Phase Transition in Count Approximation by Count-Min Sketch with Conservative Updates

Éric Fusy, Gregory Kucherov

Phase transition in count approximation by Count-Min sketch with conservative updates

Éric Fusy and Gregory Kucherov[0000−0001−5899−5424]

LIGM, CNRS, Univ. Gustave Eiffel, Marne-la-Vallée, France
{Eric.Fusy,Gregory.Kucherov}@univ-eiffel.fr

Abstract. Count-Min sketch is a hashing-based data structure to represent a dynamically changing associative array of counters. Wse analyse the counting version of Count-Min under a stronger update rule known as *conservative update*, assuming the uniform distribution of input keys. We show that the accuracy of conservative update strategy undergoes a phase transition, depending on the number of distinct keys in the input as a fraction of the size of the Count-Min array. We prove that below the threshold, the relative error is asymptotically $o(1)$ (as opposed to the regular Count-Min strategy), whereas above the threshold, the relative error is $\Theta(1)$. The threshold corresponds to the peelability threshold of random k-uniform hypergraphs. We demonstrate that even for small number of keys, peelability of the underlying hypergraph is a crucial property to ensure the $o(1)$ error. To our knowledge, this relationship has not been observed previously. Finally, we provide experimental data on the behavior of the average error for Zipf's distribution compared with the uniform one.

1 Introduction

Count-Min sketch is a hash-based data structure to represent a dynamically changing associative array $\boldsymbol{a}$ of counters in an approximate way. The array $\boldsymbol{a}$ can be seen as a mapping from some set K of keys to $\mathbb{N}$, where K is drawn from a (large) universe U. The goal is to support *point queries* about the (approximate) current value of $\boldsymbol{a}(p)$ for a key p. Count-Min is especially suitable for the streaming framework, when counters associated to keys are updated dynamically. That is, *updates* are (key,value) pairs (p, ℓ) with the meaning that $\boldsymbol{a}(p)$ is updated to $\boldsymbol{a}(p) + \ell$.

Count-Min sketch was proposed in [13], see e.g. [11] for a survey. A similar data structure was introduced earlier in [10] named *Spectral Bloom filter*, itself closely related to *Counting Bloom filters* [20]. The difference between Count-Min sketch and Spectral Bloom filter is marginal: while a Count-Min sketch requires hash functions to have disjoint codomains (rows of Count-Min matrix), a Spectral Bloom filter has all hash functions mapping to the same array. This difference is the same as between partitioned [2] and regular Bloom filters. In this paper, we will deal with the Spectral Bloom filter version but will keep the term Count-Min sketch as more common in the literature.

Count-Min sketch supports negative update values ℓ provided that at each moment, each counter $\boldsymbol{a}(p)$ remains non-negative (so-called *strict turnstile model* [27]). When updates are positive, the Count-Min update algorithm can be modified to a stronger version leading to smaller errors in queries. This modification, introduced in [18] as *conservative update*, is mentioned in [11], without any formal analysis given in those papers. This variant is also discussed in [10] under the name *minimal increase*, where it is claimed that it decreases the probability of a positive error by a factor of the number of hash functions, but no proof is given. We discuss this claim in the concluding part of this paper.

The case of positive updates is widespread in practice. In particular, a very common instance is *counting* where all update values are 1. This task occurs in different scenarios in network traffic monitoring, as well as other applications related to data stream mining [18]. In bioinformatics, we may want to maintain, on the fly, multiplicities of fixed-length words occurring in a big sequence dataset [29, 3, 33]. We refer to [16] for more examples of applications.

While it is easily seen that the error in conservative update can only be smaller than in Count-Min, obtaining more precise bounds is a challenging problem. Count-Min guarantees, with high probability, that the additive error can be bounded by $\varepsilon\|\boldsymbol{a}\|_1$ for any ε, where $\|\boldsymbol{a}\|_1$ is the $L1$-norm of $\boldsymbol{a}$ [13]. In the counting setting, $\|\boldsymbol{a}\|_1$ is the length of the input stream which can be very large, and therefore this bound provides a weak guarantee in practice, unless the distribution of keys is very skewed and queries are made on frequent keys (*heavy hitters*) [27, 8, 12]. It is therefore an important practical question to analyse the improvement provided by the conservative update strategy compared to the original Count-Min sketch.

To our knowledge, the first attempt towards this goal was made in [7], under assumption that all $\binom{n}{k}$ counter combinations are equally likely at each step (n size of the Count-Min array, k number of hash functions) which amounts to assuming uniform distribution on $\binom{n}{k}$ input keys, each hashed to a distinct combination of counters. Thus, the number of distinct keys in the input is assumed to be much larger than the sketch size n. It was observed that the behavior of this model with uniformly distributed keys has important implications to non-uniformly distributed input. Another approach to bounding the error proposed in [16] is based on a simulation of spectral Bloom filters by a hierarchy of ordinary Bloom filters. However, the bounds provided are not explicit but are expressed via a recursive relation based on false positive rates of involved Bloom filters. Recent works [6, 5] propose formulas for computing error bounds depending on key probabilities assumed independent but not necessarily uniform, in particular leading to an improved precision bounds for detecting heavy hitters.

In this paper, we provide a probabilistic analysis of the conservative update scheme for counting under the assumption of *uniform distribution of keys* in the input. Our main result is a demonstration that the error in count estimates undergoes a phase transition when the number of distinct keys grows relative to the size of the Count-Min array. We show that the phase transition threshold corresponds to the *peelability threshold* for random k-uniform hypergraphs. For

the *subcritical regime*, when the number of distinct keys is below the threshold, we show that the relative error for a randomly chosen key tends to 0 asymptotically, with high probability. This contrasts with the regular Count-Min algorithm producing a relative error shown to be at least 1 with constant probability.

For the *supercritical regime*, we show that the average relative error is lower-bounded by a constant (depending on the number of distinct keys), with high probability. We prove this result for $k = 2$ and conjecture that it holds for arbitrary k as well. We provide computer simulations showing the growth of the expected relative error after the threshold, with a distribution showing a peculiar multi-modal shape. In particular, keys with small (or zero) error still occur after the threshold, but their fraction quickly decreases when the number of distinct keys grows.

After defining Count-Min sketch and conservative update strategy in Section 2 and introducing hash hypergraphs in Section 3, we formulate the conservative update algorithm (or regular Count-Min, for that matter) in terms of a hypergraph augmented with counters associated to vertices. In Section 4, we state our main results and illustrate them with a series of computer simulations. In Section 5 we outline the proof of our main result for the subcritical regime and provide a proof for the supercritical regime.

In addition, in Section 6, we study a specific family of 2-regular k-hypergraphs that are sparse but not peelable. For such graphs we show that while the relative error of every key is 1 with the regular Count-Min strategy, it is $1/k + o(1)$ for conservative update. While this result is mainly of theoretical interest, it illustrates that the peelability property is crucial for the error to be asymptotically vanishing. Finally, in Section 7, we turn to non-uniform distributions and provide a brief experimental analysis of the behavior of the average error for Zipf's distribution compared with the uniform one. Missing full proofs and additional experimental data can be found in [22].

2 Count-Min and Conservative Update

We consider a (counting version of) Count-Min sketch to be an array A of size n of counters initially set to 0, together with hash functions $h_1, \ldots, h_k$ mapping keys from a given universe to $[1..n]$. To count key occurrences in a stream of keys, regular Count-Min proceeds as follows. To process a key p, each of the counters $A[h_i(p)]$, $1 \leq i \leq k$, is incremented by 1. Querying the occurrence number $\boldsymbol{a}(p)$ of a key p returns the estimate $\hat{\boldsymbol{a}}_{CM}(p) = \min_{1 \leq i \leq k}\{A[h_i(p)]\}$. It is easily seen that $\hat{\boldsymbol{a}}_{CM}(p) \geq \boldsymbol{a}(p)$. A bound on the overestimate of $\boldsymbol{a}(p)$ is given by the following result adapted from [13].

Theorem 1 ([13]). *For $\varepsilon > 0$, $\delta > 0$, consider a Count-Min sketch with $k = \lceil \ln(\frac{1}{\delta}) \rceil$ and size $n = k\frac{e}{\varepsilon}$. Then $\hat{\boldsymbol{a}}_{CM}(p) - \boldsymbol{a}(p) \leq \varepsilon N$ with probability at least $1 - \delta$, where N is the size of the input stream.*

While Theorem 1 is useful in some situations, it has a limited utility as it bounds the error with respect to the stream size which can be very large.

Conservative update strengthens Count-Min by increasing only the smallest counters among $A[h_i(p)]$. Formally, for $1 \leq i \leq k$, $A[h_i(p)]$ is incremented by 1 if and only if $A[h_i(p)] = \min_{1\leq j\leq k}\{A[h_j(p)]\}$ and is left unchanged otherwise. The estimate of $\boldsymbol{a}(p)$, denoted $\hat{\boldsymbol{a}}_{CU}(p)$, is computed as before: $\hat{\boldsymbol{a}}_{CU}(p) = \min_{1\leq i\leq k}\{A[h_i(p)]\}$. It can be seen that $\hat{\boldsymbol{a}}_{CU}(p) \geq \boldsymbol{a}(p)$ still holds, and that $\hat{\boldsymbol{a}}_{CU}(p) \leq \hat{\boldsymbol{a}}_{CM}(p)$. The latter follows from the observation that on the same input, an entry of counter array A under conservative update can never get larger than the same entry under Count-Min.

3 Hash hypergraphs and CU process

With a counter array $A[1..n]$ and hash functions $h_1, ..., h_k$ we associate a k-uniform *hash hypergraph* $H = (V, E)$ with vertex-set $V = \{1, ..., n\}$ and edge-set $E = \{\{h_1(p), ..., h_k(p)\}\}$ for all distinct keys p. Let $\mathcal{H}^k_{n,m}$ be the set of k-uniform hypergraphs with n vertices and m edges. We assume that the hash hypergraph is a uniformly random Erdős-Rényi hypergraph in $\mathcal{H}^k_{n,m}$, which we denote by $H^k_{n,m}$, where m is the number of distinct keys in the input (for $k = 2$, we use the notation $G_{n,m} = H^2_{n,m}$). Even if this property is not granted by hash functions used in practice, it is a reasonable and commonly used hypothesis to conduct the analysis of sketch algorithms.

Below we show that the behavior of a sketching scheme depends on the properties of the associated hash hypergraph. It is well-known that depending on the m/n ratio, many properties of Erdős-Rényi (hyper)graphs follow a phase transition phenomenon [21]. For example, the emergence of a giant component, of size $O(n)$, occurs with high probability (hereafter, *w.h.p.*) at the threshold $\frac{m}{n} = \frac{1}{k(k-1)}$ [26].

Particularly relevant to us is the *peelability* property. Let $H = (V, E)$ be a hypergraph. The peeling process on H is as follows. We define $H_0 = H$, and iteratively for $i \geq 0$, we define V_i to be the set of leaves (vertices of degree 1) or isolated vertices in H_i, E_i to be the set of edges of H_i incident to vertices in V_i, and H_{i+1} to be the hypergraph obtained from H_i by deleting the vertices of V_i and the edges of E_i. A vertex in V_i is said to have *peeling level* i. The process stabilizes from some step I, and the hypergraph H_I is called the *core* of H, which is the largest induced sub-hypergraph whose vertices all have degree at least 2. If H_I is empty, then H is called *peelable*.

It is known [30] that peelability undergoes a phase transition. For $k \geq 3$, there exists a positive constant λ_k such that, for $\lambda < \lambda_k$, the random hypergraph $H^k_{n,\lambda n}$ is w.h.p. peelable as $n \to \infty$, while for $\lambda > \lambda_k$, the core of $H^k_{n,\lambda n}$ has w.h.p. a size concentrated around αn for some $\alpha > 0$ that depends on λ. The first peelability thresholds are $\lambda_3 \approx 0.818$, $\lambda_4 \approx 0.772$, etc., λ_3 being the largest.

For $k = 2$, for $\lambda < 1/2$, w.h.p. a proportion $1 - o(1)$ of vertices are in trees of size $O(1)$, (and a proportion $o(1)$ of the vertices are in the core), while for $\lambda > 1/2$, the core size is w.h.p. concentrated around αn for $\alpha > 0$ that depends on λ [32].

We note that properties of hash hypergraphs determine the behavior of some other hash-based data structures, such as Cuckoo hash tables [31] and Cuckoo filters [19], Minimal Perfect Hash Functions and Static Functions [28], Invertible Bloom filters [24], and others. We refer to [34] for an extended study of relationships between properties of hash hypergraphs and some of those data structures. In particular, peelability is directly relevant to certain constructions of Minimal Perfect Hash Functions as well as to good functioning of Invertible Bloom filters. However, its relation to Count-Min sketches is less direct and has not been observed earlier.

The connection to hash hypergraphs allows us to reformulate the Count-Min algorithm with conservative updates as a process, which we call CU-process, on a random hypergraph $H^k_{n,m}$, where n, m, k correspond to counter array length, number of distinct keys, and number of hash functions, respectively. Let $H = (V, E)$ be a hypergraph. To each vertex v we associate a counter c_v initially set to 0. At each step $t \geq 1$, a *CU-process* on H chooses an edge $e = \{v_1, \ldots, v_k\} \in E$ in H, and increments by 1 those c_{v_i} which verify $c_{v_i} = \min_{1 \leq j \leq k} c_{v_j}$. For $t \geq 0$ and $v \in V$, $c_v(t)$ will denote the value of the counter c_v after t steps, and $o_e(t)$ the number of times edge $e \in E$ has been drawn in the first t steps. The counter $c_e(t)$ of an edge $e = \{v_1, \ldots, v_k\}$ is defined as $c_e(t) = \min_{1 \leq i \leq k} c_{v_i}(t)$. Clearly, for each t and each e, $o_e(t) \leq c_e(t)$. The *relative error* of e at time t is defined as $R_e(t) = \frac{c_e(t) - o_e(t)}{o_e(t)}$. The following lemma can be easily proved by induction on t.

Lemma 1. *Let $H = (V, E)$ be a hypergraph on which a CU-process is run. At every step t, for each vertex v, there is at least one edge e incident to v such that $c_e(t) = c_v(t)$.*

Observe that, when H is a graph ($k = 2$), Lemma 1 is equivalent to the property that vertex counters cannot have a strict local maximum, i.e., at every step t, each vertex v has at least one neighbour u such that $c_u(t) \geq c_v(t)$.

4 Phase transition of the relative error

4.1 Main results

Let $H = (V, E)$ be a hypergraph, $|V| = n, |E| = m$. Let $N \geq 1$. We consider two closely related models of input to perform the CU-process. In the *N-uniform model*, the CU process is performed on a random sequence of keys (edges in E) of length $N \cdot m$, each key being drawn independently and uniformly in E. In the *N-balanced model*, the CU-process is performed on a random sequence of length $N \cdot m$, such that each $e \in E$ occurs exactly N times, and the order of keys is random. In other words, the input sequence of keys is a random permutation of the multiset made of N copies of each key of E. Clearly, both models are very close, since the number of occurrences of any key in the N-uniform model is concentrated around N (with Gaussian fluctuations of order $\sqrt{N}$) as N gets large. For both models, we use the notation $c_v^{(N)} = c_v(Nm)$ for the resulting counter of $v \in V$, $o_e^{(N)} = o_e(Nm)$ for the resulting number of occurrences of

$e \in E$, $c_e^{(N)} = c_e(Nm)$ for the resulting counter of $e \in E$, and $R_e^{(N)} = R_e(Nm) = (c_e^{(N)} - o_e^{(N)})/o_e^{(N)}$ for the resulting relative error of e. In the N-balanced model, since each key $e \in E$ occurs N times, we have $R_e^{(N)} = (c_e^{(N)} - N)/N$.

Our main result is the following.

Theorem 2 (subcritical regime). *Let $k \geq 2$, and let $\lambda < \lambda_k$, where $\lambda_2 = 1/2$, and for $k \geq 3$, λ_k is the peelability threshold as defined in Section 3. Consider a CU-process on a random hypergraph $H^k_{n,\lambda n}$ under either N-uniform or N-balanced model, and consider the relative error $R_e^{(N)}$ of a random edge in $H^k_{n,\lambda n}$. Then $R_e^{(N)} = o(1)$ w.h.p., as both n and N grow*[1].

Note that with the regular Count-Min algorithm (see Section 2), in the N-balanced model, the counter value of a node v is $\tilde{c}_v^{(N)} = N \cdot \deg(v)$, and the relative error $\tilde{R}_e^{(N)}$ of an edge $e = (v_1, \ldots, v_k)$ is always (whatever $N \geq 1$) equal to $\min(\deg(v_1), \ldots, \deg(v_k)) - 1$, and is thus always a non-negative integer. For fixed $k \geq 2$ and $\lambda > 0$, and for a random edge e in $H^k_{n,\lambda n}$, the probability that all k vertices belonging to e have at least one incident edge apart from e converges to a positive constant $c(\lambda, k) = (1 - e^{-k\lambda})^k$. Therefore, $\tilde{R}_e$ is a nonnegative integer whose probability to be non-zero converges to $c(\lambda, k)$. Thus, Theorem 2 ensures that, for $\lambda < \lambda_k$, conservative updates lead to a drastic decrease of the error, from $\Theta(1)$ to $o(1)$.

For a given hypergraph $H = (V, E)$ with m edges, we define $\mathrm{err}_N(H) = \frac{1}{m} \sum_{e \in E} R_e^{(N)}$ the *average* error over the edges of H. Formally, Theorem 2 does not imply that $\mathrm{err}_N(H)$ is $o(1)$, as it might possibly happen that a small fraction of edges have very large errors, yielding $\mathrm{err}_N(H)$ larger than $o(1)$. However, we believe that this is not the case. From the previous remark, it follows that the error of an edge $e = (v_1, \ldots, v_k)$ is upper-bounded by $\min(\deg(v_1), \ldots, \deg(v_k)) - 1$. Since the expected maximal degree in $H^k_{n,\lambda n}$ grows very slowly with n, one can expect that any set of $o(n)$ edges should have a contribution $o(1)$ w.h.p.. This is also supported by experiments given in the next section.

Based on Theorem 2 and the above discussion, we propound that a phase transition occurs for the average error, in the sense that it is $o(1)$ in the subcritical regime $\lambda < \lambda_k$, and $\Theta(1)$ in the supercritical regime $\lambda > \lambda_k$, w.h.p.. Regarding the supercritical regime, we are able to show that this indeed holds for $k = 2$ in the N-balanced model.

Theorem 3 (supercritical regime, case $k = 2$). *Let $\lambda > 1/2$. Then there exists a positive constant $f(\lambda)$ such that, in the N-balanced model, $\mathrm{err}_N(G_{n,\lambda n}) \geq f(\lambda)$ w.h.p., as n grows*[2].

Our proof of Theorem 3 extends to $k \geq 3$ for $\lambda > \widetilde{\lambda_k}$, where $\widetilde{\lambda_k}$ is the threshold beyond which the giant component of $H^k_{n,\lambda n}$ has w.h.p. more edges than vertices.

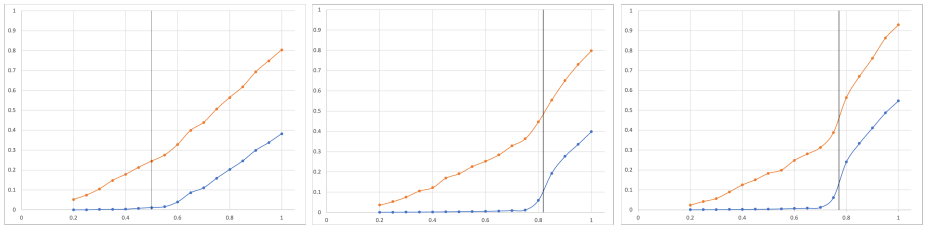
[1] Formally, for any $\epsilon > 0$, there exists M such that $\mathbb{P}(R_e^{(N)} \leq \epsilon) \geq 1 - \epsilon$ if $n \geq M$ and $N \geq M$.

[2] Formally, for any $\epsilon > 0$, there exists M such that $\mathbb{P}(\mathrm{err}_N(G_{n,m}) \geq f(\lambda)) \geq 1 - \epsilon$ if $N \geq 1$ and $n \geq M$.

The analysis given in [4] ensures that $\widetilde{\lambda_k}$ exists and is explicitly computable, $\widetilde{\lambda_3} \approx 0.94, \widetilde{\lambda_4} \approx 0.98$. We believe however that the peelability threshold λ_k constitutes the right critical value in Theorem 3 for $k \geq 3$ as well, which is supported by simulations presented below. Proving this would require a different kind of argument than we use in our proof though.

4.2 Simulations

Here we provide several experimental results illustrating the phase transition stated in Theorems 2 and 3. Figure 1 shows plots for the average relative error $\mathrm{err}_N(H^k_{n,m})$ as a function of $\lambda = m/n$, for $k \in \{2, 3, 4\}$ for regular Count-Min and the conservative update strategies. Experiments were run for $n = 1,000$ with the N-uniform model (each edge drawn independently with probability $1/m$) and $N = 50,000$ (number of steps $N \cdot |E|$). For each λ, an average is taken over 15 random graphs.



(a) $k = 2$ (b) $k = 3$ (c) $k = 4$

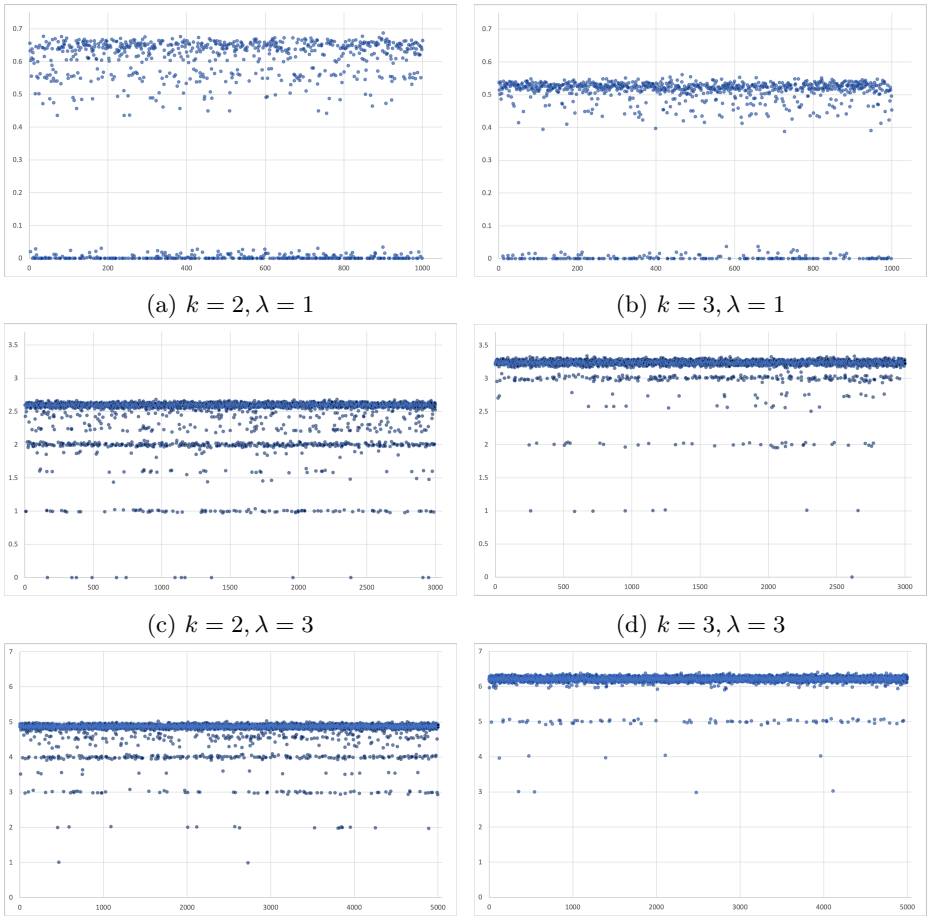
Fig. 1: Average relative error as a function of $\lambda = m/n$ for regular Count-Min (orange) and conservative update (blue), for $k \in \{2, 3, 4\}$. Vertical line shows the peelability threshold.

The phase transitions are clearly seen to correspond to the critical threshold 0.5 for $k = 2$, and, for $k \in \{3, 4\}$, to the peelability thresholds $\lambda_3 \approx 0.818$, $\lambda_4 \approx 0.772$. Observe that the transition looks sharper for $k \geq 3$, which may be explained by the fact that the core size undergoes a discontinuous phase transition for $k \geq 3$, as shown in [30] (e.g. for $k = 3$, the fraction of vertices in the core jumps from 0 to about 0.13).

For the supercritical regime, we experimentally studied the empirical distribution of individual relative errors, which turns out to have an interesting multimodal shape for intermediate values of λ. Typical distributions for $k \in \{2, 3\}$ are illustrated in Figure 2 where each point corresponds to an edge, and the edges are randomly ordered along the x-axis. Each plot corresponds to an individual random graph.

When λ grows beyond the peelability threshold, a fraction of edges with small errors still remains but vanishes quickly: these include edges incident to at least one leaf (these have error 0) and peelable edges (these have error $o(1)$),

as follows from our proof of Theorem 2. For intermediate values of λ, the distribution presents several modes: besides the main mode (largest concentration on plots of Figure 2), we observe a few other concentration values which are typically integers. While this phenomenon is still to be analysed, we explain it by the presence of certain structural graph motifs that involve disparities in node degrees. Note that the fraction of values concentrated around the main mode is dominant: for example, for $k = 3, \lambda = 3$ (Figure 2d), about 90% of values correspond to the main mode (≈ 3.22). Finally, when λ becomes larger, these "secondary modes" disappear, and the distribution becomes concentrated around a single value. More data about concentration is given in the full version [22].



(a) $k = 2, \lambda = 1$ (b) $k = 3, \lambda = 1$

(c) $k = 2, \lambda = 3$ (d) $k = 3, \lambda = 3$

(e) $k = 2, \lambda = 5$ (f) $k = 3, \lambda = 5$

Fig. 2: Distribution of relative errors of individual edges shown in a random order along x-axis.

Our analysis suggests that a positive average error in the supercritical regime is due to a large core — a non-peelable subgraph with $\Theta(n)$ nodes — which exists in this regime. To test this claim (for $k = 3$), we simulated the CU-process on sparse random non-peelable 3-hypergraphs, namely 2-regular 3-hypergraphs with $2n$ edges and $3n$ vertices (n parameter). These are sparsest possible non-peelable 3-hypergraphs, with degree 2 of each vertex. In a separate experiment, we observed that the average error for such graphs is concentrated around a constant value of ≈ 0.217. While these graphs fall to the subcritical regime ($\lambda = 2/3 < \lambda_3 \approx 0.818$), they still generate an average error bounded away from 0. Along with our results of Section 6, this supports that peelability is crucial for the error to be $o(1)$, and the presence of large non-peelable subgraphs results in a $\Theta(1)$ error.

5 Proofs of the main results

5.1 Sketch of proof of Theorem 2

Here we only provide main steps to show Theorem 2, the full proof is given in [22]. Theorem 2 relies on properties of random hypergraphs.

Case $k = 2$ corresponds to Erdős-Rényi random graphs $G_{n,\lambda n}$ [17] which have been extensively studied [21]. In particular, it is well known that when $\lambda < 1/2$ and n gets large, $G_{n,\lambda n}$ is, w.h.p., a union of small connected components most of which are constant-size trees. That is, a random edge in $G_{n,\lambda n}$ is, w.h.p., in a tree of size $O(1)$. Thus, the proof amounts to showing that, for a fixed tree T and an edge $e \in T$, we have w.h.p. $R_e^{(N)} = o(1)$ (as N gets large), both in the N-uniform and in the N-balanced model (performed on T alone). Let m be the number of edges in T. We first give a proof for the N-uniform model. Since $o_e^{(N)}$ follows a $\mathrm{Bin}(Nm, 1/m)$ distribution, we have w.h.p. $o_e^{(N)}/N = 1 + o(1)$. Hence, it is enough to prove that, for each vertex $v \in T$, we have w.h.p. $c_v^{(N)}/N = 1 + o(1)$. The proof is done by induction on the peeling level i of v. If $i = 0$, then v is a leaf. Letting e be its incident edge, we have $c_v^{(N)} = o_e^{(N)}$, hence w.h.p. $c_v^{(N)}/N = 1 + o(1)$. To let the induction work for $i \geq 1$, we actually have to carry a stronger property. Namely, we show that for each $v \in T$, there exist absolute positive constants a_v, b_v such that, for any $N \geq 1$ and $x > 0$, we have

$$\mathbb{P}\Big(\max_{t\in[0..Nm]} |c_v(t) - t/m| \geq x\sqrt{N}\Big) \leq a_v \exp(-b_v x^2). \tag{1}$$

The proof of (1) for v at level 0 follows from the fact that, for each $e \in T$ (in particular, the one incident to v), $o_e^{(N)}$ follows a $\mathrm{Bin}(Nm, 1/m)$ distribution, so that one can apply Hoeffding's inequality combined with Doob's maximal martingale inequality. This yields (1) for v at level 0, where $a_v = 2$ and $b_v = 2/m$. For v at level $i \geq 1$, we have the property that there is an edge e incident to v such that all the other neighbors $v_1, \ldots, v_h$ of v (i.e., the neighbors not incident to e) have level smaller than i, and thus satisfy (1) by induction. We then have to check that $c_v(t)$ stays close to t/m for $t \in [0..Nm]$ (in the sense of (1)).

For the lower bound part, we use the fact that $c_v(t) \geq o_e(t)$, and that $o_e(t)$ stays close to t/m. For the upper bound part we use the following argument. Letting $d_v(t) = \max(c_{v_1}(t), \ldots, c_{v_h}(t))$, we can show that if $c_v(t_0) \geq d_v(t_0) + M$ at some time t_0, then there exists $t' \leq t_0$ such that $|o_e(t') - t'/m| \geq M/4$ or $|d_v(t') - t'/m| \geq M/4$ (the crucial point to establish this property, specific to the CU-process, is that in the regime where $c_v(t) > d_v(t)$, any step where $c_v(t)$ increases occurs when picking e). Since $o_e(t)$ and $d_v(t)$ stay close to t/m (for $d_v(t)$ we use induction on i, and the union bound), this property ensures that $c_v(t)$ is unlikely to exceed t/m. Given the lower bound part, $c_v(t)$ hence stays close to t/m (in the sense of (1)). Estimate (1) then guarantees that, in the N-uniform model, we have $|c_v^{(N)}/N - 1| \leq N^{-1/3}$ with probability exponentially close to 1. The same holds in the N-balanced model, by noting that the N-balanced model is the N-uniform model conditioned on the event that each edge is chosen N times, which occurs with a probability of order $N^{-m/2}$ (thus, any event that is almost sure with exponential rate in the N-uniform model is also almost sure with exponential rate in the N-balanced model).

The proof for $k \geq 3$ is analogous but requires some more ingredients. An additional difficulty is that, for $\lambda < \lambda_k$, a random edge e in $H^k_{n,\lambda n}$ may be in the giant component (if $\lambda \in (\frac{1}{k(k-1)}, \lambda_k)$). However, we rely on the fact that the peeling level of e is $O(1)$ w.h.p., and prove that for a vertex v of bounded level, we have $c_v^{(N)}/N = 1 + o(1)$ w.h.p. as $N \to \infty$, where the $o(1)$ term does not depend on the size of the giant component.

5.2 Proof of Theorem 3

The *excess* of a hypergraph H is $\mathrm{exc}(H) = |E| - |V|$.

Lemma 2. *Let $H = (V, E)$ be a k-uniform hypergraph. Then, for the N-balanced model, we have $\sum_{e \in E} R_e^{(N)} \geq \frac{1}{k} \mathrm{exc}(H)$.*

Proof. During the CU process, each time an edge is drawn, the counter of at least one of its extremities is increased by 1. Hence $\sum_{v \in V} c_v^{(N)} \geq N|E|$. Hence, with the notation $R_v^{(N)} := c_v^{(N)}/N - 1$, we have $\sum_{v \in V} R_v^{(N)} \geq \mathrm{exc}(H)$. Now, by Lemma 1, for each $v \in V$, there exists an edge e_v incident to v such that $c_{e_v}^{(N)} = c_v^{(N)}$ (if several incident edges have this property, an arbitrary one is chosen). Hence, $\sum_{v \in V} R_{e_v}^{(N)} \geq \mathrm{exc}(H)$. Note that, in this sum, every edge occurs at most k times (since it has k extremities), thus $\sum_{e \in E} R_e^{(N)} \geq \frac{1}{k}\mathrm{exc}(H)$. □

For $k = 2$ and $\lambda > 1/2$, it is known [32, Theorem 6] that there is an explicit constant $\tilde{f}(\lambda) > 0$ such that the excess of the giant component $G' = (V', E')$ of $G_{n,\lambda n}$ is concentrated around $\tilde{f}(\lambda)n$, with fluctuations of order $\sqrt{n}$. Thus, $\mathrm{exc}(G') \geq \frac{1}{2}\tilde{f}(\lambda)n$ w.h.p. as $n \to \infty$. Hence, by Lemma 2, w.h.p. as $n \to \infty$ (and for any $N \geq 1$), we have

$$\mathrm{err}_N(G_{n,\lambda n}) \geq \frac{1}{\lambda n} \sum_{e \in E'} R_e^{(N)} \geq \frac{1}{2\lambda n} \mathrm{exc}(G') \geq \frac{1}{4\lambda} \tilde{f}(\lambda) =: f(\lambda),$$

which implies Theorem 3.

6 Analysis for some non-peelable hypergraphs

Analysing the asymptotic behaviour of the relative error of the CU-process on arbitrary hypergraphs seems to be a challenging task, even if we restrict ourselves to N-uniform and N-balanced models, as we do in this paper. Based on simulations, we expect that, for a fixed connected k-hypergraph $H = (V, E)$, and for $v \in V$, we have $c_v^{(N)}/N = C_v + o(1)$ w.h.p. as $N \to \infty$, for an explicit constant $C_v \in [1, \deg(v)]$. Since the number of increments at each step lies in $[1..k]$, constants C_v must verify $1 \leq \frac{1}{|E|} \sum_{v \in V} C_v \leq k$, where $\frac{1}{|E|} \sum_{v \in V} C_v$ can be seen as the average number of increments at a step. If H is peelable, then Theorem 2 implies that this concentration holds, with $C_v = 1$. We expect that, if no vertex of H is peelable, and if H is "sufficiently homogeneous", then the constants C_v should be all equal to the same constant $C > 1$, and thus the relative error $R_e^{(N)}$ of every edge is concentrated around $C - 1 > 0$ w.h.p. as $N \to \infty$. This, in particular, is supported by an experiment reported at the end of Section 4.2.

In this Section, we show that this is the case for a family of regular hypergraphs which are very sparse ($O(\sqrt{|V|})$ edges) but have a high order (an edge contains $O(\sqrt{|V|})$ vertices). The *dual* of a hypergraph H is the hypergraph H' where the roles of vertices and edges are interchanged: the vertices of H' are the edges of H, and the edges of H' are the vertices of H so that an edge of H' corresponding to a vertex v of H contains those vertices that correspond to edges incident to v in H.

Here we consider the hypergraph K'_n dual to the complete graph K_n. It is a $(n-1)$-uniform hypergraph with n edges and $\binom{n}{2}$ vertices, all of degree 2, therefore no vertex is peelable. For a fixed $n \geq 3$, we consider a CU-process on K'_n, in the N-balanced model. Note that with regular Count-Min, the relative error of every edge is 1, since all vertices have degree 2 and $c_v^{(N)} = 2N$ for every vertex v of K'_n. We prove that with conservative updates, the relative error is reduced to a smaller constant $1/(n-1)$. The proof is omitted and can be found in [23].

Theorem 4. *For any fixed $n \geq 2$, in the N-uniform model (resp. in the N-balanced model), the counter of each vertex $v \in K'_n$ satisfies $c_v^{(N)}/N = n/(n-1) + o(1)$ w.h.p. as $N \to \infty$. Hence, the relative error $R_e^{(N)}$ of each edge e in K'_n satisfies $R_e^{(N)} = 1/(n-1) + o(1)$ w.h.p. as $N \to \infty$.*

7 Non-uniform distributions

An interesting and natural question is whether the phase transition phenomenon holds for non-uniform distributions as well. This question is of practical importance, as in many practical situations keys are not distributed uniformly. In

particular, Zipfian distributions often occur in various applications and are a common test case for Count-Min sketches [14, 7, 16, 9, 6]. We mention a recent learning-based variant of CountMin [25] (learning heavy hitters) and its study under a Zipfian distribution [1, 15].

In Zipf's distributions, key probabilities in descending order are proportional to $1/i^\beta$, where i is the rank of the key and $\beta \geq 0$ is the *skewness* parameter. Note that for $\beta = 0$, Zipf's distribution reduces to the uniform one. It is therefore a natural question whether the phase transition occurs for Zipf's distributions with $\beta > 0$.

One may hypothesize that the answer to the question should be positive, as under Zipf's distribution, frequent keys tend to have no error, as it has been observed in earlier papers [7, 6, 5]. On the other hand, keys of the tail of the distribution have fairly similar frequencies, and therefore might show the same behavior as for the uniform case.

However, this hypothesis does not hold. Figure 3 shows the behavior of the average error for Zipf's distributions with $\beta \in \{0.2, 0.5, 0.7, 0.9\}$ vs. the uniform distribution ($\beta = 0$). The average error is defined here as the average error of all keys weighted by their frequencies[3], i.e. $\mathrm{err}_N(H) = \frac{1}{mN} \sum_{e \in E} o_e^{(N)} \frac{c_e^{(N)} - o_e^{(N)}}{o_e^{(N)}} = \frac{1}{mN} \sum_{e \in E} (c_e^{(N)} - o_e^{(N)})$. In other words, $\mathrm{err}_N(H)$ is the expected error of a randomly drawn key from the entire input stream of length mN (taking into account multiplicities).

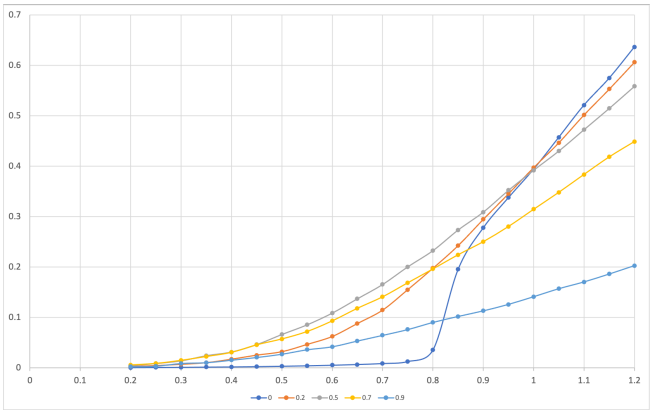


Fig. 3: Average error as a function of $\lambda = m/n$, for Zipf's distributions with $\beta \in \{0.0, 0.2, 0.5, 0.7, 0.9\}$. Plots obtained for $n = 1000$, $k = 3$, $N = 50,000$.

[3] This definition is natural for non-uniform distributions, as the error for a frequent key should have a larger contribution. Note that it is consistent with the definition of Section 4.1 in the N-balanced case, and in the N-uniform case it presents a negligible difference when N gets large.

We observe that the phase transition behavior disappears for $\beta > 0$. It turns out that even in the subcritical regime, frequent elements, while having no error themselves, heavily affect the error of certain rare elements, which raises the resulting average error. This phenomenon is analysed in more detail in our follow-up paper [23]. In the supercritical regime ($\lambda > 1$ in Figure 3) the opposite happens: the uniform distribution shows the largest average error. This is because an increasingly large fraction of the keys (those in the core of the associated hypergraph) contribute to the error, while for skewed distributions, frequent keys tend to have no error, and thus the larger β (with frequent keys becoming more predominant) the smaller the average error. Note that this is in accordance with the observation of [7] that the estimates for the uniform distribution majorate the estimates of infrequent keys for skewed distributions.

8 Concluding remarks

We presented an analysis of conservative update strategy for Count-Min sketch under the assumption of uniform distribution of keys in the input stream. Our results show that the behaviour of the sketch heavily depends on the properties of the underlying hash hypergraph. Assuming that hash functions are fully independent, the error produced by the sketch follows two different regimes depending on the density of the underlying hypergraph, that is the number of distinct keys relative to the size of the sketch. When this ratio is below the threshold, the conservative update strategy produces a $o(1)$ relative error when the input stream and the number of distinct keys both grow, while the regular Count-Min produces a positive constant error. This gap formally demonstrates that conservative update achieves a substantial improvement over regular Count-Min.

We showed that the above-mentioned threshold corresponds to the peelability threshold for k-uniform random hypergraphs. One practical implication of this is that the best memory usage is obtained with three hash functions, since λ_3 is maximum among all λ_k, and therefore $k = 3$ leads to the minimum number of counters needed to deal with a given number of distinct keys.

In [10] it is claimed, without proof, that the rate of positive errors of conservative update is k times smaller than that of regular Count-Min. This claim does not appear to be true. Note that Count-Min does not err on a key represented in the sketch if and only if the corresponding edge of the hypergraph includes a leaf (vertex of degree 1), while the conservative update can return an exact answer even for an edge without leaves. However, this latter event depends on the relative frequencies of keys and therefore on the specific distribution of keys and the input length. On the other hand, our experiments with uniformly distributed keys show that this event is relatively rare, and the rate of positive error for Count-Min and conservative update are essentially the same.

One important assumption of our analysis is the uniform distribution of keys in the input. We presented an experimental evidence that for skewed distributions, in particular for Zipf's distribution, the phase transition disappears when the skewness parameter grows. Therefore, the uniform distribution presents the

smallest error in the subcritical regime. The situation is the opposite in the supercritical regime when the number of distinct keys is large compared to the number of counters: here the uniform distribution presents the largest average error. As mentioned earlier, for Zipf's distribution, frequent keys have essentially no error, whereas in the supercritical regime, low frequency keys have all similar overestimates. This reveals another type of phase transition in error approximation for Zipf's distribution, occurring between frequent and infrequent elements, having direct application to accurate detection of *heavy hitters* in streams. We refer to our follow-up work [23] for further insights regarding this issue.

Acknowledgments We thank Djamal Belazzougui who first pointed out to us the conservative update strategy.

References

1. Aamand, A., Indyk, P., Vakilian, A.: (Learned) frequency estimation algorithms under Zipfian distribution. CoRR **abs/1908.05198** (2019)
2. Almeida, P.S.: A case for partitioned Bloom filters. CoRR **abs/2009.11789** (2020)
3. Behera, S., Gayen, S., Deogun, J.S., Vinodchandran, N.: Kmerestimate: A streaming algorithm for estimating k-mer counts with optimal space usage. In: Proc. ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics. pp. 438–447 (2018)
4. Behrisch, M., Coja-Oghlan, A., Kang, M.: Local limit theorems for the giant component of random hypergraphs. Comb. Probab. Comput. **23**(3), 331–366 (2014)
5. Ben Mazziane, Y., Alouf, S., Neglia, G.: Analyzing count min sketch with conservative updates. Computer Networks **217**, 109315 (2022)
6. Ben Mazziane, Y., Alouf, S., Neglia, G.: A formal analysis of the count-min sketch with conservative updates. CoRR **abs/2203.14549** (2022)
7. Bianchi, G., Duffy, K., Leith, D.J., Shneer, V.: Modeling conservative updates in multi-hash approximate count sketches. In: Proc. 24th International Teletraffic Congress, ITC 2012, Kraków, Poland, September 4-7, 2012. pp. 1–8. IEEE (2012)
8. Charikar, M., Chen, K., Farach-Colton, M.: Finding frequent items in data streams. Theoretical Computer Science **312**(1), 3–15 (2004)
9. Chen, P., Wu, Y., Yang, T., Jiang, J., Liu, Z.: Precise error estimation for sketch-based flow measurement. In: Proc. 21st ACM Internet Measurement Conference. p. 113–121 (2021)
10. Cohen, S., Matias, Y.: Spectral Bloom filters. In: Halevy, A.Y., Ives, Z.G., Doan, A. (eds.) Proc. 2003 ACM SIGMOD International Conference on Management of Data. pp. 241–252 (2003)
11. Cormode, G.: Count-min sketch. In: Liu, L., Özsu, M.T. (eds.) Encyclopedia of Database Systems, Second Edition. Springer (2018)
12. Cormode, G., Hadjieleftheriou, M.: Finding frequent items in data streams. Proceedings of the VLDB Endowment **1**(2), 1530–1541 (2008)
13. Cormode, G., Muthukrishnan, S.: An improved data stream summary: the count-min sketch and its applications. Journal of Algorithms **55**(1), 58–75 (Apr 2005)
14. Cormode, G., Muthukrishnan, S.: Summarizing and mining skewed data streams. In: Proc. 2005 SIAM International Conference on Data Mining, SDM 2005, Newport Beach, CA, USA, April 21-23, 2005. pp. 44–55 (2005)

15. Du, E., Wang, F., Mitzenmacher, M.: Putting the "learning" into learning-augmented algorithms for frequency estimation. In: Proc. 38th International Conference on Machine Learning, ICML 2021. vol. 139, pp. 2860–2869 (2021)
16. Einziger, G., Friedman, R.: A formal analysis of conservative update based approximate counting. In: International Conference on Computing, Networking and Communications, ICNC 2015. pp. 255–259 (2015)
17. Erdos, P., Rényi, A., et al.: On the evolution of random graphs. Publ. Math. Inst. Hung. Acad. Sci **5**(1), 17–60 (1960)
18. Estan, C., Varghese, G.: New directions in traffic measurement and accounting. In: Mathis, M., Steenkiste, P., Balakrishnan, H., Paxson, V. (eds.) Proc. ACM SIGCOMM 2002 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication. pp. 323–336. ACM (2002)
19. Fan, B., Andersen, D.G., Kaminsky, M., Mitzenmacher, M.D.: Cuckoo filter: Practically better than bloom. In: Proc. 10th ACM International on Conference on Emerging Networking Experiments and Technologies. p. 75–88 (2014)
20. Fan, L., Cao, P., Almeida, J., Broder, A.: Summary cache: a scalable wide-area web cache sharing protocol. IEEE/ACM Transactions on Networking **8**(3), 281–293 (2000)
21. Frieze, A., Karoński, M.: Introduction to Random Graphs. Cambridge University Press (2015)
22. Fusy, É., Kucherov, G.: Phase transition in count approximation by Count-Min sketch with conservative updates. CoRR **abs/2203.15496** (2022)
23. Fusy, É., Kucherov, G.: Count-min sketch with variable number of hash functions: an experimental study. CoRR **abs/2302.05245** (2023)
24. Goodrich, M.T., Mitzenmacher, M.: Invertible Bloom lookup tables. In: Proc. 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton). pp. 792–799. IEEE (2011)
25. Hsu, C., Indyk, P., Katabi, D., Vakilian, A.: Learning-based frequency estimation algorithms. In: Proc. 7th International Conference on Learning Representations, ICLR 2019 (2019)
26. Karoński, M., Łuczak, T.: The phase transition in a random hypergraph. Journal of Computational and Applied Mathematics **142**(1), 125–135 (2002)
27. Liu, H., Lin, Y., Han, J.: Methods for mining frequent items in data streams: an overview. Knowledge and information systems **26**(1), 1–30 (2011)
28. Majewski, B.S., Wormald, N.C., Havas, G., Czech, Z.J.: A family of perfect hashing methods. The Computer Journal **39**(6), 547–554 (1996)
29. Mohamadi, H., Khan, H., Birol, I.: ntCard: a streaming algorithm for cardinality estimation in genomics data. Bioinformatics **33**(9), 1324–1330 (2017)
30. Molloy, M.: Cores in random hypergraphs and boolean formulas. Random Structures & Algorithms **27**(1), 124–135 (2005)
31. Pagh, R., Rodler, F.F.: Cuckoo hashing. Journal of Algorithms **51**(2), 122–144 (2004)
32. Pittel, B., Wormald, N.C.: Counting connected graphs inside-out. Journal of Combinatorial Theory, Series B **93**(2), 127–172 (2005)
33. Shibuya, Y., Kucherov, G.: Set-Min sketch: a probabilistic map for power-law distributions with application to k-mer annotation. bioRxiv (2020), `https://www.biorxiv.org/content/10.1101/2020.11.14.382713v1`
34. Walzer, S.: Random hypergraphs for hashing-based data structures. Ph.D. thesis, Technische Universität Ilmenau, Germany (2020)