PA4: Threading and Synchronization

Brandon DeAlmeida

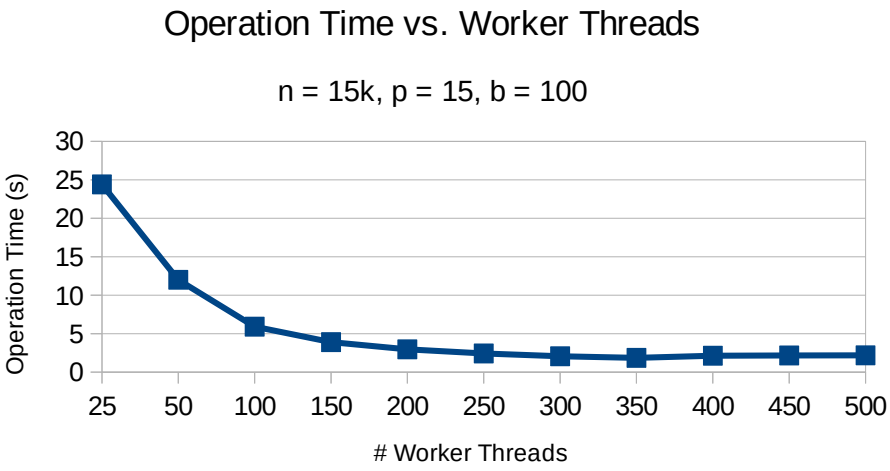
CSCE 313-508

Instructor: Tanzir Ahmed

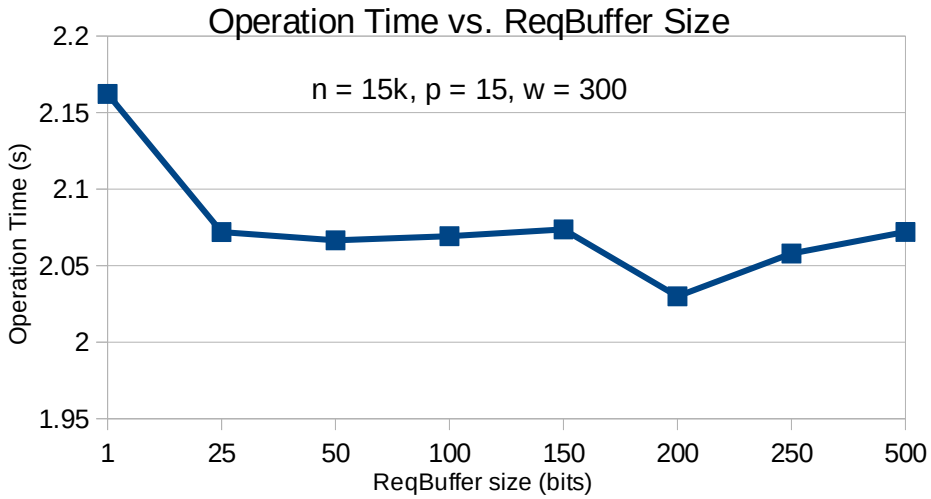
TA: Feras Khemakhem

Data:

Data Requests:



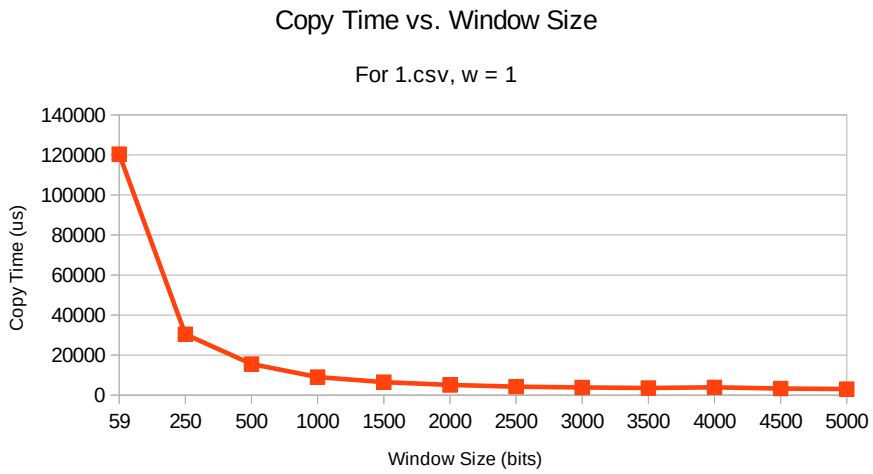
In increasing the number of worker threads present we see a somewhat exponential decrease in the time it takes to copy all data points until about 250 threads; from there the time remains relatively constant. The initial curve is likely due to the fact that we have more ‘hands’ moving data from the server through their own individual channels leading to a much faster end time. This curve also likely flattens out due to the fact that too many worker threads either leads to an always empty request buffer (leaving a lot of idle time for these workers) or a server that’s unable to keep up with that many channels at once.



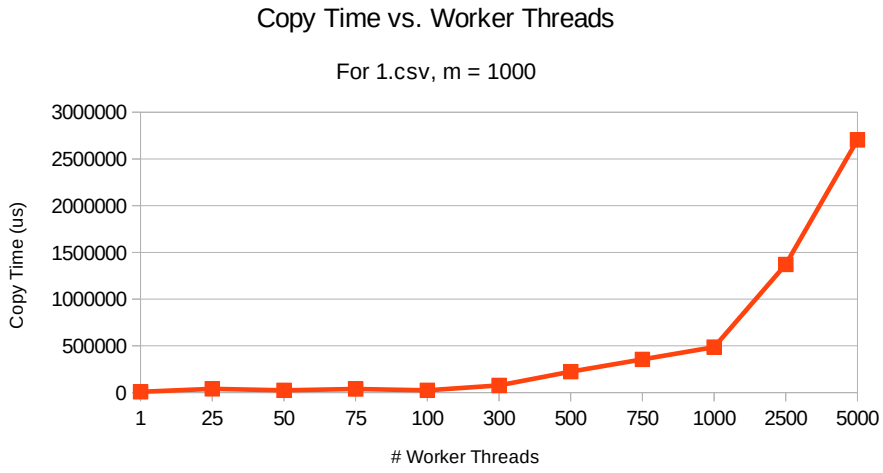
A relatively flat curve was found while increasing just the request buffer size, the only large deviation can be seen for a very little size (<25 requests). This shows that the number of worker threads tend to be the larger factor in the operation time of copying these data points. A likely reason for this

trend is that since the bounded request buffer is constantly being emptied and filled by both the worker and patient threads it likely doesn't generally stay full for long enough to slow down performance.

File Requests:



Once again here we see a somewhat exponential trend between the test cases, indicating that the window, or max, buffer size plays a large time in the time it takes to copy a file. This is likely attributed to the fact that a smaller window means more requests to the server: these requests are the lengthiest parts of our copy adding more of them can significantly drive up the copy time.



Here we see a relatively flat trend for the worker treads until about 300 when the trend shifts to a line. This implies that while the number of worker threads doesn't have a big impact on the copy time itself, at a larger number of threads there are worker threads that never copy any data to make the time faster and actually increase our overall time since we still have to close them out (hence the linear increase in time).