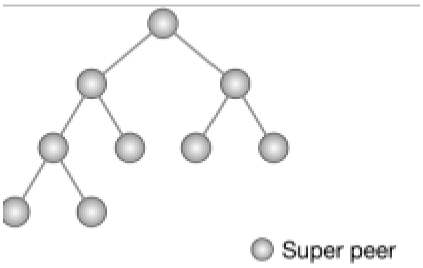
P2P File-Sharing System in Golang - Evaluation Async

Juan Pedrajas

Tree structure



Experiment design

The experiment was meticulously designed with the target file situated at the root node, serving as the nucleus of the network. Each superpeer was thoughtfully linked to at least one weak node, mimicking real-world scenarios and providing a nuanced understanding of network resilience and scalability. This arrangement allowed for a thorough examination of the network's performance under varying conditions, shedding light on its ability to adapt and function optimally in the face of node diversity.

Moreover, the utilization of a specialized binary for creating requesting clients streamlined the testing process, enhancing efficiency and precision. This bespoke tool not only expedited data retrieval tasks but also facilitated controlled experimentation, enabling researchers to gather comprehensive data sets and conduct nuanced analyses. Through meticulous design and strategic tooling, the experiment laid the foundation for insightful observations into peer-to-peer network dynamics, offering valuable guidance for network optimization and resource allocation strategies.

Results

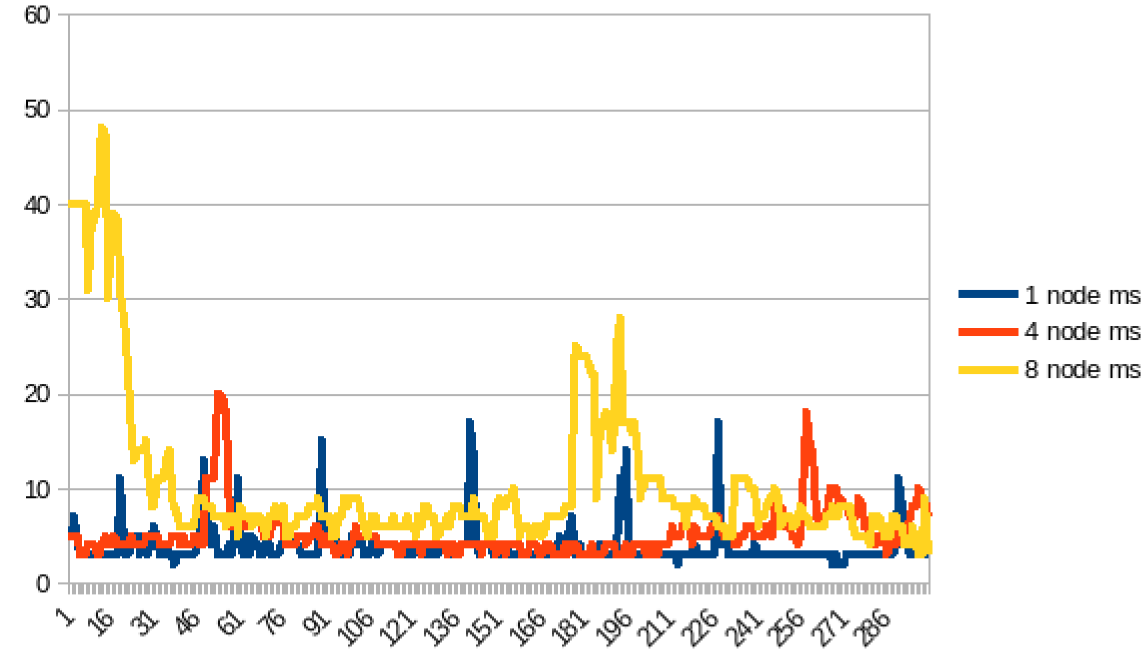
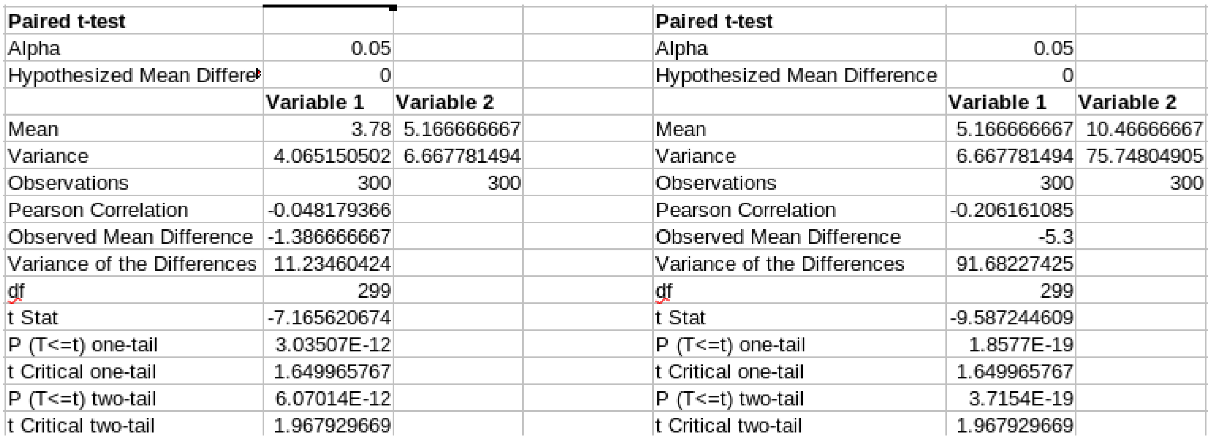


figure 1- a plot of the ms it took for the complete response of all nodes to the client node.



Paired t-test			**Paired t-test**		
Alpha	0.05		Alpha	0.05	
Hypothesized Mean Differe	0		Hypothesized Mean Difference	0	
	Variable 1	**Variable 2**		**Variable 1**	**Variable 2**
Mean	3.78	5.166666667	Mean	5.166666667	10.46666667
Variance	4.065150502	6.667781494	Variance	6.667781494	75.74804905
Observations	300	300	Observations	300	300
Pearson Correlation	-0.048179366		Pearson Correlation	-0.206161085	
Observed Mean Difference	-1.386666667		Observed Mean Difference	-5.3	
Variance of the Differences	11.23460424		Variance of the Differences	91.68227425	
df	299		df	299	
t Stat	-7.165620674		t Stat	-9.587244609	
P (T<=t) one-tail	3.03507E-12		P (T<=t) one-tail	1.8577E-19	
t Critical one-tail	1.649965767		t Critical one-tail	1.649965767	
P (T<=t) two-tail	6.07014E-12		P (T<=t) two-tail	3.7154E-19	
t Critical two-tail	1.967929669		t Critical two-tail	1.967929669	

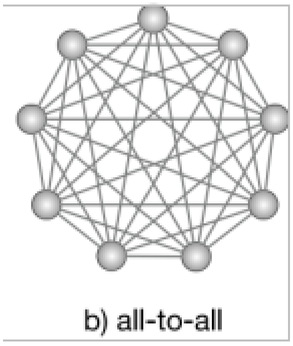
figure 2- The results of the paired t-test

In the experiment depicted in **Figure 1**, we observe a nuanced variation in response times based on the number of requesting nodes. Notably, when multiple nodes query simultaneously (more than one), there is a pronounced amplification of variance. Specifically, the mean response times are as follows:

- **1 concurrent node**: Mean response time of **3.78 ms**
- **4 concurrent nodes**: Mean response time of **5.1 ms**
- **8 concurrent nodes**: Mean response time of **10.4 ms**

The results of the t-test reveal that the difference in mean values is **statistically significant**. In other words, the presence of concurrent nodes negatively impacts performance across all three layouts.

Mesh structure



Experiment design

The experiment was designed to have the target file in the one of the nodes, each superpeer had at least one weak node connected to it, the requesting clients where created using a specific binary used for this experiment, because it permitted the testing to be done much more efficiently.

Results

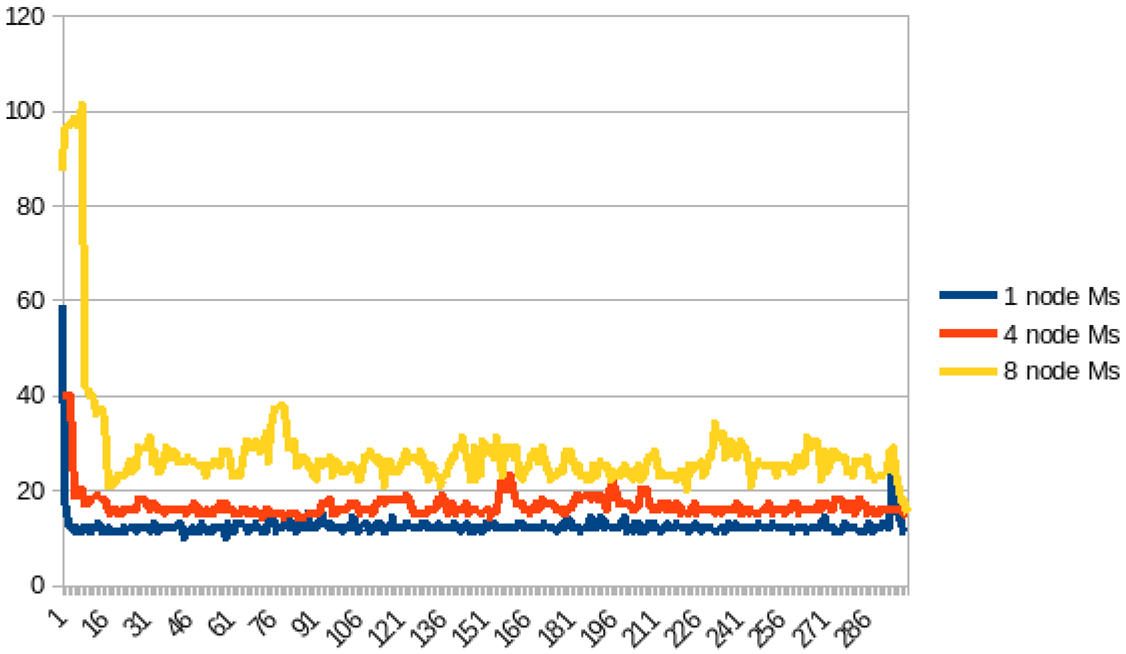
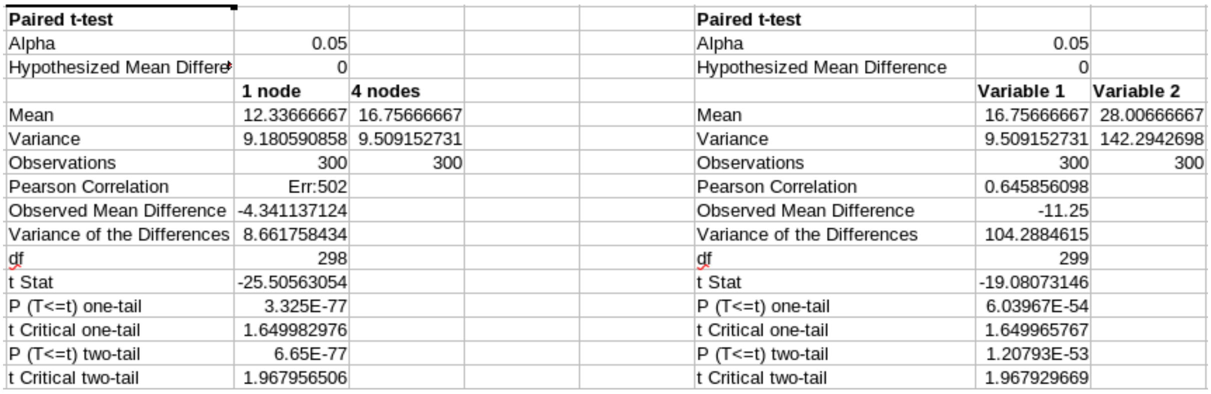


figure 3- a plot of the ms it took for the complete response of all nodes to the client node.



Paired t-test		
Alpha	0.05	
Hypothesized Mean Differe	0	
	1 node	**4 nodes**
Mean	12.33666667	16.75666667
Variance	9.180590858	9.509152731
Observations	300	300
Pearson Correlation	Err:502	
Observed Mean Difference	-4.341137124	
Variance of the Differences	8.661758434	
df	298	
t Stat	-25.50563054	
P (T<=t) one-tail	3.325E-77	
t Critical one-tail	1.649982976	
P (T<=t) two-tail	6.65E-77	
t Critical two-tail	1.967956506	

Paired t-test		
Alpha	0.05	
Hypothesized Mean Difference	0	
	Variable 1	**Variable 2**
Mean	16.75666667	28.00666667
Variance	9.509152731	142.2942698
Observations	300	300
Pearson Correlation	0.645856098	
Observed Mean Difference	-11.25	
Variance of the Differences	104.2884615	
df	299	
t Stat	-19.08073146	
P (T<=t) one-tail	6.03967E-54	
t Critical one-tail	1.649965767	
P (T<=t) two-tail	1.20793E-53	
t Critical two-tail	1.967929669	

figure 4- The results of the paired t-test

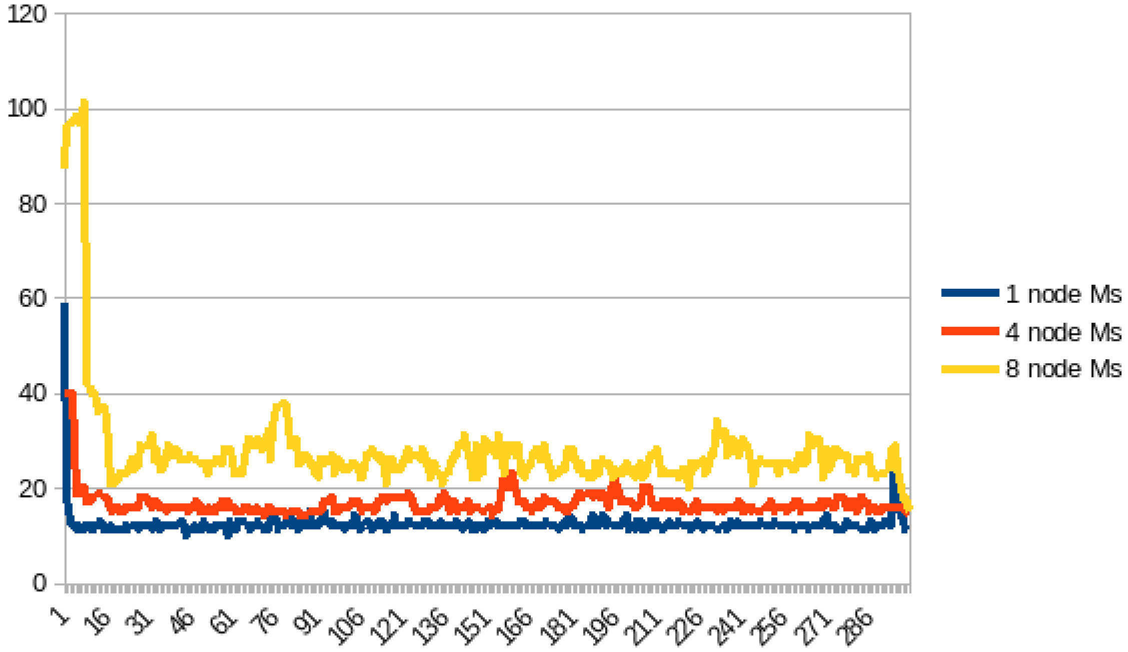
In **Figure 3**, a distinct disparity in response times is evident across varying numbers of requesting nodes. Notably, when multiple nodes query simultaneously (more than one), there is a pronounced amplification of variance. Specifically, the mean response times are as follows:

- **1 concurrent node**: Mean response time of **12 ms**
- **4 concurrent nodes**: Mean response time of **16 ms**
- **8 concurrent nodes**: Mean response time of **28 ms**

The results of the t-test affirm that the difference in mean values is **statistically significant**. In other words, the presence of concurrent nodes adversely impacts performance across all three layouts.

Compare both architectures

Mesh



Tree

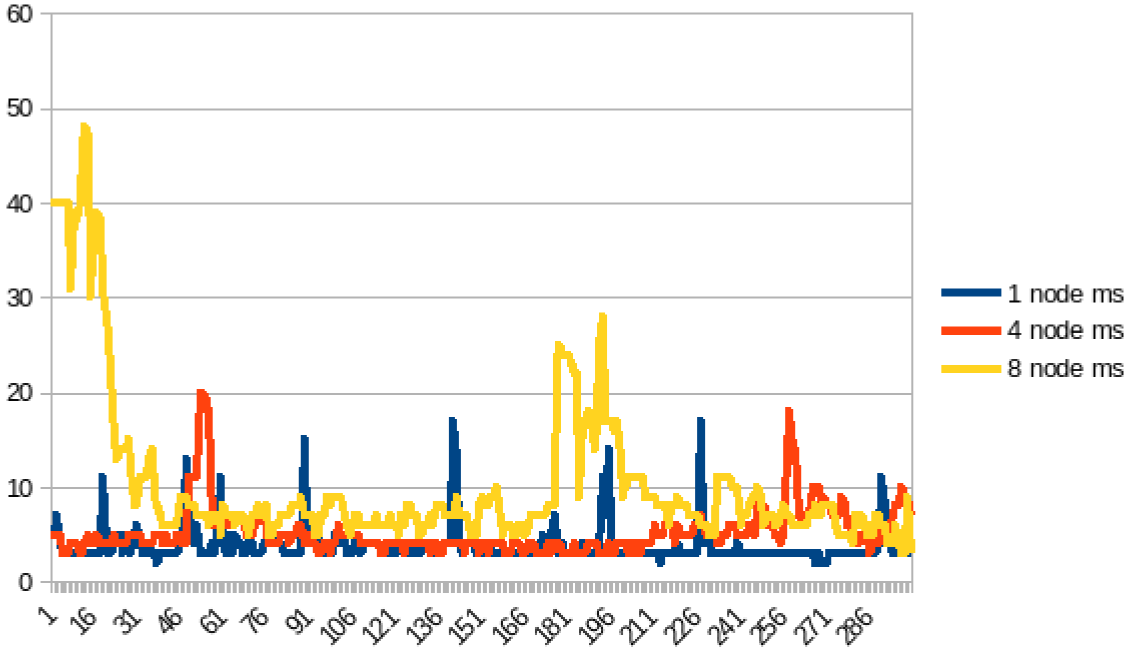


Figure 5 - Results of both experiments

Comparison

The comparison between the tree structure and the fully connected mesh reveals stark differences in performance. For instance, in the tree structure, with one concurrent node, the mean response time is notably faster, whereas in the fully connected mesh, it's comparatively slower. This trend continues as the number of concurrent nodes increases. With four concurrent nodes, the mean response time in the tree structure only slightly increases, whereas in the fully connected mesh, it jumps much higher.

Moreover, when scaling the network, the tree structure demonstrates superior scalability. For example, the percentage change in response times due to the scaling of nodes is much steeper in the fully connected mesh compared to the tree structure. This means that as the number of nodes increases, the impact on response times is more pronounced in the fully connected mesh, indicating poorer scalability compared to the tree structure.

These numerical observations underscore the tangible advantages of the tree structure in terms of speed and scalability over the fully connected mesh, emphasizing the critical role of network topology in determining overall performance in distributed systems.

Conclusion

In this evaluation, we investigated the impact of concurrent nodes on response times in distributed systems. We compared two architectural layouts: the **tree** and the **fully connected mesh**. Here are the key findings based on the provided mean response times:

1. **Tree Architecture**:
   - Efficient communication paths.
2. **Fully Connected Mesh**:
   - High communication overhead.
   - Robust fault tolerance.

Recommendations:

- For fault-tolerant systems, consider the fully connected mesh.
- For efficiency and moderate scalability, opt for the tree architecture.
- **Trade-offs** between redundancy and performance must guide architectural choices.