Albert-Ludwigs-Universität, Inst. für Informatik
Prof. Dr. Fabian Kuhn
Philipp Bamberger

Distributed Systems, Summer Term 2020
Exercise Sheet 12

1 Borůvka's MST Algorithm in the Strongly Sublinear MPC Model

We consider the strongly sublinear memory regime of the MPC model, i.e., given an input graph with n nodes and m edges, the set of edges is uniformly distributed among $O(m/S)$ machines with a memory size $S = n^\alpha$ for an $0 < \alpha < 1$. To simplify the calculations, we will assume that initially, the edges are partitioned amog at most $4m/S$ machines.

In class, we sketched how to implement one phase of Borůvka's MST algorithm in $O(1)$ time in this setting. We will now look at this more closely. We first recap what a single phase of the algorithm has to achieve:

1. The nodes are partitioned into the already computed fragments of the MST. Assume that each MST fragment has a unique fragment ID and that at the beginning of the phase, for each edge $\{u, v\} \in E$, the machine that stores the edge $\{u, v\} \in E$ also knows the fragment IDs of the nodes u and v.

2. For each fragment F, we need to find the minimum weight edge that connects a node in F with a node in another fragment. (Recall that we assume that the edge weights are distinct and therefore the minimum weight outgoing edge of a fragment F is unique.)

3. We randomly label each fragment either red or blue (independently with probability 1/2 each). If fragment F is red and its minimum weight outgoing edge e connects to a blue fragment F', we add e to the MST. The new fragments after the phase are given by the forest that is induced by the old fragments and the newly added edges.

4. The algorithm needs to compute a new unique fragment ID for each of the new fragments and it needs to make sure that for every edge $\{u, v\} \in E$, the machine that stores the edge $\{u, v\} \in E$ also knows the new fragment IDs of the nodes u and v.

For implementing a phase, we intend to use a structure consisting of $O(\log_S m)$ levels of $O(m/S)$ machines each (for simplicity, assume that we have disjoint sets of machines for the different levels). We number the levels from bottom to top from 1 to $\ell = O(\log_S m)$. Assume that level 1 is exactly given by the $O(m/S)$ machines that initially store the graph. We say that a machine μ on a level i is responsible for a fragment ID x if it participates in the computation for the fragment with ID x. On level 1, a machine μ is responsible for fragment ID x if and only if it initially stores an edge $\{u, v\}$ such that either u or v are in fragment with ID x.

For each fragment ID x and each level i for $1 \leq i < \ell$, each machine μ on level i that is responsible for x has a parent machine w.r.t. fragment ID x on level $i + 1$. On level $i \geq 2$, a machine μ is called responsible for fragment ID x if there exists a machine μ' on level $i - 1$ such that μ is the parent w.r.t. x of μ'. The indegree of a machine μ on level $i \geq 2$ is the number of pairs (μ', x) such that μ is the parent w.r.t. x of machine μ' on level $i - 1$. We make the following assumptions:

1. The indegree of every machine μ on some level $i \geq 2$ is at most S

2. For each fragment ID x, the number of machines responsible for fragment ID x on level $i \geq 1$ is $m/(S/4)^i$ (for simliciy, we ignore rounding non-integer values to integer values). Further, on the top level ℓ, there is exactly one machine that is responsible for each fragment ID x.

(a) Argue in detail how a single phase of the MST algorithm can be carried out in $O(1)$ rounds in the MPC model if above structure is given.

(b) As the second part of the question, we discuss how to build the structure on machines described above. Assume that the machines have access to shared randomness. Assume that on each level we use $4m/S$ machines. For each fragment ID x, the set of machines responsible for fragment ID x on level $i \geq 2$ is a uniformly random subset of size $m/(S/4)^i$ of the $4m/S$ machines of level i (on level ℓ, we choose one random machine for each fragment ID x). Such a set can be chosen by using the shared randomness (you do not have to worry about the details of this). Each machine on level $1 \leq i < \ell$ chooses its parents on level $i+1$ as follows. For each fragment ID x for which a machine μ on level i is responsible, the machine μ chooses the parent μ' w.r.t. x on level $i+1$ uniformly at random among the machines responsible for x on level $i+1$. Show that with this randomized construction, for every level i and every machine μ on level i, the expected indegree of μ is at most $S/2$.

Sample Solution

a) For each fragment ID x, each machine on level 1 sends the minimum weight edge with exactly one endpoint in x that it stores to its parent. All machines on the other levels do the same once they received edges from their children. Let $\{u, v\}$ be the minimum weight each that the machine on the top level receives for x, with u belonging to x and v belonging to another fragment y. The machines colors x and y red/blue with probability 1/2 each (using shared randomness, this can be done consistently in parallel on different machines). If x is red and y is blue, $\{u, v\}$ is added to the MST and fragment ID x is changed to y. This information is broadcasted down the tree to every machine that stores an edge belonging to x.

The depth of the tree is

$$\log_{S/4}(m) = \frac{\log_S(m)}{\log_S(S/4)} = \frac{\log_S(m)}{1 - \log_S(4)} \overset{(*)}{\leq} 2\log_S(m) \leq 2\log_S(n^2) = 4\log_S(n) = 4\log_{n^\alpha}(n) = \frac{4}{\alpha} ,$$

so a constant number of communication rounds is needed.

$(*)$: If $S \geq 16$, then $\log_S(4) \leq 1/2$, i.e., $1 - \log_S(4) \geq 1/2$.

b) Let μ be a machine on level $i > 1$ and x the ID of some fragment. On level i, there are $m/(S/4)^i$ uniformly random machines among $m/(S/4)$ machines responsible for x.

$\Pr(\mu \text{ is resp. for } x) = \frac{m/(S/4)^i}{m/(S/4)} = \frac{4^{i-1}}{S^{i-1}}$

$E[\#\text{children of } \mu \mid \mu \text{ is resp. for } x] = \frac{\#\text{ machines resp. for } x \text{ on level } i-1}{\#\text{ machines resp. for } x \text{ on level } i} = \frac{S}{4}$

$E[\#\text{children of } \mu \text{ for } x] = \Pr(\mu \text{ is resp. for } x) \cdot E[\#\text{children of } \mu \mid \mu \text{ is resp. for } x] = \frac{4^{i-2}}{S^{i-2}}$

$E[\text{indegree of } \mu] = \#\text{fragments} \cdot \frac{4^{i-2}}{S^{i-2}} \leq n \cdot \frac{4^{i-2}}{S^{i-2}} = S^{1/\alpha} \cdot \frac{4^{i-2}}{S^{i-2}} \overset{2 \leq i \leq 4/\alpha}{\leq} \cdots \leq \frac{S}{2}$

Alternative:

$$
\begin{aligned}
\Pr(\mu' \text{ is child of } \mu) &\leq \sum_x \big(\Pr(\mu' \text{ and } \mu \text{ are resp. for } x) \cdot \Pr(\mu' \text{ chooses } \mu \mid \text{ both are resp. for } x)\big) \\
&= \sum_x \big(\Pr(\mu' \text{ is resp. for } x) \cdot \Pr(\mu \text{ is resp. for } x) \cdot \Pr(\mu' \text{ chooses } \mu \mid \text{ both are resp. for } x)\big) \\
&= \sum_x \frac{m/(S/4)^i}{m/(S/4)} \cdot \frac{m/(S/4)^{i-1}}{m/(S/4)} \cdot \frac{1}{m/(S/4)^i} = \sum_x \frac{m/(S/4)^{i-1}}{(m/(S/4))^2} = \sum_x \frac{1}{m \cdot (S/4)^{i-3}} \\
&\leq \frac{n}{m \cdot (S/4)^{i-3}}
\end{aligned}
$$

$E[\#\text{children of } \mu] \leq \frac{4m}{S} \cdot \frac{n}{m \cdot (S/4)^{i-3}} = n \cdot \frac{4^{i-2}}{S^{i-2}}$