Separations and Equivalences Between Turnstile Streaming and Linear Sketching

John Kallaugher Eric Price

The University of Texas at Austin

THE UNIVERSITY OF TEXAS AT AUSTIN

Relations Between Streaming Models

Insertion-only Algorithms $\supsetneq$ Turnstile Algorithms $\supsetneq$? Linear Sketches

Q: Do turnstile algorithms beat linear sketches?

Turnstile Algorithms and Linear Sketches

Li, Nguyễn, Woodruff '14: Turnstile algorithms **do not** beat linear sketches... kind of.

- If they work on streams of very long length.
- And support unrestricted intermediate states.

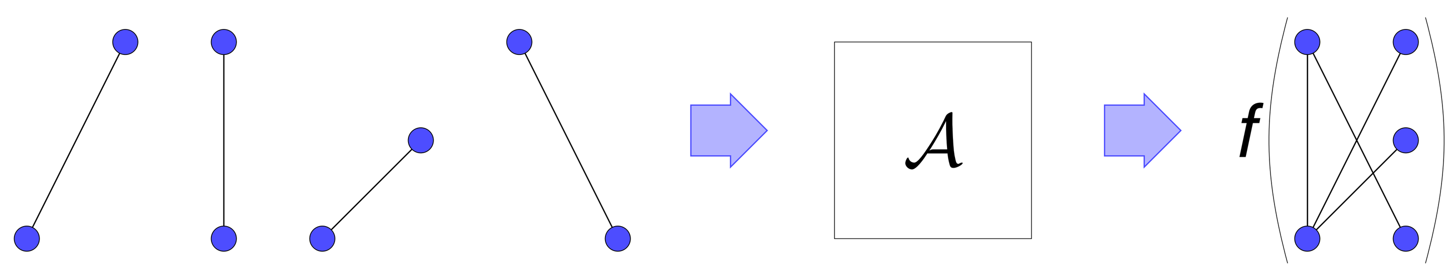
Furthermore, the reduction is not *constructive*—the sketch exists, but there is no guarantee it can be efficiently computed.

Our Results:

- If either [LNW '14] requirement is removed, turnstile algorithms **can** beat linear sketching.
- Otherwise, we strengthen the LNW equivalence to be constructive.

Background: Streaming Algorithms

Algorithms for very large datasets that arrive "one piece at a time".



Models of streaming computation:

- Insertion only: Only *positive* updates.

$$\binom{+1}{0}\binom{0}{+1}\binom{+1}{0}\binom{+1}{0} \Rightarrow \binom{3}{1}$$

- Turnstile: Both insertions and deletions.

$$\binom{+1}{0}\binom{0}{+1}\binom{-1}{0}\binom{+1}{0} \Rightarrow \binom{1}{1}$$

- Linear sketching: We may store only a *linear* function of the input stream.

$$\binom{+1}{0}\binom{0}{+1}\binom{+1}{0}\binom{-1}{0} \Rightarrow \boxed{Mv} \Rightarrow g(Mv)$$

Separations

Theorem (*Separation*)

There is a problem that requires $\Omega(n^{1/3})$ space for linear sketches, but that is solvable in $O(\log n)$ in turnstile streaming, as long as the stream either:

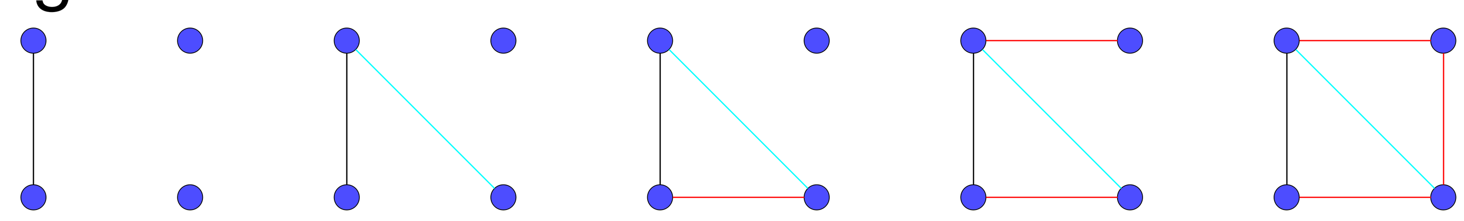
- *has length $O(n)$*
- *is binary at all intermediate states.*

Based on *bounded-degree triangle counting*: distinguish between a bounded-degree graph with n triangles and with 0.

- $O(\log n)$ in insertion-only. [Jowhari, Ghodsi '05]
- $\Omega(n^{1/3})$ in linear sketching. [Kallaugher, Kapralov, Price '18]

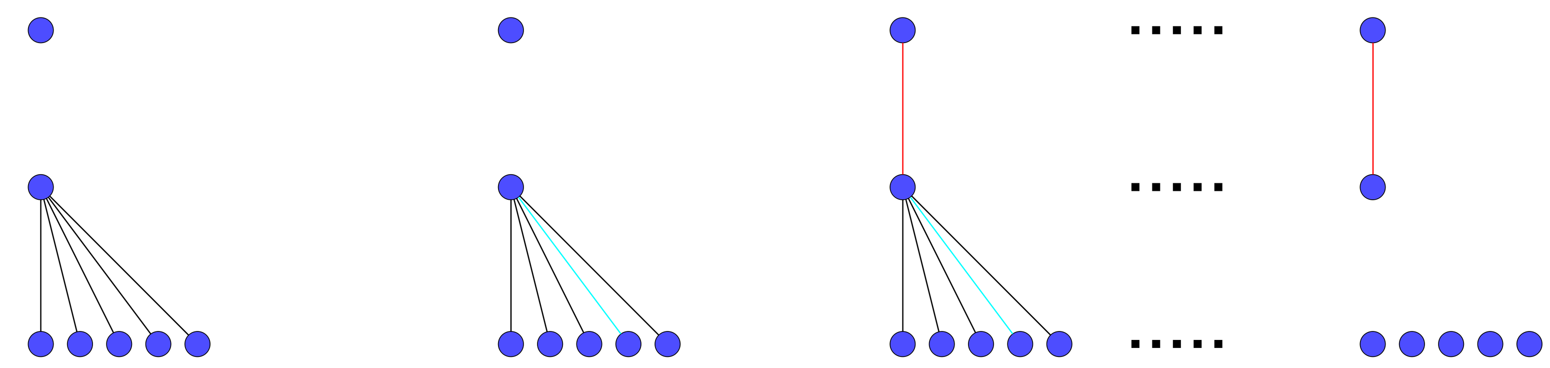
Goal: turnstile algorithm matching insertion-only.

[JG '05]: sample "seed" edges and keep their neighbors.



Sample seed Keep neighbors

Algorithm fails in turnstile because of high-degree vertices from intermediate steps.



Algorithm can be made to work if:

- Intermediate states are bounded-degree.
- Or stream is short.

Equivalences

[LNW '14] result proves a sketch *exists*. Can we get an explicit computable reduction?

Theorem (*Constructive Equivalence*)

Suppose there is a deterministic algorithm solving a streaming problem P that works on streams of all lengths and uses S space. Then there is a linear sketching algorithm for P that uses $O(S \log n)$ space.

With a random oracle, this can be extended to randomized algorithms.