基于MGCBF算法的长流信息统计[1]

周明中 龚俭 丁伟 程光
（东南大学计算机系 江苏南京 210096）
（江苏省计算机网络技术重点实验室）

摘　要：为提高流测量系统运行效率，减小其所需存储资源，本文在分析网络中流长分布特性的基础上，提出一种新颖的用于测量长流数量并维护其流信息的算法——多粒度计数 Bloom Filter (MGCBF)。利用较少的固定存储空间，MGCBF 可以在保持较小误差比例的情况下，对所有到达的流基于报文计数。本文在 MGCBF 算法的基础上以指定报文数为阈值建立了一个长流信息统计模型，并对该模型所需的存储空间，计算复杂度和计算误差进行了详细分析和讨论。通过将其分别应用于来自不同网络的 TRACE——CERNET 和 CESCAI，验证了该算法在保证测量精度的同时可以大幅度减小维护流信息所需的系统资源。

关键词：网络流量测量；流长计数；信息维护; MGCBF 算法
中图分类号：TP393　　**文献标识码：**A

Long Flows' Information Statistics based on MGCBF Algorithm

ZHOU Mingzhong, GONG Jian, DING Wei, CHENG Guang
(Department of Computer Science, Southeast Univ., Jiangsu, Nanjing 210096 China)
(Jiangsu Province Key Laboratory of Computer Networking Technology)

Abstract: In order to improve the performance and reduce the resource usage of flow-based measurement systems, this paper presents a novel long flow counting and information maintenance algorithm called Multi-Granularity Counting Bloom Filter (MGCBF) based on the distribution and characteristics analysis of long flows in the Internet. With less fixed memory used, MGCBF maintains the counters for all incoming flows with small error probability, and keeps long flow information identified with a fixed packet number threshold through an expanding data structure, by which a statistical model for long flow information can be built up. This paper also analyzes the space used, calculation complexity and error probability of this model. The experiments which applied this model on the different TRACEs from CERNET and CESCAI show that the algorithm MGCBF can reduce dramatically the resource usage in flows counting and information maintenance with losing little accuracy.

Keywords: Network Traffic Measurement; Counting Flow Length; Information Maintenance; Multi-Granularity Counting Bloom filter (MGCBF)

1 绪论

随着对基于流的网络流量测量的需求不断增加（如流量计费，安全分析，入侵检测，流量工程等），各种针对不同应用背景的流测量方法也陆续出现，长流识别和计数作为其中的重要组成部分被许多文献广泛研究[1][2][3][4]。由于应用背景各不相同，这些文献在处理长流是所关心的重点也有较大差异。

目前使用最广泛的流识别方法是 IETF 的实时流测量工作组（RTFM）[1]提出的，用以在路由设备中采集全部或部分流的信息，但是由于采样报文数量的限制，其扩展性比较差，对准确性要求较高或者长时间粒度观测的应用而言，RTFM 工作组提供的方法显然不能满足需求。Shaikh A, Rexford J, Shin K. G. [2]为实现负载均衡，维护每个流的状态信息，通过在一定时间内到达的报文数量判断流是否属于长流，这种方式比较直接但并不十分高效。Kim S.I, Reddy A. L. [3]提出了一种叫做最近最少使用（LRU）的算法，在边界路由器上用以分辨长期存在并且所占流量比例较高的流，并用作负载均衡，该算法只维护短时间的长流信息并且随时更新，也不对相关的长流信息做长期保存。Estan C, Varghese G. [4]提出了两种算法用于发现长流：*sample and hold* 和 *multistage filters*，有效地解决了如何在报文抽样情况下获取和维护流信息的问题，在文献中还对以前相关算法做了比较详细的分析，但是这两种算法都只统计占用网络中较大带宽的若干巨流的信息。Kumar

基金项目：国家重点基础研究发展计划（973 计划）课题（2003CB314804），江苏省网络与信息安全重点实验室（BM2003201）资助，教育部科学技术重点研究项目（105084）
作者简介：周明中（1976-），男，博士生；龚俭（联系人），男，博士，教授，博士生导师，jgong@njnet.edu.cn

A, Xu J, et al.在[5]中给出了一种用于流数量统计的算法——SCBF，使用有限的资源在一定误差的范围内存储所有流的长度信息，并使用极大似然估计（MLE）和平均值估计（MVE）来推测特定流的长度，但是这些算法都只关心流长的统计或估计而并不维护相关的流信息，这就不能完全满足一些特定应用（譬如负载均衡，流量计费等）的需求。

2 基于 MGCBF 算法的流长统计和信息维护

MGCBF算法使用一系列的Counting Bloom Filter（CBF）[6][7][8]，MGCBF={cbf_0,cbf_1, …, cbf_{h-1}}，对集合S中各个元素s_i出现的次数进行统计，每一个CBF采用不同的计量单位C={1,c_1,c_2, …, c_{h-1}}。该算法主要适用于“iceberg查询”——即在取自于一个大集合的元素序列中（如在一定时间段内经过特定观测点的报文），查询出现频率高于某个阈值的元素。本文提出的MGCBF算法主要是针对不同元素出现频率呈重尾分布的序列进行iceberg查询操作，所谓重尾分布是指这些相异元素大部分出现的次数很小，但有一小部分出现的次数（频率）十分高，以至于它们总体数量占集合中的绝大部分。算法的基本设计思想描述如下：

1) 当一个元素x加入到MGCBF时，cbf_0的向量空间V_0上对应的计数单元{$v_1^0(x)$,$v_2^0(x)$, …, $v_{k0}^0(x)$}均增加 1；（k0 为cbf_0中哈希函数的个数，并假设$v_1^0(x)\leqslant v_2^0(x)\leqslant\ldots\leqslant v_{k0}^0(x)$）；
2) 当$v_1^0(x)=c_1$时，将对应各计数单元都减去c_1，{0,$v_2^0(x)-c_1$, …, $v_{k1}^0(x)-c_1$}，然后在cbf_1的向量空间V_1上对应的计数单元{$v_1^1(x)$,$v_2^1(x)$, …, $v_{k1}^1(x)$}都增加 1，{ $v_1^1(x)+1$,$v_2^1(x)+1$, …, $v_{k1}^1(x)+1$}；
3) 当cbf_1中对应x的计数单元$v_1^1(x)=c_2$时，执行类似的操作，向cbf_2对应的计数单元进位，并依次递推直到cbf_h；假设元素x在不同cbf中的最小计数单元值为M(x)={$min_0(x)$,$min_1(x)$, …, $min_{h-1}(x)$}，则此时x元素在S中出现的次数为：

$$Counter(x) = min_0(x)+min_1(x)*c_1+ \ldots+ min_{h-1}(x)* \prod_{i=1}^{h-1} c_i。 \qquad (1)$$

由于 Internet 主干网中，流长分布呈重尾特性[2][3][4][9]，占流数总量极小部分的长流占用了绝大部分网络带宽。所以使用 MGCBF 算法替代传统的网络流计数方式有较大优势。图 1 是基于 MGCBF 算法的网络长流统计和流信息维护模型的示意图，图上部的 MGCBF 数据结构用于维护一定时间段内到达的每一个流所对应的报文数目，下部通过哈希表和链表的结合维护需要采集流信息。具体流程如下：

1) 当一个报文到达时，使用 MGCBF 对其进行维护，并获得其所属流包含的报文数量；
2) 当其中某个流的报文数量到达 threshold 时，首先将相应的计数减小 threshold 值，然后通过一次 Hash 运算和相关流 ID 比较将其存储到一个哈希表+链表构成的数据结构中，如果该流信息已经被维护，则更新相关流信息；如果不存在则新建流信息；
3) 对存储在表中的流信息使用超时方式进行维护，规定一定的间隔扫描存储空间，及时地发现已经结束的流，并将相关的流信息写入永久存储设备。

通过设定适当 threshold 值，可以针对不同的应用需求对任意长度（报文数≥2）的流维护其相关信息，具有很好的可扩展性。随着 threshold 值的减小，MGCBF 所使用的层数也递减以减小算法的开销；当 threshold 减小到一定程度时（譬如，threshold≤10），MGCBF 就退化成单个 CBF。

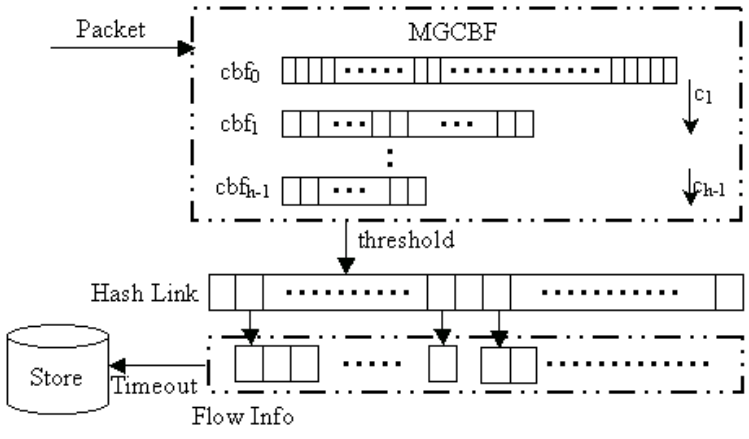

图 1　基于 MGCBF 算法的流长统计和信息维护模型

2.1 算法改进和优化

由于哈希函数固有冲突的存在，使用 MGCBF 计算集合中元素出现的次数时，不可避免会产生误差。本节主要介绍两种方法对 MGCBF 算法进行优化，以提高算法性能，减小算法的误差率。

(1) 定时更新方法（refresh periodically）

当待测集合 S 所包含的元素个数十分庞大时（如主干网中某个观测点一天内获取的报文数量），使用传统的 MGCBF 对集合 S 进行分析，所需维护向量空间 V 可能已经超过目前普通服务器所能承受的范围，即使可以承受，也会由于其性价比太低而没有实际使用的价值。

本文采用的方法是将集合S分为若干个子集，$S=\{S_1,S_2,\ldots,S_\gamma\}$，对每个子集$S_i$采用MGCBF算法进行统计计算，这样原有向量空间V被减小为V/γ，当完成一个子集的操作后初始化MGCBF的向量空间，为下一个子集操作做准备。由于在流测量中每个子集被定义为一段时间内到达的报文，所以这种方法被称之为定时更新。这样做的代价是割裂前后两个子集S_i和S_{i+1}元素之间的关系，从而导致了测量的误差，同时也增加系统计算开销，相关的误差分析和计算代价将在§2.2 中具体讨论。

(2) 重复最小值方法（recurring minimum）

重复最小值方法的基本思想是利用一个附加的CBF（CBF_t）来减小计数累加误差，当一个元素x_i到来时，如果它在原始CBF（CBF_p）中对应的计数器存在两个或两个以上的最小值，则将其加入到CBF_p中（称x_i存在重复最小值），否则将其加入到CBF_t中。假设$P(R_x)$是CBF_p出现重复最小值的比例，$P(E_x|R_x)$是在给定重复最小值的情况下误差的比例，E^s为CBF_t的计算误差，那么使用重复最小值方法的CBF的误差为：

$$E_{RM} = P(R_x)P(E_x \mid R_x) + (1 - P(R_x))E^s$$

文献[6]中给出了一个例子，CBF的各个参数设置如下：k=5，n=1000，m/n=k*ln(2)=0.7k，m^s=m/2，通过实验获得E_{RM}<E/18。在针对流长统计和信息维护模型的MGCBF算法实现中，只在高层CBF采用重复最小值的方法，因为高层CBF的误差对整个算法计数值的影响比较大，而且高层CBF的向量空间V和元素集合S也比较小，这样就可以尽量减小维护所需代价；而低层CBF由于向量空间和元素集合都很大，算法对其计数值误差的敏感性较小，所以使用传统的CBF比较合适。

2.2 算法的性能分析和误差估计

(1) 算法的性能分析

假设需要获取一段时间内报文数超过 1000 的流的信息（*Threshold*=1000），考虑性能和效率本文设计MGCBF为 2 层（*h*=2），在第二层CBF采用重复最小值的方法较小误差；由于Internet中流长分布呈重尾特性，报文小于 16 的流的数量要超过流总量 90%以上，所以如果对第一层CBF使用counter的计数值为 16（c_1=16），那么第二层所需的向量空间V_2可以缩小为V_1的 1/10（$m_2=m_1/10$）而保证不引入更多的误差；根据 §2.2 等式(1)，由于Threshold=1000，c_1=16，所以可以计算$MAX(min_1)=(Threshold-min_0)/c_1 = (1000-16)/16=61.5<64=2^6$，也就是设定$c_2$=64。依据Fan L, Cao P, Almeida J, et. al.在文献[7]中提出的建议，当一组无重复的元素插入到一个CBF中时，当取CBF的计数器大小为 16 也就是 4 比特时，就可以基本保证计数器不会因为累加而产生溢出：

$$Pr(\max(c)\geqslant 16) \leqslant 1.37\times 10^{-15}\times m$$

其中不等式左边为产生溢出的可能性，m为向量空间V的大小。定义第一层CBF的计数器大小为log(16)+4=8bit，第二层CBF计数器大小为$\log(2^6)$+4=10bit，那么MGCBF数据结构所需使用的空间为：

$$M_{MGCBF}=8\times m_1+10\times m_2+1/2\times(10\times m_2)=9.5m_1$$

而使用传统 CBF 维护流信息需要空间为：

$$M_{CBF} = (\log(1000)+4)m_1=14m_1$$

在 threshold 取值为 1000 的情况下，MGCBF 和 CBF 相比，所使用的空间节省了 1/3；如果采用 threshold 的取值继续增加时，可以在引入很小的计算复杂度和误差比例的情况下，节省更多的存储空间。

假设平均到达c_1个报文才更新一次cbf_1，但实际要高于这个比例（因为有较大一部分流长不能达到c_1）；对每个报文进行一次CBF插入或取出的计算时间为 τ，那么对流观测适用MGCBF对平均每个报文的处理时间为 $\tau_{MGCBF}<\tau_0+1/c_1\times\tau_1$。由于在高层为保证$cbf_1$的更高精度，设置$k_1=\alpha$

$k_0>k_0$，而且由于采用重复最小值方法使cbf_2的计算复杂度增加了大约 20％左右[6]，则 $\tau_1=1.2\alpha\tau_0$，那么MGCBF处理每个报文的平均计算开销为 $\tau_{MGCBF}<(1+1.2\alpha/c_1)\tau_0$。当参数$c_1=16$, $\alpha=4/3$ 时，$\tau_{MGCBF}<1.1\tau_0$，也就是说平均每个报文大约增加了不到 1/10 的计算开销。同理可以证明，随着层数的继续增加，平均每个报文增加的计算开销比例将以数量级减小。

(2) 算法误差估计

使用MGCBF算法的统计流信息所产生的误差可以分为两类：1）MGCBF算法所固有的误差；2）采用定时更新割裂两个子集间流信息关联所产生的误差。MGCBF算法产生误差比例可由各层分别计算：cbf_0为E_0，cbf_1为E_1，跟传统的CBF相比，只是增加了cbf_1的误差比例E_1，由于cbf_1由于采用重复最小值的方法，可以使所产生的误差比例$E_2<<E_1$，而总的误差为：$E_0+1/c_1\times E_1$，所以从整体来看MGCBF误差率与CBF相比，误差大概升高 1/288，从升高的比例来看是可以基本可以忽略不记。因此MGCBF算法所产生的误差比例基本上等于其第一层CBF（cbf_0）的误差比例。计算采用定时更新所导致的测量误差，假设$\eta(v,t)$是指其对应速率v为threshold/(To-t)的长流所占比例，单位时间到达长流的数量为s'，流的超时为To，则受影响的长流数目 $s_f^{'}$ 为：

$$s_f^{'}=\int_0^{+\infty}\int_0^{To}\eta(v,t)s'dtdv$$

对于绝大多数长流而言，只有在 $To-t<<To$ 时，$\eta(v,t)$才有可能大于 0，从整体来看 $s_f^{'}<<(s'\times To)$。在§3 中，通过对不同时段的 TRACE 的实验证明，由于定时更新而导致的第二类误差比例不到 1％。因此，本文提出的定时更新方法，可以在很少影响流识别精度的情况下，以少量的计算开销，将所需的存储空间控制很小的范围内。

3 实验结果

实验所使用的数据集分别来自于采集自 CERNET 某主干节点和 NLANR 公开提供的 TRACE：CERNET 和 CESCAI[10]，通过 5 元组和超时为 64 秒的定义方式[9]，经过统计获得数据如表 1 所示。

表 1. 实验使用 TRACE 流分布统计

	流总数	长流数（threshold ≥1000）	比例
CERNET	17164783	30316	0.18%
CESCAI	21146889	32955	0.16%

首先需要根据网络中单位时间内活跃的流数目来确定MGCBF算法的各个参数。一般来说在正常情况下网络中平均流长是固定的，在 5 元组和超时为 64 秒的定义下在CERNET和CESCAI中流长均为 20 左右，也就是说平均每个流包含 20 个报文，所以可以依据到达的报文数确定MGCBF各个向量空间V的大小。实验所使用的MGCBF的层数为 2，设定其各个参数如下：threshold=1000，$m_1=16\times10^6$, $m_2=16\times10^5$，$k_1=k_2=6$；对MGCBF采用定时更新的方法，当一段时间内报文数量达到 $n=m_1/8\times20=40\times10^6$时更新一次，也就是说在pps为 200×10^3时，定时更新的时间间隔大约为 3.3 分钟；对MGCBF的第二层采用重复最小值的方法进一步减小测量误差：$m^s=m_2/2$。

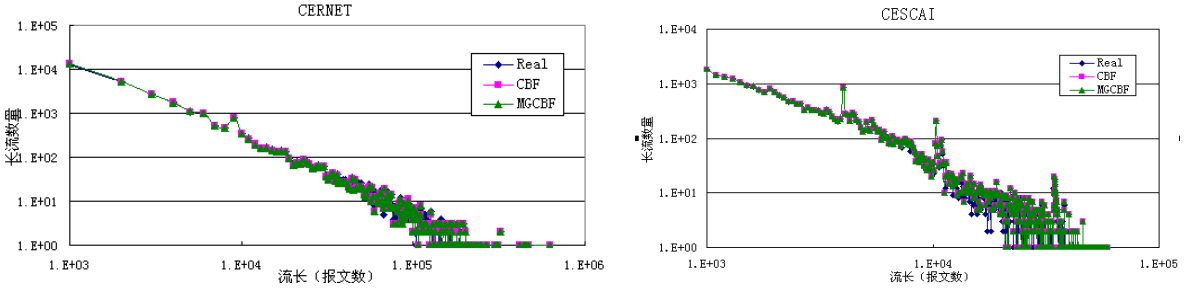

图 2　不同测量方法所得 threshold=1000 长流分布对比图

实验结果显示，使用 MGCBF 和 CBF 对 1000 个报文以上的长流估计和信息维护差别不到 1％（CERNET 为 0.26%，CESCAI 为 0.71%），而这些差别是由定时更新和第二层 CBF 的测量误差引起的。

从资源利用状况来看，由于任意时刻平均活动流的数目为 700,000 左右，传统流计数算法所需存储空间为 29.05×10^6字节；CBF算法所需要的存储空间为：224.03×10^6字节；而MGCBF算法

所需得存储空间为：19.32×10^6字节。由于哈希运算本身计算复杂性要远小于数值比较，所以在计算时间上，CBF和MGCBF算法要优于传统的流信息维护方法，而当后者的哈希映射不够均匀而导致链表长度比较大时，这种情况更为明显。因此基于MGCBF算法的长流测量方法在保证长流数目统计误差位于 2%以内的情况下，有效地提高了流信息维护效率，减小了所需使用的存储空间。

4 结论

本文针对网络中流长呈重尾分布的特点，提出了基于 MGCBF 算法的长流信息统计及其流维护模型。与目前流行的流信息维护方法相比，本模型可以在保证长流信息基本完整的情况下，以较小计算代价减小所需要的存储空间；同时也解决了其他一些流分布或估计算法不能维护相应流信息的问题。本文提出的模型具有很好的扩展性，可以用来对报文数大于 2 的所有流信息进行统计。本文所提出的 MGCBF 算法不仅可以应用在网络长流信息统计，还可以应用到其他相关领域，只要待处理的数据集中元素出现频度满足重尾分布。

参考文献(References)

[1] Brownlee N, Mills C, Ruth G. Traffic Flow Measurement: Architecture[P]. IETF, RFC 2722. 1999-10.

[2] Shaikh A, Rexford J, Shin K. G. Load-Sensitive Routing of Long-Lived IP Flows[A]. In: Proc. of the conference on Applications, technologies, architectures, and protocols for computer communication [C].New York: ACM Press,1999. 215–226

[3] Kim I, Reddy A. L. N. Analyzing Network Traces to Identify LongTerm High-Rate Flows[R]. In: Technical Repor TAMU. 2001.

[4] Cristian Estan, George Varghese: New directions in traffic measurement and accounting: Focusing on the elephants, ignoring the mice. ACM Trans. Comput. Syst. 2003, 21(3): 270-313

[5] Kumar A, Xu J, et al. Space-Code Bloom Filter for Efficient Per-Flow Traffic Measurement [C]. In: IEEE INFOCOM 2004 - The Conference on Computer Communications[M]. 2004. 1763-1774

[6] Bloom B. Space/Time trade-offs in hash coding with allowable errors [J]. *Commun. of ACM.* 1970, 13(7): 422-426

[7] Fan L, Cao P, Almeida J, et al. Summary Cache: A Scalable Wide-Area Web Cache Sharing Protocol [J]. *IEEE/ACM Transactions on Networking*. 2000, 8(3): 281-293.

[8] Cohen S, Matias Y. Spectral Bloom Filters[A]. In: Halevy AY, Ives ZG, Doan AH. Proc. of the 2003 ACM SIGMOD Int'l Conf. on Management of Data[C].San Diego： ACM Press, 2003.241-252.

[9] Claffy K.C, Braun H.W, Polyzos G.C. A Parameterizable Methodology for Internet Traffic Flow Profiling[J]. *In IEEE Journal on Selected Areas In Communications*.1995, 12(8):1481-1494.

[10] NLANR. CESCAI TRACE[EB/OL]. http://pma.nlanr.net/Special/cesc1.html. 2004-12-21/2005-8-15