Internet Mathematics Vol. 3, No. 1: 63-78

Estimating Entropy and Entropy Norm on Data Streams

Amit Chakrabarti, Khanh Do Ba, and S. Muthukrishnan

Abstract. We consider the problem of computing information-theoretic functions, such as entropy, on a data stream, using sublinear space.

Our first result deals with a measure we call the *entropy norm* of an input stream: it is closely related to entropy but is structurally similar to the well-studied notion of frequency moments. We give a polylogarithmic-space, one-pass algorithm for estimating this norm under certain conditions on the input stream. We also prove a lower bound that rules out such an algorithm if these conditions do not hold.

Our second group of results is for estimating the empirical entropy of an input stream. We first present a sublinear-space, one-pass algorithm for this problem. For a stream of m items and a given real parameter α, our algorithm uses space $\widetilde{O}(m^{2\alpha})$ and provides an approximation of $1/\alpha$ in the worst case and $(1+\varepsilon)$ in "most" cases. We then present a two-pass, polylogarithmic-space, $(1+\varepsilon)$-approximation algorithm. All our algorithms are quite simple.

1. Introduction

Algorithms for computational problems on data streams have been the focus of plenty of recent research in several communities, such as theory, databases, and networks [Alon et al. 96, Estan and Varghese 03, Babcock et al. 06, Muthukrishnan 06]. In this model of computation, the input is a stream of "items" that is too long to be stored completely in memory, and a typical problem involves computing some statistics on this stream. The main challenge is to design algorithms that are efficient not only in terms of running time, but also in terms of

1542-7951/05 $0.50 per page

space (i.e., memory usage): sublinear space is a must and polylogarithmic space is often the goal.

The seminal paper of Alon, Matias, and Szegedy [Alon et al. 96] considered the problem of estimating the *frequency moments* of the input stream: if a stream contains m_i occurrences of item i (for $1 \leq i \leq n$), its kth frequency moment is denoted F_k and is defined by $F_k := \sum_{i=1}^{n} m_i^k$. Alon et al. showed that F_k could be estimated arbitrarily well in sublinear space for all nonnegative integers k and in polylogarithmic (in m and n) space for $k \in \{0, 1, 2\}$. Their algorithmic results were subsequently improved by Coppersmith and Kumar [Coppersmith and Kumar 04] and Indyk and Woodruff [Indyk and Woodruff 05].

In this work, we first consider a somewhat related statistic of the input stream, inspired by the classic information-theoretic notion of entropy. We consider the *entropy norm* of the input stream, denoted F_H and defined by $F_H := \sum_{i=1}^{n} m_i \lg m_i$.[1] We prove (see Theorem 2.2) that F_H can be estimated arbitrarily well in polylogarithmic space provided that its value is not "too small," a condition that is satisfied if, e.g., the input stream is at least twice as long as the number of distinct items in it. We also prove (see Theorem 2.5) that F_H cannot be estimated well in polylogarithmic space if its value *is* "too small."

Second, we consider the estimation of entropy itself, as opposed to the entropy norm. Any input stream implicitly defines an *empirical* probability distribution on the set of items it contains; the probability of item i being m_i/m, where m is the length of the stream. The *empirical entropy* of the stream, denoted H, is defined to be the entropy of this probability distribution:

$$H \;:=\; \sum_{i=1}^{n} (m_i/m) \lg(m/m_i) \;=\; \lg m - F_H/m \,. \tag{1.1}$$

An algorithm that computes F_H exactly clearly suffices to compute H as well. However, since we are only able to approximate F_H in the data stream model, we need new techniques to estimate H. We prove (see Theorem 3.1) that H can be approximated using sublinear space. Although the space usage is not polylogarithmic in general, our algorithm provides a tradeoff between space and approximation factor and can be tuned to use space arbitrarily close to polylogarithmic space.

The standard data stream model allows us only one pass over the input. If, however, we are allowed *two passes* over the input but still restricted to small space, we have an algorithm that approximates H to within a factor of $1+\varepsilon$ and uses polylogarithmic space (see Theorem 3.5).

[1] Throughout this paper "lg" denotes logarithm to the base 2.

1.1. Background and Motivation

Both entropy and entropy norm are natural statistics to approximate on data streams. Arguably, entropy-related measures are even more natural than L_p norms or frequency moments F_k. In addition, they have direct applications. The quintessential need arises in analyzing IP network traffic at the packet level on high-speed routers. In monitoring IP traffic, one cares about anomalies. In general, anomalies are hard to define and detect since there are subtle intrusions, sophisticated dependence amongst network events, and agents gaming the attacks. A number of recent results in the networking community have, however, converged on monitoring entropy as a reasonable approach [Gu et al. 05, Wagner and Plattner 05, Xu et al. 05] to detect sudden changes in the network behavior and as an indicator of anomalous events. The rationale is well explained elsewhere, chiefly in Section 2 of [Wagner and Plattner 05], but the summary is that there are intimate connections between the "randomness" of traffic sequences (formalized as the entropy) and the propagation of malicious events such as worms and various attacks. The current research in this area [Wagner and Plattner 05, Gu et al. 05, Xu et al. 05] relies on full-space algorithms for entropy calculation; this is a serious bottleneck in high-speed routers where high-speed memory is at a premium. Indeed, this is the bottleneck that motivated data-stream algorithms and their applications to IP network analysis [Estan and Varghese 03, Muthukrishnan 06]. Our small-space algorithms can immediately make entropy estimation at line speed practical on high-speed routers. Our algorithms are quite simple and rely on sampling and sketching that are already part of operational traffic analysis systems such as Gigascope at AT&T [Cranor et al. 03, Cormode et al. 04, Johnson et al. 05]. Thus, we expect our algorithms to prove useful in real IP network traffic analysis systems.

1.2. Related Work and Comparison to Our Work

To the best of our knowledge, our upper and lower bound results for the entropy norm are the first of their kind. Recently, and independently of our work, Guha, McGregor, and Venkatasubramanian [Guha et al. 06] considered approximation algorithms for the entropy of a given distribution under various models, including the data stream model in which we work. They obtain a $\left(\frac{e}{e-1}+\varepsilon\right)$-approximation for the entropy H of an input stream provided that H is at least a sufficiently large constant, using space $\widetilde{O}(1/(\varepsilon^2 H))$, where the $\widetilde{O}$-notation hides factors polylogarithmic in m and n. They observe that a limitation of their technique is that "there will always be a constant bias between the entropy and [their] estimate" and that "[their] particular method alone is unlikely to yield better results." Our work, in particular, shows that H can be $(1+\varepsilon)$-approximated in $\widetilde{O}(1/\varepsilon^2)$ space for $H \geq 1$ (see Remark 3.2). More importantly, our work shows that the most challenging inputs for the entropy estimation problem are those that lead to $H < 1$, and we obtain efficient sublinear-space approximation algorithms that handle these cases as well. Our space bounds are independent of H.

Guha et al. [Guha et al. 06] also give a *two-pass*, $(1+\varepsilon)$-approximation algorithm for entropy, using $\widetilde{O}(1/(\varepsilon^2 H))$ space. In our work, we do the same using only $\widetilde{O}(1/\varepsilon^2)$ space (see Theorem 3.5). Finally, Guha et al. consider the entropy estimation problem in the *random streams model*, where it is assumed that the items in the input stream are presented in a uniform random order. Under this assumption, they obtain a $(1+\varepsilon)$-approximation using $\widetilde{O}(1/\varepsilon^2)$ space. We study adversarial data stream inputs only.

The algorithms behind our Theorems 3.1 and 3.5 are simpler and easier to analyze than earlier work.

2. Estimating the Entropy Norm

In this section we present a polylogarithmic-space, $(1+\varepsilon)$-approximation algorithm for entropy norm that assumes that the norm is sufficiently large, and we prove a matching lower bound if the norm is in fact not as large.

2.1. Upper Bound

Our algorithm is inspired by the work of Alon et al. [Alon et al. 96]. Their first algorithm, for the frequency moments F_k, has the following nice structure to it (some of the terminology is ours). A subroutine computes a *basic estimator*, which is a random variable X whose mean is exactly the quantity we seek and whose variance is small. The algorithm itself uses this subroutine to maintain $s_1 s_2$ independent basic estimators $\{X_{ij} : 1 \leq i \leq s_1, 1 \leq j \leq s_2\}$, where each X_{ij} is distributed identically to X. It then outputs a *final estimator* Y defined by

$$Y \;:=\; \operatorname*{median}_{1\leq j\leq s_2} \left(\frac{1}{s_1} \sum_{i=1}^{s_1} X_{ij} \right).$$

The following lemma, implicit in [Alon et al. 96], gives a guarantee on the quality of this final estimator.

Lemma 2.1. *Let $\mu := \mathrm{E}[X]$. For any $\varepsilon, \delta \in (0,1)$, if $s_1 \geq 8\,\mathrm{Var}[X]/(\varepsilon^2\mu^2)$ and $s_2 = 4\lg(1/\delta)$, then the above final estimator deviates from μ by no more than*

$\varepsilon\mu$ with probability at least $1-\delta$. The above algorithm can be implemented to use space $O(S\log(1/\delta)\operatorname{Var}[X]/(\varepsilon^2\mu^2))$, provided that the basic estimator can be computed using space at most S.

Proof. The claim about the space usage is immediate from the structure of the algorithm. Let $Y_j = \frac{1}{s_1}\sum_{i=1}^{s_1} X_{ij}$. Then, $\mathrm{E}[Y_j] = \mu$ and $\operatorname{Var}[Y_j] = \operatorname{Var}[X]/s_1 \le \varepsilon^2\mu^2/8$. Applying Chebyshev's Inequality gives us

$$\Pr[|Y_j - \mu| \ge \varepsilon\mu] \;\le\; 1/8\,.$$

Now, if fewer than $s_2/2$ of the Y_j deviate by as much as $\varepsilon\mu$ from μ, then Y must be within $\varepsilon\mu$ of μ. So we upper bound the probability that this does not happen. Define s_2 indicator random variables I_j, where $I_j = 1$ iff $|Y_j - \mu| \ge \varepsilon\mu$, and let $W = \sum_{j=1}^{s_2} I_j$. Then, $\mathrm{E}[W] \le s_2/8$. Applying a standard Chernoff bound [Motwani and Raghavan 95, Theorem 4.1] gives

$$\Pr\Big[|Y-\mu| \ge \varepsilon\mu\Big] \;\le\; \Pr\Big[W \ge \frac{s_2}{2}\Big] \;\le\; \left(\frac{e^3}{4^4}\right)^{s_2/8} \;=\; \left(\frac{e^3}{4^4}\right)^{\frac{1}{2}\lg(1/\delta)} \;\le\; \delta\,,$$

which completes the proof. □

We use the following subroutine to compute a basic estimator X for the entropy norm F_H.

Algorithm 1. (Basic Estimator for the Entropy Norm.)

```
Input stream: $A = \langle a_1, a_2, \ldots, a_m\rangle$, where each $a_j \in \{1,\ldots,n\}$.
Choose $p$ uniformly at random from $\{1,\ldots,m\}$.
Compute $r = |\{q : a_q = a_p, p \le q \le m\}|$. Note that $r \ge 1$.
Output $X = m\big(r\lg r - (r-1)\lg(r-1)\big)$, with the convention that $0\lg 0 = 0$.
```

Our algorithm for estimating the entropy norm outputs a final estimator based on the basic estimator described in Algorithm 1. This gives us the following theorem.

Theorem 2.2. *For any $\Delta > 0$, if $F_H \ge m/\Delta$, the above one-pass algorithm can be implemented so that its output deviates from F_H by no more than εF_H with probability at least $1-\delta$, and so that it uses space*

$$O\left(\frac{\log(1/\delta)}{\varepsilon^2}\log m(\log m + \log n)\Delta\right).$$

In particular, taking Δ to be a constant, we have a polylogarithmic-space algorithm that works on streams whose F_H is not "too small."

Proof. We first check that the expected value of X is indeed the desired quantity:

$$\begin{aligned} \mathrm{E}[X] &= \frac{m}{m}\sum_{i=1}^{n}\sum_{r=1}^{m_i}\big(r\lg r-(r-1)\lg(r-1)\big) \\ &= \sum_{i=1}^{n}(m_i\lg m_i - 0\lg 0) \;=\; F_H\,. \end{aligned}$$

The approximation guarantee of the algorithm now follows from Lemma 2.1. To bound the space usage, we must bound the variance $\mathrm{Var}[X]$ and for this we bound $\mathrm{E}[X^2]$. Let $f(r) := r\lg r$, with $f(0) := 0$, so that X can be expressed as $X = m(f(r)-f(r-1))$. Then,

$$\begin{aligned} \mathrm{E}[X^2] &= m\sum_{i=1}^{n}\sum_{r=1}^{m_i}\big(f(r)-f(r-1)\big)^2 \\ &\le m\cdot\max_{1\le r\le m}\big(f(r)-f(r-1)\big)\cdot\sum_{i=1}^{n}\sum_{r=1}^{m_i}\big(f(r)-f(r-1)\big) \\ &\le m\cdot\sup\{f'(x) : x\in(0,m]\}\cdot F_H \qquad (2.1) \\ &= (\lg e+\lg m)\,mF_H \qquad (2.2) \\ &\le (\lg e+\lg m)\,\Delta F_H^2\,, \end{aligned}$$

where (2.1) follows from the Mean Value Theorem.

Thus, $\mathrm{Var}[X]/E[X]^2 = O(\Delta\lg m)$. Moreover, the basic estimator can be implemented using space $O(\log m+\log n)$: $O(\log m)$ to count m and r, and $O(\log n)$ to store the value of a_p. Plugging these bounds into Lemma 2.1 yields the claimed upper bound on the space of our algorithm. □

Let F_0 denote the number of distinct items in the input stream (this notation deliberately coincides with that for frequency moments). Let $f(x) := x\lg x$ as used in the proof of Theorem 2.2. Observe that f is convex on $(0,\infty)$, whence, via Jensen's inequality, we obtain

$$F_H \;=\; \frac{F_0}{F_0}\sum_{i=1}^{n}f(m_i) \;\ge\; F_0 f\left(\frac{1}{F_0}\sum_{i=1}^{n}m_i\right) \;=\; m\lg\frac{m}{F_0}\,. \qquad (2.3)$$

Thus, if the input stream satisfies $m\ge 2F_0$ (or the simpler, but stronger, condition $m\ge 2n$), then we have $F_H\ge m$. As a direct corollary of Theorem 2.2 (for $\Delta = 1$), we obtain a $(1+\varepsilon)$-approximation algorithm for the entropy norm in space $O((\log(1/\delta)/\varepsilon^2)\log m(\log m+\log n))$. However, we can do slightly better.

Theorem 2.3. *If $m \geq 2F_0$, then the above one-pass, $(1+\varepsilon)$-approximation algorithm can be implemented in space*

$$O\left(\frac{\log(1/\delta)}{\varepsilon^2}\log m \log n\right)$$

without a priori knowledge of the stream length m.

Proof. We follow the proof of Theorem 2.2 up to the bound (2.2) to obtain $\mathrm{Var}[X] \leq (2\lg m)mF_H$, for m large enough. We now make the following claim:

$$\frac{\lg m}{\lg(m/F_0)} \;\leq\; 2\max\{\lg F_0, 1\}\,. \tag{2.4}$$

Assuming the truth of this claim and using (2.3), we obtain

$$\mathrm{Var}[X] \;\leq\; (2\lg m)mF_H \;\leq\; \frac{2\lg m}{\lg(m/F_0)}F_H^2 \;\leq\; 4\max\{\lg F_0, 1\}F_H^2 \;\leq\; (4\lg n)F_H^2\,.$$

Plugging this into Lemma 2.1 and proceeding as before, we obtain the desired space upper bound. Note that we no longer need to know m before starting the algorithm, because the number of basic estimators used by the algorithm is now independent of m. Although maintaining each basic estimator seems, at first, to require prior knowledge of m because a_p needs to be chosen uniformly at random from the input stream, a careful implementation can avoid this, as shown by Alon et al. [Alon et al. 96]. Specifically, as we read the stream, we maintain a "current" choice for a_p from the items already seen and, upon the arrival of the jth item a_j, replace our choice by a_j with probability $1/j$. It follows easily by induction that after m items have been seen, our choice is uniformly distributed over the m items.

We turn to proving our claim (2.4). We will need the assumption $m \geq 2F_0$. If $m \leq F_0^2$, then

$$\lg m \;\leq\; 2\lg F_0 \;=\; 2\lg F_0 \lg(2F_0/F_0) \;\leq\; 2\lg F_0 \lg(m/F_0)$$

and we are done. On the other hand, if $m \geq F_0^2$, then $F_0 \leq m^{1/2}$ so that

$$\lg(m/F_0) \;\geq\; \lg m - (1/2)\lg m \;=\; (1/2)\lg m$$

and we are done as well. □

Remark 2.4. Theorem 2.2 generalizes to estimating quantities of the form $\hat{\mu} = \sum_{i=1}^n \hat{f}(m_i)$, for any monotone increasing (on integer values), differentiable function $\hat{f}$ that satisfies $\hat{f}(0) = 0$. Assuming $\hat{\mu} \geq m/\Delta$, it gives us a one-pass, $(1+\varepsilon)$-approximation algorithm that uses $\widetilde{O}(\hat{f}'(m)\Delta)$ space. For instance, this space usage is polylogarithmic in m if $\hat{f}(x) = x\,\mathrm{polylog}(x)$.

2.2. Lower Bound

The following lower bound shows that the algorithm of Theorem 2.2 is optimal, up to factors polylogarithmic in m and n.

Theorem 2.5. *Suppose that Δ and c are integers with $4 \leq \Delta \leq o(m)$ and $0 \leq c \leq m/\Delta$. On input streams of size at most m, a randomized algorithm able to distinguish between $F_H \leq 2c$ and $F_H \geq c + 2m/\Delta$ must use space at least $\Omega(\Delta)$. In particular, the upper bound in Theorem 2.2 is tight in its dependence on Δ.*

Proof. We present a reduction from the classic problem of (two-party) set disjointness in communication complexity. For more on communication complexity and the set disjointness problem, we refer the reader to the textbook by Kushilevitz and Nisan [Kushilevitz and Nisan 97].

Suppose that Alice has a subset X and Bob a subset Y of $\{1, 2, \ldots, \Delta - 1\}$, such that X and Y either are disjoint or intersect at exactly one point. Let us define the mapping

$$\phi : x \longmapsto \left\{ \frac{(m-2c)x}{\Delta} + i : i \in \mathbb{Z},\, 0 \leq i < \frac{m-2c}{\Delta} \right\}.$$

Alice creates a stream A by listing all elements in $\bigcup_{x \in X} \phi(x)$ and concatenating the c special elements $\Delta + 1, \ldots, \Delta + c$. Similarly, Bob creates a stream B by listing all elements in $\bigcup_{y \in Y} \phi(y)$ and concatenating the same c special elements $\Delta + 1, \ldots, \Delta + c$. Now, Alice can process her stream (with the hypothetical entropy norm estimation algorithm) and send over her memory contents to Bob, who can then finish the processing. Note that the length of the combined stream $A \circ B$ is at most

$$2c + |X \cup Y| \cdot ((m - 2c)/\Delta) \leq m.$$

We now show that, based on the output of the algorithm, Alice and Bob can tell whether or not X and Y intersect. Since the set disjointness problem has communication complexity $\Omega(\Delta)$, even when $|X \cap Y|$ is known to be either 0 or 1, we get the desired space lower bound.

Suppose that X and Y are disjoint. Then, the items in $A \circ B$ are all distinct except for the c special elements, which appear twice each. So $F_H(A \circ B) = c \cdot (2 \lg 2) = 2c$. Now, suppose that $X \cap Y = \{z\}$. Then, the items in $A \circ B$ are all distinct except for the $(m - 2c)/\Delta$ elements in $\phi(z)$ and the c special elements, each of which appears twice. So

$$F_H(A \circ B) = 2(c + (m - 2c)/\Delta) \geq c + 2m/\Delta,$$

since $\Delta \geq 4$. □

Remark 2.6. Notice that Theorem 2.5 rules out even a polylogarithmic-space, *constant-factor* approximation to F_H that can work on streams with "small" F_H. This can be seen by setting $\Delta = m^\gamma$ for some constant $\gamma > 0$.

3. Estimating the Empirical Entropy

We now turn to the estimation of the empirical entropy H of a data stream, defined as in Equation (1.1): $H = \sum_{i=1}^{n}(m_i/m)\lg(m/m_i)$. Although H can be computed exactly from F_H, as shown in (1.1), a $(1+\varepsilon)$-approximation of F_H can yield a poor estimate of H when H is small (sublinear in its maximum value, $\lg m$). We therefore present a different sublinear-space, one-pass algorithm that directly computes entropy.

Our data structure takes a user parameter $\alpha > 0$ and consists of three components. The first (A1) is a sketch in the manner of Section 2, with basic estimator

$$X \;=\; m\left(\frac{r}{m}\lg\frac{m}{r} - \frac{r-1}{m}\lg\frac{m}{r-1}\right), \tag{3.1}$$

and a final estimator derived from this basic estimator using $s_1 = (8/\varepsilon^2)m^{2\alpha}\lg^2 m$ and $s_2 = 4\lg(1/\delta)$. The second component (A2) is an array of $m^{2\alpha}$ counters (each counting from 1 to m) used to keep exact counts of the first $m^{2\alpha}$ distinct items seen in the input stream. The third component (A3) is a *Count-Min Sketch*, as described by Cormode and Muthukrishnan [Cormode and Muthukrishnan 05], which we use to estimate k, defined to be the number of items in the stream that are *different* from the most frequent item; i.e.,

$$k \;:=\; m - \max\{m_i : 1 \le i \le n\}. \tag{3.2}$$

The Count-Min Sketch answers point queries, which we exploit to keep track of the most frequent item, x, as follows: upon arrival of an item, i, in the stream, we query m_i and replace x by i if $\hat{m}_i > \hat{m}_x$, where $\hat{m}_i$ and $\hat{m}_x$ are the approximations given by the Sketch for m_i and m_x, respectively. Our estimate of k is then just $\hat{k} = m - \hat{m}_x$.

The algorithm itself works as follows. Recall that F_0 denotes the number of distinct items in the stream.

Algorithm 2. (Estimation of Entropy.)

```
Input  : data stream of length $m$
Output: an estimate of the empirical entropy of the stream
Maintain A1, A2, and A3 as described above. When queried (or at end of input):
if $F_0 \le m^{2\alpha}$ then return exact $H$ from A2.
else
    let $\hat{k}$ = estimate of $k$ from A3.
    if $\hat{k} \ge (1-\varepsilon)m^{1-\alpha}$ then return final estimator, $Y$, of A1.
    else return $(\hat{k} \lg m)/m$.
end
```

Theorem 3.1. *Algorithm 2 uses*

$$O\left(\frac{\log(1/\delta)}{\varepsilon^2} m^{2\alpha} \log^2 m(\log m + \log n)\right)$$

space and outputs a random variable Z that satisfies the following properties:

1. *If $k \le m^{2\alpha} - 1$, then $Z = H$.*
2. *If $k \ge m^{1-\alpha}$, then $\Pr\left[|Z - H| \ge \varepsilon H\right] \le 2\delta$.*
3. *Otherwise (i.e., if $m^{2\alpha} \le k < m^{1-\alpha}$), Z is a $(1/\alpha)$-approximation of H.*

Remark 3.2. Under the assumption $H \ge 1$, an algorithm that uses only the basic estimator in A1 and sets $s_1 = (8/\varepsilon^2)\lg^2 m$ suffices to give a $(1+\varepsilon)$-approximation in $\widetilde{O}(1/\varepsilon^2)$ space.

To prove this theorem, we need the following technical lemma.

Lemma 3.3. *Given that the most frequent item in the input stream A has count $m - k$, the minimum entropy $H_{\min}$ is achieved when all the remaining k items are identical, and the maximum $H_{\max}$ is achieved when they are all distinct. Therefore,*

$$\begin{aligned} H_{\min} &= \frac{m-k}{m} \lg \frac{m}{m-k} + \frac{k}{m} \lg \frac{m}{k}\,, \quad \textit{and} \\ H_{\max} &= \frac{m-k}{m} \lg \frac{m}{m-k} + \frac{k}{m} \lg m\,. \end{aligned}$$

Proof. Consider a minimum-entropy stream $A_{\min}$, and suppose that, apart from its most frequent item, it has at least two other items with positive count. Without loss of generality, let $m_1 = m - k$ and $m_2, m_3 \geq 1$. Modify $A_{\min}$ to A' by letting $m_2' = m_2 + m_3$ and $m_3' = 0$ and keeping all other counts the same. Then,

$$\begin{aligned} H(A') - H(A_{\min}) &= \left(\lg m - \frac{F_H(A')}{m}\right) - \left(\lg m - \frac{F_H(A_{\min})}{m}\right) \\ &= \frac{F_H(A_{\min}) - F_H(A')}{m} \\ &= m_2 \lg m_2 + m_3 \lg m_3 - (m_2 + m_3)\lg(m_2 + m_3) \\ &< 0\,, \end{aligned}$$

since $x \lg x$ is convex and monotone increasing (on integer values), giving us a contradiction. The proof for the maximum-entropy distribution is similar. □

Proof of Theorem 3.1. The space bound is clear from the specifications of A1, A2, and A3 and Lemma 2.1. Note, in particular, that a Count-Min Sketch requires only $O(\varepsilon^{-1}\log(m/\delta))$ space, which is easily absorbed into the bound of the theorem. We now prove the three claimed properties of the output, Z.

Property 1. This follows directly from the fact that $F_0 \leq k + 1$.

Property 2. The Count-Min Sketch guarantees that $\hat{k} \leq k$ and, with probability at least $1 - \delta$, $\hat{k} \geq (1 - \varepsilon)k$. The condition in Property 2 therefore implies that $\hat{k} \geq (1 - \varepsilon)m^{1-\alpha}$, that is, $Z = Y$, with probability at least $1 - \delta$. It remains to show that Y is a $(1 + \varepsilon)$-approximation of H with probability at least $1 - \delta$.

Consider Equation (3.1), and note that for any r, $|X| \leq \lg m$. Thus, if $\mathrm{E}[X] = H \geq 1$, then $\mathrm{Var}[X]/\mathrm{E}[X]^2 \leq \mathrm{E}[X^2] \leq \lg^2 m$ and our choice of s_1 is sufficiently large to give us the desired $(1+\varepsilon)$-approximation, by Lemma 2.1.[2] On the other hand, if $H < 1$, then $k < m/2$, by a simple argument similar to the proof of Lemma 3.3. Using the expression for $H_{\min}$ from Lemma 3.3, we then have

$$H_{\min} \;=\; \lg\frac{m}{m-k} + \frac{k}{m}\lg\frac{m-k}{k} \;\geq\; -\lg\left(1 - \frac{k}{m}\right) \;\geq\; \frac{k}{m} \;\geq\; m^{-\alpha}\,,$$

which gives us $\mathrm{Var}[X]/\mathrm{E}[X]^2 \leq \mathrm{E}[X^2]/m^{-2\alpha} \leq (\lg^2 m)m^{2\alpha}$. Again, plugging this and our choice of s_1 into Lemma 2.1 gives us the desired $(1 + \varepsilon)$-approximation.

Property 3. By assumption, $m^{2\alpha} \leq k < m^{1-\alpha}$. We shall only need the upper bound on k and not the lower bound. If $\hat{k} \geq (1 - \varepsilon)m^{1-\alpha}$, then $Z = Y$ and the

[2] This observation, that $H \geq 1 \Longrightarrow \mathrm{Var}[X] \leq \lg^2 m$, proves the statement in Remark 3.2.

analysis proceeds as for Property 2. Otherwise, $Z = (\hat{k} \lg m)/m \leq (k \lg m)/m$. This time, again by Lemma 3.3, we have

$$H_{\min} \;\geq\; \frac{k}{m} \lg \frac{m}{k} \;\geq\; \frac{k}{m} \lg\left(m^{\alpha}\right) \;=\; \frac{\alpha k}{m} \lg m$$

and

$$\begin{aligned} H_{\max} &= \frac{m-k}{m} \lg \frac{m}{m-k} + \frac{k}{m} \lg m \\ &= \lg \frac{m}{m-k} + \frac{k}{m} \lg(m-k) \\ &\leq \frac{k}{m} \lg m + O\left(\frac{k}{m}\right), \end{aligned}$$

which, for large m, implies $H - o(H) \leq Z \leq H/\alpha$ and gives us Property 3. □

Corollary 3.4. *For $\alpha = 1/3$, the third case (Property 3) never occurs, so we have a $\widetilde{O}(m^{2/3})$-space, $(1+\varepsilon)$-approximation algorithm.*

The ideas involved in the proof of Theorem 3.1 can be used to yield a very efficient *two-pass* algorithm for estimating H, as follows. In the first pass, we use an estimator as in Remark 3.2, which gives us a good estimate provided that the stream does not have a *majority item*, i.e., $k \geq m/2$ (whence $H \geq 1$). During the second pass we handle the case $k < m/2$ by maintaining a very similar estimator but only working on the substream consisting of all items except the majority item. To make this work, we need to dovetail our estimator computations with a standard two-pass algorithm for finding a majority item, such as the one by Boyer and Moore [Boyer and Moore 82].

Theorem 3.5. *Algorithm 3 uses space $O(\varepsilon^{-2} \log(1/\delta) \log^2 m)$, and its output differs from H by more than εH with probability at most δ.*

Proof. The space bound is immediate from Lemma 2.1. If the input stream has no majority item, we must have $H \geq 1$. The algorithm will output Z, which, as mentioned in Remark 3.2, gives a $(1+\varepsilon)$-approximation to H.

Algorithm 3. (Estimation of Entropy Using Two Passes.)

```
Input  : data stream $A$ of length $m$, two passes allowed
Output: an estimate of the empirical entropy of the stream
Make one pass over $A$, computing two things:
    a majority candidate, $x$, for $A$ as in Boyer and Moore [Boyer and
    Moore 82].
    a final estimator $Z$ from the basic estimator given by
    Equation (3.1), $s_1 = (8/\varepsilon^2)\lg^2 m$ and $s_2 = 4\lg(1/\delta)$.
Make a second pass over $A$, computing two things:
    the frequency $m_x$ of the majority candidate in $A$.
    a final estimator $Y$ that is identical to $Z$ except that it is
    produced by sampling from the substream $A \setminus \{x\}$.
if $m_x \leq m/2$ then  return $Z$.
else
    let $k = m - m_x$.
    return $\frac{kY}{m} + \frac{m-k}{m} \lg \frac{m}{m-k}$.
end
```

We now consider the other case, i.e., $k < m/2$. Assume w.l.o.g. that item 1 is the majority item in the input stream. Then,

$$\begin{aligned} \mathrm{E}[Y] &= \frac{m}{k} \sum_{i=2}^{n} \frac{m_i}{m} \lg \frac{m}{m_i} \\ &\geq \frac{m}{k} \left(\frac{k}{m} \lg \frac{m}{k} \right) \\ &= \lg(m/k) \\ &\geq 1\,, \end{aligned}$$

where the first inequality follows in a manner similar to the proof of Lemma 3.3, using $\sum_{i=2}^{n} m_i = k$.

Also, $|Y| \leq \lg m$, whence $\mathrm{Var}[Y]/\,\mathrm{E}[Y]^2 \leq \mathrm{E}[Y^2] \leq \lg^2 m$. By Lemma 2.1, Y is ε-close to $\mathrm{E}[Y]$ with probability at least $1 - \delta$. From the definition of H and Y, we see that

$$H = \frac{k}{m}\,\mathrm{E}[Y] + \frac{m-k}{m} \lg \frac{m}{m-k}\,,$$

whence H is ε-close to the output of the algorithm with probability at least $1 - \delta$. □

4. Conclusions

Entropy and entropy norms are natural measures with direct applications in IP network traffic analysis for which one-pass, streaming algorithms are needed. We have presented one-pass, sublinear-space algorithms for approximating the entropy norms as well as the empirical entropy. We have also presented a two-pass algorithm for empirical entropy that has a stronger approximation guarantee and space bound. We believe that our algorithms will be of interest in the practice of data stream systems. It will be of interest to study these problems on streams in the presence of inserts and deletes.

Very recently, we have learned of a work in progress [Indyk 05] that may lead to a one-pass, polylogarithmic-space algorithm for approximating H to within a $(1+\varepsilon)$-factor.

Acknowledgments. The first author was supported by an NSF CAREER award and Dartmouth College startup funds. Work by the second author was partly done while visiting DIMACS in the REU program, supported by NSF ITR 0220280, DMS 0354600, and a Dean of Faculty Fellowship from Dartmouth College. The third author was supported by NSF ITR 0220280 and DMS 0354600.

References

[Alon et al. 96] N. Alon, Y. Matias and M. Szegedy. “The Space Complexity of Approximating the Frequency Moments.” In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, pp. 20–29. New York: ACM Press, 1996.

[Babcock et al. 06] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. “Models and Issues in Data Stream Systems.” In *Proceedings of the Twenty-First ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pp. 1–16. New York: ACM Press, 2002.

[Boyer and Moore 82] R. Boyer and J. Moore. “MJRTY—A Fast Majority Vote Algorithm.” Technical Report ICSCA-CMP-32, Institute for Computing Science and Computer Applications, University of Texas at Austin, 1982.

[Coppersmith and Kumar 04] D. Coppersmith and R. Kumar. “An Improved Data Stream Algorithm for Frequency Moments.” In *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 151–156. Philadelphia: SIAM, 2004.

[Cormode et al. 04] Graham Cormode, Theodore Johnson, Flip Korn, S. Muthukrishnan, Oliver Spatscheck, and Divesh Srivastava. “Holistic UDAFs at Streaming Speeds.” In *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data*, pp. 35–46. New York: ACM Press, 2004.

[Cormode and Muthukrishnan 05] G. Cormode and S. Muthukrishnan. "An Improved Data Stream Summary: The Count-Min Sketch and Its Applications." *J. Algorithms* 55:1 (2005), 58–75.

[Cranor et al. 03] Charles D. Cranor, Theodore Johnson, Oliver Spatscheck, and Vladislav Shkapenyuk. "The Gigascope Stream Database." *IEEE Data Eng. Bull.* 26:1 (2003), 27–32.

[Estan and Varghese 03] C. Estan and G. Varghese. "New Directions in Traffic Measurement and Accounting: Focusing on the Elephants, Ignoring the Mice." *ACM Trans. Comput. Syst.* 21:3 (2003), 270–313.

[Gu et al. 05] Y. Gu, A. McCallum and D. Towsley. "Detecting Anomalies in Network Traffic Using Maximum Entropy Estimation." In *Proceedings of the Internet Measurement Conference*. Available from the World Wide Web (http://www.usenix.org/events/imc05/tech/qu.html), 2005.

[Guha et al. 06] S. Guha, A. McGregor, and S. Venkatasubramanian. "Streaming and Sublinear Approximation of Entropy and Information Distances." In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 733–742. New York: ACM Press, 2006.

[Indyk 05] P. Indyk. Personal e-mail communication. September 2005.

[Indyk and Woodruff 05] P. Indyk and D. Woodruff. "Optimal Approximations of the Frequency Moments of Data Streams." In *Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing*, pp. 202–208. New York: ACM Press, 2005.

[Johnson et al. 05] T. Johnson, S. Muthukrishnan, and I. Rozenbaum. "Sampling Algorithms in a Stream Operator." In *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*, pp. 1–12. New York: ACM Press, 2005.

[Kushilevitz and Nisan 97] E. Kushilevitz and N. Nisan. *Communication Complexity*. Cambridge, UK: Cambridge University Press, 1997.

[Motwani and Raghavan 95] R. Motwani and P. Raghavan. *Randomized Algorithms*. New York: Cambridge University Press, 1995.

[Muthukrishnan 06] S. Muthukrishnan. "Data Streams: Algorithms and Applications." Manuscript. Available from World Wide Web (http://www.cs.rutgers.edu/~muthu/stream-1-1.ps), 2006.

[Wagner and Plattner 05] A. Wagner and B. Plattner. "Entropy Based Worm and Anomaly Detection in Fast IP Networks." In *14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise*, pp. 172–177. Los Alamitos, CA: IEEE Press, 2005.

[Xu et al. 05] K. Xu, Z. Zhang, and S. Bhattacharya. "Profiling Internet Backbone Traffic: Behavior Models and Applications." In *Proceedings of the 2005 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pp. 169–180. New York: ACM Press, 2005.

Amit Chakrabarti, 6211 Sudikoff Lab., Dartmouth College, Hanover, NH 03755
(ac@cs.dartmouth.edu)

Khanh Do Ba, 6211 Sudikoff Lab., Dartmouth College, Hanover, NH 03755
(khanh.do.ba.06@alum.dartmouth.org)

S. Muthukrishnan, Department of Computer Science, Rutgers, The State University
of New Jersey, 110 Frelinghuysen Road, Piscataway, NJ 08854-8019
(muthu@cs.rutgers.edu)

Received September 27, 2005; accepted November 7, 2005.