Exercise 1: Aggregation in the MPC Model

Assume you are given a number of $M \in \tilde{O}(\frac{N}{S})$ machines (you may freely choose the hidden poly(log $N$) factor in the number of machines), where $N$ is the number of aggregation messages that are collectively stored by the machines $M_i$, $i \in [M]$ and each machine $M_i$ has a memory large enough to store $S$ messages. We have $N \gg S$ and $S \in \tilde{\Omega}(1)$ (for a hidden poly(log $N$) factor of your choosing). By definition of the MPC model every machine can send and receive at most $S$ aggregation messages.

Each aggregation message has encoded within it a target machine $i \leq N$. Additionally each aggregation message has a value associated with it. The aggregation problem is solved as soon as each machine learns an aggregation message that has minimal value among all aggregation messages of which it is the target. Carefully formulate a (randomized) algorithm that solves said aggregation problem in $O(\log N)$ and prove its correctness.

Hint: You may assume that messages have encoded within them the total number of messages with the same target. Machines have numbers $1, \ldots, M$ that they are aware of. This allows that machine no. 1 computes and distributes public random bits (assume that you have arbitrary public randomness).

Sample Solution

Solution 1: (Randomized solution).
Our algorithm works as follows. First we generate a sufficiently long string of random bits (combining those with some hash functions, $O(\log N)$ random bits suffice) at some machine, which distributes it to all other machines. Our algorithm will work for $M := \ell N/S$ and $S \geq 2\ell$, where $\ell := (c \log N)^2$ is polylogarithmic in $N$. Let $N_i$ be the number of messages with target node $i$ (which is known by the respective machines due to the hint). The following aggregation protocol proceeds in iterations $j = 1, \ldots, T \in O(\log N)$ (picture this as the $j$-th layer of an aggregation tree).

Step 1 Based on the public randomness we determined earlier, each machine picks a random subset $M_{i,j} \subseteq [M]$ with $|M_{i,j}| = \frac{N_i \ell}{S}$, such that the chosen sets $M_{i,j}$ are equal for all machines.

Step 2 Then every machine sends the/a minimal message intended for target $i$ to some machine picked uniformly at random from the set $M_{i,j}$.

Step 3 Subsequently, every machine keeps only the messages it receives and deletes all other messages. In the next round all machines do the same with $j = j+1$.

After step $j$ all (minimal) messages from machines with target node $i$ will be consolidated in the set $M_{i,j}$. Finally, in step $T$ all machines send their minimal messages to their intended targets.

The correctness of the algorithm is implied by the following claims, which we will prove subsequently.

(i) After $T \in O(\log N)$ iterations we have $|M_{i,j}| \leq S$. 
(ii) No machine will receive more than $S$ messages in step 2, w.h.p.

Towards (i): Since we required $S \geq 2\ell$, we have that $|\mathcal{M}_{i,j}| \leq N_i/2^j$. Therefore, after at most $\log_2(N_i)$ steps, we have $|\mathcal{M}_{i,j}| \leq S$. Note that if we are given much larger $S \in \Omega(N^\alpha)$ for some constant $\alpha > 0$, the number of iterations will be much smaller in $O(\log S N = 1/\alpha)$.

Towards (ii): We chose $|\mathcal{M}_{i,j}| = N_i \ell^j / S^j$. Hence, the probability that some machine will be put into a fixed set $\mathcal{M}_{i,j}$ is $p_{i,j} = |\mathcal{M}_{i,j}| / M$ (a priori, every machine has the same chance to be picked for $\mathcal{M}_{i,j}$).

Let $X_j$ be the random number of sets $\mathcal{M}_{i,j}$ for $i \in [N]$ that a given machine will be a member of.

$$\mathbb{E}(X_j) = \sum_{i=1}^{N} p_{i,j} = \sum_{i=1}^{N} \frac{|\mathcal{M}_{i,j}|}{M} = \sum_{i=1}^{N} \frac{N_i \ell^j}{M} \cdot \frac{S^j}{S^j} = \frac{N_i \ell^j}{M} \cdot \frac{S^j}{S^j} \leq \frac{N \ell}{M} \cdot \frac{S}{S} = 1.$$ 

With a Chernoff bound, the probability that a machine is in more than $c \log N$ sets $\mathcal{M}_{i,j}$ is at most

$$\mathbb{P}(X_j \geq c \log N) \leq \mathbb{P}(X_j \geq (1 + \frac{c}{2} \log N) \mathbb{E}(X_j)) \leq \frac{1}{N^{c/6}}.$$ 

Let $q_{i,j}$ be the probability that a machine that is within $\mathcal{M}_{i,j}$ is picked as target of some message that is sent to $\mathcal{M}_{i,j}$. We have $q_{i,j} = 1/|\mathcal{M}_{i,j}|$. Let $Y_{i,j}$ be the according number of messages with target $i$ received by a node in $\mathcal{M}_{i,j}$. The expectation is $\mathbb{E}(Y_{i,j}) = |\mathcal{M}_{i,j-1}| / |\mathcal{M}_{i,j}| = S/\ell$. With a Chernoff bound, the probability that a machine receives more than $S$ messages is

$$\mathbb{P}(Y_{i,j} \geq S/c \log N) = \mathbb{P}(Y_{i,j} \geq c \log N \mathbb{E}(Y_{i,j}))$$

$$\leq \mathbb{P}(Y_{i,j} \geq (1 + \frac{c}{2} \log N) \mathbb{E}(Y_{i,j}))$$

$$= \exp\left(-\frac{c \log N \mathbb{E}(Y_{i,j})^2}{6}\right)$$

$$= \exp\left(-\frac{c \log N \cdot S^2}{6 \ell}\right) \leq \exp\left(-\frac{c \log N}{3}\right) = \frac{1}{N^{c/3}}.$$ 

We union bound over all of the above events (the number of events is polynomial in $N$) so we have that in any iteration $j$, any machine is in at most $c \log N$ sets $\mathcal{M}_{i,j}$ and receives at most $S/c \log N$ messages within each set $\mathcal{M}_{i,j}$, w.h.p. Thus, any node receives at most $S$ aggregation messages, w.h.p.