arXiv:2302.05245v2 [cs.DS] 7 Sep 2023

具有可变数量哈希函数的 Count-Min 草图：一项实验研究

Éric Fusy[1] 和 Gregory Kucherov[1]

[1]LIGM，CNRS，古斯塔夫·埃菲尔大学，马恩拉瓦莱，法国
{Eric.Fusy|Gregory.Kucherov}@univ-eiffel.fr

摘要

保守型 Count-Min 是广受欢迎的 Count-Min 草图的增强版本 [Cormode, Muthukrishnan 2005], ，是一种在线维护的基于哈希的草图，用于概括数据流中元素频率的信息。尽管已有多项研究尝试分析保守型 Count-Min 的误差，但其行为仍未得到充分理解。在 [Fusy、Kucherov 2022], 的研究中，我们证明了在输入元素服从均匀分布的情况下，保守型 Count-Min 的误差根据其负载因子呈现出两种不同的状态。

在本研究中，我们提出了一系列结果，为深入理解保守型 Count-Min 的行为提供了新的见解。我们的贡献有两方面。一方面，我们在不同状态下以及输入元素服从多种具有代表性的概率分布时，对 Count-Min 草图进行了详细的实验分析。另一方面，我们证明了通过为不同元素分配可变数量的哈希函数，可以实现性能改进。具体而言，这包括在仍能保持较小误差的同时，减少数据结构所占用的空间。

1 引言

从最一般的意义上说，*Count-Min* 草图 是一种数据结构，用于表示一个以从大型全集中选取的元素（键）为索引的数字关联数组。该数组通过一系列（键，值）更新构成的数据流提供，使得与某个键关联的当前值等于此前对该键进行的所有更新值之和。本文重点讨论的 Count-Min 最常见应用场景或许是计数 场景，其中所有更新值均为 $+1$。在这种情况下，某个键的值就是它的计数 , 表示该键在数据流中出现的次数。换言之，Count-Min 可以看作表示一个多重集，即将键的一个子集映射到非负整数。按照后一种解释，每次更新都称为插入。Count-Min 主要支持的查询是获取给定键的计数；返回的估计值可能并不精确，但只会高于真实计数。

Count-Min 的计数版本应用于与数据流挖掘和数据摘要相关的各种实际问题。例如，在数据流中跟踪频繁项（重频项） [26, 9, 13]。它也用于网络流量监控 [20], 缓存使用优化 [19]。此外，它还应用于非数据流的大数据应用，例如生物信息学 [28, 3, 32]。

Count-Min 使用哈希函数，但与经典哈希表不同，它不存储元素，只存储计数信息（因此称为草图）。它由 [14], 提出，不过早在 [11] 就有人提出了一个非常相似的数据结构，称为谱布隆过滤器。后者又

与计数布隆过滤器 [22]密切相关。在这项工作中，我们采用 [11]中的定义，但仍称之为 Count-Min，以与文献中通常采用的名称保持一致。关于 Count-Min 的综述可参见 [12]。

本文研究了 Count-Min 的一个更强版本，称为保守型。Count-Min 的这一改进由 [20] 提出，称为保守更新，参见 [12]。 [11] 也讨论了这一改进，并称之为最小增量。保守型 Count-Min 使用相同的内存，却能提供严格更精确的计数估计，因此严格优于原始版本。代价是无法处理删除操作（负更新）；而原始 Count-Min 也能处理删除操作，前提是累积计数保持非负（这一条件称为严格旋转门模型 [26]）。

分析保守型 Count-Min 的误差是一项困难的工作，其结果会直接影响实际应用。下文第 2.2 节将更详细地综述已知的相关结果。在我们先前的工作 [23], 中，我们通过研究其与随机超图之间的关系来探讨这一问题。我们特别证明，若数据结构所表示的元素在输入中均匀分布，则误差会呈现两种不同的规律，具体取决于底层可剥离性 这一性质，即哈希超图的可剥离性。随机超图的性质与某些数据结构密切相关，这一点早已为人所知（见第 2.3 节），但此前尚未发现它与 Count-Min 有关。

基于这些结果，本文从多个方面扩展并深化了这项分析，并通过实验验证支持我们的论点。我们的首要目标是细致分析保守型 Count-Min 的“解剖结构”，描述其在不同区间中的行为。我们的主要创新贡献在于证明：为不同输入元素分配不同数量的哈希函数，可以显著降低误差，从而节省内存。另一项重要扩展涉及输入元素的概率分布：我们也研究非均匀分布，尤其是阶梯分布和齐夫分布，并分析 Count-Min 在这些分布下的行为。这项分析之所以重要，不仅因为非均匀分布在实践中十分常见，还因为它能为 Count-Min 的主要应用——检测最高频元素（有时称为重频项 问题 [26, 9, 13]）——提供重要见解。具体而言，我们考虑“小内存区间”（按我们的术语，即超临界区间）：不同的表示元素数量远大于数据结构的规模，并分析在何种条件下，最高频元素的估计误差可忽略。这对频繁元素问题有直接应用。

作为引言的结尾，我们指出，分析所具有的实验性并不影响结果的普适性；这些结果适用于广泛的参数范围。这一点既源于所检验假设的一般性，也有基于先前研究的理论依据。

2 背景与相关工作

2.1 ConservativeCount-Min：定义

Count-Min 草图是一个大小为 n 的计数器数组 A，并配有一组哈希函数，将给定全集 U 中的元素（键）映射到 $[1..n]$。在本研究中，通常可以为每个元素 $e \in U$ 分配不同数量 k_e 的哈希函数。假设哈希函数是完全随机的，因此不失一般性，假设元素 e 被分配了哈希函数 h_1、.....、h_{k_e}。

初始化时，计数器 $A[i]$ 被设为 0。处理输入元素 e 的一次插入时，基本版 Count-Min 会将每个计数器 $A[h_i(e)]$、$1 \le i \le k_e$ 增加 1。Count-Min 的保守版本只将所有 $A[h_i(e)]$ 中最小的值增加 1。也就是说，当且仅当 $A[h_i(e)] =$ 等于 $_{1\le j\le k_e}\{A[h_j(e)]\}$ min 时，$A[h_i(e)]$ 才会增加 1，否则保持不变。

在两个版本中，查询元素 e 出现次数的估计值 由 $c(e) = \min_{1\le i\le k_e}\{A[h_i(e)]\}$ 计算得出。很容易看出，对于任意元素输入序列，原始 Count-Min 计算出的估计值大于或等于保守版本计算出的估计值。这是因为，对于相同的输入，计数器数组 A中采用保守更新的某个条目，永远不会大于 Count-Min 中对应的条目。

本文研究 Count-Min 的保守版本。令 H 表示一组选定的哈希函数 $H = \{h_1, h_2, \ldots\}$。考虑由 N 次插入操作构成的输入序列 I，并令 E为出现在 I中的不同元素集合。元素 e 的相对误差 定义为 $\mathrm{err}_{H,I}(e) = (c(e) - occ(e))/occ(e)$，其中 $occ(e)$ 是 e 在输入中的出现次数。组合错误是对 I 中所有元素的误差进行加权平均，权重为各元素的出现次数，即

$$\mathrm{err}_{H,I} = \frac{1}{N}\sum_{e\in E} occ(e)\cdot \mathrm{err}(e) = \frac{1}{N}\sum_{e\in E}(c(e) - occ(e)).$$

我们假设 I 是从元素集合 $E \subseteq U$上的某个概率分布中独立同分布抽取的随机序列。一个关键参数是 E 相对于 A的大小 n 。类比哈希表，$\lambda = |E|/n$ 称为负载因子，或简称负载。

2.2 保守型 Count-Min 的分析：先前研究

受流量监测应用的启发，[7] 可能是首篇致力于分析计数场景下保守型 Count-Min 的研究。他们的模型假设所有 $\binom{n}{k}$种计数器组合出现的概率均等，并对每个元素应用 k 个哈希函数。这意味着所处的情形为 $|E| \gg n$。[7] 重点分析计数器的增长率，即每次插入对应的平均计数器增量，所用方法基于马尔可夫链和微分方程。[19]提出了另一种方法：用一系列层级排列的普通布隆过滤器模拟保守型 Count-Min sketch。所得误差界并非显式形式，而是通过基于相应布隆过滤器假阳性率的递归关系表示。

近期研究 [5, 4] 提出了一种分析方法，用于计算取决于元素概率的误差界；这些概率假定彼此独立，但不一定均匀分布，尤其能够提高检测重频项时的精度界。然而，当所有元素的概率都较小时，这种技术的效率会受到更大限制。特别是，如果输入分布是均匀的，该方法并未比原始 Count-Min 已知的一般误差界带来任何改进。

在我们最近的工作中，[23], 我们基于保守型 Count-Min 与随机超图之间的关系，对保守型 Count-Min 进行了分析。我们将在第 2.4 节中总结这项工作的主要结果。

2.3 哈希超图

许多基于哈希的数据结构都天然地与哈希超图相关联，因此超图的性质直接关系到数据结构能否正常工作。布谷鸟哈希 [30] 、布谷鸟过滤器 [21], 、最小完美哈希函数和静态

函数 [27], 、可逆布隆查找表 [24], 以及其他一些结构。 [33]对哈希超图与其中一些数据结构之间的关系进行了更深入的研究。

一个 Count-Min 草图与一个哈希超图 $H = (V, E)$相关联，其中 $V = \{1..n\}$ 和 $E = \{\{h_1(e) ...h_{k_e}(e)\}\}$ 遍历所有互不相同的输入元素 e。对于具有n 个顶点和 m 条边的超图，我们采用记号 $\mathcal{H}_{n,\ m}$ ；对于 k均匀的此类超图，则采用记号$\mathcal{H}^k_{n,\ m}$ ，其中所有边的基数均为k。在后一种情况下，由于我们假设哈希函数是完全随机的，因此哈希超图是一个k均匀的 Erdős-Rényi 随机超图。

由于假设插入的元素取自某个随机分布，因此，将 Count-Min 草图的运行视为相关联的哈希超图[23]上的一个随机过程会很方便。每个顶点都持有一个初始值为零的计数器，因此每条边都关联着一组由相应顶点持有的计数器。插入一个元素，就是将对应边上最小的计数器递增；而检索某个元素的估计值，则返回对应边上计数器中的最小值。从现在起，在本文中，我们将交替使用相关联的哈希超图中的不同元素和边，以及计数器和顶点这两组说法。因此，我们将相应计数器的值称为*顶点值*，将相应元素的估计值称为*边值*。此外，我们还会讨论超图的*负载*，即其密度 $|E|/|V|$。

2.4 超图可剥离性与误差相变

若从 H 开始反复执行以下步骤，最终得到空图，则称超图 $H = (V, E)$ 为*可剥离*：如果图中存在度为 1 或 0 的顶点，就删除该顶点及与之关联的边（如果有）。与随机超图的许多其他性质一样，可剥离性也会发生相变。考虑 Erdős-Rényi k-均匀超图模型，其中图从 $\mathcal{H}^k_{n,\ m}$ 中均匀随机抽取。文献 [29] 证明，在（可计算的）可剥离性阈值 λ_k 处会发生相变：当 $\lambda < \lambda_k$ 时，从 $\mathcal{H}^k_{n,\ \lambda n}$ 中随机抽取的图以高概率（w.h.p.）可剥离；当 $\lambda > \lambda_k$ 时，以高概率不可剥离。前几个值为 $\lambda_2 = 0.5$、 $\lambda_3 \approx 0.818$、 $\lambda_4 \approx 0.772$ 等，其中 λ_3 最大。请注意， $k = 2$ 是可剥离性的一个例外：对于 $\lambda < \lambda_2$，剥离后仍会残留比例可忽略不计的顶点。

已知，可剥离性与某些最小完美哈希函数的构造直接相关 [27] ，也与可逆布隆过滤器 [24]的正常运行有关。在 [23], 中，我们证明它也与 Count-Min 有关。

定理 1 ([23])。 *考虑一种保守型 Count-Min，其中每个元素都使用 k 个随机哈希函数进行哈希。假设长度为 N 的输入 I 是从包含 $E \subseteq U$ 个元素的集合上的均匀 分布中抽取的，并令 $\lambda = |E|/n$，其中 n 为计数器的数量。若 $\lambda < \lambda_k$，则对于随机选取的元素 e，误差 $(c(e) - occ(e))/occ(e)$ 以 $o(1)$ 的概率发生，其中 n 和 N/n 均趋于无穷。*

在互补的情形 $\lambda > \lambda_k$ 下，我们在 [23], 中证明，在一些额外假设下，$\mathrm{err}_{H,I}$ 为 $\Theta(1)$。因此，对于输入服从均匀分布的情形，随机哈希超图的可剥离性阈值对应于保守型 Count-Min 所产生误差的相变。我们分别将 $\lambda < \lambda_k$ 和 $\lambda > \lambda_k$ *亚临界* 和 *超临界*情形称为相应区域。

2.5 可变数量的哈希函数：混合超图

最佳可剥离性阈值 $\lambda_3 \approx 0.818$ 至少可以通过两种不同方式加以提高。一种方法是使用经过精心定义的哈希函数类，以特定的非均匀采样取代对 k-边的均匀采样。因此， [17] 证明，使用一种特殊的超图类时，可将可剥离性阈值提高到 ≈ 0.918 （适用于 $k = 3$ ），对于较大的 k，则可提高到 ≈ 0.999 。

另一个有些出人意料的想法，也是我们在本文中采用的方法，是对不同元素使用不同数量的哈希函数，也就是说，考虑非均匀超图。Following [16],[31] 证明，非均匀超图的可剥离性阈值可能高于均匀超图。更具体地说， [31] 证明，包含两类基数不同且各自占所有边中固定比例的边的混合超图 ，可能具有更高的可剥离性阈值：例如，边中有 ≈ 0.887 的基数为 3，其余边的基数为 21 的超图，其可剥离性阈值为 ≈ 0.920，高于均匀超图所达到的最佳阈值 0。818。我们采用 [31] 关于混合超图的记号：写作 $k = (k_1, k_2)$ 表示该超图包含基数为 k_1 和 k_2的边，而 $k = (k_1, k_2; \alpha)$ 则进一步指定， k_1- 边所占比例为 α。

针对不同元素使用不同数量哈希函数的想法也出现在数据结构设计中。 [8] 提出了加权布隆过滤器 ，它根据元素被查询的频率以及元素属于该集合的概率，为元素应用不同数量的哈希函数。其思路是为查询频率较高、出现概率较低的元素分配更多哈希函数。研究表明，这样可以降低假阳性概率，其中假阳性概率按查询频率加权定义。 [34], 进一步完善了这一想法，随后 [6], 又将其进一步发展，并称之为*Daisy* 布隆过滤器。

3 结果

3.1 均匀分布

我们先讨论输入元素服从均匀分布的情形，即相关哈希超图中的边被处理以进行更新的概率相同。

3.1.1 亚临界区间

定理 1 与第 2.5 节的结果共同表明，针对不同元素使用不同数量的哈希函数，或许可以“扩展” Count-Min 草图的 $o(1)$误差区间；这一点可以严格表述（为简便起见，我们仅给出两种不同边基数的情况）。

定理 2 （将定理 1 推广到混合基数）。 考虑一个含有n 个计数器的保守型 *Count-Min*。假设长度为 N 的输入取自 $E \subseteq U$ 上的均匀分布，并令 $\lambda < \lambda_k$。进一步假设， E 中的元素按照混合超图模型$k =$ （k_1, k_2； α）进行哈希。令 c_k 为与 k相关联的可剥离率。那么，当 $\lambda < c_k$时，随机选取的键的错误率为 $o(1)$ ，以高概率成立；其中 n 和 N/n均趋于无穷大。

证明。 关键在于， [23] 的分析首先确立了一个确定性结果（扩展版中的引理 5），该结果可表述如下：在任意固定超图中，可剥离边 e 的错误率为 $o(1)$，且其收敛至 0 的速率由必须先于剥离 e 而剥离的边数 M_e 决定。这保证了，对于任意

随机超图模型，若超图中的随机边 e 以高概率可剥离，且 M_e 以高概率为 $O(1)$，则 e 的错误率为 $o(1)$。随后， [16] 中进行的分析表明，具有混合基数的随机超图模型在可剥离性阈值以下具有这一性质。

□

图 1 显示了三种超图的平均相对误差随负载因子变化的情况：2-均匀、3-均匀，以及混合超图，其中 0.885 的边的基数为 3，其余边的基数为 14。2-均匀和 3-均匀超图分别在趋近于 0.5 和 ≈ 0.818 的负载因子处呈现相变，这两个值分别是 2-均匀和 3-均匀超图的可剥离性阈值。显然可以看出，混合超图的相变发生在更大的值 ≈ 0.898 附近，该值是此类超图的可剥离性阈值 [31]。

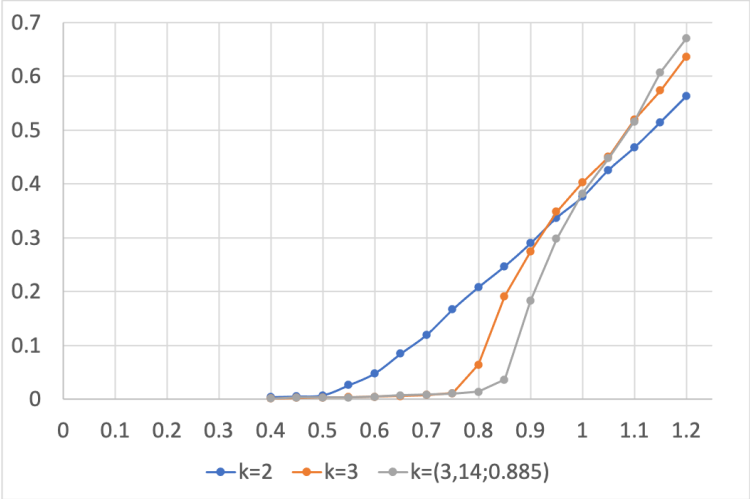


图 1：对于均匀分布和不同类型的超图，在较小 $\lambda = m/n$ 下的 $\mathrm{err}_{H,I}$ ： 2-均匀、3-均匀，以及 (3,14)-混合超图，其中 3-边所占比例为 0.885（参数取自 [31]）。数据由 $n = 1000$ 获得。每次实验的输入规模为边数的 5,000 倍。每个平均值均基于 10 个随机超图计算得出。

尽管这一结果可由结合 [31]和 [23], 的结果推导得出，但此前尚未有人发现，而且具有重要的实际意义：在 *Count-Min sketch* 中使用数量可变的哈希函数，可以在保持误差小到可忽略不计的同时提高负载因子。具体而言，对于相同的输入，与均匀情形相比，这样可以节省空间。

请注意，参数 $k = (3, 14; 0。885)$ 取自 [31]，以确保相变对应于 [31]中得到的可剥离性阈值。在实践中，可以选择更“简单”的参数；例如，我们发现 $k = (2, 5; 0。5)$ 得到的曲线与 $k = (3, 14; 0。885)$ 得到的曲线基本相同。

3.1.2 超临界区域

当负载因子变大（超临界区域）时，情况会发生显著变化。当负载因子刚刚超过阈值时，某些边的评估误差仍然很小或为零，而对于其他边，误差则会变得很大。这一“中间区域”已在

[23]中进行了说明。有趣的是，在该区域内，边值以一种非常特殊的方式分布，集中在若干个值附近（示例请参见 [23] 中的图 3）。这些值必定是由某些尚待阐明的图结构模式所决定的。

当负载因子进一步增大时，边值的多层模式会消失，所有边值都集中在同一个值附近。我们将这一现象称为饱和。例如，对于 $k = 3$，饱和大约发生在 $\lambda = 6$（数据未展示）。在这种状态下，哈希超图足够稠密，其具体拓扑结构很可能无关紧要，最高计数器层级会“渗透”到所有顶点计数器中。换言之，所有计数器都以相同的速率增长，不会有任何计数器因特定的图结构模式（例如边包含叶顶点）而“落后”。

Bianchi 等人 [7] 的分析假设每条 $\binom{n}{k}$ 边在每一步被处理的概率相同。这模拟了负载因子非常大且超图已饱和的情形。 [7] 关注的是增长率，即处理一条边时计数器增量的期望值。显然，这个值介于 1 和 k 之间； [7] 证明， k 的值越大，增长率也越大。请注意，增长率决定了 $\mathrm{err}_{H,\,I}$ 关于 λ 的线性关系的斜率。

情形 $k = 1$ 尚未被 [7]考虑。在这种情况下，增长率显然为 1，因为每次插入恰好会使一个计数器递增。此外，在这种情况下可以轻松推导出 $\mathrm{err}_{H,\,I}$，因为给定键的误差由哈希到同一计数器的键数决定。这类键的数量可用参数为 $\frac{m}{n} = \lambda$、期望值为 λ 的泊松分布近似。由于键均匀分布，$\mathrm{err}_{H,\,I} = \lambda$ 是平均相对误差的良好估计值。

然而，要看出这与中等负载因子情况下 $k = 2$ 的误差有何不同，并非易事。图 2 澄清了这一情况。图中显示，在达到某个临界点之前， $k = 1$ 产生的估计值较大；之后则低于 $k = 2$ 的估计值。这个“中间区域”大致对应于 $n \ln n$ 集邮者界，即仍有大量计数器为零的区域。在这种情况下，即使 $k = 1$ 的计数器总和小于 $k = 2$，后者的分布也更均匀，类似于资源分配中广为人知的选择的力量 现象 [2]。由于空计数器与平均相对误差无关，因此 $k = 2$ 的误差小于 $k = 1$。请注意，对于 $k = 1$，我们观察到误差曲线与对角线吻合。这可以解释为：对于任意给定的键 x，与 x 落入同一计数器的其他键数（它决定了对 x 的估计值的误差）通常用参数为$\lambda = m/n$、期望值为 λ 的泊松随机变量近似。

请注意，配置 $k = 1$ 并未受益于保守更新策略相较于常规 Count-Min 的优势，从实际角度来看，其意义也十分有限。不过，有趣的是，可以观察到，在有限的 λ 下， $k = 2$ 仍然优于 $k = 1$。

3.1.3 混合超图

我们刚刚看到，当负载足够大时，配置 $k = 1$ 会产生最小的错误。接下来要探讨的问题是，使用数量不一的哈希函数（混合超图）能否进一步减小错误，类似于亚临界区间：在该区间中，这种做法扩大了能够支持 $o(1)$ 错误的负载值范围。乍一看，这个问题似乎无关紧要，因为配置$k = 1$ 显然使计数器总和达到可能的最小值：每次插入操作至少会使一个计数器递增，因此 $k = 1$ 似乎能得到最小的估计值。

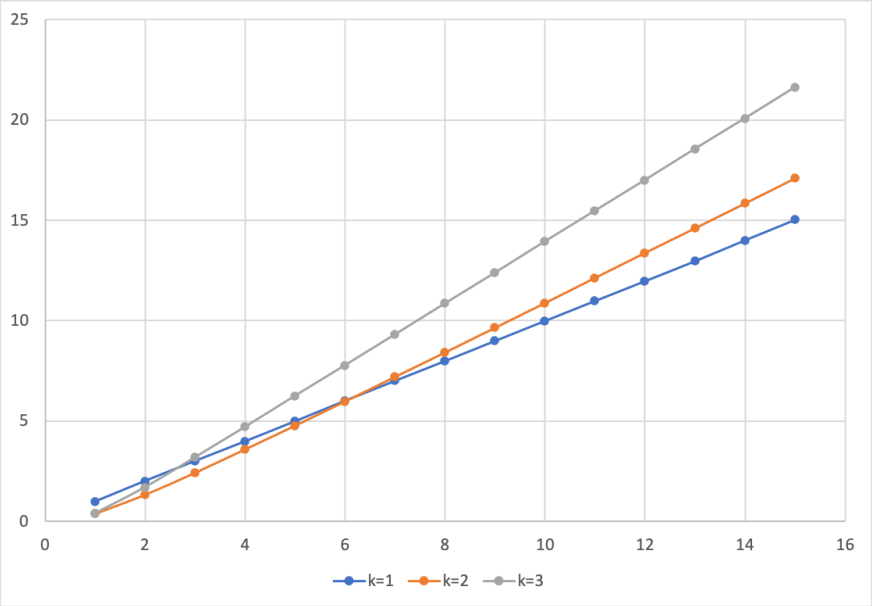


图 2：在均匀分布和超临界区域中，$\mathrm{err}_{H,\,I}$ 随 $\lambda = m/n$ 的变化。

不过，这一论证适用于超图已饱和（所有计数器都被命中）的情形，并且 λ 足够大，使计数器的值趋于集中。请注意，如前所述，计数器值近似服从泊松($k\lambda$)分布；当 λ 变大时，分布会集中在 $k\lambda$附近。也许令人意外的是，在 λ 达到饱和点之前，对于一大段 λ 取值范围，使用数量不一的哈希函数可以使 $k = 1$ 产生的错误更小。图 3 展示了这一现象。它比较了均匀情形 $k = 1$ 和一种混合配置：80% 的边的基数为 1，20% 的边的基数为 3。对于不超过约 50 的 λ 取值，后者产生的错误更小。与上一节类似，这可以用选择的力量 效应来解释：基数为 3 的边会“平滑”计数器值，从而减小组合错误；在尚未达到渐近状态的区间 $\lambda < 50$ 中，这一效应略微强于基数为 3 的边会使计数器值总和增大的影响（因为某些插入操作会使多个计数器递增）。

3.2 阶梯分布

对输入元素服从均匀分布时保守型 Count-Min 行为的分析表明，在超临界区域中，草图产生的误差会随负载因子线性增长。这意味着该草图在此区域的实用性有限。另一方面，在许多实际情况下，输入元素出现的频率并不相同。因此，有必要将 Count-Min 应用于非均匀分布，尤其是用于检测和分析输入流中的高频元素。一个广受关注的问题是计算重频项，Count-Min 草图此前已被应用于此 [14]。

在本节中，我们聚焦于最简单的非均匀分布——阶梯分布——以考察存在频率各异的元素时 Count-Min 草图的行为。我们的模型如下。我们假设输入元素分为两组，分别称为

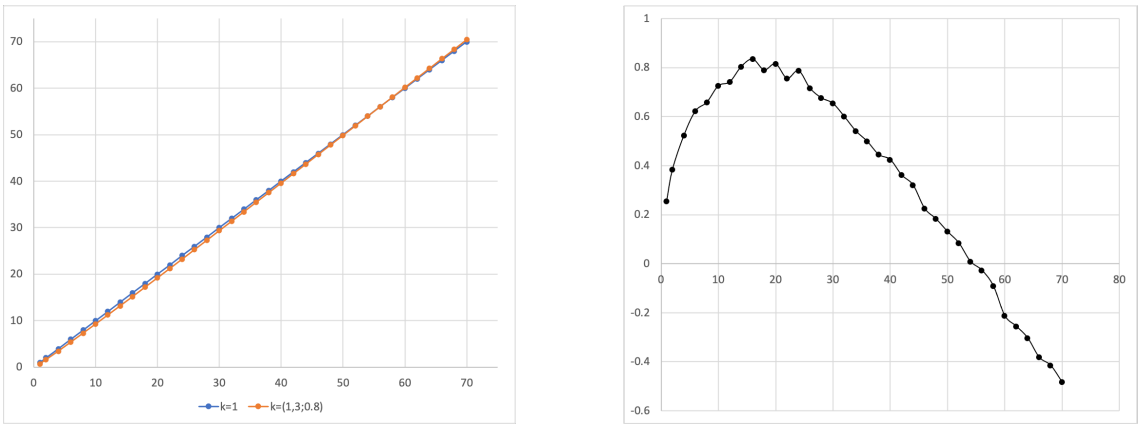


图 3：$\mathrm{err}_{H,\,I}$ 作为 λ 的函数，在均匀分布和超临界区域下。均匀配置 $k=1$（蓝色曲线）与混合配置 $k=$（1，3；0.8）（橙色曲线）的对比。为清晰起见，右图展示了前者与后者之间的差异。

热元素和冷元素，其中热元素的出现概率高于冷元素。请注意，我们假设已先验知道给定元素属于热元素还是冷元素。这种设定类似于针对结合先验成员资格信息和查询概率的布隆过滤器所进行的研究 [6]。请注意，我们对 $\mathrm{err}_{H,\,I}$ 的定义假设某个元素的查询概率与其在输入中的出现概率相等。

我们假设热元素和冷元素的负载因子分别为 λ_h 和 λ_c。也就是说，哈希超图中有 $\lambda_h n$ 条热边和 $\lambda_c n$ 条冷边。 $G>1$ 称为间隔因子，表示热元素与冷元素概率之比。令 p_h（分别为 p_c）表示输入元素为热元素（分别为冷元素）的概率。那么 $p_h/p_c = G\lambda_h/\lambda_c$，并且由于$p_h+p_c=1$，我们有

$$p_h=\frac{G\lambda_h}{\lambda_c+G\lambda_h},\quad p_c=\frac{\lambda_c}{\lambda_c+G\lambda_h}.$$

例如，如果不同的冷元素数量是热元素的 10 倍（$\lambda_h/\lambda_c=0.1$），但每个热元素出现的频率是冷元素的 10 倍（$G=10$），那么输入中热元素和冷元素所占比例大致相同（$p_h=p_c=0.5$）。

在本节余下部分，我们将关注仅由热元素产生的组合错误，记为 $\mathrm{errhot}_{H,\,I}$。如果 $E_h\subseteq E$ 是热元素的子集，且 N_h 是输入中热元素出现次数的总数，那么 $\mathrm{errhot}_{H,\,I}$ 定义为

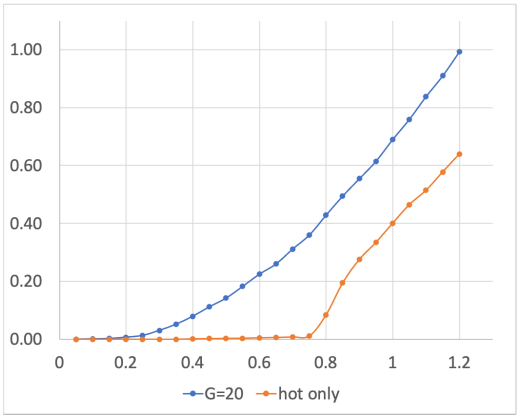
$$\mathrm{errhot}_{H,I}=\frac{1}{N_h}\sum_{e\in E_h} occ(e)\cdot \mathrm{err}(e)=\frac{1}{N_h}\sum_{e\in E_h}(c(e)-occ(e)).$$

3.2.1 热元素与冷元素的“交互”

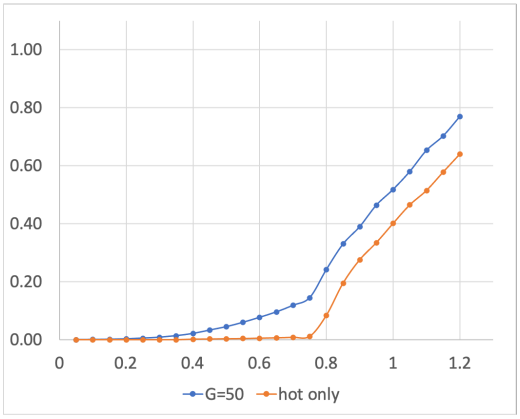
将元素划分为热点和冷元素，会使底层哈希超图相应地划分为两个子图，分别称为热点和冷子图。由于热点元素的计数较大，人们可能会推测，与热点边相关联的计数器大于冷元素的计数，因此不会因冷元素而递增。于是，$\mathrm{errhot}_{H,\,I}$ 完全由

热点子图在元素均匀分布下的情况决定。具体而言，作为 λ_h 的函数，errhot$_{H,\,I}$ 的表现应与均匀分布下的 err$_{H,\,I}$ 相同（参见第 3.1 节）。

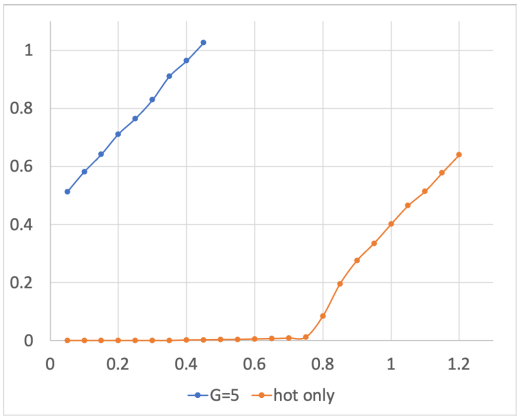
然而，这一猜想一般来说并不成立。原因之一是，所有冷边的节点也同时连接到热点边的概率为正。因此，“热计数器”（即与热点边相连的计数器）会因冷边而获得额外增量，而后者会导致对热边计数的高估。图 4a 展示了这一点。对于间隔值 $G = 20$，它展示了在存在 $\lambda_c = 5$ 的冷元素时，$k = 3$、errhot$_{H,\,I}$ 随 λ_h 的变化。为便于比较，橙色曲线展示了均匀分布下的误差（如图 1 所示），即在不存在冷元素的情况下，热点元素本会产生的误差。我们清楚地观察到，冷元素会对误差产生影响，即使是在低于可剥离性阈值的负载区间内也是如此。



（a） $G = 20$



（b） $G = 50$



（c） $G = 5$

图 4：errhot$_{H,\,I}$ 随 λ_h 变化时的 $k = 3$，存在 $\lambda_c = 5$ 冷元素（蓝色曲线）和不存在任何冷元素（橙色曲线）的情况。

图 4b 表明，当间隔变大时（此处为 $G = 50$），冷元素的影响会减弱，曲线也会趋近于均匀分布对应的曲线。间隔越大，热元素的值越大，因此冷元素的相对影响越小。

上述猜想可能不成立的另一个原因是：即使

热点元素的数量非常少，但如果间隔因子不够大，那么只要 λ_c 足够大，冷边就可能导致计数器的数值变大，尤其是在第 3.1.2 节所述的饱和区间。因此，冷边产生的计数器“背景水平”可能高于热点边的真实计数，从而导致对其估计值偏高。举例来说，再次考虑图 4 中具有 $k = 3$ 和 $\lambda_c = 5$ 的配置。图 2 显示，单独考虑冷元素时，平均误差约为 6（准确来说是 ≈ 6.25，数据未展示），这意味着估计值高出约 $7\times$ 。由于图在此区间已饱和（见第 3.1.2 节），这意味着大多数计数器的数值约为冷边计数的 7 倍。现在，如果某个热元素的出现频率仅为冷元素的 5 倍，那么它们的估计值就会高出约 $1.4\times$ ，也就是说误差约为 0.4。参照图 4，这意味着在这种情况下，蓝色曲线一开始就处于正误差值，而不是从零开始。图 4c 展示了这种情况。

3.2.2 混合超图

上述分析表明，在由大量冷元素构成的“背景”存在时，即使冷元素的出现频率远低于热元素，热元素的误差也会比不存在冷元素时，在小得多的负载因子下开始增大。受第 3.1.1 节结果的启发，我们可以探讨能否通过采用可变数量的哈希函数来扩大误差可忽略区间。请注意，在这里，将不同数量的哈希函数分别分配给热元素和冷元素，是更自然的做法。

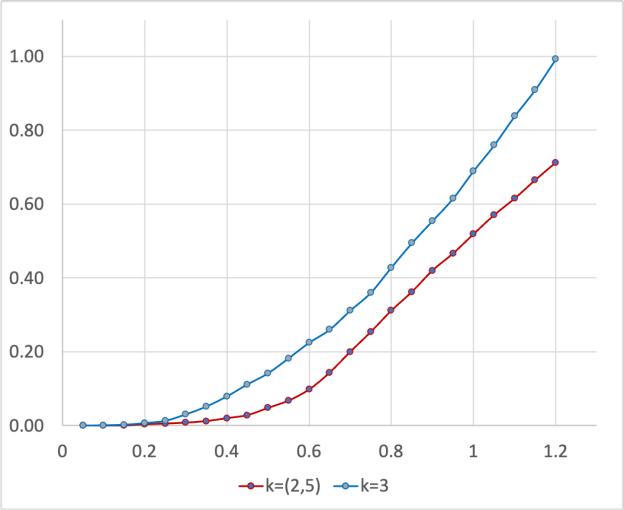


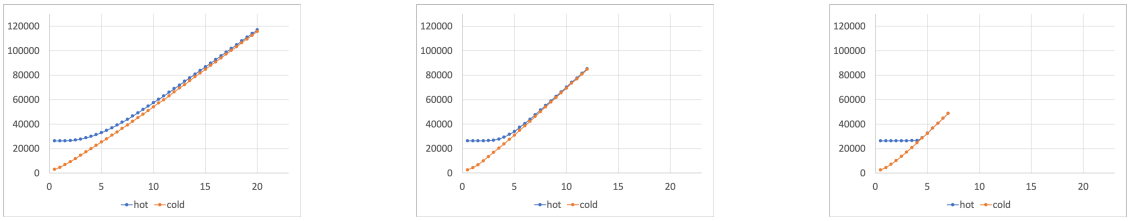
图 5：errhot$_{H,\ I}$ 作为 λ_h 的函数，其中 $k = 3$、$\lambda_c = 5$ 和 $G = 20$ （蓝色曲线，与图 4a 相同）；对于热元素和冷元素，分别为 $k = (2,\ 5)$（红色曲线）

图 5 展示了这一点确实可行：为热点元素分配较少的哈希函数，为冷元素分配较多的哈希函数。显然，误差接近于零的区间扩大了。这是因为，当热子图的密度不太高时，增加冷边的基数会提高这样的概率：至少有一个此类边的顶点不与热边关联。因此，该元素不会影响热边的误差。出于相同原因，减小热边的基数（此处从 3 减至 2）也能改善误差，因为这会提高不与热边关联的顶点所占的比例。

3.2.3 超临界区域中的饱和

在第 3.1.2 节中，我们讨论了负载值较大时出现的饱和区间：当负载增长到足够大时，也就是说，哈希超图变得足够稠密时，所有计数器都会达到相同的水平，从而抹去边与边之间的差异。在此区间内，假设负载（图密度）固定且输入服从均匀分布，则边值仅取决于输入规模，而与图结构无关（这一结论以较高概率成立）。正是在这一背景下，Bianchi 等人 [7] 研究了边值随输入规模变化的增长率。

这种饱和现象是否也适用于非均匀分布，是一个有趣、自然且具有重要实际意义的问题，因为这直接关系到区分不同频率元素的能力。对此问题作出全面而精确的回答不在本文的研究范围之内。我们认为，至少对于分段均匀分布，答案是肯定的：边被划分为若干类，每一类内等概率，且每一类都包含所有元素中线性比例的元素。下面我们用阶梯分布来说明这一观点。



(a) 2-均匀　　(b) 3-均匀　　(c) (2,5)-混合

图 6：2-均匀（6a）、2-均匀（6b）和（2,5）-混合超图中热元素与冷元素平均估计值的收敛情况。x轴显示所有情况下的总负载 $\lambda = \lambda_h + \lambda_c$ with $\lambda_h = 0.1 \cdot \lambda$and $\lambda_c = 0.9 \cdot \lambda$ and $G = 10$。所有实验均针对 $n = 1000$ 运行，输入流长度为边数的 5，000 倍（即 $5 \cdot 10^6 \cdot \lambda$）。每个平均值均取自 10 个随机超图。

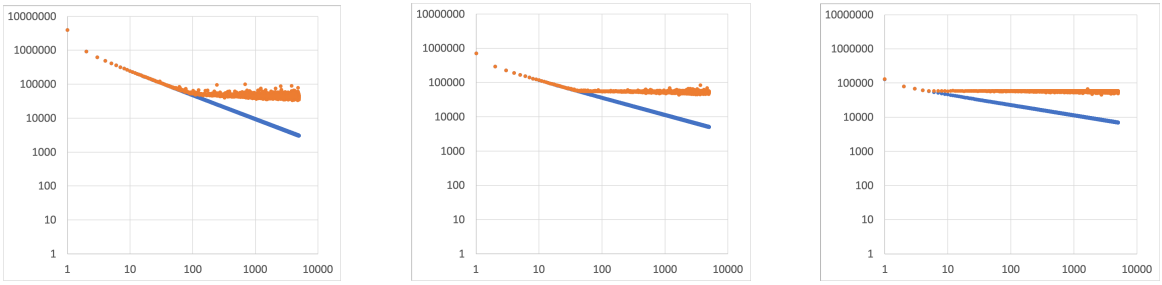
图 6 展示了饱和现象，给出了三种不同配置下热点边和冷边的平均值（$G = 10$）：2-均匀、3-均匀和 (2,5)-混合。请注意，这里的x轴表示总负载 $\lambda = \lambda_h + \lambda_c$，其中 $\lambda_h = 0.1 \cdot \lambda$ 和 $\lambda_c = 0.9 \cdot \lambda$。也就是说，随着边的总数增长，热点边和冷边的数量都呈线性增长。

可以观察到，在所有配置下，热点边和冷边的数值都趋于一致，这体现了饱和现象。有趣的是，“收敛速度”很大程度上取决于配置：均匀配置下的收敛较“慢”，而在混合配置下，热点边在小误差区间之后便立即收敛。

3.3 齐夫分布

幂律分布在实际应用中无处不在。其中最简单的是齐夫分布，它常被用作多种算法的测试用例，包括 Count-Min 草图[15, 7, 19, 10, 4]。我们还提及一种近期基于学习的方法 CountMin [25]（学习重频项），以及对其在 Zipf分布下的研究 [1, 18]。在齐夫分布下，元素概率按降序排列后与 $1/i^{\beta}$ 成正比，其中 i 是键的秩，$\beta \geq 0$ 是 偏斜度 参数。请注意，当 $\beta = 0$ 时，齐夫分布会退化为均匀分布。

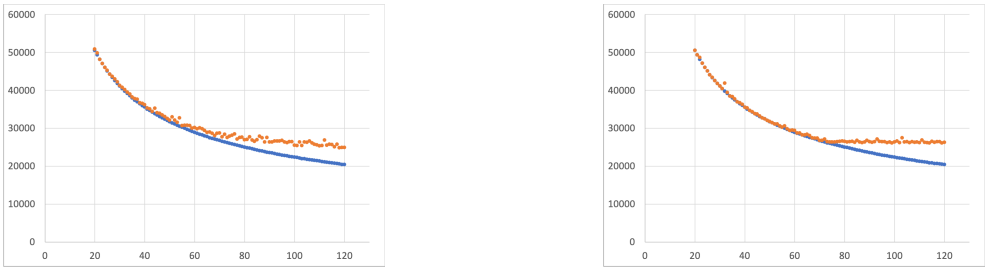
齐夫分布也是我们研究中的一个重要测试用例，因为它会使若干（少数）最高频元素的计数非常大，同时使大量元素（重尾）的计数较小；这些计数值随元素秩的增加仅按多项式级别递减，因此数量级相同。Bianchi 等人 [7, 图 1] 观察到，在超临界区间内，齐夫分布的估计值呈现“瀑布型行为”：最高频元素的估计值基本精确，而其他元素的估计值则几乎相同，与其频率无关。图 7 展示了不同偏斜度下的这一现象。



(a) $\beta = 0.7$ (b) $\beta = 0.5$ (c) $\beta = 0.3$

图 7：齐夫分布中边值的精确值（蓝色）和估计值（橙色）随元素频率秩变化的关系，采用双对数尺度绘制。所有图均针对 $n = 1000$、$\lambda = 5$、$k = 2$ 和输入规模 $50 \cdot 10^6$ 得出。估计值取 10 次哈希函数抽样的平均值。

前文各节提出的分析很好地解释了齐夫分布的瀑布型行为。数值的“瀑布池水位”（在 [7] 中称为误差下限）是重尾元素饱和所造成的结果。少数“异常高频”的元素数量太少（其数量为 $\ll n$），不足以影响饱和水平；结果，它们构成水位之上的“峰值”，因此估计值没有误差。自然，偏斜度较小会使分布不那么陡峭，并减少“异常高频”元素的数量。例如，根据图 7，对于 $\lambda = 5$ 和 $k = 2$，约有 50 个最高频元素在 $\beta = 0.7$ 时估计无误，$\beta = 0.5$ 时约有 40 个，而 $\beta = 0.3$ 时只有 5 个。



(a) $k = 2$ (b) $k = (2{,}5;\ 0.2)$

图 8：齐夫分布的精确边值（蓝色）和估计边值（橙色）随元素频率秩的变化情况，适用于 $n = 1000$、$\lambda = 2$ 和输入规模 $20 \cdot 10^6$。仅显示秩 20 至 120 的值。估计值为 50 次哈希函数抽样的平均值。

根据前几节的结果，我们研究了使用数量可变的哈希函数是否能够扩大以较小误差估计高频元素的范围。我们发现，对于

中等负载 λ，确实可以做到。图 8 展示了 $\lambda = 2$ 的这一情况。它放大显示了 $k = 2$ 与 $k =$ (2,5; 0.2) 在“转折点”附近的情况（见图 7）。为清晰起见，图中采用线性刻度，且仅显示秩为 20 至 120 的元素。图中表明，$k =$ (2,5; 0.2) 的转折更为明显（另见图 6）。因此，尽管 $k =$ (2,5; 0.2) 的饱和水平高于 $k = 2$，使用 $k =$ (2,5; 0.2) 时，仍能以较小误差估计约 70 个最高频元素；而使用 $k = 2$ 时，约为 50 个。

4 结论

本文展示了一系列实验结果，为深入了解保守型 Count-Min 草图的行为提供了新的洞见。其中一些结果可直接应用于该数据结构的实际使用。主要结果总结如下。

- 对于输入元素服从均匀分布的情况，为不同元素分配不同数量的哈希函数，可以扩展支持相对误差渐近趋于零的亚临界区间（负载因子范围 λ）。这立即意味着，满足该区间条件的 Count-Min 配置可以节省空间。对于非均匀分布，改变哈希函数的数量可以扩展最高频元素的可忽略误差区间，

- 在“足够均匀的分布”下，包括均匀分布和阶梯分布，当 λ 足够大时，Count-Min 草图会达到饱和区间。在该区间内，计数器会集中在同一数值附近，具有不同频率的元素将无法区分，

- 能够以较小误差进行估计的高频元素，可以看作是那些超过由其他大多数元素形成的饱和水平的元素。例如，对于齐夫分布，这些元素是少数“异常高频元素”，而饱和则由重尾元素造成。对于中等负载 λ，采用数量可变的哈希函数可以增加这类元素的数量。

许多这类结果缺乏严谨的数学分析。对于 Count-Min 的实际应用而言，或许最相关的问题是如何估计不同分布下的“瀑布池水位”，这对重频项类应用至关重要。事实上，在“高度超临界”的情形下，估计值往往会收敛到同一水位，正如第 3.2.3 节中针对阶梯分布所展示的那样。另一方面，只有少数频率“异常高”、超过该水位的元素，其误差才较小。齐夫分布的例子展示了这一点（第 3.3 节）。Bianchi 等人 [7] 观察到，对于输入元素服从非均匀分布的情形，“瀑布池水位”不超过计数器数值的饱和水位；其中输入分布被建模为在所有 $\binom{n}{k}$ 可能的边中均匀选择。后者通过一种基于马尔可夫链和微分方程的方法计算得出 [7] 。我们认为，这种方法也可以扩展到混合图的情形，并将其留待未来研究。不过，如何计算包括齐夫分布在内的非均匀分布下的“瀑布池水位”，仍是一个开放问题。

参考文献

[1] Anders Aamand、Piotr Indyk 和 Ali Vakilian。Zipf分布下的（学习型）频率估计算法。*CoRR*，abs/1908.05198，2019。网址：http://arxiv.org/abs/1908. 05198，arXiv:1908.05198。[2] Yossi Azar、Andrei Z. Broder、Anna R. Karlin 和 Eli Upfal。平衡分配。*SIAM J.Co mput.*，29(1):180–200，1999。doi:10.1137/S0097539795288490。[3] Sairam Behera、Sutanu Gayen、Jitender S Deogun 和 NV Vinodchandran。KmerEstimate：一种以最优空间用量估计 k-mer 计数的流式算法。载于*2018* 年 *ACM* 生物信息学、计算生物学与健康信息学国际会议论文集，第 438–447 页，2018 年。[4] Younes Ben Mazziane、Sara Alouf 和 Giovanni Neglia。采用保守更新的 Count-Min 草图的形式化分析。载于*IEEE INFOCOM WNA 2022*——第二届网络算法研讨会（*WNA*），美国纽约，2022 年 5 月。亦见 arXiv.2203.14549。doi:10.1109/INFOCOMWKSHPS54753.2022.9798146。[5] Younes Ben Mazziane、Sara Alouf 和 Giovanni Neglia。分析采用保守更新的 Count-Min 草图。计算机网络，217:109315，2022 年。网址：https://www.sciencedirect. com/science/article/pii/S1389128622003607。[6] Ioana O. Bercea、Jakob Bæk Tejs Houen 和 Rasmus Pagh。Daisy 布隆过滤器。*CoRR*，abs/2205.14894，2022 年。arXiv:2205.14894。[7] Giuseppe Bianchi、Ken Duffy、Douglas J. Leith 和 Vsevolod Shneer。多哈希近似计数草图中保守更新的建模。载于第 *24* 届国际电信业务大会（*ITC 2012*），波兰克拉科夫，*2012* 年 *9* 月 *4–7* 日，第 1–8 页。IEEE，2012 年。网址：https:// ieeexplore.ieee.org/document/6331813/。[8] Jehoshua Bruck、Jie Gao 和 Anxiao Jiang。加权布隆过滤器。载于*2006* 年 *IEEE* 信息论国际研讨会论文集（*ISIT 2006*），美国华盛顿州西雅图市 *Westin Seattle* 酒店，*2006* 年 *7* 月 *9–14* 日，第 2304–2308 页。IEEE，2006 年。doi:10.1109/ISIT.2006. 261978。[9] Moses Charikar、Kevin Chen 和 Martin Farach-Colton。查找数据流中的频繁项。理论计算机科学，312(1):3–15，2004 年。[10] Peiqing Chen、Yuhan Wu、Tong Yang、Junchen Jiang 和 Zaoxing Liu。基于草图的流量测量的精确误差估计。载于第 *21* 届 *ACM* 互联网测量会议论文集，IMC '21，第 113–121 页，美国纽约州纽约市，2021 年。计算机协会。doi:10.1145/3487552.3487856。[11] Saar Cohen 和 Yossi Matias。谱布隆过滤器。载于 Alon Y. Halevy、Zachary G. Ives 和 AnHai Doan（编），*2003* 年 *ACM SIGMOD* 数据管理国际会议论文集，美国加利福尼亚州圣迭戈，*2003* 年 *6* 月 *9–12* 日，第 241–252 页。ACM，2003 年。doi:10.1145/872757.872787。

[12] Graham Cormode。Count-Min 草图。载于 Ling Liu 和 M. Tamer Özsu 主编的《数据库系统百科全书》第二版。Springer，2018年。doi:10.1007/978-1-4614-8265-9_87.[13] Graham Cormode 和 Marios Hadjieleftheriou。数据流中的频繁项查找。*VLDB*基金会论文集，1(2):1530–1541，2008年。[14] Graham Cormode 和 S. Muthukrishnan。一种改进的数据流摘要：Count-Min 草图及其应用。算法杂志，55(1):58–75，2005年4月。doi:10.1016/j.jalgor.2003.12.001.[15] Graham Cormode 和 S. Muthukrishnan。偏斜数据流的摘要与挖掘。载于 Hillol Kargupta、Jaideep Srivastava、Chandrika Kamath 和 Arnold Goodman 主编的*2005*年*SIAM*国际数据挖掘会议论文集，*SDM 2005*，美国加利福尼亚州纽波特比奇，*2005*年*4*月*21*日至*23*日，第44–55页。SIAM，2005年。doi:10.1137/1. 9781611972757.5.[16] Martin Dietzfelbinger、Andreas Goerdt、Michael Mitzenmacher、Andrea Montanari、Rasmus Pagh 和 Michael Rink。通过 XORSAT 得到布谷鸟哈希的紧确阈值。载于 Samson Abramsky、Cyril Gavoille、Claude Kirchner、Friedhelm Meyer auf der Heide 和 Paul G. Spirakis 主编的自动机、语言与编程：第*37*届国际研讨会，*ICALP 2010*，法国波尔多，*2010*年*7*月*6*日至*10*日，论文集，第一部分，第6198卷计算机科学讲义，第213–225页。Springer，2010年。doi:10.1007/978-3-642-14165-2_19.[17] Martin Dietzfelbinger 和 Stefan Walzer。稠密可剥离随机均匀超图。载于 Michael A. Bender、Ola Svensson 和 Grzegorz Herman 主编的第*27*届欧洲算法年度研讨会，*ESA 2019*，*2019*年*9*月*9*日至*11*日，德国慕尼黑/加兴，第144卷*LIPIcs*，第38:1–38:16页。达格施图尔城堡—莱布尼茨信息学中心，2019年。doi:10.4230/LIPIcs.ESA.2019.38.[18] Elbert Du、Franklyn Wang 和 Michael Mitzenmacher。将“学习”融入用于频率估计的学习增强算法。载于 Marina Meila 和 Tong Zhang 主编的第*38*届国际机器学习会议论文集，*ICML 2021*，*2021*年*7*月*18*日至*24*日，线上会议，第139卷机器学习研究论文集，第2860–2869页。PMLR，2021年。URL: http://proceedings.mlr.press/v139/du21d.html.[19] Gil Einziger 和 Roy Friedman。对基于保守更新的近似计数进行形式化分析。载于国际计算、网络与通信会议，*ICNC 2015*，美国加利福尼亚州加登格罗夫，*2015*年*2*月*16*日至*19*日，第255–259页。IEEE计算机学会，2015年。doi:10.1109/ICCNC.2015.7069350.[20] Cristian Estan 和 George Varghese。流量测量与计量的新方向。载于 Matthew Mathis、Peter Steenkiste、Hari Balakrishnan 和 Vern Paxson 主编的*ACM SIGCOMM 2002*计算机通信应用、技术、架构与协议会议论文集，*2002*年*8*月*19*日至*23*日，美国宾夕法尼亚州匹兹堡，第323–336页。ACM，2002年。doi:10.1145/633025.633056.[21] Bin Fan、Dave G. Andersen、Michael Kaminsky 和 Michael D. Mitzenmacher。布谷鸟过滤器：实际性能优于布隆过滤器。载于第*10*届*ACM*国际会议论文集

新兴网络实验与技术，CoNEXT '14，第75–88页，美国纽约，2014年。计算机协会。doi:10.1145/2674005.2674994.[22] Li Fan、Pei Cao、J. Almeida 和 A.Z. Broder。摘要缓存：一种可扩展的广域网 Web 缓存共享协议。*IEEE/ACM* 网络学报，8(3):281–293，2000年。doi:10.1109/90.851975.[23] Éric Fusy 和 Gregory Kucherov。采用保守更新的 Count-Min 草图进行计数近似时的相变。载于 Marios Mavronicolas 主编的第*13*届国际算法与复杂性会议论文集（*CIAC'23*），塞浦路斯拉纳卡，*2023*年*6*月*13*日至*16*日，第13898卷计算机科学讲义，第232–246页。Springer，2023年。完整版本见 arxiv:2203.15496，2022年。[24] Michael T Goodrich 和 Michael Mitzenmacher。可逆布隆查找表。载于*2011*年第*49*届年度阿勒顿通信、控制与计算会议（*Allerton*），第792–799页。IEEE，2011年。[25] Chen-Yu Hsu、Piotr I ndyk、Dina Katabi 和 Ali Vakilian。基于学习的频率估计算法。载于第*7*届国际学习表征会议，*ICLR 2019*，美国路易斯安那州新奥尔良，*2019*年*5*月*6*日至*9*日。OpenReview.net，2019年。URL: https://openreview.net/ forum?id=r1lohoCqY7.[26] Hongyan Liu、Yuan Lin 和 Jia wei Han。数据流中频繁项挖掘方法综述。知识与信息系统，26(1):1–30，2011年。[27] Bohdan S Majewski、Nicholas C Wormald、George Havas 和 Zbigniew J Czech。一类完美哈希方法。计算机杂志，39(6):547–554，1996年。[28] Hamid Mohamadi、Hamza Khan 和 Inanc Birol。ntCard：一种用于基因组学数据基数估计的流式算法。生物信息学，33(9):1324–1330，2017年。[29] Michael Molloy。随机超图和布尔公式中的核心。随机结构与算法，27(1):124–135，2005年。[30] Rasmus Pagh 和 Flemming Friche Rodler。布谷鸟哈希。算法杂志，51(2):122–144，2004年。[31] Michael Rink。用于线性时间构造更稠密基于哈希的数据结构的混合超图。载于 Peter van Emde Boas、Frans C. A. Groen、Giuseppe F. Italiano、J erzy R. Nawrocki 和 Harald Sack 主编的*SOFSEM 2013*：计算机科学理论与实践，第*39*届计算机科学理论与实践当前趋势国际会议，捷克共和国什平德莱鲁夫姆林，*2013*年*1*月*26*日至*31*日。论文集，第7741卷计算机科学讲义，第356–368页。Springer，2013年。doi:10.1007/978-3-642-35843-2_31.[32] Yoshihiro Shibuya 和 Gregory Kucherov。Set-Min 草图：一种用于幂律分布的概率映射，应用于 k-mer注释。*bioRxiv*，第2020.11.14.382713页，2020年11月。doi:10.1101/2020.11.14.382713。

[33] Stefan Walzer。用于基于哈希的数据结构的随机超图。博士论文，伊尔梅瑙工业大学，德国，2020年。URL：https://www.db-thueringen.de/receive/dbt_mods_00047127。

[34] Xiujun Wang、Yusheng Ji、Zhe Dang、Xiao Zheng 和 Baohua Zhao。改进的加权布隆过滤器及近似成员查询算法的空间下界分析。收录于 Matthias Renz、Cyrus Shahabi、Xiaofang Zhou 和 Muhammad Aamir Cheema 编，高级应用数据库系统——第*20*届国际会议，*DASFAA 2015*，河内，越南，*2015*年*4*月*20*日至*23*日，论文集，第二部分，《计算机科学讲义》第9050卷，第346–362页。Springer，2015年。doi:10.1007/978-3-319-18123-3_21。