[流计算背景]

流计算最初出现在20世纪末，是为了解决数据量大，数据不能完全保存在内存中的计算问题。通过将数据集转化为数据流，就可以依次处理流过的数据，而无需在内存中存储数据集，这样就解决了数据量大的问题。一开始，能够解决的问题领域也比较有限，主要都是针对数据流的计算。数据流上的算法研究成果有很多，如computation of frequency moments [1]，histograms[2]，Wavelet transforms[3]，distinct elements count[4]等

由于图数据和图算法的复杂性，图流计算晚几年才出现。到目前为止，图流上进行的图算法有Connectivity，Triangle Count[5]，PageRank[6]，Matching等

无论是图流计算还是数据流计算，研究都集中在估计计算。估计计算主要有以下优点：1. 可以不存或者存储少量的数据，相对于整个图数据来说。这点对于Internet和社交网络等方面的分析有重大意义，因为它们的顶点和边数量庞大，甚至可能存在264条边，完全存储数据集是不现实的。2. 更新时间短，流算法对每个流数据要进行相应的计算和更新，而采用估计的流算法更新和计算时间短。3. 一些问题没有必要进行准确的计算，只要结论正确即可，比如connectivity，triangle count，pagerank等。就拿triangle count来说，它的意义在于社区发现，判断社区是否稳定，而一个大致准确的triangle数可以和一个准确的triangle数得到相同的结论。4. 估计算法的准确性可以接受，通过对估计算法的改进，可以得到让人比较满意的准确性。在分析流计算估计模型之前，我们先看一下流模型。

[流模型]

依据流中数据的表达形式，主要有两种典型的模型[7]：

1. Cash Register Model : 流中的每一项仅仅是数据集中一项，比如在distinct elements count 中，每一项就是一个数。数据集中的每一项以任意顺序形成数据流。
2. Turnstile Model : 在该模型中，我们有一个初始化为空的集合D，流中的数据由两项组成，一项是数据集的某一项，另一项是一个标志位，可以对集合D进行动态改变。

例如，流图中的每一项为(x, U)，如果U为＋，就将x加入D，如果U为－，就将x从D删除。这种模型更符合现实的一些场景，近期的参考文献也将这种模型成为动态流(dynamic stream)。

[流计算模型]

现在我们转向流的估计计算模型，下列模型主要是针对图流进行分析，当然一些模型可以应用于数据流，这里不细区分。依据模型是否存储流中的数据，将模型分为两类：

1. 采样(sampling) : 该模型完全不存储流中的数据，在流经过时，对流数据进行采样和计算。该模型的内存消耗主要在采样线程上，要采n个样本，就要起n个线程对流进行采样，即每个线程只能采一个样。依赖该模型的算法的最终计算结果，取决于图结构和采样结果。例如，采用该模型的典型算法为triangle count，见[5]，该论文定义了两类图流arbitrary stream和incidence stream，并分别提出了1-pass和3-pass采样算法，以arbitrary stream的1-pass为例，该算法可以通过令采样次数，使得的概率为，其中，，
2. 概要(summarization) : 该模型通过将图结构转化为简单的数据结构，保存图中元素，使得消耗内存量远远小于原图。同时，结构随数据流进行更新。依据做概要的方式，可以将该模型分为三类：
3. 生成树(spanner) [8, 9]: 针对图流来说，该模型仅保留边的一个集合(set)，即将图结构转化为集合(set)，可以用于判断图的连通性(connectivity)和图中任意两点的距离。
4. 稀疏图(sparsifier) [10, 11]: 针对图流来说，该模型仅保留边的一个权重矩阵，即将图结构转化为矩阵，可以用于估计图中每个连通分量(connected components)的权重。
5. 草图(sketch) : 该模型又分为线性草图(linear sketch)[12, 13]和同构草图(homomorphic sketch)[14]，针对图流来说，线性草图仅保留点的一个向量和边的一个向量，即将图结构转化向量，因为丢失了图结构，线性草图支持的查询有限，如边权重和点的入度等；同构草图保留多个顶点矩阵，即将图结构转化为多个矩阵，保留了图结构，可以支持的查询有顶点查询，边查询，路径查询和子图查询等。

基于流的特征，我们可以将流图计算扩展到实时计算上，基本上上述的模型与算法均可以应用在实时计算，除了multi-pass的算法。这样，就可以满足现代社会日益增长的实时计算的需求，研究有意义。

[参考文献]

[1] Noga Alon, Yossi Matias, and Mario Szegedy, The space complexity of approximating the frequency moments. J. Comput. Syst. Sci. 58 (1999), no. 1, 137–147.

[2] Sudipto Guha, Nick Koudas, and Kyuseok Shim, *Data-streams and histograms*, ACM Symposium on Theory of Computing, 2001, pp. 471–475.

[3] Anna C. Gilbert, Yannis Kotidis, S. Muthukrishnan, and Martin Strauss, *Surfing wavelets on streams: One-pass summaries for approximate aggregate queries.* Proc. of the 27th VLDB, 2001, pp. 79–88.

[4] Ziv Bar-Yossef, T. S. Jayram, Ravi Kumar, D. Sivakumar, and Luca Trevisan. Counting distinct elements in a data stream. In *6th Annual European Symposium (ESA’02)*, pages 1–10, 2002.

[5] Luciana S. Buriol, Gereon Frahling, Stefano Leonardi, Alberto Marchetti-Spaccamela, and Christian Sohler. Counting triangles in data streams. In *25th Symposium on Principles of Database*

*Systems (PODS’06)*, pages 253–262, 2006.

[6] Atish Das Sarma, Sreenivas Gollapudi, and Rina Panigrahy. Estimating pagerank on graph streams. *J. ACM*, 58(3):13, 2011.

[7] S. Muthukrishnan. Data Streams: Algorithms and Applications. *Foundations and Trends in Theoretical Computer Science*, 1(2), 2005.

[8] S. Baswana. Streaming algorithm for graph spanners – single pass and constant processing time per edge. *Inf. Process. Lett.* 106(3):110–114, 2008.

[9] M. Elkin. Streaming and fully dynamic centralized algorithms for constructing and maintaining sparse spanners. *ACM Transactions on Algorithms*, 7(2):20, 2011.

[10] A. A. Bencz′ur and D. R. Karger. Approximating *s-t* minimum cuts in ˜*O*(*n*2) time. In *ACM Symposium on Theory of Computing*, pages 47–55, 1996.

[11] D. A. Spielman and S.-H. Teng. Spectral sparsification of graphs. *SIAM J. Comput.*, 40(4):981–1025, 2011.

[12] G. Cormode and S. Muthukrishnan. An improved data stream summary: the count-min sketch and its applications. J. Algorithms, 55(1):58–75, 2005.

[13] P. Zhao, C. C. Aggarwal, and M. Wang. gSketch: On query estimation in graph streams. PVLDB,

5(3):193–204, 2011.

[14] Nan Tang, Qing Chen, Prasenjit Mitra. Graph Stream Summarization: From Big Bang to Big Crunch. SIGMOD '16 Proceedings of the 2016 International Conference. pages 1481-1496, 2016