Simpler algorithm for estimating frequency moments of data streams

Lakshminath Bhuvanagiri[*] Sumit Ganguly[†] Deepanjan Kesh[‡] Chandan Saha[§]

Abstract

The problem of estimating the k^{th} frequency moment F_k over a data stream by looking at the items exactly once as they arrive was posed in [1, 2]. A succession of algorithms have been proposed for this problem [1, 2, 6, 8, 7]. Recently, Indyk and Woodruff [11] have presented the first algorithm for estimating F_k, for $k > 2$, using space $\tilde{O}(n^{1-2/k})$, matching the space lower bound (up to poly-logarithmic factors) for this problem [1, 2, 3, 4, 13] (n is the number of distinct items occurring in the stream.) In this paper, we present a simpler 1-pass algorithm for estimating F_k.

1 Introduction

Data streaming systems have many natural applications, such as database systems, network monitoring, sensor networks, RF-id data management, etc.. These applications are characterized by rapidly arriving voluminous data which makes it difficult to store the data in its entirety for either online processing or post-processing. Therefore, there has been a substantial interest in the design of algorithms that process data streams using single-pass (or online) algorithms that require sub-linear space.

We view a data stream as a sequence of arrivals of the form (i, v), where, i is the identity of an item that is a member of the domain $\mathcal{D} = \{0, 1, \ldots, N-1\}$, and v is the change in the frequency of the item. The value v may be either greater than or less than zero; $v \geq 1$ signifies v insertions of the item i, and $v \leq -1$ signifies v deletions of i. The frequency of an item i is denoted by f_i and is defined as the sum of the changes to its frequency since the inception of the stream, that is, $f_i = \sum_{(i,v) \text{ appears in stream}} v$. The k^{th} frequency moment of the stream, denoted by F_k, is defined as $F_k = \sum_i f_i^k$. The problem is to design an algorithm, parameterized by accuracy parameter ϵ and confidence parameter δ that returns an estimate $\hat{F}_k$ of F_k, for $k > 2$, in a single pass over the data stream, such that, $\mathsf{Pr}\left\{|\hat{F}_k - F_k| \leq \epsilon \cdot F_k\right\} \geq 1 - \delta$. Let n denote the number of items in the stream with positive frequencies and let m denote $\sum_{i \in \mathcal{D}} f_i$.

The problem was introduced in [1, 2] who also present the first sub-linear space algorithm with space complexity $\tilde{O}(n^{1-1/k})$. (We say that $f(n)$ is $\tilde{O}(g(n))$ if $f(n) = O\left(\left(\frac{1}{\epsilon}\right)^{O(1)} (\log m)^{O(1)} (\log n)^{O(1)} g(n)\right).$) [6, 8] present algorithms with space complexity $\tilde{O}(n^{1-1/(k-1)})$ and [7] presents an algorithm with space complexity $\tilde{O}(n^{1-2/(k+1)})$. A space lower bound of $\Omega(n^{1-\frac{2}{k}})$ is shown for this problem in a series of contributions [1, 2, 3, 4] (see[13] for a closely related problem). Recently, Indyk and Woodruff [11] have presented the first algorithm for estimating F_k using space $\tilde{O}(n^{1-2/k})$, matching the space lower bound (up to poly-logarithmic factors) for this problem [4].

The space complexity of the algorithm of Indyk and Woodruff has high constants and poly-logarithmic factors. Specifically, their 2-pass algorithm has space complexity of $O(\frac{1}{\epsilon^{12}}\ n^{1-\frac{2}{k}}(\log^2 n)(\log^6 m))$. The 1-pass algorithm that is derived from this algorithm further multiplies the constant and poly-logarithmic factors.

In this paper, we present a simpler algorithm for estimating F_k, for $k > 2$, whose space complexity is $O(\frac{k^2}{\epsilon^{2+4/k}}\ n^{1-\frac{2}{k}}(\log^2 m)(\log m + \log n))$. Broadly speaking, we use the seminal idea of Indyk and Woodruff [11] to classify items into groups, based on frequency. However, Indyk and Woodruff define groups whose boundaries are randomized; this technique contributes to the complexity of their algorithm. In our algorithm, the group boundaries are deterministic. Our analysis is simpler and uses the more traditional approach of directly calculating the expectation and the variance of the estimator for F_k. Finally, our algorithm is naturally a one-pass algorithm.

The remainder of the paper is organized as follows. In Section 2, we briefly review the CountSketch algorithm[5] and the estimator for the residual second moment [9]. The data structure for the F_k estimator is presented in Section 3. The analysis of the estimator is presented in Section 4. Finally, we conclude in Section 6.

[*] Indian Institute of Technology, Kanpur.
[†] Indian Institute of Technology, Kanpur.
[‡] Indian Institute of Technology, Kanpur.
[§] Indian Institute of Technology, Kanpur.

2 Preliminaries

In this section, we review the COUNTSKETCH algorithm for finding frequent items in a data stream and an algorithm to estimate the residual second moment of a data stream [9].

The residual second moment [5] of a data stream, denoted by $F_2^{res}(k)$, is defined as the second moment of the stream after the top-k frequencies have been removed. For $r = 1, 2, \ldots, n$, let $\text{rank}(r)$ denote an item whose frequency is the r^{th} largest frequency (ties are broken arbitrarily). Then, $F_2^{res}(k) = \sum_{r>k} f^2_{\text{rank}(r)}$.

A sketch [1, 2] is a random integer $X = \sum_i f_i \cdot x_i$, where, $x_i \in \{-1, +1\}$, for $i \in \mathcal{D}$ and the family of variables $\{x_i\}_{i \in \mathcal{D}}$ that are either pair-wise or 4-wise independent. The family of random variables $\{x_i\}_{i \in \mathcal{D}}$ is referred to as the *sketch basis.* 4-wise independent sketches were used by [1, 2] to present an elegant and efficient algorithm for estimating the second moment of a stream.

Pair-wise independent sketches are used in [5] to design the COUNTSKETCH algorithm for finding the top-k frequent items in an insert-only stream. The data structure consists of a collection of $s_3 = O(\log \frac{m}{\delta})$ independent hash tables each consisting of A buckets. A pair-wise independent hash function is associated with each hash table that maps items randomly to one of the A buckets. The structure can be used to estimate the frequency $\hat{f}_i$ of an item i. The estimation guarantees are stated as a function Δ of the residual second moment and is summarized below.

$$\Delta(s, A) = 8 \left(\frac{F_2^{res}(s)}{A} \right)^{1/2}$$

THEOREM 2.1. ([5]) *Let $s_3 = O(\log \frac{m}{\delta})$ and let $\Delta = \Delta(\frac{A}{8}, A)$. Then, for every item i, $\Pr\left\{ |\hat{f}_i - f_i| \le \Delta \right\} \ge 1 - \frac{\delta}{2 \cdot m}$. The space used is $O(A \cdot \log \frac{m}{\delta} \cdot (\log(m \cdot N))$ bits, and the time taken to process a stream update is $O(\log \frac{m}{\delta})$.* □

[9] presents an algorithm to estimate $F_2^{res}(s)$ to within an accuracy of $(1 \pm \epsilon)$ with confidence $1 - \delta$ using space $O(\frac{s}{\epsilon^2} \log(m \cdot N) \log \frac{m}{\delta})$ bits. The data structure used is identical to the COUNTSKETCH structure.

3 Estimator for F_k

In this section, we present our estimator for F_k.

3.1 Data structure and algorithm We use a data structure called the *hierarchical samples from sketches* structure, denoted by HSAMPLESKETCH, and parameterized by $A = O(\frac{1}{\epsilon^{4/k}} n^{1-\frac{2}{k}})$. The structure is divided into $L + 1$ levels, numbered from 0 through L, where, $L = \min(\lceil \log \frac{m^2}{A} \rceil, \log N)$. Each level uses a COUNTSKETCH data structure [5] with $s_1 = O(\log(m \cdot L \cdot A))$ independent hash tables of size B buckets each. A random hash function $h : \mathcal{D} \rightarrow \mathcal{D}$ is chosen from a t-wise independent hash family (t is fixed later) and is used to define a random mapping of items $i \in \mathcal{D} \ldots \{1, \ldots, L\}$ as follows: $\text{level}(i) = \text{lsb}(h(i))$, where, $\text{lsb}(a)$ denotes the least significant bit position of a. Each stream update of the form (i, v) is propagated to the COUNTSKETCH structure at level 0 and to the COUNTSKETCH structures at each level l, where, $l \le \text{level}(i)$. Thus, item i maps to level 0 with probability 1, to level 1 with probability $\frac{1}{2}$, to level 2 with probability $\frac{1}{4}$, etc..

Using the algorithm of [9], we obtain an estimate $\hat{F}_2^{res}(A)$ for $F_2^{res}(A)$ that is accurate to within a factor of $1 \pm \frac{\epsilon}{4k}$ with probability at least $1 - \frac{1}{30}$. For each level l, $0 \le l \le L$, we define a threshold S_l as follows.

$$S_l = 8 \left(\frac{\hat{F}_2^{res}(A)}{A \cdot 2^{l+1}} \right)^{\frac{1}{2}}$$

Thus, $S_l = \frac{S_0}{2^{l/2}}$, $l = 1, 2, \ldots, L$. Let $\hat{f}_{i,l}$ denote the estimated frequency of item i obtained from the COUNTSKETCH structure at level l, assuming that i has mapped to level l. The table size parameter B of the COUNTSKETCH algorithm is chosen such that if $\hat{f}_{i,l}$ crosses the threshold S_l, then, with probability $1 - \frac{1}{30mAL}$, the estimate is within a factor of $(1 \pm \epsilon)$ of the true frequency f_i. If the estimated frequency of an item i crosses the thresholds S_l at multiple levels, then we *disambiguate* the estimated frequency in favor of the estimate that is returned from the lowest among such levels l where the estimate crosses the threshold S_l. The estimated frequency of an item i is denoted by $\hat{f}_i$ (after possible disambiguation). Items whose estimated frequencies do not cross the threshold S_l at any level l are said to have undefined estimates. Items with defined estimates are classified into sample groups $\hat{G}_l$, for $0 \le l \le L$, according to the following rule. Let $T_0 = 2^{\frac{1}{4}} S_0$ and $T_l = \frac{S_{l-1}}{2^{1/4}}$, for $0 \le l \le L$ (i.e., $T_l = (S_{l-1} S_l)^{1/2}$).

$$\hat{G}_0 = \{ i \mid \hat{f}_i \ge T_0 \},$$

$$\hat{G}_l = \{ i \mid T_l \le \hat{f}_i < T_{l-1} \text{ and } \text{level}(i) \ge l \}, 1 \le l \le L \ .$$

The estimate of F_k is obtained as follows.

$$\hat{F}_k = \sum_{l=0}^{L} \sum_{i \in \hat{G}_l} \hat{f}_i^k \cdot 2^l \qquad (3.1)$$

There are $L + 1$ levels per structure, where, $L = O(\log \frac{m^2}{A})$. Each level keeps $O(\log m)$ hash tables, each of size $B = O(\frac{k^2 A}{\epsilon^2})$ buckets, where, $A = O(\frac{1}{\epsilon^{4/k}} n^{1-\frac{2}{k}})$. Each bucket stores a sketch of size $\log m$ bits. The total space complexity is therefore $O(\frac{k^2}{\epsilon^{2+4/k}} n^{1-\frac{1}{k}} \log^2 m)$ bits. An additional factor of $O(\log m)$ is introduced due to our use of Nisan's pseudo-random generator[12] (see § 5).

3.2 Discussion The level function randomly maps item i to level l with probability $\frac{1}{2^l}$. The COUNTSKETCH structure at each level is used to retrieve the top-A items in terms of their estimated frequencies from the set of items that map to level l. The size of the hash table $B = O(\frac{k^2}{\epsilon^2} A)$ used by the COUNTSKETCH structures is selected in such a way that the error of estimation at level l is at most $\frac{\epsilon S_l}{4k}$, that is, $|\hat{f}_i - f_i| < \frac{\epsilon S_l}{4k}$, with probability at least $1 - \frac{1}{30mAL}$. Our first part of the analysis is essentially devoted to establishing this fact.

We classify items into groups G_0 through G_L based on their frequencies, by extending the classification used by the algorithm, as follows: $G_0 = \{i \mid f_i \geq T_0\}$ and $G_l = \{i \mid T_l \leq f_i \leq T_{l-1}\}$. Let i belong to group G_l. We say that an item i is on the left margin of G_l, provided, $T_l \leq f_i \leq T_l + \frac{\epsilon}{4k} S_l$. Since, the error of estimation at level l is bounded by $\frac{\epsilon S_l}{4k}$, it follows that the estimated frequency of such an item may cross the boundary of group G_l and may be misclassified to belong to group G_{l+1}. The right margin of a group is defined analogously. The remainder of the group is called the middle-region; items that lie in the middle region of a group are correctly classified into the same group (with probability $1 - \frac{1}{30mAL}$).

The thresholds T_l and S_l are constructed such that if an item i is on the left margin of a group G_l, then, its estimate $\hat{f}_{i,l'}$ does not cross the threshold $S_{l'}$ for accurate estimation at any level $l' < l$ (with probability $1 - \frac{1}{30mAL}$). This can be seen by substituting the definitions of the thresholds, as follows. We assume that $\epsilon \leq 1$ and $k \geq 2$.

$$\hat{f}_{i,l'} \leq f_i + \frac{\epsilon}{4k} S_{l'} \leq (1 + \frac{\epsilon}{4k}) T_l + \frac{\epsilon}{4k} S_{l'} < S_{l'} \ .$$

Therefore, an item that belongs to the left margin of group G_l is not "discovered" at any level lower than l. Further, if i maps to level l, then its estimate crosses the threshold at level l (with probability $1 - \frac{1}{30mAL}$) implying that it is "discovered" at this level.

$$\hat{f}_{i,l} \geq f_{i,l} - \frac{\epsilon}{4k} S_l \geq T_l (1 - \frac{\epsilon}{4k}) > S_l \text{ since, } \frac{\epsilon}{4k} < \frac{1}{\sqrt{2}}.$$

Thus we can conclude that if an item belongs to the left margin of some group G_l and it hashes to level l, then, its estimate $\hat{f}_i$ is obtained from the sub-structure at level l. This however, does not mean that such items are not misclassified. Indeed, due to errors of estimation, items that lie in the left margin of a group G_l may be classified into group $\hat{G}_l$ or in the group $\hat{G}_{l+1}$. However, in either case, its estimate $\hat{f}_i$ is obtained (with probability $1 - \frac{1}{30mAL}$) from the structure at level l. An analogous discussion can be carried out for right margin items.

4 Analysis

In this section, we present an analysis of the estimator for F_k. We show that with probability at least $1 - \frac{1}{30mAL}$, every item that hashes to level l and whose true frequency exceeds S_l is estimated accurately with an error of at most $\pm \frac{\epsilon S_l}{4k}$. We then calculate the expectation and the variance of the estimator.

4.1 Accurate Retrieval of frequent items We first show that the number of items that cross the threshold S_l at level l is $O(A)$. Let H_l denote the set of items that map to level l and whose estimated frequencies cross the threshold S_l. Assume that $\epsilon \leq 1$.

LEMMA 4.1. *$|H_0| \leq \frac{5A}{4}$ and $|H_l| \leq A$, for $1 \leq l \leq L$, with combined probability at least $1 - \frac{1}{30}$.*

Proof. Since, the estimation error at level l is $\frac{\epsilon}{4k} S_l$, each item that maps to level l and whose estimated frequency crosses the threshold T_l has true frequency at least $U'_l = T_l - \frac{\epsilon}{4k} S_l = 8\left(2^{1/4} - \frac{\epsilon}{4k}\right)\left(\frac{\hat{F}_2^{res}(A)}{A \cdot 2^{l+1}}\right)^{1/2}$. By an application of the residual second moment algorithm, $\hat{F}_2^{res}(A) \geq F_2^{res}(A)\left(1 - \frac{\epsilon}{4k}\right)$. Thus, $U'_l \geq 8\left(1 - \frac{\epsilon}{4k}\right)^{1/2}\left(2^{1/4} - \frac{\epsilon}{4k}\right)\left(\frac{F_2^{res}(A)}{A \cdot 2^{l+1}}\right)^{1/2}$. Therefore, the number of items in f with true frequency exceeding U'_l is at most $n'_l = A + \frac{2^{l+1} A}{8 \cdot \left(1 - \frac{\epsilon}{4k}\right)^{1/2}\left(2^{1/4} - \frac{\epsilon}{4k}\right)} \leq A + 2^{l-2} A$, since, $\epsilon \leq 1$. Therefore, $|H_0| \leq \frac{5A}{4}$ and $\mathsf{E}\big[|H_l|\big] \leq \frac{n'_l}{2^l} \leq \frac{A}{2^l} + \frac{A}{4}$. By Chernoff's bounds, it follows that $|H_l| \leq A$, for $l \geq 1$. □

LEMMA 4.2.

$$F_2^{res}(l, s) \leq \max\left(\frac{F_2^{res}(2^{l-1} s)}{2^{l-1}}, O(\log(mAL)\right) \ .$$

Proof. Let $s = \Omega(\log m)$. Consider the top-$2^{l-1}s$ items. The expected number of these items that map to level l is $\frac{2^{l-1}s}{2^l} = \frac{s}{2}$. By Chernoff's bounds, the actual number of these items that map to level l is at most $\frac{3s}{4}$ (since, $s = O(\log m)$), with probability $1 - \frac{1}{30mAL}$. Therefore, the removal of the top-s frequencies from those items that map to level l removes all the items that are in the top-$2^{l-1}s$ ranks. Therefore, $\mathsf{E}\big[F_2^{res}(l, s)\big] \leq \frac{F_2^{res}(2^{l-2}s)}{2^l}$, since, each item is mapped to level l with probability 2^l, with probability $1 - \log\frac{1}{30mAL}$. Using Hoeffding's bounds, $F_2^{res}(l,s) \le \max(2\mathsf{E}\big[F_2^{res}(l,s)\big], O(\log(mAL))$. □

The above inference can also be obtained using generalizations of Chernoff-Hoeffding bounds [14], by assuming that h is $O(\log L)$-wise independent. However, we later use Nisan's generator to "simulate" full independence using fewer random bits.

We can now show that the COUNTSKETCH structure at each level can be given sufficient space so that all items of H_l can be retrieved with error at most $\frac{\epsilon \cdot S_l}{4k}$ (with probability $1 - \frac{1}{30mAL}$). By Theorem 2.1, this is possible, provided, $8 \cdot \left(\frac{F_2^{res}(l, \frac{B}{8})}{B}\right)^{1/2} \le \frac{\epsilon S_l}{4k}$ and $|H_l| \le \frac{B}{8}$. By Lemmas 4.1 and 4.2, this is implied by $B \ge \frac{16k^2 A}{\epsilon^2}$.

4.2 Expectation and Variance For $i \in \mathcal{D}$ and $0 \le l \le L$, let $x_{i,l}$ denote an indicator random variable, that is 1 if i is classified into group $\hat{G}_l$ and i hashes to level l. In the following, we assume that the COUNTSKETCH algorithm at level l makes two-sided errors of magnitude at most $\frac{\epsilon S_l}{4k}$ with probability 1; the error probabilities will be added later using the union bound.

LEMMA 4.3. *Assume that the hash function mapping the items to levels is fully independent. Let i be an item that lies on the left margin of the group G_l. Then,* $\mathsf{E}\big[2^l x_{i,l} + 2^{l+1}x_{i,l+1}\big] = 1$.

Proof. From the above discussion, we note that since f_i belongs to the left margin of group G_l, item i is discovered as a candidate frequent item (i.e., $\hat{f}_i \ge S_l$) only at level l and not earlier. Given that i maps to level l, let p be the probability that i is classified into $\hat{G}_l$. Since, the disambiguation of multiple estimates is done in favor of the estimate obtained from the lowest level, therefore, p remains the probability (assuming fully independent hash function) that i is classified into $\hat{G}_l$, irrespective of whether i maps to level $l+1$ or not. The probability that i maps to both levels l and $l+1$ is $\frac{1}{2^{l+1}}$ and the probability that i maps to only level l is also $\frac{1}{2^{l+1}}$. Therefore, $\mathsf{Pr}\{x_{i,l} = 1\} \cdot 2^l + \mathsf{Pr}\{x_{i,l+1} = 1\} \cdot 2^{l+1} = \left(\frac{p}{2^{l+1}} + \frac{p}{2^{l+1}}\right) \cdot 2^l + \frac{1-p}{2^{l+1}} \cdot 2^{l+1} = 1$. □

An analogous property holds for items that lie on the right margin of a group. Items that lie in the middle region of group G_l are never misclassified (with probability at least $1 - \frac{1}{30mAL}$), and therefore, $\mathsf{Pr}\{x_{i,l} = 1\} = \frac{1}{2^l}$, and $\mathsf{Pr}\{x_{i,l'} = 1\} = 0$, for $l' \ne l$. It follows that

$$\mathsf{E}\big[\sum_{r=0}^{L} x_{i,r} \cdot 2^r\big] = 1, \quad \text{for } i \in \text{ stream.}$$

The estimator for F_k can be rewritten as follows:

$$\hat{F}_k = \sum_{i \in \mathcal{D}} \hat{f}_i^k \cdot x_{i,l} \cdot 2^l \ .$$

Let $\tilde{F}_k$ denote the following expression:

$$\tilde{F}_k = \sum_{i \in \mathcal{D}} f_i^k x_{i,l} 2^l \ .$$

For each item i that is included in any one of the groups, $|\hat{f}_i - f_i| \le \frac{\epsilon}{4k} f_i$ (with probability $1 - \frac{1}{30mAL}$), which implies that $|\hat{f}_i^k - f_i^k| \le \frac{\epsilon}{2} f_i^k$. Therefore,

$$\text{(4.2)} \quad |\hat{F}_k - \tilde{F}_k| \le \sum_{i \in \mathcal{D}} |\hat{f}_i^k - f_i^k| x_{i,l} 2^l \le \sum_{i \in \mathcal{D}} \frac{\epsilon f_i^k}{2} \cdot x_{i,l} \cdot 2^l = \frac{\epsilon \tilde{F}_k}{2}, \text{ with probability } 1 - \frac{1}{20} \ .$$

The total error probability is at most $\frac{1}{20}$, since, by Lemma 4.1, there are at most $\frac{5A}{4}$ frequent items discovered at each level. Adding the error probability of $\frac{1}{30mAL}$ for each of these items for each of $L+1$ levels gives the stated probability.

LEMMA 4.4. $\mathsf{E}\big[\tilde{F}_k\big] = F_k$.

Proof. By Lemma 4.3, it follows that

$$\begin{aligned} \mathsf{E}\big[\tilde{F}_k\big] &= \mathsf{E}\big[\sum_{i \in \mathcal{D}} \sum_{l=0}^{L} f_i^k x_{i,l} \cdot 2^l\big] \\ &= \sum_{i \in \mathcal{D}} f_i^k \ \mathsf{E}\big[\sum_{l=0}^{L} x_{i,l} \cdot 2^l\big] \\ &= \sum_{i \in \mathcal{D}} f_i^k = F_k. \quad \square \end{aligned}$$

We now consider the calculation for $\mathsf{Var}\big[\tilde{F}_k\big]$. Let $F_r(G_l)$ be the r^{th} frequency moment of the items in the group G_l, that is, $F_r(G_l) = \sum_{i \in G_l} f_i^r$. Let $\mathsf{lmargin}(G_l)$ denote the left margin of the group G_l, that is, $\mathsf{lmargin}(G_l) = \{i \mid T_l \le f_i \le T_l + \frac{\epsilon}{4k} S_l\}$.

LEMMA 4.5. $\mathsf{Var}\big[\tilde{F}_k\big] \le (F_k(\mathsf{lmargin}(G_0)))^2 + \sum_{l=1}^{L} F_{2k}(G_l) 2^{l+1}$.

Proof.

$$
\begin{aligned}
\mathsf{Var}\big[\tilde{F}_k\big] &= \mathsf{E}\big[(\tilde{F}_k)^2\big] - (\mathsf{E}\big[\tilde{F}_k\big])^2 \\
&= \mathsf{E}\Big[\left(\sum_{l=0}^{L}\sum_{i\in\mathcal{D}} f_i^k \cdot x_{i,l}\cdot 2^l\right)^2\Big] - F_k^2 \\
&= \mathsf{E}\Big[\sum_{i\in\mathcal{D}} f_i^{2k}(\sum_{l=0}^{L} x_{i,l}\cdot 2^l)^2 \\
&\qquad + \sum_{i\neq j} f_i^k f_j^k (\sum_{r=0}^{L} x_{i,r}2^r)\cdot(\sum_{s=0}^{L} x_{i,s}2^s)\Big] - F_k^2 \\
&= \sum_{l=0}^{L}\sum_{i\in G_l} f_i^{2k}\sum_{r=0}^{L}\mathsf{Pr}\left\{x_{i,r}=1\right\}2^{2r} \\
&\qquad + \sum_{i\neq j} f_i^k f_j^k \mathsf{E}\big[(\sum_{r=0}^{L} x_{i,r}2^r)\cdot(\sum_{s=0}^{L} x_{i,s}2^s)\big] - F_k^2
\end{aligned}
$$

by linearity of expectation, and, since $x_{i,l}=1$ for at most one level l. For an item i, we now calculate $\sum_{l=0}^{L}\mathsf{Pr}\left\{x_{i,l}=1\right\}2^{2l}$ and for $i\neq j$, $\mathsf{E}\big[(\sum_{r=0}^{L} x_{i,r}2^r)\cdot(\sum_{s=0}^{L} x_{i,s}2^s)\big]$.

Consider a calculation similar to that of Lemma 4.3. If i belongs to the middle region of some group G_l, then i is correctly classified into group l. Thus,

$$\sum_{r=0}^{L}\mathsf{Pr}\left\{x_{i,r}=1\right\}2^{2r} = \mathsf{Pr}\left\{x_{i,l}=1\right\}2^{2l} = \frac{2^{2l}}{2^l} = 2^l.$$

Now suppose that i lies on the left margin of a group G_l and let p be the probability that i is classified into group G_l given that i maps to level l. Then,

$$
\begin{aligned}
&\sum_{r=0}^{L}\mathsf{Pr}\left\{x_{i,r}=1\right\}2^{2r} \\
&= \mathsf{Pr}\left\{x_{i,l}=1\right\}2^{2l} + \mathsf{Pr}\left\{x_{i,l+1}=1\right\}2^{2l+2} \\
&= \frac{p\cdot 2^{2l}}{2^l} + \frac{(1-p)\cdot 2^{2l+2}}{2^{l+1}} \\
&= p\cdot 2^l + (1-p)\cdot 2^{l+1} \\
&\leq 2^{l+1}
\end{aligned}
$$

assuming that the hash function is fully independent. An analogous argument can be made for items that belong to the right margin of groups. We therefore have the following.

$$
\mathsf{E}\big[\sum_{r=0}^{L} x_{i,r}2^{2r}\big] \text{ is } \begin{cases} \leq \;\; 2^{l+1} & \text{if } i\in G_l \text{ and } l\geq 1 \\ \leq \;\; 2 & \text{if } i\in \mathsf{lmargin}(G_0) \\ = \;\; 1 & \text{if } i\in G_0 - \mathsf{lmargin}(G_0)). \end{cases}
$$

For $i\neq j$, by pair-wise independence, $\mathsf{E}\big[(\sum_{r=0}^{L} x_{i,r}2^r)\cdot(\sum_{s=0}^{L} x_{j,s}2^s)\big] = \mathsf{E}\big[\sum_{r=0}^{L} x_{i,r}2^r\big]\cdot\mathsf{E}\big[\sum_{s=0}^{L} x_{j,s}2^s\big] = 1\cdot 1 = 1$. Therefore, the expression for $\mathsf{Var}\big[\tilde{F}_k\big]$ can be written as follows.

$$
\begin{aligned}
\mathsf{Var}\big[\tilde{F}_k\big] &\leq \sum_{l=0}^{L}\sum_{i\in G_l}\mathsf{E}\big[\sum_{r=0}^{L} x_{i,r}2^{2r}\big]f_i^{2k} + \sum_{i\neq j} f_i^k f_j^k - F_k^2 \\
&= \sum_{l=0}^{L}\sum_{i\in G_l}\mathsf{E}\big[\sum_{r=0}^{L} x_{i,r}2^{2r}\big]f_i^{2k} - F_{2k} \\
&\leq \sum_{l=1}^{L}\sum_{i\in G_l} f_i^{2k}2^{l+1} + \sum_{i\in\mathsf{lmargin}(G_0)} 2\cdot f_i^{2k} \\
&\qquad + \sum_{i\in G_0-\mathsf{lmargin}(G_0)} f_i^{2k} - F_{2k} \\
&\leq (F_k(\mathsf{lmargin}(G_0)))^2 + \sum_{l=1}^{L} F_{2k}(G_l)2^{l+1} \quad \square
\end{aligned}
$$

Lemma 4.6. *If $\epsilon \leq 1$ and $A \geq \frac{2^{3+\frac{8}{k}}}{\epsilon^{4/k}}(n^{1-\frac{2}{k}})$, then, $\mathsf{Var}\big[\tilde{F}_k\big] \leq \frac{\epsilon^2}{32}F_k^2$.*

Proof. $\frac{F_k^{res}(A)}{n-A} \geq \left(\frac{F_2^{res}(A)}{n-A}\right)^{k/2}$. Therefore, $(F_2^{res}(A))^k \leq (n-A)^{k-2}(F_k^{res}(A))^2 < n^{k-2}(F_k^{res}(A))^2$.

$$
\begin{aligned}
&\sum_{l=1}^{L} F_{2k}(G_l)2^{l+1} \\
&\leq \sum_{l=1}^{L} T_{l-1}^k F_k(G_l)2^{l+1}, \; \{\text{ by defn. of group } G_l\} \\
&\leq \sum_{l=1}^{L} 8^k\left(\frac{\hat{F}_2^{res}(A)}{A\cdot 2^l}\right)^{k/2} 2^{l+1}, \quad \{\text{ by defn. of } T_{l-1}\} \\
&\leq 2\cdot 8^k\left(1+\frac{\epsilon}{4k}\right)^{k/2}\sum_{l=1}^{L}(F_2^{res}(A))^{k/2}\frac{2^l}{A^{k/2}2^{kl/2}}F_k(G_l) \\
&\qquad\qquad \{\text{ using } \hat{F}_2^{res}A \leq (1+\frac{\epsilon}{4k})F_2^{res}(A)\} \\
&< 2\cdot 8^k(1+\frac{\epsilon}{2})\sum_{l=1}^{L}\frac{n^{k-2}}{A^k}F_k^{res}(A)F_k(G_l)\frac{1}{2^{l(k/2-1)}} \\
&\qquad\qquad \{\text{ using } (F_2^{res}(A))^k < n^{k-2}(F_k^{res}(A))^2\} \\
&= \frac{\epsilon^2}{64}F_k^{res}(A)\sum_{l=1}^{L}\frac{F_k(G_l)}{2^{l(k/2-1)}} \\
&\leq \frac{\epsilon^2}{64}F_k^{res}(A)\sum_{l=1}^{L}F_k(G_l), \qquad \{\text{ since, } k\geq 2\ \} \\
&\leq \frac{\epsilon^2}{64}F_k^{res}(A)F_k \quad \leq \quad \frac{\epsilon^2}{64}F_k^2\ .
\end{aligned}
$$

Analogously, it can be shown that

$$F_{2k}(\mathsf{lmargin}(G_0)) \le (T_0)^k(1+\frac{\epsilon}{2})F_k(\mathsf{lmargin}(G_0)) \le \frac{\epsilon^2}{64}F_k(\mathsf{lmargin}(G_0))F_k^{res}(A) \le \frac{\epsilon^2}{64}F_k^2 \ .$$

By Lemma 4.5, we obtain that

$$\mathsf{Var}\big[\tilde{F}_k\big] \le \sum_{l=1}^{L} F_{2k}(G_l)2^{l+1} + F_{2k}(\mathsf{lmargin}(G_0)) \le \frac{\epsilon^2}{64}F_k^2 + \frac{\epsilon^2}{64}F_k^2 = \frac{\epsilon^2}{32}F_k^2 \ . \ \square$$

Theorem 4.1. *If $\epsilon \le 1$ and $A \ge \frac{2^{3+\frac{8}{k}}}{\epsilon^{4/k}}(n^{1-\frac{2}{k}})$, then,* $\mathsf{Pr}\left\{|\hat{F}_k - F_k| \le \epsilon F_k\right\} \ge \frac{2}{3}$.

Proof. By Chebychev's inequality and Lemma 4.6

$$\mathsf{Pr}\{|\tilde{F}_k - F_k| \le \frac{\epsilon F_k}{3}\} \ge 1 - \frac{9\mathsf{Var}\big[\tilde{F}_k\big]}{\epsilon^2 F_k^2} \ge 1 - \frac{9}{32}$$

since, $\mathsf{E}\big[\tilde{F}_k\big] = F_k$ (by Lemma 4.4). Therefore, using triangle inequality,

$$|\hat{F}_k - F_k| \le |\hat{F}_k - \tilde{F}_k| + |\tilde{F}_k - F_k| \le \frac{\epsilon\tilde{F}_k}{2} + \frac{\epsilon F_k}{2}, \text{ by } (4.2)$$

with probability $1-\frac{9}{32}-\frac{1}{20} > \frac{2}{3}$. Since, $\tilde{F}_k \le (1+\frac{\epsilon}{3})F_k$, with error probability that is already accounted for in $\frac{9}{32}+\frac{1}{20} < \frac{1}{3}$, we have,

$$|\hat{F}_k - F_k| \le \left(\frac{\epsilon}{2}\cdot(1+\frac{\epsilon}{3}) + \frac{\epsilon}{2}\right)F_k \le \epsilon F_k$$

with probability at least $\frac{2}{3}$. Therefore, $\mathsf{Pr}\left\{|\hat{F}_k - F_k| > \epsilon F_k\right\} \ge \frac{2}{3}$. □

5 Reducing Randomness by using PRG

We use a standard technique of reducing the randomness by using a pseudo-random generator (PRG) of Nisan [12] along the lines of Indyk in [10] and Indyk and Woodruff in [11].

Notation on PRG [12, 10]. Let M be a finite state machine that uses S bits and has running time R. Assume that M uses the random bits in k segments, each segment consisting of kb bits. Let U^r be a uniform distribution over $\{0,1\}^r$ and for a discrete random variable X, let $\mathcal{F}[X]$ denote the probability distribution of X, treated as a vector. Let $M(x)$ denote the state of M after using the random bits in x. The generator $G : \{0,1\}^u \rightarrow \{0,1\}^{kb}$ expands a "small" number of u bits that are truly random to a sequence of kb bits that "appear" random to M. G is said to be a pseudo-random generator for a class $\mathcal{C}$ of finite state machines with parameter ϵ, provided, for every $M \in \mathcal{C}$

$$\left|\mathcal{F}[M_{x\in U^{kb}}(x)] - \mathcal{F}[M_{x \in U^m}(G(x))]\right|_1 \le \epsilon$$

where, $|y|_1$ denotes the L_1 norm of the vector y. Nisan [12] shows the following property (the version is from [10]).

Theorem 5.1. ([12]) *There exists a* PRG *G for Space(S) and Time(R) with parameter $\epsilon = 2^{-O(S)}$ that requires $O(S)$ bits such that G expands $O(S\log R)$ bits into $O(R)$ bits.* □

The PRG G given by the above theorem can be used to obtain the random bits for our algorithm. The space required by the algorithm is $S = O(\frac{k^2}{\epsilon^{2+2/k}}n^{1-\frac{2}{k}}\log^2 m)$ bits. The running time of the algorithm is at most $O(m\cdot n)$, since, corresponding to each of the m stream updates, the data structure can be updated in time $O(n)$. Therefore, by Theorem 5.1, there exists a PRG G that expands $O(S\log R) = O(\frac{k^2}{\epsilon^{2+4/k}} \ n^{1-\frac{2}{k}}\log^2 m(\log m + \log n))$ truly random bits into $O(R)$ bits. We summarize the main result of the paper as follows.

Theorem 5.2. *There exists an algorithm that computes $\hat{F}_k$ satisfying* $\mathsf{Pr}\left\{|\hat{F}_k - F_k| \le \epsilon F_k\right\} \ge \frac{2}{3}$ *using $O(\frac{k^2}{\epsilon^{2+4/k}}\cdot n^{1-\frac{2}{k}}\cdot(\log^2 m)\cdot(\log m + \log n))$ bits.* □

6 Conclusions

We present an algorithm to estimate the kth frequency moment of a data stream using space $O(\frac{k^2}{\epsilon^{2+2/k}}n^{1-\frac{2}{k}}\log^2 m(\log m + \log n))$ bits that matches the space lower bound of $\Omega(n^{1-\frac{2}{k}})$ bits for this problem [3, 4] up to logarithmic factors. It is simpler than the first algorithm for this problem presented by Indyk and Woodruff in [11].

References

[1] Noga Alon, Yossi Matias, and Mario Szegedy. "The Space Complexity of Approximating the Frequency Moments". In *Proceedings of the 28th Annual ACM Symposium on the Theory of Computing (STOC)*, May 1996.

[2] Noga Alon, Yossi Matias, and Mario Szegedy. "The space complexity of approximating frequency moments". *Journal of Computer Systems and Sciences*, 58(1):137–147, 1998.

[3] Ziv Bar-Yossef, T.S. Jayram, Ravi Kumar, and D. Sivakumar. "An information statistics approach to data stream and communication complexity". In *Proceedings of the 34th ACM Symposium on Theory of Computing (STOC)*, 2002.

[4] Amit Chakrabarti, Subhash Khot, and Xiaodong Sun. "Near-Optimal Lower Bounds on the Multi-Party Communication Complexity of Set Disjointness". In *Proceedings of the 18th Annual IEEE Conference on Computational Complexity, CCC 2003*, 2003.

[5] Moses Charikar, Kevin Chen, and Martin Farach-Colton. "Finding frequent items in data streams". In *Proceedings of the 29th International Colloquium on Automata Languages and Programming*, 2002.

[6] Don Coppersmith and Ravi Kumar. "An improved data stream algorithm for estimating frequency moments". In *Proceedings of the Fifteenth ACM SIAM Symposium on Discrete Algorithms*, 2004.

[7] Sumit Ganguly. "A hybrid technique for estimating frequency moments over data streams". Manuscript, July, 2004.

[8] Sumit Ganguly. "Estimating Frequency Moments of Update Streams using Random Linear Combinations". *Proceedings of the 8th International Workshop on Randomized Algorithms (RANDOM)*, August 2004.

[9] Sumit Ganguly, Deepanjan Kesh, and Chandan Saha. "Practical Algorithms for Tracking Database Join Sizes". In *Proceedings of the International Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2005, (To appear)*, pages 294–305, December 2005.

[10] Piotr Indyk. "Stable Distributions, Pseudo Random Generators, Embeddings and Data Stream Computation". In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, pages 189–197, November 2000.

[11] Piotr Indyk and David Woodruff. "Optimal Approximations of the Frequency Moments". In *Proceedings of the 37th ACM Symposium on Theory of Computing (STOC)*, June 2005.

[12] Noam Nisan. "Pseudo-Random Generators for Space Bounded Computation". In *Proceedings of the 30th Annual ACM Symposium on the Theory of Computing (STOC), 1990*, pages 204–212, May 1990.

[13] M. Saks and X. Sun. "Space lower bounds for distance approximation in the data stream model". In *Proceedings of the 34th ACM Symposium on Theory of Computing (STOC)*, 2002.

[14] Jeanette Schmidt, Alan Siegel, and Aravind Srinivasan. "Chernoff-Hoeffding Bounds with Applications for Limited Independence". In *Proceedings of the 3rd Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 331–340, 1992.