Statistics. Assignment 1 report.

Alexandr Grichshenko

Innopolis University, Innopolis 420500, Russia

This section introduces the chosen method for this assignment as described in the paper by Chen and Zhang [1]. I have decided to implement a *q-digest* method for quantile computations in distributed systems. The algorithm utilises a binary tree structure and compression technique to store data in an efficient way, reducing the storage space. Two main parameters of the *q-digest* data structure are the size of the fixed-universe σ and the compression factor k. Since *q-digest* is meant to be used in distributed applications, the number of separate trees $|v|$ present in the system is also relevant.

According to the paper, the time complexity of quantile computation from the merged trees is $O(|v|log(\sigma))$. In my implementation I have simulated the merging and quantile computations with $|v| \in [2, 20]$, $\sigma = 10$. The resulting normalised time complexity plot is presented in Figure 1

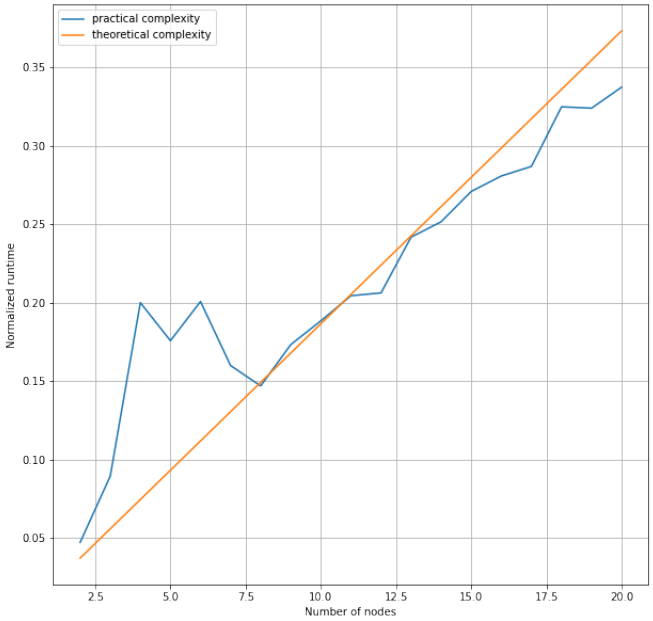

Fig. 1: Time complexity plot

From the graph one can observe that the implementation follows the theoretical complexity boundaries presented in the paper.

References

1. Chen, Z., Zhang, A.: A survey of approximate quantile computation on large-scale data. IEEE Access **8**, 34585–34597 (2020)