On Estimating Path Aggregates over Streaming Graphs

Sumit Ganguly and Barna Saha

Indian Institute of Technology, Kanpur
{sganguly,barna}@cse.iitk.ac.in

Abstract. We consider the updatable streaming graph model, where edges of a graph arrive or depart in arbitrary sequence and are processed in an online fashion using sub-linear space and time. We study the problem of estimating aggregate path metrics P_k defined as the number of pairs of vertices that have a simple path between them of length k. For a streaming undirected graph with n vertices, m edges and r components, we present an $\tilde{O}(m(m-r)^{-1/4})$ space[1] algorithm for estimating P_2 and an $\Omega(\sqrt{m})$ space lower bound. We show that estimating P_2 over directed streaming graphs, and estimating P_k over streaming graphs (whether directed or undirected), for any $k \geq 3$ requires $\Omega(n^2)$ space. We also present a space lower bound of $\Omega(n^2)$ for the problems of (a) deterministically testing the connectivity, and, (b) estimating the size of transitive closure, of undirected streaming graphs that allow both edge-insertions and deletions.

1 Introduction

The data streaming model has gained popularity as a computational model for a variety of *monitoring* applications, where, data is generated rapidly and continuously, and must be analyzed very efficiently and in an online fashion using space that is significantly sub-linear in the data size. An emerging class of monitoring applications is concerned with *massive dynamic graphs*. For example, consider the dynamic web graph, where nodes are web-pages and edges model hyperlinks from one page to another. The edges in the web-graph are generated in a streaming fashion by web-crawlers [8]. Significant changes in the size, connectivity and path properties of web-communities of interest can be glimpsed by computing over these stream of edges. Another example is the citations graph [7], where, nodes are published articles and directed edges denote a citation of one article by another. Consider the query: find the top-k *second-level* frequent citations, where the second-level citation number of an article A is the number of (distinct) articles C that cite an article B that cite A.

[1] $f(m)$ is said to be $\tilde{O}(g(m))$ if $f(m) = O(\frac{1}{\epsilon^{O(1)}}(\log m)(\log n)(\log \frac{1}{\delta})^{O(1)}g(n))$. Similarly, $f(m)$ is said to be $\tilde{\Omega}(g(m))$ if $g(m)$ is $\tilde{O}(f(m))$.

Graph Streaming Models. In the *updatable* edge-streaming model of graphs, the stream is viewed as a sequence of tuples of the form $(u, v, +)$ or $(u, v, -)$, corresponding, respectively, to the insertion or the deletion of the edge (u, v). In the updatable model, once an edge (u, v) is inserted, it remains *current* in the graph until a tuple of the form $(u, v, -)$ appears in the stream to delete the edge. The current state of the graph $G = (V, E)$ is defined by the set of current edges E; the set of vertices V are those vertices that are incident to any of the current edges. Multi-graphs are modelled by allowing an edge to be inserted multiple times. Edges may be inserted and deleted in arbitrary order; however, an edge may be deleted at most as many times as it is inserted. The *insert-only* streaming model [1, 3, 7] only allows tuples of the form $(u, v, +)$ to appear in the stream. Graph streaming models that allow use of external memory and extra passes over stored data have been proposed—these include the semi-streaming graph model [2] and the W-stream model [1]. In this paper, we do not consider computational models over streaming graphs that allow multiple passes.

Path Aggregates. The path aggregate P_k is defined as the number of pairs of vertices (u, v) such that there is a simple path of length k from u to v. In this work, we consider the problem of estimating the path aggregate P_k, for $k \geq 2$ over updatable streaming graphs. The continuous monitoring of path aggregates enables online detection of changing path properties of a dynamic graph. For example, an article can be said to be frequently cited at level l, provided, the number of its level l-citations exceeds P_l/s, for a parameter s. The problem also has applications in database query size estimation. For example, let $R(A, B)$ be a binary relation over attributes A and B, over the same domain. Then, the P_2 over the binary relation R viewed as a graph represents the number of distinct pairs in the self-join (the *distinct self join*) of its relations.

Prior work in estimating path aggregates. [5] presents the JDSKETCH algorithm for estimating the *Join-Distinct size* of two data streams, $R = R(A, B)$ and $S(B, C)$ defined as $\text{JD}(R, S) = |\pi_{A,C}(R \bowtie S)|$. If $R = S$, then, $\text{JD}(R, R) = P_2$ and therefore, the JDSKETCH algorithm can be used to estimate P_2. The space requirement of the JDSKETCH algorithm is $\tilde{O}(m^2/P_2)$ [5]. In particular, for complete bi-partite graphs, chain graphs, etc., the JDSKETCH requires $\Omega(m)$ space.

Contributions. We present the *LDRS* algorithm for estimating P_2 for undirected streaming graphs and multi-graphs to within accuracy factors of $1 \pm \epsilon$ and confidence $1 - \delta$, where, $0 < \epsilon, \delta < 1$. For a graph with n vertices, m edges and r-components, the algorithm requires $O(\frac{1}{\epsilon^2} \frac{m}{(m-r)^{-1/4}} (\log n)(\log \frac{1}{\delta}))$ bits. For graphs with $\frac{m}{2}$ or less components, the space complexity of the algorithm is $\tilde{O}(m^{3/4})$ bits. We present a lower bound of $O(\sqrt{m})$ bits for estimating P_2 for undirected and connected streaming graphs. For directed streaming graphs, we show that the estimating P_k, for any $k \geq 2$, to within any approximation factor, requires $\Omega(m)$ bits of space. We also show that estimating P_k, for $k \geq 3$, for undirected streaming graphs to within a factor of $1 \pm \frac{3}{4}$, requires $\Omega(n^2)$ bits of

space. Finally, we present a space lower bound of $\Omega(n^2)$ for the problems of (a) deterministically testing the connectivity, and, (b) estimating the size of transitive closure, of undirected streaming graphs that allow both edge-insertions and deletions.

Organization. Section 2 presents the *LDRS* algorithm for estimating P_2 and Section 3 presents lower bound results.

2 Estimating P_2

In this section, we present the *RS* algorithm for estimating P_2 for *undirected* graphs and multi-graphs. We first consider insert-only streaming graphs and prove Theorem 1 and then generalize it to updatable edge streaming graphs.

Theorem 1. *For $0 \leq \epsilon < \frac{1}{6}$ and $0 < \delta < 1$, there exists an algorithm that takes as input an insert-only streaming graph with r components, m edges and n vertices and returns an estimate $\hat{P}_2$ satisfying $\mathsf{Pr}\{|\hat{P}_2 - P_2| \leq \epsilon P_2\} \geq 1 - \delta$ using $O(\epsilon^{-2} m(m-r)^{-1/4}(\log \frac{1}{\delta})(\log n))$ bits.*

2.1 Random Subgraph *RS* of graph streams

Given a graph $G = (V, E)$, the random subgraph *RS* is obtained by sampling the *vertices* of V uniformly and independently with probability p, and storing the adjacency list of each sampled vertex . We now design an *adaptive RS* structure for streaming graphs, given a *sampling probability function* $p(m)$ (for e.g., $p(m) = \frac{1}{\sqrt{m}}$) and space function $s = s(m) = 8mp(m)$.

Data Structure. The *current level counter* l_{curr} is initialized to 1 and takes increasing values between 1 and $\log|F|$. The *current sampling probability*, denoted by p_{curr}, is given by $p_{\text{curr}} = 2^{-l_{\text{curr}}+1}$. The current upper limit on the number of edges of the graph is given by m_{curr} that is initialized to $O(1)$. We maintain the invariant that $m_{\text{curr}} = \max(4m, O(1))$. The value of m_{curr} is doubled periodically as necessary. The counter s_{curr} denotes the *current space* provided to the portion of the data structure that stores the adjacency list of the sampled vertices. The invariant $s_{\text{curr}} = s(m_{\text{curr}})$ is maintained. Let S denotes the actual space (in words) used to store the adjacency lists of the sampled vertices and is initialized to 0. The set V_l stores the current set of sampled vertices. For every vertex in V_l, its adjacency list is also stored. The value of m is tracked by the data structure. This can be done exactly for simple graphs; for multi-graphs, an ϵ-approximation to the number m of distinct edges m can be tracked using space $O(\frac{1}{\epsilon^2}(\log n)(\log \frac{1}{\delta}))$ using a standard technique for counting the number of distinct items in a data stream [4, 6].

Let $e = \{u, v\}$ be an incoming streaming edge. The set of vertices that are adjacent to a given vertex $u \in V$ is denoted by $adj(u)$. If $u \in V_l$, then we add v to $adj(u)$. If $u \notin V_l$, then, we insert u into V_l with probability p_{curr} and initialize

$adj(u)$ as $\{v\}$. If u is not sampled, then, no further action is taken. The procedure is repeated for v independently and the space incurred S is incremented suitably. After processing an incoming edge, we check whether $S < s_{\text{curr}}$, that is, whether there is room for further insertions. If not, then, we perform a sub-sampling operation, if $m < \frac{m_{\text{curr}}}{2}$, or, increase available space, if $m \geq \frac{m_{\text{curr}}}{2}$. In the former case, we *sub-sample*, that is, the sampling probability p_{curr} is halved and for every $u \in V_l$, we retain u and its adjacency list with probability 1/2 (and, otherwise, u and and its adjacency list are dropped). In the latter case, if $m \geq \frac{m_{\text{curr}}}{2}$ and $S = s_{\text{curr}}$, then, we increase the available space from s_{curr} to $s_{\text{curr}} = s(2m_{\text{curr}})$ and update $m_{curr} = 2m_{curr}$.

Analysis. It is quite straightforward to see that the algorithm maintains the following invariants: $s_{\text{curr}} = s(m_{\text{curr}})$ and $m_{\text{curr}} \leq \max(O(1), 4m)$. The first invariant holds at initialization and at all subsequent space increases. Therefore, space used (in words) is $S = O(s_{\text{curr}}) = O(s(m_{\text{curr}})) = O(s(4m)) = O(s(m))$, since, $s(m)$ is a sub-linear function, and, therefore, $s(4m) \leq 4s(m)$.

For $u \in V$, define an indicator variable y_u that is 1 iff $u \in V_l$ and is 0 otherwise. The space used by the data structure (in words of size $\log n$ bits) is $S = \sum_{u \in V} \deg(u) y_u$. Thus, $\mathsf{E}\big[S\big] = \sum_{u \in V} \deg(u) \mathsf{Pr}\{y_u = 1\} = (2m)p_{\text{curr}}$. By Markov's inequality, $\mathsf{Pr}\{S \leq 4\mathsf{E}\big[S\big]\} = \mathsf{Pr}\{S \leq 8mp_{\text{curr}}\} \geq \frac{3}{4}$. Therefore, $\mathsf{Pr}\{p_{\text{curr}} \geq \frac{S}{8m}\} = \mathsf{Pr}\{S \leq 8mp_{\text{curr}}\} \geq \frac{3}{4}$. In view of this calculation, we keep $s_2 = O(\log \frac{1}{\delta})$ independent copies of the data structure. Suppose we call the current state of the data structure as *concise* if $p_{\text{curr}} \geq \frac{S}{8m}$. At the time of inference, we consider only the *concise* copies, obtain estimates of P_2 from the concise copies and return the median of these estimates. By Chernoff's bounds, the number of concise copies is $O(\log \frac{1}{\delta})$ with probability $1 - \frac{\delta}{2}$. The space requirement is $O(m \cdot p(m)(\log \frac{1}{\delta})(\log n))$. The above data structure can be *extended to updatable streaming graphs* using a combination of existing data structures [9].

Estimator. An estimate $\hat{P}_2$ is obtained from a concise copy of the RS structure with sampling probability $p = p(m)$ as follows. Let EP_2 denote the number of unordered vertex pairs u and v that are both sampled and have a common neighbor.

$$\hat{P}_2 = \frac{1}{p^2} EP_2 = \frac{1}{p^2} |\{\{u, v\} \mid u, v \in V_l \text{ and } adj(u) \cap adj(v) \neq \phi\}|$$

Finally, we return the median of $t = O(\log \frac{1}{\delta})$ independent estimates.

2.2 Analysis: Graph Based Properties of P_2

For an undirected simple graph $G = (V, E)$ and a vertex $u \in V$, let $\deg(u)$ denote the degree of u in G and let $\deg_2(u)$ denote the number of vertices in $V - \{u\}$ that can be reached from u in two hops.

Lemma 2. *In any graph $G = (V, E)$, $\deg_2(u) \leq (4P_2)^{3/4}$.*

Proof. Let r denote $\deg(u)$ and let T be the set of vertices, not including u, that can be reached from u in two hops. Let $s = |T|$. The vertices adjacent to u contribute $A = \binom{r}{2}$ to P_2. Let B denote the contribution to P_2 by vertex pairs in T. For each fixed value of s, B is minimized if each vertex of $adj(u)$ has either $\lceil \frac{s}{r} \rceil$ or $\lfloor \frac{s}{r} \rfloor$ neighbors in T and no two vertices of $adj(u)$ has any common neighbor (except u). Therefore, $B \geq r\binom{s/r}{2}$. Since, each vertex pair may be counted at most twice, that is once in both A and B, $P_2 \geq \frac{1}{2}(A+B) \geq \frac{1}{2}\binom{r}{2} + \frac{r}{2}\binom{s/r}{2}$. The expression in the *RHS* attains a minimum at $r \approx \frac{s^{2/3}}{2^{1/3}}$ and the corresponding minimum value of P_2 is greater than $\frac{s^{4/3}}{4}$. Thus, $s = \deg_2(u) \leq (4P_2)^{3/4}$. □

Lemma 3 presents a lower bound on the value of P_2 for simple undirected graph.

Lemma 3. *For a connected graph $G = (V, E)$ such that $|E| = m$, $P_2 \geq m - \sqrt{m}$. For a graph with r components, $P_2 \geq m - \sqrt{mr}$.*

Proof. We first show, by induction, that for a connected graph $G = (V, E)$ with m edges, $P_2 \geq m - \sqrt{m}$. Base Case: A connected graph G with one edge, that is, $m = 1$, $0 = P_2 \geq 1 - \sqrt{1} = 0$.

Induction Case. Suppose that the statement of the theorem holds true for graphs with number of edges between 1 and $m-1$. Consider a connected graph G with m edges. Let x be a lowest degree vertex among all vertices in the connected graph G that are not cut-vertices and let $\deg(x) = q$. (Note that in any graph G, the end vertices of any longest path are not cut-vertices; hence, we can always find x.). Let $y_1, y_2, \ldots, y_q$ denote the neighbors of x. Let $z_1, z_2, \ldots, z_s$ be the set of neighboring vertices of $y_1, \ldots, y_q$, not including x.

Suppose $s \geq q$. Since x is not a cut-vertex of G, deleting x from G leaves G connected. In the resulting graph, G', there are $m-q$ edges, and therefore, by the induction hypothesis, $P_2(G') \geq m - q - \sqrt{m-q}$. In G, x is connected by a path of length 2 to $z_1, z_2, \ldots, z_s$ respectively. Therefore, $P_2 \geq m - q - \sqrt{m-q} + s \geq m - \sqrt{m}$, since, $s \geq q$.

Suppose $s < q$. We first claim that none of $y_1, y_2, \ldots, y_q$ are cut-vertices. To prove this, suppose that y_j is a cut-vertex. Then, by removing y_j from G, $G - \{y_j\}$ has two or more components. Thus, in $G - \{y_j\}$, there is a z_k that is in a different component than x and z_k is adjacent to y_j. The component in $G - \{y_j\}$ that contains x also contains $y_1, \ldots, y_{j-1}, y_{j+1}, \ldots, y_q$. Therefore, there is no edge between y_i and z_k, for, $1 \leq i \leq q, i \neq j$ or between x and z_k. Thus, among the y_i's, z_k is attached only to y_j. Continuing this argument, we can show that if $y_{j_1}, y_{j_2}, \ldots, y_{j_p}$ are cut-vertices in G, then, there exist vertices $z_{k_1}, z_{k_2}, \ldots, z_{k_p}$ distinct from each other such that z_{k_r} is attached to y_{j_r} only and to none of the other y_i's or to x.

Not all of the y_i's can be cut vertices, since, this implies that the number of z_k's is at least q, which contradicts the assumption that $s < q$. Therefore, there exists at least one of the y_i's that is not a cut-vertex, say y_a. Suppose further that there is at least one cut-vertex y_j. Let y_j be attached to z_k such that z_k and x lie in different components in the graph $G - \{y_j\}$. Consider the degree of y_a. It is attached to x and is not attached to z_k. Therefore, $\deg(y_a) \leq 1 + (s-1) = s$.

Since, $s < q$, $\deg(y_a) < q = \deg(x)$. By assumption, x is the vertex with the smallest degree among all vertices that are not cut-vertices in G. Since, y_a is not a cut-vertex, and $\deg(y_a) < \deg(x)$, this is a contradiction. Thus, the only conclusion possible is that none of the y_i's are cut-vertices, proving the claim.

Further, since, none of the vertices y_i are cut-vertices, their degree is at least $\deg(x) = q$. Therefore, other than x, each y_i is connected to at least $q-1$ of the z_i's. Since $s < q$, this implies that $s = q-1$, and each of $y_1, y_2, \ldots, y_q$ is attached to each of x and $z_1, z_2, \ldots, z_{q-1}$. The subgraph of the y_i's in one partition and the z_j's and x in the other partition (y_i's and z_j's are disjoint, otherwise $s \geq q$, since G is a simple graph) is the complete bi-partite subgraph $K_{q,q}$. If there are no other edges in the graph, then, we can calculate m and P_2 for $K_{q,q}$ as follows.

$$m = q^2,\ P_2 = q(q-1),\ \text{and}\ P_2 = m - \sqrt{m}$$

which satisfies the statement of the lemma.

Suppose there are edges in addition to the $K_{q,q}$ subgraph formed above. Note that since, $s = q-1$, if there is any edge in the graph G other than the $K_{q,q}$ subgraph, then, there must be an edge attaching some z_k to some vertex u (since, vertices x and $y_1, \ldots, y_q$ are saturated with respect to degree). The vertex u is neither x nor one of $y_1, \ldots, y_q$. We now remove the vertex y_1 from G. The reduced graph G' is still connected since y_1 was not a cut-vertex and has $m - \deg(y_1) = m - q$ edges. Therefore, by the induction hypothesis, $P_2(G') \geq m - q - (m-q)^{1/2}$. In G, y_1 is at distance 2 from each of $y_2, \ldots, y_q$. In addition, y_1, by virtue of the edges (y_1, z_k) and (z_k, u), has a path of length 2 to u. Therefore, $\deg_2(y_1) \geq q-1+1 = q$. Thus, $P_2 \geq (m-q)-(m-q)^{1/2}+q \geq m-\sqrt{m}$.

We can now prove Lemma 3. Let m_c denote the number of edges of component number c, $1 \leq c \leq r$. Since, each component is connected, therefore, $P_2 \geq \sum_{c=1}^{r}(m_c - \sqrt{m_c}) \geq r\left(\frac{m}{r} - \sqrt{\frac{m}{r}}\right) = m - \sqrt{rm}$. □

2.3 Analysis: Space usage of the estimator

For $u \in V$, define an indicator random variable x_u such that $x_u = 1$ iff $u \in V_l$.

Lemma 4. $\mathsf{E}\big[EP_2\big] = p^2 P_2$ *and* $\mathsf{E}\big[\hat{P}_2\big] = P_2$.

Proof. $EP_2 = \sum_{\{u,v\} \in P_2} x_u x_v$. So, $\mathsf{E}\big[EP_2\big] = p^2 P_2$ and $\mathsf{E}\big[\hat{P}_2\big] = \mathsf{E}\big[\frac{EP_2}{p^2}\big] = P_2$. □

Lemma 5. $\mathsf{Var}\big[\hat{P}_2\big] = \frac{P_2}{p^2} + \frac{1}{2p}\sum_{u \in V} \deg_2^2(u)$.

Proof. Since, $EP_2 = \sum_{\{u,v\} \in P_2} x_u x_v$,

$$EP_2^2 = \Big(\sum_{\{u,v\} \in P_2} x_u x_v\Big)^2 = \sum_{\{u,v\} \in P_2} x_u x_v + \sum_{\substack{\{u,v\} \in P_2 \\ \{u,v'\} \in P_2 \\ v \neq v'}} x_u x_v x_{v'} + \sum_{\substack{\{u,v\} \in P_2 \\ \{u',v'\} \in P_2 \\ \{u,v\} \cap \{u',v'\} = \phi}} x_u x_v x_{u'} x_{v'}$$

Taking expectations,

$$\mathsf{E}\big[EP_2^2\big] \le p^2P_2 + \sum_{u\in V}\sum_{\substack{v\in adj(u)\\ v'\in adj(u)\\ v\neq v'}} p^3 + \sum_{\substack{\{u,v\}\in P_2\\ \{u',v'\}\in P_2\\ \{u,v\}\cap\{u',v'\}=\phi}} p^4$$
$$\le p^2P_2 + p^3\sum_{u\in V}\binom{\deg_2(u)}{2} + (p^2P_2)^2 \ .$$

Using Lemma 4,

$$\mathsf{Var}\big[EP_2\big] = \mathsf{E}\big[EP_2^2\big] - (\mathsf{E}\big[EP_2\big])^2 \le p^2P_2 + p^3\sum_{u\in V}\binom{\deg_2(u)}{2} \ .$$

So, $\mathsf{Var}\big[\hat{P}_2\big] = \mathsf{Var}\big[\frac{EP_2}{p^2}\big] = \frac{1}{p^4}\mathsf{Var}\big[EP_2\big] < \frac{P_2}{p^2} + \frac{1}{2p}\sum_{u\in V}\deg_2^2(u)$. □

Lemma 6. $\mathsf{Pr}\{|\hat{P}_2 - P_2| > \epsilon P_2\} < \frac{2}{9}$, *if* $p \ge \max(\frac{3}{\epsilon\sqrt{P_2}}, \frac{6}{\epsilon^2P_2^2}\sum_{u\in V}\deg_2^2(u))$.

Proof. By Chebychev's inequality, $\mathsf{Pr}\{|\hat{P}_2 - \mathsf{E}\big[\hat{P}_2\big]| > \epsilon P_2\} \le \frac{\mathsf{Var}\big[\hat{P}_2\big]}{25\epsilon^2P_2^2} < \frac{1}{p^2\epsilon^2P_2} + \frac{\sum_{u\in V}\deg_2^2(u)}{\epsilon^2P_2^2p} + \frac{2}{25} < \frac{1}{9} + \frac{1}{9}$. □

Lemma 7. *Let G have r components, m edges and n vertices. Then,* $\mathsf{Pr}\{|\hat{P}_2 - P_2| \le 6\epsilon P_2\} \ge 1-\delta$. *The space requirement is* $O(\frac{m}{\epsilon^2(m-r)^{1/4}}(\log\frac{1}{\delta})(\log n))$ *bits, with probability* $1-\delta$.

Proof. The space requirement is $O(mp)$, where, by Lemma 6, $mp = O(\max(\frac{m}{\epsilon^2\sqrt{P_2}}, \frac{m}{\epsilon^2P_2^2}\sum_{u\in V}\deg_2^2(u))))$. By Lemma 3, $P_2 \ge m - \sqrt{rm}$. Therefore, $\frac{m}{\epsilon\sqrt{P_2}} = \frac{m}{\epsilon(m-\sqrt{rm})} = \frac{m^{1/2}}{(m-r)^{1/2}}$. Further, since, $\sum_{u\in V}\deg_2(u) = 2P_2$, we have, by Lemma 2,

$$\sum_{u\in V}\deg_2^2(u) \le (\max_{w\in V}\deg_2(w))\sum_{u\in V}\deg_2(u) \le (4P_2)^{3/4}(2P_2) \le 8P_2^{7/4}.$$

By Lemma 3, $P_2 \ge m - \sqrt{mr} = \sqrt{m}(\sqrt{m} - \sqrt{r}) = \sqrt{m}\frac{m-r}{\sqrt{m}+\sqrt{r}} \ge \frac{m-r}{2}$, since, $r \le m$. Using this, it follows that $\frac{m}{\epsilon^2P_2^2}\sum_u\deg_2^2(u) \le \frac{8m}{\epsilon^2P_2^{1/4}} \le \frac{16m}{(m-r)^{1/4}}$. To boost the confidence to $1-\delta$, we keep $O(\log\frac{1}{\delta})$ independent copies and return the median from the concise copies. The space required is therefore $O(\frac{m}{\epsilon^2(m-r)^{1/4}}(\log\frac{1}{\delta})(\log n))$ bits. □

3 Lower Bounds

In this section, we present space lower bounds.

Lemma 8. *An algorithm that estimates P_2 for undirected and connected streaming graphs in the insert-only model to within a factor of $1 \pm \frac{1}{8}$ with probability $\frac{2}{3}$ requires $\Omega(n + \sqrt{m})$ bits.*

Proof. We reduce a special case of the two-party set disjointness problem in which parties A and B are each given a subset of $\{0, 1, \ldots, n-1\}$ of size at least $\frac{n}{3}$ with the promise that the subsets are either disjoint or have exactly one element in common. The parties have to determine whether the sets are disjoint. This problem has communication complexity $\Omega(n)$ bits. Suppose there is an algorithm $\mathcal{A}$ satisfying the premises of the lemma. A and B each construct in their local memory a complete graph whose nodes correspond to the items in the subset given to it. A inserts the edges corresponding to its complete graph into the data structure for $\mathcal{A}$ and sends it to B. B inserts the edges of its complete graph into the data structure of $\mathcal{A}$ and estimates P_2. If the sets are disjoint, then, $P_2 \leq \frac{5n^2}{16}$, and otherwise, $P_2 \geq \frac{7n^2}{16}$, allowing $\mathcal{A}$ to distinguish between the two cases. Hence, $\mathcal{A}$ requires $\Omega(n)$ bits. In the constructed graph, $m = \Theta(n^2)$, hence, the space complexity is $\Omega(\sqrt{m})$.

In the above construction, the graph is either connected (when the subsets intersect) or has two components (disjoint case). An additional tree-structure ensures that the graph is always connected. For $i \in \{0, 1, \ldots, n-1\}$, B inserts new vertices v_{2i} and v_{3i}, with edges between v_i and v_{2i} and between v_{2i} and v_{3i}. The nodes $\{v_{3i} : 0 \leq i \leq n-1\}$ are then made the leaf nodes of a complete binary tree (as much as possible) by adding new vertices. The resulting graph is connected. The contribution to P_2 by the new vertices is as follows. $\deg_2(v_{2i}) = 1 + \deg(v_i)$, for $0 \leq i \leq n-1$, and the contribution to P_2 by the remaining tree vertices is at most $n-1$ (vertex pairs at the same level) $+ n - 3$ (vertex pairs where one vertex is a grandparent of the other) $= 2n - 4$. Thus, total P_2 of the new graph is $n + 2n - 4 + 2$ oldP_2, where, oldP_2 is the P_2 of the graph prior to the addition of the tree structure. The rest of the argument proceeds as before. □

Lemma 9. *Deterministically estimating P_2 over streaming graphs to within factor of $(1 \pm \frac{1}{4})$ requires $\Omega(m)$ space.*

Proof. For any set S, we define the undirected graph G_S on the set S as $G_S = (V, E)$, where $V = S \cup a$, $E = \{(a, i), \forall i \in S\}$ and a does not belong to the domain of elements from where S is chosen. Therefore number of edges in the graph G_S is same as the number of vertices in the set S. Consider $|D| = 4m$ and select sets from them of size m such that any two of them have atmost $m/2$, elements as common. Number of such sets is $2^{\Omega(m)}$(can be established from known results of coding theory). For each such set, create the corresponding graphs. This gives a family of graphs $\mathcal{G}$ of size $2^{\Omega(m)}$, where each graph in the family has m edges. Select randomly $G_1, G_2 \in \mathcal{G}$ and create the graph streams $A(G_1, G_1)$ and $A(G_2, G_1)$ respectively, where $A(S, T)$ indicates in the stream A the edges of S arrive before the edges of T. We see $P_2(A(G_1, G_1)) = \binom{m}{2}$ and $P_2(A(G_2, G_1)) = \binom{(3m/2)}{2}$. So any deterministic algorithm that estimates P_2, within accuracy $1 \pm \frac{1}{4}$, must be able to distinguish between these two cases. By

pigeonhole principle if the space usage of any such deterministic algorithm is $< log|\mathcal{G}|$, then there exists at least one pair of graphs G_1 and G_2 in $\mathcal{G}$, such that the contents of the memory after reading G_1 and G_2 are same. Therefore the algorithm will give the same result for both $A(G_1, G_1)$ and $A(G_2, G_1)$, by making an error in at least one of them. So the space requirement of any deterministic algorithm that estimates P_2, with accuracy $1 \pm \frac{1}{4}$, must use $\log |\mathcal{G}| = \Omega(m)$ space. □

Estimating P_k over directed streaming graphs. We show that for *directed* streaming graphs, estimating P_k for $k \geq 2$, to any multiplicative factor requires $\Omega(m)$ space. The reductions use the standard bit vector index problem: Party A is given a bit-vector v of size r and party B is given an index i, $1 \leq i \leq r$. B has to determine whether $v[i] = 1$. The communication allowed is one-way from A to B. This problem has communication complexity of $\Omega(r)$ [7, 3].

Lemma 10. *Estimating P_k for directed streaming graphs to within any multiplicative accuracy factor requires $\Omega(m)$ bits.*

Proof. We will reduce a special case of the bit-vector index problem, where, it is given that exactly $\frac{r}{2}$ bits of v have value 1. The communication complexity of this problem is also $\Omega(r)$. Let $r = 2n$ and let $\mathcal{A}$ be an algorithm for estimating P_2. For every $v[i] = 1$ in the bit-vector v, party A inserts a directed edge $(r+1, i)$ to the summary structure of algorithm $\mathcal{A}$. A then sends the summary structure to B. Given index j, B adds the set of directed edges $\{(j, k) \mid r+2 \leq k \leq r+n+2\}$, to the summary structure that it received from A. If $v[j] = 1$, then $P_2 = n$, else $P_2 = 0$, proving the claim for P_2. The extension for P_k is analogous. □

Lemma 11. *For $k \geq 3$, estimating P_k to within factor of $1 \pm \frac{1}{3}$ with probability $\frac{3}{4}$ over undirected streaming graphs with n vertices requires $\Omega(n^2)$ bits.*

Proof. We reduce the bit-vector index problem to the problem of estimating P_3. Let $r = \frac{n(n-1)}{2}$ and let $v[1 \ldots r]$ be the given vector of 0's and 1's. Let $\mathcal{B}$ be an algorithm for estimating P_2 with the specified accuracy and confidence. Each index $1 \leq r \leq \frac{n(n-1)}{2}$ is written uniquely as a pair of distinct numbers, (u, w), each lying between 0 and $n-1$. This mapping is used to create a graph $G = (V, E)$, where, $V = \{1, 2, \ldots, 9n\}$. For every index $j = (u, w)$ such that $v[j] = 1$, we add an edge $(u, w) \in E$. Next, for the given index $i = (c, d)$, we add $8n$ new vertices to the graph, and attach $4n$ of them to c and $4n$ of them to d. These edges are given as input stream to $\mathcal{B}$. We now use $\mathcal{B}$ to estimate P_3. $v[b] = 1$ iff there is an edge between c and d in G. In this case, $P_3 \geq 16n^2$, and, otherwise, $P_3 \leq 8n^2 + \binom{n}{2}$. Therefore the space requirement by P_3 is $\Omega(r) = \Omega(n^2)$. The proof can be easily extended to P_k, $k > 3$. □

Theorem 12. *A deterministic algorithm for testing connectivity of an undirected graph in the updatable streaming graph model requires $\Omega(n^2)$ space.*

Proof. Let $G = (V, E)$ be a connected graph and let $G' = (V, E')$ be the edge-complement graph on the same set of vertices. Consider the family of graphs

for which G and G' are both connected. For this family of graphs, checking for edge-membership can proceed as below. (u, v) is an edge in G iff there is a sequence of edges $e_1, \ldots, e_{k-1}, e_k = (u, v)$ in G, such that after the deletion of $e_1, e_2, \ldots, e_{k-1}$ in sequence, the graph remains connected, but gets disconnected after $e_k = (u, v)$ is deleted thereafter. The sequence of edges $e_1, \ldots, e_{k-1}$ can be thought of as a certificate of membership of (u, v) in G. Analogously, if (u, v) is not in G, then, it is in G', and therefore, there exists a certificate for membership of (u, v) in G'. This certificate serves as a certificate that (u, v) is not in G. Hence checking edge membership reduces to connectivity testing problem.

Given an algorithm that maintains a summary structure for testing connectivity of a streaming graph, we use it to maintain a pair of summaries corresponding to G and its complement G'. This is easily done by letting $E = \phi$ and $E' = K_n$, where K_n is the clique of n vertices. Corresponding to each edge update, we propagate the update to the summary structure for G and propagate the complement of the update to the summary structure for G'.

We now obtain a lower bound on the number of graph-complement pairs (G, G') over n vertices such that both G and G' are connected. Consider the complete graph K_n on n vertices, for $n > 2$. Choose a spanning tree C of K_n that is a chain. Consider the remaining graph defined by the set of edges in $K_n - C$. This graph remains connected for $n \geq 4$. Let D be a spanning tree of the graph defined by edges in $K_n - C$. Place the set of edges in C in G and the set of edges in D in G'. The number of remaining edges is $\binom{n}{2} - 2(n-1)$. Each of these edges can be placed either in G or in G' in $2^{\binom{n}{2}-2(n-1)}$ ways. Each of these ways gives a different (G, G') pair. By construction, G and G' contain C and D respectively, and are therefore connected. Therefore, the number of graph-complement pairs (G, G') over n vertices such that both G and G' are connected is at least $2^{\binom{n}{2}-2(n-1)}$.

The algorithm that tests for edge-membership must have a different memory pattern for each of the graph-complement pairs (G, G'). Otherwise, given two distinct pairs (G, G') and (H, H'), there are edge pairs (e, e') that distinguish them. Mapping them to the same pattern causes the algorithm to make at least one error when presented with the certificates of the edges e and e', respectively. Hence checking edge-membership requires space $\Omega(\log(2^{\binom{n}{2}-2(n-1)})) = \Omega(n^2)$ bits . Since edge-membership can be reduced to connectivity testing, the statement of the lemma follows. □

Corollary 13. *Deterministic algorithms for the following problems require $\Omega(n^2)$ space in the updatable graph streaming model: (a) estimating the size of the transitive closure of an undirected graph to within a factor of $1 \pm \frac{1}{5}$, and (b) estimating the diameter of an undirected graph to within any approximation factor.*

Proof. Let G be a graph consisting of n vertices, $N = \{1, 2, .., n\}$. Two chain graphs, $P_a = a_1 - a_2 - ... - a_{n/2}$ and $P_b = b_1 - b_2 - ... - b_{n/2}$ are created, where $a_1, b_1 \in N, a_1 \neq b_1$, and the other vertices do not belong to N. If G is connected, then transitive closure of G is $\binom{2n-2}{2} \approx 2n^2$. Otherwise, there exists a_1 and b_1, such that they belong to different components of G. In that case

transitive closure size is $< \binom{3n/2}{2} + \binom{n/2}{2} \approx \frac{5n^2}{4}$. If $\epsilon \leq \frac{1}{5}$, then for all a_1 and b_1, if G is connected, transitive closure size is $\geq 2n^2(1-\frac{1}{5}) = \frac{8n^2}{5}$. Else there exists a_1 and b_1, for which transitive closure size $< \frac{5n^2}{4}(1+\frac{1}{5}) = \frac{3n^2}{2}$. An algorithm that measures transitive closure with accuracy $1 \pm \frac{1}{5}$, can distinguish between these two cases. So we have a reduction from transitive closure to connectivity testing. Since connectivity testing requires $\Omega(n^2)$ bits of space, any deterministic algorithm estimating transitive closure with accuracy $1 \pm \frac{1}{5}$ must use space $\Omega(n^2)$ bits.

(b) If the graph is connected, then its diameter is at most $n-1$, otherwise it is ∞. This reduces the connectivity testing problem to diameter estimation. □

Note that testing connectivity and maintaining the size of transitive closure is easily solved using $O(n \log n)$ space in the insert-only streaming model [7, 3].

References

1. C. Demetrescu, I. Finocchi, and A. Ribichini. "Trading off space for passes in graph streaming problems". In *Proceedings of ACM SODA*, 2006.
2. J. Feigenbaum, S. Kannan, A. McGregor, S. Suri, and J. Zhang. On graph problems in a semi-streaming model. In *Proceedings of ICALP*, pages 531–543, 2004.
3. J. Feigenbaum, S. Kannan, A. McGregor, S. Suri, and J. Zhang. Graph distances in the streaming model: the value of space. In *Proceedings of ACM SODA*, 2005.
4. Philippe Flajolet and G.N. Martin. "Probabilistic Counting Algorithms for Database Applications". *J. Comp. Sys. and Sc.*, 31(2):182–209, 1985.
5. S. Ganguly, M.N. Garofalakis, A. Kumar, and R. Rastogi. "Join-distinct aggregate estimation over update streams". In *Proceedings of ACM PODS*, 2005.
6. P. B. Gibbons and S. Tirthapura. "Estimating simple functions on the union of data streams". In *Proceedings of ACM SPAA*, 2001.
7. M. Henzinger, P. Raghavan, and S. Rajagopalan. Computing on data streams. Technical Note 1998-011, Digital Systems Research, Palo Alto, CA, May 1998.
8. S. Muthukrishnan. *"Data Streams: Algorithms and Applications"*. Foundations and Trends in Theoretical Computer Science, *Vol. 1, Issue 2*, 2005.
9. Barna Saha. "Space Complexity of Estimating Aggregate Path Metrics over Massive Graph Streams and Related Metrics". Master's thesis, IIT Kanpur, Computer Science, 2006.