Tight Trade-offs for the Maximum k-Coverage Problem in the General Streaming Model

Piotr Indyk
indyk@mit.edu
CSAIL, MIT

Ali Vakilian
vakilian@mit.edu
CSAIL, MIT

ABSTRACT

We study the *maximum k-coverage* problem in the general *edge-arrival* streaming model: given a collection of m sets $\mathcal{F}$, each subset of a ground set of elements $\mathcal{U}$ of size n, the task is to find k sets whose coverage is maximized. The sets are specified as a sequence of (element, set) pairs in an *arbitrary* order. Our main result is a *tight* (up to polylogarithmic factors) trade-off between the space complexity and the approximation factor $\alpha \in (1/(1-1/e), \widetilde{\Omega}(\sqrt{m})]$ of any single-pass streaming algorithm that estimates the maximum coverage size. Specifically, we show that the optimal space bound is $\widetilde{\Theta}(m/\alpha^2)$. Moreover, we design a single-pass algorithm that reports an α-approximate solution in $\widetilde{O}(m/\alpha^2 + k)$ space.[1]

Our algorithm heavily exploits data stream sketching techniques, which could lead to further connections between vector sketching methods and streaming algorithms for combinatorial optimization tasks.

CCS CONCEPTS

• **Theory of computation → Streaming models**; **Sketching and sampling**; Lower bounds and information complexity.

KEYWORDS

max k-cover, sketching/streaming, heavy hitters

ACM Reference Format:
Piotr Indyk and Ali Vakilian. 2019. Tight Trade-offs for the Maximum k-Coverage Problem in the General Streaming Model. In *38th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems (PODS '19), June 30–July 5, 2019, Amsterdam, Netherlands.* ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3294052.3319691

[1] $\widetilde{O}, \widetilde{\Omega}, \widetilde{\Theta}$ suppress polylog factors.

PODS '19, June 30–July 5, 2019, Amsterdam, Netherlands

ACM ISBN 978-1-4503-6227-6/19/06…$15.00
https://doi.org/10.1145/3294052.3319691

1 INTRODUCTION

Maximum k-coverage (Max k-Cover) is a classic problem in combinatorial optimization. Given a ground set $\mathcal{U}$ of n elements, a family of m sets $\mathcal{F}$ (each is a subset of $\mathcal{U}$), and a parameter k, the goal is to select k sets in $\mathcal{F}$ whose union has the largest cardinality. Max k-Cover is an important problem in submodular maximization, with applications in many areas, including operations research, machine learning, information retrieval and data mining [1, 19, 37].

The classic greedy algorithm for this problem [35] guarantees an approximation ratio of $1/(1-1/e)$, which is known to be tight under P ≠ NP [23]. Unfortunately, the standard greedy algorithm does not scale very well to massive data sets [19]. This has led to considerable interest in developing maximum coverage algorithms tailored to modern architectures specifically designed for massive data processing. In particular [37] gave the first algorithm for the problem in the data streaming model, where the algorithm is required to make a single pass over the input $\mathcal{F}$ while using a sub-linear amount of memory. Since then, there has been a large body of work designing space-efficient streaming algorithms for maximum coverage [6, 12, 33, 34], as well as its dual variant, *set cover* [6, 7, 12, 17, 21, 22, 26–28].

The initial algorithms were developed in the *set arrival* model, where the input sets are listed contiguously. This restriction is natural from the perspective of submodular optimization, but limits the applicability of the algorithms[2]. Avoiding this limitation can be difficult, as streaming algorithms can no longer operate on sets as "unit objects". As a result, the first maximum coverage algorithm for the general *edge arrival* model, where pairs of (set, element) can arrive in arbitrary order, have been developed recently. In particular [12] presented a one-pass algorithm with space linear in m and constant approximation factor[3]. We remark that many of

[2] For example, consider a situation where the sets correspond to neighborhoods of vertices in a directed graph. Depending on the input representation, for each vertex, either the ingoing edges or the outgoing edges might be placed non-contiguously.

[3] Note that in [12] m denotes the number of elements and n denotes the number of sets.

the prior bounds (both upper and lower bounds) on set cover and max k-cover problems in set-arrival streams also work in edge arrival streams (e.g. [6, 7, 21, 26, 27, 34]). However, the design of efficient streaming algorithms for the coverage problems on edge arrival streams was first studied explicitly in [12].

A particularly interesting line of research in set arrival streaming set cover and max k-cover is to design efficient algorithms that only use $\widetilde{O}(n)$ space [9, 17, 22, 34, 37]. Previous work have shown that we can adopt the existing greedy algorithm of max k-cover to achieve constant factor approximation in $\widetilde{O}(n)$ space [9, 37] (which later improved to $\widetilde{O}(k)$ by [34]). However, the complexity of the problem in the "low space" regime is very different in edge-arrival streams: [12] showed that as long as the approximation factor is a constant, any algorithm must use $\Omega(m)$ space. Still, our understanding of approximation/space trade-offs in the general case is far from complete. Table 1 summarizes the known results.

Other related work. Another important related question in this area is to design a "low-approximation" (i.e., better than the 2-approximation guarantee of the greedy approach) streaming algorithm for the max k-cover problem in the *set arrival* setting. Recently, Norouzi-Fard et al. [36] presented the first streaming algorithm that improves upon 2-approximation guarantee of greedy approach on *random arrival* streams. Very recently, Agrawal et al. [2] achieved an almost $1/(1-1/e)$-approximation in $\widetilde{O}(n)$ space which is essentially the optimal bound [34][4]. Still it is an important question to design such algorithms on adversarial order streams. We also remark that the algorithms of [2, 36] do not work on *edge arrival* streams.

In many scenarios, space is the most critical factor, and thus the question becomes: what approximation guarantees are possible within the given space bounds? This question has been studied before in the context of *set cover* in set arrival streams (e.g. [17, 22]), leading to poly(n, m)-factor approximation algorithms.

Our results. In this paper, we complement the work of [12] by designing space-efficient algorithms for super-constant approximation factors α. In fact, we show a tight (up to poly-logarithmic factors) trade-off between the two: the optimal space bound is $\widetilde{O}(\frac{m}{\alpha^2})$ for estimating the maximum coverage value, and $\widetilde{O}(\frac{m}{\alpha^2}+k)$ for reporting an approximately optimal solution[5]. The approximation factor α can take any value in $(1/(1-1/e), \widetilde{\Omega}(\sqrt{m})]$.

Our techniques. In the edge arrival model, elements of each set can arrive irregularly and out of order. This necessitates the use of methods that aggregate the information about the input sets, or their coverage. In particular, distinct element sketches were used both in [12] (implicitly) and [34] (explicitly). In this paper we expand the use of sketching toolkit. Specifically, in addition to distinct element estimation [5, 11, 13, 30, 31], we also need algorithms for *heavy hitters* with respect to the L_2 norm [14, 15, 18, 39], as well as a frequency-based partitioning of elements, and detecting sets that "substantially contribute" to the solution [29]. Application of vector-sketching techniques (e.g. L_p-sampling/estimation and heavy hitters) in graph streaming settings have been studied extensively (e.g. [3, 4, 8, 20, 25, 32]). We believe that our algorithms can lead to further connections between vector sketching methods and streaming algorithms for the coverage problems.

Lower bound. Our algorithm was inspired by the lower bound. Specifically, it was previously shown by [12] that approximating Max k-Cover by a factor better than 2 requires $\Omega(m)$ space. Similar approach works for larger values of α, by showing a reduction from the α-player *set disjointess* problem ($\mathsf{DSJ}_{[m]}$) with *unique intersection* guarantee (i.e., either players' sets are disjoint or there is a unique item that appears in all sets) to the task of α-approximating Max k-Cover.

The specific hard instances in the aforementioned lower bound can be distinguished in the streaming model using space $O(\frac{m}{\alpha^2})$. To this end, we compute an α-approximation to the L_∞-norm of a vector v that, for each element e, counts the number of sets e belongs to. This problem can be solved in $O(\frac{m}{\alpha^2})$ space, by using L_2-norm sketches [5]. This suggests that it might be possible to solve the *general* Max k-Cover using sketching techniques as well.

Upper bound. We start our algorithm with a "coverage boosting" *universe reduction* technique which constructs a reduced size instance (i.e. with reduced ground set) whose optimal k-cover has constant fraction coverage (see Section 3.1). This step is particularly important as the space complexity of the existing methods for Max k-Cover is proportional to the reciprocal of the fraction of covered elements in an optimal solution.

Once we have a constant fraction coverage guarantee, our algorithm exploits three different approaches so that on any instance, at least one of them reports a "good" approximate solution.

[4]Both [2, 36] study the more general problem of submodular maximization and their results are stated with different notation and assuming *oracle* access. Here, we state their guarantees for max cover on set arrival streams

[5]We note that similar trade-offs were previously obtained for the set cover problem, as [7] showed a $\Theta(mn/\alpha^2)$ bound for estimation, and a $\Theta(mn/\alpha)$ bound for reporting. Interestingly, the $1/\alpha^2$ vs. $1/\alpha$ gap does not occur for our problem.

Estimation/ Reporting	Set/Edge Arrival	Approximation	Space Upper Bound	Space Lower Bound
Estimation	Set Arrival	$1/(1-\varepsilon)$	–	$\widetilde{\Omega}(\frac{m}{\varepsilon^2})$[6]
Estimation	Set Arrival	$1/(1-1/e-\varepsilon)$	–	$\Omega(\frac{m}{k^2})$[34]
Reporting	Edge Arrival	$1/(1-1/e-\varepsilon)^a$	$\widetilde{O}(\frac{m}{\varepsilon^3})$[12], $\widetilde{O}(\frac{m}{\varepsilon^2})$[34]	–
Reporting	Set Arrival	4 [37], 2 [9]^b	$\widetilde{O}(n)$	–
Reporting	Set Arrival	$2+\varepsilon$	$\widetilde{O}(\frac{k}{\varepsilon^3})$[34]	–
Estimation	Edge Arrival	α	$\widetilde{O}(\frac{m}{\alpha^2})$ **[here]**	$\Omega(\frac{m}{\alpha^2})$ **[here]**,[12]
Reporting	Edge Arrival	α	$\widetilde{O}(\frac{m}{\alpha^2}+k)$ **[here]**	–

Table 1: The summary of known results on the complexity of single-pass streaming algorithms of Max k-Cover**.**

[a]Allowing exponential runtime, the approximation factor becomes $\frac{1}{1-\varepsilon}$ which matches the information theoretic lower bound of [6] up to a factor of $\frac{1}{\varepsilon}$.
[b]Their result works for the general submodular maximization assuming access to a value oracle that given a collection of sets computes their coverage. A careful adoption of their result to Max k-Cover (without the value oracle) uses $\widetilde{O}(n)$ space.

Multi-layered set sampling. By extending the *set sampling* approach (see Section 2.1) and trying a larger range of sampling rate, $[\frac{k}{m}, \frac{\alpha k}{m}]$, we design a *smooth* variant of set sampling: a collection of sets sampled uniformly and independently[6] at rate $\widetilde{O}(\frac{\beta k}{m})$ w.h.p., covers all elements that appear in at least $\frac{m}{\beta k}$ sets. Besides expanding the application of set sampling in finding α-approximate k-cover, this smooth variant implies more structure on the number of elements in a wider range of frequency levels which is specifically crucial in our approach for detecting sets with "low contribution".

Unlike the set sampling based technique whose success in finding an α-approximate k-cover *only* depends on the structure of the set system, the performance of the next two approaches rely on the structure of optimal solutions as well: whether the majority of the coverage (in a specific optimal k-cover) is due to (few) "large" sets or, (many) "small" sets.

Heavy hitters and contributing frequencies. The high level idea in this approach is that *if in an optimal solution, a sufficiently*[7] *small number of sets cover the majority of the elements (covered by the optimal solution), it is enough to find a single large set*, which naturally hints the use of ideas related to heavy hitters. For the sake of efficiency (in space complexity), we *randomly* partition sets into *supersets* of size at most k. However, once we merge sets into a single superset, we can no longer distinguish between their coverage and their total size. Since we combine sets at random, if all elements have "low" frequency in the set system, then the gap between the total size of all sets in a superset and their coverage is just $\widetilde{O}(1)$. This observation implies that if there is no "common" element in the set system, then we can use the total size of the sets in a superset as an estimate of its coverage size. To get around the case with (many) "common" elements, we show that performing the heavy hitter based algorithm on a sampled set of elements will find a sufficiently large superset as desired (see Section B).

Detecting k-covers with many small sets. Finally, we address the case in which an optimal k-cover consists of many "small" sets. In this case, we can show that after subsampling sets uniformly with probability $1/\alpha$, a $(\frac{k}{\alpha})$-cover with coverage at least $\Theta(1/\alpha)$ times the coverage of an optimal k-cover survives. This sampling method will save us a factor of α in the memory usage of the algorithm. Further, by exploiting the structural property guaranteed due to the multi-layered set sampling[8], we can show that element sampling can save another factor of α in the space complexity once applied to find a constant factor approximate Max $(\frac{k}{\alpha})$-Cover of the subsampled sets.

2 PRELIMINARIES AND NOTATIONS

2.1 Sampling Methods for Max k-Cover

Here we describe two sampling methods that have been used widely in the design of streaming algorithms for Max k-Cover and Set Cover [6, 7, 12, 21, 26, 27, 33, 34]. For a collection of sets $\mathcal{Q}$, we define $C(\mathcal{Q})$ to denote the set of elements that are covered by $\mathcal{Q}$; $C(\mathcal{Q}) := \bigcup_{S\in\mathcal{Q}} S$. Moreover, we denote an optimal k-cover of $(\mathcal{U}, \mathcal{F})$ by OPT.

Set Sampling. Roughly speaking, it says that by selecting sets *uniformly at random*, with high probability, all elements that appear in large number of sets will be covered.

Definition 2.1 (λ-common element). An element $e \in \mathcal{U}$ is called λ-*common* if it appears in at least $c \cdot m \cdot \text{polylog}(m,n)/\lambda$

[6]In fact, $O(\log mn)$-wise independent is sufficient for all applications in this paper.
[7]Depending on how large our desired approximation factor α is.
[8]If the multi-layered set sampling fails to return an α-approximate estimate, we can infer strong conditions on the maximum number of elements that belong to each frequency level in $[\frac{m}{k}, \frac{m}{\alpha k}]$.

sets in $\mathcal{F}$. Furthermore, We denote the set of λ-common elements by $\mathcal{U}^{\text{cmn}}_{\lambda}$.

Observation 2.2. *For any $0 \le \lambda_1 \le \lambda_2$, $\mathcal{U}^{\text{cmn}}_{\lambda_1} \subseteq \mathcal{U}^{\text{cmn}}_{\lambda_2}$.*

Lemma 2.3 (Set Sampling [21]). *Consider a set system $(\mathcal{U}, \mathcal{F})$ and let $\mathcal{F}^{\text{rnd}} \subseteq \mathcal{F}$ be a collection of sets such that each set S is picked in $\mathcal{F}^{\text{rnd}}$ with probability $\frac{\lambda}{m}$. With high probability, $\mathcal{F}^{\text{rnd}}$ covers all elements that appear in $\widetilde{\Omega}(m/\lambda)$ sets (i.e. λ-common elements).*

Observation 2.4. *Let $\mathcal{Q}$ be a subset of $\mathcal{F}$ of size (βk). Then, in any partitioning of $\mathcal{Q}$ into β groups, there exists one group with coverage at least $|C(\mathcal{Q})|/\beta$. In particular, an optimal k-cover in $\mathcal{Q}$ covers at least $|C(\mathcal{Q})|/\beta$.*

This simple observation is in particular interesting because it relates the task of α-approximating Max k-Cover to solving instances of Max (βk)-Cover where $\beta \le \alpha$.

Element Sampling for Max k-Cover. This sampling method shows that if we sample elements of $\mathcal{U}$ uniformly with a large enough rate (i.e. proportional to $(k|\mathcal{U}|)/|C(\text{OPT})|$), then a constant factor approximate k-cover over the sampled elements w.h.p., is a constant factor approximate solution of the original instance.

Lemma 2.5 (Element Sampling Lemma [21, 33]). *Consider an instance of* Max k-Cover$(\mathcal{U}, \mathcal{F})$. *Let's assume that an optimal k-cover of $(\mathcal{U}, \mathcal{F})$ covers $(1/\eta)$-fraction of $\mathcal{U}$. Let $\mathcal{L} \subset \mathcal{U}$ be a set of elements of size $\widetilde{\Theta}(\eta k)$ picked uniformly at random. Then, with high probability, a $\Theta(1)$-approximate k-cover of $(\mathcal{L}, \mathcal{F})$ is a $\Theta(1)$-approximate k-cover of $(\mathcal{U}, \mathcal{F})$.*

2.2 HeavyHitters and Contributing Classes

Suppose that a sequence of items $p_1, \cdots, p_T$ arrive in a data stream where for each $j \le T$, $p_j \in [m]$. We can think of the stream as a sequence of (insertion only) updates on an initially zero vector $\vec{a}$ such that upon arrival of p_j in the stream, $\vec{a}[j] \leftarrow \vec{a}[j] + 1$. Here, we review the notion of F_2-*heavy hitter* and *contributing coordinates* that are used in our algorithm for approximating Max k-Cover.

*Definition 2.6 (F_2-*HeavyHitters*).* Given an m-dimensional vector $\vec{a}$, an item j (corresponding to $\vec{a}[j]$) is a ϕ-HeavyHitter of $F_2(\vec{a})$, if $\vec{a}[j]^2 \ge \phi \cdot F_2(\vec{a}) = \phi \cdot \sum_{j \in [m]} \vec{a}[j]^2$. Intuitively, the set of items that appear frequently in the stream are the heavy hitters.

We conceptually partition coordinates of $\vec{a}$ into classes

$$R_i = \{j \mid 2^{i-1} < \vec{a}[j] \le 2^i\}.$$

Definition 2.7 (γ-contributing coordinates). A class of coordinates R_t is **γ-contributing** if $|R_t| \cdot 2^{2t} \ge \gamma F_2(\vec{a}) = \gamma \sum_{j \in [m]} \vec{a}[j]^2$.

Let R_{t^*} be a γ-contributing class and let n_{t^*} denote the size of R_{t^*}; $n_{t^*} = |R_{t^*}|$. Further, let's assume that $i^* = \lceil \log n_{t^*} \rceil$; $2^{i^*-1} < n_{t^*} \le 2^{i^*}$. Let $h : [m] \to [(12m \log m)/2^{i^*}]$ be a function chosen uniformly at random from a family of $\Theta(\log(mn))$-wise independent hash functions. We define $\mathcal{S}_{i^*}$ as a *sampled substream* of the input stream with rate $1/2^{i^*}$. More precisely, $\mathcal{S}_{i^*}$ only contains the updates corresponding to the coordinates $\mathcal{F}_{i^*} = \{j \mid h(j) = 1\}$ that are mapped to one under h. Next, we show that the survived coordinates of R_{t^*} ($j \in R_{t^*}$) in $\vec{a}_{i^*}$, which is the vector $\vec{a}$ restricted to the items in $\mathcal{F}_{i^*}$, are $\widetilde{\Omega}(\gamma)$-HeavyHitters of $F_2(\vec{a}_{i^*})$; $\vec{a}[j]^2 \ge \widetilde{\Omega}(\gamma) \cdot F_2(\vec{a}_{i^*})$. Roughly speaking, if we subsample the stream so that only polylog(m) coordinates of R_{t^*} survive, then with high probability these coordinates are $\widetilde{\Omega}(\gamma)$-HeavyHitters of the sampled substream.

Claim 2.8. *With probability at least $1 - m^{-1}$, the number of survived coordinates in the sampled substream $\mathcal{S}_{i^*}$ is at least $(6m \log m)/2^{i^*}$.*

Lemma 2.9. *With probability at least $1 - 2/(9 \log^2 n \log^c m)$, a coordinate $j \in R_{t^*}$ is a $(\frac{\gamma}{162 \log^2 n \log^{c+1} m})$-*HeavyHitter *in the sampled substream $\mathcal{S}_{t^*}$.*

Next, we can use the exiting algorithms for F_2-HeavyHitters to complete this section.

Theorem 2.10 (F_2-heavy hitters algorithm [14, 15, 18, 39]). *Let's assume that $\vec{a}$ is a m-dimensional vector initialized to zero. Let $\mathcal{S}$ be a stream of items $p_1, \cdots, p_T$ where for each $j \in [T]$, $p_j \in [m]$. Then, there is a single pass algorithm F_2-HeavyHitter that uses $\widetilde{O}(1/\gamma)$ space and with high probability returns all coordinates i such that $\vec{a}[i]^2 \ge \gamma F_2(\vec{a})$. In addition, it returns $(1 \pm \frac{1}{2})$-approximate values of these coordinates.*

Finally, there exists an algorithm that with probability at least $1 - 2/(9 \log n \log^c m)$, finds at least one coordinate in each γ-contributing class of $\vec{a}$ using $\widetilde{O}(1/\gamma)$ space.

Theorem 2.11 (γ-contributing algorithm [29]). *Let's assume that $\vec{a}$ is an m-dimensional vector initialized to zero. Let $\mathcal{S}$ be a stream of items $p_1, \cdots, p_T$ where for each $j \in [T]$, $p_j \in [m]$. Moreover, let's assume no item in $\mathcal{S}$ has frequency more than n. There exists a single pass algorithm F_2-Contributing that uses $\widetilde{O}(1/\gamma)$ space and with probability at least $1 - 2/(9 \log n \log^c m)$ returns a coordinate i from each γ-contributing class. In addition, it returns $(1 \pm \frac{1}{2})$-approximate frequency of these coordinates.*

Proof. There are at most $\log n$ (the total number of classes) γ-contributing classes for $\vec{a}$ and for each γ-contributing class R_t, by Lemma 2.9, with probability at least $1 - 2/(9 \log^2 n \log^c m)$, a coordinate in R_t will be a $\widetilde{\Omega}(\gamma)$-HeavyHitter of $F_2(\vec{a}_{i^*})$ (where $i^* = \lceil \log(n_t) \rceil$). By trying all values of $i^* \in [\log n]$,

with probability at least

$$1 - \log n\left(\frac{2}{9\log^2 n \log^c m}\right) \geq 1 - \frac{2}{9\log n \log^c m}$$

F_2-Contributing algorithm outputs a coordinate from each γ-contributing class. □

F_2-**Contributing**(γ, r):
for each $n_t \in \{2^i \mid i \in [\log \mathrm{r}]\}$ **do** in parallel
  ▷ n_t : #coordinates in a γ-contributing class
  let $\phi \leftarrow \left(\frac{\gamma}{432 \log n \log^{c+1} m}\right)$ ▷ ϕ-HeavyHitter
  let HH be a instance of F_2-HeavyHitter(ϕ)
  let $\rho \leftarrow (12 \log m)/2^i$ ▷ sample rate
  pick $h : [m] \to [m/\rho]$ from a family of
    $\Theta(\log(mn))$-wise independent hash functions
  for each i in the data stream
    if $h(i) = 1$ **then** feed i to HH
  ▷ heavy coordinates with their approximate freq.
  return output of HH

2.3 L_0-Estimation

Norm estimation is one of the fundamental problems in the area of streaming algorithms where we are given an m-dimensional vector $\vec{a}$ which is initialized to zero and a sequence of items $p_1, \cdots, p_T$ (updates for the vector $\vec{a}$) where for each $j \in [T]$, $p_j \in [m]$ arrive in a data stream. In the well-studied task L_0-estimation (also known as Count-distinct problem), the goal is to output a $(1 \pm \varepsilon)$-estimate of the number of distinct elements (i.e., $L_0(\vec{a}) := |\{i \mid \vec{a}[i] \neq 0\}|$) after reading the whole stream.

Theorem 2.12 (L_0-estimation [5, 11, 13, 30, 31]). *Let's assume that $\vec{a}$ is an m-dimensional vector initialized to zero. Let $\mathcal{S}$ be a stream of items $p_1, \cdots, p_T$ where for each $j \in [T]$, $i_j \in [m]$. There exists a single pass algorithm that returns a $(1 \pm 1/2)$-approximation of $L_0(\vec{a})$ and uses $\widetilde{O}(1)$ space.*

3 ESTIMATING SIZE OF MAXIMUM COVERAGE

In this section, we describe the outline of our single-pass algorithm that approximates the coverage size of an optimal k-cover of $(\mathcal{U}, \mathcal{F})$ within a factor of α using $\widetilde{O}(m/\alpha^2)$ space in *arbitrary order* edge arrival streams. The input to our algorithm is $k, \alpha, n = |\mathcal{U}|$ and $m = |\mathcal{F}|$. In high level, we perform three different subroutines in parallel and show that for any given Max k-Cover instance, at least one of the subroutines estimates the optimal coverage size within the desired factor in the promised space.

Theorem 3.1. *For any $\alpha \in [\widetilde{O}(1), \widetilde{\Omega}(\sqrt{m})]$, the single-pass algorithm EstimateMaxCover uses $\widetilde{O}(m/\alpha^2)$ space and with probability at least 3/4 computes the* **size of an optimal coverage** *of* Max k-Cover$(\mathcal{U}, \mathcal{F})$ *within a factor of α in edge-arrival streams.*

Note that Theorem 3.1 together with the $O(1)$-approximation algorithms of [12, 34] that use $\widetilde{O}(m)$ space, imply that for any $\alpha \in (1/(1-1/e), \widetilde{\Omega}(\sqrt{m})]$, there exists a single-pass streaming algorithm that computes an α-approximation of the optimal coverage size of Max k-Cover$(\mathcal{U}, \mathcal{F})$ in $\widetilde{O}(\frac{m}{\alpha^2})$ space. In the longer version of the paper, we extend our approach further to achieve a single pass algorithm that computes an α-approximate k-cover in $\widetilde{O}(\frac{m}{\alpha^2} + k)$ space.

Theorem 3.2. *For any $\alpha \in [\widetilde{O}(1), \widetilde{\Omega}(\sqrt{m})]$, there exists a single-pass algorithm that uses $\widetilde{O}(\frac{m}{\alpha^2} + k)$ space and with probability at least 3/4 returns an α-***approximate solution** *of* Max k-Cover *$(\mathcal{U}, \mathcal{F})$ in edge-arrival streams.*

Finally, we complement our upper bounds with a matching lower bound in Section 5.

Theorem 3.3. *Any single pass (possibly randomized) algorithm on edge-arrival streams that α-approximates the optimal coverage size of* Max k-Cover *requires $\Omega(\frac{m}{\alpha^2})$ space.*

As a first step, we provide a mapping from the ground set $\mathcal{U}$ to a small size set of *pseudo-elements* such that the optimal k-cover on the pseudo-elements covers a constant fraction of the pseudo-elements. This reduction is in particular useful for bounding the number of required samples in methods such as element sampling.

3.1 Universe Reduction

In this section we show that in order to solve Max k-Cover on *edge-arrival* streams, it suffices to solve the instances whose optimal coverage size are at least a constant fraction of $|\mathcal{U}|$. This reduction is particularly important as the space complexity of the existing methods for Max k-Cover is proportional to the reciprocal of the fraction of covered elements in an optimal solution. To this end, suppose that we have an algorithm $\mathcal{A}$ for Max k-Cover in edge-arrival streams with the following properties:

Definition 3.4 ((α, δ, η)-oracle for Max k-Cover*).* An algorithm $\mathcal{A}$ is an *(α, δ, η)-oracle* for Max k-Cover if it satisfies the following properties (α denotes the approximation guarantee, δ denotes the failure probability and η is the promised coverage of an optimal k-cover):

- If the optimal coverage size of Max k-Cover$(\mathcal{U}, \mathcal{F})$ is at least $|\mathcal{U}|/\eta$, then with probability at least $1 - \delta$, $\mathcal{A}$ returns an α-approximation of the optimal coverage size.
- If $\mathcal{A}$ returns z, then an optimal solution of Max k-Cover $(\mathcal{U}, \mathcal{F})$ with high probability, has coverage at least z.

EstimateMaxCover (as in Figure 1) is an $\widetilde{O}(\alpha)$-approximation algorithm for Max k-Cover on edge arrival streams that invokes (α, δ, η)-oracles for Max k-Cover.

EstimateMaxCover(k, α):
if $k\alpha \geq m$ **then do return** n/α ▷ trivial bound
▷ different guesses of optimal coverage size
for each $z \in \{2^i \mid i \in [\log n]\}$ **do in parallel**
  $\text{est}_z \leftarrow 0$
  repeat $\log(\frac{1}{\delta})$ times ▷ boosting success probability
    pick $h : \mathcal{U} \to [z]$ from a family of 4-wise independent hash functions.
    for each (S, e) in the data stream
      feed $(S, h(e))$ to (α, δ, η)-**oracle** $\mathcal{A}$ ▷ $\mathcal{S}_h$
    $\text{est}_z \leftarrow \max(\text{output of } \mathcal{A} \text{ on } \mathcal{S}_h, \text{est}_z)$
return $\max\{\text{est}_z \mid \text{est}_z \geq z/(4\alpha)\}$

Figure 1: A single-pass algorithm that computes an $\widetilde{O}(\alpha)$-approximation of the optimal coverage size of Max k-Cover**.**

As in EstimateMaxCover, let $h : \mathcal{U} \to [z]$ be a hash function picked uniformly at random from a family of 4-wise independent hash functions mapping the ground set $\mathcal{U}$ onto *pseudo-elements* $\mathcal{V} = \{1, \cdots, z\}$. Furthermore, for a subset of elements S, we define $h(S) := \bigcup_{e \in S} h(e)$.

Lemma 3.5. *Let $h : \mathcal{U} \to [z]$ be a hash function picked uniformly at random from a family of 4-wise independent hash functions where $z \geq 32$. Further, let S be a subset of $\mathcal{U}$ of size at least z. Then, with probability at least $3/4$, $|h(S)| \geq z/4$.*

Proof. For any pair of elements $e_i, e_j \in S$, let $X_{i,j}$ be a random variable which is *one* if $h(e_i) = h(e_j)$ (i.e. they collide) and *zero* otherwise. Let $X := \sum_{e_i, e_j \in S} X_{i,j}$ denote the the total number of collision among the elements of S under h.

First, we show that if $X \leq |S|^2/\gamma$, then $|h(S)| \geq \gamma/4$. Let's assume $h(S) = \{v_1, \cdots, v_q\}$ and let n_i denote the number of elements in S that are mapped to v_i by h. Then, the total number of collision, X, is

$$X = \sum_{i=1}^{q} \binom{n_i}{2} \geq \sum_{i=1}^{q} (\frac{n_i}{2})^2 \geq \frac{1}{4} \cdot q \cdot (\frac{|S|}{q})^2 = \frac{|S|^2}{4q}.$$

This implies that $q = |h(S)| \geq \gamma/4$. Using this observation, it only remains to show that with probability at least $3/4$, $|X| \leq |S|^2/z$. Since h is selected from a family of 4-wise independent hash functions, $\{X_{i,j}\}_{e_i, e_j \in S}$ are *pairwise independent*. Hence,

$$\mathbf{E}[X] = \sum_{e_i, e_j \in S} \mathbf{E}[X_{i,j}] = \binom{|S|}{2} \cdot (\frac{1}{z}) \leq \frac{|S|^2}{2z},$$

$$\mathbf{Var}[X] = \sum_{e_i, e_j \in S} \mathbf{Var}[X_{i,j}] = \binom{|S|}{2} \cdot (\frac{1}{z} - \frac{1}{z^2}) \geq \frac{z}{8}.$$

Applying Chebyshev's inequality,

$$\Pr(X > |S|^2/z) \leq \Pr(X > \mathbf{E}[X] + \mathbf{Var}[X]) \leq \frac{1}{\mathbf{Var}[X]} \leq \frac{8}{z} \leq \frac{1}{4}.$$

Hence, with probability at least $3/4$, $X \leq |S|^2/z$ which implies that with probability at least $3/4$, $|h(S)| \geq z/4$. □

Theorem 3.6. *Suppose that there exists a (α, δ, η)-oracle for* Max k-Cover *on edge-arrival streams that uses $f(m, \alpha)$ space with $\eta \geq 4$ where m denotes the number of sets in the input. Then, EstimateMaxCover is an $O(\alpha)$-approximation algorithm for* Max k-Cover *with failure probability at most $4\delta \log n$ that uses $\widetilde{O}(f(m, \alpha))$ space on edge-arrival streams.*

Proof. Let OPT denote an optimal solution of Max k-Cover on $(\mathcal{U}, \mathcal{F})$. First, we show that with high probability, EstimateMaxCover returns $\Omega(|C(\text{OPT})|/\alpha)$. Note that for each guess on the optimal coverage size $z \leq |C(\text{OPT})|$, by Lemma 3.5, the probability that in none of $\log(1/\delta)$ iterations $|h(C(\text{OPT}))| > |C(\text{OPT})|/4$ is at most δ (i.e., none of the iterations preserve the optimal coverage size up to a factor of 4). Moreover, by the guarantee of (α, δ, η)-oracles for Max k-Cover, each run of $\mathcal{A}$ fails with probability at most δ. Thus, by an application of union bound, with probability at least $1 - 2\delta \log n$, est_z is at least $z/(4\alpha)$ for all $z \leq |C(\text{OPT})|$. This in particular implies that the solution returned by EstimateMaxCover is at least $|C(\text{OPT})|/(8\alpha)$. Moreover, since the coverage of a k-cover never increases after applying the "universe reduction" step (i.e. for each $S \subseteq \mathcal{U}$, $|h(C(S))| \leq |S|$) and the estimate returned by the (α, δ, η)-oracle $\mathcal{A}$ is with high probability less than the optimal coverage size, the output of EstimateMaxCover is in $[|C(\text{OPT})|/(8\alpha), |C(\text{OPT})|]$ with probability at least $1 - 4\delta \log n$.

Finally, since EstimateMaxCover runs $\log n \log(\frac{1}{\delta})$ instances of $\mathcal{A}$ with parameter (α, δ, η) in parallel and each instance has m sets, the total space of EstimateMaxCover is $\widetilde{O}(f(m, \alpha))$. □

The *universe reduction* step basically enables us to only focus on the instances of Max k-Cover in which the optimal solution covers a constant fraction of the ground set, namely at least $|\mathcal{U}|/4$ elements. Next, in Section 4, we design an $\widetilde{O}(m/\alpha)$-space (α, δ, η)-oracle for Max k-Cover with $\alpha = \widetilde{\Omega}(1)$, $\eta \geq 4$ and $\delta = O(1/\log n)$, which together with Theorem 3.6 complete the proof of Theorem 3.1. Our (α, δ, η)-oracle for Max k-Cover performs three different subroutines in parallel that together guarantee the required properties of (α, δ, η)-oracles and only use $\widetilde{O}(m/\alpha^2)$ space:

Set sampling based approach. This subroutine which provides the guarantee of (α, δ, η)-oracles when the number of *common* elements is large (see Definition 2.1) is an application of a "multi-layered" variant of set sampling. This subroutine is presented in Section 4.1.

HeavyHitter based approach. We relate the problem of α-estimating/approximating of Max k-Cover to the problem of finding *contributing classes* and *heavy hitters* on properly sampled substreams (see Section 2.2) when the main contribution of an optimal solution of Max k-Cover is due to "large" sets. In particular, this subroutine finds an α-estimation of the optimal coverage size when $\alpha = \Omega(k)$. This approach is presented in Section 4.2.

Element sampling based approach. Finally, we employ element sampling together with a new sampling technique that samples a collection of sets to find a desired estimate of Max k-Cover on instances for which the main contribution to an optimal solution comes from "small" sets. Here, we also take advantage of the structure guaranteed by the multi-layered set sampling on the number of elements in different frequency levels. This subroutine is presented in Section 4.3.

4 (α, δ, η)-ORACLE OF MAX k-COVER

In this section, we design the promised (α, δ, η)-oracle for Max k-Cover. Let OPT denote an optimal solution of Max k-Cover $(\mathcal{U}, \mathcal{F})$. As described in Definition 3.4, the solution returned by a (α, δ, η)-oracle with high probability, is smaller than $|C(\text{OPT})|$ and if $|C(\text{OPT})| \geq |\mathcal{U}|/\eta$, with probability at least $(1-\delta)$, it outputs a value not smaller than $|C(\text{OPT})|/\alpha$. The following Theorem together with Theorem 3.6 prove Theorem 3.1.

Theorem 4.1. *Oracle(α, k) performs a single pass on edge arrival streams and implements a $(\widetilde{O}(\alpha), (\log n \,\text{polylog}(m))^{-1}, \eta)$-oracle of* Max k-Cover$(\mathcal{U}, \mathcal{F})$ *using $\widetilde{O}(m/\alpha^2)$ space.*

Proof. The proof follows from the guarantees of LargeCommon (Theorem 4.4), LargeSet (Theorem 4.8) and SmallSet (Theorem 4.22). The total space of the algorithm is clearly $\widetilde{O}(m/\alpha^2)$ which is the space complexity of the each of subroutines invoked by Oracle. □

To design the promised (α, δ, η)-oracle, we design different subroutines such that each guarantees the properties required by the oracle if certain conditions based on the the size/value of following notions hold.

Common elements. An important property in design of our oracle is whether there exists $\beta \leq \alpha$ such that the number of (βk)-*common* elements is relatively large (see Definition 2.1).

We also take advantage of another useful notion which is a property of a k-cover (though, here we only describe it for optimal k-covers).

Contribution to the optimal coverage. Given the input argument α and a parameter s as defined in Table 2, we define the following notion of *contribution* for the sets in an (optimal) k-cover.

Definition 4.2. For a k-cover $\text{OPT} = \{O_1, \cdots, O_k\}$, we consider an arbitrary ordering of the sets in OPT and define the *contribution* of O_i to $C(\text{OPT})$ as $|O'_i|$ where $O'_i := O_i \setminus \bigcup_{1 \leq j < i} O_i$. Note that O'_i are disjoint and $|\bigcup_{i \in [k]} O'_i| = |C(\text{OPT})| = z$. We (conceptually) define $\text{OPT}_{\text{large}}$ to be the collection of all sets in OPT that contribute more than $z/(\text{s}\alpha)$ to $C(\text{OPT})$ according to O'_is; $\text{OPT}_{\text{large}} = \{O_i \in \text{OPT} \mid |O'_i| \geq z/(\text{s}\alpha)\}$ for $\text{s} < 1$ (as in Table 2). Note that since O'_is are disjoint, $|\text{OPT}_{\text{large}}| \leq \text{s}\alpha$.

$\text{w} = \min\{k, \alpha\}$,	$\text{s} = \frac{9}{5000\sqrt{2\eta \log(\text{s}\alpha)} \log^2(mn)} \cdot \frac{\text{w}}{\alpha}$
$\text{f} = 7\log(mn)$,	$\sigma = \frac{1}{2500 \log^2(mn)}$
$\text{t} = \frac{5000 \log^2(mn)}{\text{s}}$,	$\eta = 4$

Table 2: Values of the parameters used in this section.

Design of (δ, α, η)-oracle of max k-cover. Here we sketch a high-level outline of our (δ, α, η)-oracle for Max k-Cover (refer to Figure 2 for a formal description). In the following cases, $\sigma = \Omega(\frac{1}{\log^2(mn)})$ (as in Table 2).

I. If there exists a $\beta \leq \alpha$ such that $|\mathcal{U}^{\text{cmn}}_{\beta k}| \geq \frac{\sigma\beta|\mathcal{U}|}{\alpha}$. In this case, by Observation 2.4, to approximate the optimal solution size within a factor of $\widetilde{O}(\alpha)$, it suffices to find βk sets that cover $\mathcal{U}^{\text{cmn}}_{\beta k}$ which can be done via set sampling (see Section 4.1).

II. $|C(\mathbf{OPT}_{\text{large}})| \geq \frac{|C(\mathbf{OPT})|}{2}$ and $\forall \beta \leq \alpha, |\mathcal{U}^{\text{cmn}}_{\beta k}| < \frac{\sigma\beta|\mathcal{U}|}{\alpha}$. The subroutine for this case which is presented in Section 4.2, handles the instances of the problem in which $\text{s}\alpha \geq 2k$ or, $\text{s}\alpha < 2k$ and there exists an optimal solution OPT such that $|C(\text{OPT}_{\text{large}})| \geq |C(\text{OPT})|/2$.

Claim 4.3. *If $\text{s}\alpha \geq 2k$, then $|C(OPT_{\text{large}})| \geq |C(OPT)|/2$.*

Proof. Consider the optimal solution OPT and ignore the sets in OPT whose contribution to the coverage is less than $|C(\text{OPT})|/(2k)$. Note that the survived sets belong to $\text{OPT}_{\text{large}}$ and their total coverage is at least $|C(\text{OPT})| - k \cdot \frac{|C(\text{OPT})|}{2k} \geq |C(\text{OPT})|/2$. □

III. $|C(\mathbf{OPT}_{\text{large}})| < \frac{|C(\mathbf{OPT})|}{2}$ and $\forall \beta \leq \alpha, |\mathcal{U}^{\text{cmn}}_{\beta k}| < \frac{\sigma\beta|\mathcal{U}|}{\alpha}$. In this case, the main contribution to the coverage of OPT comes from "small" sets. This enables us to show that if we sample sets with probability $1/\alpha$, then $\widetilde{\Omega}(1/\alpha)$-fraction of sets in OPT survive and with high probability, their coverage is $\widetilde{\Omega}(|C(\text{OPT})|/\alpha)$. In Section 4.3, we show that element sampling method with some new ideas can take care of this case which can only happen when $\text{s}\alpha < 2k$.

Oracle(k, α):
▷ For instances in which $\exists \beta \le \alpha$ s.t. $|\mathcal{U}^{\text{cmn}}_{\beta k}| \ge \frac{\sigma\beta|\mathcal{U}|}{\alpha}$
$\text{sol}_{\text{cmn}} \leftarrow$ LargeCommon(k, α)
if $s\alpha \ge 2k$ **then do**
  ▷ If $s\alpha \ge 2k$, then $|\text{OPT}_{\text{large}}| \ge |\text{OPT}|/2$
  $\text{sol}_{\text{HH}} \leftarrow$ LargeSet(k, α, k)
else do
  ▷ For instances with $s\alpha < 2k$ and $|\text{OPT}_{\text{large}}| \ge |\text{OPT}|/2$
  $\text{sol}_{\text{HH}} \leftarrow$ LargeSet(k, α, α)
  ▷ For instances with $|\text{OPT}_{\text{large}}| < |\text{OPT}|/2$
  $\text{sol}_{\text{small}} \leftarrow$ SmallSet(k, α)
return $\max(\text{sol}_{\text{cmn}}, \text{sol}_{\text{HH}}, \text{sol}_{\text{small}})$

Figure 2: An (α, δ, η)-oracle of Max k-Cover.

4.1 Multi-layered Set Sampling

Here, we first guess the value of β (more precisely, a 2-approximate estimate of β) and then pick βk sets $\mathcal{F}^{\text{rnd}}_{\beta}$ at random and compute their coverage in one pass using $\widetilde{O}(1)$ space. To get the desired space complexity, we use the implementation of set sampling with $O(\log(mn))$ random bits as described in Section A.1.

Theorem 4.4. *Consider an instance $(\mathcal{U}, \mathcal{F})$ of* Max k-Cover. *The* LargeCommon *algorithm uses $\widetilde{O}(1)$ space and if there exists $\beta \le \alpha$ such that $|\mathcal{U}^{\text{cmn}}_{\beta k}| \ge \frac{\sigma\beta|\mathcal{U}|}{\alpha}$, then with high probability, the algorithm returns at least $\sigma|\mathcal{U}|/(6\alpha)$. Moreover, with high probability the output of* LargeCommon *is smaller than the coverage size of an optimal solution of* Max k-Cover$(\mathcal{U}, \mathcal{F})$.

LargeCommon(k, α):
for each $\beta_g \in \{2^i \mid 1 \le i \le \log \alpha\}$ **do in parallel**
  ▷ Perform set sampling in one pass using $\widetilde{O}(1)$ space.
  pick $h_g : \mathcal{F} \to [\frac{cm \log m}{\beta_g k}]$ at random from $\Theta(\log(mn))$-wise independent hash functions
  let DE_g be a $(1 \pm 1/2)$-approximation streaming algorithm of L_0-estimation
  for each (S, e) in the data stream **do**
    if $h_g(S) = 1$ **then** feed e to DE_g ▷ computes $C(\mathcal{F}^{\text{rnd}}_{\beta_g})$
  if $\text{VAL}(\text{DE}_g) \ge \sigma\beta_g|\mathcal{U}|/(4\alpha)$ **then return** $2\text{VAL}(\text{DE}_g)/(3\beta_g)$
return infeasible ▷ $\nexists \beta \in [\alpha]$ s.t. $|\mathcal{U}^{\text{cmn}}_{\beta k}| \ge \frac{\sigma\beta|\mathcal{U}|}{\alpha k}$

Figure 3: A (α, δ, η)-oracle of Max k-Cover that handles the case where the number of common elements is large.

Claim 4.5. *For each $\beta_g \in \{2^i \mid 1 \le i \le \log \alpha\}$, with high probability, $|\mathcal{F}^{\text{rnd}}_{\beta_g}| \le \beta_g k$.*

Lemma 4.6. *If there exists $\beta \le \alpha$ such that $|\mathcal{U}^{\text{cmn}}_{\beta k}| \ge \sigma\beta|\mathcal{U}|/\alpha$, then with high probability the output of* LargeCommon *is at least $\sigma|\mathcal{U}|/(6\alpha)$.*

Proof. Let 2^i be the smallest power of two which is larger than or equal to β; $i := \lceil \log \beta \rceil$. Consider the iteration of LargeCommon in which $\beta_g = 2^i$. Since $2\beta > \beta_g \ge \beta$ and by Observation 2.2,

$$|\mathcal{U}^{\text{cmn}}_{\beta_g k}| \ge |\mathcal{U}^{\text{cmn}}_{\beta k}| \ge \frac{\sigma\beta|\mathcal{U}|}{\alpha} \ge \frac{\sigma\beta_g|\mathcal{U}|}{2\alpha}.$$

Hence, by the guarantee of existing streaming algorithms for L_0-estimation (Theorem 2.12) and set sampling (Lemma 2.3 and A.7), w.h.p., $\text{VAL}(\text{DE}_g) \ge \frac{1}{2} \cdot \frac{\sigma\beta_g|\mathcal{U}|}{2\alpha} = \frac{\sigma\beta_g|\mathcal{U}|}{4\alpha}$. Hence, the estimate returned by the algorithm which is a lower bound on the coverage of the best k sets in $\mathcal{F}^{\text{rnd}}_{\beta_g}$ (see Observation 2.4), w.h.p., is at least $\frac{2}{3} \cdot \frac{1}{\beta_g} \cdot \frac{\sigma\beta_g|\mathcal{U}|}{4\alpha} = \frac{\sigma|\mathcal{U}|}{6\alpha}$.

Moreover, it is straightforward to check that by the guarantee of the streaming algorithm for L_0-estimation (Theorem 2.12), the value returned by LargeCommon with high probability is not more than the actual coverage of the best k-cover in the collection of sampled sets using h_g. □

Lemma 4.7. *If* LargeCommon *returns* **infeasible**, *then with high probability, for all $\beta \le \alpha$, $|\mathcal{U}^{\text{cmn}}_{\beta k}| \le \frac{\sigma \cdot \beta \cdot |\mathcal{U}|}{\alpha}$.*

Proof. Since the algorithm returns **infeasible**, by the guarantee of the $(1 \pm 1/2)$-approximation algorithm for L_0-estimation (Theorem 2.12) and set sampling (Lemma 2.3), for all values of $\beta_g \in \{2^i \mid i \le \log \alpha\}$, with high probability,

$$|\mathcal{U}^{\text{cmn}}_{\beta_g k}| \le |C(\mathcal{F}^{\text{rnd}}_{\beta_g})| \le 2\text{VAL}(\text{DE}_g) < \frac{\sigma \cdot \beta_g \cdot |\mathcal{U}|}{2\alpha}. \quad (1)$$

Now, for any given value $\beta \le \alpha$, consider $\beta_g := 2^{\lceil \log \beta \rceil}$ (i.e. set β_g to be the smallest power of two which is larger than or equal to β). By Observation 2.2, $|\mathcal{U}^{\text{cmn}}_{\beta k}| \le |\mathcal{U}^{\text{cmn}}_{\beta_g k}|$ which together with Eq. 1 imply that $|\mathcal{U}^{\text{cmn}}_{\beta k}| \le \frac{\sigma \cdot \beta_g \cdot |\mathcal{U}|}{2\alpha} \le \frac{\sigma \cdot \beta \cdot |\mathcal{U}|}{\alpha}$. □

Proof of Theorem 4.4. The guarantee on the quality of the output follows from Lemma 4.6. Moreover, by Theorem 2.12, the total amount of space to compute the coverage of each collection $\mathcal{F}^{\text{rnd}}_{\beta_g}$ (via existing L_0-estimation algorithms in streams) is $\widetilde{O}(1)$. Hence, the total space to compute the coverage of all $\log \alpha$ collections considered in LargeCommon is $\widetilde{O}(1)$. □

4.2 Heavy Hitters and Contributing Classes: $|C(\text{OPT}_{\text{large}})| \ge |C(\text{OPT})|/2$.

In this section, we show that if there exists an optimal solution OPT of Max k-Cover$(\mathcal{U}, \mathcal{F})$ such that the main contribution in the coverage of OPT is due to *large* sets, which

are formally defined to be the sets whose contribution to $C(\text{OPT})$ is at least $|C(\text{OPT})|/(s\alpha)$, then we can approximate the optimal coverage size within a factor of $\widetilde{O}(\alpha)$ by detecting $\widetilde{\Omega}(\frac{\alpha^2}{m})$-HeavyHitters in properly sampled substreams. Following is the main result of this section.

Theorem 4.8. *Consider an instance $(\mathcal{U}, \mathcal{F})$ of* Max k-Cover. *In a single pass, LargeSet uses $\widetilde{O}(m/\alpha^2)$ space and if the optimal coverage size of the instance is $\Omega(|\mathcal{U}|)$, then with probability at least $1 - (\log n \,\text{polylog}(m))^{-1}$, it returns at least $\widetilde{\Omega}(|\mathcal{U}|/\alpha)$. Moreover, with high probability, the estimate returned by LargeSet is smaller than the optimal coverage size.*

We defer the proof of Theorem 4.8 to Section B. In this section, we prove the same guarantees on the performance of a simplified variant of LargeSet, LargeSetSimple, when $\mathcal{U}$ contains no "common" elements (wee will define them formally later in this section) which essentially presents the main technical ideas.

Partitioning sets into supersets. We partition the sets of $\mathcal{F}$ *randomly* into $\frac{cm\log m}{\mathsf{w}}$ *supersets* $\mathcal{Q} := \{\mathcal{D}_1, \cdots, \mathcal{D}_{\frac{cm\log m}{\mathsf{w}}}\}$ via a hash function $h : \mathcal{F} \to [(cm\log m)/\mathsf{w}]$ chosen from a family of $\Theta(\log(mn))$-wise independent hash functions. More precisely, each set $S \in \mathcal{F}$ belongs to the superset $\mathcal{D}_{h(S)}$.

The parameter w denotes the desired upper bound on the maximum number of sets in a superset in $\mathcal{Q}$ defined by h and is set to $\min(\alpha, k)$. In fact, given w, we define h to be a function picked uniformly at random from a family of $\Theta(\log(mn))$-wise independent hash functions $\{\mathcal{F} \to [(cm\log m)/\mathsf{w}]\}$.

Claim 4.9. *W.h.p., no superset in $\mathcal{Q}$ has more than w sets.*

Claim 4.10. *With high probability, for each $e \in \mathcal{U} \setminus \mathcal{U}^{\text{cmn}}_{\mathsf{w}}$ and $\mathcal{D} \in \mathcal{Q}$, the number of sets in $\mathcal{D}$ that contain e is at most f where $\mathsf{f} = \Theta(\log(mn))$.*

This implies if $\mathcal{U}^{\text{cmn}}_{\mathsf{w}} = \emptyset$, to get an $\widetilde{O}(\alpha)$-approximation of Max k-Cover$(\mathcal{U}, \mathcal{F})$, it suffices to find a superset whose total size of its sets is $\widetilde{\Omega}(1/\alpha)$ times the optimal coverage size. Now, we are ready to exploit the results on F_2-heavy hitters and F_2-contributing classes mentioned in Section 2.2 to describe our (α, δ, η)-oracle for Max k-Cover assuming $\mathcal{U}^{\text{cmn}}_{\mathsf{w}} = \emptyset$. Later in Section B, we show how to remove this assumption by performing our algorithm on a set of sampled elements in $\mathcal{U}$ instead.

Partitioning supersets by their total size. First, setting $z = |C(\text{OPT})|$, we partition the supersets in $\mathcal{Q}$ (conceptually) according to the total size of their sets into $O(\log\alpha)$ classes as follows:

$$\mathcal{Q}_0 = \{\mathcal{D} \mid \sum_{S\in\mathcal{D}} |S| \geq \frac{z}{2}\}, \tag{2}$$

$$\mathcal{Q}_i = \{\mathcal{D} \mid \frac{z}{2^{i+1}} \leq \sum_{S\in\mathcal{D}} |S| < \frac{z}{2^i}\}, \quad \forall i \in [1, \log(\alpha))\ (3)$$

$$\mathcal{Q}_{\text{small}} = \{\mathcal{D} \mid \sum_{S\in\mathcal{D}} |S| < z/\alpha\}. \tag{4}$$

Further, let n_i denote the number of supersets in $\mathcal{Q}_i$; $n_i = |\mathcal{Q}_i|$. Next, we define the vector $\vec{v}$ of size $(cm\log m)/\mathsf{w}$ such that $\vec{v}[i] = \sum_{S\in\mathcal{D}_i} |S|$ denotes the total size of the sets in $\mathcal{D}_i$. In the following, we show that a subset of supersets with large total size form an $\widetilde{\Omega}(\frac{m}{\alpha^2})$-contributing class of $F_2(\vec{v})$ and any superset in this $\widetilde{\Omega}(\frac{m}{\alpha^2})$-contributing class is an *α-approximate k-cover of $(\mathcal{U}, \mathcal{F})$*.

We consider the following two cases depending on whether the coordinates corresponding to small supersets, $\mathcal{Q}_{\text{small}}$, contribute to $F_2(\vec{v})$; $F_2(\vec{v}_{\text{small}}) \geq F_2(\vec{v})/2$ where $\vec{v}_{\text{small}}$ denotes the vector $\vec{v}$ restricted to the coordinates corresponding to supersets in $\mathcal{Q}_{\text{small}}$.

Case 1: Supersets with total size less than z/α contribute to $F_2(\vec{v})$. This implies that

$$F_2(\vec{v}) \leq 2F_2(\vec{v}_{\text{small}}) \leq \frac{2cm\log m}{\mathsf{w}} \cdot \frac{z^2}{\alpha^2}. \tag{5}$$

Claim 4.11. *If $F_2(\vec{v}_{\text{small}}) \geq F_2(\vec{v})/2$, then there exists an $\widetilde{\Omega}(\frac{\alpha^2}{m})$-contributing class $\mathcal{Q}_{i^*}$ of $F_2(\vec{v})$ for an index $i^* < \log(s\alpha)$.*

Proof. Since each set in $\text{OPT}_{\text{large}}$ has contribution at least $\frac{z}{s\alpha}$ to $C(\text{OPT})$, sets in $\text{OPT}_{\text{large}}$ land in one of $\mathcal{Q}_0, \cdots, \mathcal{Q}_{\log(s\alpha)-1}$. Moreover, since $\text{OPT}_{\text{large}}$ has coverage at least $z/2$,

$$\sum_{i=0}^{\log(s\alpha)-1} n_i \cdot \frac{z}{2^i} \geq \sum_{O_i\in\text{OPT}_{\text{large}}} |O_i| \geq |C(\text{OPT}_{\text{large}})| \geq z/2,$$

which implies that there exists an index $i^* < \log(s\alpha)$ such that $n_{i^*} \geq 2^{i^*}/(2\log(s\alpha))$. Hence, $\mathcal{Q}_{i^*}$ is an $\widetilde{\Omega}(\frac{\alpha^2}{m})$-contributing class of $F_2(\vec{v})$:

$$\begin{aligned}
|\mathcal{Q}_{i^*}| \cdot (\frac{z}{2^{i^*+1}})^2 &\geq \frac{2^{i^*}}{2\log(s\alpha)} \cdot \frac{z^2}{4(2^{i^*})^2} \\
&\geq \frac{\mathsf{w}}{2cm\log m} \cdot \alpha^2 \cdot \frac{1}{2^{i^*+3}\log(s\alpha)} \cdot F_2(\vec{v}) \ \triangleright \text{By (5)} \\
&\geq (\frac{\mathsf{w}}{s\alpha} \cdot \frac{1}{8c\log(s\alpha)\log m}) \cdot \frac{\alpha^2}{m} \cdot F_2(\vec{v})
\end{aligned}$$

More formally, since $\frac{\mathsf{w}}{s\alpha} = \widetilde{\Omega}(1)$ (see Table 2), $\mathcal{Q}_{i^*}$ is a ϕ_1-contributing class of $F_2(\vec{v})$ where

$$\phi_1 = (\frac{\mathsf{w}}{s\alpha} \cdot \frac{1}{8c\log(s\alpha)\log m}) \cdot \frac{\alpha^2}{m} = \widetilde{\Omega}(\frac{\alpha^2}{m}). \tag{6}$$

□

Hence, by Theorem 2.11, a superset of total size at least $\frac{2}{3} \cdot \frac{1}{2} \cdot \frac{z}{s\alpha}$ will be identified by the subroutine F_2-Contributing$(\phi_1, s\alpha)$ using $\widetilde{O}(m/\alpha^2)$ space.

Remark 4.12. *Recall that in order to find a coordinate in a ϕ-contributing class R_{t^*}, F_2-Contributing subsamples the stream proportional to $1/|R_{t^*}|$ (so that only $O(1)$ coordinates of R_{t^*} survive) and then with high probability any survived coordinate of R_{t^*} becomes a $\widetilde{\Omega}(\phi)$-HeavyHitter in the sampled substream. However, here we show that there exists a ϕ-contributing class R_{t^*} whose intersection with OPT_{large} is a ϕ-contributing class of coordinates too. Hence, it suffices to only search for a coordinate in a ϕ-contributing class of size at most $|OPT_{\text{large}}| \le s\alpha$.*

We emphasis that bounding the size of a ϕ-contributing class is not required for the simplified case of this section (i.e., $\mathcal{U}^{\text{cmn}}_{w} = \emptyset$). However, since we estimate the coverage size of a superset by the total size of its sets, in Section B where we solve the general case (i.e., $\mathcal{U}^{\text{cmn}}_{w} \neq \emptyset$) it is crucial to only consider ϕ-contributing classes of small size. Otherwise, in the superset returned by our algorithm, the gap between its actual coverage size and total size of the sets in the superset can be very large.

Case 2. Supersets with coverage less than z/α do not contribute to $F_2(\vec{v})$.

Claim 4.13. *If $F_2(\vec{v}_{\text{small}}) < F_2(\vec{v})/2$, then there exists an $\widetilde{\Omega}(1)$-contributing class $\mathcal{Q}_{i^*}$ of $F_2(\vec{v})$ for an index $i^* < \log \alpha$.*

Proof. In this case, since supersets in $\mathcal{Q}_{\text{small}}$ are not contributing, there exists an index $i^* < \log(\alpha)$ (note that we consider all classes $\mathcal{Q}_0, \cdots, \mathcal{Q}_{\log \alpha - 1}$ in this case) such that

$$n_{i^*} \cdot \left(\frac{z}{2^{i^*}}\right)^2 \ge \frac{F_2(\vec{v})}{2 \log \alpha};$$

in other words, $\mathcal{Q}_{i^*}$ is a ϕ_2-contributing class of $F_2(\vec{v})$ where $\phi_2 = (\frac{1}{2 \log \alpha})$. □

Note that, by Theorem 2.11, a superset of total size at least $\frac{2}{3} \cdot \frac{1}{2} \cdot \frac{z}{\alpha}$ will be identified by F_2-Contributing$(\phi_2, \frac{cm \log m}{w})$ using $\widetilde{O}(1)$ space.

Lemma 4.14. *If $|C(OPT)| \ge \frac{|\mathcal{U}|}{\eta}$, then w.p. at least $1 - 1/(3 \log n \log^c m)$, the estimate returned by LargeSetSimple with parameters $(\mathcal{V} = \mathcal{U}, w, \text{thr}_1 = \frac{|\mathcal{U}|}{\eta s \alpha}, \text{thr}_2 = \frac{|\mathcal{U}|}{\eta \alpha})$ has coverage at least $|\mathcal{U}|/(3f\eta\alpha) = \widetilde{\Omega}(|\mathcal{U}|/\alpha)$.*

Proof. By Theorem 2.11, w.p. at least $1 - 2/(9 \log n \log^c m)$, the algorithm returns a superset whose total size is at least $\frac{2}{3} \cdot \frac{|\mathcal{U}|}{2\eta\alpha} \ge \frac{|\mathcal{U}|}{3\eta\alpha}$ (in fact, if it is in Case 1, then the estimate is at least $\frac{|\mathcal{U}|}{3s\alpha\eta}$). Then, by Claim 4.10 and assumption $\mathcal{U}^{\text{cmn}}_{w} = \emptyset$, the coverage of the reported superset is at least $\frac{1}{f} \cdot \frac{|\mathcal{U}|}{3\eta\alpha}$. □

LargeSetSimple$(\mathcal{V}, w, \text{thr}_1, \text{thr}_2)$:
▷ **Input:** w is a bound on the number of sets in a superset
▷ **Parameters:** $\phi_1 = \widetilde{\Omega}(\alpha^2/m)$ and $\phi_2 = \widetilde{\Omega}(1)$
▷ For Case 1:
$\text{Cntr}_{\text{small}} \leftarrow$ instance of F_2-Contributing$(\phi_1, s\alpha)$
▷ For Case 2:
$\text{Cntr}_{\text{large}} \leftarrow$ instance of F_2-Contributing$(\phi_2, \frac{cm \log m}{w})$
pick $h : \mathcal{F} \to [(cm \log m)/w]$ from $\Theta(\log(mn))$-wise independent hash functions
for each (S, e) in the data stream **do**
  if $e \in \mathcal{V}$ **then feed** $h(S)$ to both $\text{Cntr}_{\text{small}}$ and $\text{Cntr}_{\text{large}}$
▷ output(Cntr) returns $(1 \pm 1/2)$-estimate of the frequencies
▷ $\forall i \in$ output(Cntr), $\tilde{v}_i$ denotes the estimated frequency of i
if there exists $i \in \text{output}(\text{Cntr}_{\text{small}})$ such that $\tilde{v}_i \ge \frac{1}{2} \cdot \text{thr}_1$
  return $2\tilde{v}_i/(3f)$
if there exists $i \in \text{output}(\text{Cntr}_{\text{large}})$ such that $\tilde{v}_i \ge \frac{1}{2} \cdot \text{thr}_2$
  return $2\tilde{v}_i/(3f)$
return infeasible

Figure 4: An (α, δ, η)-oracle of Max k-Cover that handles the case in which the majority of the coverage in an optimal solution is by the sets whose coverage contributions are at least $1/(s\alpha)$ fraction of the optimal coverage size.

Lemma 4.15. *The amount of space used by LargeSetSimple is $\widetilde{O}(m/\alpha^2)$.*

Proof. By Theorem 2.11, the amount of space to perform $\text{Cntr}_{\text{small}}$ and $\text{Cntr}_{\text{large}}$ as defined in Figure 4 is respectively $\widetilde{O}(1/\phi_1) = \widetilde{O}(m/\alpha^2)$ and $\widetilde{O}(1/\phi_2) = \widetilde{O}(1)$. □

4.3 Element Sampling: $|C(\mathbf{OPT}_{\text{large}})| < \frac{|C(\mathbf{OPT})|}{2}$

We design an (α, δ, η)-oracle of Max k-Cover with the desired parameters for the case $|C(\text{OPT}_{\text{small}})| > |C(\text{OPT}_{\text{large}})|$. Intuitively speaking, in this case, after sampling each set in $\mathcal{F}$ with probability $\widetilde{\Theta}(\frac{1}{\alpha})$ still we can find $O(\frac{k}{\alpha})$ sets whose coverage size is at least $\widetilde{\Omega}(\frac{|C(\text{OPT})|}{\alpha})$. As proved in Claim 4.3, if $\alpha = \widetilde{\Omega}(k)$, then the main contribution to the coverage of OPT is due to $\text{OPT}_{\text{large}}$ and we can α-approximate the optimal coverage size by LargeSet. Hence, in this section we assume that $\alpha = \widetilde{O}(k)$. Moreover, throughout this section we assume that for all $\beta \le \alpha$, $|\mathcal{U}^{\text{cmn}}_{\beta k}| < \frac{\sigma \cdot \beta \cdot |\mathcal{U}|}{\alpha}$ (otherwise, our multi-layered set sampling approach described in Section 4.1 returns an α-approximation of the optimal coverage size).

Lemma 4.16. *Consider an instance of Max k-Cover $(\mathcal{U}, \mathcal{F})$. Suppose that $\mathcal{D}$ is a collection of k disjoint sets with coverage z such that no $S \in \mathcal{D}$ has size more than $z/(s\alpha)$ where $s < 1$*

and $\mathrm{s} = \widetilde{\Omega}(1)$. *Let's assume that* $\mathcal{D}_{\mathsf{smp}} := \mathcal{D} \cap \mathcal{M}$ *where each* $S \in \mathcal{F}$ *survives in* $\mathcal{M}$ *with probability* $c/(\mathrm{s}\alpha)$ *where* $c > 1$ *is a fixed constant. Then, with probability at least* $(1 - 6/c)$, $\mathcal{D}_{\mathsf{smp}}$ *has size at most* $(2ck)/(\mathrm{s}\alpha)$ *and covers at least* $(cz)/(2\mathrm{s}\alpha)$ *elements.*

Proof. Let $\mathcal{D} = \{S'_1, \cdots, S'_k\}$ and for each i, let X_i to be the random variable corresponding to S'_i such that $X_i = |S'_i|$ if $S'_i \in \mathcal{D}_{\mathsf{smp}}$ and zero otherwise.

Claim 4.17. $\mathbf{E}[X_i] = \frac{c}{\mathrm{s}\alpha} \cdot |S'_i|$ *and* $\mathbf{Var}[X_i] \leq \frac{c}{\mathrm{s}\alpha} \cdot |S'_i|^2$.

Next, we define $X := X_1 + \cdots + X_k$. Note that $\mathbf{E}[X] = (cz)/(\mathrm{s}\alpha)$ and, by the pairwise independence of X_is and the assumption that $|S'_i| \leq z/(\mathrm{s}\alpha)$,

$$\mathbf{Var}[X] \leq \frac{c}{\mathrm{s}\alpha} \cdot \sum_{i=1}^{k} |S'_i|^2 \leq \frac{c}{\mathrm{s}\alpha} \cdot \mathrm{s}\alpha \cdot (\frac{z}{\mathrm{s}\alpha})^2 = c \cdot (\frac{z}{\mathrm{s}\alpha})^2,$$

Finally, applying Chebyshev inequality,

$$\Pr[X < \frac{cz}{2\mathrm{s}\alpha}] = \Pr[X < (\frac{cz}{\mathrm{s}\alpha} - \frac{\sqrt{c}}{2} \cdot (\frac{\sqrt{c}}{\mathrm{s}\alpha} \cdot z)] < 4/c.$$

Hence, with probability at least $1 - 4/c$, $\mathcal{D}_{\mathsf{smp}}$ covers at least $(cz)/(2\mathrm{s}\alpha)$ elements.

Next, we show that with probability at least $1 - 2/c$, $0 < |\mathcal{D}_{\mathsf{smp}}| < (2ck)/(\mathrm{s}\alpha)$. For each i, let Y_i denote the random variable corresponding to S_i which is equal to one if $S_i \in \mathcal{D}_{\mathsf{smp}}$ and zero otherwise.

Claim 4.18. $\mathbf{E}[Y_i] = (c/\mathrm{s}\alpha)$ *and* $\mathbf{Var}[Y_i] \leq (c/\mathrm{s}\alpha)$.

We define $Y = Y_1 + \cdots + Y_\ell$ which denotes the size of $\mathcal{D}_{\mathsf{smp}}$. Then by pairwise independence of Y_is, $\mathbf{E}[Y] = (ck)/(\mathrm{s}\alpha)$ and $\mathbf{Var}[Y] \leq (ck)/(\mathrm{s}\alpha)$. Applying Chebyshev inequality ($\Pr[|Y - \mathbf{E}[Y]| \geq t\mathbf{Var}[Y]] \leq 1/(t^2\mathbf{Var}[Y])$), with probability at least $1 - (\mathrm{s}\alpha)/(ck) \geq 1 - 2/c$ (since in this case, $\alpha \leq 2k/\mathrm{s}$), $0 < |\mathcal{D}_{\mathsf{smp}}| < (2ck)/(\mathrm{s}\alpha)$.

Hence, with probability at least $(1 - 6/c)$, $\mathcal{D}_{\mathsf{smp}}$ is a subset of size at most $(2ck)/(\mathrm{s}\alpha)$ that covers at least $(cz)/(2\mathrm{s}\alpha)$ elements. □

Corollary 4.19. *Consider an instance* $(\mathcal{U}, \mathcal{F})$ *of* Max k-Cover *and let OPT be an optimal solution of this instance such that* $|C(OPT_{\mathsf{small}})| \geq \frac{1}{2} \cdot |C(OPT)| \geq |\mathcal{U}|/(2\eta)$. *Moreover, let* $\mathcal{M} \subset \mathcal{F}$ *be a collection of* $\widetilde{O}(|\mathcal{F}|/\alpha)$ *pairwise independent sets picked uniformly at random such that each* $S \in \mathcal{F}$ *belongs to* $\mathcal{M}$ *with probability* $\frac{18}{\mathrm{s}\alpha}$. *With probability at least* 2/3, Max $(\frac{36k}{\mathrm{s}\alpha})$-Cover$(\mathcal{U}, \mathcal{M})$ *has an optimal solution with coverage size at least* $\frac{9|\mathcal{U}|}{\mathrm{s}\cdot\alpha\cdot\eta}$.

Proof. By definition of $\mathrm{OPT}_{\mathsf{small}}$, for each $O \in \mathrm{OPT}_{\mathsf{small}}$, the contribution of O to OPT (i.e. O') is at most $z/(\mathrm{s}\alpha)$ (see Definition 4.2). Then, the result follows from an application of Lemma 4.16 on collection $\mathcal{D} := \{O' \mid O \in \mathrm{OPT}_{\mathsf{small}}\}$ by setting $c = 18$ and $z = |\mathrm{OPT}_{\mathsf{small}}| \geq |\mathcal{U}|/(2\eta)$. □

Next, we show that we can perform "element sampling" and find an $\widetilde{O}(1)$-approximation of Max $(\frac{36k}{\mathrm{s}\alpha})$-Cover of the specified instance in Corollary 4.19, $(\mathcal{U}, \mathcal{M})$, in one pass and using $\widetilde{O}(m/\alpha^2)$ space. To this end, first we compute the space complexity of $(\mathcal{L}, \mathcal{F})$ where $\mathcal{L} \subseteq \mathcal{U}$ is a subset of size $\widetilde{O}(k)$ which is picked by element sampling.

Lemma 4.20. *Suppose that coverage size of an optimal solution of* Max $(\frac{k}{\alpha})$-Cover$(\mathcal{U}, \mathcal{F})$ *is* $|\mathcal{U}|/\gamma = \widetilde{\Omega}(|\mathcal{U}|/\alpha)$. *Let* $\mathcal{L} \subset \mathcal{U}$ *be a collection of elements of size* $\widetilde{O}(\frac{k\gamma}{\alpha})$ *picked uniformly at random. With high probability, the total amount of space to store the set system* $(\mathcal{L}, \mathcal{F})$ *is* $\widetilde{O}(m/\alpha)$.

Proof. Recall that $(\mathcal{U}, \mathcal{F})$ has the property that for all $\beta \leq \alpha$, $|\mathcal{U}^{\mathsf{cmn}}_{\beta k}| < \sigma\beta|\mathcal{U}|/\alpha$ (otherwise, the result of Section 4.1 can be applied). Next, we (conceptually) partition the elements in $\mathcal{U}$ into $\log\alpha + 1$ groups as follows:

$$\mathcal{W}_0 = \mathcal{U} \setminus \mathcal{U}^{\mathsf{cmn}}_{\alpha k}, \text{ and } \mathcal{W}_i = \mathcal{U}^{\mathsf{cmn}}_{(\frac{\alpha}{2^{i-1}})k} \setminus \mathcal{U}^{\mathsf{cmn}}_{(\frac{\alpha}{2^i})k} \quad \forall i \in [\log\alpha].$$

Note that $|\mathcal{W}_0| \leq |\mathcal{U}|$ and for each $i \in [\log\alpha]$, $|\mathcal{W}_i| \leq \sigma|\mathcal{U}|/2^{i-1}$. Since each element $e \in \mathcal{U}$ survives in $\mathcal{L}$ with probability $\widetilde{O}(\frac{k\gamma}{\alpha|\mathcal{U}|})$, w.h.p., for each $i \in [\log\alpha]$, $|\mathcal{W}_i \cap \mathcal{L}| = \widetilde{O}(1 + \frac{\sigma\cdot\gamma\cdot k}{\alpha 2^{i-1}})$. Furthermore, since each element in $\mathcal{W}_i$ appears in at most $\widetilde{O}(\frac{2^i m}{\alpha k})$ sets in $\mathcal{F}$, the total amount of space required to store $(\mathcal{L}, \mathcal{F})$ is at most

$$\begin{aligned}\mathbf{S}(\mathcal{L}, \mathcal{F}) &= \sum_{i=0}^{\log\alpha} |\mathcal{W}_i \cap \mathcal{L}| \cdot \max_{e\in\mathcal{W}_i} \mathrm{freq}(e) \\ &= \widetilde{O}(\frac{k\gamma}{\alpha}) \cdot \widetilde{O}(\frac{m}{\alpha k}) + \sum_{i=1}^{\log\alpha} \widetilde{O}(1 + \frac{\sigma\gamma k}{\alpha 2^{i-1}}) \cdot \widetilde{O}(\frac{2^i m}{\alpha k}) \\ &= \widetilde{O}(\frac{\gamma m}{\alpha^2}) + \sum_{i=1}^{\log\alpha} \widetilde{O}(\frac{2^i m}{\alpha k} + \frac{\sigma\gamma m}{\alpha^2}) = \widetilde{O}(m/\alpha).\end{aligned}$$

□

Next, we show that after subsampling the sets by a factor of $\widetilde{\Theta}(1/\alpha)$, we can save another factor of $\widetilde{\Omega}(\alpha)$ in the space complexity; in other words, $(\mathcal{L}, \mathcal{M})$ uses $\widetilde{O}(\frac{m}{\alpha^2})$ space. Note that since $k\alpha$ may be as large as $\widetilde{\Omega}(m)$ we cannot hope to show directly that each element in $\mathcal{W}_i$ appears in at most $\widetilde{O}(\frac{m}{2^i\alpha k})$. However, we can show that the total size of the intersection of all sets in $\mathcal{M}$ with $\mathcal{L}$ is $\widetilde{O}(\frac{m}{\alpha^2})$ using the properties of the max cover instance.

Lemma 4.21. *Suppose that the coverage size of an optimal solution of* Max $(\frac{k}{\alpha})$-Cover$(\mathcal{U}, \mathcal{F})$ *is* $|\mathcal{U}|/\gamma = \widetilde{\Omega}(|\mathcal{U}|/\alpha)$. *Let* $\mathcal{L} \subset \mathcal{U}$ *be a collection of elements of size* $\widetilde{O}(\frac{k\gamma}{\alpha})$ *picked uniformly at random and let* $\mathcal{M} \subset \mathcal{F}$ *be a collection of sets of size* $\widetilde{O}(m/\alpha)$ *picked uniformly at random. With high probability, the total amount of space required to store the set system* $(\mathcal{L}, \mathcal{M})$ *is* $\widetilde{O}(m/\alpha^2)$.

Proof. First note that since an optimal $(\frac{k}{\alpha})$-cover of $(\mathcal{U}, \mathcal{F})$ has coverage $|\mathcal{U}|/\gamma$, with high probability, for each set $S \in \mathcal{M}$, $|S \cap \mathcal{L}| = \widetilde{O}(k/\alpha)$. Moreover, by Lemma 4.20, the size of the intersection of all sets in $\mathcal{F}$ with $\mathcal{L}$ is $\widetilde{O}(m/\alpha)$. Next, we (conceptually) partition the sets of $\mathcal{F}$ into $O(\log k)$ groups based on their intersection size with $\mathcal{L}$ as follows ($c = \widetilde{O}(1)$):

$$\mathcal{Q}_i = \{S \in \mathcal{F} \mid \frac{1}{2^i} \cdot \frac{ck}{\alpha} \leq |S \cap \mathcal{L}| < \frac{1}{2^{i-1}} \cdot \frac{ck}{\alpha}\}, \ \forall 1 \leq i \leq \log k$$

Since the total size of the intersection of all sets with the sampled set $\mathcal{L}$ is w.h.p. $\widetilde{O}(m/\alpha)$, for each $i \leq \log k$, $|\mathcal{Q}_i| \leq \frac{\widetilde{O}(m/\alpha)}{(ck)/(2^i\alpha)} = \widetilde{O}(\frac{2^i \cdot m}{k})$. Since we have the assumption that $\frac{m}{k\alpha} \geq 1$ (we took care of the case $m < k\alpha$ in the first line of EstimateMaxCover separately), w.h.p., for each $\mathcal{Q}_i$, $|\mathcal{Q}_i \cap \mathcal{M}| = \widetilde{O}(\frac{2^i m}{k\alpha})$. Hence, the total amount of space to store $(\mathcal{L}, \mathcal{M})$ is at most

$$\sum_{i=1}^{\log k} \frac{1}{2^{i-1}} \cdot \frac{ck}{\alpha} \cdot \widetilde{O}(\frac{2^i m}{k\alpha}) = \widetilde{O}(\frac{m}{\alpha^2}).$$

□

SmallSet(k, α):
▷ estimate of the optimal coverage of $\widetilde{O}(\frac{k}{\alpha})$-cover
for each $\gamma_g \in \{2^{-i} \mid i \in [\log \alpha]\}$ **do in parallel**:
  repeat $\log n$ times **in parallel**:
    $\mathcal{M} \leftarrow$ uniformly selected samples of size $\widetilde{\Theta}(m/\alpha)$ from $\mathcal{F}$
    $\mathcal{L} \leftarrow$ uniformly selected samples of size $\widetilde{\Theta}(\gamma_g \cdot (\frac{k}{\alpha}))$ from $\mathcal{U}$
    ▷ S$(\mathcal{L}, \mathcal{M})$ stores $(\mathcal{L}, \mathcal{M})$
    initialize S$(\mathcal{L}, \mathcal{M})$ to be an empty set
    for each (S, e) in the data stream **do**
      if $S \in \mathcal{M}$ and $e \in \mathcal{L}$ **then**
        add (S, e) to S$(\mathcal{L}, \mathcal{M})$
      if S$(\mathcal{L}, \mathcal{M}) > \widetilde{O}(m/\alpha^2)$ **then**
        terminate
  $\text{sol}_{\gamma_g} \leftarrow \max_{\mathcal{L}, \mathcal{M}}\{O(1)$-approximation of the coverage of Max $(\frac{36k}{s\alpha})$-Cover(S$(\mathcal{L}, \mathcal{M}))\}$
return $\max_{\gamma_g}\{(\frac{|\mathcal{U}|}{\gamma_g(k/\alpha)} \cdot \text{sol}_{\gamma_g}) \mid \text{sol}_{\gamma_g} = \widetilde{\Omega}(k/\alpha)\}$

Figure 5: A single pass streaming algorithm that estimates the optimal coverage size of Max k-Cover$(\mathcal{U}, \mathcal{F})$ **within a factor of** $\widetilde{O}(\alpha)$ **in** $\widetilde{O}(\frac{m}{\alpha^2})$ **space.**

Theorem 4.22. *If* $|C(OPT_{\text{small}})| \geq |\mathcal{U}|/(2\eta)$ *and for all* $\beta \leq \alpha$, $|\mathcal{U}^{\text{cmn}}_{\beta k}| < \frac{\sigma \cdot \beta \cdot |\mathcal{U}|}{\alpha}$, *then with high probability,* SmallSet *outputs an* $\widetilde{O}(\alpha)$*-approximation of the size of an optimal solution of* Max k-Cover$(\mathcal{U}, \mathcal{F})$ *in* $\widetilde{O}(m/\alpha^2)$ *space.*

Proof. By Corollary 4.19, for any sampled collection of sets $\mathcal{M}$ of size $\widetilde{\Theta}(m/\alpha)$ (as in SmallSet), with probability at least 2/3, there exists a subset of size at most $(\frac{36k}{s\alpha})$ in $\mathcal{M}$ whose coverage is $9|\mathcal{U}|/(s \cdot \alpha \cdot \eta) = |\mathcal{U}|/\gamma$. Moreover, by the guarantee of element sampling, Lemma 2.5, when $\gamma/2 < \gamma_g \leq \gamma$, an $O(1)$-approximate solution of Max k-Cover$(\mathcal{L}, \mathcal{M})$ w.h.p. is an $O(1)$-approximate solution of Max k-Cover$(\mathcal{U}, \mathcal{M})$. Hence, with probability $1 - n^{-1}$, in at least one of the $(\log n)$ instances with the desired γ_g, sol_{γ_g} has coverage $\widetilde{\Omega}(\frac{|\mathcal{U}|}{\gamma} \cdot \gamma_g \cdot \frac{(k/\alpha)}{|\mathcal{U}|}) = \widetilde{\Omega}(k/\alpha)$ over the sampled elements $\mathcal{L}$. Note that we need to scale sol_{γ_g} by a factor of $\widetilde{\Theta}(|\mathcal{U}|/(\gamma_g \cdot \frac{k}{\alpha}))$ to reflect its coverage on $\mathcal{U}$.

Further, by Lemma 4.21, the amount of space required to store each $(\mathcal{L}, \mathcal{M})$ with high probability is $\widetilde{O}(\frac{m}{\alpha^2})$ and since SmallSet stores $\widetilde{O}(1)$ different instances $(\mathcal{L}, \mathcal{F})$, the total space of the algorithm is $\widetilde{O}(\frac{m}{\alpha^2})$. □

To complete the SmallSet is indeed an (α, δ, η)-oracle with the desired parameters, we need to show that it never overestimates the optimal coverage size.

Lemma 4.23. *The output of* SmallSet *with high probability is not larger than the optimal coverage size of* Max k-Cover$(\mathcal{U}, \mathcal{F})$.

Proof. The proof follows from the fact that if the optimal coverage size Max $(\widetilde{O}(\frac{k}{\alpha}))$-Cover$(\mathcal{L}, \mathcal{M})$ is not $\widetilde{\Omega}(|\mathcal{U}|/(\alpha \cdot \gamma_g))$, then with high probability in none of the $\log n$ iterations $\text{sol}_{\gamma_g} = \widetilde{\Omega}(k/\alpha)$. Hence, SmallSet with high probability does not overestimate the size of an optimal $\widetilde{O}(k/\alpha)$-cover in $(\mathcal{U}, \mathcal{F})$. □

5 LOWER BOUND FOR ESTIMATING MAXIMUM k-COVERAGE IN EDGE ARRIVAL STREAMS

By the result of [12], it is known that estimating the size of an optimal coverage of Max k-Cover within a factor of two requires $\Omega(m)$ space. Their argument relies on a reduction from Set Disjointness problem and implies the mentioned bound for 1-cover instances. In the following, we generalize their approach and provide lower bounds for the all range of approximation guarantees α smaller than $\sqrt{m}$. We remark that both our lower bound result and the lower bound result of [12] are basically similar to the lower bound of L_∞ and L_k estimation first proved in [5, 10].

The lower bound result we explain in this section is based on the well-known r-player Set Disjointness problem with unique set intersection promise which has been studied extensively in communication complexity (e.g. [11, 16, 24]). The setting of the problem is as follows: There are r players and each has a set $T_i \subseteq [m]$. The promise is that the input is in one of the following forms:

- **No Case:** There is a unique element $j \in [m]$ such that for all $i \leq r$, $j \in T_i$.
- **Yes Case:** All sets are pair-wise disjoint.

Moreover, a round of communication consists of each player i sending a message to player $i+1$ in order from $i = 1$ to $r-1$. The goal is that at end of a single round, player r be able to correctly output whether the input belongs to the family of **Yes** instances or **No** instances. Chakrabarti et al. [16] showed the following tight lower bound on the one-way communication complexity of the r-player Set Disjointness problem with unique set intersection promise.

Theorem 5.1 (From [16]). *Any randomized one-way protocol that solves r-player* Set Disjointness(m) *with success probability at least $2/3$ requires $\Omega(m/r)$ bits of communication.*

We remark that the same $\Omega(m/r)$ communication lower bound was later proved for the general model (i.e. with multiple rounds) by Gronemeier [24]. However, for our application, the lower bound on the one-way communication model suffices.

Corollary 5.2. *Any single-pass streaming algorithm that solves r-player* Set Disjointness(m) *with success probability at least $2/3$ consumes $\Omega(m/r^2)$ space.*

Next, we sketch a reduction from r-player Set Disjointness(m) to Max k-Cover with m sets such that an α-approximation protocol of Max k-Cover solves the corresponding instance of r-player Set Disjointness(m). To this end, consider an arbitrary instance $\mathcal{I}$ of α-player Set Disjointness(m) problem in which each player i has a set $T_i \subset [m]$. Define $\mathcal{U}_{\mathcal{I}} = \{e_1, \cdots, e_\alpha\}$ to be the set of elements in the Max 1-Cover instance and for each player i if $j \in T_i$ then add (e_i, S_j) to the stream. In other words, in the constructed Max 1-Cover $(\mathcal{U}_{\mathcal{I}}, \mathcal{F}_{\mathcal{I}} := \{S_1, \cdots, S_m\})$ instance, we have an element e_i corresponding to each player i and there exists a set S_j corresponding to each item $j \in [m]$. Moreover, each set S_j in the Max 1-Cover instance $(\mathcal{U}_{\mathcal{I}}, \mathcal{F}_{\mathcal{I}})$ denotes the set of players in the Set Disjointness(m) instance $\mathcal{I}$ whose input sets contain j; $S_j := \{i \in [\alpha] \mid j \in T_i\}$.

Claim 5.3. *If $\mathcal{I}$ is a* **No** *instance, then the optimal coverage of the* Max 1-Cover *instance $(\mathcal{U}_{\mathcal{I}}, \mathcal{F}_{\mathcal{I}})$ is α.*

Proof. In this case, by the unique intersection promise, there exists an item j that belongs to all T_i (for $i \in [\alpha]$). Hence, by the construction of the Max 1-Cover instance, S_j covers the whole $\mathcal{U}_{\mathcal{I}}$. Thus, the optimal 1-cover has size α. □

Claim 5.4. *If $\mathcal{I}$ is a* **Yes** *instance, then the optimal coverage of the* Max 1-Cover *instance $(\mathcal{U}_{\mathcal{I}}, \mathcal{F}_{\mathcal{I}})$ is* 1.

Proof. Since T_is are disjoint, for each $j \in [m]$, the set S_j has cardinality one. □

Corollary 5.2 together with Claims 5.3 and 5.4 imply the stated lower bound on α-approximating the optimal coverage size of Max k-Cover in edge-arrival streams in Theorem 3.3: *Any single pass (possibly randomized) algorithm on edge-arrival streams that α-approximates the optimal coverage size of* Max k-Cover *requires $\Omega(\frac{m}{\alpha^2})$ space.*

ACKNOWLEDGMENTS

This work was supported by grants from the NSF and the Simons Investigator award.

REFERENCES

[1] Z. Abbassi, V. S. Mirrokni, and M. Thakur. 2013. Diversity maximization under matroid constraints. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 32–40.

[2] Shipra Agrawal, Mohammad Shadravan, and Cliff Stein. 2019. Submodular Secretary Problem with Shortlists. In *10th Innovations in Theoretical Computer Science Conference (ITCS)*. 1:1–1:19.

[3] K. J. Ahn, S. Guha, and A. McGregor. 2012. Analyzing graph structure via linear measurements. In *Proc. 23rd ACM-SIAM Sympos. Discrete Algs.* (SODA). 459–467.

[4] K. J. Ahn, S. Guha, and A. McGregor. 2012. Graph sketches: sparsification, spanners, and subgraphs. In *Proc. 31st ACM Sympos. on Principles of Database Systems* (PODS). 5–14.

[5] N. Alon, Y. Matias, and M. Szegedy. 1999. The space complexity of approximating the frequency moments. *Journal of Computer and system sciences* 58, 1 (1999), 137–147.

[6] S. Assadi. 2017. Tight Space-Approximation Tradeoff for the Multi-Pass Streaming Set Cover Problem. In *Proc. 36th ACM Sympos. on Principles of Database Systems* (PODS). 321–335.

[7] S. Assadi, S. Khanna, and Y. Li. 2016. Tight Bounds for Single-Pass Streaming Complexity of the Set Cover Problem. In *Proc. 48th Annu. ACM Sympos. Theory Comput.* (STOC). 698–711.

[8] S. Assadi, S. Khanna, Y. Li, and G. Yaroslavtsev. 2016. Maximum matchings in dynamic graph streams and the simultaneous communication model. In *Proc. 27th ACM-SIAM Sympos. Discrete Algs.* (SODA). 1345–1364.

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

[10] Z. Bar-Yossef, T. S. Jayram, R. Kumar, and D. Sivakumar. 2004. An information statistics approach to data stream and communication complexity. *J. Comput. Sys. Sci.* 68, 4 (2004), 702–732.

[11] Z. Bar-Yossef, T. S. Jayram, R. Kumar, D. Sivakumar, and L. Trevisan. 2002. Counting distinct elements in a data stream. In *International Workshop on Randomization and Approximation Techniques in Computer Science*. 1–10.

[12] M. Bateni, H. Esfandiari, and V. S. Mirrokni. 2017. Almost Optimal Streaming Algorithms for Coverage Problems. In *Proc. 29th ACM Sympos. Parallel Alg. Arch.* (SPAA). 13–23.

[13] J. Blasiok. 2018. Optimal streaming and tracking distinct elements with high probability. In *Proc. 29th ACM-SIAM Sympos. Discrete Algs.* (SODA). 2432–2448.

[14] V. Braverman, S. R. Chestnut, N. Ivkin, J. Nelson, Z. Wang, and D. P. Woodruff. 2017. BPTree: An ℓ_2 Heavy Hitters Algorithm Using Constant Memory. In *Proc. 36th ACM Sympos. on Principles of Database Systems* (PODS). 361–376.

[15] V. Braverman, S. R. Chestnut, N. Ivkin, and D. P. Woodruff. 2016. Beating CountSketch for heavy hitters in insertion streams. In *Proc. 48th Annu. ACM Sympos. Theory Comput.* (STOC). 740–753.
[16] A. Chakrabarti, S. Khot, and X. Sun. 2003. Near-optimal lower bounds on the multi-party communication complexity of set disjointness. In *18th Annual IEEE Conference on Computational Complexity*. 107–117.
[17] A. Chakrabarti and A. Wirth. 2016. Incidence Geometries and the Pass Complexity of Semi-Streaming Set Cover. In *Proc. 27th ACM-SIAM Sympos. Discrete Algs.* (SODA). 1365–1373.
[18] M. Charikar, K. Chen, and M. Farach-Colton. 2002. Finding frequent items in data streams. In *Proc. 29th Int. Colloq. Automata Lang. Prog.* (ICALP). 693–703.
[19] F. Chierichetti, R. Kumar, and A. Tomkins. 2010. Max-cover in map-reduce. In *Proc. 19th Int. Conf. World Wide Web* (WWW). 231–240.
[20] R. Chitnis, G. Cormode, H. Esfandiari, M. Hajiaghayi, A. McGregor, M. Monemizadeh, and S. Vorotnikova. 2016. Kernelization via sampling with applications to finding matchings and related problems in dynamic graph streams. In *Proc. 27th ACM-SIAM Sympos. Discrete Algs.* (SODA). 1326–1344.
[21] E. D. Demaine, P. Indyk, S. Mahabadi, and A. Vakilian. 2014. On Streaming and Communication Complexity of the Set Cover Problem. In *Proc. 28th Int. Symp. Dist. Comp.* (DISC), Vol. 8784. 484–498.
[22] Y. Emek and A. Rosén. 2014. Semi-Streaming Set Cover. In *Proc. 41st Int. Colloq. Automata Lang. Prog.* (ICALP) *(Lect. Notes in Comp. Sci.)*, Vol. 8572. 453–464. https://doi.org/10.1007/978-3-662-43948-7_38
[23] U. Feige. 1998. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)* 45, 4 (1998), 634–652.
[24] A. Gronemeier. 2009. Asymptotically Optimal Lower Bounds on the NIH-Multi-Party Information Complexity of the AND-Function and Disjointness. In *26th International Symposium on Theoretical Aspects of Computer Science, STACS 2009, February 26-28, 2009, Freiburg, Germany, Proceedings*. 505–516.
[25] S. Guha, A. McGregor, and D. Tench. 2015. Vertex and hyperedge connectivity in dynamic graph streams. In *Proc. 34th ACM Sympos. on Principles of Database Systems* (PODS). 241–247.
[26] S. Har-Peled, P. Indyk, S. Mahabadi, and A. Vakilian. 2016. Towards Tight Bounds for the Streaming Set Cover Problem. In *Proc. 35th ACM Sympos. on Principles of Database Systems* (PODS). 371–383.
[27] P. Indyk, S. Mahabadi, R. Rubinfeld, J. Ullman, A. Vakilian, and A. Yodpinyanee. 2017. Fractional Set Cover in the Streaming Model. *Approximation, Randomization, and Combinatorial Optimization* (APPROX/RANDOM) (2017), 198–217.
[28] P. Indyk, S. Mahabadi, R. Rubinfeld, A. Vakilian, and A. Yodpinyanee. 2018. Set Cover in Sub-linear Time. In *Proc. 29th ACM-SIAM Sympos. Discrete Algs.* (SODA). 2467–2486.
[29] P. Indyk and D. P. Woodruff. 2005. Optimal approximations of the frequency moments of data streams. In *Proc. 37th Annu. ACM Sympos. Theory Comput.* (STOC). 202–208.
[30] D. M. Kane, J. Nelson, and D. P. Woodruff. 2008. Revisiting norm estimation in data streams. *arXiv preprint arXiv:0811.3648* (2008).
[31] D. M. Kane, J. Nelson, and D. P. Woodruff. 2010. An optimal algorithm for the distinct elements problem. In *Proc. 29th ACM Sympos. on Principles of Database Systems* (PODS). 41–52.
[32] M. Kapralov, Y. T. Lee, C. Musco, C. Musco, and A. Sidford. 2017. Single pass spectral sparsification in dynamic streams. *SIAM J. Comput.* 46, 1 (2017), 456–477.
[33] S. Lattanzi, B. Moseley, S. Suri, and S. Vassilvitskii. 2011. Filtering: a method for solving graph problems in mapreduce. In *Proc. 23rd ACM Sympos. Parallel Alg. Arch.* (SPAA). ACM, 85–94.
[34] A. McGregor and H. T. Vu. 2017. Better Streaming Algorithms for the Maximum Coverage Problem. In *20th International Conference on Database Theory (ICDT)*. 22:1–22:18.
[35] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. 1978. An analysis of approximations for maximizing submodular set functions–I. *Mathematical Programming* 14, 1 (1978), 265–294.
[36] Ashkan Norouzi-Fard, Jakub Tarnawski, Slobodan Mitrovic, Amir Zandieh, Aidasadat Mousavifar, and Ola Svensson. 2018. Beyond 1/2-Approximation for Submodular Maximization on Massive Data Streams. In *International Conference on Machine Learning (ICML)*. 3826–3835.
[37] B. Saha and L. Getoor. 2009. On Maximum Coverage in the Streaming Model & Application to Multi-topic Blog-Watch. In *Proc. SIAM Int. Conf. Data Mining* (SDM). 697–708.
[38] J. P. Schmidt, A. Siegel, and A. Srinivasan. 1995. Chernoff–Hoeffding bounds for applications with limited independence. *SIAM Journal on Discrete Mathematics* 8, 2 (1995), 223–250.
[39] M. Thorup and Y. Zhang. 2012. Tabulation-based 5-independent hashing with applications to linear probing and second moment estimation. *SIAM J. Comput.* 41, 2 (2012), 293–331.
[40] S. Vadhan. 2012. Pseudorandomness. *Foundations and Trends® in Theoretical Computer Science* 7, 1–3 (2012), 1–336.

A CHERNOFF BOUND FOR APPLICATIONS WITH LIMITED INDEPENDENCE

In this section, we mention some of the results in [38] on applications of Chernoff bound with limited independence that are used in our analysis.

Definition A.1 (Family of d-wise Independent Hash Functions). A family of functions $\mathcal{H} = \{h : [m] \to [n]\}$ is d-wise independent, if for any set of d distinct values $x_1, \cdots, x_d$, the random variables $h(x_1), \cdots, h(x_d)$ are independent and uniformly distributed in $[n]$ when h is picked uniformly at random from $\mathcal{H}$.

Next, we exploit the results that show for small values of d, we can store a family of d-wise hash function in small space and in the same time it suffices for our application of Chernoff bound.

Lemma A.2 (Corollary 3.34 in [40]). *For every values of m, n, and d, there is a family of d-wise independent hash functions $\mathcal{H} = \{h : [m] \to [n]\}$ such that a selecting a random function from $\mathcal{H}$ only requires $d \cdot \log(mn)$.*

Lemma A.3 (Theorem 5 in [38]). *Let $X_1, \cdots X_n$ be binary d-wise independent random variables and let $X := X_1 + \cdots + X_n$. Then,*

$$\Pr(|X - \mathbf{E}[X]| \geq \delta \mathbf{E}[X]) \leq \begin{cases} e^{-\frac{\mathbf{E}[X]\delta^2}{3}} & : \text{ if } \delta < 1 \text{ and } d = \Omega(\delta^2 \mathbf{E}[X]); \\ e^{-\frac{\mathbf{E}[X]\delta}{3}} & : \text{ if } \delta \geq 1 \text{ and } d = \Omega(\delta \mathbf{E}[X]). \end{cases}$$

Lemma A.4 (Theorem 6 in [38]). *Let $X_1, \cdots, X_n$ and $Y_1, \cdots, Y_n$ be Bernoulli trials such that for each i, $\mathbf{E}[X_i] = \mathbf{E}[Y_i] = p_i$. Let assume that Y_is are independent, but X_is are only d-wise independent. Further, let $p(r)$ and $p_d(r)$ respectively denote $\Pr(\sum_{i=1}^n Y_i = r)$ and $\Pr(\sum_{i=1}^n X_i = r)$ and let $\mu = \sum_{i=1}^n p_i$ be the expected number of success in the trials.*

If $d \geq e\mu + \ln(1/p(0)) + r + D$, then $|p_d(r) - p(r)| \leq e^{-D} p(r)$.

A.1 An Application: Set Sampling with $\Theta(\log(mn))$-wise Independence

Consider a set system $(\mathcal{U}, \mathcal{F})$ and let $h : \mathcal{F} \to [(cm \log m)/\gamma]$ be a function selected uniformly at random from a family of $\Theta(\log(mn))$-wise independent hash functions where c is a sufficiently large constant. Then, we think of our randomly selected sets $\mathcal{F}^{\text{rnd}}$ to be the collection of sets in $\mathcal{F}$ that are mapped to one by h; $\mathcal{F}^{\text{rnd}} := \{S \in \mathcal{F} \mid h(S) = 1\}$.

Lemma A.5. *Assuming $\gamma \geq 6c \log^2 m$, with probability at least $1 - m^{-1}$, $|\mathcal{F}^{\text{rnd}}| \leq \gamma$.*

Proof. Let X_i be a random variable which is one if $S_i \in \mathcal{F}^{\text{rnd}}$ and zero otherwise. We define $X := X_1 + \cdots + X_m$. Note that X_i are $\Theta(\log(mn))$-wise independent and $\mathbf{E}[X] = \gamma/(c \log m)$. Then, by an application of Chernoff bound with limited independence (Lemma A.3),

$$\Pr(X > \gamma) \leq \Pr(X > \underbrace{(1 + \sqrt{\frac{6c}{\gamma}} \log m)}_{\leq 2\gamma/c} \mathbf{E}[X]) < m^{-1}.$$

Hence, with high probability, $\mathcal{F}^{\text{rnd}}$ has size at most γ. □

Next, we show that $\mathcal{F}^{\text{rnd}}$ covers the set of elements $\mathcal{U}^{\text{cmn}}_{\gamma}$ (see Definition 2.1).

Lemma A.6. *With probability at least $1 - n^{-1}$, $\mathcal{F}^{\text{rnd}}$ covers $\mathcal{U}^{\text{cmn}}_{\gamma}$.*

Proof. Let $e \in \mathcal{U}^{\text{cmn}}_{\gamma}$ and let $S_1, \cdots, S_q$ be the sets in $\mathcal{F}$ that cover e: for each $i \leq q$, $e \in S_i$. We define X_i to be a random variable which is one if $S_i \in \mathcal{F}^{\text{rnd}}$ and is zero otherwise. We also define $X := X_1 + \cdots + X_q$ to denote the number of sets in $\mathcal{F}^{\text{rnd}}$ that cover e. Note that X_i are $\Theta(\log(mn))$-wise independent and $\mathbf{E}[X] = (\gamma/(cm \log m)) \cdot q \geq \log n \log(mn)$. Then, applying Chernoff bound on random variables with limited independence (Lemma A.3),

$$\Pr(X < 1) < \Pr(X < (1 - \underbrace{\sqrt{(6 \log n)/\mathbf{E}[X]}}_{\leq 1/2})\mathbf{E}[X]) < n^{-2}.$$

Hence, by union bound over all elements in $\mathcal{U}^{\text{cmn}}_{\gamma}$, with high probability, $\mathcal{F}^{\text{rnd}}$ covers $\mathcal{U}^{\text{cmn}}_{\gamma}$. □

Lemma A.7 (Set Sampling with limited independence). *$\Theta(\log(mn))$ random bits suffice to implement set sampling method.*

B GENERALIZATION OF SECTION 4.2

In this section we generalize the approach of Section 4.2 which relates the results on heavy hitters and contributing classes to approximating the optimal coverage size. In Section 4.2, to simplify the presentation, we had the assumption that $\mathcal{U}^{\text{cmn}}_{\text{w}}$ is empty. This is in particular important since we can then assume that the total size of k-covers are roughly the same as their coverage size (up to polylogarithmic factors).

Here, we take care of the case in which $\mathcal{U}^{\text{cmn}}_{\text{w}}$ is non-empty and complete the description of our (α, δ, η)-oracle of Max k-Cover for the case $|C(\text{OPT}_{\text{large}})| \geq |C(\text{OPT})|/2$. The high-level idea is to sample enough number of elements so that the algorithm using heavy hitters and contributing classes still works but in the same time no w-common element is among the sampled elements with at least a constant probability.

Step 1. Sampling Elements. We sample a subset $\mathcal{L} \subset \mathcal{U}$ in which each element e is in $\mathcal{L}$ with probability $\rho = (\text{ts} \cdot \alpha\eta)/|\mathcal{U}| = \widetilde{O}(\alpha/|\mathcal{U}|)$ where $\text{t} = \widetilde{O}(1)$ (see Table 2 for the exact values). We implement the process of sampling $\mathcal{L}$ via a hash function from a family of $\Theta(\log(mn))$-wise independent functions $\mathcal{H} = \{h : \mathcal{U} \to [\frac{n}{\text{t} \cdot \text{s}\alpha \cdot \eta}]\}$ such that $\mathcal{L} = \{e \in \mathcal{U} \mid h(e) = 1\}$.

Claim B.1. *With high probability, $\frac{\rho|\mathcal{U}|}{2} \leq |\mathcal{L}| \leq \frac{3\rho|\mathcal{U}|}{2}$.*

For each collection of set $\mathcal{D}$, we define $\mathcal{D}'$ to be the intersection of $\mathcal{D}$ with $\mathcal{L}$; $\mathcal{D}' := \{S \cap \mathcal{L} \mid S \in \mathcal{D}\}$.

Claim B.2. *If $|C(\mathcal{D})| \geq |\mathcal{U}|/(54\text{f}\eta\alpha)$, then with probability at least $1 - m^{-2}$, $|C(\mathcal{D}')| \geq \rho|C(\mathcal{D})|/2$. Moreover, if $|C(\mathcal{D})| < |\mathcal{U}|/(54\text{f}\eta\alpha)$, then with probability at least $1 - m^{-2}$, $|C(\mathcal{D}')| < \text{ts}/(36\text{f})$.*

Proof. Since $\text{ts} \geq 27 \cdot 24\text{f} \log m$ (as in Table 2), it follows from two applications of Chernoff bound on random variables with limited independence (Lemma A.3). □

Similarly to Lemma 4.14, we have the following guarantee for LargeSetSimple using the sampled set of elements $\mathcal{L}$.

Lemma B.3. *If $|C(OPT)| \geq |\mathcal{U}|/\eta$ and $\mathcal{L} \cap \mathcal{U}^{\text{cmn}} = \emptyset$, then with probability at least $1 - (2 \log n \, \text{polylog}\, m)^{-1}$, the output of LargeSetSimple with parameters $(\mathcal{L}, \text{w}, \text{r}_1 = \text{s}_{\mathcal{L}}\alpha, \text{r}_2 = \frac{cm \log m}{\text{w}}, \text{thr}_1 = \frac{|\mathcal{L}|}{18\eta\text{s}\alpha}, \text{thr}_2 = \frac{|\mathcal{L}|}{6\eta\alpha})$ is a superset whose coverage is at least $|\mathcal{U}|/(54\text{f}\eta\alpha)$.*

Proof. By Claim B.1, $(\rho|\mathcal{U}|)/2 \leq |\mathcal{L}| \leq (3\rho|\mathcal{U}|)/2$. Moreover, by Claim B.2 and since $|C(\text{OPT})| \geq |\mathcal{U}|/\eta$, with probability at least $1 - m^{-2}$, $|C(\text{OPT}_{\text{large}}) \cap \mathcal{L}| \geq \rho|C(\text{OPT})|/4 \geq |\mathcal{L}|/(6\eta)$. We define $\eta_{\mathcal{L}} := 6\eta$ to denote the coverage of $\text{OPT}_{\text{large}}$ on the sampled elements $\mathcal{L}$.

Now, consider the collection $\text{OPT}_{\text{large}} := \{O_1, \cdots, O_q\}$. Since for each $i \leq q$, the contribution of O_i to the coverage of OPT is at least $1/(\text{s}\alpha)$ fraction (i.e. $|O_i \setminus \bigcup_{j<i} O_j| \geq |C(\text{OPT})|/(\text{s}\alpha) \geq |\mathcal{U}|/(\eta\text{s}\alpha)$), by Claim B.2, with probability at least $1 - m^{-1}$, for all $i \leq q$,

$$|(O_i \setminus \bigcup_{j<i} O_j) \cap \mathcal{L}| \geq \rho|C(\text{OPT})|/(2\text{s}\alpha).$$

This implies that $\{O_i \cap \mathcal{L} \mid O_i \in \mathrm{OPT}_{\mathrm{large}}\}$ is a collection of sets whose contribution to $C(\mathrm{OPT}) \cap \mathcal{L}$ w.h.p. is at least $(\frac{\rho|C(\mathrm{OPT})|}{2\mathsf{s}\alpha})/(\frac{3\rho|C(\mathrm{OPT})|}{2}) = 1/(3\mathsf{s}\alpha)$. We define $\mathsf{s}_{\mathcal{L}} := 3\mathsf{s}$ which denotes the contribution of sets in $\mathrm{OPT}_{\mathrm{large}}$ compared to the coverage of OPT on the sampled elements $\mathcal{L}$.

By an application of Lemma 4.14 with parameters ($\mathcal{V} := \mathcal{L}, \mathrm{thr}_1 := \frac{|\mathcal{L}|}{\eta_{\mathcal{L}}\mathsf{s}_{\mathcal{L}}\alpha}, \mathrm{thr}_2 := \frac{|\mathcal{L}|}{\eta_{\mathcal{L}}\alpha}$), with probability at least $1 - 1/(3\log n \log^c m)$, the algorithm returns a superset $\mathcal{D}_i'$ whose coverage on the sampled set $\mathcal{L}$ is at least

$$\frac{|\mathcal{L}|}{3\mathsf{f}\eta_{\mathcal{L}}\alpha} \geq \frac{\rho|\mathcal{U}|}{36\mathsf{f}\alpha\eta} = \frac{\mathsf{ts}}{36\mathsf{f}}.$$

Then, by Claim B.2, with probability at least $1 - m^{-2}$, $\mathcal{D}_i$ has coverage at least $|\mathcal{U}|/(54\mathsf{f}\eta\alpha)$. □

Step 2. Handling Common Elements. Next, we turn our attention to the case $\mathcal{L} \cap \mathcal{U}_{\mathsf{w}}^{\mathsf{cmn}} \neq \emptyset$. Although common elements may be covered $\widetilde{\Omega}(1)$ times within a single superset, the contribution of common elements to all supersets is roughly the same.

Claim B.4. *Let $\mathcal{L}_{\mathsf{w}}^{\mathsf{cmn}} := \mathcal{L} \cap \mathcal{U}_{\mathsf{w}}^{\mathsf{cmn}}$ be the set of* w*-common elements that are sampled in $\mathcal{L}$. Then, with high probability, for each superset $\mathcal{D}$, the total number of times that elements of $\mathcal{L}_{\mathsf{w}}^{\mathsf{cmn}}$ appear in $\mathcal{D}$ (counting duplicates) belongs to $[P, 2P]$ where P is a fixed number larger than* $\log n$.

Proof. Let $e \in \mathcal{U}_{\mathsf{w}}^{\mathsf{cmn}}$ be a w-common element and let $S_1, \cdots, S_q$ be the collection of sets that contain e. For a superset $\mathcal{D}_j$, define $X_{i,j}$ to be a binary random variable that denotes whether $S_i \in \mathcal{D}_j$. Moreover, let $Y_{j,e} := X_{1,j} + \cdots + X_{q,j}$. Then, $\mathbf{E}[Y_{j,e}] = \frac{\mathsf{w}q}{cm\log m}$. By an application of Chernoff bound with limited independence (Lemma A.3) and since $\mathsf{w}q \geq cm\log m \log n \log(mn)$ (see Definition 2.1),

$$\Pr(|Y_{j,e} - \mathbf{E}[Y_{j,e}]| \geq \underbrace{\sqrt{\frac{6c\log(mn)m\log m}{\mathsf{w}q}}}_{\leq 1/3}\mathbf{E}[Y_{j,e}]) \leq (mn)^{-2}. \tag{7}$$

Note that for any w-common element e and any pair of supersets $\mathcal{D}_j, \mathcal{D}_i$, $\mathbf{E}[Y_{j,e}] = \mathbf{E}[Y_{i,e}]$. In particular, we define $Y_e := \mathbf{E}[Y_{j,e}]$ whose value is independet of the supersets. Next, we define $Y_{\mathsf{cmn}} := \sum_{e\in\mathcal{U}_{\mathsf{w}}^{\mathsf{cmn}}} Y_e$ to denote the expected contribution of w-common elements to any superset. Hence, for each superset $\mathcal{D}_j$, with probability at least $1 - 1/(nm^2)$, the total number of times that w-common elements are covered by $\mathcal{D}_j$ belongs to the range $[2Y_{\mathsf{cmn}}/3, 4Y_{\mathsf{cmn}}/3]$. Hence, with probability at least $1 - (mn)^{-1}$, the contribution of sampled w-common elements to each superset belongs to $[2Y_{\mathsf{cmn}}/3, 4Y_{\mathsf{cmn}}/3]$ where $Y_{\mathsf{cmn}} \geq \log n \log(mn)|\mathcal{L}_{\mathsf{w}}^{\mathsf{cmn}}| \geq \log n$. □

Next, we show that if a w-common element is picked in $\mathcal{L}$ the algorithm still does not return a superset with small coverage (though it may missed all large supersets). To this end, we modify LargeSetSimple and design a new subroutine LargeSetComplete as in Figure 6. The high-level idea is to guarantee that if the main contribution of a superset is just from the duplicate counts of w-common elements ($\propto P$), it will not be returned. To achieve this, unlike LargeSetSimple we do not allow F_2-Contributing to consider contributing classes of any size (up to $|Q|$). Instead, we set parameters r_1 and r_2 which denotes how large the size of a contributing class that we are looking for is. To handle the case in which the size of a contributing class is large (i.e. larger than r_2), we sample supersets proportional to $1/\mathsf{r}_2$ and compute their coverage by existing L_0-estimation algorithms.

Lemma B.5. *Even if $\mathcal{L} \cap \mathcal{U}_{\mathsf{w}}^{\mathsf{cmn}} \neq \emptyset$, with probability at least $1 - m^{-1}$, none of the solutions returned by* LargeSetComplete *with parameters ($\mathcal{L}, \mathsf{w}, \mathsf{r}_1 = \mathsf{s}_{\mathcal{L}}\alpha, \mathsf{r}_2 = \widetilde{\Theta}(\frac{cm\log m}{\mathsf{w}}), \mathrm{thr}_1 = \frac{|\mathcal{L}|}{18\eta\mathsf{s}\alpha}, \mathrm{thr}_2 = \frac{|\mathcal{L}|}{6\eta\alpha}$) is a superset with coverage less than $|\mathcal{U}|/(54\mathsf{f}\cdot\eta\cdot\alpha)$.*

Proof. Here, we need to revisit Case 1 and Case 2 of Section 4.2 and redo the calculations with respect to the sampled elements $\mathcal{L}$.

Case 1. Suppose that F_2-Contributing$(\widetilde{\Omega}(\frac{\alpha^2}{m}), \mathsf{s}_{\mathcal{L}}\cdot\alpha)$ returns a solution whose coverage is less than $|\mathcal{U}|(54\mathsf{f}\cdot\eta\cdot\alpha)$.

Let $\vec{r}$ be a vector of size at most $(cm\log m)/\mathsf{w}$ whose ith entry denotes the total size of the intersection of sets in $\mathcal{D}_i$ and $\mathcal{L}_{\mathsf{w}}^{\mathsf{rare}} := \mathcal{L} \setminus \mathcal{U}_{\mathsf{w}}^{\mathsf{cmn}}$; $\vec{r}[i] := \sum_{S\in\mathcal{D}_i} |S \cap \mathcal{L}_{\mathsf{w}}^{\mathsf{rare}}|$. By Claim B.2, if $|C(\mathcal{D}_j)| < |\mathcal{U}|/(54\mathsf{f}\eta\alpha)$, with probability at least $1 - m^{-2}$, $|C(\mathcal{D}_i) \cap \mathcal{L}_{\mathsf{w}}^{\mathsf{rare}}| < |C(\mathcal{D}_i) \cap \mathcal{L}| < \mathsf{ts}/(36\mathsf{f})$. Hence, together with Claim 4.10, with probability at least $1 - 2m^{-2}$, $\vec{r}[i] < \mathsf{ts}/36$.

Similarly, let $\vec{v}$ be a vector of size $(cm\log m)/\mathsf{w}$ whose ith entry denotes the total size of the intersection of sets in $\mathcal{D}_i$ with the sampled elements $\mathcal{L}$; $\vec{v}[i] := \sum_{S\in\mathcal{D}_i} |S \cap \mathcal{L}|$. Note that Since $P \geq 1$, for each superset $\mathcal{D}_i$ with coverage less than $|\mathcal{U}|/(54\mathsf{f}\cdot\eta\cdot\alpha)$, $\vec{v}[i] \leq 2P + \vec{r}[i] \leq (\mathsf{ts}/36)P$. On the other hand, by Claim B.4, with probability at least $1 - m^{-1}$, the value of $\vec{v}[j]$ for all j in the sampled substream is at least P.

Moreover, since the size of contributing classes in this case is at most $\mathsf{s}_{\mathcal{L}}\cdot\alpha = 3\mathsf{s}\alpha$ (more precisely, we can always find at most $3\mathsf{s}\alpha$ sets that are $\widetilde{\Omega}(\frac{\alpha^2}{m})$-contributing), by Claim 2.8, with probability at least $1 - \frac{\log m}{m}$, all sampled substreams considered by F_2-Contributing$(\phi_1, \mathsf{s}_{\mathcal{L}}\cdot\alpha)$ have size at least $cm\log m/(\mathsf{w}\cdot\mathsf{s}\cdot\alpha)$. Hence, by Claim B.4, with probability at least $1 - \frac{2\log m}{m}$,

$$F_2(\vec{v}_{\mathsf{smp}}) \geq (\frac{cm\log m}{\mathsf{w}\cdot\mathsf{s}\cdot\alpha})P^2,$$

LargeSetComplete$(\mathcal{V}, \mathsf{w}, \mathsf{r}_1, \mathsf{r}_2, \mathrm{thr}_1, \mathrm{thr}_2)$:
▷ **Input:** w is an upper bound on the size of a superset
▷ **Parameters:** $\phi_1 = \widetilde{\Omega}(\alpha^2/m)$ and $\phi_2 = \widetilde{\Omega}(1)$
▷ For Case 1
let $\mathsf{Cntr}_{\mathsf{small}}$ be an instance of F_2-Contributing(ϕ_1, r_1)
▷ For Case 2
let $\mathsf{Cntr}_{\mathsf{large}}$ be an instance of F_2-Contributing(ϕ_2, r_2)
pick $h : \mathcal{F} \to [(cm \log m)/\mathsf{w}]$ from $\Theta(\log(mn))$-wise independent hash functions
for each (S, e) in the data stream **do**
  if $e \in \mathcal{V}$ **then feed** $h(S)$ to both $\mathsf{Cntr}_{\mathsf{small}}$ and $\mathsf{Cntr}_{\mathsf{large}}$
▷ output(Cntr) returns $(1 \pm 1/2)$-estimate of frequencies
if there exists $i^* \in \mathrm{output}(\mathsf{Cntr}_{\mathsf{small}})$ such that $\tilde{v}_{i^*} \geq \frac{1}{2} \cdot \mathrm{thr}_1$
  return $2\tilde{v}_{i^*}/(3\mathsf{f})$
▷ add **return** $\{S \mid h(S) = i^*\}$ to get a k-cover
if there exists $i^* \in \mathrm{output}(\mathsf{Cntr}_{\mathsf{large}})$ such that $\tilde{v}_{i^*} \geq \frac{1}{2} \cdot \mathrm{thr}_2$
  return $2\tilde{v}_{i^*}/(3\mathsf{f})$
▷ add **return** $\{S \mid h(S) = i^*\}$ to get a k-cover

▷ Case 2: if size of the contributing class is large; $\widetilde{\Omega}(|\mathcal{Q}|)$
let $\mathcal{M} \subset \mathcal{Q}$ be a collection of size $12|\mathcal{Q}| \log m/\mathsf{r}_2$ picked uniformly at random
for each $i \in \mathcal{M}$ **do**
  ▷ DE estimates the coverage of the supersets in $\mathcal{L}$
  let DE_i be a $(1/2)$-approximation algorithm of L_0-estimation initialized to zero
pick $h : \mathcal{F} \to [(cm \log m)/\mathsf{w}]$ from $\Theta(\log(mn))$-wise independent hash functions
for each (S, e) in the data stream **do**
  if $e \in \mathcal{V}$ and $h(S) \in \mathcal{M}$ **then** feed $h(S)$ to $\mathsf{DE}_{h(S)}$
if there exists $i^* \in \mathcal{M}$ such that $\mathrm{VAL}(\mathsf{DE}_{i^*}) \geq \frac{1}{2} \cdot \mathrm{thr}_2$
  return $2\mathrm{VAL}(\mathsf{DE}_{i^*})/3$
  ▷ add **return** $\{S \mid h(S) = i^*\}$ to get the k-cover
return infeasible

Figure 6: A (α, δ, η)-oracle of Max k-Cover that handles the case in which the majority of the coverage in an optimal solution is due to the sets whose coverage contributions are at least $1/(\mathsf{s}\alpha)$ fraction of the optimal coverage size.

where $\vec{v}_{\mathsf{smp}}$ is a vector corresponding to a sampled substream considered in F_2-Contributing$(\phi_1, \mathsf{s}_{\mathcal{L}} \cdot \alpha)$. Since $\mathsf{s}^4\mathsf{t}^2 \leq \frac{81}{2\eta \log(\mathsf{s}\alpha)}$ (see Table 2) and by the value of ϕ_1 as in Eq. (6), the following holds:

$$(\frac{\mathsf{ts}}{36})^2 P^2 < \phi_1 \cdot \frac{cm \log m}{\mathsf{w} \cdot \mathsf{s} \cdot \alpha} P^2,$$

which implies that with probability at least $1 - 3\log m/m$, an entry corresponding to a superset with coverage less than $|\mathcal{U}|/(54\mathsf{f} \cdot \eta \cdot \alpha)$ cannot be a ϕ_1-HeavyHitter in any of the sampled substreams considered in F_2-Contributing$(\phi_1, \mathsf{s}_{\mathcal{L}} \cdot \alpha)$.

Case 2. The high-level idea in this case is similar to the previous case. In Case 1, we heavily used the fact that there exists a class containing at most $\mathsf{s}_{\mathcal{L}} \cdot \alpha$ coordinates that is $\widetilde{\Omega}(\frac{\alpha^2}{m})$-contributing.

This observation is crucial because then we can show that all sampled substreams considered in F_2-Contributing with parameters $(\phi_1, \mathsf{s}_{\mathcal{L}}\alpha)$ have size at least $\widetilde{\Omega}(m/(\mathsf{w} \cdot \mathsf{s}_{\mathcal{L}} \cdot \alpha))$ which rules out the possibility that a coordinate corresponding to a small superset is a $\widetilde{\Omega}(\frac{\alpha^2}{m})$-HeavyHitter for sufficiently small values of s (recall that $\mathsf{s}_{\mathcal{L}} = 3\mathsf{s}$).

However, in this case, a contributing class may have size $\widetilde{\Omega}(m/\mathsf{w})$ which results in a sampled substream with only $\widetilde{O}(1)$ coordinates in the run of F_2-Contributing! To address the issue, we handle the case in which a contributing class has more than $\mathsf{r}_2 : \frac{cm \log m}{\mathsf{w}} \cdot \gamma$ coordinates separately:

1. $\widetilde{\Omega}(1)$-contributing class has size less than r_2. Since $P \geq 1$ and by Claim B.2 and 4.10, for each superset $\mathcal{D}_j$ with coverage less than $|\mathcal{U}|/(54\mathsf{f} \cdot \eta \cdot \alpha)$, with probability at least $1 - 2m^{-2}$, $\vec{v}[j] \leq (\mathsf{ts}/36)P$. On the other hand, by Claim B.4, with probability at least $1 - m^{-1}$, the value of $\vec{v}[j]$ for all j in the sampled substream is at least P. Moreover, by Claim 2.8, with probability at least $1 - \frac{\log m}{m}$, all sampled substreams considered in F_2-Contributing$(\phi_2 = \frac{1}{2\log(\alpha)}, \mathsf{r}_2)$ invoked by LargeSetComplete (which is to handle Case 2) have at least $(\frac{3cm \log m}{\mathsf{w} \cdot \mathsf{r}_2}) = \frac{3}{\gamma}$ coordinates. Hence, $F_2(\vec{v}_{\mathsf{smp}}) \geq \frac{3}{\gamma} \cdot P^2$. By setting

$$\gamma < \frac{3\phi_2}{(\mathsf{ts}/36)^2} = \frac{1944}{\log(\alpha)\mathsf{t}^2\mathsf{s}^2}, \tag{8}$$

$\vec{v}[j]^2 \leq (\mathsf{ts}/36)^2 P^2 < (\frac{1}{2\log\alpha} F_2(\vec{v}_{\mathsf{smp}}))$, which implies that an entry corresponding to a superset with coverage less than $\frac{|\mathcal{U}|}{27\mathsf{f} \cdot \eta \cdot \alpha}$ cannot be a ϕ_2-HeavyHitter in any of the sampled substream considered in F_2-Contributing(ϕ_2, r_2).

2. $\widetilde{\Omega}(1)$-contributing class has size at least r_2. Here, we need to consider an extra case compared to Lemma 4.14 and B.3 because we do not allow r_2 to try all values up to $(\frac{cm \log m}{\mathsf{w}})$. To address the case in which the number of coordinates in a contributing class is larger than r_2, we sample $\ell = (12 \log m)|\mathcal{Q}|/\mathsf{r}_2$ supersets $\mathcal{M}$ uniformly at random from $\mathcal{Q}$; with high probability, $\mathcal{M}$ contains a superset from the contributing class. Then, we compute the coverage of sampled supersets via an existing algorithm for L_0-estimation. By Claim 4.13, the coverage of supersets corresponding to the ϕ_2-contributing class whose size is larger than r_2 on the sampled set $\mathcal{L}$ is at least $|\mathcal{L}|/(\eta_{\mathcal{L}} \cdot \alpha) \geq \mathsf{ts}/(12\mathsf{f})$. Hence, the algorithm finds a superset with coverage at least $\mathsf{ts}/(36\mathsf{f})$ on $\mathcal{L}$ which by Claim B.2, it implies that the returned superset has coverage at least $|\mathcal{U}|/(54\mathsf{f} \cdot \eta \cdot \mathsf{s} \cdot \alpha)$. □

Theorem B.6. *If $|C(OPT)| \geq |\mathcal{U}|/\eta$, then with probability at least $1-(\log n \operatorname{polylog} m)^{-1}$, LargeSet$(k,\alpha)$ returns at least $\frac{|\mathcal{U}|}{54\mathsf{f}\cdot\eta\cdot\alpha}$. Moreover, if LargeSet$(k,\alpha)$ returns a value other than* **infeasible**, *then with probability at least $1-4m^{-1}$, $|C(OPT)| \geq \frac{|\mathcal{U}|}{54\mathsf{f}\cdot\eta\cdot\alpha}$.*

Proof. By Lemma B.5, with probability at least $1-3m^{-1}$, LargeSet never returns a superset with coverage less than $|\mathcal{U}|/(27\mathsf{f}\eta\alpha)$; either it returns a large enough estimate or returns **infeasible**. Here, we show that if $|C(\text{OPT})| > |\mathcal{U}|/\eta$, then the algorithm will return an estimate at least $|\mathcal{U}|/(54\mathsf{f}\cdot\eta\cdot\alpha)$ with probability at least $1-1/(2\log n\log^c m)-n^{-1}$. To this end, we show that with high probability, in one of the $O(\log n)$ parallel runs of LargeSet, the sampled sets of elements $\mathcal{L}$ does not contain any common element. Then, by Lemma B.3, the algorithm with probability at least $1-1/(2\log n\log^c m)$ returns $|\mathcal{U}|/(54\mathsf{f}\cdot\eta\cdot\alpha)$ in the iteration in which the sampled set of elements that does not contain any common elements.

Now, we show that with probability at least $1-n^{-1}$, in one of $O(\log n)$ parallel runs of LargeSet, the sampled set does not contain any common elements. Let $q = |\mathcal{U}^{\text{cmn}}_{\mathsf{w}}|$ and define $Y_1,\cdots,Y_q$ to be independent Bernoulli trials with probability of success equal to ρ. Recall that, we have the assumption that $|\mathcal{U}^{\text{cmn}}_{k}| \leq \frac{\sigma|\mathcal{U}|}{\alpha}$ and since $\mathsf{w}\leq k$, $|\mathcal{U}^{\text{cmn}}_{\mathsf{w}}| \leq |\mathcal{U}^{\text{cmn}}_{k}| \leq \frac{\sigma|\mathcal{U}|}{\alpha}$,

$$\mu = \mathbf{E}[\sum_{i=1}^{q} Y_i] \leq \frac{\sigma|\mathcal{U}|}{\alpha}\cdot\rho = \mathsf{t}\cdot\mathsf{s}\cdot\eta\cdot\sigma$$

$$\Pr(\sum_{i=1}^{q} Y_i = 0) = (1-\rho)^q \geq e^{-2\rho q} \geq e^{-2\mathsf{t}\cdot\mathsf{s}\cdot\eta\cdot\sigma}$$

Next, let's assume that $\mathcal{U}^{\text{cmn}}_{\mathsf{w}} = \{e_1,\cdots,e_q\}$. Further, define $X_1,\cdots,X_q$ to be random variables such that $X_i = 1$ if $e_i \in \mathcal{L}$. Hence, $X_1,\cdots,X_q$ are $\Theta(\log(mn))$-wise independent Bernoulli trials with success probability $\mathbf{E}[X_i] = \rho$. Next, by Lemma A.4 with the following parameters:

$$r = 0, \ln(1/p(0)) \leq 2\mathsf{t}\cdot\mathsf{s}\cdot\eta\cdot\sigma, \mu = \mathsf{t}\cdot\mathsf{s}\cdot\eta\cdot\sigma, D = 12\log(mn),$$

and show that

$$\Pr(\mathcal{L}\cap\mathcal{U}^{\text{cmn}}_{\mathsf{w}} = \emptyset) = \Pr(\sum_{i=1}^{q} X_i = 0) \geq \Pr(\sum_{i=1}^{q} Y_i)(1-e^{-D}) \geq e^{-2\mathsf{t}\cdot\mathsf{s}\cdot\eta\cdot\sigma}/2.$$

Hence, since $\mathsf{t}\cdot\mathsf{s}\cdot\eta\cdot\sigma = \eta = O(1)$ (see Table 2),

$$\Pr(\text{In all runs, } \mathcal{L}\cap\mathcal{U}^{\text{cmn}} \neq \emptyset) \leq (1-e^{-2\mathsf{t}\cdot\mathsf{s}\cdot\eta\cdot\sigma})^{O(\log n)} \leq n^{-1}.$$

The second property follows from Lemma B.5: if the algorithm returns a value other than **infeasible**, then with probability at least $1-4m^{-1}$, $|C(\text{OPT})| \geq |\mathcal{U}|/(54\mathsf{f}\cdot\eta\cdot\alpha)$. □

LargeSet(k,α,w):
▷ Run LargeSetComplete on sampled set of elements
repeat $O(\log n)$ times **in parallel**
  let $\mathcal{L}\subseteq\mathcal{U}$ s.t. each $e\in\mathcal{L}$ w.p. $\rho = \frac{\mathsf{t}\cdot\mathsf{s}\alpha\cdot\eta}{|\mathcal{U}|}$
  $\mathsf{r}_1 \leftarrow \mathsf{s}_{\mathcal{L}}\alpha$, $\mathsf{r}_2 \leftarrow \frac{cm\log m}{\mathsf{w}}\cdot\gamma$ ▷ $\gamma := \frac{1944}{\mathsf{t}^2\mathsf{s}^2\log\alpha}$
  $\text{thr}_1 \leftarrow |\mathcal{L}|/(18\eta\mathsf{s}\alpha)$, $\text{thr}_2 \leftarrow |\mathcal{L}|/(6\eta\alpha)$
  sol $\leftarrow$ LargeSetComplete$(\mathcal{L},\mathsf{w},\mathsf{r}_1,\mathsf{r}_2,\text{thr}_1,\text{thr}_2)$
    if sol $\neq$ **infeasible then return** $|\mathcal{U}|/(54\mathsf{f}\cdot\eta\cdot\alpha)$
return infeasible

Figure 7: A single pass streaming algorithm.

Lemma B.7. *The amount of space used by LargeSet is $\widetilde{O}(\frac{m}{\alpha^2})$.*

Proof. Note that LargeSet performs $O(\log n)$ instances of LargeSetComplete in parallel. Hence, the total amount of space use by LargeSet is $O(\log n)$ times the space complexity of LargeSetComplete.

Similarly to the space analysis of LargeSetSimple, the amount of space to perform $\text{Cntr}_{\text{small}}$ and $\text{Cntr}_{\text{large}}$ as defined in LargeSetComplete is respectively $\widetilde{O}(1/\phi_1) = \widetilde{O}(m/\alpha^2)$ and $\widetilde{O}(1/\phi_2) = \widetilde{O}(1)$. Moreover, for the last case in which the contributing class has size larger than r_2, by Theorem 2.12, in total $\widetilde{O}(\frac{m}{\mathsf{w}\cdot\mathsf{r}_2}) = \widetilde{O}(1)$ space is required to compute the coverage of sampled supersets in $\mathcal{M}$. Note that, in all cases, by Lemma A.2, the algorithm can store h in $\widetilde{O}(1)$ space.

Hence, the total amount of space required to implement LargeSet is $\widetilde{O}(\frac{m}{\alpha^2})$. □

Proof of Theorem 4.8. The guarantee on the quality of the returned estimate follows from Theorem B.6 with $\mathsf{w} = \min\{\alpha,k\}$ and $\mathsf{s} = \widetilde{O}(\mathsf{w}/\alpha)$ (as in Table 2). Moreover, Lemma B.7 shows that the space complexity of LargeSet is $\widetilde{O}(\frac{m}{\alpha^2})$.

Moreover, since with high probability the estimate returned by the algorithm is a lower bound on the coverage size of a k-cover of $\mathcal{F}$, the output of LargeSet with high probability, is smaller than the optimal coverage size of Max k-Cover$(\mathcal{U},\mathcal{F})$. □