CS 7880 Special Topics in TCS: Sublinear Algorithms (Fall'22)   Northeastern University

Lecture 19

November 18, 2022

Instructor: Soheil Behnezhad   *Scribe: William Schultz*

Disclaimer: *These notes have not been edited by the instructor.*

1 MPC Algorithms with Strongly Sublinear Space

Given a graph G, we focus on algorithms in the MPC model using $O(n^{\epsilon})$ space per machine for $0 < \epsilon < 1$. We refer to this as the *strongly sublinear* regime.

Theorem 1 ([GU19])**.** *For any $0 < \delta < 1$ there is an $O(\sqrt{\log n} \cdot \log\log n)$ algorithm for finding a $(1+\epsilon)$ approximate (weighted) matching or maximal independent set and a 2-approximate minimum vertex cover or maximal matching, using $O(n^{\delta})$ space.*

> **Remark.** It is open whether the $O(\sqrt{\log n} \cdot \log\log n)$ bound can be improved on. There are reasons, however, to believe that the correct round complexity is $O(\log\log n)$.

1 vs. 2 cycle problem: Suppose that a given graph input is either 1 cycle on n nodes or 2 cycles on $n/2$ nodes each. How many rounds do we need to distinguish the two possible inputs with strongly sublinear space per machine?

Claim 2 (1 vs. 2 cycle conjecture)**.** *Solving the 1 vs 2 cycle problem requires $\Omega(\log n)$ rounds. (**Open**)*

Theorem 3. *For any fixed $0 < \delta < 1$, there is an $O(\log n)$ round MPC algorithm that returns the number of connected components in a given graph, using $O(n^{\delta})$ space per machine.*

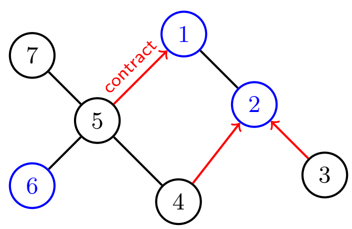
> **Remark.** Note that the best known lower bound for solving the 1 vs. 2 cycle problem with $O(n^{\delta})$ space is $\Omega(1/\delta)$. Furthermore, proving a better than $\Omega(\frac{1}{\delta})$ lower bound for any problem in P would imply $NC^1 \subsetneq P$ (i.e. that there exist problems in P not in NC^1[a]) The problem of whether $NC^1 \subsetneq P$ has been open for several decades, so seems an indication of the difficulty of proving these kinds of lower bounds [BIS90].
>
> [a] Recall that NC^1 is the class of problems that can be solved with poly-size circuits of logarithmic depth.

We now consider a sketch of an algorithm for solving the 1 vs. 2 cycle problem in the strongly sublinear regime.

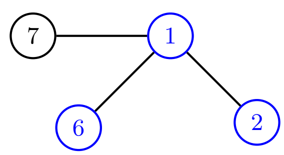
1 vs. 2 cycle Algorithm (1v2ALG)

- For some $O(\log n)$ steps
  1. Mark every vertex with prob. $\frac{1}{2}$ independently.
  2. Every unmarked vertex v that has at least one marked neighbor u is contracted to u, breaking ties arbitrarily.
- Return the number of remaining vertices

For example, consider executing this algorithm on the graph below, where vertices $\{1, 2, 6\}$ are marked initially, and the neighbors of these marked nodes are then contracted:



resulting in the following graph (post contraction):



Claim 4. *Take any vertex v not in a singleton component. Then v is removed in the next iteration of 1v2ALG with probability at least $1/4$.*

Proof. Take any neighbor u of v. If v is not marked and u is marked, then v is removed. This will occur with probability $1/4$, since each vertex is marked with prob. $1/2$. $\square$

Consider a component C_i in iteration i where $|C_i| > 1$. Let C_{i+1} be the same component after round $i+1$ completes. From the above we get that

$$\mathbb{E}\left[|C_{i+1}|\right] \leq \frac{3}{4}|C_i|$$

Now, take a component C_0 in the original graph. We have that

$$\mathbb{E}\left[|C_i|\right] \leq \frac{3}{4}\,\mathbb{E}\left[|C_{i-1}|\right] \leq \left(\frac{3}{4}\right)^2 \mathbb{E}\left[|C_{i-2}|\right] \leq \dots$$

so if we do this for $r = 10 \log n$ rounds, we have

$$\mathbb{E}\left[|C_r|\right] \leq \frac{3}{4}^{10 \log n} |C_0| \leq \frac{1}{n^3} \cdot n = \frac{1}{n^2}$$

So $Pr[C_r > 1] \leq \frac{1}{n^2}$ i.e., a connected component has size larger than 1 after r rounds with low probability. Taking a union bound over at most n components implies that with prob. $1 - \frac{1}{n}$ all components become singleton after $O(\log n)$ rounds.

1.1 Diameter Parameterization

The *diameter* D of a graph G is the "longest shortest path" in G. We can parameterize the complexity based on graph diameter.

> **Remark.** It is open whether there is an $O(\log D)$ round algorithm for graph connectivity using $O(n^\epsilon)$ space.

Theorem 5 ([ASS+18]). *Graph connectivity can be solved in $O(\log D \cdot \log\log n)$ rounds with $O(m)$ total space and $O(n^\epsilon)$ space per machine.*

Theorem 6 ([BDE+19]). *Graph connectivity can be solved in $O(\log D)$ rounds with $O(m)$ total space and $O(n^\epsilon)$ space per machine.*

Note that both of the algorithms from the above results use randomization.

Theorem 7 ([CC22]). *Graph connectivity can be solved in $O(\log D + \log\log n)$ rounds with $O(m)$ total space and $O(n^\epsilon)$ local space deterministically.*

The following key ideas are used in the results of Theorem 5.

- **Idea 1 (Random contraction)**: If every vertex has degree $\geq b$, then we can mark every vertex with probability $\frac{1}{b}$ as opposed to marking with prob. $\frac{1}{2}$.
- **Idea 2 (Graph exponentiation)**: Within $O(\log D)$ rounds we can increase the degree of every vertex to $\sqrt{b}$ using $O(b)$ space per machine.

References

[ASS+18] Alexandr Andoni, Zhao Song, Clifford Stein, Zhengyu Wang, and Peilin Zhong. Parallel graph connectivity in log diameter rounds. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 674–685, 2018. 3

[BDE+19] Soheil Behnezhad, Laxman Dhulipala, Hossein Esfandiari, Jakub Lacki, and Vahab Mirrokni. Near-optimal massively parallel graph connectivity. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1615–1636. IEEE, 2019. 3

[BIS90] David A Mix Barrington, Neil Immerman, and Howard Straubing. On uniformity within nc1. *Journal of Computer and System Sciences*, 41(3):274–306, 1990. 1

[CC22] Sam Coy and Artur Czumaj. Deterministic massively parallel connectivity. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 162–175, 2022. 3

[GU19] Mohsen Ghaffari and Jara Uitto. Sparsifying distributed algorithms with ramifications in massively parallel computation and centralized local computation. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '19, page 1636–1653, USA, 2019. Society for Industrial and Applied Mathematics. 1