Low-Congestion Shortcuts

Part-wise Aggregation

Collection of vertex disjoint $S_1,...,S_k$
each $G[S_i]$ is connected.
Every vertex $v_i \in S_j$ holds $O(\log n)$-bit $x_i$
and it is required for every $v_i \in S_i$ to learn
the aggregate function of all $x_k$'s for $v_k \in S_j$.

Low-Congestion Shortcuts (LCS)

$G = (V,E)$ and $S_1,...,S_k$, each $G[S_i]$ is connected.
A collection of subgraphs $\mathcal{H} = \{H_1,...,H_k\}$ is an $(\alpha,\beta)$ LCS if:

1. $\text{Diam}\ (G[S_i] \cup H_i) \leq \alpha$.
2. Each edge $e \in G$ appears on at most $\beta$ $H_i$-subgraphs.

Theorem: Given a $T$-round alg for computing $(\alpha,\beta)$ shortcuts for $S_1,...,S_k$,
then one can solve the partwise aggregation problem in $\tilde{O}(T+\alpha+\beta)$.

Remark: This also implies MST in $\tilde{O}(T+\alpha+\beta)$ rounds

<table>
<thead>
<tr>
<th>Graph Family</th>
<th>Quality of LCS</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Graphs</td>
<td>$\alpha+\beta = O(D+\sqrt{n})$</td>
</tr>
<tr>
<td>Planar Graphs</td>
<td>$\alpha+\beta = \tilde{O}(D)$</td>
</tr>
<tr>
<td>Expander Graphs</td>
<td>$2^{\frac{(\log n)}{\log 2}}$</td>
</tr>
<tr>
<td>Graph with $D=O(1)$</td>
<td>$\alpha+\beta = \tilde{O}(n^{\frac{D-2}{20-2}})$</td>
</tr>
</tbody>
</table>
Approximate min-cut

**Cut** is a graph partitioning into $(S, V\setminus S)$

The value of the cut $(S, V\setminus S)$ is

$$\delta_G(S) = |\{(u,v) | u \in S, v \in V\setminus S\}|$$

**Min-cut** is $\min_{S \subseteq V} \delta_G(S)$

**O(\log n) Approximation**

**Karger's edge sampling**: Given graph $G = (V, E)$ with min cut $\lambda$.

Let $G[p]$ be the subgraph obtained by sampling each edge $e \in E$ indep. w.p.

$$p = \frac{\log n}{\varepsilon^2 \lambda}.$$  

Then: w.h.p. for $S \subseteq V$:

$$\delta_{G[p]}(S) \in (1 \pm \varepsilon) \delta_G(S) \cdot p$$

**Cor:** $G[p]$ for $p = \Theta(\frac{\log n}{\lambda})$ then w.h.p. $G[p]$ is connected.

**Algorithm**

For $i = 1$ to $\lceil \log n \rceil$

For $j = 1$ to $\ell = O(\log n)$

$$G_{ji} = G[p_j] \text{ for } p = 2^{-j}$$

$$X_{ji} = 1 \text{ iff } G_{ji} \text{ is connected.}$$

$$j^* = \max j \text{ s.t. } \sum_{i=1}^{l} X_{ji} \geq 0.9 \cdot \ell$$

Return $2^{j^*}$
**FACs 2020:** Exact min-cut in $\tilde{O}(D+\sqrt{n})$. $(1+\epsilon)$ approximation with shortcuts.

**LB:** for unweighted min-cut: $\Omega(n \cdot D) \nRightarrow \text{Poly}(D)$

**K-sparse certificate**

$H \subseteq G$ is a $K$-sparse certificate if:

1) $\forall S \subseteq V$, $\frac{\delta_H(S)}{\delta_H(G)} \geq \min\{K, \delta_G(S)\}$

2) $|E(H)| \leq K \cdot n$.

**Centralized Construction**

$H \leftarrow \emptyset$, $G' \leftarrow G$

For $i=1$ to $k$:

$F_i = \text{MSF in } G'$

$H \leftarrow H \cup F_i$

$G' \leftarrow G' \setminus F_i$

**Correctness:**

Focus on a specific cut $(S, V \setminus S)$

In every step $i$: collect at least one cut edge (if exists).

**Distributed Implementation**

**Phase** $i$: edges in $F_i \cup \ldots \cup F_{i-1}$ have weight $\infty$

edges in $G \setminus \bigcup_{j=1}^{i-1} F_j$ have weight 1

Compute MST.

**Lemma:** Can compute $K$-certificate in $\tilde{O}(K(D+\sqrt{n}))$ rounds.

**Next steps:**

- assume $\lambda$ is known.
- provide alg' that works in $\tilde{O}(\lambda(D+\sqrt{n}))$ rounds.
High Level Idea of Centralized Construction (Matula's)

- Compute $K'$-certificate $H \subset G$
  
  $K' = (1 + \frac{\epsilon}{10}) \lambda$

- Define $G'$ obtained by contracting all edges in $G \setminus H$.

  **Case 1:** $|V(G')| \geq \frac{n}{1 + \frac{\epsilon}{5}}$

  Avg deg in $G' \leq \frac{2|E(G')|}{|V(G')|} \leq \frac{2(1 + \frac{\epsilon}{40})\lambda n}{\frac{n}{1 + \frac{\epsilon}{5}}} \leq (2 + \epsilon) \lambda$

  $\implies$ node (possibly super-node) with deg $\leq (2 + \epsilon) \lambda$ =) defines a cut.

  **Case 2:** $|V(G)| \leq \frac{n}{1 + \frac{\epsilon}{5}}$

  $\implies$ continue recursively.

Distributed Implementation

$G \setminus H$ to be contracted:

- Every super-node in $G'$ corresponds to a connected comp. of $H$.

Compute certificate for $G'$.

- assign weight of 0 to all edges inside super-node.

Complexity: $\tilde{O}(\lambda(D + \sqrt{n}))$ rounds.

Goal: $(2 + \epsilon)$-approx in $\tilde{O}(D + \sqrt{n})$ rounds.

Idea: Use Karger to sparsify the graph. $G' = G[\rho]$ for $\rho = \frac{200 \log n}{\epsilon^2 \lambda}$:

$\forall S \subseteq V, \exists_{G'}(s) \in (1 + \epsilon) \Delta_{G}(s) \cdot \rho$

$\implies$ Apply Matula's alg' on $G'$ in $\tilde{O}(D + \sqrt{n})$ rounds.

Omit assumption on $\lambda$

Try $\lambda = (1 + \frac{\epsilon}{40})^i$ as a guess for $i = 1, 2, 3, ...$