This talk will be about the Ullman-Yannakakis algorithm, a parallel randomized algorithm to solve unweighted shortest paths.

0 Recap from Part 1

- Sequential BFS and time bounds.
- Parallel BFS and time bounds.
- Tropical matrix multiplication and time bounds – note that this algorithms solves all pairs shortest paths in weighted graphs.

1 Preliminary: Parallel $k$-limited search

- Just parallel BFS, but stopping at level $k$.
- Takes $O(m/p + k)$ time and $O(m + kp)$ work.

2 Ullman-Yannakakis algorithm

We take advantage of the tradeoff of the first two parallel algorithms by using the high-work matrix multiplication algorithm a smaller graph, then stitching the shortest path together using this information.

- Choose $2\sqrt{n}$ distinguished vertices at random, and include $s$ in the list of distinguished vertices.
- Perform a $(\sqrt{n}\log n)$-limited search from each distinguished vertex. Using $O(m/\sqrt{n})$ processors per vertex, this can be done in $O(\sqrt{n}\log n)$ time and $O(m\sqrt{n}\log n)$ work.
- Create an auxiliary weighted graph $H$ out of the distinguished vertices, with edge weight $w_{ij}$ equal to the minimum distance from $s_i$ to $s_j$ found in the previous step, or $\infty$ if no path was found.
- Use the tropical matrix multiplication algorithm to compute all-pairs shortest paths in $H$. This takes $O(\log n)$ time using $O(n^{3/2})$ processors.
- Output the minimum over all distinguished vertices $u$ of $d(s, u; H) + d(u, t)$, where $d(s, u; H)$ is the shortest path length from $s$ to $u$ in $H$ that was found in the previous step.
- Total time: $O(\sqrt{n}\log n)$; total work: $O(m\sqrt{n}\log n)$.

Correct of this algorithm is based on the following lemma:
Lemma (Ullman and Yannakakis). With probability at least $1 - \frac{1}{n}$, there is a shortest s-t path such that every subpath of size $\sqrt{n \log n}$ contains a distinguished vertex.

The lemma leads to the following theorem.

Theorem (Ullman and Yannakakis). This algorithm is correct with probability at least $1 - \frac{1}{n}$.

Proofs of Lemma and Theorem. Discussed in lecture.