Streaming k-Submodular Maximization under Noise subject to Size Constraint

Lan N. Nguyen [1] **My T. Thai** [1]

Abstract

Maximizing on k-submodular functions subject to size constraint has received extensive attention recently. In this paper, we investigate a more realistic scenario of this problem that (1) obtaining exact evaluation of an objective function is impractical, instead, its noisy version is acquired; and (2) algorithms are required to take only one single pass over dataset, producing solutions in a timely manner. We propose two novel streaming algorithms, namely DSTREAM and RSTREAM, with their theoretical performance guarantees. We further demonstrate the efficiency of our algorithms in two applications in Influence Maximization and Sensor Placement, showing that our algorithms can return comparative results to state-of-the-art non-streaming methods while using a much fewer number of queries.

1. Introduction

Maximizing a k-submodular function subject to size constraint (**M**k**SC**) has been studied by Ohsaka & Yoshida (2015); Zhou et al. (2019); Qian et al. (2017a); Sakaue (2017). In this problem, a finite set V is given, let $(k+1)^V = \{(X_1, X_2, ...X_k) \mid X_i \subseteq V \; \forall i \in [k], X_i \cap X_j = \emptyset \; \forall i \neq j\}$ be a family of k disjoint sets. Given a k-submodular function $f : (k+1)^V \to \mathbb{R}$ and a positive integer B, **M**k**SC** asks for $\mathbf{s} = \{S_1, ..., S_k\} \in (k+1)^V$ subject to $|S_1 \cup ... \cup S_k| \leq B$ that maximizes $f(\mathbf{s})$.

Many applications of **M**k**SC** have been investigated in literature. For example, in **Influence Maximization** with k topics (Ohsaka & Yoshida, 2015; Zhou et al., 2019; Qian et al., 2017a), the problem asks for B social users, each initially adopts a topic, that maximizes the expected number of users, who are eventually activated by at least one topic.

[1]Department of Computer and Information Science and Engineering, University of Florida, Gainesville, Florida, United States. Correspondence to: Lan N. Nguyen <lan.nguyen@ufl.edu>, My T. Thai <mythai@cise.ufl.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

Another readily application is **Sensor Placement** with k types of measures (Ohsaka & Yoshida, 2015; Qian et al., 2017a), which asks for B locations within n given locations and each with what type of sensors that maximizes the information gain. Another applications of **M**k**SC** can also be found in information coverage problem (Qian et al., 2017a) and coupled feature selection (Singh et al., 2012).

However, existing literature has largely ignored the fact that querying f may be expensive and could only be achieved with some errors. For example, in the Influence Maximization or Information Coverage problem, exact computing $f(\mathbf{s})$ is #P-hard (Kempe et al., 2003), which requires at least exponential number of computations.

Therefore, instead of exact querying, an approximate oracle of f, which approximates f within an amount of error, should be considered. In this paper, we consider a noisy version of f, denoted as F, which is accessible with a *multiplicative error* (Qian et al., 2015) ϵ, guarantee $(1-\epsilon)f(\mathbf{s}) \leq F(\mathbf{s}) \leq (1+\epsilon)f(\mathbf{s})$ for all $\mathbf{s} \in (k+1)^V$.

Moreover, in many applications, data volumes are increasing massive in scale, making it impractical to store a whole dataset in computer memory. It is critical to process data one by one in a streaming fashion, which not only reduces the burden on memory storage but also be able to produce solutions in a timely manner. Motivated by these observation, we aim to solve the noisy **M**k**SC** problem in the streaming fashion. Our algorithms require *only one single scan* over V while providing performance guarantees in terms of approximation ratios, memory, and query complexity.

Solving **M**k**SC** with noisy f in streaming fashion is quite challenging, indeed. First, F may shadow f's properties, i.e. F is not k-submodular and even decreasing when f is increasing. Also, the noise ϵ can mislead a process of constructing solution by magnifying the marginal gain of a selection whose contribution may be insignificant. On another hand, even streaming algorithms for the monotone submodular function maximization (a special case of **M**k**SC**) has been studied, directly applying them to noisy **M**k**SC** gives unclear performance guarantees due to intrinsic differences between submodularity and k-submodularity. That leaves a task of solving noisy **M**k**SC** in the streaming fashion widely open and to our knowledge, we are the first one studying the problem.

Our Contribution. We propose two novel streaming algorithms to approximate noisy **MkSC**. Both our algorithms have an approximation ratio of $O((1-\epsilon)^{-2}\epsilon B)$ when f is **monotone**; and $O((1-\epsilon)^{-3}\epsilon B)$ when f is **non-monotone**. To be specific, our main contributions are:

- Our first algorithm is DSTREAM, a deterministic streaming method, which exploits a Greedy concept to work in the streaming scenario. In general, for each $e \in V$ when being observed, DSTREAM will greedily put e into set i that maximizes the ϵ-estimate of f as long as the estimated value is sufficiently large in comparison with an estimate of the optimal solution.
- The second algorithm is RSTREAM, a randomized streaming method, which exploits the randomized framework for unconstrained k-submodular maximization and introduces a new probability distribution with a constraint to bound solution size. In general, for each $e \in V$ when being observed, RSTREAM will randomly add e into set i with probability proportional to an upper bound of marginal gain on f, as long as the bound is sufficiently large in comparison with an estimate of the optimal solution. Since the bound may be significantly larger than f's actual marginal gain due to ϵ, RSTREAM proposes a *denoise* step to improve returned solution quality.
- We experimentally investigate our algorithms' performance in comparison with existing methods on two applications of Noisy **MkSC**: Influence Maximization with k topics and Sensor Placement with k measures. The experimental results show our algorithms perform comparatively to state-of-the-art non-streaming algorithms in quality of solutions but outperform them in term of number of queries. We further investigate the trade-offs between quality of solution versus number of queries of our algorithms on different settings.

Organization. In Section 2, we review definition of k-submodular functions and most recent works related to our paper. Section 3 and 4 provide detail description and analysis of our two streaming algorithms. Section 5 shows experimental results and Section 6 concludes our paper.

2. Preliminaries

2.1. Problem Definition

Given a finite set V, let $(k+1)^V = \{(X_1, X_2, ... X_k) \mid X_i \subseteq V \; \forall i \in [k], X_i \cap X_j = \emptyset \; \forall i \neq j\}$ be a family of k disjoint subsets of V. For simplicity, we call a tuple of k disjoint subsets a k**-set**.

Given a k-set $\mathbf{x} = \{X_1, X_2, ..., X_k\}$, $\mathbf{x}$ can be written as a mapping that $\mathbf{x}(e) = i$ if $e \in X_i$; $\mathbf{x}(e) = 0$ if $e \notin \cup_{i\in[k]} X_i$. Also, let $supp(\mathbf{x}) = \cup_{i\in[k]} X_i$. For simplicity, denote $|\mathbf{x}| = |supp(\mathbf{x})|$.

A function $f : (k+1)^V \to \mathbb{R}$ is k**-submodular** iff for any $\mathbf{x} = (X_1, ..., X_k)$ and $\mathbf{y} = (Y_1, ..., Y_k) \in (k+1)^V$

$$f(\mathbf{x}) + f(\mathbf{y}) \geq f(\mathbf{x} \sqcap \mathbf{y}) + f(\mathbf{x} \sqcup \mathbf{y})$$

where $\mathbf{x} \sqcap \mathbf{y} = (X_1 \cap Y_1, ..., X_k \cap Y_k)$; and $\mathbf{x} \sqcup \mathbf{y} = (Z_1, ... Z_k)$ where $Z_i = X_i \cup Y_i \setminus (\bigcup_{j\neq i} X_j \cup Y_j)$.

A function $f : (k+1)^V \to \mathbb{R}$ is **monotone** iff for any $\mathbf{x} \in (k+1)^V$, $e \notin supp(\mathbf{x})$ and $i \in [k]$:

$$\begin{aligned}\Delta_{e,i} f(\mathbf{x}) = f(X_1, ..., X_{i-1}, X_i \cup \{e\}, X_{i+1}, ..., X_k) \\ - f(X_1, ..., X_k) \geq 0\end{aligned}$$

A function $f : (k+1)^V \to \mathbb{R}$ is **normalized** iff $f(\{\emptyset, \emptyset, ... \emptyset\}) = 0$.

With $\mathbf{x} = \{X_1, ..., X_k\} \in (k+1)^V$ that $X_i = \{e\}$ and $X_j = \emptyset \; \forall j \neq i$, denote $\mathbf{x}$ as $\langle e, i \rangle$. Therefore, given $\mathbf{x} = \{X_1, ..., X_k\}$ and $e \notin supp(\mathbf{x})$, adding e into X_i can be represented by $\mathbf{x} \sqcup \langle e, i \rangle$.

Given $\mathbf{x} = \{X_1, ..., X_k\}$ and $\mathbf{y} = \{Y_1, ..., Y_k\}$, denote $\mathbf{x} \sqsubseteq \mathbf{y}$ iff $X_i \subseteq Y_i$ for all $i \in [k]$.

A function $F : (k+1)^V \to \mathbb{R}$ is an ϵ**-estimate** of f iff $(1-\epsilon) f(\mathbf{x}) \leq F(\mathbf{x}) \leq (1+\epsilon) f(\mathbf{x})$ for all $\mathbf{x} \in (k+1)^V$.

The noisy **MkSC** problem is formally defined as follows:

Definition 1. (**Noisy MkSC**) Given a finite set V, an ϵ-estimate F of a normalized k-submodular function $f : (k+1)^V \to \mathbb{R}$ and a positive integer B, identify $\mathbf{s} \in (k+1)^V$ such that $|\mathbf{s}| \leq B$ and $f(\mathbf{s})$ is maximized.

Denote $\mathbf{o}$ as an optimal solution, i.e $f(\mathbf{o}) = \max_{|\mathbf{s}| \leq B} f(\mathbf{s})$.

With $k = 1$, **MkSC** becomes submodular maximization under cardinality constraint. Thus, in this paper, we only consider $k \geq 2$.

2.2. Related Work

To our knowledge, we are the first one studying Noisy **MkSC** in the streaming fashion. In the following discussion, we pay attention to recent studies on the **MkSC** problem, optimization on problems involving noisy function or requiring streaming processing.

Maximizing k-submodular function: k-submodular concept was first introduced by Singh et al. (2012) and further investigated by Ward & Živný (2014; 2016); Iwata et al. (2016); Oshima (2017); Soma (2019). However, the authors only focused on studying unconstrained k-submodular maximization, a special case of **MkSC** where $B = |V|$.

MkSC was first studied by Ohsaka & Yoshida (2015), in which the authors extended the greedy framework of Ward

& Živnỳ (2014) to solve **MkSC** with monotone objective function. Two algorithms, proposed by the authors, were basically variation of classical greedy and both have approximation ratio of 2. Greedy algorithm also showed to be efficient on k-submodular maximization subject to matroid constraint (Sakaue, 2017). In Appendix B, we prove that with Noisy **MkSC**, Greedy is able to provide $\frac{2+2\epsilon B}{1-\epsilon}$ approximation ratio when f is monotone and $\frac{(2+2\epsilon+4\epsilon B)(1+\epsilon)}{(1-\epsilon)^2}+1$ in case of non-monotonicity. Greedy would be used as a baseline to compare with our two streaming algorithms.

MkSC with monotone function was then further studied by Zhou et al. (2019); Qian et al. (2017a) using multi-objective evolutionary algorithms. In term of theoretical performance guarantee, their algorithms run in $8eB$ iterations, each may take $O(k|V|\ln^2 B)$ queries in expectation, in order to obtain 2-ratio.

Noisy Function Optimization. Although there is no study on Noisy **MkSC** in general, its special case with $k = 1$ (submodular function) has been studied extensively recently. Submodular optimization under noise was first introduced by Hassidim & Singer (2017) and further studied by Horel & Singer (2016); Singer & Hassidim (2018); Crawford et al. (2019); Qian et al. (2017b); Qian (2019). However, existing algorithms are either: (1) impractical to apply to Noisy **MkSC** due to intrinsic difference between submodularity and k-submodularity; or (2) lack of noise awareness; or (3) too expensive in runtime complexity.

Also, in many application of noisy **MkSC** such as Influence maximization (Ohsaka & Yoshida, 2015; Zhou et al., 2019; Qian et al., 2017b) or Information Coverage (Qian et al., 2017b), the noise ϵ can be obtained. In appendix E, we illustrate how to to control the value of ϵ in the application of Influence Maximization with k topics. The same technique can also be applied to control ϵ's value in the Information Coverage application.

Indeed, known ϵ is very critical. Considering an instance of Noisy **MkSC** that: due to extreme noise, F decreases when f increases, and increases when f decreases; existing algorithms without ϵ awareness like Greedy could end up with elements which decrease f. Meanwhile, our algorithms, by utilizing the fact that the noise ϵ is known, can add elements that increase f by considering the bound of their gains, which involves ϵ.

Streaming Optimization. Although there is no study for solving **MkSC** in the streaming fashion in general, two of its special cases have been investigated, when $B = |V|$ (unconstrained k-submodular maximization) and when $k = 1$ (submodular maximization with cardinality constraint).

In unconstrained k-submodular maximization, without directly stating, one can trivially see that algorithms proposed by Ward & Živnỳ (2014; 2016); Iwata et al. (2016); Soma (2019); Oshima (2017) work in the streaming manner, which requires only one pass over dataset. However, those algorithms cannot be directly applied to **MkSC** because they utilized a fact that: a k-submodular function f implies *pairwise monotonicity*. Thus, f always reach maximum at $|\mathbf{o}| = |V|$. Indeed, RSTREAM exploits the randomized framework proposed by Ward & Živnỳ (2014) and introduces a new probability distribution, restricting elements with insignificant contribution and obeying the size constraint of Noisy **MkSC**.

Streaming submodular maximization with cardinality constraint has been studied by Gomes & Krause (2010); Kumar et al. (2015); Bateni et al. (2017); Badanidiyuru et al. (2014); Yang et al. (2019) but inapplicable to **MkSC** due to intrinsic differences between submodularity and k-submodularity. Indeed, our work was inspired by Badanidiyuru et al. (2014); Yang et al. (2019), in which we also first assume the optimal solution is known in order to sequentially make decision for each element observation. That assumption is removed by establishing a sequence of estimate on the optimum.

However, in contrast to Badanidiyuru et al. (2014); Yang et al. (2019), solving **MkSC** struggles from the fact that there could be multiple favourable sets that satisfy a selection condition. A fair set choice with approximation guarantee is of interest. Furthermore, their algorithms work by exploiting the submodularity of the objective function, which does not exist on k-submodular functions. For example, a union of the optimal solution with any solution may not have better f value than the optimal one, i.e. $f(\mathbf{o} \sqcup \mathbf{x})$ may be less than $f(\mathbf{o})$ if $\mathbf{o}(e) \neq \mathbf{x}(e) \neq 0\, \forall e \in V$. Instead, our algorithms work by exploiting a sequence of k-sets $\{\mathbf{o}^j\}_j$, created from $\mathbf{o}$ and a sequence of k-sets obtained by the algorithms. The novelty of our approaches comes from mathematically modelling relationship between $\mathbf{o}^j$ and the returned solution with size constraint, no matter the returned solution reaches budget B or not after a single pass over V.

3. DSTREAM Algorithm

In general, DSTREAM is a streaming algorithm by taking only one single pass over V. For each element $e \in V$ when being observed, DSTREAM works in greedy manner by putting e into a set i that guarantees the ϵ-estimate of f is maximized and large enough in comparison with $f(\mathbf{o})$. For an ease of presentation, we first assume $f(\mathbf{o})$ is known to investigate DSTREAM approximation ratio. Then, the assumption is removed by adapting a lazy estimation method (Badanidiyuru et al., 2014) to the Noisy **MkSC**.

3.1. DSTREAM with known $f(\mathbf{o})$

Description. With $f(\mathbf{o})$ is known, DSTREAM receives an input o that satisfies $f(\mathbf{o}) \geq o \times B \geq \frac{1}{1+\gamma} f(\mathbf{o})$. The

Algorithm 1 DSTREAM with known $f(\mathbf{o})$

Input F, B, k and o that $f(\mathbf{o}) \geq o \times B \geq f(\mathbf{o})/(1+\gamma)$

```
s^0 = {∅, ...∅}; t = 0
for each e in V do
    if |s^t| < B then
        i = argmax_{i'∈[k]} F(s^t ⊔ ⟨e, i'⟩)
        if F(s^t ⊔ ⟨e,i⟩)/(1−ε) ≥ (t+1) o/M then
            s^{t+1} = s^t ⊔ ⟨e, i⟩
            t = t + 1
```

Return $\mathbf{s}^t$ if f is **monotone**; $argmax_{\mathbf{s}^j; j\leq t} F(\mathbf{s}^j)$ if f is **non-monotone**.

algorithm starts with $\mathbf{s}$, which is initially empty.

DSTREAM takes a single pass over V. For each $e \in V$ when being observed, DSTREAM identifies $i \in [k]$ that maximizes $F(\mathbf{s} \sqcup \langle e, i\rangle)$. $\langle e, i\rangle$ is added into $\mathbf{s}$ if $\frac{F(\mathbf{s} \sqcup \langle e,i\rangle)}{1-\epsilon} \geq (|\mathbf{s}|+1)\frac{o}{M}$ where M is the algorithm's parameter. This constraint is to filter out elements with insignificant contribution from being added into $\mathbf{s}$. The factor $\frac{1}{1-\epsilon}$ of $F(\mathbf{s} \sqcup \langle e, i\rangle)$ is to present the largest possible value of $f(\mathbf{s} \sqcup \langle e, i\rangle)$.

After a single pass over V, DSTREAM returns $\mathbf{s}$ if f is monotone. Otherwise, the algorithm returns $argmax_{\mathbf{s}^j; j\leq t} F(\mathbf{s}^j)$, where $\mathbf{s}^j \sqsubseteq \mathbf{s}$ is a k-set containing the first j elements selected by DSTREAM.

DSTREAM's pseudocode with known $f(\mathbf{o})$ is fully presented in Alg. 1. The algorithm's approximation ratio with known $f(\mathbf{o})$ is presented in Proposition 1 and detail proofs are provided in Appendix C.1.

Proposition 1. Given an instance of Noisy **MkSC** with input V, k, B, o, M and F is an ϵ-estimate of the **monotone** k-submodular objective function f. If o satisfies $f(\mathbf{o}) \geq oB \geq f(\mathbf{o})/(1+\gamma)$, $\mathbf{s}$ is an output of Alg. 1, then $f(\mathbf{o}) \leq \max\left((1+\gamma)(1+\epsilon), \frac{2+4B\epsilon}{M-1}\right)\frac{M}{1-\epsilon} f(\mathbf{s})$.

Proof overview. Observe that after a single pass over V, t may or may not reach a value of B. If $t = B$, then

$$f(\mathbf{s}) \geq \frac{F(\mathbf{s}^B)}{1+\epsilon} \geq \frac{1-\epsilon}{1+\epsilon} B \frac{o}{M} \geq \frac{1-\epsilon}{1+\epsilon} \frac{f(\mathbf{o})}{(1+\gamma)M} \tag{1}$$

and the result follows. Thus, for the rest of the proof, we focus on the case $t < B$.

For each $j \in \{1, ..., t\}$, define $\mathbf{o}^j = (\mathbf{o} \sqcup \mathbf{s}^j) \sqcup \mathbf{s}^j$. The key points of this proof come from two following claims.

Claim 1. $f(\mathbf{o}) - f(\mathbf{o}^t) \leq \frac{1+\epsilon+2B\epsilon}{1-\epsilon} f(\mathbf{s})$

This claim is obtained by exploiting the k-submodularity and monotonicity of the objective function f.

Moreover, as $t < B$, there exists no $e \in V \setminus supp(\mathbf{s}^t)$ and $i \in [k]$ that $F(\mathbf{s}^t \sqcup \langle e, i\rangle) \geq (t+1)\frac{o}{M}$, which is critical to obtain the following claim

Claim 2. $f(\mathbf{o}^t) - f(\mathbf{s}) \leq \frac{1}{M} f(\mathbf{o}) + \frac{2\epsilon B}{1-\epsilon} f(\mathbf{s})$

The ratio is obtained by combing these two claims. □

3.2. DSTREAM without known $f(\mathbf{o})$

In this part, we remove the assumption that $f(\mathbf{o})$ is known and investigate DSTREAM's memory and query complexity.

To find o such that $f(\mathbf{o}) \geq oB \geq \frac{f(\mathbf{o})}{1+\gamma}$, DSTREAM adapts a lazy estimation method (Badanidiyuru et al., 2014) to Noisy **MkSC** as follows: First, it is a natural observation that:

$$f(\mathbf{o}) \leq B \cdot \max_{e\in V, i\in k} f(\langle e, i\rangle) \leq B \cdot \max_{e\in V, i\in k} \frac{F(\langle e, i\rangle)}{1-\epsilon}$$

Furthermore, $f(\mathbf{o})$ can be lower bounded by:

$$f(\mathbf{o}) \geq \max_{e\in V, i\in k} f(\langle e, i\rangle) \geq \max_{e\in V, i\in k} \frac{F(\langle e, i\rangle)}{1+\epsilon}$$

Let $\Delta_u = \max_{e,i} \frac{F(\langle e,i\rangle)}{1-\epsilon}$; $\Delta_l = \max_{e,i} \frac{F(\langle e,i\rangle)}{(1+\epsilon)(1+\gamma)}$ and define $O = \{(1+\gamma)^j \mid j \in \mathbb{Z}^{\geq}, \frac{\Delta_l}{B\cdot M} \leq (1+\gamma)^j \leq \frac{\Delta_u}{M}\}$.

As $f(\mathbf{o}) \in [\Delta_l, B\Delta_u]$, there should exist $v \in O$ such that $v \in [\frac{f(\mathbf{o})}{M\cdot B(1+\gamma)}, \frac{f(\mathbf{o})}{M\cdot B}]$. vM is an ideal value for o. We can simply run Alg. 1 with input $o = vM$ to produce a candidate solution for each $v \in O$ and return the best solution obtained.

The only remaining challenge now is: identifying Δ_u and Δ_l requires at least one pass over V. DSTREAM handles that by lazily maintaining $\Delta_u = \frac{\max_{e\in V_p, i\in[k]} F(\langle e,i\rangle)}{1-\epsilon}$ and $\Delta_l = \frac{\max_{e\in V_p, i\in[k]} F(\langle e,i\rangle)}{(1+\epsilon)(1+\gamma)}$ where V_p is a set of elements that were already observed; and considering all $v = (1+\gamma)^j \in [\frac{\Delta_l}{BM}, (1+\epsilon)\Delta_u]$. Note that the upper bound of that range is $(1+\epsilon)\Delta_u$ instead of $\frac{\Delta_u}{M}$ in order to cover the largest v that DSTREAM can produce a non-empty solution with already observed elements, i.e. $\frac{F(\mathbf{s})}{1-\epsilon} \leq (1+\epsilon)|\mathbf{s}|\Delta_u$ $\forall \mathbf{s} \in (1+k)^V$ that $supp(\mathbf{s}) \subseteq V_p$. That helps DSTREAM preserve potential solution if that range shifts forward when observing next elements.

The full pseudo code of DSTREAM is presented in Alg. 2. In Alg. 2, t_j and $\{\mathbf{s}^i_j\}_i$ are to keep track the number of selected elements and k-sets produced by DSTREAM when v is estimated by $(1+\gamma)^j$. Theorem 1 shows DSTREAM's memory and query complexity; and approximation ratio when f is monotone increasing.

Theorem 1. Given an instance of Noisy **MkSC** with input V, k, B, γ, M and F is an ϵ-estimate of the k-submodular objective function f. DSTREAM has query

Algorithm 2 DSTREAM

Input $V, F, k, B, M > 1, \gamma > 0$

$\Delta_u = \Delta_l = \Delta = 0$; $t_j = 0\ \forall j \in \mathbb{Z}^+$
for each e in V **do**
  $\Delta = \max\big(\Delta, \max_{j\in[k]} F(\langle e, j\rangle)\big)$
  $\Delta_u = \Delta/(1-\epsilon)$; $\Delta_l = \Delta/\big((1+\epsilon)(1+\gamma)\big)$
  $O = \{(1+\gamma)^j \mid \frac{\Delta_l}{B\cdot M} \le (1+\gamma)^j \le (1+\epsilon)\Delta_u\}$
  for each j that $(1+\gamma)^j \in O$ **do**
    $o = M(1+\gamma)^j$
    if $t_j < B$ **then**
      $i = argmax_{j'\in[k]} F(\mathbf{s}_j^{t_j} \sqcup \langle e, j'\rangle)$
      if $\frac{F(\mathbf{s}_j^{t_j} \sqcup \langle e, i\rangle)}{1-\epsilon} \ge (t_j+1)\frac{o}{M}$ **then**
        $\mathbf{s}_j^{t_j+1} = \mathbf{s}_j^{t_j} \sqcup \langle e, i\rangle$
        $t_j = t_j + 1$

Return $argmax_{\mathbf{s}_j^{t_j}; j\in O} F(\mathbf{s}_j^{t_j})$ if f is **monotone**; $argmax_{\mathbf{s}_j^{i}; i \le t_j, j\in O} F(\mathbf{s}_j^{i})$ if f is **non-monotone**.

complexity of $O\big(\frac{|V|k}{\gamma}\log(\frac{(1+\epsilon)(1+\gamma))}{1-\epsilon}BM)\big)$ and takes $O\big(\frac{B}{\gamma}\log(\frac{(1+\epsilon)(1+\gamma)}{1-\epsilon}BM)\big)$ memory. If f is **monotone**, then $f(\mathbf{o}) \le \frac{1+\epsilon}{1-\epsilon}\min_{x\in(1,M]} h(x) f(\mathbf{s})$ where $\mathbf{s}$ is an output of DSTREAM, $\mathbf{o}$ is an optimal solution and:

$$h(x) = \max\left((1+\gamma)(1+\epsilon), \frac{2+4B\epsilon}{x-1}\right)\frac{x}{1-\epsilon}$$

Proof. Although we have proven that the approximation guarantee of the algorithm is $\max\big((1+\gamma)(1+\epsilon), \frac{2+4B\epsilon}{M-1}\big)\frac{M}{1-\epsilon}$ if $f(\mathbf{o})$ is known, the final pseudocode has shown that M only plays a role in deciding a lower bound of $(1+\gamma)^j$. Thus, the algorithm with a value of M would achieve a ratio no worse than the one with smaller M. Thus:

$$\min_{x\in(1,M]} h(x) \max_{\mathbf{s}_j^{t_j}} f(\mathbf{s}_j^{t_j}) \ge f(\mathbf{o})$$

The final ratio follows by observing that $f(\mathbf{s}) \ge \frac{F(\mathbf{s})}{1+\epsilon} \ge \frac{1}{1+\epsilon}\max_{\mathbf{s}_j^{t_j}} F(\mathbf{s}_j^{t_j}) \ge \frac{1-\epsilon}{1+\epsilon}\max_{\mathbf{s}_j^{t_j}} f(\mathbf{s}_j^{t_j})$.

As $\Delta_u = \frac{(1+\epsilon)(1+\gamma)}{1-\epsilon}\Delta_l$, the maximum number of j's values to be considered is $l = O(\log(\frac{(1+\epsilon)(1+\gamma)}{1-\epsilon}BM)\frac{1}{\gamma})$. Thus, for each $e \in V$, the algorithm queries f at most lk times. Also, due to constraint $|\mathbf{s}_j| \le B$, the algorithm occupies at most Bl memory to store $\mathbf{s}_j$. $\square$

We further investigate DSTREAM's approximation ratio when f is **non-monotone**. The ratio is stated in Theorem 2 and proof is provided in Appendix. C.2.

Theorem 2. Given an instance of Noisy **MkSC** with input V, k, B, γ, M and F is an ϵ-estimate of the k-submodular objective function f. If f is **non-monotone**, then $f(\mathbf{o}) \le \frac{1+\epsilon}{1-\epsilon}\min_{x\in(1,M]}\max\big(a(x), b(x)\big) f(\mathbf{s})$, where $\mathbf{s}$ is an output of DSTREAM, $\mathbf{o}$ is an optimal solution and

$$a(x) = \frac{(1+\gamma)(1+\epsilon)}{1-\epsilon}x$$
$$b(x) = \frac{3+4\epsilon+6\epsilon B+\epsilon^2+2\epsilon^2 B}{(1-\epsilon)^2}\frac{x}{x-1}$$

4. RSTREAM Algorithm

RSTREAM is also a streaming algorithm, taking only one single pass over V. For each $e \in V$ when being observed, $\langle e, i\rangle$ is randomly selected by RSTREAM with a probability proportional to its upper bound on the marginal gain. However, RSTREAM is vulnerable to ϵ. Thus a **denoise** stage is proposed to improve RSTREAM's performance.

Description. Similar to DSTREAM, RSTREAM also requires to know o, which satisfies $f(\mathbf{o}) \ge o \times B \ge \frac{1}{1+\gamma}f(\mathbf{o})$. Thus RSTREAM uses a similar lazy estimation as in DSTREAM to obtain o. Without loss of generality, in the following algorithm description, we assume that o is already obtained. RSTREAM starts with $\mathbf{s}$ as an empty set initially.

RSTREAM takes a single scan over V. For each $e \in V$ when being observed, a probability for e to be put into set i of $\mathbf{s}$ is derived from:

$$d_i = \frac{F(\mathbf{s} \sqcup \langle e, i\rangle)}{1-\epsilon} - \frac{F(\mathbf{s})}{1+\epsilon}$$

The factor $\frac{1}{1-\epsilon}$ of $F(\mathbf{s} \sqcup \langle e, i\rangle)$ and $\frac{1}{1+\epsilon}$ of $F(\mathbf{s})$ is to cover the worst case that $F(\mathbf{s} \sqcup \langle e, i\rangle) < F(\mathbf{s})$ even f is monotone increasing. To filter out elements with insignificant contribution, RSTREAM ignores $\langle e, i\rangle$ that $d_i < \frac{o}{M}$, i.e. $d_i = 0$ if $d_i < \frac{o}{M}$, otherwise d_i keeps its value. Then, the probability for e to be added into set i is computed as:

$$\Pr[\langle e, i\rangle \text{ is added to } \mathbf{s}] = d_i^{T-1} \Big/ \sum_{j\in[k]} d_j^{T-1}$$

where $T = |\{j : d_j \ge \frac{o}{M}\}|$. Intuitively, T is for scaling up placements with high d_i to be likely selected.

However, the algorithm still struggles that ϵ could make d_i much larger than $\Delta_{e,i} f(\mathbf{s})$. For example, if $F(\mathbf{s}) \approx f(\mathbf{s}) = f(\mathbf{s}\sqcup\langle e, i\rangle) \approx F(\mathbf{s}\sqcup\langle e, i\rangle)$, then $\frac{1}{1-\epsilon}F(\mathbf{s}\sqcup\langle e, i\rangle) - \frac{1}{1+\epsilon}F(\mathbf{s}) \approx \frac{2\epsilon}{1-\epsilon^2}f(\mathbf{s})$, which could exceed $\frac{o}{M}$ and $\langle e, i\rangle$ has chance to be added into $\mathbf{s}$ even $\Delta_{e,i} f(\mathbf{s}) = 0$.

To tackle such challenges, RSTREAM proposes a *denoise* step, introducing a parameter η whose role is to fragment ϵ into thresholds $\epsilon' = \epsilon$, $\frac{(\eta-2)\epsilon}{\eta-1}$, $\frac{(\eta-3)\epsilon}{\eta-1}$, ...0 and treat F as an ϵ'-estimate of f. Note that, with $\eta = 1$, RSTREAM only considers $\epsilon' = \epsilon$. RSTREAM then runs multiple copies, each corresponding to a hypothesis that F is an ϵ'-estimate of

f and produces a candidate solution for each ϵ'. Finally, RSTREAM returns the best solution obtained.

The full RSTREAM's pseudocode is presented in Alg. 3. In Alg. 3, $t_{j,\epsilon'}$ and $\{\mathbf{s}^i_{j,\epsilon'}\}$ keep track the number of selected elements and k-sets produced by RSTREAM when o is estimated by $M(1+\gamma)^j$ and assuming that F is ϵ'-estimate of f. There is a slight difference to DSTREAM on the way RSTREAM lazily estimates Δ_u. To be specific, $\Delta_u = \frac{(1+\epsilon)^2+4\epsilon B}{(1-\epsilon^2)(1-\epsilon)} \max_{e\in V_p, i\in[k]} F(\langle e,i\rangle)$ to guarantee $\frac{F(\mathbf{s}\sqcup\langle e,i\rangle)}{1-\epsilon} - \frac{F(\mathbf{s})}{1+\epsilon} \leq \Delta_u$ for all $e \in V_p, i \in [k]$ and $\mathbf{s} \in (1+k)^V$ that $supp(\mathbf{s}) \subseteq V_p$ and $|\mathbf{s}| \leq B$.

Theorem 3 shows RSTREAM's performance guarantee when f is monotone. Detail proofs are provided in Appendix D.1.

Theorem 3. Given an instance of Noisy **MkSC** with input V, k, B, γ, M and F is an ϵ-estimate of the k-submodular objective function f. RSTREAM has query complexity of $O\big(\frac{\eta|V|k}{\gamma}\log(\frac{(1+\gamma)((1+\epsilon)^2+4\epsilon B)}{(1-\epsilon)^2}BM)\big)$ and takes $O\big(\frac{\eta B}{\gamma}\log(\frac{(1+\gamma)((1+\epsilon)^2+4\epsilon B)}{(1-\epsilon)^2}BM)\big)$ memory. If f is **monotone**, then $\frac{1+\epsilon}{1-\epsilon}\min_{x\in(1,M]}\max(\alpha(x),\beta(x))\mathrm{E}[f(\mathbf{s})] \geq f(\mathbf{o})$ where $\mathbf{s}$ is an output of RSTREAM and:

$$\alpha(x) = (1+\epsilon+2B\epsilon)(1+\gamma)\,x\,/\,(1-\epsilon)$$
$$\beta(x) = \Big(\frac{(1+\epsilon)^2+4B\epsilon}{1-\epsilon^2}\big(1-\frac{1}{k}\big)+1\Big)\frac{kx}{kx-k-1}$$

Proof overview. Without loss of generality, we abuse the notation and simply write t and $\mathbf{s}^i$ to indicate $t_{j,\epsilon'}$ and $\mathbf{s}^i_{j,\epsilon'}$ when $\epsilon' = \epsilon$ and $o = M(1+\gamma)^j$ that $f(\mathbf{o}) \geq oB \geq \frac{f(\mathbf{o})}{1+\gamma}$.

The key of our proof is to show that:

$$\max\big(\alpha(M),\beta(M)\big)\mathrm{E}[f(\mathbf{s}^t)] \geq f(\mathbf{o}) \tag{2}$$

Then, the approximation ratio of RSTREAM can be trivially inferred as in DSTREAM proof.

Equ. 2 is proven by using Lemma 1 and Lemma 2.

Lemma 1. If $t = B$ then $\alpha(M)\ f(\mathbf{s}^t) \geq f(\mathbf{o})$

To prove lemma 1, we expand $F(\mathbf{s}^t)$ to sum of sequences of $F(\mathbf{s}^i) - F(\mathbf{s}^{i-1})$ for $i = 0 \to t$ and utilize the filter that $\frac{F(\mathbf{s}^i)}{1-\epsilon} - \frac{F(\mathbf{s}^{i-1})}{1+\epsilon} \geq \frac{o}{M}$.

Lemma 2. If $t < B$ then $\beta(M)\ \mathrm{E}[f(\mathbf{s}^t)] \geq f(\mathbf{o})$.

Define $\mathbf{o}^j = (\mathbf{o} \sqcup \mathbf{s}^j) \sqcup \mathbf{s}^j$ for $i = 0 \to t$. The key proof of lemma 2 comes from the following claims.

Claim 3. For all $j \in \{1,...,t\}$, $f(\mathbf{o}^{j-1}) - \mathrm{E}[f(\mathbf{o}^j)] \leq \big(1-\frac{1}{k}\big)\big(\frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})\big) + \frac{o}{kM}$

In general, claim 3 is proven by considering all cases of relation between $\mathbf{o}^j$ and $\mathbf{o}^{j-1}$, and novelly applying AM-GM theorem (Hirschhorn, 2007) to bound $f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j)$ in term of $f(\mathbf{s}^j)$ and $f(\mathbf{s}^{j-1})$, in which scaling d_i by exponent T plays a critical role. Claim 3 allows us to obtain a key relation between $\mathbf{o}$ and $\mathbf{o}^t$ as follows.

Claim 4.

$$f(\mathbf{o}) - \mathrm{E}[f(\mathbf{o}^t)] \leq \big(1-\tfrac{1}{k}\big)\tfrac{(1+\epsilon)^2+4B\epsilon}{1-\epsilon^2}\mathrm{E}[f(\mathbf{s}^t)] + \tfrac{1}{kM}f(\mathbf{o})$$

Furthermore, as $\mathbf{s} \sqsubseteq \mathbf{o}^t$, we have the following claim

Claim 5. $\mathrm{E}[f(\mathbf{o}^t)] - \mathrm{E}[f(\mathbf{s}^t)] \leq \frac{1}{M}f(\mathbf{o})$

Lemma 2 follows by combining claims 4 and 5.

Memory and query complexity of RSTREAM can be trivially inferred as in the proof of DSTREAM. □

Theorem 4 shows RSTREAM's ratio when f is **non-monotone**. Detail proof is provided in Appendix. D.2.

Theorem 4. Given an instance of Noisy **MkSC** with input V, k, B, γ, M and F is an ϵ-estimate of the k-submodular objective function f. If f is **non-monotone**, then $\frac{1+\epsilon}{1-\epsilon}\min_{x\in(1,M]}\max(\alpha(x),\beta(x))\mathrm{E}[f(\mathbf{s})] \geq f(\mathbf{o})$, where $\mathbf{s}$ is an output of RSTREAM, $\mathbf{o}$ is an optimal solution

$$\alpha(x) = (1+\epsilon+2B\epsilon)(1+\gamma)x\,/\,(1-\epsilon)$$
$$\beta(x) = \frac{(3k-2)(1+\epsilon)^2+(8k-8)\epsilon B}{(1-\epsilon)^2}\,\frac{x}{kx-k-2}$$

5. Experimental Evaluation

In this section, we compare our two algorithms with existing methods on two applications of Noisy **MkSC**[1]. In general, our algorithms return solutions approximately close to that of Greedy, the current best **non-streaming** algorithm for **MkSC**, and outperform any other algorithms in term of number of queries. We further investigate the trade-off between quality of solution and number of queries of our algorithms with different parameter settings.

5.1. Influence Maximization with k Topics

In this problem, a social network is modeled under an directed graph $G = (V, E)$ where V is a set of social users; and each edge $(u, v) \in E$ $(u, v \in V)$ is associated with weights $\{w^i_{u,v}\}_{i\in[k]}$, where $w^i_{u,v}$ represents the strength of influence from user u to v on topic i.

We use Linear Threshold (LT) Model (Kempe et al., 2003) to model a diffusion process of a topic. The process is as

[1]The source code is available at `https://github.com/lannn2410/streamingksubmodular`

Algorithm 3 RSTREAM

Input $F, k, B, M > 1, \gamma > 0, \eta \geq 1$

$\Delta_u = \Delta_l = \Delta = 0$; $t_{j,\epsilon'} = 0 \; \forall j \in \mathbb{Z}^+, \epsilon' \in \mathbb{R}^+$
for each e in V **do**
  $\Delta = \max\left(\Delta, \max_{j\in[k]} F(\langle e, j\rangle)\right)$
  $\Delta_u = \frac{(1+\epsilon)^2+4\epsilon B}{(1-\epsilon^2)(1-\epsilon)}\Delta$; $\Delta_l = \Delta/\big((1+\epsilon)(1+\gamma)\big)$
  $O = \{(1+\gamma)^j \mid \frac{\Delta_l}{B\cdot M} \leq (1+\gamma)^j \leq \Delta_u\}$
  for each j that $(1+\gamma)^j \in O$ **do**
    $o = M(1+\gamma)^j$
    for each $\epsilon' = \epsilon, \frac{(\eta-2)\epsilon}{\eta-1}, \frac{(\eta-3)\epsilon}{\eta-1}, ...0$ **do**
      if $t_{j,\epsilon'} < B$ **then**
        for each $i \in [k]$ **do**
          $d_i = \frac{F(\mathbf{s}_{j,\epsilon'}^{t_{j,\epsilon'}} \sqcup \langle e,i\rangle)}{1-\epsilon'} - \frac{F(\mathbf{s}_{j,\epsilon'}^{t_{j,\epsilon'}})}{1+\epsilon'}$
          $d_i = 0$ if $d_i < \frac{o}{M}$, d_i otherwise
        $T =$ no. d_i that $d_i > 0$
        $D = \sum_{i\in[k]} d_i^{T-1}$
        if $t_{j,\epsilon'} < B$ and $T > 0$ **then**
          if $T = 1$ **then**
            $i =$ the only one that $d_i > 0$
          else
            $i =$ selected with prob. d_i^{T-1}/D
          $\mathbf{s}_{j,\epsilon'}^{t_{j,\epsilon'}+1} = \mathbf{s}_{j,\epsilon'}^{t_{j,\epsilon'}} \sqcup \langle e, i\rangle$
          $t_{j,\epsilon'} = t_{j,\epsilon'} + 1$

Return $argmax_{\mathbf{s}_{j,\epsilon'}^{t_{j,\epsilon'}}, j\in O} F(\mathbf{s}_{j,\epsilon'}^{t_{j,\epsilon'}})$ if f is **monotone**;
$argmax_{\mathbf{s}_{j,\epsilon'}^{i} | i < t_{j,\epsilon'}, j\in O} F(\mathbf{s}_{j,\epsilon'}^{i})$ if f is **non-monotone**;

follows: Given a seed k-set $\mathbf{s} = \{S_1, ...S_k\} \in (k+1)^V$, at first all users in S_i become active by i. Each $v \in V$ has a threshold θ_v^i chosen uniformly at random in $[0, 1]$ and v becomes active by i if $\sum_{\text{active } u} w_{u,v}^i \geq \theta_v^i$. The diffusion process of a topic is independent from other topics.

The problem's objective is to maximize an expected number of users who eventually become active in at least one of k diffusion processes given a seed k-set $\mathbf{s}$. To be specific, the objective function is $\mathbb{I}(\mathbf{s}) = \mathrm{E}\big[| \cup_{i\in[k]} \mathcal{D}_i(S_i)|\big]$ where $\mathcal{D}_i(S_i)$ is a random variable representing the set of active users in the diffusion process of topic i with seed S_i. The problem asks for $\mathbf{s}$ subject to $|\mathbf{s}| \leq B$ that maximizes $\mathbb{I}(\mathbf{s})$.

Ohsaka & Yoshida (2015) have shown that $\mathbb{I}(\cdot)$ under Independent Cascade model is monotone k-submodular. It is trivial that $\mathbb{I}(\cdot)$ under LT Model is also monotone k-submodular.

However, exactly computing $\mathbb{I}(\mathbf{s})$ is extremely expensive. Kempe et al. (2003) has shown that computing influence spread of a single topic is #P-hard already. Therefore, to reduce computational cost, it is common to estimate $\mathbb{I}(\mathbf{s})$ using sampling, although it comes with a cost on estimate error. And in this paper context, that error is represented by ϵ. Sampling detail is presented in Appendix E.

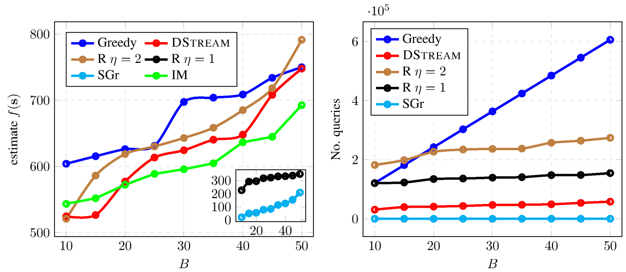
Settings. We use Facebook dataset from SNAP database (Leskovec & Krevl, 2014), an undirected graph with 4,039 nodes and 88,234 edges. Since it is undirected, we treat each edge as two directed edges. Weights $\{w_{u,v}^i\}_{i\in[k]}$ of the directed edge (u, v) is randomly shuffled from values of $\{\frac{1}{k\cdot d_v}, \frac{2}{k\cdot d_v}, \dots \frac{k}{k\cdot d_v}\}$ where d_v is in-degree of v. We set $k = 3, \epsilon = 0.5$ and compare our algorithms with the following:

- Greedy - the current best algorithm for **MkSC** in a non-streaming setting. In Appendix B, we prove that Greedy is able to achieve a ratio of $\frac{2+2\epsilon B}{1-\epsilon}$ for monotone Noisy **MkSC**. Since we can only obtain ϵ-estimate of $\mathbb{I}(\cdot)$, it is impractical to apply the lazy greedy method (Minoux, 1978) to reduce number of queries.
- Influence Maximization (IM) with a single topic. With $k = 1$, this problem becomes the classical IM problem which has been investigated extensively under several variants (Nguyen et al., 2016a; Dinh et al., 2014; Zhang et al., 2016; Shen et al., 2012; Zhang et al., 2014; Nguyen et al., 2013; Li et al., 2017; Nguyen et al., 2019). To apply an IM algorithm into Noisy **MkSC**, we first randomly select a topic i and run the algorithm to find k nodes initially adopting i in order to maximize the number of activated users. We use the SSA algorithm (Nguyen et al., 2016b; 2018) to solve IM.
- Streaming Greedy (SGr) - a simple streaming heuristic. SGr takes a single scan over V and for each $e \in V$ when being observed, if the size constraint is not violated, the algorithm picks e with probability $\frac{B}{|V|}$ and puts e into a set i of $\mathbf{s}$ that maximizes $F(\mathbf{s} \sqcup \langle e, i\rangle)$.

Although we have shown how to obtain values of M, γ that our algorithms can get their best ratio, we still varied values of M and γ to show a trade-off in their performance between solution quality and the number of queries. To be specific, we vary M between $3, 4, 5$ and γ between $0.5, 1.0, 1.5$. Elements in V are observed in random order. We compare algorithms' performance in various values of B. Results are averaged over 3 repetitions.

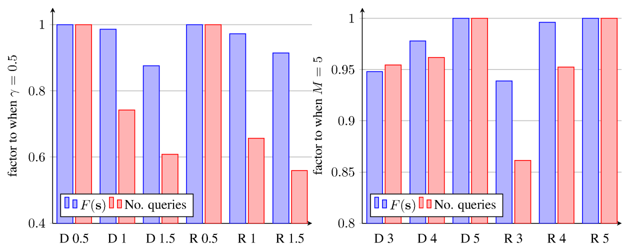
Results. Figure 1 shows a comparison of our two streaming algorithms ($M = 3$ and $\gamma = 1$) with Greedy and IM in term of quality of solutions and number of queries. We also measure RSTREAM's performances with $\eta = 2$ and $\eta = 1$ (no denoise step). With quality of solutions (Fig. 1a), DSTREAM and RSTREAM ($\eta = 2$) outperformed IM in almost cases and totally dominated SGr. Moreover, RSTREAM ($\eta = 2$) was able to perform approximately to Greedy and even outperformed Greedy with $B = 50$.

Figure 1b shows the number of queries of our algorithms versus Greedy. Since SSA worked in a manner that samples



(a) Quality of solution (b) No. queries

Figure 1. Comparison in Influence Maximization with k topics. For shorter legends, R stands for RSTREAM.



(a) Tradeoff w.r.t γ (b) Tradeoff w.r.t M

Figure 2. Algorithms' tradeoff. Labels in x-axis represented algorithms' initials and their parameter values, i.e. D for DSTREAM, R for RSTREAM. In Fig. 2a, D 0.5 means DSTREAM with $\gamma = 0.5$. In Fig. 2b, R 5 means RSTREAM with $M = 5$

a sufficient large number of graph realization and solve Max-Coverage on top of samples, it is unfair and impractical to measure and compare its number of queries with other algorithms. Therefore, there is no plot of IM in Fig. 1b. Except when $B \leq 20$, it is easy to see that our two streaming algorithms total outperformed Greedy by a huge margin in term of number of queries. For example, with $B = 50$, Greedy takes 2 times more queries than RSTREAM with $\eta = 2$ and 6 times more than DSTREAM.

Fig. 1 also shows the important role of the denoise step in RSTREAM. RSTREAM without denoise ($\eta = 1$) returned the worst solutions comparing to any other algorithms except SGr. That was due to ϵ mislead RSTREAM's estimation of f's marginal gain. Thus RSTREAM tends to add elements who have no contribution to $f(\mathbf{s})$ and prematurely reaches budget B before finishing passing V.

With the presence of the denoise step in RSTREAM, RSTREAM performed better than DSTREAM in term of solution quality but takes approximately 4 times more queries. That was due to: (1) RSTREAM is less likely to prematurely reach B; and (2) Δ_u of DSTREAM is much smaller than the one of RSTREAM, so DSTREAM needs fewer $(1+\gamma)^j$'s values than RSTREAM to estimate the optimal solution. That is also reflected in the two algorithms' query complexity.

We further investigate our two streaming algorithms' performance with different settings. Figure 2a shows their performance with $M = 4$ and $B = 20$. The tradeoff between solution quality and number of queries can easily be seen in both algorithm. For example, with DSTREAM, the solution quality drops by 13% for $\gamma = 0.5$ to 1.5 but the number of queries decreases by almost 40% . That can be explained intuitively by the fact that larger γ means fewer values of j that satisfy $\frac{\Delta_l}{B \cdot M} \leq (1+\gamma)^j \leq \Delta_u$. Thus the algorithms take fewer queries but acquire lower solution quality, which is also reflected by their approximation ratio.

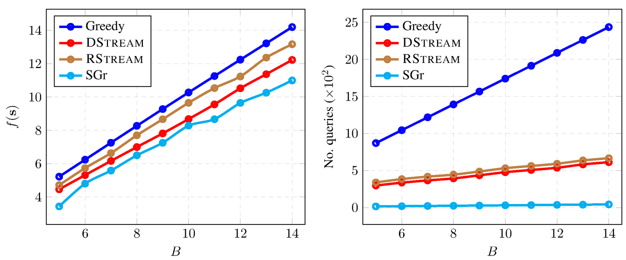
It looks like M does not impact much on the two streaming algorithms' performance. Figure 2b shows their measurements with $\gamma = 1.0$ and $B = 50$. Both algorithms' results show that increasing M slightly improves solution quality but also takes a few more queries. That is also reflected theoretically by their approximation ratio - the ratio with larger value of M should be at least better than the ratio with smaller one. In term of number of queries, increasing M reduces $\frac{\Delta_l}{B \cdot M}$, thus the algorithms have to consider a few more values of j that $\frac{\Delta_l}{B \cdot M} \leq (1+\gamma)^j \leq \Delta_u$.

5.2. Sensor placement with k types of measures.

In this part, we study a problem: given k types of sensors for different measures and a set V of n locations, each of which can be instrumented with exactly one sensor. The problem asks for B locations with their types of sensor placement (represented by s) that maximizes the *entropy* of s. In another words, denote X_e^i as the random variable representing the observation collected from a sensor of kind i if it is placed at location e. Then the problem asks for $\mathbf{s} \in (k+1)^V$ that maximizes $f(\mathbf{s}) = H\big(\cup_{e \in supp(\mathbf{s})} \{X_e^{\mathbf{s}(e)}\}\big)$ subject to $|\mathbf{s}| \leq B$, where $H(\cdot)$ is entropy function. Ohsaka & Yoshida (2015); Qian et al. (2017a) have proven that f is monotone k-submodular.

Settings. We re-implemented exact settings as in Ohsaka & Yoshida (2015). We also use Intel Lab dataset (Bodik et al., 2004), containing a log of readings collected from 58 sensors deployed in the Intel Berkeley research lab between February 28th and April 5th, 2004. Temperature, humidity and light values are extracted from each reading and discretized into bins of 2 degrees Celsius each, 5 points each and 100 luxes each, respectively. Since f can be computed exactly in polynomial number of computations, $\epsilon = 0$ in this experiment and the denoise step is no more necessary, thus $\eta = 1$ for RSTREAM. Again, we compare our algorithms with SGr and Greedy.

Results. Fig. 3 shows the entropy achieved by the algorithms with $M = 3$ and $\gamma = 0.5$. We observe the same pattern as in Influence Maximization experiments that our



(a) Quality of solution (b) No. queries

Figure 3. Comparison in Sensor Placement

algorithms performed closely to Greedy in term of solution quality but totally outperformed Greedy in the number of queries. An only difference is that RSTREAM now takes only a slight more queries than DSTREAM. That is due to RSTREAM and DSTREAM have a same value of Δ_u with $\epsilon = 0$; and a k-set constructed by RSTREAM would also satisfy DSTREAM's selection condition but not vice versa.

6. Conclusion

In this work, we propose two novel streaming algorithms for maximizing a noisy k-submodular function subject to size constraint; and investigate their theoretical performance guarantees. Experimental results have shown that our algorithms require a much smaller number of queries than that of the state-of-the-art non-streaming algorithm, while returning comparable solutions in term of quality.

Acknowledgements

This work was supported in part by the National Science Foundation (NSF) grants IIS-1908594, CNS-1814614, and the University of Florida Informatics Institute Fellowship Program. We would like to thank the anonymous reviewers for their helpful feedback.

References

Badanidiyuru, A., Mirzasoleiman, B., Karbasi, A., and Krause, A. Streaming submodular maximization: Massive data summarization on the fly. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 671–680. ACM, 2014.

Bateni, M., Esfandiari, H., and Mirrokni, V. Almost optimal streaming algorithms for coverage problems. In *Proceedings of the 29th ACM Symposium on Parallelism in Algorithms and Architectures*, pp. 13–23. ACM, 2017.

Bodik, P., Hong, W., Guestrin, C., Madden, S., Paskin, M., and Thibaux, R. Intel lab data. http://db.csail.mit.edu/labdata/labdata.html, 2004.

Borgs, C., Brautbar, M., Chayes, J., and Lucier, B. Maximizing social influence in nearly optimal time. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pp. 946–957. SIAM, 2014.

Crawford, V. G., Kuhnle, A., and Thai, M. T. Submodular cost submodular cover with an approximate oracle. In *International Conference on Machine Learning*, 2019.

Dinh, T. N., Shen, Y., Nguyen, D. T., and Thai, M. T. On the approximability of positive influence dominating set in social networks. *Journal of Combinatorial Optimization*, 27(3):487–503, 2014.

Gomes, R. and Krause, A. Budgeted nonparametric learning from data streams. In *ICML*, volume 1, pp. 3, 2010.

Hassidim, A. and Singer, Y. Submodular optimization under noise. In *Conference on Learning Theory*, pp. 1069–1122, 2017.

Hirschhorn, M. D. The am-gm inequality. *Mathematical Intelligencer*, 29(4):7–7, 2007.

Horel, T. and Singer, Y. Maximization of approximately submodular functions. In *Advances in Neural Information Processing Systems*, pp. 3045–3053, 2016.

Iwata, S., Tanigawa, S.-i., and Yoshida, Y. Improved approximation algorithms for k-submodular function maximization. In *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*, pp. 404–413. Society for Industrial and Applied Mathematics, 2016.

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

Kumar, R., Moseley, B., Vassilvitskii, S., and Vattani, A. Fast greedy algorithms in mapreduce and streaming. *ACM Transactions on Parallel Computing (TOPC)*, 2(3):14, 2015.

Leskovec, J. and Krevl, A. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

Li, X., Smith, J. D., Dinh, T. N., and Thai, M. T. Why approximate when you can get the exact? optimal targeted viral marketing at scale. In *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*, pp. 1–9. IEEE, 2017.

Minoux, M. Accelerated greedy algorithms for maximizing submodular set functions. In *Optimization techniques*, pp. 234–243. Springer, 1978.

Nguyen, D. T., Zhang, H., Das, S., Thai, M. T., and Dinh, T. N. Least cost influence in multiplex social networks: Model representation and analysis. In *2013 IEEE 13th International Conference on Data Mining*, pp. 567–576. IEEE, 2013.

Nguyen, H. T., Dinh, T. N., and Thai, M. T. Cost-aware targeted viral marketing in billion-scale networks. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pp. 1–9. IEEE, 2016a.

Nguyen, H. T., Thai, M. T., and Dinh, T. N. Stop-and-stare: Optimal sampling algorithms for viral marketing in billion-scale networks. In *Proceedings of the 2016 International Conference on Management of Data*, pp. 695–710. ACM, 2016b.

Nguyen, H. T., Dinh, T. N., and Thai, M. T. Revisiting of 'revisiting the stop-and-stare algorithms for influence maximization'. In *International Conference on Computational Social Networks*, pp. 273–285. Springer, 2018.

Nguyen, L. N., Zhou, K., and Thai, M. T. Influence maximization at community level: A new challenge with non-submodularity. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 327–337. IEEE, 2019.

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

Oshima, H. Derandomization for k-submodular maximization. In *International Workshop on Combinatorial Algorithms*, pp. 88–99. Springer, 2017.

Qian, C. Distributed pareto optimization for large-scale noisy subset selection. *IEEE Transactions on Evolutionary Computation*, 2019.

Qian, C., Yu, Y., and Zhou, Z.-H. Subset selection by pareto optimization. In *Advances in Neural Information Processing Systems*, pp. 1774–1782, 2015.

Qian, C., Shi, J.-C., Tang, K., and Zhou, Z.-H. Constrained monotone k-submodular function maximization using multiobjective evolutionary algorithms with theoretical guarantee. *IEEE Transactions on Evolutionary Computation*, 22(4):595–608, 2017a.

Qian, C., Shi, J.-C., Yu, Y., Tang, K., and Zhou, Z.-H. Subset selection under noise. In *Advances in neural information processing systems*, pp. 3560–3570, 2017b.

Sakaue, S. On maximizing a monotone k-submodular function subject to a matroid constraint. *Discrete Optimization*, 23:105–113, 2017.

Shen, Y., Dinh, T. N., Zhang, H., and Thai, M. T. Interest-matching information propagation in multiple online social networks. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pp. 1824–1828, 2012.

Singer, Y. and Hassidim, A. Optimization for approximate submodularity. In *Advances in Neural Information Processing Systems*, pp. 396–407, 2018.

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

Soma, T. No-regret algorithms for online k-submodular maximization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1205–1214, 2019.

Ward, J. and Živnỳ, S. Maximizing bisubmodular and k-submodular functions. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1468–1481. Society for Industrial and Applied Mathematics, 2014.

Ward, J. and Živnỳ, S. Maximizing k-submodular functions and beyond. *ACM Transactions on Algorithms (TALG)*, 12(4):47, 2016.

Yang, R., Xu, D., Cheng, Y., Gao, C., and Du, D.-Z. Streaming submodular maximization under noises. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 348–357. IEEE, 2019.

Zhang, H., Mishra, S., Thai, M. T., Wu, J., and Wang, Y. Recent advances in information diffusion and influence maximization in complex social networks. *Opportunistic Mobile Social Networks*, 37(1.1):37, 2014.

Zhang, H., Zhang, H., Kuhnle, A., and Thai, M. T. Profit maximization for multiple products in online social networks. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pp. 1–9. IEEE, 2016.

Zhou, Z.-H., Yu, Y., and Qian, C. *Evolutionary Learning: Advances in Theories and Algorithms*. Springer, 2019.

A. Organization of the Appendix

Appendix B describes the Greedy algorithm to solve Noisy **M**k**SC** and its approximation ratio.

Appendix C provides omitted proofs from Section 3.

Appendix D provides omitted proofs from Section 4.

Appendix E presents details of the sampling method for Influence Maximization with k topics.

B. Greedy Algorithm

In this part, we investigate performance guarantee of Greedy algorithm. Greedy has been proven to obtain an approximation ratio of 2 for **M**k**SC** with monotone non-noisy objective function (Ohsaka & Yoshida, 2015). We extend the authors' proof to **M**k**SC** under noise and show that Greedy is able to obtain approximation of $\frac{2+2\epsilon B}{1-\epsilon}$ when f is monotone and $\frac{(2+2\epsilon+4\epsilon B)(1+\epsilon)}{(1-\epsilon)^2}+1$ in case of non-monotonicity.

The pseudo code of Greedy is presented by Alg. 4

Algorithm 4 Greedy Algorithm

Input F, k, B

1: $\mathbf{s}^0 = \{\emptyset, \emptyset, ...\emptyset\}$
2: **for** $t = 1 \to B$ **do**
3: $\quad e, i = argmax_{e\in V; i\in[k]} F(\mathbf{s}^{t-1} \sqcup \langle e, i\rangle)$
4: $\quad \mathbf{s}^t = \mathbf{s}^{t-1} \sqcup \langle e, i\rangle$

Return $\mathbf{s}^B$ if f is **monotone**; $argmax_{\mathbf{s}^i; i\in\{1,\ldots,B\}} F(\mathbf{s}^i)$ if f is **non-monotone**.

Theorem 5. Given an instance of Noisy **M**k**SC** with input V, k, B and F is an ϵ-estimation of the **monotone** k-submodular objective function f. If $\mathbf{s}$ is an output of the Greedy algorithm and $\mathbf{o}$ is an optimal solution, then

$$f(\mathbf{o}) \leq \frac{2+2\epsilon B}{1-\epsilon} f(\mathbf{s}) \tag{3}$$

Proof. Let e^j and i^j be a selection in iteration j of the Greedy algorithm, we construct a sequence $\{\mathbf{o}^j\}$ as follows:

- $\mathbf{o}^0 = \mathbf{o}$
- With $j > 0$, let $S^j = supp(\mathbf{o}^{j-1}) \setminus supp(\mathbf{s}^{j-1})$. Let $o^j = e^j$ if $e^j \in S^j$, otherwise let o^j to be an arbitrary element in S^j.
  - Define $\mathbf{o}^{j-1/2}$ as a k-set that $\mathbf{o}^{j-1/2}(e) = \mathbf{o}^{j-1}(e)\ \forall e \in V \setminus \{o^j\}$ and $\mathbf{o}^{j-1/2}(o^j) = 0$.
  - Define $\mathbf{o}^j$ as a k-set that $\mathbf{o}^j(e) = \mathbf{o}^{j-1/2}(e)\ \forall e \in V \setminus \{e^j\}$ and $\mathbf{o}^j(e^j) = i^j$
  - Define $\mathbf{s}^{j-1/2}$ as a k-set that $\mathbf{s}^{j-1/2}(e) = \mathbf{s}^{j-1}(e)\ \forall e \in V \setminus \{o^j\}$ and $\mathbf{s}^{j-1/2}(o^j) = i^j$.

It is trivial that $\mathbf{o}^B = \mathbf{s}^B$. We have:

$$\begin{aligned}
f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) &\leq f(\mathbf{o}^{j-1}) - f(\mathbf{o}^{j-1/2}) && (4)\\
&\leq f(\mathbf{s}^{j-1/2}) - f(\mathbf{s}^{j-1}) && (5)\\
&\leq \frac{1}{1-\epsilon} F(\mathbf{s}^{j-1/2}) - f(\mathbf{s}^{j-1}) && (6)\\
&\leq \frac{1}{1-\epsilon} F(\mathbf{s}^{j}) - f(\mathbf{s}^{j-1}) && (7)\\
&\leq \frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^{j}) - f(\mathbf{s}^{j-1}) && (8)
\end{aligned}$$

The inequality (4) is due to f is monotone, thus $f(\mathbf{o}^j) \geq f(\mathbf{o}^{j-1/2})$. The inequality (5) is from k-submodularity of f. The inequality (7) is due to greedy selection. Therefore:

$$f(\mathbf{o}) - f(\mathbf{s}) = \sum_{j=1}^{B} \Big(f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) \Big) \leq \sum_{j=1}^{B} \Big(\frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^j) - f(\mathbf{s}^{j-1}) \Big)$$
$$\leq \frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}) + \sum_{j=1}^{B} \frac{2\epsilon}{1-\epsilon} f(\mathbf{s}^j) \leq \frac{1+\epsilon+2\epsilon B}{1-\epsilon} f(\mathbf{s})$$

Thus $f(\mathbf{o}) \leq \frac{2+2\epsilon B}{1-\epsilon} f(\mathbf{s})$, which completes the proof. $\square$

Theorem 6. Given an instance of Noisy **M**k**SC** with input V, k, B and F is an ϵ-estimation of the **non-monotone** k-submodular objective function f. If $\mathbf{s}$ is an output of the Greedy algorithm and $\mathbf{o}$ is an optimal solution, then

$$f(\mathbf{o}) \leq \frac{3+\epsilon+4\epsilon B}{1-\epsilon} f(\mathbf{s}) \tag{9}$$

Proof. We use the same definition of $\mathbf{o}^j, \mathbf{o}^{j-1/2}, \mathbf{s}^j$ as in proof of Theorem 5.

Although f is non-monotone, f is *pairwise-monotone* due to k-submodularity (Ward & Živnỳ, 2016). To be specific, given a k-set $\mathbf{x}$ and $e \notin supp(\mathbf{x})$, we have

$$\Delta_{e,i} f(\mathbf{x}) + \Delta_{e,j} f(\mathbf{x}) \geq 0 \; \forall i, j \in [k] \text{ and } i \neq j \tag{10}$$

Let consider a value of $j \in [1, B]$, due to pairwise-monotonicity, there should exist $i' \in [k]$ that $f(\mathbf{s}^{j-1} \sqcup \langle e^j, i' \rangle) \geq f(\mathbf{s}^{j-1})$. Moreover, due to greedy selection, $f(\mathbf{s}^{j-1} \sqcup \langle e^j, i' \rangle) \leq \frac{1}{1-\epsilon} F(\mathbf{s}^{j-1} \sqcup \langle e^j, i' \rangle) \leq \frac{1}{1-\epsilon} F(\mathbf{s}^j) \leq \frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^j)$. Thus $\frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^j) \geq f(\mathbf{s}^{j-1})$. We consider 2 following cases:

- $\mathbf{o}^j = \mathbf{o}^{j-1}$. In this case $f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) = 0 \leq \frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^j) - f(\mathbf{s}^{j-1})$
- $\mathbf{o}^j \neq \mathbf{o}^{j-1}$. There would be 2 sub-cases:
  - $\mathbf{o}^{j-1}(e^j) = 0$. Then let $i' \in [k]$ be an arbitrary number and $i' \neq i^j$. Then

$$f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) = f(\mathbf{o}^{j-1}) + f(\mathbf{o}^{j-1/2} \sqcup \langle e^j, i' \rangle) - 2f(\mathbf{o}^{j-1/2}) - \Big(f(\mathbf{o}^{j-1/2} \sqcup \langle e^j, i' \rangle) + f(\mathbf{o}^j) - 2f(\mathbf{o}^{j-1/2}) \Big) \tag{11}$$
$$\leq f(\mathbf{o}^{j-1}) + f(\mathbf{o}^{j-1/2} \sqcup \langle e^j, i' \rangle) - 2f(\mathbf{o}^{j-1/2}) \tag{12}$$
$$\leq f(\mathbf{s}^{j-1/2}) + f(\mathbf{s}^{j-1} \sqcup \langle e^j, i' \rangle) - 2f(\mathbf{s}^{j-1}) \tag{13}$$
$$\leq \frac{1}{1-\epsilon} F(\mathbf{s}^{j-1/2}) + \frac{1}{1-\epsilon} F(\mathbf{s}^{j-1} \sqcup \langle e^j, i' \rangle) - 2f(\mathbf{s}^{j-1}) \tag{14}$$
$$\leq \frac{2}{1-\epsilon} F(\mathbf{s}^j) - 2f(\mathbf{s}^{j-1}) \leq 2\frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^j) - 2f(\mathbf{s}^{j-1}) \tag{15}$$

  The inequality 12 is due to pairwise-monotonicity of f. The inequality 13 is from k-submodularity and inequality 15 is due to greedy selection.
  - $\mathbf{o}^{j-1}(e^j) \neq i^j$. Then:

$$f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) = 2f(\mathbf{o}^{j-1}) - 2f(\mathbf{o}^{j-1/2}) - \Big(f(\mathbf{o}^{j-1}) + f(\mathbf{o}^j) - 2f(\mathbf{o}^{j-1/2}) \Big)$$
$$\leq 2f(\mathbf{o}^{j-1}) - 2f(\mathbf{o}^{j-1/2}) \leq 2f(\mathbf{s}^{j-1/2}) - 2f(\mathbf{s}^{j-1}) \leq \frac{2}{1-\epsilon} F(\mathbf{s}^{j-1/2}) - 2f(\mathbf{s}^{j-1})$$
$$\leq \frac{2}{1-\epsilon} F(\mathbf{s}^j) - 2f(\mathbf{s}^{j-1}) \leq 2\frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^j) - 2f(\mathbf{s}^{j-1})$$

Both cases imply that $f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) \leq 2\frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^j) - 2f(\mathbf{s}^{j-1})$. Therefore:

$$f(\mathbf{o}) - f(\mathbf{s}) = \sum_{j=1}^{B} \Big(f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) \Big) \leq 2 \sum_{j=1}^{B} \Big(\frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^j) - f(\mathbf{s}^{j-1}) \Big)$$
$$\leq 2 \times \Big(\frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^B) + \sum_{j=1}^{B} \frac{2\epsilon}{1-\epsilon} f(\mathbf{s}^j) \Big) \leq \frac{2+2\epsilon+4\epsilon B}{1-\epsilon} \frac{1+\epsilon}{1-\epsilon} f(\mathbf{s})$$

Thus, $f(\mathbf{o}) \leq \big(\frac{(2+2\epsilon+4\epsilon B)(1+\epsilon)}{(1-\epsilon)^2} + 1 \big) f(\mathbf{s})$, which completes the proof. □

C. DSTREAM Algorithm

In this part, we present in detail omitted proofs of DSTREAM's approximation ratio. There would be 2 separate sub-parts: one is for when f is **monotone** - which we describe omitted proofs of claims in Proposition 1; the other is for when f is **non-monotone** - which we would fully prove the Theorem 2.

C.1. Approximation ratio of DSTREAM when f is monotone

Proof of Claim 1. Denote $\langle e^j, i^j \rangle$ as j-th addition of Alg. 1, i.e $\mathbf{s}^j = \mathbf{s}^{j-1} \sqcup \langle e^j, i^j \rangle$. Define:

$$\mathbf{o}^{j-1/2} = (\mathbf{o} \sqcup \mathbf{s}^j) \sqcup \mathbf{s}^{j-1}$$

Furthermore, define $\mathbf{s}^{j-1/2}$ as:

- If $e^j \in supp(\mathbf{o})$, let $i' = \mathbf{o}(e^j)$. Then $\mathbf{s}^{j-1/2} = \mathbf{s}^{j-1} \sqcup \langle e^j, i' \rangle$
- Otherwise, $\mathbf{s}^{j-1/2} = \mathbf{s}^{j-1}$

We have:

$$\begin{aligned} f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) &\leq f(\mathbf{o}^{j-1}) - f(\mathbf{o}^{j-1/2}) && (16)\\ &\leq f(\mathbf{s}^{j-1/2}) - f(\mathbf{s}^{j-1}) && (17)\\ &\leq \frac{1}{1-\epsilon} F(\mathbf{s}^{j-1/2}) - f(\mathbf{s}^{j-1}) && (18)\\ &\leq \frac{1}{1-\epsilon} F(\mathbf{s}^{j}) - f(\mathbf{s}^{j-1}) && (19)\\ &\leq \frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^{j}) - f(\mathbf{s}^{j-1}) && (20) \end{aligned}$$

The Inequality 16 is due to monotonicity of f; the inequality 17 is due to its k-submodularity and the inequality 19 is from selection of Alg. 1 that guarantees $i^j = argmax_{i\in[k]} F(\mathbf{s}^{j-1} \sqcup \langle e^j, i \rangle)$. Therefore:

$$f(\mathbf{o}) - f(\mathbf{o}^t) = \sum_{j=1}^{t} \Big(f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) \Big) \leq \sum_{j=1}^{t} \Big(\frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}^j) - f(\mathbf{s}^{j-1}) \Big)$$
$$= \frac{1+\epsilon}{1-\epsilon} f(\mathbf{s}) + \sum_{i=1}^{t-1} \Big(\frac{1+\epsilon}{1-\epsilon} - 1 \Big) f(\mathbf{s}^i) \leq \frac{1+\epsilon+2B\epsilon}{1-\epsilon} f(\mathbf{s})$$

which completes the proof.

Proof of Claim 2. As $\mathbf{s} \sqsubseteq \mathbf{o}^t$, denote $\{\langle u_1, j_1 \rangle, ... \langle u_r, j_r \rangle\}$ as a set of elements and their placement in $\mathbf{o}^t$ that are not in $\mathbf{s}$.

For each $\langle u_i, j_i \rangle$, denote $\mathbf{s}_i$ as $\mathbf{s}$ when Alg. 1 encounters u_i. As u_i was not added into $\mathbf{s}$, $\frac{F(\mathbf{s}_i \sqcup \langle u_i, j_i \rangle)}{1-\epsilon} \leq (|\mathbf{s}_i| + 1) \frac{o}{M}$. While $\frac{F(\mathbf{s}_i)}{1-\epsilon} \geq |\mathbf{s}_i| \frac{o}{M}$, we have:

$$\frac{F(\mathbf{s}_i \sqcup \langle u_i, j_i\rangle)}{1-\epsilon} - \frac{F(\mathbf{s}_i)}{1+\epsilon} \leq \frac{o}{M} + \frac{2\epsilon F(\mathbf{s}_i)}{1-\epsilon^2}$$

Denote $\mathbf{u}_i = \mathbf{s} \sqcup \{\langle u_1, j_1\rangle, ...\langle u_i, j_i\rangle\}$. We have

$$\begin{aligned}
f(\mathbf{o}^t) - f(\mathbf{s}) &= \sum_{i=1}^{B-t}\Big(f(\mathbf{u}_i) - f(\mathbf{u}_{i-1})\Big) \leq \sum_{i=1}^{B-t}\Big(f(\mathbf{s}_i \sqcup \langle u_i, j_i\rangle) - f(\mathbf{s}_i)\Big) \\
&\leq \sum_{i=1}^{B-t}\Big(\frac{1}{1-\epsilon}F(\mathbf{s}_i \sqcup \langle u_i, j_i\rangle) - \frac{1}{1+\epsilon}F(\mathbf{s}^i)\Big) \leq \sum_{i=1}^{B-t}\left(\frac{o}{M} + \frac{2\epsilon}{1-\epsilon^2}F(\mathbf{s}_i)\right) \leq \frac{1}{M}f(\mathbf{o}) + \frac{2\epsilon B}{1-\epsilon}f(\mathbf{s})
\end{aligned}$$

which complete the proof.

C.2. Approximation of DSTREAM when f is non-monotone

Proof of Theorem 2 We still use definition of $\mathbf{o}^j, \mathbf{o}^{j-1/2}, e^j, i^j, \mathbf{s}^j, \mathbf{s}^{j-1/2}$ as in the monotone proof. Also, for simplicity, we first prove the approximation ratio of Alg. 1 (assume $f(\mathbf{o})$ is known).

If $t = B$, $f(\mathbf{s}) \geq \frac{1}{1+\epsilon}\max_{i\in\{1,..,B\}} F(\mathbf{s}^i) \geq \frac{1}{1+\epsilon}F(\mathbf{s}^B) \geq \frac{1-\epsilon}{1+\epsilon}B\frac{o}{M} \geq \frac{1-\epsilon}{1+\epsilon}\frac{1}{(1+\gamma)M}f(\mathbf{o})$. The rest of the proof will focus on case $t < B$.

Due to pairwise-monotonicity of a k-submodular function, for any $j \in \{1, ..., t\}$, there exists no pair $i_1 \neq i_2 \in [k]$ that $\Delta_{e^j, i_1} f(\mathbf{s}^{j-1}) < 0$ and $\Delta_{e^j, i_2} f(\mathbf{s}^{j-1}) < 0$. Therefore $\max_{i\in[k]} f(\mathbf{s}^{j-1} \sqcup \langle e^j, i\rangle) \geq f(\mathbf{s}^{j-1})$. Thus

$$f(\mathbf{s}^j) \geq \frac{F(\mathbf{s}^j)}{1+\epsilon} = \frac{\max_{i\in[k]} F(\mathbf{s}^{j-1} \sqcup \langle e^j, i\rangle)}{1+\epsilon} \geq \frac{1-\epsilon}{1+\epsilon}\max_{i\in[k]} f(\mathbf{s}^{j-1} \sqcup \langle e^j, i\rangle) \geq \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1}) \tag{21}$$

Let's consider relation between $\mathbf{o}^{j-1}$ and $\mathbf{o}^j$, there are 2 cases:

- $|supp(\mathbf{o}^{j-1})| < |supp(\mathbf{o}^j)|$, which means $e^j \notin supp(\mathbf{o})$. We randomly pick $i' \in [k]$ and $i' \neq i^j$, we have
$$\begin{aligned}
f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) &= f(\mathbf{o}^{j-1} \sqcup \langle e^j, i'\rangle) - f(\mathbf{o}^{j-1}) - \big(f(\mathbf{o}^j) + f(\mathbf{o}^{j-1} \sqcup \langle e^j, i'\rangle) - 2f(\mathbf{o}^{j-1})\big) \\
&\leq f(\mathbf{o}^{j-1} \sqcup \langle e^j, i'\rangle) - f(\mathbf{o}^{j-1}) \leq f(\mathbf{s}^{j-1} \sqcup \langle e^j, i'\rangle) - f(\mathbf{s}^{j-1}) \\
&\leq \frac{1}{1-\epsilon}F(\mathbf{s}^{j-1} \sqcup \langle e^j, i'\rangle) - f(\mathbf{s}^{j-1}) \leq \frac{1}{1-\epsilon}F(\mathbf{s}^j) - f(\mathbf{s}^{j-1}) \\
&\leq \frac{1+\epsilon}{1-\epsilon}f(\mathbf{s}^j) - f(\mathbf{s}^{j-1})
\end{aligned}$$
- $|supp(\mathbf{o}^{j-1})| = |supp(\mathbf{o}^j)|$, which means $e^j \in supp(\mathbf{o})$. We have 2 sub-cases
  - $\mathbf{o}^{j-1}(e^j) = i^j$. Then $f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) = 0 \leq \frac{1+\epsilon}{1-\epsilon}f(\mathbf{s}^j) - f(\mathbf{s}^{j-1})$
  - $\mathbf{o}^{j-1}(e^j) \neq i^j$, we have
$$\begin{aligned}
f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) &= 2f(\mathbf{o}^{j-1}) - 2f(\mathbf{o}^{j-1/2}) - \big(f(\mathbf{o}^{j-1}) + f(\mathbf{o}^j) - 2(\mathbf{o}^{j-1/2})\big) \\
&\leq 2f(\mathbf{o}^{j-1}) - 2f(\mathbf{o}^{j-1/2}) \leq 2f(\mathbf{s}^{j-1/2}) - 2f(\mathbf{s}^{j-1}) \leq 2\frac{1+\epsilon}{1-\epsilon}f(\mathbf{s}^j) - 2f(\mathbf{s}^{j-1})
\end{aligned}$$

Therefore, in overall $f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) \leq 2\frac{1+\epsilon}{1-\epsilon}f(\mathbf{s}^j) - 2f(\mathbf{s}^{j-1})$. We have

$$f(\mathbf{o}) - f(\mathbf{o}^t) = \sum_{j=1}^{t}\Big(f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j)\Big) \leq \sum_{j=1}^{t}2\Big(\frac{1+\epsilon}{1-\epsilon}f(\mathbf{s}^j) - f(\mathbf{s}^{j-1})\Big) \tag{22}$$

$$= 2\frac{1+\epsilon}{1-\epsilon}f(\mathbf{s}^t) + 2\sum_{i=1}^{t-1}\frac{2\epsilon}{1-\epsilon}f(\mathbf{s}^i) \leq \frac{2+2\epsilon+4\epsilon B}{1-\epsilon}\max_{i\leq t} f(\mathbf{s}^i) \leq \frac{(1+\epsilon)(2+2\epsilon+4\epsilon B)}{(1-\epsilon)^2}f(\mathbf{s}) \tag{23}$$

Also, similar to monotone case, as $\mathbf{s}^t \sqsubseteq \mathbf{o}^t$, denote $\{\langle u_1, j_1\rangle, ...\langle u_r, j_r\rangle\}$ as a set of elements and their placement in $\mathbf{o}^t$ that are not in $\mathbf{s}^t$. For each $\langle u_i, j_i\rangle$, denote $\mathbf{s}_i$ as $\mathbf{s}^t$ when Alg. 1 encounters u_i. Denote $\mathbf{u}_i = \mathbf{s}^t \sqcup \{\langle u_1, j_1\rangle, ...\langle u_i, j_i\rangle\}$. We have

$$f(\mathbf{o}^t) - f(\mathbf{s}^t) = \sum_{i=1}^{B-t}\Big(f(\mathbf{u}^i) - f(\mathbf{u}^{i-1})\Big) \leq \sum_{i=1}^{B-t}\Big(f(\mathbf{s}_i \sqcup \langle u_i, j_i\rangle) - f(\mathbf{s}_i)\Big) \tag{24}$$

$$\leq \sum_{i=1}^{B-t}\left(\frac{F(\mathbf{s}_i \sqcup \langle u_i, j_i\rangle)}{1-\epsilon} - \frac{F(\mathbf{s}_i)}{1+\epsilon}\right) \leq \sum_{i=1}^{B-t}\left(\frac{o}{M} + \frac{2\epsilon}{1-\epsilon^2}F(\mathbf{s}^i)\right) \tag{25}$$

$$\leq \frac{1}{M}f(\mathbf{o}) + \frac{2\epsilon B}{1-\epsilon^2}F(\mathbf{s}) \leq \frac{1}{M}f(\mathbf{o}) + \frac{2\epsilon B}{1-\epsilon}f(\mathbf{s}) \tag{26}$$

Combining Equ. 23 and 26, we have

$$(1-\frac{1}{M})f(\mathbf{o}) \leq \Big(\frac{(1+\epsilon)(2+2\epsilon+4\epsilon B)}{(1-\epsilon)^2} + \frac{2\epsilon B}{1-\epsilon} + \frac{1+\epsilon}{1-\epsilon}\Big)f(\mathbf{s}) = \frac{3+4\epsilon+6\epsilon B+\epsilon^2+2\epsilon^2 B}{(1-\epsilon)^2}f(\mathbf{s})$$

The approximation ratio of DSTREAM when discarding assumption of known $f(\mathbf{o})$ is trivially follows as in the proof of the monotone case.

D. RSTREAM Algorithm

In this part, we present in detail omitted proofs of RSTREAM's approximation ratio. There would be 2 separate sub-parts: one is for when f is **monotone** - which we describe omitted proofs of claims in Theorem 3; the other is for when f is **non-monotone** - which we would fully prove the Theorem 4.

Note that in proofs related to RSTREAM, we abuse the notation and simply write t and $\mathbf{s}^i$ to indicate $t_{j,\epsilon'}$ and $\mathbf{s}^i_{j,\epsilon'}$ in Alg. 3 when o is estimated by $M(1+\gamma)^j$ that satisfies $f(\mathbf{o}) \geq oB \geq \frac{f(\mathbf{o})}{1+\gamma}$; and $\epsilon' = \epsilon$.

Similar to DSTREAM's proof, denote $\langle e^j, i^j\rangle$ as j-th addition of Alg. 3, i.e $\mathbf{s}^j = \mathbf{s}^{j-1} \sqcup \langle e^j, i^j\rangle$. Define:

$$\mathbf{o}^{j-1/2} = (\mathbf{o} \sqcup \mathbf{s}^j) \sqcup \mathbf{s}^{j-1}$$

Furthermore, define $\mathbf{s}^{j-1/2}$ as:

- If $e^j \in supp(\mathbf{o})$, let i' be an index of a set containing e^j in $\mathbf{o}$. Then $\mathbf{s}^{j-1/2} = \mathbf{s}^{j-1} \sqcup \langle e^j, i'\rangle$
- Otherwise, $\mathbf{s}^{j-1/2} = \mathbf{s}^{j-1}$

D.1. Approximation ratio of RSTREAM when f is monotone

Proof of Lemma 1 We have:

$$F(\mathbf{s}) = F(\mathbf{s}^B) = \sum_{i=1}^{B}\Big(F(\mathbf{s}^i) - F(\mathbf{s}^{i-1})\Big) = \sum_{i=1}^{B}\Big(F(\mathbf{s}^i) - \frac{1-\epsilon}{1+\epsilon}F(\mathbf{s}^{i-1})\Big) - \frac{2\epsilon}{1+\epsilon}\sum_{i}^{B-1}F(\mathbf{s}^i)$$
$$\geq (1-\epsilon)\sum_{i=1}^{B}\frac{o}{M} - 2\epsilon\sum_{i}^{B-1}f(\mathbf{s}^i) \geq \frac{(1-\epsilon)f(\mathbf{o})}{(1+\gamma)M} - 2\epsilon B f(\mathbf{s})$$

Therefore, $\frac{(1+\gamma)(1+\epsilon+2\epsilon B)M}{1-\epsilon}f(\mathbf{s}) \geq f(\mathbf{o})$

Proof of Claim 3 Let's consider Alg. 3 before adding e^j into $\mathbf{s}^{j-1}$. Denote $d_i = \frac{1}{1-\epsilon}F(\mathbf{s}^{j-1} \sqcup \langle e^j, i\rangle) - \frac{1}{1+\epsilon}F(\mathbf{s}^{j-1})$. We consider the following cases:

- $|supp(\mathbf{o}^{j-1})| = |supp(\mathbf{o}^j)|$, which means $e^j \in supp(\mathbf{o})$. Let $p = \mathbf{o}^{j-1}(e^j)$.

  Let $I \subseteq [k]$ be a set of values of i that $d_i > \frac{o}{M}, T = |I|$.

  We define $\mathbf{o}_i^j$ as a k-set that $\mathbf{o}_i^j(e) = \mathbf{o}^j(e)\ \forall e \in V \setminus \{e^j\}$ and $\mathbf{o}_i^j(e^j) = i$. Define $\mathbf{o}^{j-1/2}$ as a k-set that $\mathbf{o}^{j-1/2}(e) = \mathbf{o}^j(e)\ \forall e \in V \setminus \{e^j\}$ and $\mathbf{o}^{j-1/2}(e^j) = 0$.

  Let $\mathbf{s}_i^j = \mathbf{s}^{j-1} \sqcup \langle e^j, i\rangle$. We have two following sub-cases:

  - $d_p \geq \frac{o}{M}$, then if $T = 1$, $\mathbf{o}^{j-1} = \mathbf{o}^j$ and $f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j) = 0 \leq \left(1 - \frac{1}{k}\right)\left(\frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})\right)$. Thus, we assume $T > 1$, in which we have

$$
\begin{aligned}
f(\mathbf{o}^{j-1}) - \mathrm{E}[f(\mathbf{o}^j)] &= \frac{1}{D}\sum_{i \in I\setminus\{p\}} \Big(f(\mathbf{o}^{j-1}) - f(\mathbf{o}_i^j)\Big) d_i^{T-1} \leq \frac{1}{D}\sum_{i \in I\setminus\{p\}} \Big(f(\mathbf{o}^{j-1}) - f(\mathbf{o}^{j-1/2})\Big) d_i^{T-1} && (27)\\
&\leq \frac{1}{D}\sum_{i \in I\setminus\{p\}} \Big(f(\mathbf{s}_p^j) - f(\mathbf{s}^{j-1})\Big) d_i^{T-1} \leq \frac{1}{D}\sum_{i \in I\setminus\{p\}} d_p d_i^{T-1} && (28)\\
&\leq \frac{1}{D}\Big(1 - \frac{1}{T}\Big)\sum_{i \in I} d_i^T && (29)\\
&= \frac{1}{D}\Big(1 - \frac{1}{T}\Big)\sum_{i \in I}\Big(\frac{1}{1-\epsilon}F(\mathbf{s}_i^j) - \frac{1}{1+\epsilon}F(\mathbf{s}^{j-1})\Big) d_i^{T-1} && (30)\\
&\leq \frac{1}{D}\Big(1 - \frac{1}{T}\Big)\sum_{i \in I}\Big(\frac{1+\epsilon}{1-\epsilon}f(\mathbf{s}_i^j) - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})\Big) d_i^{T-1} && (31)\\
&\leq \Big(1 - \frac{1}{k}\Big)\Big(\frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})\Big) && (32)
\end{aligned}
$$

    Inequality 29 comes from AM-GM inequality, defined as:

    Theorem 7. (Hirschhorn, 2007) Given n non-negative numbers $x_1, ...x_n$

$$x_1 + ... + x_n \geq n\sqrt[n]{x_1 \times ... \times x_n}$$

    Thus, directly apply the theorem given us $d_p d_i^{T-1} \leq \frac{1}{T}(d_p^T + (T-1)d_i^T)$.

  - $d_p < \frac{o}{M}$, which also means $d_p \leq \frac{o}{M} \leq \mathrm{E}[d_{i^j}] \leq \frac{1+\epsilon}{1-\epsilon}E[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})$. So:

$$
\begin{aligned}
f(\mathbf{o}^{j-1}) - \mathrm{E}[f(\mathbf{o}^j)] \leq f(\mathbf{s}_p^j) - f(\mathbf{s}^{j-1})] \leq d_p &\leq \Big(1 - \frac{1}{k}\Big)\mathrm{E}[d_{i^j}] + \frac{1}{k}\frac{o}{M}\\
&\leq \Big(1 - \frac{1}{k}\Big)\Big(\frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})\Big) + \frac{1}{k}\frac{o}{M}
\end{aligned}
$$

- $|supp(\mathbf{o}^{j-1})| < |supp(\mathbf{o}^j)|$, then $e^j \notin supp(\mathbf{o})$. Due to monotonicity of f.

$$f(\mathbf{o}^{j-1}) - \mathrm{E}[f(\mathbf{o}^j)] \leq 0 \leq \Big(1 - \frac{1}{k}\Big)\Big(\frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})\Big)$$

Therefore, in overall $f(\mathbf{o}^{j-1}) - \mathrm{E}[f(\mathbf{o}^j)] \leq \left(1 - \frac{1}{k}\right)\left(\frac{1+\epsilon}{1-\epsilon}E[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})\right) + \frac{o}{kM}$

Proof of Claim 4 We have

$$
\begin{aligned}
f(\mathbf{o}) - \mathrm{E}[f(\mathbf{o}^t)] &= \sum_{j=1}^{t}\mathrm{E}\Big[f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j)\Big] \leq \sum_{j=1}^{t}\Big(\big(1 - \frac{1}{k}\big)\big(\frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}\mathrm{E}[f(\mathbf{s}^{j-1})]\big) + \frac{o}{kM}\Big)\\
&\leq \Big(1 - \frac{1}{k}\Big)\Big(\frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s})] + \sum_{j=1}^{t-1}\frac{4\epsilon}{1-\epsilon^2}\mathrm{E}[f(\mathbf{s}^j)]\Big) + \frac{oB}{kM}\\
&\leq \Big(1 - \frac{1}{k}\Big)\frac{(1+\epsilon)^2 + 4B\epsilon}{1-\epsilon^2}\mathrm{E}[f(\mathbf{s})] + \frac{1}{kM}f(\mathbf{o})
\end{aligned}
$$

which completes the proof.

Proof of Claim 5 Denote $\{\langle u_1, j_1\rangle, ...\langle u_r, j_r\rangle\}$ as a set of elements and their placements in $\mathbf{o}^t$ that are not in $\mathbf{s}$.

For each $\langle u_i, j_i\rangle$, denote $\mathbf{s}_i$ as $\mathbf{s}$ when Alg. 3 encounters u_i. As u_i was not added into $\mathbf{s}_i$, $\frac{1}{1-\epsilon}F(\mathbf{s}_i \sqcup \langle u_i, j_i\rangle) - \frac{1}{1+\epsilon}F(\mathbf{s}_i) < \frac{o}{M}$

Let $\mathbf{u}_i = \mathbf{s} \sqcup \{\langle u_1, j_1\rangle, ...\langle u_i, j_i\rangle\}$. We have

$$\begin{aligned}
\mathrm{E}[f(\mathbf{o}^t)] - \mathrm{E}[f(\mathbf{s})] &= \sum_{i=1}^{B-t} \mathrm{E}\Big[f(\mathbf{u}_i) - f(\mathbf{u}_{i-1})\Big] \leq \sum_{i=1}^{B-t} \mathrm{E}\Big[f(\mathbf{s}_i \sqcup \langle u_i, j_i\rangle) - f(\mathbf{s}^i)\Big] \\
&\leq \sum_{i=1}^{B-t} \mathrm{E}\Big[\frac{1}{1-\epsilon}F(\mathbf{s}_i \sqcup \langle u_i, j_i\rangle) - \frac{1}{1+\epsilon}F(\mathbf{s}_i)\Big] \leq \sum_{i}^{B-t} \frac{o}{M} \leq \frac{1}{M}f(\mathbf{o})
\end{aligned}$$

which completes the proof.

D.2. Approximation ratio of RSTREAM when f is non-monotone

Proof of Theorem 4 Similar to monotone case, the key of our proof is to show that:

$$\max\big(\alpha(M), \beta(M)\big)\mathrm{E}[f(\mathbf{s}^t)] \geq f(\mathbf{o}) \tag{33}$$

If $t = B$, we have:

$$f(\mathbf{s}) \geq \frac{F(\mathbf{s})}{1+\epsilon} \geq \frac{F(\mathbf{s}^B)}{1+\epsilon} = \frac{1}{1+\epsilon}\sum_{i=1}^{B}\Big(F(\mathbf{s}^i) - F(\mathbf{s}^{i-1})\Big) \tag{34}$$

$$= \frac{1}{1+\epsilon}\sum_{i=1}^{B}\Big(F(\mathbf{s}^i) - \frac{1-\epsilon}{1+\epsilon}F(\mathbf{s}^{i-1})\Big) - \frac{2\epsilon}{(1+\epsilon)^2}\sum_{i}^{B-1}F(\mathbf{s}^i) \tag{35}$$

$$\geq \frac{1-\epsilon}{1+\epsilon}\sum_{i=1}^{B}\frac{o}{M} - \frac{2\epsilon B}{(1+\epsilon)^2}F(\mathbf{s}) \geq \frac{(1-\epsilon)f(\mathbf{o})}{(1+\epsilon)(1+\gamma)M} - \frac{2\epsilon B}{1+\epsilon}f(\mathbf{s}) \tag{36}$$

Therefore, with $t = B$, $\frac{(1+\epsilon+2\epsilon B)(1+\gamma)M}{1-\epsilon}f(\mathbf{s}) \geq f(\mathbf{o})$. The rest of the proof would focus on when $t < B$. We re-use notations $\mathbf{o}_i^j, \mathbf{s}_i^j$ as in the monotone proof and still compare $\mathbf{o}^{j-1}$ and $\mathbf{o}^j$, we have two following cases.

- $|supp(\mathbf{o}^{j-1})| < |supp(\mathbf{o}^j)|$, which means $e^j \notin supp(\mathbf{o})$. We consider 2 sub-cases:
  - If $T < k$, which means there exists $i' \in [k]$ such that $\frac{F(\mathbf{s}_{i'}^j)}{1-\epsilon} - \frac{F(\mathbf{s}^{j-1})}{1+\epsilon} < \frac{o}{M} \leq \frac{\mathrm{E}[F(\mathbf{s}^j)]}{1-\epsilon} - \frac{F(\mathbf{s}^{j-1})}{1+\epsilon}$. Then

$$\begin{aligned}
f(\mathbf{o}^{j-1}) - \mathrm{E}[f(\mathbf{o}^j)] &= f(\mathbf{o}_{i'}^j) - f(\mathbf{o}^{j-1}) - \Big(\mathrm{E}[f(\mathbf{o}^j)] + f(\mathbf{o}_{i'}^j) - 2f(\mathbf{o}^{j-1})\Big) \\
&\leq f(\mathbf{o}_{i'}^j) - f(\mathbf{o}^{j-1}) \leq f(\mathbf{s}_{i'}^j) - f(\mathbf{s}^{j-1}) \leq \frac{F(\mathbf{s}_{i'}^j)}{1-\epsilon} - \frac{F(\mathbf{s}^{j-1})}{1+\epsilon} \\
&\leq \frac{\mathrm{E}[F(\mathbf{s}^j)]}{1-\epsilon} - \frac{F(\mathbf{s}^{j-1})}{1+\epsilon} \leq \frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})
\end{aligned}$$

- If $T = k$, define a permutation $\pi : [k] \to [k]$ such that $\pi(i) \neq i$ for all $i \in [k]$. Then

$$\begin{aligned} f(\mathbf{o}^{j-1}) - \mathrm{E}[f(\mathbf{o}^j)] &= \frac{1}{D}\sum_{i\in[k]}\Big(f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j_i)\Big)d_i^{T-1} \\ &= \frac{1}{D}\sum_{i\in[k]}\Big(f(\mathbf{o}^j_{\pi(i)}) - f(\mathbf{o}^{j-1}) - \big(f(\mathbf{o}^j_{\pi(i)}) + f(\mathbf{o}^j_i) - 2f(\mathbf{o}^{j-1})\big)\Big)d_i^{T-1} \\ &\le \frac{1}{D}\sum_{i\in[k]}\Big(f(\mathbf{o}^j_{\pi(i)}) - f(\mathbf{o}^{j-1})\Big)d_i^{T-1} \le \frac{1}{D}\sum_{i}\Big(f(\mathbf{s}^j_{\pi(i)}) - f(\mathbf{s}^{j-1})\Big)d_i^{T-1} \\ &\le \frac{1}{D}\sum_{i\in[k]}d_{\pi(i)}d_i^{T-1} \le \frac{1}{D}\sum_{i\in[k]}d_i^T = \frac{1}{1-\epsilon}\mathrm{E}[F(\mathbf{s}^j)] - \frac{1}{1+\epsilon}F(\mathbf{s}^{j-1}) \\ &\le \frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1}) \end{aligned}$$

- $|supp(\mathbf{o}^{j-1})| = |supp(\mathbf{o}^j)|$. We re-use notation $p, I, \mathbf{o}^{j-1/2}$ as in the monotone case and consider other two sub-cases:
  - If $d_p \ge \frac{o}{M}$, then

$$\begin{aligned} f(\mathbf{o}^{j-1}) - \mathrm{E}[f(\mathbf{o}^j)] &= \frac{1}{D}\sum_{i\in I\setminus\{p\}}\Big(f(\mathbf{o}^{j-1}) - f(\mathbf{o}^j_i)\Big)d_i^{T-1} \\ &= \frac{1}{D}\sum_{i\in I\setminus\{p\}}\Big(2f(\mathbf{o}^j_p) - 2f(\mathbf{o}^{j-1/2}) - \big(f(\mathbf{o}^j_p) + f(\mathbf{o}^j_i) - 2f(\mathbf{o}^{j-1/2})\big)\Big)d_i^{T-1} \\ &\le \frac{2}{D}\sum_{i\in I\setminus\{p\}}\Big(f(\mathbf{s}^j_p) - f(\mathbf{s}^{j-1})\Big)d_i^{T-1} \le \frac{2}{D}\sum_{i\in I\setminus\{p\}}d_p d_i^{T-1} \le \frac{2}{D}\Big(1-\frac{1}{T}\Big)\sum_{i\in I}d_i^T \\ &\le \Big(2-\frac{2}{k}\Big)\Big(\frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})\Big) \end{aligned}$$

  - If $d_p = 0$, which also means $\frac{1}{1-\epsilon}F(\mathbf{s}^j_p) - \frac{1}{1+\epsilon}F(\mathbf{s}^{j-1}) < \frac{o}{M} \le \frac{1}{1-\epsilon}\mathrm{E}[F(\mathbf{s}^j)] - \frac{1}{1+\epsilon}F(\mathbf{s}^{j-1})$. So:

$$\begin{aligned} f(\mathbf{o}^{j-1}) - \mathrm{E}[f(\mathbf{o}^j)] &= 2f(\mathbf{o}^j_p) - 2f(\mathbf{o}^{j-1/2}) - \Big(f(\mathbf{o}^j_p) + \mathrm{E}[f(\mathbf{o}^j)] - 2f(\mathbf{o}^{j-1/2})\Big) \\ &\le 2f(\mathbf{o}^j_p) - 2f(\mathbf{o}^{j-1/2}) \le 2f(\mathbf{s}^j_p) - 2f(\mathbf{s}^{j-1})] \\ &\le \frac{2}{1-\epsilon}F(\mathbf{s}^j_p) - \frac{2}{1+\epsilon}F(\mathbf{s}^{j-1}) \le \Big(2-\frac{2}{k}\Big)\Big(\frac{\mathrm{E}[F(\mathbf{s}^j)]}{1-\epsilon} - \frac{F(\mathbf{s}^{j-1})}{1+\epsilon}\Big) + \frac{2}{k}\frac{o}{M} \\ &\le \Big(2-\frac{2}{k}\Big)\Big(\frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})\Big) + \frac{2o}{kM} \end{aligned}$$

Since $k \ge 2$, in overall, $f(\mathbf{o}^{j-1}) - \mathrm{E}[f(\mathbf{o}^j)] \le \Big(2-\frac{2}{k}\Big)\Big(\frac{1+\epsilon}{1-\epsilon}E[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})\Big) + \frac{2o}{kM}$. We have

$$f(\mathbf{o}) - \mathrm{E}[f(\mathbf{o}^t)] = \sum_{j=1}^{t}\Big(\mathrm{E}[f(\mathbf{o}^{j-1})] - \mathrm{E}[f(\mathbf{o}^j)]\Big) \le \sum_{j=1}^{t}\Big(\big(2-\frac{2}{k}\big)\big(\frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^j)] - \frac{1-\epsilon}{1+\epsilon}f(\mathbf{s}^{j-1})\big) + \frac{2o}{kM}\Big) \tag{37}$$

$$= \Big(2-\frac{2}{k}\Big)\Big(\frac{1+\epsilon}{1-\epsilon}\mathrm{E}[f(\mathbf{s}^t)] + \sum_{i=1}^{t-1}\frac{4\epsilon}{1-\epsilon^2}\mathrm{E}[f(\mathbf{s}^i)]\Big) + \frac{2oB}{kM} \tag{38}$$

$$\le \Big(2-\frac{2}{k}\Big)\frac{(1+\epsilon)^2 + 4\epsilon B}{1-\epsilon^2}\max_{i\le t}\mathrm{E}[f(\mathbf{s}^i)] + \frac{2}{kM}f(\mathbf{o}) \tag{39}$$

$$\le \Big(2-\frac{2}{k}\Big)\frac{(1+\epsilon)^2 + 4\epsilon B}{(1-\epsilon)^2}\mathrm{E}[f(\mathbf{s})] + \frac{2}{kM}f(\mathbf{o}) \tag{40}$$

Also, similar to monotone case, as $\mathbf{s}^t \sqsubseteq \mathbf{o}^t$, denote $\{\langle u_1, j_1\rangle, ...\langle u_r, j_r\rangle\}$ as a set of elements and their placement in $\mathbf{o}^t$ that are not in $\mathbf{s}^t$. For each $\langle u_i, j_i\rangle$, denote $\mathbf{s}_i$ as $\mathbf{s}^t$ when Alg. 1 encounters u_i. Denote $\mathbf{u}_i = \mathbf{s}^t \sqcup \{\langle u_1, j_1\rangle, ...\langle u_i, j_i\rangle\}$. We have

$$\mathrm{E}[f(\mathbf{o}^t)] - \mathrm{E}[f(\mathbf{s}^t)] = \sum_{i=1}^{B-t} \Big(\mathrm{E}[f(\mathbf{u}^i) - f(\mathbf{u}^{i-1})] \Big) \leq \sum_{i=1}^{B-t} \Big(\mathrm{E}[f(\mathbf{s}_i \sqcup \langle u_i, j_i \rangle) - f(\mathbf{s}_i)] \Big) \tag{41}$$

$$\leq \sum_{i=1}^{B-t} \mathrm{E}\left[\frac{F(\mathbf{s}_i \sqcup \langle u_i, j_i \rangle)}{1-\epsilon} - \frac{F(\mathbf{s}_i)}{1+\epsilon} \right] \leq \sum_{i=1}^{B-t} \frac{o}{M} \leq \frac{1}{M} f(\mathbf{o}) \tag{42}$$

Combining Equ. 40 and 42, we have

$$f(\mathbf{o}) \leq \frac{M}{kM-k-2} \frac{(3k-2)(1+\epsilon)^2 + (8k-8)\epsilon B}{(1-\epsilon)^2} \mathrm{E}[f(\mathbf{s})]$$

The approximation ratio of RSTREAM when discarding assumption of known $f(\mathbf{o})$ trivially follows as in the proof of the monotone case.

E. Sampling Method for Influence Maximization with k topics

In this part, we would present the sampling method that helps obtaining $F(\mathbf{s})$ satisfying $(1-\epsilon)\mathbb{I}(\mathbf{s}) \leq F(\mathbf{s}) \leq (1+\epsilon)\mathbb{I}(\mathbf{s})$ with high probability. We adopt a concept of Reverse Influence Sampling (RIS) (Borgs et al., 2014) to the problem as follows:

Given a social network $G = (V, E)$ and $w^i_{u,v}$ is a weight of edge (u, v) on topic i, a random Reverse Reachable (RR) sample $\mathcal{R} = \{R_1, ..., R_k\}$ is generated from G by: (1) selecting a random node $v \in V$; (2) generating sample graph $\{g_1, g_2, ...g_k\}$ from G, where g_i is generated from weights $\{w^i_{u,v}\}$ for all $(u, v) \in E$; (3) return $\mathcal{R} = \{R_1, ...R_k\}$ where R_i is a set of nodes that can reach v in g_i. A k-set $\mathbf{s} = \{S_1, ...S_k\}$ would activate the random sample $\mathcal{R}$ iff there exists $i \in [k]$ that $S_i \cap R_i \neq \emptyset$. For simplicity, denote $\mathbf{s} \triangleright \mathcal{R}$ as an indicator variable whether $\mathbf{s}$ activates $\mathcal{R}$. Using a similar proof as **Observation 3.2** (Borgs et al., 2014), we have $\mathbb{I}(\mathbf{s}) = |V| \cdot \Pr_{\mathcal{R}}[\mathbf{s} \triangleright \mathcal{R}]$.

To estimate $\mathbb{I}(\mathbf{s})$, we generates multiple samples $\mathcal{R}_1, ...\mathcal{R}_n$ and let $F(\mathbf{s}) = \frac{|V|}{n} \sum_{i=1}^n \mathbf{s} \triangleright \mathcal{R}_i$. We apply Chernoff Bound [2] to bound the number of samples, which guarantee $F(\mathbf{s})$ is an ϵ-estimate of $\mathbb{I}(\mathbf{s})$ with high probability. The Chernoff Bound theorem is stated as follows.

Theorem 8. (Chernoff bound) Suppose $X_1, ...X_n$ are independent random variables taking values in $\{0, 1\}$. Let X denote their sum and let $\mu = \mathrm{E}[X]$. Then for $\epsilon \in [0, 1]$ we have

$$Pr\Big(X \leq (1-\epsilon)\mu\Big) \leq e^{-\frac{\epsilon^2 \mu}{2}}$$
$$Pr\Big(X \geq (1+\epsilon)\mu\Big) \leq e^{-\frac{\epsilon^2 \mu}{3}}$$

Therefore, the number of samples that helps us obtaining ϵ-estimate of $\mathbb{I}(\mathbf{s})$ is stated in the following lemma.

Lemma 3. Given a seed set $\mathbf{s}$, by generating at least $n = \frac{3|V|}{\epsilon^2 |\mathbf{s}|} \ln \frac{1}{1-\sqrt{\lambda}}$ RR samples, $F(\mathbf{s})$ is ϵ-estimate of $\mathbb{I}(\mathbf{s})$ with probability at least λ.

Proof. A slight change in algebra in Chernoff bound helps us obtaining:

$$\Pr\Big((1-\epsilon)\mathbb{I}(\mathbf{s}) \leq F(\mathbf{s}) \leq (1+\epsilon)\mathbb{I}(\mathbf{s})\Big) \geq \big(1 - e^{-\frac{\epsilon^2 \mathbb{I}(\mathbf{s}) n}{3|V|}}\big)^2 \tag{43}$$

Since $\mathbb{I}(\mathbf{s}) \geq |\mathbf{s}|$, $n \geq \frac{3|V|}{\epsilon^2 |\mathbf{s}|} \ln \frac{1}{1-\sqrt{\lambda}}$ guarantees $F(\mathbf{s})$ is ϵ-estimate of $\mathbb{I}(\mathbf{s})$ with probability at least λ ☐

In experiment, we set $\lambda = 0.8$ whenever the algorithms query $\mathbb{I}(\cdot)$.

[2] http://math.mit.edu/ goemans/18310S15/chernoff-notes.pdf