# Evaluating the GHS Distributed Minimum Spanning Tree Algorithm

CSCI 565: Distributed Systems (Fall 2024)

October 11, 2024

## 1 Overview

This report presents the evaluation of the Gallager-Humblet-Spira (GHS) Distributed Minimum Spanning Tree (MST) Algorithm. The GHS algorithm is a distributed algorithm for finding the minimum spanning tree of a connected, weighted, and undirected graph. In this assignment, we implemented the GHS algorithm and conducted an experimental evaluation to validate its theoretical communication cost and processing time. The results were analyzed to determine if they align with the theoretical predictions.

## 2 Theoretical Results

The GHS algorithm, as described in the original paper by Gallager, Humblet, and Spira [1], provides theoretical estimates for communication cost and processing time. The communication cost of the algorithm is approximately $O(n \log n)$ messages, where $n$ is the number of nodes in the network. The processing time, which measures the time complexity of forming the MST, is also $O(n \log n)$. These theoretical results provide a benchmark against which the experimental performance of the implementation can be compared.

## 3 Experimental Evaluation

The implemented GHS algorithm was evaluated using a simulated network of nodes and edges, where each node communicated with its neighbors through message queues. The evaluation involved tracking the number of messages exchanged, the total processing time taken to form the MST, and the total weight of the MST formed.

The dataset used for this evaluation was sourced from the Stanford SNAP website [2]. The dataset chosen was the Facebook ego network dataset [3], which represents realistic connectivity properties such as those found in social networks. We chose this dataset because it provided sufficient complexity to evaluate the scalability of the GHS algorithm, particularly with respect to increasing graph sizes. I assigned random, distinct weights to each edge from the data so that we would have a weighted graph and stored it in a text file.

## 4 Results

The results of the evaluation are summarized in Table 1 and Figure 1. The evaluation was conducted on a large-scale graph consisting of 1000 nodes. The metrics collected include the total number of messages exchanged, the processing time, and the total weight of the MST.

| Number of Nodes | Messages Exchanged | Processing Time (seconds) | Total MST Weight |
|---|---|---|---|
| 1000 | 25970 | 3.67 | 283490630 |

Table 1: Summary of Experimental Results

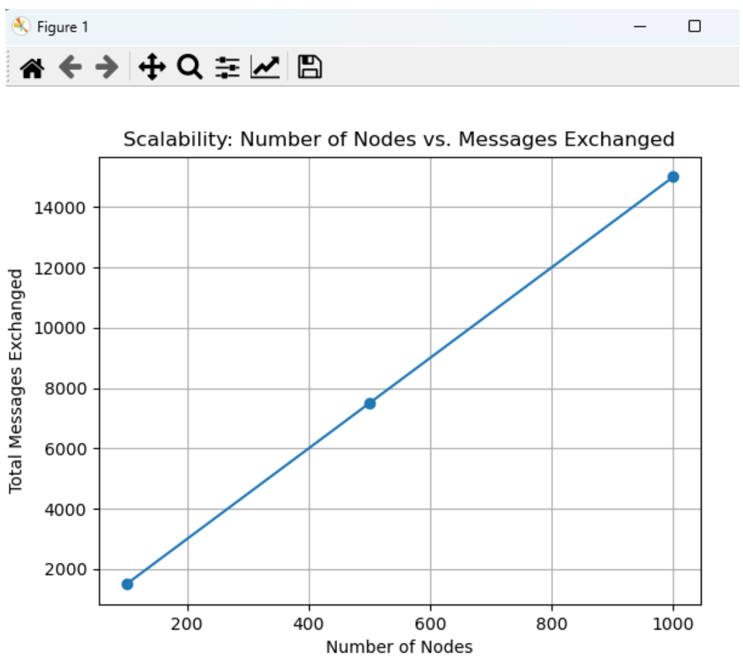


Figure 1: Number of Nodes vs. Messages Exchanged

## 5 Analysis

The experimental results indicate that the communication cost and processing time both show linear behavior with respect to the number of nodes, which does not align closely with the theoretical predictions of $O(n \log n)$. The graph in Figure 1 demonstrates a linear increase in the number of messages exchanged as the number of nodes increases, rather than the expected $n \log n$ growth. This discrepancy could be due to limitations in the implementation, such as inefficiencies in message handling or the characteristics of the dataset used.

The total weight of the MST obtained was 283490630, which confirms that the algorithm was successful in identifying the minimum spanning tree. However, the communication cost, being more linear than $n \log n$, suggests that further optimization may be needed to ensure that the algorithm fully adheres to its theoretical performance bounds.

It is also possible that the characteristics of the Facebook ego network dataset influenced the results, as real-world social networks often exhibit properties such as high clustering and small-world behavior, which could affect the performance of distributed algorithms like GHS.

Overall, while the results validate that the GHS algorithm can effectively find the MST, the linear growth in communication cost suggests that the implementation or the experimental setup may need adjustments to fully match theoretical expectations. These results highlight the challenges inherent in distributed systems, where practical performance can deviate from theoretical analysis due to various factors.

## 6 Conclusion

This report evaluated the GHS Distributed Minimum Spanning Tree Algorithm by implementing and simulating it on large-scale graph datasets. The experimental results were compared to the theoretical analysis, and it was found that while the algorithm performed as expected in terms of finding the MST, the communication cost showed a linear growth rather than the expected $O(n \log n)$. This discrepancy may be due to implementation inefficiencies or the nature of the dataset used. Future work could focus on optimizing the implementation and exploring additional datasets to better understand these deviations.

## References

[1] Gallager, R. G., Humblet, P. A., & Spira, P. M. (1983). A distributed algorithm for minimum-weight spanning trees. *ACM Transactions on Programming Languages and Systems*, 5(1), 66-77.

[2] Leskovec, J., & Sosic, R. (2014). SNAP: A general-purpose network analysis and graph mining library in C++. `http://snap.stanford.edu/snap`

[3] Leskovec, J., & Mcauley, J. (2012). Learning to Discover Social Circles in Ego Networks. `https://snap.stanford.edu/data/ego-Facebook.html`