A Simple and Efficient Estimation Method for Stream Expression Cardinalities

Aiyou Chen, Jin Cao and Tian Bu
Bell Labs, Alcatel-Lucent
(aychen,jincao,tbu)@research.bell-labs.com

ABSTRACT

Estimating the cardinality (i.e. number of distinct elements) of an arbitrary set expression defined over multiple distributed streams is one of the most fundamental queries of interest. Earlier methods based on probabilistic sketches have focused mostly on the sketching algorithms. However, the estimators do not fully utilize the information in the sketches and thus are not statistically efficient. In this paper, we develop a novel statistical model and an efficient yet simple estimator for the cardinalities based on a continuous variant of the well known Flajolet-Martin sketches. Specifically, we show that, for two streams, our estimator has almost the same statistical efficiency as the Maximum Likelihood Estimator (MLE), which is known to be optimal in the sense of Cramer-Rao lower bounds under regular conditions. Moreover, as the number of streams gets larger, our estimator is still computationally simple, but the MLE becomes intractable due to the complexity of the likelihood. Let N be the cardinality of the union of all streams, and $|\mathcal{S}|$ be the cardinality of a set expression $\mathcal{S}$ to be estimated. For a given relative standard error δ, the memory requirement of our estimator is $O(\delta^{-2}|\mathcal{S}|^{-1}N\log\log N)$, which is superior to state-of-the-art algorithms, especially for large N and small $\frac{|\mathcal{S}|}{N}$ where the estimation is most challenging.

1. INTRODUCTION

Massive and distributed data streams are increasingly prevalent in many modern applications. In a backbone IP network composed of hundreds or even thousands of nodes, packets arrive and depart at the nodes at very high speed. In a web content delivery system composed of many servers (such as Akamai), the user requests to websites are distributed among the servers based on the user location and server loads. Other application domains that give rise to these massive and distributed streams include financial applications and sensor networks.

Due to their massive and distributed nature, query answering for these data streams poses a unique challenge. Often, exact query computation is infeasible due to the memory requirement and the communication overhead. Thus approximate query answering that can provide probabilistic guarantees becomes the only viable option. One of the most fundamental queries of interest is to estimate the cardinality (i.e., number of distinct elements) of an arbitrary set expression defined over multiple distributed streams. (Note: the notion of set in this paper means a stream of elements and it allows multiple appearance of the same elements.) For instance, in the context of IP network management, the number of distinct flows in a network sharing the same characteristics is of high interest to network operators, where a packet flow can be defined as a sequence of packets that have the same 5-tuple, IP addresses/ports of the two communicating peers and the protocol. Moreover, the flow ID of a packet can be derived from the 5-tuple. The number of distinct common flows between a node pair i and j, which is a special case of the *traffic matrix*, can be formulated as the stream cardinality of $\mathcal{T}_i \cap \mathcal{T}_j$, where $\mathcal{T}_i$ and $\mathcal{T}_j$ are the streams of packet flow IDs seen at node i, j respectively. The traffic matrix is used by the network operators for network provisioning and optimization. Another example is the total number of distinct flows to a particular destination, i.e, $\cup_i \mathcal{T}_i$, where $\mathcal{T}_i$ is the stream of packet flow IDs to the same destination seen at node i. A significant increase of the cardinality of $\cup_i \mathcal{T}_i$ may indicates an underlying network anomaly such as a Denial of Service (DoS) attack.

In general, let $\mathcal{T}_j$, $1 \le j \le J$, denote the distributed streams under consideration. Let $\cup, \cap, \backslash$ represent the set union, intersection and difference, respectively. We use $|\cdot|$ to denote the cardinality of a set. For a stream expression that involves an arbitrary combination of unions, intersections, and/or differences of $\mathcal{T}_j, 1 \le j \le J$, (e.g., $(\mathcal{T}_1 \cup \mathcal{T}_2) \cap \mathcal{T}_3 \backslash \mathcal{T}_4$), our objective is to estimate its cardinality and provide probabilistic guarantees of the estimation accuracy whereas minimize the computational overhead and the memory usage.

There have been a few proposals in the literature for an approximate answer of the cardinality queries of stream expressions. These methods seek to develop estimators based on compact sketches of the distributed streams. Such sketches can be sampling based [15], or hash based probabilistic sketches [6, 19, 13, 12]. Comparison between some of these methods has been made by [19], which showed that the probabilistic approaches have a performance advantage over the sampling based approaches.

The works that are more relevant to ours are those based on probabilistic sketches. For a single stream cardinality counting, Flajolet and Martin [11] proposed an estimator

VLDB '07, September 23-28, 2007, Vienna, Austria.

using a hash-based synopsis data structure with $O(\log N)$ space where N is the true cardinality. To date, the Flajolet-Martin (FM) technique remains one of the most effective approaches to the single stream cardinality counting problem, and has seen several applications [7, 14]. In [8], the FM method is further improved by reducing the space requirement to only $O(\log\log N)$; other enhancements can also be found by [1, 3, 12].

To extend the one-stream cardinality counting algorithms to set expressions over multiple streams, a straightforward solution is to re-express the cardinality using the cardinalities of stream unions, and count the cardinality of each set union, where the set union cardinalities can be estimated easily. This is the approach adopted by [6, 19] using an improved version of the FM algorithm [8] and the bitmap algorithms [9] which solves the one-stream cardinality problem. More recently, [13] proposed a novel 2-level hash algorithm for estimating stream expression cardinalities that does not simply use one-stream cardinality algorithms. This is further improved in [12] by reducing the per bucket storage. Their approach can be applied to general update streams instead of the insert only streams for most of other algorithms.

These probabilistic sketch based solutions largely focus on deriving novel sketching algorithms. But little attention has been paid to develop *statistically efficient* estimators. As a result, their estimators do not fully utilize the information contained in the sketches. For example, the estimators proposed by [19, 6] based on one-stream cardinality algorithms do not fully explore the correlation information between the sketches of different streams and thus are quite inefficient. We defer the detailed theoretical justifications to Section 5.

Popularized by the Fisher information theory and celebrated Cramer-Rao lower bound [4], statistical efficiency is one of the most fundamental concepts in statistical inference. It measures the quality of an estimator by comparing its variance to the Cramer-Rao lower bound that is the lower bound of variance for unbiased estimators, achievable under regular conditions. It has been shown that the Maximum-Likelihood Estimator (MLE) is *asymptotically efficient* as it can achieve the Cramer-Rao lower bound with increasing sample sizes. However, in many cases, MLE is computationally expensive or even intractable if the likelihood function is complex. Such cases call for alternative methods for deriving statistically efficient estimators. The focus of this paper is to bring formal statistical inference techniques to the forefront of the stream expression cardinality problem, and develop simple yet statistically efficient estimators that are superior to existing ones.

1.1 Our Contributions

Let $\mathcal{S}$ be a stream expression under study, N be the cardinality of the union of all streams, and $p = |\mathcal{S}|/N$ be the proportion of $|\mathcal{S}|$. Let δ be a specified value of the relative standard error that is desirable. Our contributions can be summarized as follows.

1. Using a continuous variant of the FM sketches, we develop a statistically efficient yet simple *proportional-union* estimator for the stream expression cardinality problem. We show that the memory requirement of our estimator is $O(\delta^{-2}p^{-1}\log\log N)$, where the $\log\log N$ order is achieved by discretizing a continuous version of the FM sketches.
2. For cardinalities defined over two streams, we formally analyze the statistical efficiency of our proportional-union estimator, as compared to the Maximum Likelihood Estimator (MLE). We show that the proportional-union estimator has almost the same statistical efficiency as the MLE, yet is computationally much simpler than MLE. For cardinality defined over higher number of streams, our estimator is still computationally simple, but the MLE becomes intractable due to the complexity of the likelihood.
3. We demonstrate both analytically and by simulation that our proportional-union estimator has a superior performance compared to existing methods by [19, 6, 13, 12]. We show that for algorithms [19, 6] based on one stream cardinality counting, the required memory for a given relative standard error δ grows proportional to p^{-2} for a large value of p^{-1} whereas our method is p^{-1} scaling. On the other hand, for the given ϵ value, the required memory for [12] (an improved version of [13]) grows proportionally to $O(\log^2 N)$ for a large N, which is larger than $O(\log\log N)$ required by ours. These differences are significant for large N and small p where the query is more challenging. Furthermore, the results from our Internet trace driven simulation validate our theoretical analysis very well.

Finally, we note that although our estimator has a superior performance over existing methods, we only handles the insert only stream but not general update stream as [13, 12] do.

1.2 Paper Outline

The rest of this paper is organized as follows. In Section 2, we review some basic statistical concepts and describe a continuous variant of the FM sketches that our proposed method is based on. In Section 3, we develop a proportional-union estimator from the continuous FM sketches, and characterize its performance analytically. In Section 4, we analyze the statistical efficiency of the proportional-union estimator by comparing it with the maximum likelihood estimator for expressions over two streams. Section 5 gives a systematic analytic performance comparison between the existing methods and the proposed method, which shows that our method is superior. This is further demonstrated in Section 6 using simulation studies, and empirical evaluations of a real network traffic matrix estimation example. Finally, we concludes in Section 7.

1.3 Notations

There are a few notations that are used throughout the paper. We use $P(\cdot)$ to present the probability function, $\mathrm{E}(\cdot)$ and $\mathrm{var}(\cdot)$ to present the expectation and variance of a random variable, $\mathrm{corr}(\cdot,\cdot)$ or $\mathrm{cov}(\cdot,\cdot)$ to represent the correlation or covariance between two random variables, and $\xrightarrow{d}$ to represent the convergence in distribution. We also use $\doteq$ to represent definitions and $a \approx b$ to represent $a/b \approx 1$. We use $Exp(r)$ to represent an exponential distribution with rate r and $Normal(\mu, \sigma^2)$ to represent a Gaussian distribution with mean μ and variance σ^2. We use log to denote logarithm with base 2 unless it is specified.

2. PRELIMINARIES

In this section, we first give a brief introduction of some statistical concepts that are used later in the paper, including the notion of statistical efficiency. Then we describe a continuous variant of the Flajolet-Martin (FM) sketch that we use to develop an efficient proportional-union estimator for stream expression cardinalities.

2.1 Statistical Inference and Efficiency of Estimators

Let $f_\theta(x)$ be a probability density/mass function for a random variable x parameterized by θ. Given a random sample of size n, say $x_1, x_2, ..., x_n$, from $f_\theta(\cdot)$, the *likelihood function* $\mathcal{L}(\theta)$ of θ is given by

$$L(\theta) = \Pi_{i=1}^n f_\theta(x_i).$$

Let $I(\theta)$ be the Fisher information defined by

$$I(\theta) = \frac{1}{n}\mathrm{E}\left[\left(\frac{\partial}{\partial\theta}\log L(\theta)\right)^2\right], \tag{1}$$

and let $\widehat{\theta}$ be an unbiased estimator of θ based on the given sample $x_1, x_2, ..., x_n$. Under regular conditions, the variance of $\widehat{\theta}$ is then bounded by the reciprocal of the Fisher information $I(\theta)$, i.e.,

$$\mathrm{var}\left(\widehat{\theta}\right) \geq \frac{1}{I(\theta)}.$$

This is the well-known Cramer-Rao inequality, also known as the Cramer-Rao lower bound (CRB) [4]. Now define the efficiency of $\widehat{\theta}$ using the Cramer-Rao lower bound by

$$eff(\widehat{\theta}) = \frac{1/I(\theta)}{\mathrm{var}(\widehat{\theta})}. \tag{2}$$

From the Cramer-Rao inequality, $eff(\widehat{\theta}) \leq 1$. A good statistical method seeks to find an efficient estimator $\widehat{\theta}$ that has a large value of efficiency. One such method is the popular *Maximum Likelihood Estimate* (MLE). The MLE of θ, say $\hat{\theta}_{MLE}$, is defined as the maximizer of $L(\theta)$, i.e.,

$$\hat{\theta}_{MLE} = \mathrm{argmax}_\theta L(\theta).$$

It can be shown that as the number of samples increases to infinity, the MLE is asymptotically unbiased and efficient, i.e., it achieves the Cramer-Rao lower bound.

2.2 Continuous Flajolet-Martin Sketches for Single Stream Cardinality Counting

In a seminal paper [11], Flajolet and Martin proposed an estimator for counting distinct values in a single stream, using a hash-based probabilistic sketches with $O(\log N)$ space where N is the true cardinality. In the original version of the Flajolet-Martin (FM) sketching algorithm, a hash function is used to map an element in the stream to an integer that follows a geometric distribution. Here we describe a continuous variant of the algorithm by replacing the geometric random number with a uniform random number. We use the continuous variant here to simplify the statistical analysis that we present later. To generate independent replicates of the statistics used for counting the cardinalities, we also employ a technique referred to as *stochastic averaging* in [8] by randomly distributing the elements over an array of buckets.

For a single stream $\mathcal{T}$, let $[M]$ be the data domain of its element t. In the IP flow counting example, t is the packet flow ID, and $[M]$ is the set of all possible values of flow IDs. We generate the continuous FM sketch $Y[k], k = 1, \ldots, m$, an array of size m, as follows. First, we initialize $Y[k]$ with 1 for all k. Let $h : [M] \to \{1, \cdots, m\}$ be a universal hash function that maps an element uniformly over an array of m buckets, and let $g : [M] \to [0, 1]$ be a universal hash function that maps the element t to a uniform random number in $[0, 1]$, independent of h. For each incoming element t, let $k = h(t)$ be the bucket that t is mapped to, then we update $Y[k]$ by

$$Y[k] \quad \leftarrow \quad \min(Y[k], g(t)). \tag{3}$$

A bucket value will remain as 1 if no element is hashed to that bucket. Algorithm 1 summarizes the continuous FM sketch generation for stream $\mathcal{T}$.

Algorithm 1 Continuous FM Sketch for a Stream $\mathcal{T}$

1: Initialize a hash array Y of size m with values 1.
2: **for** each incoming element t of $\mathcal{T}$ **do**
3: Hash t into a bucket $k = h(t)$ uniformly, where $k \in \{1, 2, \ldots, m\}$
4: Generate a random number $g(t)$ uniformly on the interval $[0, 1]$.
5: Update $Y[k]$ by $\min(Y[k], g(t))$.
6: Return Y at the end of the stream.

Let $\mu = m^{-1}|\mathcal{T}|$ be the mean number of distinct items in each bucket for $\mathcal{T}$. The following lemma characterizes the statistical properties of the continuous FM sketch $Y[k], k = 1, \ldots, m$. The proof is provided in the appendix.

LEMMA 1. *For a large m, $Y[k]$ follows approximately a right censored exponential distribution with rate μ, i.e.,*

$$P(Y[k] \geq y) \approx e^{-\mu y}, \quad \textit{for } y \in [0, 1],$$

and $P(Y[k] = 1) \approx e^{-\mu}$. In addition,

$$corr(Y[k], Y[j]) \approx -\frac{1}{|\mathcal{T}|}, \quad \textit{for } k \neq j.$$

Obviously, when μ is large, $Y[k], k = 1, \ldots, m$ approximates an independent sample of m exponential random variables with rate μ.

By ignoring the weak dependence among $\{Y[k], 1 \leq k \leq m\}$, the likelihood function of μ can be written as

$$L(\mu) \quad = \quad e^{-\mu \sum_{k=1}^m I(Y[k]=1)} \prod_{Y[k]<1} \mu e^{-\mu Y[k]},$$

where $I(\cdot)$ is the indicator function. Thus the MLE of μ and hence $|\mathcal{T}|$ are given by

$$\widehat{\mu} = \frac{\sum_{k=1}^m I(Y[k] < 1)}{\sum_{k=1}^m Y[k]}, \quad \widehat{|\mathcal{T}|} = m\widehat{\mu}. \tag{4}$$

LEMMA 2. *As m goes to infinity,*

$$\sqrt{m}\left(\frac{\widehat{|\mathcal{T}|}}{|\mathcal{T}|} - 1\right) \xrightarrow{d} Normal\left(0, (1 - e^{-\mu})^{-1}\right).$$

Note that for a large μ, say $\mu > 10$, the limiting variance in Lemma 2 is approximately 1, which is close to the original FM algorithm reported in [10].

Finally we note that [16] have also considered using continuous random numbers for the FM sketches, but their sketch is based on higher order statistics instead of the minimal statistic described in the above. In addition, their estimators are not based on likelihood.

We assume throughout that two universal hash functions h and g are available that produce random independent numbers. To be more realistic, similar to [13], we may need to use t-wise independent hashing [2, 17, 5] which requires additional storage cost for storing an appropriate seed. Here t depends on desired accuracy, which we do not pursue detailed analysis in this paper.

3. PROPORTIONAL UNION METHOD

Below we develop a novel method for estimating the cardinality of a general stream expression. To make the description clearer, we first demonstrate our method for a set expression over two streams $\mathcal{T}_1, \mathcal{T}_2$, and then generalize it to arbitrary number of streams $\mathcal{T}_j, j = 1, \ldots, J$.

3.1 Set Expressions over Two Streams

We first state a simple lemma that we use to derive our proportional-union estimation method.

Lemma 3. *Let* $\overline{E}_1 = \min(E_1, 1)$ *and* $\overline{E}_2 = \min(E_2, 1)$ *be two right censored exponentials, where* $E_1 \sim Exp(r_1)$, $E_2 \sim Exp(r_2)$, *and* E_1 *is independent of* E_2. *Then*

$$\min(\overline{E}_1, \overline{E}_2) \sim \min(Exp(r_1 + r_2), 1),$$

and

$$P(\overline{E}_1 = \min(\overline{E}_1, \overline{E}_2)) = e^{-(r_1+r_2)} + (1 - e^{-(r_1+r_2)})\frac{r_1}{r_1 + r_2}.$$

To formally present our proportional-union method, we first introduce some notations. For each stream $\mathcal{T}_j, j = 1, 2$, let $Y_j[k], 1 \le k \le m$ be the corresponding continuous FM sketch, obtained using Algorithm 1 with the same hash functions h and g. The stream union $\mathcal{T}_1 \cup \mathcal{T}_2$ can be partitioned into three subsets, $\mathcal{T}_1 \cap \mathcal{T}_2$, $\mathcal{T}_1 \backslash \mathcal{T}_2$, and $\mathcal{T}_2 \backslash \mathcal{T}_1$ (shown in Figure 1). Let $X_0[k], X_1[k], X_2[k], (k = 1, \ldots, m)$ be the virtual continuous FM sketches corresponding to the subsets $\mathcal{T}_1 \cap \mathcal{T}_2$, $\mathcal{T}_1 \backslash \mathcal{T}_2$, and $\mathcal{T}_2 \backslash \mathcal{T}_1$, respectively, obtained also by Algorithm 1 but not observable. To simplify the presentation, in the following, we shall omit $[k]$ when we refer to these sketches.

Notice that $\mathcal{T}_1 = (\mathcal{T}_1 \cap \mathcal{T}_2) \cup (\mathcal{T}_1 \backslash \mathcal{T}_2)$, and similarly, $\mathcal{T}_2 = (\mathcal{T}_1 \cap \mathcal{T}_2) \cup (\mathcal{T}_2 \backslash \mathcal{T}_1)$ (again see Figure 1), it is obvious that

$$Y_1 = \min(X_0, X_1), \quad Y_2 = \min(X_0, X_2). \tag{5}$$

Furthermore, by Lemma 1, when m is large enough such that $(1 - m^{-1})^m \approx e^{-1}$, X_0, X_1, X_2 approximates a right censored exponential, i.e.,

$$X_0 \sim \min(1, Exp\left(m^{-1}|\mathcal{T}_1 \cap \mathcal{T}_2|\right)),$$
$$X_1 \sim \min(1, Exp(m^{-1}|\mathcal{T}_1 \backslash \mathcal{T}_2|)),$$
$$X_2 \sim \min(1, Exp(m^{-1}|\mathcal{T}_2 \backslash \mathcal{T}_1|)).$$

Let $N = |\mathcal{T}_1 \cup \mathcal{T}_2|$ and $\epsilon = \exp(-N/m)$. Applying Lemma 3, we have

$$\begin{aligned} P(Y_1 = Y_2) &= P(\min(X_0, X_1) = \min(X_0, X_2)) \\ &= P(X_0 = \min(X_0, X_1, X_2)) \\ &\approx \epsilon + (1 - \epsilon)N^{-1}|\mathcal{T}_1 \cap \mathcal{T}_2|. \end{aligned} \tag{6}$$

Here we use the fact that $P(X_1 = X_2 < 1) = 0$. This immediately leads to the following theorem. We omit the proof for the other two probability approximations as their proofs are very similar.

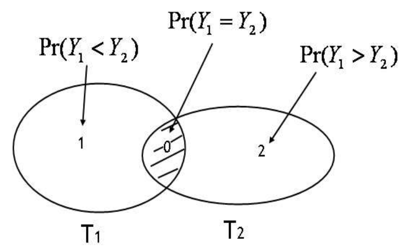

Figure 1: Proportions of set expression cardinalities

Theorem 1. *Suppose that* m *is large enough such that* $(1-m^{-1})^m \approx e^{-1}$. *Let* $N = |\mathcal{T}_1 \cup \mathcal{T}_2|$, *and* $\epsilon = \exp(-N/m)$. *Then*

$$\begin{aligned} P(Y_1 = Y_2) &\approx \epsilon + (1 - \epsilon)N^{-1}|\mathcal{T}_1 \cap \mathcal{T}_2|, \\ P(Y_1 < Y_2) &\approx (1 - \epsilon)N^{-1}|\mathcal{T}_1 \backslash \mathcal{T}_2|, \\ P(Y_1 > Y_2) &\approx (1 - \epsilon)N^{-1}|\mathcal{T}_2 \backslash \mathcal{T}_1|. \end{aligned} \tag{7}$$

Theorem 1 motivates a proportional-union estimate for the cardinality $|\mathcal{T}_1 \cap \mathcal{T}_2|$, $|\mathcal{T}_1 \backslash \mathcal{T}_2|$ and $|\mathcal{T}_2 \backslash \mathcal{T}_1|$. For example, when N is large enough such that ϵ is negligible, from Theorem 1, $|\mathcal{T}_1 \cap \mathcal{T}_2| \approx NP(Y_1 = Y_2)$. Therefore, an estimate of $|\mathcal{T}_1 \cap \mathcal{T}_2|$ can be obtained using the product of estimates of $|\mathcal{T}_1 \cup \mathcal{T}_2|$ and $P(Y_1 = Y_2)$. Notice that the continuous FM sketch (Algorithm 1) of the stream union $\mathcal{T}_1 \cup \mathcal{T}_2$ is exactly the bucket-wise minimum $\min(Y_1, Y_2)$, therefore, $|\mathcal{T}_1 \cup \mathcal{T}_2|$ can be easily estimated by using Equation 4 on the new sketch. Furthermore, $P(Y_1 = Y_2)$ can be estimated empirically from the observed sketch-pair (Y_1, Y_2).

Let $\widehat{N}$ be the estimate of N by using Equation 4 on the continuous FM sketch of the stream union $\mathcal{T}_1 \cup \mathcal{T}_2$, defined by $\min(Y_1, Y_2)$. Then for a large N such that ϵ is negligible, the cardinalities $|\mathcal{T}_1 \cap \mathcal{T}_2|$, $|\mathcal{T}_1 \backslash \mathcal{T}_2|$, and $|\mathcal{T}_2 \backslash \mathcal{T}_1|$ can be estimated by

$$\begin{aligned} \widehat{|\mathcal{T}_1 \cap \mathcal{T}_2|}^{(PU)} &= \widehat{N} \cdot \widehat{P}(Y_1 = Y_2) \\ \widehat{|\mathcal{T}_1 \backslash \mathcal{T}_2|}^{(PU)} &= \widehat{N} \cdot \widehat{P}(Y_1 < Y_2) \\ \widehat{|\mathcal{T}_2 \backslash \mathcal{T}_1|}^{(PU)} &= \widehat{N} \cdot \widehat{P}(Y_2 < Y_1), \end{aligned} \tag{8}$$

where $\widehat{P}$ represents the empirical probabilities based on the observed sketch pair $(Y_1[k], Y_2[k]), k = 1, \ldots, m$. We call this as the *proportional-union* estimator in this paper. When $\epsilon = \exp(-N/m)$ is not negligible, one can invert Equation 7 to obtain the proportional-union estimator of the cardinalities.

3.2 Set Expressions over Multiple Streams

Now we generalize the proportional-union method to estimate the cardinality of a set expression over multiple streams, $\mathcal{T}_j$, $j = 1, 2, \cdots, J$, with $J > 2$. Let $Y_j[k], k = 1, \ldots, m$ be the corresponding continuous FM sketches by Algorithm 1 for stream $\mathcal{T}_j$. As before, to simplify the presentation, we shall omit $[k]$ when we refer the sketches at a bucket location k. Define

$$Y_\cup = \min(Y_1, \cdots, Y_J). \tag{9}$$

Obviously, $Y_\cup$ is the continuous FM sketch for the stream union $\cup_{j=1}^{J}\mathcal{T}_j$. The following is an extension of Theorem 1 on $P(Y_1 = Y_2)$ to the case of multiple streams.

THEOREM 2. *Suppose m is large enough such that $(1 - m^{-1})^m \approx e^{-1}$. Let $N = |\cup_{j=1}^{J}\mathcal{T}_j|$, and $\epsilon = \exp(-N/m)$. Then for $1 \le d \le J$,*

$$P(Y_1 = Y_2 = \cdots = Y_d = Y_\cup) \approx \epsilon + (1-\epsilon)N^{-1}|\cap_{j=1}^{d}\mathcal{T}_d|. \quad (10)$$

PROOF. For simplicity, we prove the result for the case $J = 3$ and $d = 2$: the generalization to arbitrary J and d values is straightforward.

Figure 2 shows a diagram of the three streams $\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3$ under consideration. Notice that the stream union can be partitioned into 7 exclusive subsets numbered from 0 to 6: for example, the subset numbered by 0 denotes $\cap_{j=1}^{3}\mathcal{T}_j$. Let X_j be the virtual continuous FM sketch for the subset j, $j = 0, \cdots, 6$. Then by the set relations (see Figure 2), we have $Y_1 = \min(X_0, X_4, X_5, X_1)$, $Y_2 = \min(X_0, X_4, X_6, X_2)$ and $Y_3 = \min(X_0, X_5, X_6, X_3)$. Since $\min(X_0, X_4)$ contributes to both Y_1 and Y_2, it is easy to verify that $Y_1 = Y_2$ is equivalent to $\min(X_0, X_4) = \min(Y_1, Y_2)$. Hence

$$\begin{aligned} P(Y_1 = Y_2 = Y_\cup) &= P(\min(X_0, X_4) = Y_\cup) \\ &= P(\min(X_0, X_4) = \min(X_j, 0 \le j \le 6)). \end{aligned}$$

For a large m such that $(1 - m^{-1})^m \approx e^{-1}$, by Lemma 1, $\min(X_0, X_4)$ and $\min(X_j, 0 \le j \le 6)$ follow right censored exponential distributions with rate $m^{-1}|\mathcal{T}_1 \cap \mathcal{T}_2|$ and $m^{-1}|\cup_{j=1}^{3}\mathcal{T}_j|$, respectively. Now the result follows readily from Lemma 3. □

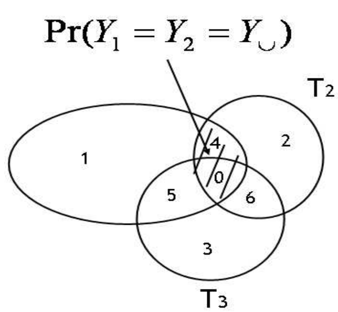

Figure 2: Diagram of set expressions over three streams

Suppose that $\mathcal{S}$ is a set expression over the J streams, whose cardinality is the subject of interest. To complete the generalization of the proportional-union method for $|\mathcal{S}|$, we need two additional techniques for dealing with set expressions, which we shall illustrate using an example. Consider $\mathcal{S} = \mathcal{T}_1 \backslash ((\mathcal{T}_2 \cap \mathcal{T}_3) \cup \mathcal{T}_4)$. The first technique is to remove the set differences appeared in the expression using the relation

$$|\mathcal{A} \backslash \mathcal{B}| = |\mathcal{A}| - |\mathcal{A} \cap \mathcal{B}|.$$

Notice that we can use this repeatedly if there are multiple set differences. In our example, this implies

$$|\mathcal{T}_1 \backslash ((\mathcal{T}_2 \cap \mathcal{T}_3) \cup \mathcal{T}_4)| \quad = \quad |\mathcal{T}_1| - |(\mathcal{T}_1 \cap ((\mathcal{T}_2 \cap \mathcal{T}_3) \cup \mathcal{T}_4))|.$$

Now without the loss of generality, we can assume that the set expression only involves unions and intersections. The second technique is to rewrite the set expression in terms of intersections of set unions, i.e,

$$(\mathcal{A} \cap \mathcal{B}) \cup \mathcal{C} = (\mathcal{A} \cup \mathcal{C}) \cap (\mathcal{B} \cup \mathcal{C}).$$

In our example, this implies

$$\mathcal{T}_1 \cap ((\mathcal{T}_2 \cap \mathcal{T}_3) \cup \mathcal{T}_4) \quad = \quad \mathcal{T}_1 \cap (\mathcal{T}_2 \cup \mathcal{T}_4) \cap (\mathcal{T}_3 \cup \mathcal{T}_4).$$

Notice that the continuous FM sketch of a set union of d streams is exactly the minimum of the d individual FM sketches. Let

$$p_\mathcal{S} = |\mathcal{S}|/N$$

be the proportion of $|S|$ in the total union. Applying Theorem 2 with the sketches Y_j replaced by the sketches corresponds to the set unions, we can derive a close approximation of $p_\mathcal{S}$. In our example, we have (for simplicity assuming ϵ ignorable)

$$\begin{aligned} N^{-1}|\mathcal{T}_1| &\approx P(Y_1 = Y_\cup), \\ N^{-1}|\mathcal{T}_1 \cap ((\mathcal{T}_2 \cap \mathcal{T}_3) \cup \mathcal{T}_4)| & \\ &\approx P(Y_1 = \min(Y_2, Y_4) = \min(Y_3, Y_4) = Y_\cup), \end{aligned}$$

and thus

$$\begin{aligned} p_\mathcal{S} &= N^{-1}|\mathcal{T}_1 \backslash ((\mathcal{T}_2 \cap \mathcal{T}_3) \cup \mathcal{T}_4)| \\ &\approx (P(Y_1 = Y_\cup) - P(Y_1 = \min(Y_2, Y_4) = \min(Y_3, Y_4) = Y_\cup)). \end{aligned}$$

Let $\widehat{N}$ be the estimate of N by $Y_\cup$, and let $\widehat{p}_\mathcal{S}$ be the empirical proportion based on the observed sketch tuple of the J streams. When N is large enough such that $\epsilon = \exp(-N/m)$ is negligible, the proportional-union estimate of $|\mathcal{S}|$ can be obtained in a straightforward way as in the two stream case:

$$\widehat{|\mathcal{S}|}^{(PU)} \quad = \quad \widehat{N} \cdot \widehat{p}_\mathcal{S}.$$

Note that ϵ can be estimated by $e^{-\hat{N}/m}$. When ϵ is not negligible, one can invert the proportion equation derived from the above procedure to obtain the proportional-union estimator of $|\mathcal{S}|$ correspondingly.

The following result states that the relative standard error for the proportional-union estimator grows linearly with $p_\mathcal{S}^{-1}$. The proof is given in the appendix.

THEOREM 3. *The relative standard error (RSE) of the proportional-union estimator for a set expression $\mathcal{S}$ is:*

$$E\left(\frac{\widehat{|\mathcal{S}|}^{(PU)}}{|\mathcal{S}|} - 1\right)^2 \quad \approx \quad \frac{1}{mp_\mathcal{S}}.$$

3.3 Memory Requirement

For the stream expression $\mathcal{S}$, let δ be a specified value of the relative standard error of our proportional-union estimate of $|\mathcal{S}|$. From Theorem 3, the required number of buckets for a given δ is

$$m \approx \delta^{-2}p_\mathcal{S}^{-1}.$$

Let E be a unit exponential random variable, and $\lambda_j = |\mathcal{T}_j|/m, j = 1, \ldots, J$. Notice that by Lemma 1, the continous FM sketch Y_j for $\mathcal{T}_j$ is a right censored exponential,

$$Y_j \sim \min(\lambda_j^{-1}E, 1). \quad (11)$$

Since $\lambda_j \sim O(N)$, this implies that Y_j requires $O(\log N)$ storage bits. In the following, we describe a procedure that can reduce the per-bucket storage of the sketch statsitics Y_j from $\log N$ to $\log\log N$, by storing $\log Y_j$ instead.

Notice that by Equation 11,

$$\log Y_j \sim \min(0, -\log(\lambda_j) + \log E). \tag{12}$$

Assume $\log\log\lambda_j$ is an integer. Thus $\log\lambda_j$ requires $\log\log\lambda_j$ storage bits, and $\log Y$ requires at most $\log\log(N) + a$ storage bits, where the a bits are used for storing the decimals of $\log E$ (for reference, we note that the 0.1% and 99.9% quantiles of $\log E$ are -6.907 and 1.933 respectively). Therefore, now the per-bucket storage is at most $O(\log\log N)$, and the total required memory is $\delta^{-2}p_{\mathcal{S}}^{-1}(\log\log N + a)$.

From experimental studies presented in Section 6, we observe that $a = 10$ bits is enough for storing the decimals of $\log E$ so that the overal accuracy of the cardinality estimate is not compromised. This can be further justified using a careful bias analysis of the probability approximation of $p_{\mathcal{S}}$ (e.g. Equation 7 and 10.) Consider $\mathcal{S} = \mathcal{T}_1 \cap \mathcal{T}_2$ over the J streams, and $p = p_{\mathcal{S}} = P(Y_1 = Y_2 = Y_\cup)$ for example. Let $p^{(I)}$ be the new probability based on the discretized sketches $Y_j, j = 1, \ldots, J$ described above. It can be shown that $p \le p^{(I)} \le p + 2^{-a+1}$. Therefore if $a = 10$, the difference in the probabilities is at most 0.002, which is negligible for practical purposes.

Finally, we make note of a direct method for computing the logarithmic sketch $\log Y_j$. By Algorithm 1, we can write

$$Y_j = \min(1, U_1, \cdots, U_B),$$

where B is a Binomial random number representing the number of distinct elements that are mapped to the bucket, and each U_i is a uniform random number in $[0, 1]$. Notice that $-\log U_i$ follows a unit exponential distribution, therefore

$$\log Y_j = -\max(0, -\log U_1, \cdots, -\log U_B).$$

Thus, to generate the logarithmic continuous FM sketches, we can replace the uniform random number generator $g(\cdot)$ by a unit exponential random number generator (with decimal truncated into a bits) and replace the minimum update by maximum. In this way, we can avoid taking the logarithm in the sketch generation. Now the initial values for the buckets become 0 instead of 1.

4. EFFICIENCY OF THE PROPORTIONAL UNION METHOD

In this section, we investigate the efficiency of our proposed proportional-union method, using the formal statistical methods described in Section 2.1. We derive the likelihood of the continuous FM sketches for the case of two streams, obtain the MLE of the cardinality parameters, and then compare its asymptotic variance with that of the proportional-union method. As we have explained earlier, MLE is asymptotically most efficient as it can achieve the Cramer-Rao lower bound. For a set expression $\mathcal{S}$ whose cardinality is the subject of interest here, again let $p = |\mathcal{S}|/N$ be the proportion of the cardinality of $\mathcal{S}$ in the total union. We show that in the two stream case, our proportional-union method is as efficient as that of MLE when p is small. As shown below, the MLE for the case of two streams is quite involved but manageable, however the MLE for the case $J > 2$ is much more complicated and we do not investigate it in this paper.

We adopt the same notation that we use in Section 3.1. Furthermore, we define

$$\lambda_0 = |\mathcal{T}_1 \cap \mathcal{T}_2|/m, \quad \lambda_1 = |\mathcal{T}_1 \backslash \mathcal{T}_2|/m, \quad \lambda_2 = |\mathcal{T}_2 \backslash \mathcal{T}_1|/m,$$

as the unknown cardinality parameters, and let

$$\theta = (\lambda_0, \lambda_1, \lambda_2)^T.$$

To simplify the presentation, in the following, we omit $[k]$ when we refer the continuous FM sketches $Y_1[k]$, $Y_2[k]$, $X_0[k]$, $X_1[k]$, $X_2[k]$ at a bucket location k. By Lemma 1 and the relation in Equation 5, for $y_1, y_2 \in [0, 1]$, we have

$$\begin{aligned} & P(Y_1 \ge y_1, Y_2 \ge y_2) \\ = \; & P(X_0 \ge \max(y_1, y_2), X_1 \ge y_1, X_2 \ge y_2) \\ \approx \; & \exp\left(-\lambda_0 \max(y_1, y_2) - \sum_{j=1}^{2} \lambda_j y_j\right), \end{aligned}$$

where "$\approx$" holds since $(1 - m^{-1})^m \approx e^{-1}$ for large m. Let $f_\lambda(\cdot)$ denote the density function of an exponential random variable with rate λ, i.e. $f_\lambda(x) = \lambda e^{-\lambda x}$, $x \ge 0$. Then the density function for the continuous FM sketches (Y_1, Y_2), i.e., $P(Y_1 = y_1, Y_2 = y_2)$, $0 \le y_1, y_2 \le 1$, can be expressed as

$$\begin{cases} e^{-(\lambda_0+\lambda_1+\lambda_2)}, & y_1 = y_2 = 1 \quad \text{(case 1)} \\ e^{-(\lambda_0+\lambda_2)} f_{\lambda_1}(y_1), & y_1 < y_2 = 1 \quad \text{(case 2)} \\ e^{-(\lambda_0+\lambda_1)} f_{\lambda_2}(y_2), & y_2 < y_1 = 1 \quad \text{(case 3)} \\ \lambda_0 e^{-(\lambda_0+\lambda_1+\lambda_2)y_1}, & y_1 = y_2 < 1 \quad \text{(case 4)} \\ f_{\lambda_2}(y_2) f_{\lambda+\lambda_1}(y_1), & y_2 < y_1 < 1 \quad \text{(case 5)} \\ f_{\lambda_1}(y_1) f_{\lambda+\lambda_2}(y_2), & y_1 < y_2 < 1 \quad \text{(case 6).} \end{cases} \tag{13}$$

Again from Lemma 1, the sketches at two different bucket locations $(Y_1[k], Y_2[k])$ and $(Y_1[j], Y_2[j])$ for $j \ne k$, are very weakly dependent. Let $l(\theta)$ be the negative logarithmic likelihood function of the continuous FM sketches $(Y_1[k], Y_2[k]), k = 1, \ldots, m$, i.e.,

$$l(\theta) = -\sum_{k=1}^{m} \log\{P(Y_1[k], Y_2[k])\}.$$

The following lemma gives the gradient and Hessian matrix of $l(\theta)$ with respect to θ, noting that the expectation of the Hessian matrix is the same as the information matrix $\mathcal{I}(\theta)$ defined in Equation 1. The proof is based on direct calculation using Equation 13 and omitted.

LEMMA 4. *The gradient* $\mathbf{g} = \frac{\partial l(\theta)}{\partial \theta}$ *is given by*

$$\begin{bmatrix} S_3 - \frac{m_6}{\lambda_0+\lambda_2} - \frac{m_5}{\lambda_0+\lambda_1} - \frac{m_4}{\lambda_0} \\ S_1 - \frac{m_2+m_6}{\lambda_1} - \frac{m_5}{\lambda_0+\lambda_1} \\ S_2 - \frac{m_3+m_5}{\lambda_2} - \frac{m_6}{\lambda_0+\lambda_2} \end{bmatrix} \tag{14}$$

where m_i are the number of buckets of case i defined in Equation 13, $S_1 = \sum_{k=1}^m Y_1[k]$, $S_2 = \sum_{k=1}^m Y_2[k]$, and $S_3 = \sum_{i=1}^m \max(Y_1[k], Y_2[k])$.

The Hessian matrix $\mathcal{H} = \frac{\partial^2 l(\theta)}{\partial\theta\partial\theta^T}$ is a 3×3 symmetric non-

negative definite matrix with elements $\mathcal{H}_{ij}$ given by:

$$\begin{cases} \mathcal{H}_{11} = \frac{m_6}{(\lambda_0+\lambda_2)^2} + \frac{m_5}{(\lambda_0+\lambda_1)^2} + \frac{m_4}{\lambda_0^2}, \\ \mathcal{H}_{12} = \mathcal{H}_{21} = \frac{m_5}{(\lambda_0+\lambda_1)^2}, \\ \mathcal{H}_{13} = \mathcal{H}_{31} = \frac{m_6}{(\lambda_0+\lambda_2)^2} \\ \mathcal{H}_{22} = \frac{m_2+m_6}{\lambda_1^2} + \frac{m_5}{(\lambda_0+\lambda_1)^2}, \\ \mathcal{H}_{23} = \mathcal{H}_{32} = 0, \\ \mathcal{H}_{33} = \frac{m_3+m_5}{\lambda_2^2} + \frac{m_6}{(\lambda_0+\lambda_2)^2}. \end{cases} \quad (15)$$

Furthermore if $m_5 > 0$ and $m_6 > 0$, $\mathcal{H}$ is strictly positive definite and thus $l(\theta)$ is a strictly convex function. If $|\mathcal{T}_1 \cap \mathcal{T}_2| > 0$, then the probability

$$P(m_5 > 0, m_6 > 0) \approx 1 - \sum_{j=1}^{2} \left(\frac{|\mathcal{T}_j|}{N}\right)^m + \left(\frac{|\mathcal{T}_1 \cap \mathcal{T}_2|}{N}\right)^m$$

is almost 1 for large m.

Unlike the proportional-union estimate, the MLE of θ that minimizes $l(\theta)$ does not have a closed-form solution. By Lemma 4, $l(\theta)$ is strictly convex with probability almost 1, and hence, its unique minimum can be located using a simple Newton-Rapson algorithm. Let $\widehat{\theta}^{(MLE)}$ be the MLE of θ by minimizing $l(\theta)$, then the MLE of cardinalities is simply

$$\left(\widehat{|\mathcal{T}_1 \cap \mathcal{T}_2|}^{(MLE)}, \widehat{|\mathcal{T}_1 \backslash \mathcal{T}_2|}^{(MLE)}, \widehat{|\mathcal{T}_2 \backslash \mathcal{T}_1|}^{(MLE)}\right) = m\widehat{\theta}^{(MLE)}. \quad (16)$$

In our experience through simulation and experimental studies, the Newton-Rapson iteration typically requires only a few steps (less than 5) before convergence is reached.

The following theorem gives the asymptotic distribution for the relative accuracy of $\widehat{\theta}^{(MLE)}$. The proof is given in the appendix.

THEOREM 4. *Let $p_{\mathcal{S}} = |\mathcal{S}|/N$. For $\lambda_0 > 0$ and large $\lambda_0 + \lambda_i$, $i = 1, 2$, as m goes to infinity, we have for $\mathcal{S} = \mathcal{T}_1 \cap \mathcal{T}_2$, $\mathcal{T}_1 - \mathcal{T}_2$ and $\mathcal{T}_2 - \mathcal{T}_1$,*

$$\sqrt{m}\left(\frac{\widehat{|\mathcal{S}|}^{(MLE)}}{|\mathcal{S}|} - 1\right) \rightarrow Normal\left(0, \frac{1}{p_{\mathcal{S}}(1+b)})\right),$$

where b is a small number given in Equation 19 in the appendix.

Comparing the relative standard error of the MLE (Theorem 4 above) and that of the proportional-union estimate (Theorem 3), the efficiency of proportional-union estimate as compared to MLE, defined by the ratio of their MSEs (i.e., Equation 2) is

$$eff\left(\widehat{|\mathcal{S}|}^{(PU)}\right) = \frac{1}{1+b},$$

which is close to 1 for a small $p_{\mathcal{S}}$, see the appendix for details. Therefore, the proportion-union estimate has almost the same efficiency as MLE.

5. COMPARISONS WITH EXISTING METHODS

In this section, we compare our proportional-union method with two existing methods for the stream expression cardinality estimation problem, both of which use hash based probabilistic sketches. In the first approach [19, 6], the cardinality of an arbitrary stream expression is expressed as sums/differences of the cardinalities of set unions, and the cardinality of each set union is counted using existing algorithms developed for single stream cardinality counting. In the second approach [13, 12], a novel *2-level hash* sketching algorithm is used to directly estimate the cardinality of an arbitrary set expression over more general update streams. We discuss performance limitations of both approaches and and demonstrate analytically that our proportional-union method is superior. Further empirical evidence for this is reported in the next section.

5.1 Algorithms based on One Stream Cardinality Counting

We give a brief description of the approach adopted by [19, 6]. Both are based on single stream cardinality counting: the difference is that the *LLog-bitmap* method by [6] used a combination of the LogLog algorithm [8] (an improved version of the FM sketch) and the bitmap sketch [9], and [19] used only the bitmap sketch. Although their approach is developed only for the cardinality estimation over two streams, specifically, $|\mathcal{T}_1 \cap \mathcal{T}_2|$, in the context of IP traffic matrix estimation, the generalization to multiple streams is straightforward.

Notice that

$$|\mathcal{T}_1 \cap \mathcal{T}_2| = |\mathcal{T}_1| + |\mathcal{T}_2| - |\mathcal{T}_1 \cup \mathcal{T}_2|. \quad (17)$$

Therefore, $|\mathcal{T}_1 \cap \mathcal{T}_2|$ can be estimated from the three cardinalities, $|\mathcal{T}_1|$, $|\mathcal{T}_2|$ and $|\mathcal{T}_1 \cup \mathcal{T}_2|$. Let Y_j be the probabilistic sketch for stream $\mathcal{T}_j$, $j = 1, 2$. It turns out that for both the original FM sketch [10], and bitmap sketch [9], the corresponding probabilistic sketch for the stream union $\mathcal{T}_1 \cup \mathcal{T}_2$ is simply $Y_\cup[k] \equiv \min(Y_1[k], Y_2[k])$. Now the estimation of $|\mathcal{T}_1 \cap \mathcal{T}_2|$ is straightforward by using the FM or bitmap cardinality counting algorithm for individual streams $\mathcal{T}_1$, $\mathcal{T}_2$ and $\mathcal{T}_1 \cup \mathcal{T}_2$.

Despite its mathematical simplicity, their approach does not fully explore the correlation among the stream sketches, and thus it is quite inefficient. Let $\mathcal{S} = \mathcal{T}_1 \cap \mathcal{T}_2$ be the set expression of interest and $\widehat{|\mathcal{S}|}^{(1D)}$ be the estimate of $|\mathcal{S}|$ using a one-stream cardinality counting algorithm (e.g. FM or bitmap sketch) described above. Let $N = |\mathcal{T}_1 \cup \mathcal{T}_2|$. The following theorem states that the relative mean-square-error of $\widehat{|\mathcal{S}|}^{(1D)}$ grows quadratically with $N/|\mathcal{S}|$ which is much worse than $N/|\mathcal{S}|$ scaling of our method (Theorem 3). In fact the result can be easily generalized to the intersections of multiple streams.

THEOREM 5. *For $\mathcal{S} = \mathcal{T}_1 \cap \mathcal{T}_2$ and an estimator $\widehat{|\mathcal{S}|}^{(1D)}$ based on a sketch of size m (such as FM or bitmap sketch),*

$$\mathrm{E}\left(\frac{\widehat{|S|}^{(1D)}}{|S|} - 1\right)^2 = O(m^{-1}p_{\mathcal{S}}^{-2}),$$

where $p_{\mathcal{S}} = |\mathcal{S}|/N$. In particular, if a continuous FM sketch (Algorithm 1) is used here, then as $m \to \infty$,

$$\sqrt{m}\left(\frac{\widehat{|S|}^{(1D)}}{|S|} - 1\right) \xrightarrow{d} Normal\left(0, \alpha p_{\mathcal{S}}^{-2}\right),$$

where α is given in Equation 20 in the appendix.

PROOF. A simple heuristic is that if $|\mathcal{T}_j|$ are known, i.e. $\widehat{|\mathcal{T}_j|} = |\mathcal{T}_j|$, then $\widehat{|\mathcal{S}|}^{(1D)}$ can be improved and $\mathrm{var}(\widehat{|\mathcal{T}_1 \cap \mathcal{T}_2|}) =$

$\text{var}(|\widehat{\mathcal{T}_1 \cup \mathcal{T}_2}|)$, i.e., $O(m^{-1}N^2)$. Below is a formal proof. Let $\epsilon_j = \sqrt{m}(\frac{|\widehat{\mathcal{T}_j}|}{|\mathcal{T}_j|} - 1)$, $j = 1, 2$ and $\epsilon_3 = \sqrt{m}(\frac{|\widehat{\mathcal{T}_1 \cup \mathcal{T}_2}|}{|\mathcal{T}_1 \cup \mathcal{T}_2|} - 1)$, then $\text{var}(\epsilon_j) = O(1)$. Let $p_j = \frac{|\mathcal{S}|}{|\mathcal{T}_j|}$, $j = 1, 2$. We have $\frac{\widehat{|\mathcal{S}|}^{(1D)}}{|\mathcal{S}|} - 1 = m^{-1/2}\left(\sum_{j=1}^{2} p_j^{-1}\epsilon_j + p_{\mathcal{S}}^{-1}\epsilon_3\right)$. Using the fact that for any two random variables ξ, η, $\text{var}(\xi) = \text{E}(\text{var}(\xi|\eta)) + var(\text{E}(\xi|\eta)) \geq \text{E}(\text{var}(\xi|\eta))$, we have

$$var\left(\frac{\widehat{|\mathcal{S}|}^{(1D)}}{|\mathcal{S}|}\right) \geq \frac{1}{m} p_{\mathcal{S}}^{-2} \text{E}\left(var\left(\epsilon_3|\epsilon_1, \epsilon_2\right)\right).$$

Notice that $\text{E}(var(\epsilon_3|\epsilon_1, \epsilon_2)) = 0$ if and only if ϵ_3 is a function of ϵ_1, ϵ_2 with probability 1, or equivalently, $|\widehat{\mathcal{T}_1 \cup \mathcal{T}_2}|$ is a function of $|\widehat{\mathcal{T}_1}|$ and $|\widehat{\mathcal{T}_2}|$. But this can not be true, because two marginal cardinalities are not sufficient for predicting the union cardinality. Thus $var\left(\frac{\widehat{|\mathcal{S}|}^{(1D)}}{|\mathcal{S}|}\right) = O(\frac{1}{m} p_{\mathcal{S}}^{-2})$. The first result follows. The proof of the second result is given in the appendix. □

5.2 Two-level Hash Method

The 2-level hash method is proposed first by [13] for estimating the cardinality of a set expression over update streams, which allow item deletion as well as insertion. The 2-level hash sketch has a hierarchical structure with two levels: after generating a string of $\log N$ binary bits for each distinct item, the first level uses the original FM sketch with $O(\log N)$ bits to store the location of the least-significant bit (LSB) in the string; for each value of LSB, the second level uses an array of size $O(\log N)$ to count the number of 1s in each bit using strings with the given LSB value. As a result, the 2-level hash sketch for a stream requires $O(\log^3 N)$ storage per bucket. This is improved in [12] by reducing the space requirement for the second level hash to $O(\log N)$ per bucket instead. The novelty of the 2-level hash is that it allows one to identify buckets with a single distinct item for general set expressions. It has been shown by [13] that the mean square error for their estimate is $O(m^{-1}p_{\mathcal{S}}^{-1})$, the same order as our proportional-union method. However, their per-bucket storage is significantly higher: $O(\log^2 N)$ instead for the $O(\log\log N)$ required by our method. Intuitively, the 2-level hash method is quite inefficient because the estimate is solely based on the information regarding to those buckets that contain a single distinct element.

5.3 Performance Comparison

Table 1 summarizes the analytic performance comparison for all three methods, LLog-bitmap [6], 2-level-hash [12], and our proporitional-union method, in terms of relative standard error and memory units. To be specific, we also give a concrete example of the required memory for estimating the cardinality of $\mathcal{S} = \mathcal{T}_1 \cup \mathcal{T}_2$ when $|\mathcal{S}| = 10^5$, $p^{-1} = N/|\mathcal{S}| = 50$. To achieve a relative standard error 0.035, our proportional-union method requires 40 Kbytes, while the LLog-bitmap and 2-level-hash require 640 Kbytes and 7 Mbytes, respectively.

6. EXPERIMENTAL STUDY

In this section, we present empirical studies of our proportional union method and compare it with the two existing methods described in the previous section [19, 6, 13, 12]. The objective is to demonstrate further that our method significantly outperforms existing ones in practical scenarios. Three experiments are reported below. The first two are synthetic simulations for estimating the cardinalities of $|\mathcal{T}_1 \cap \mathcal{T}_2|$ and $|\mathcal{T}_1 \cap \mathcal{T}_2 \backslash \mathcal{T}_3|$. In the last experiment, we apply our proportional-union method to estimate the traffic matrix of the core network in a tier-1 service provider.

6.1 Synthetic Simulation: $|\mathcal{T}_1 \cap \mathcal{T}_2|$

Let N be the cardinality of the stream union, and p is the cardinality proportion of $|\mathcal{T}_1 \cap \mathcal{T}_2|$ in N. The simulation setup is as follows. To investigate the effect of the inverse proportion p^{-1} on the estimation accuracy, we fix the cardinality of the union at $N = 3 \cdot 10^7$, and let the inverse proportion p^{-1} vary from from 2 to 300. Therefore, as p^{-1} increases, $|\mathcal{T}_1 \cap \mathcal{T}_2|$ decreases. We choose $m = 10^5$ buckets for the proportional-union method, and allocate 15 bits for each bucket, where 10 bits are devoted for storing the decimals of the logarithm of the continuous FM sketch as described in Section 3.3. Therefore, the total memory is 1.5Mbits, or 188 Kbytes.

To complete the setup, we also let $|\mathcal{T}_1 \backslash \mathcal{T}_2| = r|\mathcal{T}_2 \backslash \mathcal{T}_1|$, and fix $r = 2$. This is done purely for convenience, since as indicated by Theorem 3, the performance of the proportional-union estimate would not be affected by the value of r. Now for each p^{-1}, we simulate $\mathcal{T}_1, \mathcal{T}_2$ for 100 times, and obtain estimate of $|\mathcal{T}_1 \cap \mathcal{T}_2|$ using four methods: LLog-bitmap by [6], 2-level-hash by [12], MLE and our proportional-union method. We do not implement the methods by [19, 13] since they are earlier versions of [6] and [12]. We allocate 5 bits per bucket for LLog-Bitmap, 800 bits per bucket for 2-level-hash (25 bits for the first level hash and 32 bits for each first level bit in the second level hash). The MLE uses the same sketches as the proportion-union method.

For each method, define the empirical value of the relative standard error from the 100 runs by

$$\widehat{\delta} = \sqrt{\frac{1}{100}\sum_{k=1}^{100}\left(relative.error^{(k)}\right)^2},$$

where $relative.error^{(k)}$ is the observed relative error of the estimate in each run. Figure 3 reports the value of $\widehat{\delta}$ as a function of the inverse proportion p^{-1} for the four methods, all of which use the same amount memory 188Kbytes. The results show that the proportional-union estimates and the MLE are very close to each other, and has a better performance comparing to the LLog-bitmap and 2-level-hash methods. For example, when $p^{-1} = 100$, the relative standard errors are 0.03 for both proportional-union and MLE, 0.10 for LLog-bitmap, and 0.52 for 2-level hash. The 2-level-hash method is the worst performer simply because the per-bucket storage is much higher than the rest of methods (only 250 bucket for 188 Kbytes memory). This is supported by Figure 4, where again we show $\widehat{\delta}$ as a function of p^{-1} for the four methods. However, this time, we kept the number of buckets the same at $m = 10^4$. Notice that even in this case, the 2-level-hash method is still a few factors worse than our proportional-union method, even though it is better than. LLog-bitmap.

By Theorem 3, 4, and 5, we expect a linear increase of the relative standard error as a function of p^{-1} for the LLog-bitmap method, and a square root increase for the other three methods. However, due to the statistical variability

Algorithm	Relative Standard error	Memory units	$\lvert\mathcal{T}_1 \cap \mathcal{T}_2\rvert = 10^5$, $p^{-1} = 50$, $\delta = 0.035$
Loglog-bitmap	$O(m^{-\frac{1}{2}}p^{-1})$	$\log\log(N)$ (bits)	640 Kbytes
2-level-hash	$O(m^{-\frac{1}{2}}p^{-\frac{1}{2}})$	$\log^2(N)$ (bits)	7 Mbytes
PU	$(mp)^{-\frac{1}{2}}$	$\log\log(N) + 10$ (bits)	40 Kbytes
MLE	$(mp(1+b))^{-\frac{1}{2}}$	$\log\log(N)$+10 (bits)	40 Kbytes

Table 1: Performance comparison of three methods: LLog-bitmap, 2-level-hash and our proportional-union, in terms of accuracy and memory requirement (m: number of buckets, $p = |\mathcal{S}|/N$: the proportion of the cardinality stream expression in total union, δ: relative standard error).

in the simulation, this relationship is sometimes obscure in Figure 3 and 4, for example in the case of the 2-level-hash method. Finally we note that since the memory requirement for the 2-level hash is $O(\log^2 N)$ but the others are $O(\log\log N)$, the relative performance of 2-level hash may become worse for larger N and better for smaller N.

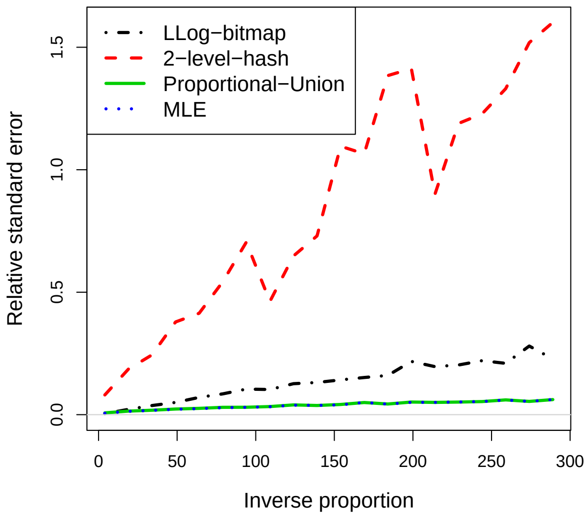

Figure 3: Relative standard errors of four methods for estimating $|\mathcal{T}_1 \cap \mathcal{T}_2|$ as a function of the inverse proportion $N/|\mathcal{T}_1 \cup \mathcal{T}_2|$ with $N = 3 \cdot 10^7$ and memory 188 Kbytes: the solid line is for proportional-union method, the dashed and dot-dashed lines are for the 2-level-hash and LLog-bitmap methods, and the MLE is in a dotted line and almost overlaps with the solid line for the proportional-union. The relative standard errors are computed based on 100 runs for each case.

6.2 Synthetic Simulation: $|\mathcal{T}_1 \cap \mathcal{T}_2 \backslash \mathcal{T}_3|$

The simulation setup is similar to the previous one. We fix $N = 3 \cdot 10^7$ and vary p^{-1} from 6 to 300. For the rest of subsets in the division of the total union (subsets labeled 1 to 6 in Figure 2), we let them have equal cardinalities. For each p^{-1}, we simulate $\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3$ for 100 times according to the specification, and obtain estimates of $|\mathcal{T}_1 \cap \mathcal{T}_2 \backslash \mathcal{T}_3|$ using three methods: LLog-bitmap, 2-level-hash and our proportional-union. Notice that we left out the MLE method since it cannot be easily obtained due to the complexity of the likelihood. The memory allocation for each method is the same as in the previous experiment over two streams (a total of 188 Kbytes).

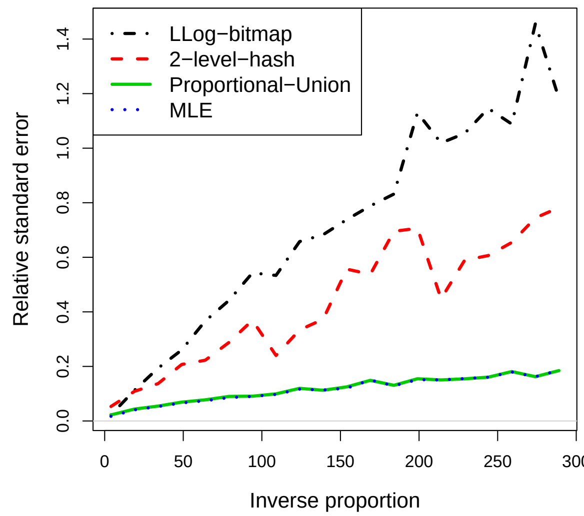

Figure 4: Relative standard errors of four methods for estimating $|\mathcal{T}_1 \cap \mathcal{T}_2|$ as a function of the inverse proportion $N/|\mathcal{T}_1 \cup \mathcal{T}_2|$ with fixed $N = 3 \cdot 10^7$ and array size $m = 10^4$: the solid line is for the proportional-union method, the dashed and dot-dashed lines are for the 2-level-hash and LLog-bitmap methods, and the MLE is in a dotted line and almost overlaps with the solid line for the proportional-union. The relative standard errors are computed based on 100 runs for each case.

Figure 5 reports the value of $\widehat{\delta}$ as a function of the inverse proportion p^{-1} for the three methods. The results show that the proportional-union estimates is better comparing to both the LLog-bitmap and 2-level hash methods. In fact, by comparing Figure 5 and 3, we notice that the performance of our proportional-union method is very similar under two scenarios for the same value of p^{-1}, which is consistent with our asymptotic results in Theorem 3.

Figure 4 also shows $\widehat{\delta}$ as a function of p^{-1}, but with a fixed number buckets $m = 10^4$. Same as the previous experiment, we conclude that our proportional-union method is the best and 2-level-hash method is the second if we do not consider per-bucket storage.

6.3 Network Traffic Matrix Estimation

In this study, we evaluate our proportional-union method in the traffic matrix estimation problem for the core network of a tier-1 service provider, where each element of traffic matrix is the number of distinct flows between an origin-destination (OD) pair in the network.

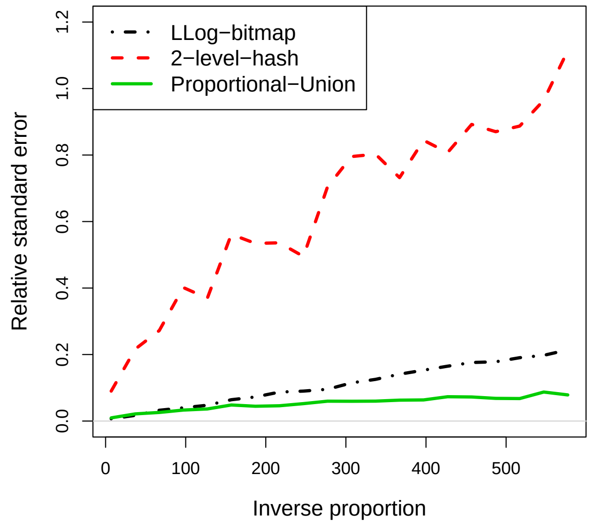

Figure 5: Relative standard errors of four methods for estimating $|\mathcal{S}| = |\mathcal{T}_1 \cap \mathcal{T}_2 \backslash \mathcal{T}_3|$ as a function of the inverse proportion $N/|\mathcal{S}|$, with $N = 3 \cdot 10^7$ and memory 188 Kbytes: the solid line is for the proportional-union method, the dashed and dot-dashed lines for the 2-level-hash and LLog-bitmap methods. The relative standard errors are computed based on 100 runs for each case.

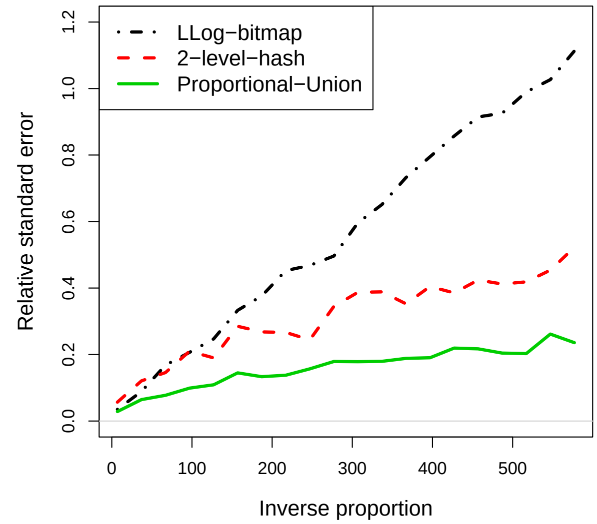

Figure 6: Relative standard errors of four methods for estimating $|\mathcal{S}| = |\mathcal{T}_1 \cap \mathcal{T}_2 \backslash \mathcal{T}_3|$, as a function of the inverse proportion $N/|\mathcal{S}|$, with $N = 3 \cdot 10^7$ and $m = 10^4$: the solid line is for the proportional-union method, the dashed and dot-dashed lines for the 2-level-hash and LLog-bitmap methods. The relative standard errors are computed based on 100 runs for each case.

Our traffic data from the tier-1 network provider includes the traffic volume in bytes on MPLS paths in every five minutes and the routing matrix for all MPLS paths that lists all the links on every path. From this, we can derive the OD traffic volume between any pair of origin and destination links by summing up the volume of the MPLS paths that traverse both links. However, the data from the service provider does not have any flow or packet level information, and therefore cannot be directly used as input to our algorithms.

In order to generate realistic flow level traces, we related some Internet packet traces to the byte volume matrix from the service provider as follows. We first preprocess the packet trace and group them at the flow level. For each MPLS path, we assign flows from the packet traces until the sum of the volume from all flows is no less than the traffic volume of the MPLS path from the traffic matrix. After this process, every flow is assigned to one MPLS path whose total volume is close to the value in the traffic matrix. We use the packet traces that are collected by [18]. There are about 1800 OD link pairs that contain OD traffic.

For each OD link pair, we apply our proportional-union method to estimate the OD flow counts in a 5-minute interval. We allocated a memory of 1M bits for the continuous FM sketch at each link with 15 bits per bucket as above, and compared the estimated OD flow counts for all link pairs to the ground truth.

Figure 7 reports the histogram of the inverse proportion p^{-1} of the OD flow counts in the union of OD flows where the five vertical lines show the 10%, 25%, 50%, 75% and 90% quantiles. For example, 50% of OD link pairs has a value beyond 64, 37% beyond 256 and 25% beyond 1024. By the synthetic simulation study for $|\mathcal{T}_1 \cap \mathcal{T}_2|$, there is a clear advantage of our method when the inverse proportion is beyond 256, and this advantage will only grow with larger values of p^{-1}.

Figure 8 shows the estimated OD flow counts with respect to the ground truth, where the two dashed lines show 10% off the ground truth values. Results show that the relative errors of our flow counts estimation for more than half of the OD pairs are less than 1%, and more than 90% OD pairs are within 10%.

7. CONCLUSION

In this paper, we have developed a simple proportional-union estimator for general stream expression cardinalities based on a variant of the well known Flajolet-Martin sketches. We have shown that for a set expression over two streams, it has almost the same statistical efficiency as the maximum likelihood estimator, which is the most efficient estimator achieving the Cramer-Rao lower bound asymptotically. For cardinalities defined over a larger number of streams, our proportional-union method is still simple to implement, but the MLE becomes intractable due to the complexity of the likelihood. We have demonstrated both analytically and by experimental studies, that our proportional-union estimator has a superior performance compared to state-of-the-art algorithms, especially for large N and small $|\mathcal{S}|/N$. Here $\mathcal{S}$ is the stream expression of interest and N is the cardinality of the union of the streams defining $\mathcal{S}$.

8. APPENDIX

8.1 Proof of Lemma 1

Let $B[k]$ denote the number of distinct items hashed into the kth bucket and let $U_1, \cdots, U_{B[k]}$ be the uniform random numbers associated with these items. Then $B[k]$ follows the

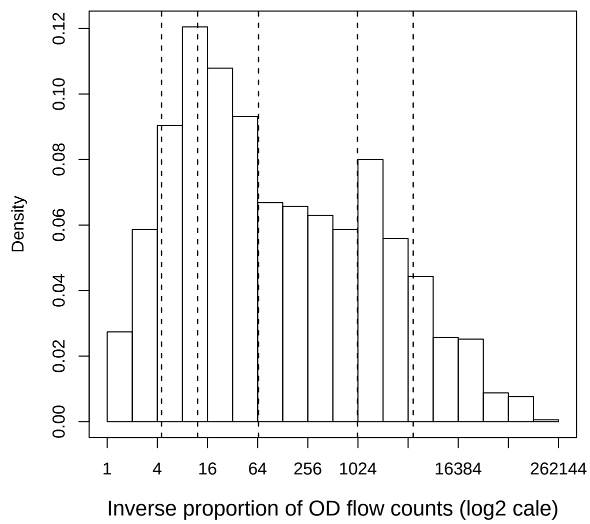

Figure 7: Histogram of the inverse proportion $p^{-1} = N/|\mathcal{T}_i \cap \mathcal{T}_j|$ for all link pairs (i, j) which have OD traffic.

Binomial distribution $Binomial(|\mathcal{T}|, \frac{1}{m})$ and $Y[k] = \min(1, U_1, \cdots, U_{B[k]})$. Thus for any $y \in [0, 1]$,

$$
\begin{aligned}
P(Y[k] \geq y) &= \sum_{i=0}^{|\mathcal{T}|} (1-y)^i P(B[k] = i) \\
&= \mathrm{E}\left[(1-y)^{B[k]}\right] \\
&= (1 - \frac{y}{m})^{|\mathcal{T}|} \approx e^{-\mu y},
\end{aligned}
$$

where the approximation is due to $(1 - \frac{y}{m})^{m/y} \approx e^{-1}$. Then the density function of $Y[k]$ at $y \in (0, 1)$ is approximately $f_\lambda(y) = \lambda e^{-\lambda y}$. This verifies the marginal distribution of $Y[k]$.

For the correlation structure of $Y[k], 1 \leq k \leq m$, notice that for any $1 \leq j \neq k \leq m$, $(B[j], B[k], |\mathcal{T}| - B[j] - B[k])$ follows a multinomial distribution $Multinomial(|\mathcal{T}|, (\frac{1}{m}, \frac{1}{m}, 1 - \frac{2}{m}))$. Using the Multinomial properties, we have

$$
\begin{aligned}
\mathrm{E}\left(Y[j]\right) &= \mathrm{E}\left[\frac{1}{1+B_j}\right] \\
&= \frac{m}{|\mathcal{T}|+1}(1 - (1 - \frac{1}{m})^{|\mathcal{T}|+1}) \qquad (18)
\end{aligned}
$$

and

$$
\begin{aligned}
\mathrm{E}\left(Y[j]Y[k]\right) &= \mathrm{E}\left[\frac{1}{(1+B_j)(1+B_k)}\right] \\
&= \frac{m^2}{(|\mathcal{T}|+1)(|\mathcal{T}|+2)} \\
&\quad \times \left[1 - 2(1 - \frac{1}{m})^{|\mathcal{T}|+2} + (1 - \frac{2}{m})^{|\mathcal{T}|+2}\right]
\end{aligned}
$$

Thus using $(1 - \frac{1}{m})^m \approx e^{-1}$, we have

$$
\begin{aligned}
\mathrm{cov}\left(Y[j], Y[k]\right) &= \mathrm{E}\left(Y[j]Y[k]\right) - \mathrm{E}\left(Y[j]\right) \cdot \mathrm{E}\left(Y[j]\right) \\
&\approx -\mu^{-2}(1 - e^{-\mu})m^{-1}[\mu^{-1} - 2e^{-\mu}].
\end{aligned}
$$

Similary it can be shown that $var(Y[j]) \approx \mu^{-2}(1 - e^{-\mu})$. Hence, $corr(Y[j], Y[k]) \approx -|\mathcal{T}|^{-1}(1 - 2\mu e^{-\mu})$.

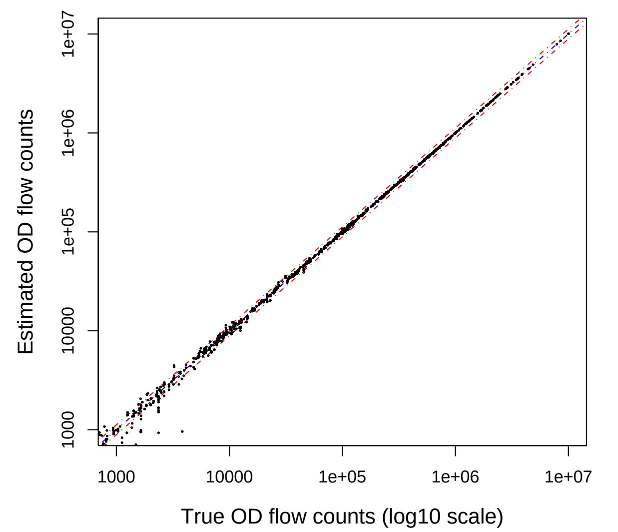

Figure 8: Estimation of OD flow counts by our proportional-union method.

8.2 Proof of Lemma 2

Let $\mu = |\mathcal{T}|/m$. Notice that

$$
\begin{aligned}
&\sqrt{m}\left(\frac{\widehat{|\mathcal{T}|}}{|\mathcal{T}|} - 1\right) \\
&= \frac{m^{-1/2}\sum_{k=1}^{m}\left(I(Y[k] < 1) - \mu Y[k]\right)}{\mu m^{-1}\sum_{k=1}^{m} Y[k]}.
\end{aligned}
$$

By the Law of Large Nunmbers, $m^{-1}\sum_{k=1}^{m} Y[k] \to E[Y]$, which by (18) is approximately equal to $\mu^{-1}(1 - e^{-\mu})$. Further, $\mathrm{E}\left(I(Y < 1) - \mu Y\right) \approx \frac{1}{|\mathcal{T}|}\left(1 - e^{-\mu}\right)$ and $\mathrm{E}\left(I(Y < 1) - \mu Y\right)^2 \approx 1 - e^{-\mu}$. Thus by the Central Limit Theorem and the Slusky Theorem, the numerator is approximately Gaussian with mean $m^{-1/2}\mu^{-1}(1 - e^{-\mu})$ and variance $1 - e^{-\mu}$. Thus for large m, $\sqrt{m}\left(\frac{\widehat{|\mathcal{T}|}}{|\mathcal{T}|} - 1\right)$ is approximately Gaussian with mean 0 and variance $\left(1 - e^{-\mu}\right)^{-1}$.

8.3 Proof of Theorem 3

Let $\epsilon_1 = \sqrt{m}\left(\frac{\hat{N}}{N} - 1\right)$ and $\epsilon_2 = \sqrt{m}\left(\frac{\hat{p}_S}{p_S} - 1\right)$. Assuming that $e^{-N/m} \approx 0$, by Theorem 3, $\epsilon_1 \approx Normal(0, 1)$ and by the Central Limit Theorem, (ϵ_1, ϵ_2) has a limit bivariate Gaussian distribution and $\epsilon_2 \approx N\left(0, \frac{1-p_S}{p_S}\right)$. Thus

$$
\begin{aligned}
\sqrt{m}\left(\frac{\widehat{|S|}^{(PU)}}{|S|} - 1\right) &= \sqrt{m}\left(\frac{\hat{N} \cdot \hat{p}_S}{N \cdot p_S} - 1\right) \\
&= \frac{\hat{p}_S}{p_S}\epsilon_1 + \epsilon_2.
\end{aligned}
$$

Using the fact that for two independent exponential random variables $\mathcal{E}_1, \mathcal{E}_2$ with rate λ_1, λ_2, the correlation between $\min(\mathcal{E}_1, \mathcal{E}_2)$ and $I(\mathcal{E}_1 = \min(\mathcal{E}_1, \mathcal{E}_2))$ is equal to 0, it can be shown that the correlation between ϵ_1 and ϵ_2 is ignorable. Note that $\hat{p}_S/p_S \approx 1$. By the Slusky Theorem, the limit distribution of $\frac{\hat{p}_S}{p_S}\epsilon_1 + \epsilon_2$ is asymptotic normal with mean 0 and variance $1 + \frac{1-p_S}{p_S} = \frac{N}{|S|}$.

8.4 Proof of Theorem 4

Let $\Lambda = \lambda_0 + \lambda_1 + \lambda_2$. After some calculation, one can show that the Fisher information matrix $\mathcal{I}(\theta) \doteq \frac{1}{m}\mathrm{E}(\mathcal{H}(\theta))$ is given with elements

$$\begin{cases} \mathcal{I}_{11} = (\frac{\lambda_1}{(\lambda_0+\lambda_2)^2} + \frac{\lambda_2}{(\lambda_0+\lambda_1)^2} + \frac{1}{\lambda})\Lambda^{-1} \\ \mathcal{I}_{22} = (\frac{1}{\lambda_1} + \frac{\lambda_2}{(\lambda_0+\lambda_1)^2}\Lambda^{-1} \\ \mathcal{I}_{33} = (\frac{1}{\lambda_2} + \frac{\lambda_1}{(\lambda_0+\lambda_2)^2})\Lambda^{-1} \\ \mathcal{I}_{12} = \mathcal{I}_{21} = \frac{\lambda_2}{(\lambda_0+\lambda_1)^2}\Lambda^{-1} \\ \mathcal{I}_{13} = \mathcal{I}_{31} = \frac{\lambda_1}{(\lambda_0+\lambda_2)^2}\Lambda^{-1} \\ \mathcal{I}_{23} = \mathcal{I}_{32} = 0. \end{cases}$$

From Fisher information theory [4], the MLE estimate of λ is asymptotically unbiased with a variance given by $(\mathcal{I}^{-1}(\theta))_{11}$. By inverting $\mathcal{I}$, we have

$$(\mathcal{I}^{-1})_{11} = \frac{\Lambda\lambda_0}{1+b},$$

where

$$b = \frac{\sigma_1}{\sigma_1\sigma_2 + (1+\sigma_2)^2} + \frac{\sigma_2}{\sigma_1\sigma_2 + (1+\sigma_1)^2} \qquad (19)$$

with $\sigma_j = \lambda_j\lambda^{-1}$, $j = 1, 2$. The result then follows.

8.5 Proof of Theorem 5 (cont)

Let $S = \mathcal{T}_1 \cap \mathcal{T}_2$. Let $p_S = \frac{|S|}{|\mathcal{T}_1 \cup \mathcal{T}_2|}$ and $p_j = \frac{|S|}{|\mathcal{T}_j|}$, $j = 1, 2$. Then the relative error w.r.t. the intersection cardinality estimator can be expressed as follows:

$$\sqrt{m}\left(\frac{\widehat{|S|}^{(1D)}}{|S|} - 1\right) = \sum_{j=1}^{2} p_j^{-1}\epsilon_j - p_S^{-1}\epsilon_\cup,$$

where σ_j ϵ_j denotes the relative error corresponding to $\widehat{|\mathcal{T}_j|}$, i.e. $\epsilon_j = \sqrt{m}\left(\frac{\widehat{|\mathcal{T}_j|}}{|\mathcal{T}_j|} - 1\right)$, and $\epsilon_\cup = \sqrt{m}\left(\frac{\widehat{|\mathcal{T}_1 \cup \mathcal{T}_2|}}{|\mathcal{T}_1 \cup \mathcal{T}_2|} - 1\right)$. By Lemma 2, when $e^{-|\mathcal{T}_j|/m}$ are ignorable, ϵ_j and $\epsilon_\cup$ approximately follow $Normal\,(0,1)$. Further calculation gives that

$$\mathrm{cov}(\epsilon_1, \epsilon_2) \approx p_S$$

and for $j = 1, 2$,

$$\mathrm{cov}(\epsilon_j, \epsilon_\cup) \approx p_S p_j^{-1}.$$

Thus $\sqrt{m}\left(\frac{\widehat{|S|}^{(1D)}}{|S|} - 1\right)$ approximately follows Normal with mean 0 and variance αp_S^{-2}, where

$$\alpha = 1 + \sum_{j=1}^{2} p_j^{-2} p_S^2 (1 - 2p_S^2) + 2p_S^{-1} p_1 p_2 > 0. \qquad (20)$$

9. REFERENCES

[1] 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 (STOC)*, pages 20–29, 1996.

[2] N. Alon and J. Spencer. *The Probabilistic Method*. Wiley, New York, 1992.

[3] Z. Bar-Yossef, T. Jayram, R. Kumar, D. Sivakumar, and L. Trevisan. Counting distinct elements in a data stream. In *In Proc. RANDOM '02*, 2002.

[4] P. Bickel and K. Doksum. *Mathematical Statistics: Basic Ideas and Selected Topics, Vol 1, (2nd Edition)*. Prentice Hall, 2000.

[5] A. Broder, M. Charikar, A. Frieze, and M. Mitzenmacher. Min-wise independent permutations. *Journal of Computer and System Sciences*, 60:630–659, 2000.

[6] M. Cai, J. Pan, Y.-K. Kwok, and K. Hwang. Fast and accurate traffic matrix measurement using adaptive cardinality counting. In *SIGCOMM'05 Workshops*, 2005.

[7] J. Considine, F. Li, G. Kollios, and J. Byers. Approximate aggregation techniques for sensor databases. In *ICDE '04: Proceedings of the 20th International Conference on Data Engineering*, page 449, Washington, DC, USA, 2004. IEEE Computer Society.

[8] M. Durand and P. Flajolet. Loglog counting of large cardinalities. In *Proceedings of European Symposium on Algorithms*, pages 605–617, 2003.

[9] C. Estan, G. Varghese, and M. Fisk. Bitmap algorithms for counting active flows on high speed links. *IEEE/ACM Trans. on Networking*, 14(5):925–937, 2006.

[10] P. Flajolet and G. Martin. Probabilistic counting algorithms for database applications. *Journal of Computer and System Sciences*, 31(2):182–209, 1985.

[11] P. Flajolet and G. N. Martin. Probabilistic counting. In *In Proc. Symp. on Foundations of Computer Science (FOCS)*, 1983.

[12] S. Ganguly. Counting distinct items over update streams. In *ISAAC*, pages 505–514, 2005.

[13] S. Ganguly, M. Garofalakis, and R. Rastogi. Tracking set-expression cardinalities over continuous update streams. *The VLDB Journal*, 13:354–369, 2004.

[14] M. Garofalakis, J. M. Hellerstein, and P. Maniatis. Proof sketches: Verifiable in-network aggregation. In *Proceedings of the IEEE 23rd International Conference on Data Engineering*, Istanbul, Turkey, 2007.

[15] P. Gibbons. Distinct sampling for highly-accurate answers to distinct values queries and event reports. *The VLDB Journal*, pages 541–550, 2001.

[16] F. Giroire. Order statistics and estimating cardinalities of massive data sets. In C. Martínez, editor, *2005 International Conference on Analysis of Algorithms*, pages 157–166. Discrete Mathematics and Theoretical Computer Science, 2005.

[17] P. Indyk. A small approximately min-wise independent family of hash functions. In *Proceedings of the 10th annual ACM-SIAM symposium on discrete algorithms*, Baltimore, MD, 1999.

[18] NLNAR. www.nlanr.net.

[19] Q. G. Zhao, A. Kumar, J. Wang, and J. J. Xu. Data streaming algorithms for accurate and efficient measurement of traffic and flow matrices. In *Proceedings of the ACM SIGMETRICS*, pages 350–361, 2005.