References for Distributed Data Streams

August 20-23 2012 at MADALGO Center, Aarhus

Graham Cormode

1 Introduction to Distributed Data Streams

The topic of (single) streams of data has had much coverage: for more information, see other surveys such as [Mut05, BBD$^+$02, GGR02]. There has been less explicit attention given to distributed data streams, although many techniques from single streams naturally distribute. The survey of Muthukrishnan lays out the important data models for describing streams: arrivals only (cash-register model) and arrivals and departures (turnstile model) [Mut05]. Different models for distributed streaming include one-shot computation, gossip-based computation, and continuous computation. Early work defining the one-shot model is due to Gibbons and Tirthapura [GT01], and a formalization of the model in the context of the MapReduce computations is due to Feldman et al. [FMS$^+$08]. For the gossip model, see [KDG03], and for the continuous model, see the survey [Cor11] and later lecture.

Some initial examples of methods in the one-shot distributed model include distributed sampling [NGSA04]; entropy estimation [CCM07]; finding frequent items/heavy hitters [MG82, ACH$^+$12]; fingerprints for equality testing [MR95]; and Bloom Filters for set membership [BM04].

References

[ACH$^+$12] Pankaj Agarwal, Graham Cormode, Zengfeng Huang, Jeff Phillips, Zheiwei Wei, and Ke Yi. Mergeable summaries. In *ACM Principles of Database Systems*, 2012.

[BBD$^+$02] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. Models and issues in data stream systems. In *ACM Principles of Database Systems*, pages 1–16, 2002.

[BM04] Andrei Z. Broder and Michael Mitzenmacher. Network applications of Bloom filters: A survey. *Internet Mathematics*, 1(4), 2004.

[CCM07] A. Chakrabarti, G. Cormode, and A. McGregor. A near-optimal algorithm for computing the entropy of a stream. In *ACM-SIAM Symposium on Discrete Algorithms*, 2007.

[Cor11] Graham Cormode. Continuous distributed monitoring: A short survey. In *Algorithms and Models for Distributed Event Processing (AlMoDEP)*, 2011.

[FMS$^+$08] Jon Feldman, S. Muthukrishnan, Anastasios Sidiropoulos, Clifford Stein, and Zoya Svitkina. On distributing symmetric streaming computations. In *ACM-SIAM Symposium on Discrete Algorithms*, 2008.

[GGR02] M. Garofalakis, J. Gehrke, and R. Rastogi. Querying and mining data streams: You only get one look. In *ACM SIGMOD International Conference on Management of Data*, 2002.

[GT01] P. Gibbons and S. Tirthapura. Estimating simple functions on the union of data streams. In *ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, pages 281–290, 2001.

[KDG03] D. Kempe, A. Dobra, and J. Gehrke. Computing aggregates using gossip. In *IEEE Conference on Foundations of Computer Science*, 2003.

[MG82] J. Misra and D. Gries. Finding repeated elements. *Science of Computer Programming*, 2:143–152, 1982.

[MR95] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.

[Mut05] S. Muthukrishnan. *Data Streams: Algorithms and Applications*. Now Publishers, 2005.

[NGSA04] S. Nath, P. B. Gibbons, S. Seshan, and Z. R. Anderson. Synopsis diffusion for robust aggrgation in sensor networks. In *ACM SenSys*, 2004.

2 Sketch Data Structures and Concentration Bounds

Many results in distributed streaming arise in the context of sketch algorithms, which are analyzed using probabilistic concentration bounds [MR95, MU05]. A general survey on sketching techniques is given in [Cor11].

Given a (distributed) stream of items i, let f_i denote the total number of occurrences of i in the stream. The kth frequency moment F_k is defined as $F_k = \sum_i |f_i|^k$. Important cases are F_0, the number of items with non-zero frequency; F_2, the sum of the squares of the frequencies; and F_∞, the largest frequency. The Count-Min sketch relies only on the Markov inequality to provide accurate estimates of item frequencies and F_∞ [CM05a]. The AMS (Alon-Matias-Szegedy) sketch estimates the second frequency moment F_2 (the sum of squared frequencies), and is analyzed via the Chebyshev and Chernoff bounds [AMS96, CCFC02].

The problem of estimating the quantity F_0, also known as distinct counting, is studied in [FM83, Gib01]. An approach based on hashing and tracking the smallest observed hash values is due to Bar-Yossef et al. [BYJK$^+$02]. Motivated by distributed computation, generalizations of these data structures have been used to build duplicate-resilient aggregates in [CLKB04, CM05b]. *Range Efficient* problems occur when the input arrives as a series of ranges of values $[a \dots b]$, and must be processed efficiently [PT07, RD06].

References

[AMS96] N. Alon, Y. Matias, and M. Szegedy. The space complexity of approximating the frequency moments. In *ACM Symposium on Theory of Computing*, pages 20–29, 1996.

[BYJK$^+$02] Z. Bar-Yossef, T.S. Jayram, R. Kumar, D. Sivakumar, and L. Trevisian. Counting distinct elements in a data stream. In *Proceedings of RANDOM 2002*, pages 1–10, 2002.

[CCFC02] M. Charikar, K. Chen, and M. Farach-Colton. Finding frequent items in data streams. In *Procedings of the International Colloquium on Automata, Languages and Programming (ICALP)*, 2002.

[CLKB04] J. Considine, F. Li, G. Kollios, and J. Byers. Approximate aggregation techniques for sensor databases. In *IEEE International Conference on Data Engineering*, 2004.

[CM05a] G. Cormode and S. Muthukrishnan. An improved data stream summary: The Count-Min sketch and its applications. *Journal of Algorithms*, 55(1):58–75, 2005.

[CM05b] G. Cormode and S. Muthukrishnan. Space efficient mining of multigraph streams. In *ACM Principles of Database Systems*, 2005.

[Cor11] Graham Cormode. Sketch techniques for massive data. In Graham Cormode, Minos Garofalakis, Peter Haas, and Chris Jermaine, editors, *Synposes for Massive Data: Samples, Histograms, Wavelets and Sketches*, Foundations and Trends in Databases. NOW publishers, 2011.

[FM83] P. Flajolet and G. N. Martin. Probabilistic counting. In *IEEE Conference on Foundations of Computer Science*, pages 76–82, 1983. Journal version in *Journal of Computer and System Sciences*, 31:182–209, 1985.

[Gib01] P. Gibbons. Distinct sampling for highly-accurate answers to distinct values queries and event reports. In *International Conference on Very Large Data Bases*, pages 541–550, 2001.

[MR95] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.

[MU05] Michael Mitzenmacher and Eli Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.

[PT07] A. Pavan and Srikanta Tirthapura. Range-efficient counting of distinct elements in a massive data stream. *SIAM Journal on Computing*, 37(2):359–379, 2007.

[RD06] Florin Rusu and Alin Dobra. Fast range-summable random variables for efficient aggregate estimation. In *ACM SIGMOD International Conference on Management of Data*, 2006.

3 L_p Sampling and Stream Verification

L_p Sampling via sketches. L_p sampling is a recently introduced technique, which asks to draw a sample i with probability (approximately) proportional to $|f_i|^p/\|f\|_p^p$. Several methods to perform this sampling have been proposed [MW10, JST11]. For L_0 sampling (sample from the set of items with non-zero frequency), other methods are known [FIS05, CMR05]. This can be applied to build sketches for graph properties such as connectivity, bipartiteness, and sparsest cut [AGM12a, AGM12b]. Methods for L_2 sampling can also be applied to estimate F_k accurately for any $k \geq 3$ [CK04, AKO11].

Stream Verification. When a large computation is outsourced, it is desirable to have some way to check the correctness of the returned results, without having to duplicate the entire computation. This leads to the area of verification. Initial results in the streaming setting were provided by Chakrabarti et al. [CCM09] for vector and matrix computations, and for graph computations by Cormode et al. [CMT10]. These results have been generalized to multiple rounds of interaction between a prover and a verifier [CTY12]. A general result shows any NP problem has a proof protocol of polynomial size [GKR08], and this protocol has been implemented in the streaming setting [CMT12].

References

[AGM12a] Kook Jin Ahn, Sudipto Guha, and Andrew McGregor. Analyzing graph structure via linear measurements. In *ACM-SIAM Symposium on Discrete Algorithms*, 2012.

[AGM12b] Kook Jin Ahn, Sudipto Guha, and Andrew McGregor. Graph sketches: sparsification, spanners, and subgraphs. In *ACM Principles of Database Systems*, 2012.

[AKO11] Alexandr Andoni, Robert Krauthgamer, and Krzysztof Onak. Streaming algorithms via precision sampling. In *IEEE Conference on Foundations of Computer Science*, 2011.

[CCM09] Amit Chakrabarti, Graham Cormode, and Andrew McGregor. Annotations in data streams. In *International Colloquium on Automata, Languages and Programming (ICALP)*, 2009.

[CK04] Don Coppersmith and Ravi Kumar. An improved data stream algorithm for frequency moments. In *ACM-SIAM Symposium on Discrete Algorithms*, 2004.

[CMR05] G. Cormode, S. Muthukrishnan, and I. Rozenbaum. Summarizing and mining inverse distributions on data streams via dynamic inverse sampling. In *International Conference on Very Large Data Bases*, 2005.

[CMT10] Graham Cormode, Michael Mitzenmacher, and Justin Thaler. Streaming graph computations with a helpful advisor. In *European Symposium on Algorithms (ESA)*, 2010.

[CMT12] Graham Cormode, Michael Mitzenmacher, and Justin Thaler. Practical verified computation with streaming interactive proofs. In *Innovations in Theoretical Computer Science (ITCS)*, 2012.

[CTY12] Graham Cormode, Justin Thaler, and Ke Yi. Verifying computations with streaming interactive proofs. In *International Conference on Very Large Data Bases*, September 2012.

[FIS05] G. Frahling, P. Indyk, and C. Sohler. Sampling in dynamic data streams and applications. In *Symposium on Computational Geometry*, June 2005.

[GKR08] Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. Delegating computation: interactive proofs for muggles. In *ACM Symposium on Theory of Computing*, 2008.

[JST11] Hossein Jowhari, Mert Saglam, and Gábor Tardos. Tight bounds for lp samplers, finding duplicates in streams, and related problems. In *ACM Principles of Database Systems*, 2011.

[MW10] Morteza Monemizadeh and David P. Woodruff. 1-pass relative-error l_p-sampling with applications. In *ACM-SIAM Symposium on Discrete Algorithms*, 2010.

4 Continuous Distributed Monitoring

In the model of continuous distributed monitoring, k observers each see a stream of observations. Their goal is to work together to compute a function of the union of their observations. A longer survey of the area is given in [Cor11]. A first problem is to count when τ events have been observed can easily be solved with $O(k^2 \log \tau)$ messages [KCR06]. The dependence on k can be improved to k, and indeed made indpendent of k when randomization is allowed [CMY08]. A general "geometric approach" allows any function to be monitored in this setting based on breaking the global function into pieces which can be monitored locally [SSK06, SSK08].

Different approaches are needed when the goal is to extract a set of items from the input, such as in drawing a sample. This has resulted in elegant protocols for sampling [CMYZ10, TW11]. Other work has used different approaches to reduce the cost of protocols, such as building prediction models using sketches [CGMR05] and tracking only recent updates in a sliding window [CLLT10]. Applications to distributed gaming are discussed in [HM09].

References

[CGMR05] G. Cormode, M. Garofalakis, S. Muthukrishnan, and R. Rastogi. Holistic aggregates in a networked world: Distributed tracking of approximate quantiles. In *ACM SIGMOD International Conference on Management of Data*, 2005.

[CLLT10] Ho-Leung Chan, Tak-Wah Lam, Lap-Kei Lee, and Hing-Fung Ting. Continuous monitoring of distributed data streams over a time-based sliding window. In *Symposium on Theoretical Aspects of Computer Science (STACS)*, 2010.

[CMY08] Graham Cormode, S. Muthukrishnan, and Ke Yi. Algorithms for distributed, functional monitoring. In *ACM-SIAM Symposium on Discrete Algorithms*, 2008.

[CMYZ10] Graham Cormode, S. Muthukrishnan, Ke Yi, and Qin Zhang. Optimal sampling from distributed streams. In *ACM Principles of Database Systems*, 2010.

[Cor11] Graham Cormode. Sketch techniques for massive data. In Graham Cormode, Minos Garofalakis, Peter Haas, and Chris Jermaine, editors, *Synposes for Massive Data: Samples, Histograms, Wavelets and Sketches*, Foundations and Trends in Databases. NOW publishers, 2011.

[HM09] Kelly Heffner and Gregory Malecha. Design and implementation of generalized functional monitoring. `http://www.people.fas.harvard.edu/~gmalecha/proj/funkymon.pdf`, 2009.

[KCR06] R. Keralapura, G. Cormode, and J. Ramamirtham. Communication-efficient distributed monitoring of thresholded counts. In *ACM SIGMOD International Conference on Management of Data*, 2006.

[SSK06] I. Sharfman, A. Schuster, and D. Keren. A geometric approach to monitoring threshold functions over distributed data streams. In *ACM SIGMOD International Conference on Management of Data*, 2006.

[SSK08] I. Sharfman, A. Schuster, and D. Keren. Shape sensitive geometric monitoring. In *ACM Principles of Database Systems*, 2008.

[TW11] Srikanta Tirthapura and David P. Woodruff. Optimal random sampling from distributed streams revisited. In *DISC*, 2011.

5 Lower Bounds.

Since its introduction by Yao, communication complexity [Yao79, KN97] has proven to be a powerful technique for proving lower bounds for a variety of computation models. Two problems in randomized communication complexity that are "hard" (require $\Omega(n)$ communication) lead to many streaming lower bounds:
INDEX: Alice holds string x of length n, Bob holds index $y \in [n]$, and Bob must output the yth bit of x, given a single message from Alice [KN97].
DISJOINTNESS: Alice and Bob both hold strings x and y, and must determine whether there is any bit location where both strings are 1 [Raz92]. Hardness of these problems can also be shown via "information complexity", which considers the information transmitted by any correct protocol [BYJKS04].

In the GAPHAMMING problem, Alice and Bob both have strings with the promise that their strings have Hamming distance either less than $N/2 - \sqrt{N}$ or greater than $N/2 + \sqrt{N}$. Via reductions, the hardness of this problem is also $\Omega(N)$ [JKS08]. The linear hardness of GAPHAMMING can show hardness of frequency moments of $\Omega(\epsilon^{-2})$ [IW03, Woo04]; and $\Omega(\epsilon^{-2}/\log(1/\epsilon))$ for entropy estimation [CCM07]. Thus the algorithms for these problems are essentially optimal in terms of their dependency on ϵ.

References

[BYJKS04] Ziv Bar-Yossef, T. S. Jayram, Ravi Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *J. Comput. Syst. Sci.*, 68(4):702–732, 2004.

[CCM07] A. Chakrabarti, G. Cormode, and A. McGregor. A near-optimal algorithm for computing the entropy of a stream. In *ACM-SIAM Symposium on Discrete Algorithms*, 2007.

[IW03] P. Indyk and D. Woodruff. Tight lower bounds for the distinct elements problem. In *IEEE Conference on Foundations of Computer Science*, 2003.

[JKS08] T. S. Jayram, Ravi Kumar, and D. Sivakumar. The one-way communication complexity of hamming distance. *Theory of Computing*, 4(1):129–135, 2008.

[KN97] E. Kushilevitz and N. Nisan. *Communication Complexity*. Cambridge University Press, 1997.

[Raz92] A. A. Razborov. On the distributional complexity of disjointness. *Theoretical Computer Science*, 106(2):385–390, 1992.

[Woo04] D. Woodruff. Optimal space lower bounds for all frequency moments. In *ACM-SIAM Symposium on Discrete Algorithms*, pages 167–175, 2004.

[Yao79] Andrew Chi-Chih Yao. Some complexity questions related to distributive computing (preliminary report). In *ACM Symposium on Theory of Computing*, pages 209–213, 1979.