Yuese Li

Comparing Message Passing Efficiency in Lab vs. Cluster Environments

This analysis explores the efficiency of message passing within two distinct computing settings: Computer Science lab and a specialized computing cluster. The time it took for messages to travel among different numbers of processes, ranging from small groups of 4 to large assemblies of 256, was measured to understand these dynamics.

The collected data clearly shows that the cluster consistently surpasses the lab in speed. Specifically, with 256 processes, the lab's average message passing time was around 0.238 seconds from 3 samples, while the cluster accomplished the same task remarkably faster, with an average of only 0.011 seconds from 3 samples. This substantial discrepancy indicates that the cluster's network infrastructure is better equipped for handling tasks that involve parallel communications.

Variability in the time measurements, represented by variance, is an important metric for assessing performance stability and reliability. The lab demonstrated greater variability at the 256-process level, with a variance of 0.000922200591 seconds^2, indicating a less consistent performance compared to the cluster at 256-process level, which had a notably lower variance of 0.0000001511743333 seconds^2. This points to the cluster providing a more stable and predictable environment for parallel processing. Therefore, I hypothesize that the Slurm cluster's superior performance is due to its advanced network infrastructure, designed specifically for high-volume, parallel processing tasks.

When comparing the data from this week with last week's, a change in the CS Lab's performance becomes evident. Before, when the number of processes grew from 64 to 256, the time decrease wasn't as big as expected—only a 47.8% drop for "Slices" and 46.7% for "Chunks." This week, the time it took to pass messages for 256 processes went up to 0.237956 seconds from 0.0765 seconds for 128 processes. This is a threefold increase, whereas normally, you'd expect the time to just double if the number of processes doubles for this task. Moreover, the variance increased significantly from 0.000007792826333 to 0.000922200591. This indicates that as more processes were added, the time increased and the lab's performance became less predictable, which might be reaching its bandwidth limitations, or synchronization overheads.

The Borg Cluster, on the other hand, maintains its display of efficiency. The previous week's data showed a nearly 50% reduction in time when doubling the number of processes for both "Slices" and "Chunks," indicative of the cluster's consistent scalability. Even with a new task this week, the cluster's average message passing time for 256 processes stayed impressively low at 0.01094333333 seconds, with a negligible variance of 0.0000001511743333 seconds^2. The sustained low variance implies that the cluster's performance remains stable across different tasks, a robustness not mirrored by the CS Lab.

The observed data suggests that the cluster's network infrastructure, with its efficient data paths and reduced competition for resources, provides a stable environment for parallel processing tasks, an advantage that grows as more processes are added. This robust network is designed for rapid data transfer between computers, featuring specialized pathways and additional capacity not present in the standard lab network. Therefore, for tasks that demand fast and reliable communication across many processors, the cluster is the clear choice. It not only completes tasks faster but also offers consistent performance, which is crucial for achieving predictable outcomes. While the lab can handle simpler tasks, the cluster stands out for complex, communication-intensive processes.

Yuese Li

AI Tools

I used ChatGPT with its data analysis function. I allowed it to analyze my code, assist me in commenting on the code, guide writing functions, and explain the functions to me. ChatGPT also made sure my charts were exactly as they should be.

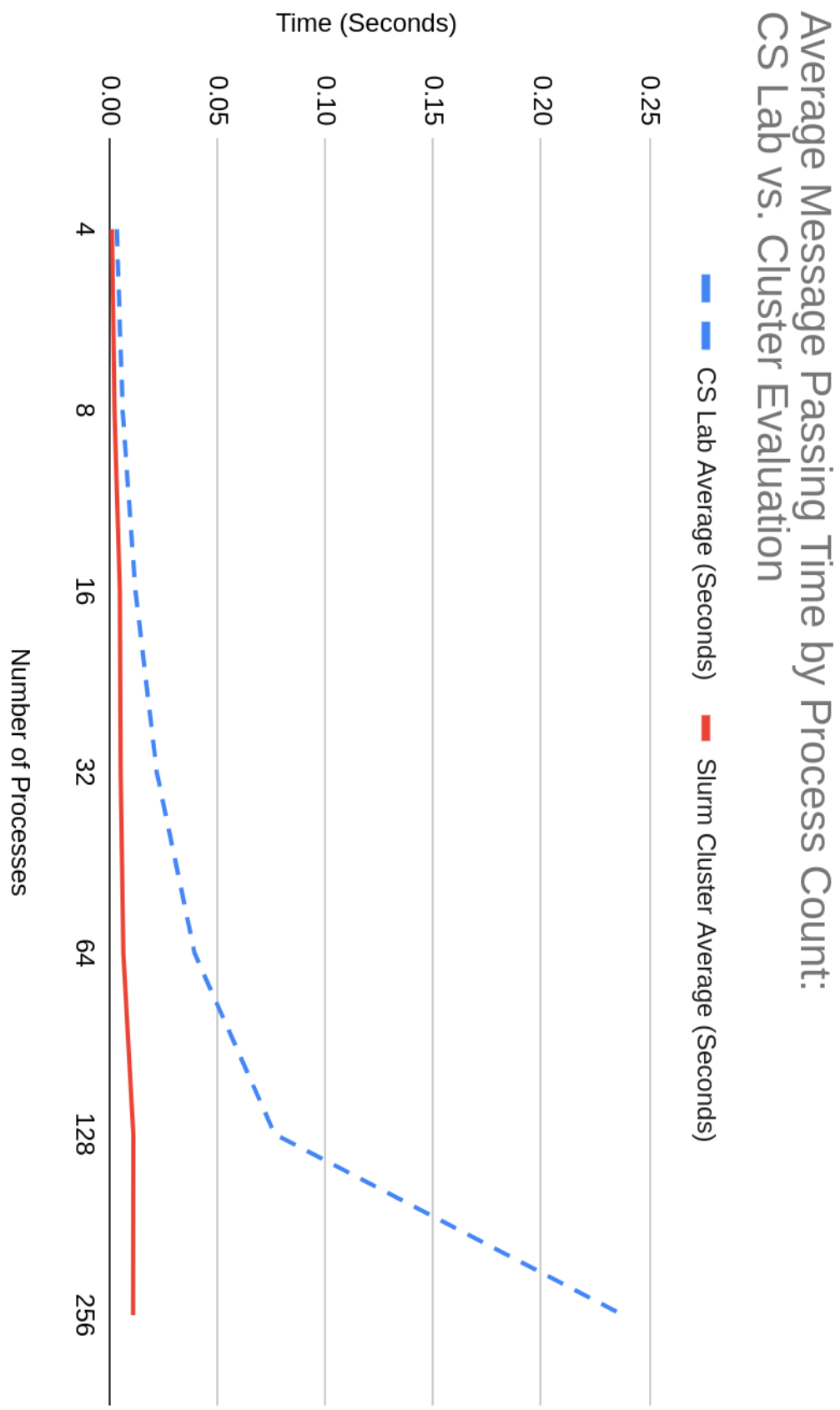
Things went more smoothly this week. Now that I'm getting the hang of the Borg cluster, it's easier than before. ChatGPT was a big help, especially when I forgot the steps to send files back. I sent the wrong one at first, and it pointed out that I needed to send the '.c' file. When I was stuck trying to figure out what went wrong, it seemed to just know the answer. It probably saved me about two hours, helping me understand things better, fixing my writing, and making my analysis clear. I wasn't sure about my analysis at first, but ChatGPT helped me keep checking that I was following the instructions. Helped me where to write hypotheses and cite collected data. It was quick to catch the small, silly mistakes I made. After watching OpenAI dev day last Saturday, it's impressive how much better this is getting, and it’s a bit concerning. It's making learning and doing tasks much easier.

Sometimes, it feels like ChatGPT isn't quite grasping my questions, but I think that's partly because I ask some pretty confusing ones. It doesn't always respond like a person would, which reminds me that there's still a gap between human and artificial intelligence, no matter how advanced it gets. But then again, it's probably just as much about the way I phrase my questions as it is about the AI's understanding.

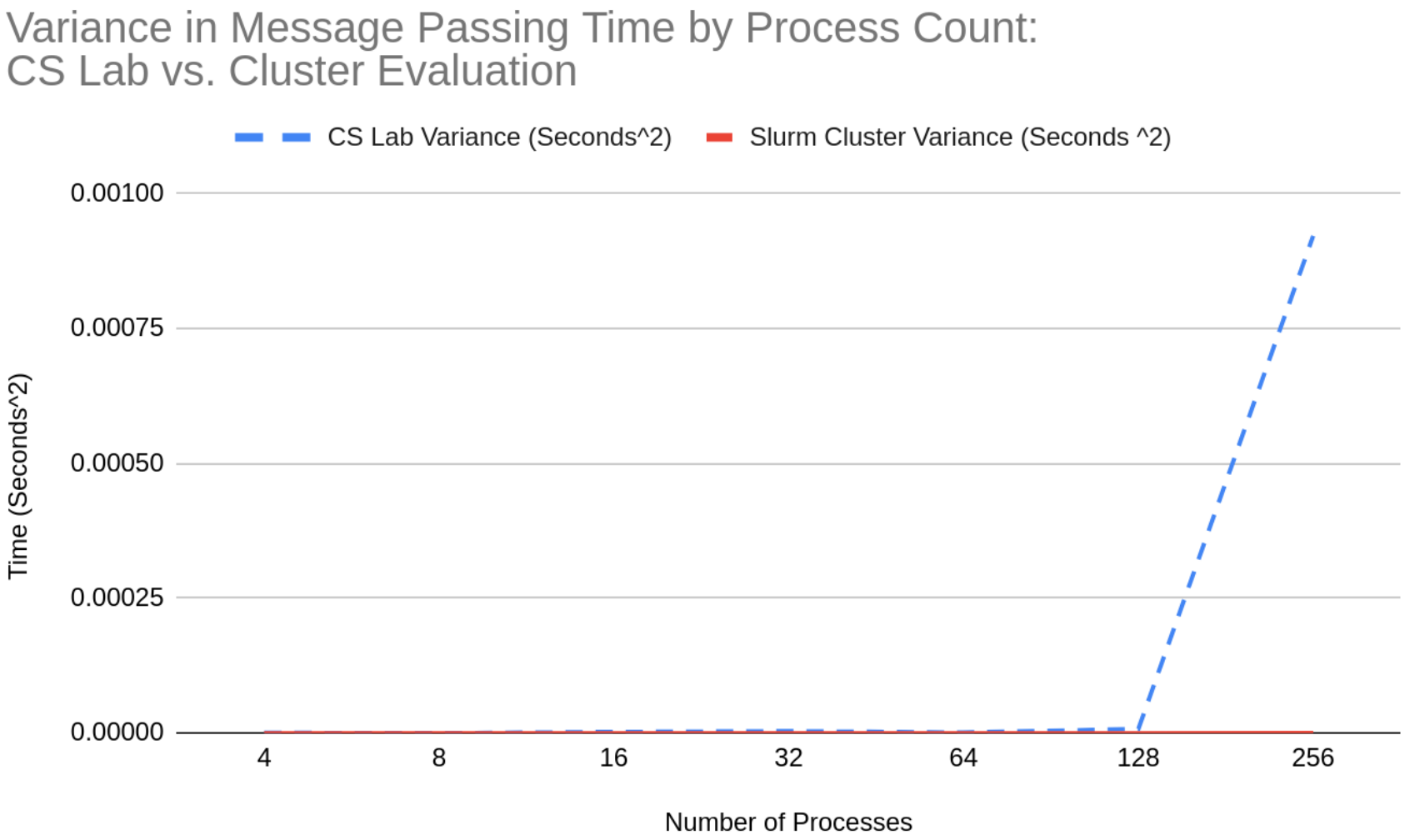
The prompts I used:

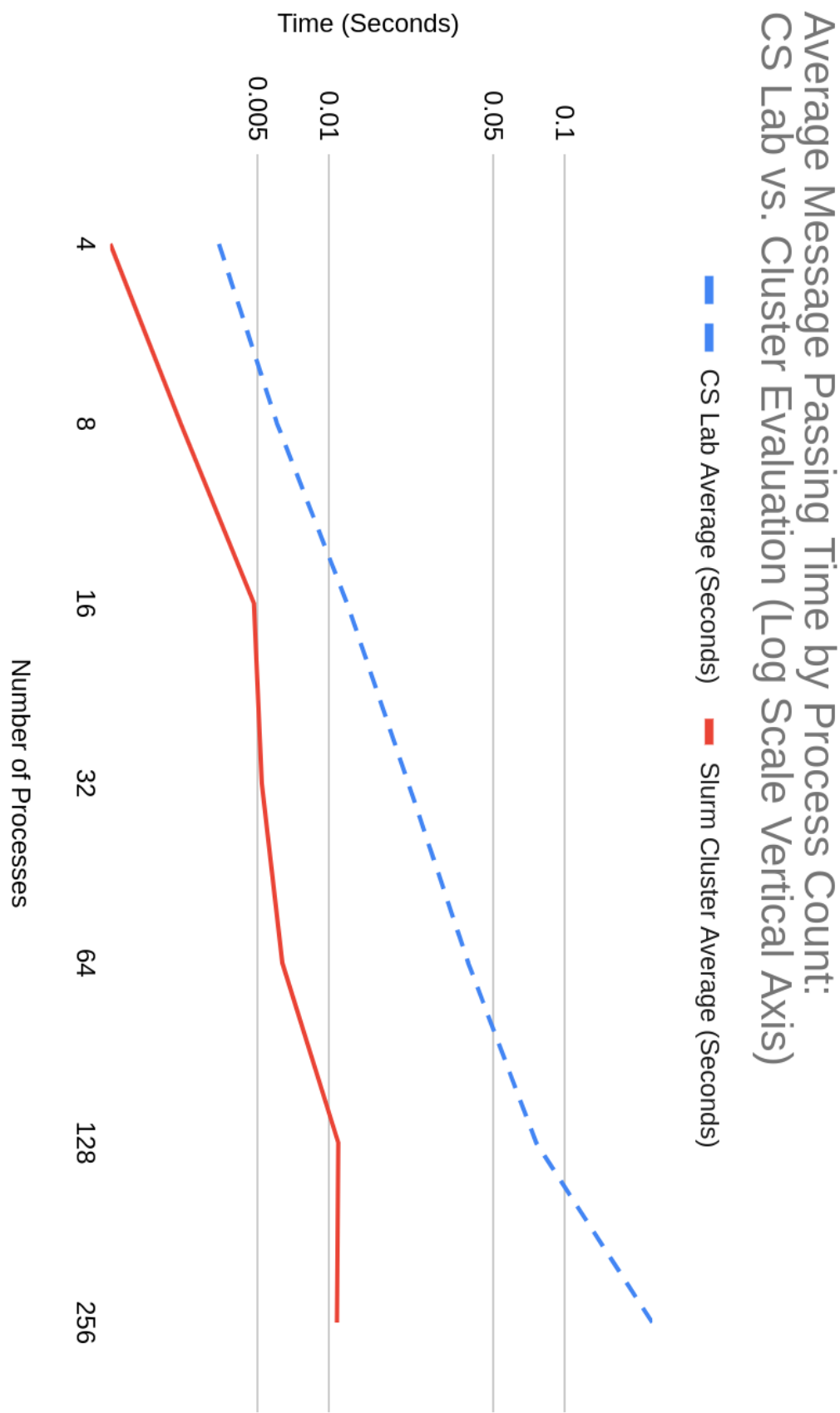
In simple words can you explain array passing and message passing are different
and explain the code
im working from 4 to 256 provesses, do you think where do i need to change?
so the only thing you changed it from char newMessage[10]; to char newMessage[5];
What should I name the file
Did I followed the instructions clearly
Is this top comment good enough could you help me make it more organized and grammarly correct
Can you give me comments
1. is this good

yl77@gold26:~/374/proj02$ mpicc ring.c -o ring -Wall -ansi -pedantic -std=c99
yl77@gold26:~/374/proj02$ mpirun -np 16 -machinefile hosts ./ring
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
time: 0.013725 secs
Why its showing error here make does not work
what should this spred sheet look like? Is there an example on how to do it on google sheets?
does this data look right to you?
did this analysis meet all the requirements listed?
What command to use to copy from slurm to my local
Why from 128 to 256 the time tripled
fix these two top comments
Do you think I did the hypothesis correctly
Where can I state it better
Can you extend this topic on comparing last week data, how is it related
fix grammar and basic logic and keep using simple words, if there are any logic error please tell me
did this analysis meet the requirement? out of 100 what would you give
explain why at 256 the mean and variance suddenly went really high what sthe hyposthesis? and what data can you provdie t oprove it

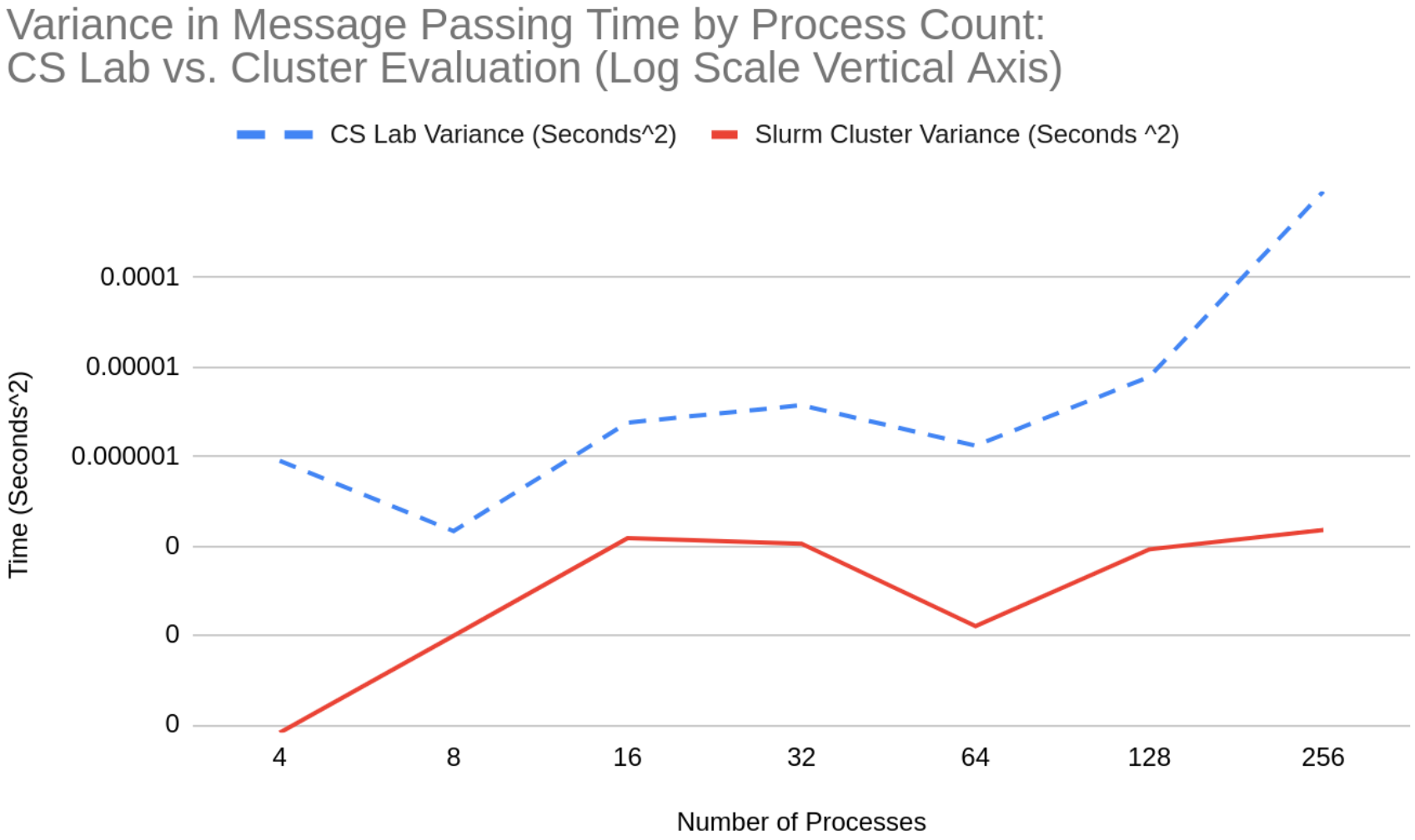

Average Message Passing Time by Process Count:
CS Lab vs. Cluster Evaluation
CS Lab Average (Seconds)
Slurm Cluster Average (Seconds)
Time (Seconds)
0.25
0.20
0.15
0.10
0.05
0.00
4
8
16
32
64
128
256
Number of Processes

Yuese Li


Variance in Message Passing Time by Process Count:
CS Lab vs. Cluster Evaluation
CS Lab Variance (Seconds^2)
Slurm Cluster Variance (Seconds ^2)
0.00100
0.00075
0.00050
0.00025
0.00000
Time (Seconds^2)
4
8
16
32
64
128
256
Number of Processes


Average Message Passing Time by Process Count:
CS Lab vs. Cluster Evaluation (Log Scale Vertical Axis)
CS Lab Average (Seconds)
Slurm Cluster Average (Seconds)
Time (Seconds)
0.1
0.05
0.01
0.005
4
8
16
32
64
128
256
Number of Processes

Yuese Li


Variance in Message Passing Time by Process Count:
CS Lab vs. Cluster Evaluation (Log Scale Vertical Axis)
CS Lab Variance (Seconds^2)
Slurm Cluster Variance (Seconds ^2)
0.0001
0.00001
0.000001
0
0
0
Time (Seconds^2)
4
8
16
32
64
128
256
Number of Processes

CS Lab Data

Number of Processes	Trial 1	Trial 2	Trial 3	CS Lab Average (Seconds)	CS Lab Variance (Seconds^2)
4	0.002601	0.004474	0.003281	0.003452	0.000000898963
8	0.005739	0.006042	0.0065	0.006093666667	0.0000001467823333
16	0.011388	0.010993	0.013849	0.01207666667	0.000002394880333
32	0.02405	0.021574	0.020233	0.02195233333	0.000003749724333
64	0.038878	0.038693	0.040771	0.03944733333	0.000001322626333
128	0.078345	0.077958	0.073328	0.07654366667	0.000007792826333
256	0.212686	0.229537	0.271645	0.237956	0.000922200591

Cluster Data

Number of Processes	Trial 1	Trial 2	Trial 3	Slurm Cluster Average (Seconds)	Slurm Cluster Variance (Seconds ^2)
4	0.00119	0.001223	0.001166	0.001193	0.000000000819
8	0.002302	0.002493	0.002348	0.002381	0.000000009937
16	0.00449	0.005183	0.004918	0.004863666667	0.0000001222763333
32	0.00561	0.004977	0.005163	0.00525	0.000000105849
64	0.006297	0.006522	0.00642	0.006413	0.000000012693
128	0.00811	0.008716	0.008428	0.008418	0.000000091884
256	0.010609	0.010851	0.01137	0.01094333333	0.0000001511743333