New Parallel and Streaming Algorithms for Directed Densest Subgraph

Slobodan Mitrović, **Theodore Pan**, Mahdi Qaempanah, Mohammad Amin Raeisi

UC Davis, **UC Davis**, Sharif University of Technology, Yale University

December 2025

NEURAL INFORMATION PROCESSING SYSTEMS

Densest subgraph problem

Given an **undirected** graph G, we define the density given a vertex set S to be

$$\rho(S) = \frac{|E_G(S)|}{|S|}.$$

Densest subgraph problem

Given an **undirected** graph G, we define the density given a vertex set S to be

$$\rho(S) = \frac{|E_G(S)|}{|S|}.$$

For a **directed** graph G, we define the density given two vertex sets S, T to be

$$\rho(S,T) = \frac{|E_G(S,T)|}{\sqrt{|S||T|}}.$$

Densest subgraph problem

Given an **undirected** graph G, we define the density given a vertex set S to be

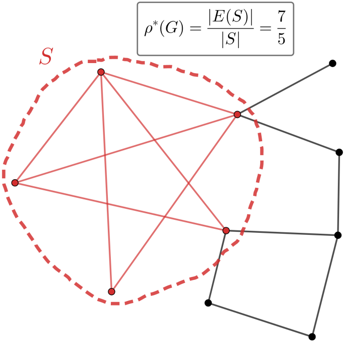
$$\rho(S) = \frac{|E_G(S)|}{|S|}.$$

For a **directed** graph G, we define the density given two vertex sets S, T to be

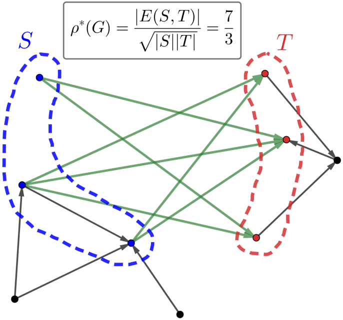
$$\rho(S, T) = \frac{|E_G(S, T)|}{\sqrt{|S||T|}}.$$

The **directed densest subgraph** are sets S^*, T^* that attain the maximum density $\rho^*(G)$. We look to find an approximate densest subgraph.

Densest subgraph problem



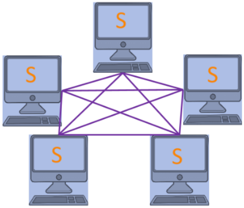
(a) Undirected graph



(b) Directed graph

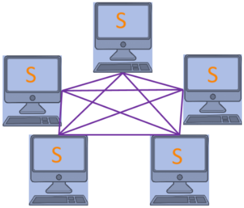
Model definitions

Massively Parallel Computation (MPC) model: Machines perform synchronous rounds of local computation and communication across N machines. Each machine has space S and messages sent and received are limited by S.



Model definitions

Massively Parallel Computation (MPC) model: Machines perform synchronous rounds of local computation and communication across N machines. Each machine has space S and messages sent and received are limited by S.



Our work is in the *sublinear regime* where $S = O(n^{\delta})$ for $\delta \in (0, 1)$.

Model definitions

Semi-streaming Model: Graph is given as a stream of edges and algorithms are constrained to $O(n \cdot \text{poly} \log n)$ internal memory.

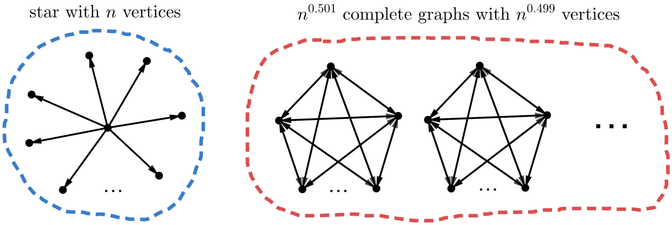
Model definitions

Semi-streaming Model: Graph is given as a stream of edges and algorithms are constrained to $O(n \cdot \text{poly} \log n)$ internal memory.

Note that $\Omega(n)$ memory is needed to even output the vertex sets of the directed densest subgraph. Our work focuses on *single-pass* semi-streaming algorithms.

Why is the directed problem difficult?

We have issues with uniform sampling. For **undirected**, it's been shown we need to only sample $O(n \log n)$ edges. However, for **directed**, we need $\Omega(n^{1.5})$ edges.



Why is the directed problem difficult?

Sublinear MPC: This issue with sampling makes it difficult to reduce neighborhoods to fit in sub-linear memory. Current state of the art for densest subgraph in sub-linear MPC:

- Undirected: $\tilde{O}(\sqrt{\log n})$ rounds [Ghaffari et al. (ICML 2019)]
- Directed: $O(\log n)$ rounds [Bahmani et al. (WAW 2014)]

Why is the directed problem difficult?

Semi-streaming: Uniform sampling easily allows a single-pass semi-streaming algorithm for undirected densest subgraph. The same sampling gives a bad approximation for directed densest subgraph though.

- Undirected: $(1+\epsilon)$-approximation [Esfandiari et al. (SPAA 2016)]
- Directed: $\tilde{\Omega}(n^{1/6})$-approximation [Esfandiari et al. (SPAA 2016)]

Our contributions

Sublinear MPC:

- We reduce directed densest subgraph from $O(\log n)$ to $\tilde{O}(\sqrt{\log n})$ rounds while attaining a $(2+\epsilon)$-approximation.
- This matches the round complexity of [Ghaffari et al. (ICML 2019)].

Our contributions

Sublinear MPC:

- We reduce directed densest subgraph from $O(\log n)$ to $\tilde{O}(\sqrt{\log n})$ rounds while attaining a $(2+\epsilon)$-approximation.
- This matches the round complexity of [Ghaffari et al. (ICML 2019)].

Semi-streaming:

- First single-pass *deterministic* algorithm for undirected or directed DS.
- Attains a $O(\log n)$-approximation.
- Also an insertion-only dynamic algorithm with $O(\text{poly} \log n)$ worst-case update time and sublinear memory.

Future work

1. In sublinear MPC, our result attains a $(2+\epsilon)$-approximation while there's a $(1+\epsilon)$-approximation for undirected. Can we match the approximation?
2. Is it possible to push past the $\tilde{O}(\sqrt{\log n})$ round complexity barrier for constant approximations in either case of undirected or directed?
3. In semi-streaming, can we attain a constant approximation in a single pass?