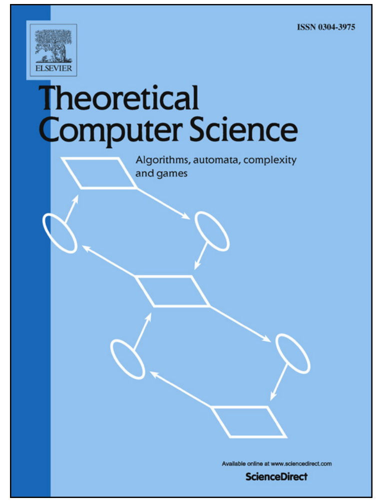
Journal Pre-proof



Novel algorithms for maximum DS decomposition

Shengminjie Chen, Wenguo Yang, Suixiang Gao and Rong Jin

PII: S0304-3975(20)30773-8
DOI: https://doi.org/10.1016/j.tcs.2020.12.041
Reference: TCS 12784

To appear in: *Theoretical Computer Science*

Received date: 15 November 2020
Revised date: 4 December 2020
Accepted date: 14 December 2020

Highlights

- Authors have studied the problem of DS decomposition which is an important set function optimization problem. We have tackled the problem of how to effectively and efficiently solve the problem of DS decomposition as it is a vital part to solve for any set function. The proposed algorithm is a Deterministic Parameter Conditioned Greedy Algorithm and Random Parameter Conditioned Greedy Algorithm. We have focused on the difference with parameter and combined it with non-negative condition and have obtained some novel approximation as well. All the proofs and detailed and also the algorithm seems to be flexible as the Conditioned Algorithm is the general framework, different users can choose the parameters that fit their problem to get a better approximation.

Novel algorithms for maximum DS decomposition

Shengminjie Chen[a], Wenguo Yang[a,*], Suixiang Gao[a], Rong Jin[b]

[a]*School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing 100049, China*
[b]*Department of Computer Science, University of Texas at Dallas, Richardson, TX, 75080, USA*

Abstract

DS decomposition plays an important role in set function optimization problem, because there is DS decomposition for any set function. How to design an efficient and effective algorithm to solve maximizing DS decomposition is a heated problem. In this work, we propose a framework called Parameter Conditioned Greedy Algorithm which has a deterministic version and two random versions. In more detail, this framework uses the difference with parameter decomposition function and combines non-negative condition. Besides, if we set the different parameters, the framework can return solution with different approximation ratio. Also, we choose two special case to show our deterministic algorithm gets $f(S_k)-(e^{-1}-c_g)g(S_k) \geq (1-e^{-1})[f(OPT)-g(OPT)]$ and $f(S_k)-(1-c_g)g(S_k) \geq (1-e^{-1})f(OPT)-g(OPT)$ respectively for cardinality constrained problem, where c_g is the curvature of monotone submodular set function. To speed the deterministic algorithm, we introduce a random sample set which can represent optimal solution set as soon as possible. Importantly, it also can get the same approximation ratio as deterministic algorithm under expectation. Further, for maximization DS decomposition without constraint, our another random algorithm gets $E[f(S_k)-(e^{-1}-c_g)g(S_k)] \geq (1-e^{-1})[f(OPT)-g(OPT)]$ and $E[f(S_k)-(1-c_g)g(S_k)] \geq (1-e^{-1})f(OPT)-g(OPT)$ respectively. Because the Parameter Conditioned Algorithm is the general framework, different users can choose the parameters that fit their problem to get a better approximation.

*Corresponding Author
Email address: yangwg@ucas.ac.cn (Wenguo Yang)

Keywords: Non-Submodularity, DS Decomposition, Parameter Conditioned Greedy

1. Introduction

In discrete optimization, submodular maximization for set functions is a hot topic, which attracts many scholars. In the past few decades, many important results for maximization of submodular set function have been proposed. Based on monotonicity and $f(\emptyset) = 0$, the pioneering work was done in 1978. Nemhauser et al. [1] firstly proposed a greedy algorithm which can get $1 - e^{-1}$ approximation ratio under cardinality constraint by adding a maximal marginal gains element to current solution in each iteration. In the same year, Fisher et al. [2] proved that the greedy strategy can get $(1 + p)^{-1}$ approximation with p matroid constraint. But, for the application, the quality of solution returned by greedy strategy is much better than the above approximation ratio. By constructing a metric called curvature $c \in [0, 1]$, Conforti et al. (1984) [3] proved that the greedy algorithm can get more tight approximation ratio $\frac{1-e^{-c}}{c}$ and $\frac{1}{1+c}$. That is why greedy strategy is excellent for monotone submodular maximization. These groundbreaking works have inspired a great deal of submodular optimization.

Beyond the monotonicity, the above algorithms are not good for submodular maximization. Until 2011, Feldman et al. [4] proposed a novel greedy which has e^{-1} approximation for non-monotone submodular maximization under matroids constraint. Further, by using double greedy, Buchbinder et al. (2014) [5] proved that non-monotone submodular maximization under cardinality constrained also has e^{-1} approximation polynomial algorithm. Besides, for maximization without constraints, Buchbinder et.al (2012) [6] also constructed an algorithm which can get 1/2 approximation.

Because these are some efficient algorithms, submodular optimization is widely utilized in data mining, machine learning, economics and operation research such as influence maximization (Kempe et al., (2003) [7]), active learning (Golovin et al., (2011) [8]), document summarization (Lin et al., (2011) [9]), image segmentation (Jegelka et al., (2011) [10]).

In applications, unfortunately, a large number of objective functions are not submodular. Thus, how to optimize a general set function is the most important problem and has puzzled many researchers.

In this paper, we study the non-submodular maximization problem based on DS decomposition about set functions, which is the difference between two monotone submodular functions. To solve this problem, we proposed Parameter Conditioned Greedy Algorithm by using the difference with parameter decomposition function and combining non-negative condition. There are general frameworks with a deterministic version and two random version and users can choose rational parameters according the property of problem.

1.1. Related Work

In this subsection, we introduce some previous works about non-submodular maximization problem which are related to our work directly.

On the one hand, some researchers proposed lots of definition about approximation of submodularity. Firstly, Krause et al. (2008) [11] constructed a metric called ϵ-Diminishing returns to evaluate what is the level of violation of marginal gains decreasing. On this basis, authors proved the standard greedy algorithm can get $f(X) \geq (1-e^{-1})(OPT - k\epsilon)$ approximation under cardinality constraint. Das and Kempe (2011) [12] introduced submodular ratio to measure violation about submodularity. Besides, they proved the standard greedy strategy can get $f(X) \geq (1-e^{-\gamma})OPT$ approximation under cardinality constraint. Horel and Singer (2016) [13] proposed ϵ-approximation submodularity to limit non-submodular function. According this metric, authors proved that the standard greedy algorithm can return $f(X) \geq \frac{1}{1+\frac{4k\epsilon}{(1-\epsilon)^2}}(1-e^{-1}(\frac{1-\epsilon}{1+\epsilon})^{2k}) \cdot OPT$ approximation under cardinality constraint. It is no surprise that computing these metrics is also a hard problem. It makes these results only theoretical meaning, but lack of practical application.

On the other hand, Lu et al. (2015) [14] proposed Sandwich Approach which chooses the best one solution of a submodular upper bound, a submodular lower bound and original problem. Because this approach can get a well-calculating parametric approximation, some researchers also applied Sandwich Approach to solve their application problems (Yang et al. (2020) [15], Yang et al. (2020) [16], Zhu et al. (2019) [17], Wang et al. (2017) [18]).

In addition, Iyer and Bilmes (2012) [19] proved that any set function can be expressed as the difference between two submodular set functions called DS decomposition. Specially, the two submodular set function are monotone and non-decreasing. What's more, authors proposed SubSup, SupSub, ModMod algorithms to solve minimum this problem.

Lemma 1 (Iyer and Bilmes 2012)**.** *Any set function $h : 2^{\Omega} \to R$ can decompose the difference of two monotone non-decreasing submodular set functions f and g, i.e. $h = f - g$.*

This excellent result has been applied in many scenarios (Han et al. (2018) [20], Maehara et al. (2015) [21] and Yu et al. (2016) [22]). Because of the correspondence between submodularity and supermodularity, Li et al. (2020) [23] proved a variation of DS decomposition, i.e. any set function can be expressed as the difference of two monotone non-decreasing supermodular functions. Similarly, they also proposed greedy strategies like Iyer and Bilmes called ModMod and SupMod.

From the perspective of application, Bai and Bilmes (2018) [24] found that some set function can be expressed as the sum of a submodular and a supermodular function called BP decomposition, both of which are non-negative monotone non-decreasing. Interestingly, they proved that greedy strategy can get a $\frac{1}{k_f}[1 - e^{-(1-k_g)k_f}]$ and $\frac{1-k_g}{(1-k_g)k_f+p}$ approximation under cardinality and matroid constrained, where k_f, k_g are curvature about submodular and supermodular functions. Harshaw et al. (2019) [25] focused on the difference between a monotone non-negative submodular set function and a monotone non-negative modular set function. Besides, they proposed Distorted Greedy to solve this problem and get $f(S) - g(S) \geq (1 - e^{-\gamma})f(OPT) - g(OPT)$ approximation.

1.2. Contribution

Although DS decomposition has a bright application prospect, about general set function, there is a problem worth studying how to solve DS decomposition efficiently and effectively. Also, The Distorted Greedy strategy inspires our idea to solve maximization of DS decomposition. The major contribution of our work are as follows:

- Deterministic Parameter Conditioned Greedy is a deterministic framework under cardinality constrained problem. Under some rational assumption, the Deterministic Conditioned Greedy can get a polynomial approximation for maximum DS decomposition.

- To speed the Deterministic Parameter Conditioned Greedy, we use sampling to cover optimal solution set which can reduce ground set to a small set called Random Parameter Conditioned Greedy I. Besides, we prove it can get the same approximation ratio as Deterministic Parameter Conditioned Greedy.

- For maximization of DS decomposition under without constraint, we propose Random Parameter Conditioned Greedy II to solve it. Under some rational assumption, the Random Parameter Conditioned Greedy can get a polynomial approximation for maximum DS decomposition.
- We choose two special parameters to show our algorithms can get two novel approximations $f(S_k) - (e^{-1} - c_g)g(S_k) \geq (1 - e^{-1})[f(OPT) - g(OPT)]$ and $f(S_k) - (1 - c_g)g(S_k) \geq (1 - e^{-1})f(OPT) - g(OPT)$ respectively for cardinality constrained problem and also can get two novel approximations $E[f(S_k) - (e^{-1} - c_g)g(S_k)] \geq (1 - e^{-1})[f(OPT) - g(OPT)]$ and $E[f(S_k) - (1 - c_g)g(S_k)] \geq (1 - e^{-1})f(OPT) - g(OPT)$ respectively for unconstrained problem.

1.3. Organization

The rest of this paper is organized as follow. In Section 2, we propose Deterministic Parameter Conditioned Greedy Algorithm to solve maximization of DS decomposition under cardinality constraint and use sample technique to speed our algorithm. As for without constraint, we design a Random Parameter Conditioned Greedy Algorithm II in Section 3. In Section 4, we choose some special parameters to analyze the efficiency of our framework. Conclusion and future works are in Section 5.

2. Maximization of DS Decomposition under Cardinality Constraint

In this section, we study the following problem, where f, g are the monotone non-decreasing submodular set functions, k is the cardinality constraint.

$$\begin{aligned} \max_{X \subseteq \Omega}\ & f(X) - g(X) \\ s.t. \quad & \|X\| \leq k \end{aligned} \tag{1}$$

2.1. Deterministic Parameter Conditioned Algorithm

Since Greedy Strategy is simple and efficient and gets some excellent constant approximation ratios for submodular optimization problem, many researchers use greedy strategy to deal with their problems. Therefore, we design a Deterministic Parameter Conditioned Greedy Algorithm to solve maximization DS decomposition with cardinality constraint. The details about Deterministic Parameter Conditioned Algorithm is in Algorithm 1.

From the algorithm, Line 2-Line 8 is the main loop which needs to be executed k times. Line 3 finds the element with maximal parameter decomposition marginal gains. Importantly, $A(i)$ and $B(i)$ are chosen by algorithm designers. In terms of practical applications, we assume $f(\emptyset) = g(\emptyset) = 0$, $A(i) \geq 0$ and $B(i) \geq 0$. Line 4 limits the maximal marginal gain must be positive, if negative means algorithm cannot adds any elements in this loop because all elements may not have enough large marginal gain. It is depended on our parameter how to choose. For example, when $A(i)$ is enough large and $B(i)$ is enough small, the condition of Line 4 is always satisfied. If designers choose well-defined parameters that are related to iteration round, it can get a wonderful approximation ratio about maximum DS decomposition. In this section, we assume $(1 - \frac{1}{k})A(i+1) - A(i) \geq 0$ and $B(i+1) - B(i) \geq 0$ specially. And then, we're going to go through the following process to explain why we're making this assumption.

To prove approximation ratio of Algorithm 1, firstly, we introduce a metric called curvature and two auxiliary functions which are useful in process of approximation proof.

Algorithm 1 Deterministic Parameter Conditioned Greedy

Input: Cardinality k, Parameters $A(i)$,$B(i)$
Output: S_k

1: Initialize $S_0 \leftarrow \emptyset$
2: **for** $i = 0$ to $k-1$ **do**
3:     $e_i \leftarrow \arg\max_{e \in \Omega} \{A(i+1)f(e|S_i) - B(i+1)g(e|\Omega \setminus e)\}$
4:     **if** $A(i+1)f(e_i|S_i) - B(i+1)g(e_i|\Omega \setminus e_i) > 0$ **then**
5:         $S_{i+1} \leftarrow S_i \cup \{e_i\}$
6:     **else**
7:         $S_{i+1} \leftarrow S_i$
8:     **end if**
9: **end for**
10: Return S_k

Definition 1 (Conforti (1984)). *Given a monotone submodular set function*

$f : 2^{\Omega} \to R$, the curvature of f is

$$c_f = 1 - min_{e\in\Omega}\frac{f(e|\Omega \setminus e)}{f(e)}$$

Definition 2. *Define two auxiliary functions:*

$$\phi_i(T) = A(i)f(T) - B(i)\sum_{e\in T} g(e|\Omega \setminus e)$$

$$\psi_i(T, e) = \max\{0, A(i+1)f(e|T) - B(i+1)g(e|\Omega \setminus e)\}$$

From Definition 2, $\psi_i(T, e)$ is the condition of the Line 4 in Algorithm 1. What's more, $\phi_i(T)$ is the surrogate objective function. Clearly, if we take $\phi_i(T)$ as our objective function, Algorithm 1 is the classical greedy algorithm just combined non-negative condition. Next, we prove an important property about surrogate objective function.

Property 1. *In each iteration*

$$\phi_{i+1}(S_{i+1}) - \phi_i(S_i) = \psi_i(S_i, e_i) + [A(i+1) - A(i)]f(S_i) - [B(i+1) - B(i)]\sum_{e\in S_i} g(e|\Omega \setminus e)$$

Proof.

$$\begin{aligned}
&\phi_{i+1}(S_{i+1}) - \phi_i(S_i) \\
=& A(i+1)f(S_{i+1}) - B(i+1)\sum_{e\in S_{i+1}} g(e|\Omega \setminus e) - A(i)f(S_i) + B(i)\sum_{e\in S_i} g(e|\Omega \setminus e) \\
=& A(i+1)[f(S_{i+1} - f(S_i))] - B(i+1)g(e_i|\Omega \setminus e_i) + [A(i+1) - A(i)]f(S_i) \\
&- [B(i+1) - B(i)]\sum_{e\in S_i} g(e|\Omega \setminus e) \\
=& \psi_i(S_i, e_i) + [A(i+1) - A(i)]f(S_i) - [B(i+1) - B(i)]\sum_{e\in S_i} g(e|\Omega \setminus e)
\end{aligned}$$

□

Using the Property 1, we construct the relationship from Deterministic Parameter Conditioned Greedy to surrogate objective function. That is why we introduce the Definition 2. Coming from the proof of the approximation ratio of greedy algorithm in the submodular maximization, analyzing the marginal gain of surrogate objective function in each iteration is essential for approximation ratio for Algorithm 1. Interestingly, the condition of Algorithm 1 has a lower bound marginal gains and we prove it in Theorem 1.

Theorem 1. $\psi_i(S_i, e_i) \geq \frac{1}{k}A(i+1)[f(OPT) - f(S_i)] - \frac{1}{k}B(i+1)g(OPT)$

Proof.

$$\begin{aligned}
&k \cdot \psi_i(S_i, e_i) = k \cdot \max_{e\in\Omega} \{0, A(i+1)f(e|S_i) - B(i+1)g(e|\Omega \setminus e)\} \\
&\geq |OPT| \cdot \max_{e\in\Omega} \{0, A(i+1)f(e|S_i) - B(i+1)g(e|\Omega \setminus e)\} \\
&\geq |OPT| \cdot \max_{e\in OPT} \{A(i+1)f(e|S_i) - B(i+1)g(e|\Omega \setminus e)\} \\
&\geq \sum_{e\in OPT} [A(i+1)f(e|S_i) - B(i+1)g(e|\Omega \setminus e)] \\
&= A(i+1) \sum_{e\in OPT} f(e|S_i) - B(i+1) \sum_{e\in OPT} g(e|\Omega \setminus e) \\
&\geq A(i+1)[f(OPT \cup S_i) - f(S_i)] - B(i+1)g(OPT) \\
&\geq A(i+1)[f(OPT) - f(S_i)] - B(i+1)g(OPT)
\end{aligned}$$

The first inequality is $k \geq |OPT|$. The second inequality is $OPT \subseteq \Omega$. The third inequality is the maximum. The fourth inequality is submodularity, i.e.$f(OPT \cup S_i) - f(S_i) \leq \sum_{e\in OPT} f(e|S_i)$ and $\sum_{e\in OPT} g(e|\Omega \setminus e) \leq g(OPT)$. The last inequality is monotony. □

Back to Deterministic Parameter Conditioned Algorithm, the following corollary is obvious by combining Property 1 and Theorem 1,

Corollary 1.

$$\begin{aligned}
&\phi_{i+1}(S_{i+1}) - \phi_i(S_i) \geq \frac{1}{k}A(i+1)f(OPT) - \frac{1}{k}B(i+1)g(OPT)+ \\
&[(1-\frac{1}{k})A(i+1) - A(i)]f(S_i) - [B(i+1) - B(i)] \sum_{e\in S_i} g(e|\Omega \setminus e)
\end{aligned}$$

According Lemma 1, we have $\sum_{e\in S_i} g(e|\Omega \setminus e) \geq 0$ and $f(S_i) \geq 0$. That is to say the proper selection of $A(i)$ and $B(i)$ is the key to the final performance of the algorithm. If $[(1-\frac{1}{k})A(i+1) - A(i)]f(S_i) - [B(i+1) - B(i)]\sum_{e\in S_i} g(e|\Omega \setminus e) \geq 0$, the result is trivial cause we can ignore them.

As for other non-trivial parameter scenarios, it is a obstacle to analyze approximation ratio. That is why we assume $(1-\frac{1}{k})A(i+1)-A(i)\geq 0$ and $B(i+1)-B(i)\geq 0$. Therefore, we can get an approximation guarantee about maximum DS decomposition under cardinality constrained using Deterministic Parameter Conditioned Greedy.

Theorem 2. *S_k is the solution of Algorithm 1 after k iteration*

$$A(k)f(S_k)-(B(0)-c_g)g(S_k)\geq \frac{1}{k}\sum_{i=0}^{k-1}[A(i+1)f(OPT)-B(i+1)g(OPT)]$$

where c_g is the curvature of function g.

Proof. $\phi_0(S_0)=A(0)f(\emptyset)-B(0)g(\emptyset)=0$ According the curvature $c_g = 1-min\frac{g(e|\Omega\setminus e)}{g(e)}$, we have $\frac{g(e|\Omega\setminus e)}{g(e)}\geq 1-c_g$,

$$\phi_k(S_k)=A(k)f(S_k)-B(k)\sum_{e\in S_k}g(e|\Omega\setminus e)\leq A(k)f(S_k)-B(k)(1-c_g)\sum_{e\in S_k}g(e)$$
$$\leq A(k)f(S_k)-B(k)(1-c_g)g(S_k)$$

Thus, we can rewrite an accumulation statement

$$A(k)f(S_k)-B(k)(1-c_g)g(S_k)\geq \phi_k(S_k)-\phi_0(S_0)=\sum_{i=0}^{k-1}[\phi_{i+1}(S_{i+1})-\phi_i(S_i)]$$
$$\geq\sum_{i=0}^{k-1}[\frac{1}{k}A(i+1)f(OPT)-\frac{1}{k}B(i+1)g(OPT)+[(1-\frac{1}{k})A(i+1)-A(i)]f(S_i)]$$
$$-\sum_{i=0}^{k-1}[[B(i+1)-B(i)]\sum_{e\in S_i}g(e|\Omega\setminus e)]$$
$$\geq\frac{1}{k}\sum_{i=0}^{k-1}[A(i+1)f(OPT)-B(i+1)g(OPT)]-[B(k)-B(0)]\sum_{e\in S_k}g(e|\Omega\setminus e)$$
$$\geq\frac{1}{k}\sum_{i=0}^{k-1}[A(i+1)f(OPT)-B(i+1)g(OPT)]-[B(k)-B(0)]g(S_k)$$

Therefore, we can conclude

$$A(k)f(S_k)-(B(0)-c_g)g(S_k)\geq \frac{1}{k}\sum_{i=0}^{k-1}[A(i+1)f(OPT)-B(i+1)g(OPT)]$$

□

2.2. Random Parameter Conditioned Algorithm I

Although greedy is an efficient algorithm to solve discrete problem, it complexity of computation is too high to accept in large scale problem such as social network influence maximization because it must consider the all elements in ground set. To speed greedy, sampling is the widely utilized technique. Also, we use a sample set to cover the optimal solution for acceleration. Firstly, we prove an essential lemma for sample set.

Lemma 2. *$Pr[S \cap OPT \neq \emptyset] \geq (1-\epsilon)\frac{|OPT|}{k}$, if $|S| \geq \frac{-n \ln \epsilon}{k}$. Where S is a sample set, OPT is the optimal solution set and n is the size of ground set.*

Proof. $Pr[S \cap OPT = \emptyset] \leq (1 - \frac{|OPT|}{n})^{|S|} \leq e^{-|S|\frac{|OPT|}{n}} = e^{-\frac{|S|k}{n}\frac{|OPT|}{k}}$. The first inequality is the Bernoulli Distribution with $|S|$ times. The second inequality is $1-x \leq e^{-x}$. Thus, $Pr[S \cap OPT \neq \emptyset] \geq 1-e^{-\frac{|S|k}{n}\frac{|OPT|}{k}} \geq (1-e^{-\frac{|S|k}{n}})\frac{|OPT|}{k} \geq (1-\epsilon)\frac{|OPT|}{k}$ i.e $\epsilon \geq e^{-\frac{|S|k}{n}}$ where $1-e^{-ax} \geq (1-e^{-a})x$, if $x \in [0,1]$. Therefore, the sample size is $|S| \geq \frac{-n \ln \epsilon}{k}$. □

Algorithm 2 Random Parameter Conditioned Greedy I

Input: Cardinality k, Parameters $A(i)$, $B(i)$, ϵ
Output: S_k

```
Initialize S_0 ← ∅
for i = 0 to k − 1 do
    S ← choose |S| elements from ground set uniformly and randomly
    e_i ← arg max_{e∈S} {A(i+1) f(e|S_i) − B(i+1) g(e|Ω \ e)}
    if A(i+1) f(e_i|S_i) − B(i+1) g(e_i|Ω \ e_i) > 0 then
        S_{i+1} ← S_i ∪ {e_i}
    else
        S_{i+1} ← S_i
    end if
end for
Return S_k
```

Comparing Algorithm 1 and Algorithm 2, we speed algorithm just by reducing ground set to sample set. Intuitively, sample set is must smaller than ground set. For the proof of approximation ratio, we only need to prove the lower bound of the marginal gain. Other proofs are the same as the Algorithm 1.

Theorem 3. $E[\psi_i(S_i,e_i)] \geq \{\frac{1}{k}A(i+1)[f(OPT)-f(S_i)]-\frac{1}{k}B(i+1)g(OPT)\}\cdot(1-\epsilon)$

Proof.

$$\begin{aligned}
&E[\psi_i(S_i,e_i)] = E[\psi_i(S_i,e_i)|S\cap OPT \neq \emptyset]Pr[S\cap OPT \neq \emptyset]\\
&+ E[\psi_i(S_i,e_i)|S\cap OPT = \emptyset]Pr[S\cap OPT = \emptyset]\\
&\geq E[\psi_i(S_i,e_i)|S\cap OPT \neq \emptyset]Pr[S\cap OPT \neq \emptyset]\\
&= \max_{e\in S}\{0, A(i+1)f(e|S_i) - B(i+1)g(e|\Omega\setminus e)\}(1-\epsilon)\frac{|OPT|}{k}\\
&\geq \max_{e\in S\cap OPT}\{0, A(i+1)f(e|S_i) - B(i+1)g(e|\Omega\setminus e)\}\cdot(1-\epsilon)\frac{|OPT|}{k}\\
&\geq \frac{1}{|S\cap OPT|}\sum_{e\in S\cap OPT}[A(i+1)f(e|S_i) - B(i+1)g(e|\Omega\setminus e)]\cdot(1-\epsilon)\frac{|OPT|}{k}\\
&= \frac{1}{|OPT|}\sum_{e\in OPT}[A(i+1)f(e|S_i) - B(i+1)g(e|\Omega\setminus e)]\cdot(1-\epsilon)\frac{|OPT|}{k}\\
&= \{\frac{1}{k}A(i+1)\sum_{e\in OPT}f(e|S_i) - \frac{1}{k}B(i+1)\sum_{e\in OPT}g(e|\Omega\setminus e)\}\cdot(1-\epsilon)\\
&\geq \{\frac{1}{k}A(i+1)[f(OPT)-f(S_i)] - \frac{1}{k}B(i+1)g(OPT)\}\cdot(1-\epsilon)
\end{aligned}$$

The second inequality is $S\cap OPT \subseteq S$. The third inequality is that the largest value is greater than average value. And the following equality is that the sample set is chosen from ground set uniformly and randomly i.e. unbiased estimation. The third inequality is $k \geq |OPT|$. The fourth inequality is submodularity and monotonicity. □

Using the Theorem 3, we give the approximation ratio for Algorithm 2 without proof because this proof is the same as Theorem 2.

Theorem 4. *S_k is the solution of Algorithm 2 after k iteration*

$$E\{A(k)f(S_k)-(B(0)-c_g)g(S_k)\} \geq \frac{1}{k}\sum_{i=0}^{k-1}[(A(i+1)-\epsilon)f(OPT)-(B(i+1)-\epsilon)g(OPT)]$$

where c_g is the curvature of function g.

3. Maximization of DS Decomposition without Constraint

Maximization for non-monotone problem without constraint is an important problem in submodular optimization. The best approximation ratio for this problem is $\frac{1}{2}$ which is firstly proved by a random algorithm. Few years later, the deterministic algorithm also can get this ratio. Since randomness can bring a lot of uncertain factors and information, in some cases, random algorithms can get better approximation than deterministic algorithms. For non-monotone non-submodular maximization problem, so far, no algorithm has been able to give an acceptable approximation ratio. Therefore, we propose a Random Parameter Conditioned Greedy Algorithm II in this section. The following statement is the unconstraint problem for DS decomposition.

$$\max_{X\subseteq\Omega} h(X) = f(X) - g(X)$$

According Definition 1, the Property 1 is also true for Random Parameter Conditioned Greedy II. We just have to modify the assumptions a little bit $(1-\frac{1}{n})A(i+1)-A(i) \geq 0$ to get the following theorems and corollary directly.

Theorem 5. $E[\psi_i(S_i,e)] \geq \frac{1}{n}A(i+1)[f(OPT)-f(S_i)] - \frac{1}{n}B(i+1)g(OPT)$

Proof.

$$
\begin{aligned}
E[\psi_i(S_i,e_i)] &= \frac{1}{n}\cdot\sum_{e_i\in\Omega}\psi_i(S_i,e_i)\\
&\geq \frac{1}{n}\cdot\sum_{e_i\in OPT}[A(i+1)f(e_i|S_i) - B(i+1)g(e_i|\Omega\setminus e_i)]\\
&= \frac{1}{n}A(i+1)\sum_{e\in OPT} f(e_i|S_i) - \frac{1}{n}B(i+1)\sum_{e\in OPT} g(e_i|\Omega\setminus e_i)\\
&\geq \frac{1}{n}A(i+1)[f(OPT\cup S_i)-f(S_i)] - \frac{1}{n}B(i+1)g(OPT)\\
&\geq \frac{1}{n}A(i+1)[f(OPT)-f(S_i)] - \frac{1}{n}B(i+1)g(OPT)
\end{aligned}
$$

□

Corollary 2. $E[\phi_{i+1}(S_{i+1})-\phi_i(S_i)] \geq \frac{1}{n}A(i+1)f(OPT) - \frac{1}{n}B(i+1)g(OPT)$ $+ [(1-\frac{1}{n})A(i+1)-A(i)]f(S_i) - [B(i+1)-B(i)]\sum_{e\in S_i} g(e|\Omega\setminus e)$

Algorithm 3 Random Parameter Conditioned Greedy II

Input: Parameters $A(i)$,$B(i)$
Output: S_n

```
Initialize S_0 ← ∅
for i = 0 to n − 1 do
    e_i ← choose an elements from ground set uniformly and randomly.
    if A(i + 1)f(e_i|S_i) − B(i + 1)g(e_i|Ω \ e_i) > 0 then
        S_{i+1} ← S_i ∪ {e_i}
    else
        S_{i+1} ← S_i
    end if
end for
Return S_n
```

Combined with Theorem 5 and Corollary 2, using the same method as Section 2, we can prove the Random Parameter Conditioned Greedy II can get the following approximation.

Theorem 6. *S_n is the solution of Algorithm 4 after n iteration*

$$E[A(n)f(S_n)-(B(0)-c_g)g(S_n)] \geq \frac{1}{n}\sum_{i=0}^{n-1}[A(i+1)f(OPT)-B(i+1)g(OPT)]$$

where c_g is the curvature of function g.

Proof.

$$
\begin{aligned}
& E[A(n)f(S_n) - B(n)(1-c_g)g(S_n)] \\
& \geq E[\phi_n(S_n)] - E[\phi_0(S_0)] = \sum_{i=0}^{n-1} E[\phi_{i+1}(S_{i+1})] - E[\phi_i(S_i)] \\
& \geq \sum_{i=0}^{n-1}[\frac{1}{n}A(i+1)f(OPT) - \frac{1}{n}B(i+1)g(OPT) + [(1-\frac{1}{n})A(i+1) - A(i)]f(S_i)] \\
& - \sum_{i=0}^{n-1}[B(i+1) - B(i)]\sum_{e\in S_i} g(e|\Omega \setminus e) \\
& \geq \frac{1}{n}\sum_{i=0}^{n-1}[A(i+1)f(OPT) - B(i+1)g(OPT)] - [B(n) - B(0)]\sum_{e\in S_n} g(e|\Omega \setminus e) \\
& \geq \frac{1}{n}\sum_{i=0}^{n-1}[A(i+1)f(OPT) - B(i+1)g(OPT)] - [B(n) - B(0)]g(S_n)
\end{aligned}
$$

Therefore, we can conclude

$$E[A(n)f(S_n) - (B(0)-c_g)g(S_n)] \geq \frac{1}{n}\sum_{i=0}^{n-1}[A(i+1)f(OPT) - B(i+1)g(OPT)]$$

□

From the Theorem 5 and Theorem 6, we find an interesting phenomenon. If we decrease the number of iterations to k , this Random Parameter Conditioned Greedy II can also be used in problem with cardinality constrained. Since this proof is similar with Theorem 6, we just give the statement without proof.

Theorem 7. *S_k is the solution of Algorithm 4 after k iteration*

$$E[A(k)f(S_k) - (B(0)-c_g)g(S_k)] \geq \frac{1}{n}\sum_{i=0}^{k-1}[A(i+1)f(OPT) - B(i+1)g(OPT)]$$

where c_g is the curvature of function g.

4. Parameter Analyzing

In this section, we choose two special case to show our Parameter Conditioned Greedy framework. What's more, we compare the marginal gains under the different assumption about $A(i)$ and $B(i)$.

4.1. Case 1

In this case, we set $A(i) = B(i) = (1 - \frac{1}{k})^{(k-i)}$. Therefore, the Definition 1 becomes the following. Obviously, these settings satisfy all conditions and assumptions in Section 2 and Section 3. Hence, the following results are clearly. Because the proofs are similar to Section 2 and Section 3, we omit them here.

$$\phi_i(T) = (1 - \frac{1}{k})^{k-i}[f(T) - \sum_{e \in T} g(e|\Omega \setminus e)]$$

$$\psi_i(T, e) = \max\{0, (1 - \frac{1}{k})^{k-(i+1)}[f(e|S_i) - g(e|\Omega \setminus e)]\}$$

Property 2. *In each iteration*

$$\phi_{i+1}(S_{i+1}) - \phi_i(S_i) = \psi_i(S_i, e_i) + \frac{1}{k}(1 - \frac{1}{k})^{-1}\phi_i(S_i)$$

Theorem 8. *In Deterministic Parameter Conditioned Algorithm,* $\psi_i(S_i, e_i) \geq \frac{1}{k}(1 - \frac{1}{k})^{k-(i+1)}[f(OPT) - f(S_i) - g(OPT)]$. *In Random Parameter Conditioned Algorithm I,* $E[\psi_i(S_i, e_i)] \geq \{\frac{1}{k}(1 - \frac{1}{k})^{k-(i+1)}[f(OPT) - f(S_i) - g(OPT)]\} \cdot (1-\epsilon)$. *In Random Parameter Conditioned Algorithm II,* $E[\psi_i(S_i, e_i)] \geq \frac{1}{n}(1 - \frac{1}{n})^{n-(i+1)}[f(OPT) - f(S_i) - g(OPT)]$

Theorem 9. *The Deterministic Parameter Conditioned Algorithm can return* $f(S_k) - (e^{-1} - c_g)g(S_k) \geq (1 - e^{-1})[f(OPT) - g(OPT)]$ *approximation ratio solution for cardinality constraint problem. The Random Parameter Conditioned Algorithm I can return* $E[f(S_k) - (e^{-1} - c_g)g(S_k)] \geq (1 - e^{-1} - \epsilon)[f(OPT) - g(OPT)]$ *approximation ratio solution for cardinality constraint problem. The Random Parameter Conditioned Algorithm II can return* $E[f(S_n) - (e^{-1} - c_g)g(S_n)] \geq (1 - e^{-1})[f(OPT) - g(OPT)]$ *for without constraint problem. Where* c_g *is the curvature of function* g.

From Theorem 9, we find an interesting result and get the following corollary, when $c_g = 0$, i.e. the submodular function g is a modular function.

Corollary 3. *If $c_g = 0$, i.e. g is modular. For non-monotone submodular maximization with cardinality constraint, Algorithm 1 has $f(S_k)-e^{-1}g(S_k) \geq (1-e^{-1})[f(OPT)-g(OPT)]$ approximation ratio, Algorithm 2 has $E[f(S_k)-e^{-1}g(S_k)] \geq (1-e^{-1}-\epsilon)[f(OPT)-g(OPT)]$ approximation ratio. For non-monotonicity submodular maximization without constraint, Algorithm 3 has $E[f(S_n)-e^{-1}g(S_n)] \geq (1-e^{-1})[f(OPT)-g(OPT)]$ approxiamtion ratio.*

4.2. Case 2

In this case, we set $A(i) = (1-\frac{1}{k})^{(k-i)}$, $B(i) = 1$. Similarly, the Definition 1 become the following and these settings satisfy all condition in Section 2 and Section 3. Because the proofs are similar with Section 2 and Section 3. In this subsection, we have omitted all proofs.

$$\phi_i(T) = (1-\frac{1}{k})^{k-i} f(T) - \sum_{e \in T} g(e|\Omega \setminus e)$$

$$\psi_i(T,e) = \max\{0, (1-\frac{1}{k})^{k-(i+1)} f(e|S_i) - g(e|\Omega \setminus e)\}$$

Property 3. *In each iteration*

$$\phi_{i+1}(S_{i+1}) - \phi_i(S_i) = \psi_i(S_i, e_i) + \frac{1}{k}(1-\frac{1}{k})^{k-(i+1)} f(S_i)$$

Theorem 10. *In Deterministic Parameter Conditioned Algorithm, $\psi_i(S_i, e_i) \geq \frac{1}{k}(1-\frac{1}{k})^{k-(i+1)}[f(OPT)-f(S_i)] - \frac{1}{k}g(OPT)$. In Random Parameter Conditioned Algorithm I, $E[\psi_i(S_i,e_i)] \geq \{\frac{1}{k}(1-\frac{1}{k})^{k-(i+1)}[f(OPT)-f(S_i)] - \frac{1}{k}g(OPT)\} \cdot (1-\epsilon)$. In Random Parameter Conditioned II, $E[\psi_i(S_i,e_i)] \geq \frac{1}{n}(1-\frac{1}{n})^{n-(i+1)}[f(OPT)-f(S_i)] - \frac{1}{n}g(OPT)$.*

Theorem 11. *The Deterministic Parameter Conditioned Algorithm can return $f(S_k)-(1-c_g)g(S_k) \geq (1-e^{-1})f(OPT)-g(OPT)$ approximation ratio solution for cardinality constraint. The Random Parameter Conditioned Algorithm I can return $E[f(S_k)-(1-c_g)g(S_k)] \geq (1-e^{-1}-\epsilon)f(OPT)-(1-\epsilon)g(OPT)$ approxiamtion solution ratio for cardinality constraint. The Random Parameter Conditioned Algorithm II can return $E[f(S_n)-(1-c_g)g(S_n)] \geq (1-e^{-1})f(OPT)-g(OPT)$ approximation ration for without constraint. Where c_g is the curvature of function g.*

Corollary 4. *If $c_g = 0$, i.e. g is modular. For non-monotone submodular maximization with cardinality constraint, Algorithm 1 has $f(S_k) - g(S_k) \geq (1-e^{-1})f(OPT) - g(OPT)$ approximation ratio, Algorithm 2 has $E[f(S_k) - g(S_k)] \geq (1 - e^{-1} - \epsilon)f(OPT) - (1 - \epsilon)g(OPT)$ approximation ratio. For non-monotone submodular maximization without constraint, Algorithm 3 has $E[f(S_n) - g(S_n)] \geq (1 - e^{-1})f(OPT) - g(OPT)$ approxiamtion ratio.*

Remark 1:. Clearly, if g is modular, then $h = f - g$ is submodular. The above approximations are different with submodular maximization under cardinality constrained problem $(1 - e^{-1})$. And also, the different parameters can also cause different approximations. We think first gap is caused by non-monotony, because h is not always monotonous. The second gap give us a clue that we can choose the appropriate parameters $A(i)$ and $B(i)$ according to the characteristics of the problem to get a better approximate ratio. The above approximations are different with non-monotone submodular maximization under unconstrained problem 1/2. But we cannot measure which one is better than others. In some cases, our approximations may be better than 1/2.

Remark 2:. The conditions in Line 4 of Algorithm 1, 3 and Line 5 in Algorithm 2 are necessary. Firstly, if we don't add non-negative condition, we may add some bad elements which has the lower marginal gains to the solution. Combined the positive condition, it makes the large marginal gains element will be added into solution, because the conditions in Line 4 of Algorithm 1, 3 and Line 5 in Algorithm 2 are a lower bound of marginal gains. Also, we assume $(1-\frac{1}{k})A(i+1) - A(i) \geq 0$ and $B(i+1) - B(i) \geq 0$ i.e. the parameter of $A(i)$ and $B(i)$ is increasing with index i increase. If an element e is added into solution in loop i, the true marginal gains will be enough large because $A(k) = B(k) = 1$.

5. Conclusions

In this paper, we propose Parameter Conditioned Greedy strategy with deterministic and random which are general frameworks for maximum DS decomposition under cardinality constrained and unconstrained respectively. Users can choose some rational parameters to fit special practical problems and get a wonderful approximation about problem. Also, we choose two special cases show our strategy can get some novel approximation. In some situations, these novel approximations are better than the best approximation at the state of art.

In the future works, how to remove the curvature parameter in approximation ratio is important, because it can make the approximation much tight. What's more, how to select $A(i)$ and $B(i)$ so that the algorithm can achieve the optimal approximation ratio is also urgent problem to be solved.

Acknowledge

This work was supported by the National Natural Science Foundation of China under Grant 11991022 and 12071459.

Reference

References

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

[2] M. L. Fisher, G. L. Nemhauser, L. A. Wolsey, An analysis of approximations for maximizing submodular set functions—ii, Mathematical Programming 8 (1) (1978) 73–87.

[3] Michele, Conforti, , , 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 (1984).

[4] M. Feldman, J. S. Naor, R. Schwartz, A unified continuous greedy algorithm for submodular maximization, IEEE Computer Society, USA, 2011. doi:10.1109/FOCS.2011.46.
URL `https://doi.org/10.1109/FOCS.2011.46`

[5] N. Buchbinder, M. Feldman, J. S. Naor, R. Schwartz, Submodular maximization with cardinality constraints, Society for Industrial and Applied Mathematics, USA, 2014.

[6] N. Buchbinder, M. Feldman, J. S. Naor, R. Schwartz, A tight linear time (1/2)-approximation for unconstrained submodular maximization, IEEE Computer Society, USA, 2012. doi:10.1109/FOCS.2012.73. URL `https://doi.org/10.1109/FOCS.2012.73`

[7] D. Kempe, Maximizing the spread of influence through a social network, Proc.of Acm Sigkdd Intl Conf.on Knowledge Discovery and Data Mining (2003).

[8] D. Golovin, A. Krause, Adaptive submodularity: Theory and applications in active learning and stochastic optimization, Journal of Artificial Intelligence Research 42 (1) (2012) 427–486.

[9] H. Lin, J. Bilmes, A class of submodular functions for document summarization, Association for Computational Linguistics, USA, 2011.

[10] S. Jegelka, J. Bilmes, Submodularity beyond submodular energies: Coupling edges in graph cuts, IEEE Computer Society, 2011.

[11] A. Krause, A. Singh, C. Guestrin, Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies, Journal of Machine Learning Research 9 (3) (2008) 235–284.

[12] A. Das, D. Kempe, Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection, Computer ence (2011).

[13] T. Horel, Y. Singer, Maximization of approximately submodular functions, in: Advances in neural information processing systems, 2016.

[14] W. Lu, W. Chen, L. V. S. Lakshmanan, From competition to complementarity: Comparative influence diffusion and maximization, proceedings of the vldb endowment 9 (2) (2015) 60–71.

[15] W. Yang, S. Chen, S. Gao, R. Yan, Boosting node activity by recommendations in social networks, Journal of Combinatorial Optimization (2020).

[16] W. Yang, Y. Zhang, D. zhu Du, Influence maximization problem: properties and algorithms, Journal of Combinatorial Optimization (2020).

[17] J. Zhu, S. Ghosh, J. Zhu, W. Wu, Near-optimal convergent approach for composed influence maximization problem in social networks, IEEE Access PP (99) (2019) 1–1.

[18] Z. Wang, Y. Yang, J. Pei, L. Chu, E. Chen, Activity maximization by effective information diffusion in social networks, IEEE Transactions on Knowledge and Data Engineering 29 (11) (2017) 2374–2387.

[19] R. Iyer, J. Bilmes, Algorithms for approximate minimization of the difference between submodular functions, with applications, AUAI Press, Arlington, Virginia, USA, 2012.

[20] K. Han, C. Xu, F. Gui, S. Tang, H. Huang, J. Luo, Discount allocation for revenue maximization in online social networks, 2018, pp. 121–130. doi:10.1145/3209582.3209595.

[21] Maehara, Takanori, Murota, Kazuo, A framework of discrete dc programming by discrete convex analysis, Mathematical Programming (2015).

[22] J. Yu, M. Blaschko, A convex surrogate operator for general non-modular loss functions (04 2016).

[23] X. Li, H. G. Du, P. M. Pardalos, A variation of ds decomposition in set function optimization, Journal of Combinatorial Optimization (2020).

[24] W. Bai, J. A. Bilmes, Greed is still good: Maximizing monotone submodular+supermodular functions (2018).

[25] C. Harshaw, M. Feldman, J. Ward, A. Karbasi, Submodular maximization beyond non-negativity: Guarantees, fast algorithms, and applications, Vol. 2019-June, Long Beach, CA, United states, 2019, pp. 4684 – 4705.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: