Abstract

We consider the problem of aggregation in overlay networks. We use a synchronous time model in which each node has polylogarithmic memory and can send at most a polylogarithmic number of messages per round. We investigate how to quickly compute the result of an aggregate function f over elements that are distributed among the nodes of the network such that the result is eventually known by a selected root node. We show how to compute *distributive* aggregate functions such as SUM, MAX, and OR in time $O\left(\frac{\log n}{\log\log n}\right)$ using a tree with degree $O(\log n)$ that is created in a pre-processing phase. If only a polylogarithmic number of data items need to be aggregated, we show how to compute the result in time $O\left(\sqrt{\frac{\log n}{\log\log n}}\right)$. Furthermore, we show how to compute *holistic* aggregate functions such as DISTINCT, SMALLEST(k), and MODE(k) in time $O\left(\frac{\log n}{\log\log n}\right)$. Finally, we show a lower bound of $\Omega\left(\sqrt{\frac{\log n}{\log\log n}}\right)$ for deterministic algorithms that compute any of the aggregate functions in the scope of the thesis.