Efficient Algorithms for Monotone Non-Submodular Maximization with Partition Matroid Constraint

Lan N. Nguyen , My T. Thai
Department of Computer and Information Science and Engineering
University of Florida, Gainesville, Florida 32611
lan.nguyen@ufl.edu, mythai@cise.ufl.edu

Abstract

In this work, we study the problem of monotone non-submodular maximization with partition matroid constraint. Although a generalization of this problem has been studied in literature, our work focuses on leveraging properties of partition matroid constraint to (1) propose algorithms with theoretical bound and efficient query complexity; and (2) provide better analysis on theoretical performance guarantee of some existing techniques. We further investigate those algorithms' performance in two applications: Boosting Influence Spread and Video Summarization. Experiments show our algorithms return comparative results to the state-of-the-art algorithms while taking much fewer queries.

1 Introduction

Maximizing classes of set functions, generalizing submodular functions, has emerged recently due to its wide range applications in real-world problems. Among those works, non-submodular maximization subject to cardinality constraint was studied the most extensively, including but not limited to [Bian *et al.*, 2017; Das and Kempe, 2011; Qian *et al.*, 2018; Kuhnle *et al.*, 2018].

However, cardinality constraint may not be sufficient to capture some natural requirements of various applications. For example, in many viral marketing campaigns, it is important to ensure the diversity and fairness among different ethnics and genders. These applications aim to distribute budget to feed information fairly among different groups of users while guaranteeing to maximize the influence spread in the network. Another example is data summarization. In many situations, a large data may be formed by elements of various classes. The problem, thus, aims to find a representative subset to cover the dataset's content as much as possible while imposing a constraint that the subset should contain a number of members of each class to guarantee diversity.

Motivated by those observation, we study the following problem: Given a ground set V, a non-negative monotone function $f : 2^V \rightarrow \mathbb{R}^{\geq}$; let $V_1, ..., V_k$ be a collection of disjoint subsets forming V (i.e. $V = V_1 \biguplus ... \biguplus V_k$), and $b_1, ..., b_k$ be k integers that $1 \leq b_i \leq |V_i| \; \forall i \in [k]$. The problem asks for:

$$\max_{S \subseteq V} \{f(S) : |S \cap V_i| \leq b_i \; \forall i \in [k]\} \qquad (\textsc{MaxMP})$$

MAXMP is formally represented as monotone non-submodular maximization with partition matroid constraint. This constraint is a special case of matroid constraint and generalizes cardinality constraint.

Non-submodular maximization beyond cardinality constraint has only received attention recently. The most recent works are [Chen *et al.*, 2018] and [Gatmiry and Gomez-Rodriguez, 2018], in which they studied the performance guarantee of GREEDY or RESIDUAL GREEDY (RESGREEDY) [Buchbinder *et al.*, 2014] on monotone non-submodular maximization subject to matroid constraint. However, those algorithms requires $O(nK)$ queries of f (K is a rank of a matroid), which may not be desirable in practice. Researchers [Mirzasoleiman *et al.*, 2016; Badanidiyuru and Vondrák, 2014; Kuhnle *et al.*, 2018] have sought ways to speed up the GREEDY algorithm. Unfortunately, these approaches were only for cardinality constraint; or relied upon the submodularity of f.

To our knowledge, there exists no specific work dedicating for non-submodular maximization subject to partition matroid constraint. That leaves us open questions on: (1) With partition matroid, does there exist an algorithm with a better ratio or can we improve the ratio of the existing algorithms, whose performance guarantees have been proven with a matroid constraint? (As partition matroid is a special case of matroid constraint, perhaps we can get a tighter ratio if we only considered the partition matroid.) (2) Can we leverage partition matroid properties to devise approximation algorithms with more query-efficient?

In this work, we focus on answering the above two questions. First, to quantify the non-submodularity of a function, we introduce Partition Matroid Curvature α and Partition Matroid Diminishing-Return ratio γ. These two quantities are derived from the same concept with the diminishing-return ratio [Lehmann *et al.*, 2006; Bogunovic *et al.*, 2017] and generalized curvature [Bian *et al.*, 2017; Conforti and Cornuéjols, 1984; Iyer *et al.*, 2013] but have more relaxed requirements.

Our main contribution is to introduce a novel approximation algorithm, named PROB, with approximation ratio of $(1/\gamma' - 1 + \alpha')(1 - 1/\Theta(\max_{i \in [k]} |V_i|)) + 1$ where γ' and α' are non-trivial and obtainable bounds of γ and α. PROB's novelty lies in a random process of selecting a new element, in which the algorithm introduces a new probability distribution among non-selected elements. That probability distribution is a key for PROB to obtain its ratio. Furthermore, by utilizing a sampling technique to reduce searching space, we propose FASTPROB, an algorithm improving from PROB with efficient query complexity of $O(n \ln^2 \sum_{i\in[k]} b_i)$.

Furthermore, we re-investigate theoretical performance guarantees of two existing techniques, GREEDY and THRESHOLD GREEDY (THRGREEDY). We proved that: with partition matroid constraint, GREEDY can obtain a ratio of $\min\big(\alpha/(1-(1-\alpha\gamma/\sum_{i\in[k]} b_i)^{\min_{i\in[k]} b_i}), (1+\gamma\alpha)/\gamma\big)$, which - in comparing with existing work of [Friedrich *et al.*, 2019] in matroid constraint - has its own advantage in some certain range of non-submodular quantification parameters.

Finally, we investigate our algorithms' performance on two applications of MAXMP: Boosting Influence Spread and Video Summarization. We provide bounds on the objective functions' partition matroid curvature and diminishing ratio to have a better insight on theoretical guarantees of our algorithms. Experimental results show our algorithms return comparable solutions to the state-of-the-art techniques while totally outperform them in the number of queries.

2 Related Work

2.1 Quantifying Non-Submodularity

To bound how close a function to submodularity, three most popular quantities in literature are: (1) weakly submodular ratio; (2) diminishing return ratio; and (3) generalized curvature. Weakly submodular ratio, denoted as γ_s, was first introduced by [Das and Kempe, 2011] and further used by [Elenberg *et al.*, 2017; Qian *et al.*, 2015; Chen *et al.*, 2018]. γ_s is defined as the maximum value in range $[0,1]$ such that $f(S\cup T)-f(S) \le \frac{1}{\gamma_s}\sum_{e\in T\setminus S}(f(S\cup\{e\})-f(S))$ for all $S,T \subseteq V$. Diminishing-return (DR) ratio γ_d [Bogunovic *et al.*, 2018; Lehmann *et al.*, 2006; Qian *et al.*, 2018; Kuhnle *et al.*, 2018] is defined as the largest value in range $[0,1]$ that guarantees $f(T\cup\{e\})-f(T) \le \frac{1}{\gamma_d}(f(S\cup\{e\})-f(S))$ for all $S\subseteq T\subseteq V$ and $e\notin T$. γ_d was proven to be at most the value of γ_s [Kuhnle *et al.*, 2018]. General curvature α_c [Bian *et al.*, 2017; Conforti and Cornuéjols, 1984; Iyer *et al.*, 2013], on another hand, is the smallest number in $[0,1]$ that $f(T\cup\{e\})-f(T) \ge (1-\alpha_c)(f(S\cup\{e\})-f(S))$.

In this work, we adapt DR-ratio and curvature but with more relaxed requirements. To be specific, instead of requiring those quantities to be applicable for all sets, we narrow down the collection of subsets $S\subseteq T$ that need to satisfy those properties to $|(T\setminus S)\cap V_i| \le b_i$ for all $i\in[k]$. If considering size constraint, this relaxation is corresponding to the definition of Greedy DR-ratio and Greedy Curvature [Bian *et al.*, 2017; Kuhnle *et al.*, 2018]. Not only this relaxation is sufficient to bound our approximation ratios; but it also helps us obtain meaningful bounds of those quantities in the MAXMP's applications in our experiments.

2.2 Beyond Cardinality Constraint

Non-submodular maximization beyond cardinality constraint has received attention recently. [Chen *et al.*, 2018] was the first one who brought up the concept of non-submodular maximization subject to matroid constraint. In this work, the authors proved that RESGREEDY can obtain the ratio of $(1+\frac{1}{\gamma_s})^2$. [Gatmiry and Gomez-Rodriguez, 2018] then proved GREEDY is able to obtain a ratio of $\frac{\sqrt{\gamma_s K}+1}{0.4\gamma_s^2}$ and $1+1/\gamma_d$.

In *submodular* maximization, the study beyond cardinality constraint is too extensive to give a comprehensive overview. Due to space limit, we only go over representative works; and refer readers to a comprehensive discussion in [Calinescu *et al.*, 2011; Buchbinder *et al.*, 2019; Friedrich *et al.*, 2019].

For decades, GREEDY- with ratio of 2 [Cornnejols *et al.*, 1977] - has been considered as the best algorithm for monotone submodular maximization subject to matroid constraint. This was up until [Calinescu *et al.*, 2011] introduced a concept of multilinear extension of submodular functions to devise a $1/(1-1/e)$ algorithm. However, their expensive complexity remains a significant bottleneck to make the algorithm be applicable; and improving it is still an intriguing open question for future research. The newest breakthrough is of [Buchbinder *et al.*, 2019], who devised an algorithm, namely SPLITGROW, with a ratio of $1/0.5008$ and $\tilde{O}(nK^2+KT)$ complexity - where T is the complexity to find a maximum weight perfect matching in a bipartite graph with $2K$ vertices.

The most recent work on partition matroid, to our knowledge, is of [Friedrich *et al.*, 2019], in which the authors proved GREEDY is able to obtain a ratio of $\alpha_c/\big(1-\exp\big[-\alpha_c\frac{\min_{i\in[k]} b_i}{\sum_{i\in[k]} b_i}\big]\big)$. We generalizes this work to non-submodular objective function by providing analysis that GREEDY can obtain a ratio of $\min\big(\alpha/(1-(1-\alpha\gamma/\sum_{i\in[k]} b_i)^{\min_{i\in[k]} b_i}), 1/\gamma+\alpha\big)$. If only considering submodular objective function, our ratio has an advantage that it is bounded by $1/\gamma+\alpha$. Therefore, its ratio does not degrade when the input is formed by many partitions.

We also provide approximation ratio of THRGREEDY. THRGREEDY has been studied by [Kuhnle *et al.*, 2018] for the problem of monotone non-submodular maximization with cardinality constraint. Since partition matroid generalizes cardinality constraint, our analysis techniques are totally different to [Kuhnle *et al.*, 2018]. If projecting our ratio to cardinality constraint, our ratio is better than the one of [Kuhnle *et al.*, 2018], which is $1/(1-e^{-\gamma_d\gamma_s(1-\epsilon)}-\epsilon)$. The keys help us obtain a better ratio are (1) γ_s is not necessary to bound inequality between obtained solutions and the optimal solution; and (2) we utilizes the general curvature to tighten the inequality equations, thus our ratio becomes better if the curvature moves away from the trivial value 1.

3 Definitions and Notations

Given a set function f, a set S and $e\notin S$, denote $\Delta_e f(S) := f(S\cup\{e\})-f(S)$.

Given the partition matroid constraint of MAXMP, including $V=V_1\uplus\ldots\uplus V_k$ and $b_1,\ldots,b_k$, denote $b=\sum_{i\in[k]} b_i$; $n=|V|$; $n_i=|V_i|\ \forall i\in[k]$. Let $\bar{n}=\max_{i\in[k]} n_i$ and $\hat{b}=\min_{i\in[k]} b_i$. A set $S\subseteq V$ is called a *maximal* set to the constraint iff $|S\cap V_i|=b_i\ \forall i\in[k]$.

Definition 1. *Given an instance of* MAXMP*, including*

Algorithm 1 PROB

Input $V = V_1 \uplus ... \uplus V_k; b_1, ..., b_k; f, \gamma', \alpha'$

```
1: I = [k]; S_0 = ∅; t = 0
2: while I ≠ ∅ do
3:     for each i ∈ I do
4:         a = ⌈ (|V_i \ S_t| + 1) / (1 − γ'(1 − α')) ⌉ − 1
5:         e_t = select from V_i \ S_t with probability
               (Δ_{e_t} f(S_t))^a / Σ_{u ∈ V_i \ S_t} (Δ_u f(S_t))^a
6:         S_{t+1} = S_t ∪ {e_t}; t = t + 1
7:         if |S_t ∩ V_i| ≥ b_i then I = I \ {i}
```

Return S_b

*$V = V_1 \uplus ... \uplus V_k; \{b_1, ..., b_k\}$ and f. The **Partition Matroid (PM) Diminishing Return** ratio γ of the objective function f is defined as the maximum value in $[0, 1]$ that guarantees $\Delta_e f(T) \leq \frac{1}{\gamma}\Delta_e f(S)$ for any $S \subseteq T$ that $|(T \setminus S) \cap V_i| \leq b_i \ \forall i \in [k]$ and $e \in V \setminus T$.*

Definition 2. *Given an instance of* MAXMP*, including $V = V_1 \uplus ... \uplus V_k; \{b_1, ..., b_k\}$ and f. The **Partition Matroid (PM) Curvature** α of the objective function f is defined as the minimum value in $[0, 1]$ that guarantees $\Delta_e f(T) \geq (1-\alpha)\Delta_e f(S)$ for any $S \subseteq T$ that $|(T \setminus S) \cap V_i| \leq b_i \ \forall i \in [k]$ and $e \in V \setminus T$.*

It is unknown in the literature on how hard it is to obtain exact values of quantities quantifying non-submodularity. γ and α are not exceptional either. Fortunately, for some applications, we can obtain non-trivial bounds of γ and α, which can help assess approximation ratios of our algorithms. We denote γ' as a lower bound of γ, e.g. $\gamma \geq \gamma' \geq 0$; and α' as a upper bound of α, e.g. $\alpha \leq \alpha' \leq 1$.

W.l.o.g, we assume the objective function f is *normalized*, i.e. $f(\emptyset) = 0$, and $b_i \leq n_i$ for all $i \in [k]$. In our algorithms' analysis, we denote S^* as an optimal solution, i.e $f(S^*) = \max_{S:|S \cap V_i| \leq b_i} f(S)$.

4 PROB and FASTPROB Algorithms

4.1 PROB Algorithm

PROB is a randomized algorithm with approximation ratio of $(1/\gamma' - 1 + \alpha')\big(1 - 1/O(\bar{n})\big) + 1$. Pseudocode of PROB is presented by Alg. 1. In general, PROB works in rounds, and at each round, one member of a group V_i is added to the obtained solution S if $|S \cap V_i| < b_i$. The key for PROB to obtain efficient performance guarantee lies in a random process, which introduces a probability distribution, defined locally for each group, to select a new element of each group to add into the obtained solution (line 6 Alg. 1). This random process allows us to construct a sequence of maximal sets in order to form a recursive relationship among changes on the f's values of the obtained solutions, which is critical to bound PROB's approximation ratio.

Theorem 1. PROB *obtains a $\left(\frac{1}{\gamma'} + \alpha' - 1\right)\left(1 - \frac{1}{\bar{n}+2}\right) +$ 1-approximation solution and has query complexity of $O(\sum_{i \in [k]} n_i b_i)$.*

Proof. Denote $\beta = \left(\frac{1}{\gamma'} + \alpha' - 1\right)\left(1 - \frac{1}{\bar{n}+2}\right)$ and $S_1, ..., S_b$ as a sequence of obtained solution by PROB. We prove the approximation ratio of PROB by constructing a sequence of maximal sets $S_0^*, ..., S_b^*$ that satisfies the following properties: (1) $S_0^* = S^*$ and $S_b^* = S_b$; (2) $S_t \subset S_t^*$ for all $t = 0, ..., b-1$ and $S_b = S_b^*$; (3) $f(S_t^*) - f(S_{t+1}^*) \leq \beta \, \mathrm{E}\Big[f(S_{t+1}) - f(S_t)\Big]$ for $t = 0 \to b-1$. Then, we have:

$$f(S^*) = \sum_{t=0}^{b-1}\Big(f(S_t^*) - f(S_{t+1}^*)\Big) + f(S_b^*)$$

$$\leq \beta \sum_{t=0}^{b-1} \mathrm{E}[f(S_{t+1}) - f(S_t)] + f(S_b) \leq (\beta+1)\mathrm{E}[f(S_b)]$$

To construct the sequence, starting with $S_0^* = S^*$, for each $t = 1, ..., b-1$, S_{t+1}^* is formed from S_t^*, S_t and e_t as follows: Let i be the index being considered at the **for** loop (line 3 Alg. 1); and e_t will be added into S_t. Since $S_t \subset S_t^*$ and $|S_t \cap V_i| < b_i$, $(S_t^* \setminus S_t) \cap V_i \neq \emptyset$. Let e' be any arbitrary element in $(S_t^* \setminus S_t) \cap V_i$. S_{t+1}^* is set as follows:

- If $e_t \in (S_t^* \setminus S_t) \cap V_i$, $S_{t+1}^* := S_t^*$.
- Otherwise, let $S_{t+1}^* := S_t^* \setminus \{e'\} \cup \{e_t\}$.

Denote $\rho_e = \Delta_e f(S_t)$ and $\mathrm{Pr}_e = \frac{\rho_e^a}{\sum_{v \in V_i \setminus S_t} \rho_v^a}$ (i.e. Pr_e is probability e is selected). We have:

$$\mathrm{E}\Big[f(S_t^*) - f(S_{t+1}^*)\Big] \tag{1}$$

$$= \sum_{u \in V_i \setminus S_t^*} \Big[f(S_t^*) - f(S_t^* \setminus \{e'\} \cup \{u\})\Big] \times \mathrm{Pr}_u \tag{2}$$

$$= \sum_{u \in V_i \setminus S_t^*} \Big[\Delta_{e'} f(S_t^* \setminus \{e'\}) - \Delta_u f(S_t^* \setminus \{e'\})\Big] \times \mathrm{Pr}_u \tag{3}$$

$$\leq \sum_{u \in V_i \setminus S_t^*} \Big[\frac{1}{\gamma}\rho_{e'} - (1-\alpha)\rho_u\Big] \times \mathrm{Pr}_u \tag{4}$$

$$= \frac{1}{\gamma} \sum_{u \in V_i \setminus S_t^*} \frac{\rho_{e'}\rho_u^a}{\sum_{v \in V_i \setminus S_t} \rho_v^a} - (1-\alpha) \sum_{u \in V_i \setminus S_t^*} \rho_u \mathrm{Pr}_u \tag{5}$$

$$\leq \frac{1}{\gamma(a+1)} \sum_{u \in V_i \setminus S_t^*} \frac{\rho_{e'}^{a+1} + a\rho_u^{a+1}}{\sum_{v \in V_i \setminus S_t} \rho_v^a} \tag{6}$$

$$- (1-\alpha) \sum_{u \in V_i \setminus S_t^*} \rho_u \mathrm{Pr}_u \tag{7}$$

$$= \frac{|V_i \setminus S_t^*|}{\gamma(a+1)} \rho_{e'} \mathrm{Pr}_{e'} + \Big(\frac{1}{\gamma}\frac{a}{a+1} + \alpha - 1\Big) \sum_{u \in V_i \setminus S_t^*} \rho_u \mathrm{Pr}_u \tag{8}$$

where Equ. (4) is from properties of γ and α; while Equ. (7) is from AM-GM inequality.

Replacing $a = \lceil \frac{|V_i \setminus S_t| + 1}{1 - \gamma'(1-\alpha')} \rceil - 1$, we have

$$\frac{|V_i \setminus S_t^*|}{\gamma(a+1)} \leq \frac{|V_i \setminus S_t|}{\gamma(|V_i \setminus S_t| + 1)/(1 - \gamma'(1-\alpha'))} \tag{9}$$

$$\leq \Big(\frac{1}{\gamma'} + \alpha' - 1\Big)\Big(1 - \frac{1}{\bar{n}+1}\Big) \tag{10}$$

$$\frac{1}{\gamma}\frac{a}{a+1}+\alpha-1 \leq \frac{1}{\gamma}\Big(1-\frac{1-\gamma'(1-\alpha')}{|V_i\setminus S_t|+2}\Big)+\alpha-1 \tag{11}$$

$$\leq \Big(\frac{1}{\gamma'}+\alpha'-1\Big)\Big(1-\frac{1}{\bar{n}+2}\Big) \tag{12}$$

Therefore, combining Equ. (10), (12) to (8), we have:

$$(8) \leq \Big(\frac{1}{\gamma'}+\alpha'-1\Big)\Big(1-\frac{1}{\bar{n}+2}\Big)\mathrm{E}\Big[f(S_{t+1})-f(S_t)\Big]$$

The query complexity of PROB can be trivially inferred from the algorithm's pseudocode. □

Due to differences in definition of the quantities quantifying non-submodularity and how algorithms' ratios depend on them, it is no clear way to compare their ratios. For example, RESGREEDY obtains $(1+\frac{1}{\gamma_s})^2$-ratio [Chen *et al.*, 2018]. Although $\gamma_s \geq \gamma \geq \gamma'$, it is unclear how this ratio is compared with PROB's ratio. However, PROB has a better query complexity than RESGREEDY ($O(nb)$).

When f is submodular ($\gamma = 1$), PROB can obtain a ratio of $1+\alpha'(1-\frac{1}{\bar{n}+2})$. Although PROB's ratio is still not comparable to the best ratio $(1-1/e)$ of [Calinescu *et al.*, 2011], their expensive complexity $O(n^8)$ remains a significant bottleneck to make their algorithm be applicable in practice. In compare with the most recent work [Buchbinder *et al.*, 2019], PROB can reach a better ratio than SPLITGROW ($\frac{1}{0.5008}$) with appropriate values of α' and $\bar{n}$; and PROB has much better query complexity than SPLITGROW ($O(nb^2)$).

4.2 FASTPROB Algorithm

PROB's query complexity can be improved by observing that the proof of Theorem 2 can non-trivially go through if e_t is selected from a set that overlaps with $(S_t^* \setminus S_t) \cap V_i$ for all $t = 1, ..., b$. This always works in Alg. 1 since e_t is selected from $V_i \setminus S_t$. Therefore, we can use sampling to reduce the space of selecting e_t as in Alg. 2.

We call Alg. 2 FASTPROB. The condition which helps FASTPROB has the same ratio as PROB with probability at least $1-\delta$, is guaranteed as stated in the following lemma.

Lemma 1. *$(S_t^* \setminus S_t) \cap R_t \neq \emptyset$ for all $t = 0, ..., b-1$ with probability at least $1-\delta$*

Proof. We prove for each $t = 0, ..., b-1$, $\Pr\Big[(S_t^*\setminus S_t)\cap R_t = \emptyset\Big] \leq \frac{\delta}{b}$. Then using union bound, $(S_t^* \setminus S_t) \cap R_t \neq \emptyset$ for all $t = 0, ..., b-1$ with probability at least $1-\delta$. This probability is trivial if $R_t = V_i \setminus S_t$. If $|R_t| = \frac{n_i-|S_t\cap V_i|}{b_i-|S_t\cap V_i|}\ln\frac{b}{\delta}$, since $S_t \subseteq S_t^*$, $|(S_t^* \setminus S_t) \cap V_i| = b_i - |S_t \cap V_i|$. We have:

$$\Pr\Big[(S_t^*\setminus S_t)\cap R_t=\emptyset\Big] \leq \Big(\frac{|V_i\setminus S_t^*|}{|V_i\setminus S_t|}\Big)^{|R_t|}$$

$$= \Big(1-\frac{|(S_t^*\setminus S_t)\cap V_i|}{|V_i\setminus S_t|}\Big)^{|R_t|} \leq e^{-|R_t|\frac{b_i-|S_t\cap V_i|}{n_i-|S_t\cap V_i|}} \leq \frac{\delta}{b}$$

which completes the proof. □

Algorithm 2 FASTPROB

Input $V = V_1 \uplus ... \uplus V_k; f, \gamma', \alpha'; b_1, ..., b_k; \delta \in [0,1]$

$I = [k]; S_0 = \emptyset; t = 0$
while $I \neq \emptyset$ **do**
  for each $i \in I$ **do**
    $R_t =$ pick $\min\big(\frac{n_i-|S_t\cap V_i|}{b_i-|S_t\cap V_i|}\ln\frac{b}{\delta}, |V_i\setminus S_t|\big)$ random elements from $V_i \setminus S_t$
    $a = \lceil\frac{|R_t|+1}{1-\gamma'(1-\alpha')}\rceil - 1$
    $e_t =$ select from R_t with probability $\frac{(\Delta_{e_t}f(S_t))^a}{\sum_{u\in R_t}(\Delta_u f(S_t))^a}$
    $S_{t+1} = S_t \cup \{e_t\}; t = t+1$
    if $|S_t \cap V_i| \geq b_i$ **then** $I = I \setminus \{i\}$

Return S_b

Theorem 2. FASTPROB *obtains a $\Big(\frac{1}{\gamma'}+\alpha'-1\Big)\Big(1-\frac{1}{\bar{n}+2}\Big)+1$-approximation solution with probability at least $1-\delta$ and has query complexity of $O(n\ln b\ln\frac{b}{\delta})$.*

Proof. The method to prove FASTPROB's approximation ratio is similar to the proof of PROB. Due to space limit and for the sake of completeness, we provide the proof of FASTPROB's ratio in Appendix [Nguyen and Thai, 2022].

In term of query complexity, it is trivial that the number of queries of FASTPROB is $\sum_{t=0}^{b-1}|R_t|$. We have:

$$\sum_{t=0}^{b-1}|R_t| \leq \sum_{i\in[k]}\sum_{j=0}^{b_i-1}\frac{n_i-j}{b_i-j}\ln\frac{b}{\delta} \tag{13}$$

$$= \ln\frac{b}{\delta}\Big(\sum_{i\in[k]}b_i + (n_i-b_i)\sum_{j=0}^{b_i-1}\frac{1}{b_i-j}\Big) \tag{14}$$

$$\leq b\ln\frac{b}{\delta} + \ln\frac{b}{\delta}\sum_{i\in[k]}(n_i-b_i)\ln b_i \leq O(\ln\frac{b}{\delta}\sum_{i\in[k]}n_i\ln b_i) \tag{15}$$

$$\leq O(n\ln\frac{b}{\delta}\ln\sum_{i\in[k]}\frac{n_ib_i}{n}) \leq O(n\ln b\ln\frac{b}{\delta}) \tag{16}$$

where Equ. (16) is from the fact that $\log x$ is a concave function, so $\sum_i \alpha_i \log x_i \leq \log \alpha_i x_i$ if $\sum_i \alpha_i = 1$; and $\sum_{i\in[k]}\frac{n_ib_i}{n} \leq \sum_{i\in[k]}\frac{n_i}{n}\sum_{i\in[k]}b_i = b$. □

5 GREEDY-like Algorithms

We re-study the theoretical performance guarantee of two algorithms, GREEDY and THRGREEDY. Our analysis provides better ratios of GREEDY than existing works on matroid constraint [Gatmiry and Gomez-Rodriguez, 2018] or submodular objective function [Friedrich *et al.*, 2019].

In general, GREEDY works in round and at each round, an element of maximal marginal gain, whose addition does not violate partition matroid constraint, is added to the obtained solution. The algorithm terminates when the obtained solution

Algorithm 3 GREEDY

Input $V = V_1 \uplus ... \uplus V_k; f; b_1, ..., b_k$

$I = [k]; S_0 = \emptyset; t = 0$
while $I \neq \emptyset$ **do**
  $e, i = \text{argmax}_{e \in V_i \setminus S_t; i \in I} \Delta_e f(S_t)$
  $S_{t+1} = S_t \cup \{e\}; t = t + 1$
  if $|S_t \cap V_i| \geq b_i$ **then** $I = I \setminus \{i\}$

Return S_b

Algorithm 4 THRGREEDY

Input $V = V_1 \uplus ... \uplus V_k; f; b_1, ..., b_k; \epsilon \in [0, 1]$

$I = [k]; S_0 = \emptyset; t = 0$
$\tau = \tau_0 = \max_{e \in V} \Delta_e f(S_0)$
while $I \neq \emptyset$ and $\tau \geq \frac{\epsilon(1-\epsilon)\tau_0}{b}$ **do**
  for each $i \in I$ and $e \in V_i \setminus S_t$ **do**
    if $\Delta_e f(S_t) \geq \tau$ **then**
      $S_{t+1} = S_t \cup \{e\}; t = t + 1$
      if $|S_t \cap V_i| \geq b_i$ **then** $I = I \setminus \{i\}$
  $\tau = \tau(1 - \epsilon)$

Return S_t

is maximal. THRGREEDY, on the other hand, works by always keeping a threshold τ, which bounds the maximum marginal gain to the objective by any non-selected elements. The algorithm runs in rounds; at each round, any element with a marginal gain at least τ will be added to the solution if it does not violate the partition matroid constraint. After each round, τ is decreased by a factor $1 - \epsilon$ in order to guarantee new elements can be added to the solution at successive rounds. The algorithm continues until the obtained solution becoming a maximal set or the threshold is below a value defined by ϵ and b. GREEDY's pseudocode is presented by Alg. 3 and THRGREEDY's is Alg. 4.

Theorem 3. GREEDY *obtains a* $\min(\frac{1}{r_1^{(g)}}, \frac{1}{r_2^{(g)}})$-*approximation solution, where*

$$r_1^{(g)} = \frac{\gamma}{1 + \gamma\alpha} \qquad r_2^{(g)} = \frac{1}{\alpha}\Big[1 - \Big(1 - \frac{\alpha\gamma}{b}\Big)^{\hat{b}}\Big]$$

and has a query complexity of $O(n\, b)$.

Theorem 4. THRGREEDY *obtains a* $\min(\frac{1}{r_1^{(t)}}, \frac{1}{r_2^{(t)}})$-*approximation solution, where*

$$r_1^{(t)} = \frac{\gamma(1-\epsilon)^2}{1 + \gamma\alpha(1-\epsilon)} \quad r_2^{(t)} = \frac{1}{\alpha}\Big[1 - \Big(1 - \frac{\alpha\gamma(1-\epsilon)}{b}\Big)^{\hat{b}}\Big]$$

and has a query complexity of $O(\frac{n}{\epsilon} \ln b)$.

Due to space limit, full proofs of Theorem 3 and 4 is provided in Appendix [Nguyen and Thai, 2022].

In case of submodular objective function, $r_2^{(g)}$ of GREEDY is identical to the ratio obtained by [Friedrich *et al.*, 2019]. With cardinality constraint, $r_2^{(g)}$ matches with the ratio of [Bian *et al.*, 2017], which was also proven to be tight. However, with $\hat{b}/b \to 0$ (e.g. the input is formed by many partitions), $r_2^{(g)}$ and $r_2^{(t)}$ approach 0 and become undesirable. In this case, $r_1^{(g)}$ and $r_1^{(t)}$ should be a better bound on the performance of GREEDY and THRGREEDY.

6 Applications and Experimental Results

In this section, we consider two applications of MAXMP: Boosting Influence Spread and Video Summarization.

6.1 Applications' Formulation and Bounded Non-Submodularity Quantities

In **Boosting Influence Spread**, a social directed graph $G = (V, E)$ is given, where V represents a set of social network users; and E represents friendship between social users in V. An information will start spreading at a set $I \subset V$ of users. The problem asks for a set S of users to strengthen the influence spread in order to maximize the number of users whom the information can reach to.

Boosting Influence Spread under size constraint has been studied by [Lin *et al.*, 2017]. In their model, each edge $e = (u, v) \in E$ is associated with two weight values p_e^0, p_e^1 ($p_e^0 \leq p_e^1 \leq 1$). The probability v adopts the information from u is p_e^1 if $v \in S$; p_e^0 otherwise. In this application, $f(S)$ measures expected number of users the information can reach if S is selected. The authors has proven that f is monotone non-submodular; but did not show how close f is to submodularity. We provide the bound γ', α' of γ and α of f as in Lemma 2, and full proof is provided in [Nguyen and Thai, 2022].

Lemma 2. *Given a Boosting Influence Spread instance, let* Δ *be the maximum in-degree of the input directed graph. For any* $S \subseteq T$ *that* $|(T \setminus S) \cap V_i| \leq b_i\ \forall i \in [k]$ *and* $u \in V \setminus T$*:*

$$\min_{|E'| \leq b\Delta} \prod_{e \in E'} \frac{1 - p_e^1}{1 - p_e^0} \leq \frac{\Delta_u f(S)}{\Delta_u f(T)} \leq \max_{|E'| \leq b\Delta} \prod_{e \in E'} \frac{p_e^1}{p_e^0} \quad (17)$$

With **Video Summarization**, given a video, the problem aims to pick a few representative frames from the video which can contains as much content as possible. The video contains n frames; each frame is represented by a p-dimensional vector. Let $X \in \mathbb{R}^{n \times n}$ be the Gramian matrix of the n resulting vectors and the Gaussian kernel; i.e. X_{ij} is the value of the Gaussian kernel between the i-th and j-th vectors. The objective function is defined as $f(S) = \det(I + X_S)$, where X_S is the submatrix of X indexed by S; and I is a unit matrix.

$f(S)$ was proved to be supermodular by [Bian *et al.*, 2017], thus its curvature $\alpha = 0$. The authors also bounded the weakly submodular ratio, which is not useful in our algorithms. We bound the value of γ as in the following lemma, and full proof is provided in Appendix [Nguyen and Thai, 2022].

Lemma 3. *Given a Video Summarization instance, let* $A = I + X$ *and* $\lambda_i(M)$ *be the i-th eigenvalue of a positive definite matrix* M *in a way that* $\lambda_1(M) \geq ... \geq \lambda_{rank(M)}(M)$*. For any* $S \subseteq T$ *that* $|(T \setminus S) \cap V_i| \leq b_i\ \forall i \in [k]$ *and* $e \in V \setminus T$*:*

$$\Delta_e f(S) \geq \Delta_e f(T) \times \frac{\lambda_n(A) - 1}{\lambda_1(A) - 1} \prod_i^b \frac{1}{\lambda_i(A)} \quad (18)$$

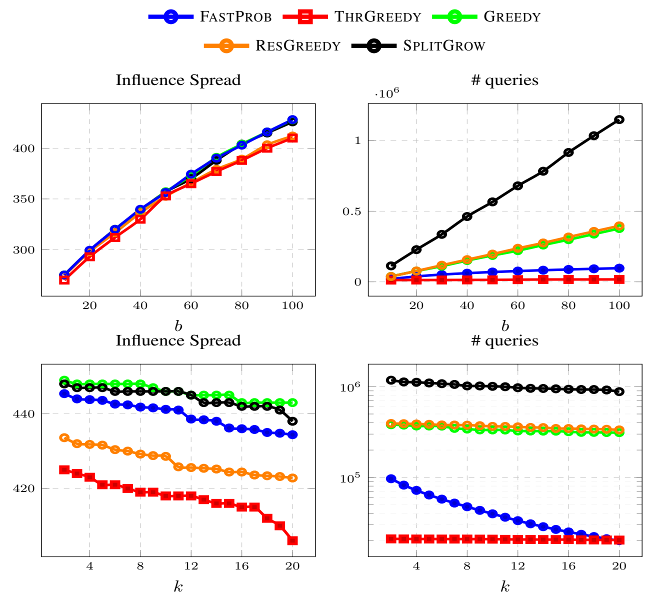

Figure 1: Performance in Boosting Influence Spread.

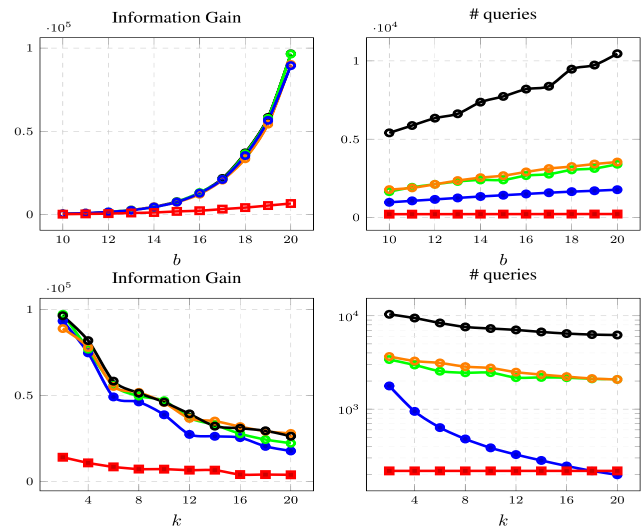

Figure 2: Performance in Video Summarization

6.2 Settings and Compared Algorithms

With Boosting Influence Spread, we use Facebook dataset from SNAP database [Leskovec and Krevl, 2014], an undirected graph with 4,039 nodes and 88,234 edges. Since it is undirected, we treat each edge as two directed edges. For each edge $e = (u, v)$, $p_e^0 = \frac{1}{d_v}$ and $p_e^1 = \frac{2}{d_v}$ where d_v is in-degree of v. Information starts spreading at a node of highest degree. Due to lack of information, a user is randomly assigned to a group V_i. The budget is distributed equally to each group, i.e. $b_1 \approx ... \approx b_k \approx \frac{b}{k}$. The objective is estimated over 100 pre-sampled graph realizations of G.

With Video Summarization, we chose a video of roughly 3.5 minutes. The video is segmented to k equal-length parts; and the algorithms will pick $\frac{b}{k}$ frames from each part.

With FastProb, we set $\delta = 0.001$, which guarantees FastProb to return solutions almost similar to Prob but be much better in the number of queries. With ThrGreedy, we set $\epsilon = 0.5$. Results were averaged over 10 repetitions.

We varied values of b and k; and compare FastProb, Greedy and ThrGreedy with ResGreedy [Chen *et al.*, 2018] and SplitGrow [Buchbinder *et al.*, 2019]. Although SplitGrow's performance is unknown if f is submodular, we used it as a heuristic to compare.

6.3 Numerical Results

Fig. 1 and 2 show experimental results of different algorithms on Boosting Influence Spread and Video Summarization. With experiments that we varied values of b, we fixed $k = 2$. With the one that k is varied, we fixed $b = 100$ in Boosting Influence Spread and $b = 20$ in Video Summarization.

In these experiments, FastProb, Greedy and SplitGrow performed approximately equal in term of solution quality while ThrGreedy was always the worst one. Especially, in Video Summarization, the supermodular objective function made the marginal gain of non-included elements increase with larger obtained solutions. Therefore, ThrGreedy easily reached a maximal solution just by one or two iterations of decreasing threshold. That explained why ThrGreedy took very few number of queries but has undesirable returned solution quality. In term of the number of queries, FastProb outperformed Greedy, ResGreedy and SplitGrow.

FastProb closed the gap or even surpassed ThrGreedy to become the best algorithm in the number of queries in the experiments with fixed b and varied k. In these experiments, we can see that the number of queries of all algorithms, except FastProb, almost did not change or just slightly decreased with larger k. FastProb's numbers, on the other hand, decreased significantly as k increased. This phenomenon is also reflected on the theoretical bound of FastProb's complexity. In Equ. (15), FastProb's complexity is bounded by $O(\ln \frac{b}{\delta} \sum_{i \in [k]} n_i \ln b_i)$. With n_is are roughly equal (the same with b_is), FastProb's complexity becomes $O(n \ln \frac{b}{k} \ln \frac{b}{\delta})$, which decreases w.r.t k.

7 Discussion

We proposed Prob and later FastProb to solve monotone non-submodular maximization with partition matroid constraint. The experimental results demonstrated that FastProb can perform closely to the best algorithms in solution quality, and outperform other algorithms (except ThrGreedy- the worst in solution quality) in the number of queries. Although there is no superior algorithm in general, FastProb should be considered as the best algorithm in scenarios that scalability issues are concerned, e.g. algorithms with fast runtime and relatively high solution quality.

There is still an open question on what is the best algorithm in approximation ratio? Prob's ratio depends on γ', α' - which can be undesirable in some settings of our experiments. However, how hard to obtain exact value of γ, α or other non-submodular quantities is unknown. And computing those quantities by enumerating all possible S, T that $T \setminus S$ satisfies partition matroid is too expensive. Therefore, the differences between Greedy, ThrGreedy, ResGreedy and Prob's ratio are still remained open.

Acknowledgements

This work was supported in part by the National Science Foundation (NSF) grants IIS-1908594 and IIS-1939725. We'd like to thank the anonymous reviewers for their helpful feedback.

References

[Badanidiyuru and Vondrák, 2014] Ashwinkumar Badanidiyuru and Jan Vondrák. Fast algorithms for maximizing submodular functions. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 1497–1514. SIAM, 2014.

[Bian *et al.*, 2017] Andrew An Bian, Joachim M Buhmann, Andreas Krause, and Sebastian Tschiatschek. Guarantees for greedy maximization of non-submodular functions with applications. In *International conference on machine learning*, pages 498–507. PMLR, 2017.

[Bogunovic *et al.*, 2017] Ilija Bogunovic, Slobodan Mitrović, Jonathan Scarlett, and Volkan Cevher. Robust submodular maximization: A non-uniform partitioning approach. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017.

[Bogunovic *et al.*, 2018] Ilija Bogunovic, Junyao Zhao, and Volkan Cevher. Robust maximization of non-submodular objectives. In *International Conference on Artificial Intelligence and Statistics*, pages 890–899, 2018.

[Buchbinder *et al.*, 2014] Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. Submodular maximization with cardinality constraints. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 1433–1452. SIAM, 2014.

[Buchbinder *et al.*, 2019] Niv Buchbinder, Moran Feldman, and Mohit Garg. Deterministic $(1/2+\varepsilon)$-approximation for submodular maximization over a matroid. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 241–254. SIAM, 2019.

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

[Chen *et al.*, 2018] Lin Chen, Moran Feldman, and Amin Karbasi. Weakly submodular maximization beyond cardinality constraints: Does randomization help greedy? In *International Conference on Machine Learning*, 2018.

[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, 1984.

[Cornnejols *et al.*, 1977] G Cornnejols, M Fisher, and G Nemhauser. Location of bank accounts of optimize float: An analytic study of exact and approximate algorithm. *Management Science*, 23:789–810, 1977.

[Das and Kempe, 2011] Abhimanyu Das and David Kempe. Submodular meets spectral: greedy algorithms for subset selection, sparse approximation and dictionary selection. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 1057–1064, 2011.

[Elenberg *et al.*, 2017] Ethan Elenberg, Alexandros G Dimakis, Moran Feldman, and Amin Karbasi. Streaming weak submodularity: Interpreting neural networks on the fly. In *Advances in Neural Information Processing Systems*, pages 4044–4054, 2017.

[Friedrich *et al.*, 2019] Tobias Friedrich, Andreas Göbel, Frank Neumann, Francesco Quinzan, and Ralf Rothenberger. Greedy maximization of functions with bounded curvature under partition matroid constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2272–2279, 2019.

[Gatmiry and Gomez-Rodriguez, 2018] Khashayar Gatmiry and Manuel Gomez-Rodriguez. Non-submodular function maximization subject to a matroid constraint, with applications. *arXiv preprint arXiv:1811.07863*, 2018.

[Iyer *et al.*, 2013] Rishabh K Iyer, Stefanie Jegelka, and Jeff A Bilmes. Curvature and optimal algorithms for learning and minimizing submodular functions. *Advances in Neural Information Processing Systems*, 2013.

[Kuhnle *et al.*, 2018] Alan Kuhnle, J David Smith, Victoria Crawford, and My Thai. Fast maximization of non-submodular, monotonic functions on the integer lattice. In *International Conference on Machine Learning*, pages 2786–2795, 2018.

[Lehmann *et al.*, 2006] Benny Lehmann, Daniel Lehmann, and Noam Nisan. Combinatorial auctions with decreasing marginal utilities. *Games and Economic Behavior*, 55(2):270–296, 2006.

[Leskovec and Krevl, 2014] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

[Lin *et al.*, 2017] Yishi Lin, Wei Chen, and John CS Lui. Boosting information spread: An algorithmic approach. In *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, pages 883–894. IEEE, 2017.

[Mirzasoleiman *et al.*, 2016] Baharan Mirzasoleiman, Morteza Zadimoghaddam, and Amin Karbasi. Fast distributed submodular cover: Public-private data summarization. In *Advances in Neural Information Processing Systems*, pages 3594–3602, 2016.

[Nguyen and Thai, 2022] Lan Nguyen and My Thai. Efficient algorithms for monotone non-submodular maximization with partition matroid constraint. *arXiv preprint arXiv:2204.13832*, 2022.

[Qian *et al.*, 2015] Chao Qian, Yang Yu, and Zhi-Hua Zhou. Subset selection by pareto optimization. In *Advances in Neural Information Processing Systems*, 2015.

[Qian *et al.*, 2018] Chao Qian, Yibo Zhang, Ke Tang, and Xin Yao. On multiset selection with size constraints. In *AAAI*, pages 1395–1402, 2018.