14 Graph Sketching: AGM Sketches for Connectivity

Graph Sketching: AGM sketch for Connectivity:

Dynamic Graph Streams: Existing edges can be deleted also.

Stream: Collection of tuples

$\langle u_i, v_i, \Delta \rangle$ for $u_i, v_i \in V$

$\Delta \in \{-1, +1\}$

$\Delta = +1 \longrightarrow$ Insert an edge between u_i & v_i.

$\Delta = -1 \longrightarrow$ delete theedge between u_i & v_i

The goal is to solve the problem on the <u>final</u> graph.

Warm-up: Can we solve any non-trivial problem?
Even find an Edge at the end of the streams?

ℓ_0-Samplers: Given a frequency vector

$$f = \begin{bmatrix} f_1 \\ \vdots \\ f_N \end{bmatrix}_N$$ with ± 1 updates

to the coordinates in the stream, Sample an element f_j from $\mathrm{supp}(f)$ uniformly at random

Step 1: Suppose $|\mathrm{supp}(f)| = 1$.

Solution: Compute $g \cdot f$ for $g = [1, 2, \ldots, N]$.

$a \cdot f$

return $\underline{g \cdot f}$ $\blacksquare$

$w = [1, \text{———}, 1]$ $\qquad w \cdot f$

Step 2. Test if $|Supp(f_s)| = 1$ or larger.

Solution: Sample vectors g_1, g_2, g_3 as follows:

$\forall i \in [n]$, pick $j \in \{1, 2, 3\}$ randomly and set $g_{j,i} = 1$, $g_{j',i}$ so

$$
\begin{array}{ll}
g_1 & [\, 1 \quad 0 \quad 0 \quad 0 \quad \cdots \,] \\
g_2 & [\, 0 \quad 1 \quad 1 \quad 0 \quad \cdots \,] \\
g_3 & [\, 0 \quad 0 \quad 0 \quad 1 \quad \cdots \,]
\end{array}
$$

Compute $g_1 \cdot f$, $g_2 \cdot f$, $g_3 \cdot f$

If answer was $(\neq 0, 0, 0)$
$(0, \neq 0, 0)$ output yes
$(0, 0, \neq 0)$ o.w No

Proof: when $|supp(f)| = 1$ this is clearly true.

when $|supp(f)| = 0$ we get $(0, 0, 0)$.

when $|supp(f)| \geq 2$.

Consider mapping of last element:

Case 1 Answer is valid so for say:

$(\neq 0, 0, 0)$; with prob $\frac{1}{3}$

be mapped to $\{2, 3\}$

making answer invalid.

Case 2 The answer is invalid:

$(\neq 0, \neq 0, 0)$ with prob $1/3$;

will be mapped to 3 keeping answer invalid.

Repeating the test $O(\lg n)$ times gives correct answer w.h.p.

Step 3: When $|supp(f)| = k$ for some $2^{\alpha} \leq k < 2^{\alpha+1}$

(we only know α)

Solution: First sample every element in N w.p. $\frac{1}{2^{\alpha+1}}$; then run previous two steps.

Formally: Let $g = [0, 1, \cdots, 0]$

$\uparrow$ is 1 w.p. $\frac{1}{}$

Let M be the matrix of $2^{\alpha+1}$ previous steps.

Compute $M \cdot g \cdot f$.

Proof. $\underline{\Pr(|\mathrm{Supp}(g \cdot f)| = 1) = \Omega(1).}$

$\Pr($exactly j belongs to $g \cdot f) =$

$$\frac{1}{2^{\alpha+1}} \cdot \left(1 - \frac{1}{2^{\alpha+1}}\right)^{k-1}$$

$\Pr(\exists j$ that only mapped to $g \cdot f) =$

$$\frac{K}{2^{\alpha+1}} \cdot \left(1-\frac{1}{2^{\alpha+1}}\right)^{K-1} \geq \frac{1}{2} \cdot \left(1-\frac{1}{2^{\alpha+1}}\right)^{2^{\alpha+1}-1}$$

$$\geq \frac{1}{4}. \qquad \blacksquare$$

Step 4: Original problem

Solution: Run the prev. alg for $\alpha=1, \alpha=2, \ldots, \alpha=\lg N$. in parallel.

Note: All random vectors can be pairwise independent instead of fully random

Summary: A polylg(N)-space alg for ℓ_0-sampling.

AGM sketches:

Solve the following problem:

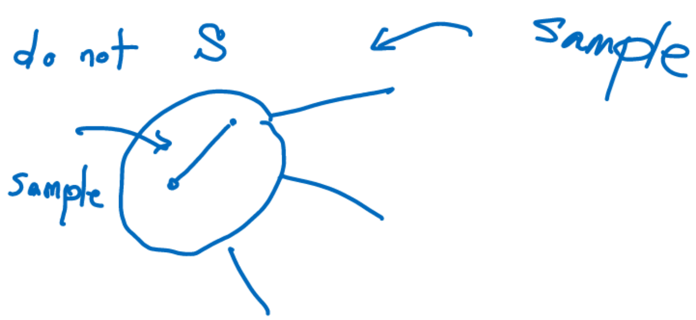
- Store polylg(N) bits per vertex during dynamic stream
- Given a set S of vertices at the end output an edge from the cut $\delta(S)$.

- The answer is only a function of sketches of S.

Toy examples:

1. only edge insertions.
2. S is a singleton vertex

Main case: S is arbitrary.



We want edges inside

we want edges inside

S to get canceled.

– These edges have Both endpoints in S.

Consider the following matrix $n \times \binom{n}{2}$.

$$B = \begin{array}{c|ccc|}
 & e_1 & (u-v) & e_{\binom{n}{2}} \\
\hline
v_1 & & 0 & \\
u & & +1 & \\
 & & 0 & \\
v & & -1 & \\
\end{array}$$

$v_n \quad 0 \quad]$

each column u,v has a

$+1$ for u & -1 for

v $(u<v)$.

b_v = vector of vertex v

For a set S of vertices

how does vector

$b_S = \sum_{v \in S} b_v$ look like?

$$\begin{array}{cc} & wz \quad uv \\ b_S : & \begin{bmatrix} 0 & \begin{matrix} -1 \\ +1 \end{matrix} & \end{bmatrix} \end{array}$$

$uv \in \delta(S)$

$wz \notin \delta(S)$

Edges inside S cancel out.

AGM sketch: each vertex

maintains $M \cdot b_v$:

ℓ_0-sampler matrix

(over $N = \binom{n}{2}$)

At the end, we use

$M \cdot b_S = M \cdot (\sum_{v \in S} b_v) = \sum_{v \in S} M \cdot b_v$. $\blacksquare$

Algorithm for Connectivity.

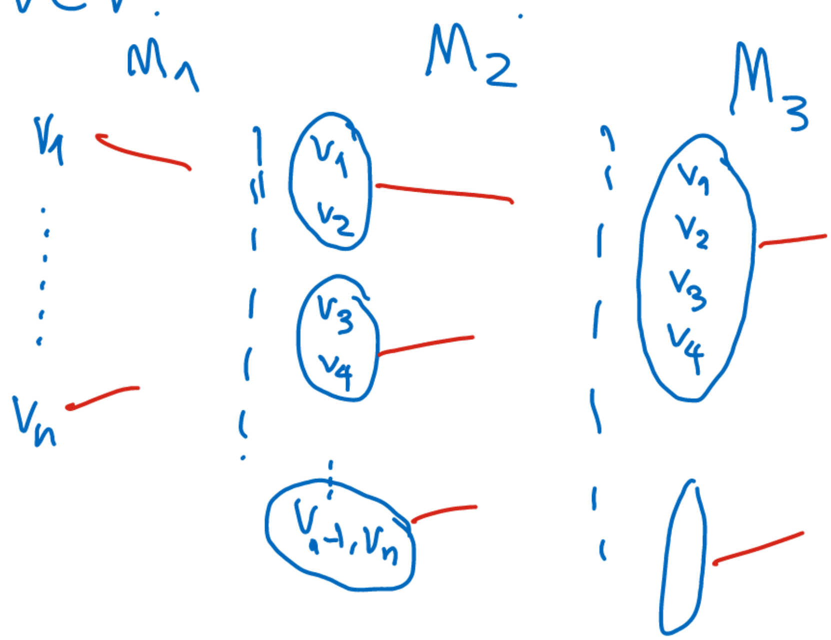
Sample AGM sketches

$M_1, \ldots, M_{t=\lg n}$ and compute

$M_i \cdot b_v$ for all $v \in V$ $i \in [t]$.

Then run the following:

Use M_1 and each vertex v individually to get an edge out of each $v \in V$.



At $M_{t = \lg n}$, we will find

all Connected Components of

G & a spanning forest for G.

total space: $O(n \cdot \text{poly} N) =$

$\tilde{O}(n)$.

Practice Problem: An $\tilde{O}(n \cdot k)$ space alg for finding a k-edge connected component.