Triangle Count Estimation in Graph Streams: Space Bounds

Summary and Intuition

Overview

Estimating the number of triangles T in an m-edge graph data stream to within a $(1 \pm \varepsilon)$-factor requires space scaling as

$$\Theta\Big(\min\{\, m^{3/2}/T,\ m/\sqrt{T}\}\Big).$$

Two distinct "hard" regimes yield these bounds: when triangles are very sparse and when they are moderately abundant.

1. Sparse-Triangle Regime (T small)

When $T \ll m^{3/2}$, the dominant term is $m^{3/2}/T$. In this case, most "wedges" (two-hop paths) are not closed into triangles, so any algorithm that samples fewer than $\Theta(m^{3/2}/T)$ wedges will see almost none of the T true triangles. Formally, distinguishing a triangle-free graph from one with T well-spread triangles requires storing $\Omega(m^{3/2}/T)$ bits [1, 2].

Why does $m^{3/2}/T$ matter?

Let's break it down:

1. There can be up to **around $m^{3/2}$ wedges** in a graph.
   This is just a known mathematical fact (you don't need to prove it now).
2. If **only a few of these wedges** are part of triangles (because T is small), then:
   - If you randomly sample wedges, you'll probably get ones that **don't form triangles**.
   - So, to have a **good chance** of seeing even a few triangles, you must sample **a lot of wedges**.
3. How many wedges must you sample?
   To see at least one triangle, you must sample around:

   $$\frac{m^{3/2}}{T}$$

   wedges.
4. And storing each wedge takes memory.
   So, you need **at least $\Omega(m^{3/2}/T)$ space**.

What's a Wedge?

A wedge is a 2-hop path: **A–B–C**. If A–C exists, it becomes a triangle.

Why Wedges?

Triangles form by **closing wedges**, so estimating triangles means counting wedges.

Key Fact

A node of degree d forms $\binom{d}{2} \approx d^2$ wedges.

Goal

Maximize wedges using a graph with m edges.

Worst Case Setup

- Build $\sqrt{m}$ nodes with degree $\sqrt{m}$.
- Each connects to $\sqrt{m}$ degree-1 nodes.
- Each high-degree node forms $\approx m$ wedges.
- $\sqrt{m} \cdot m = \boxed{m^{3/2}}$ wedges total.

Final Result

A graph with m edges can have up to

$$\boxed{\Theta(m^{3/2})\ \text{wedges}}$$

2. Heavy-Edge Regime (T large)

When T grows so that $m/\sqrt{T} < m^{3/2}/T$, the bound $m/\sqrt{T}$ takes over. Here one must detect "heavy" edges incident to many triangles. If an algorithm stores $o(m/\sqrt{T})$ edges, it will likely miss all such edges, making it impossible to approximate T. One can show a lower bound of $\Omega(m/\sqrt{T})$ by constructing graphs where a single edge participates in $\Theta(\sqrt{T})$ triangles [1].

- **For sparse triangles** ($T \ll m$): $\frac{m^{3/2}}{T}$ is the bottleneck.
- **For dense triangles** ($T \gg m$): $\frac{m}{\sqrt{T}}$ controls the space.
- **Practical implication**: Algorithms must adapt to which term is larger!

3. Matching Upper Bounds

McGregor *et al.* (PODS 2016) give two constant-pass algorithms achieving

- $O(m^{3/2}/T)$ space by sampling random wedges. [1]

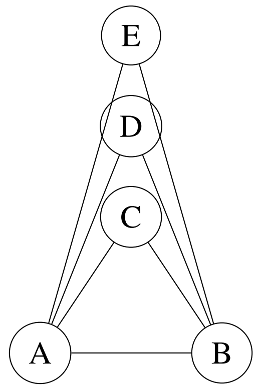

Figure 2: Edge A-B involved in multiple triangles with shared neighbors

- $O(m/\sqrt{T})$ space by sampling edges and tracking heavy hitters. [1]

Bera and Seshadhri (STACS 2017) prove corresponding lower bounds for any constant-pass, arbitrary-order algorithm [2].

4. Conclusions

The $\min(m^{3/2}/T,\ m/\sqrt{T})$ barrier is both necessary and sufficient for $(1\pm\epsilon)$-approximate streaming estimation of triangle counts with a constant number of passes over an arbitrary-order stream.

References

[1] A. McGregor, S. Vorotnikova, H. Vu. "Better Algorithms for Counting Triangles in Data Streams." PODS 2016.

[2] S. Bera, C. Seshadhri. "Towards Tighter Space Bounds for Counting Triangles..." STACS 2017.

[3] A. McGregor, M. Hu. "The Complexity of Counting Cycles in the Adjacency List Streaming Model." PODS 2019.