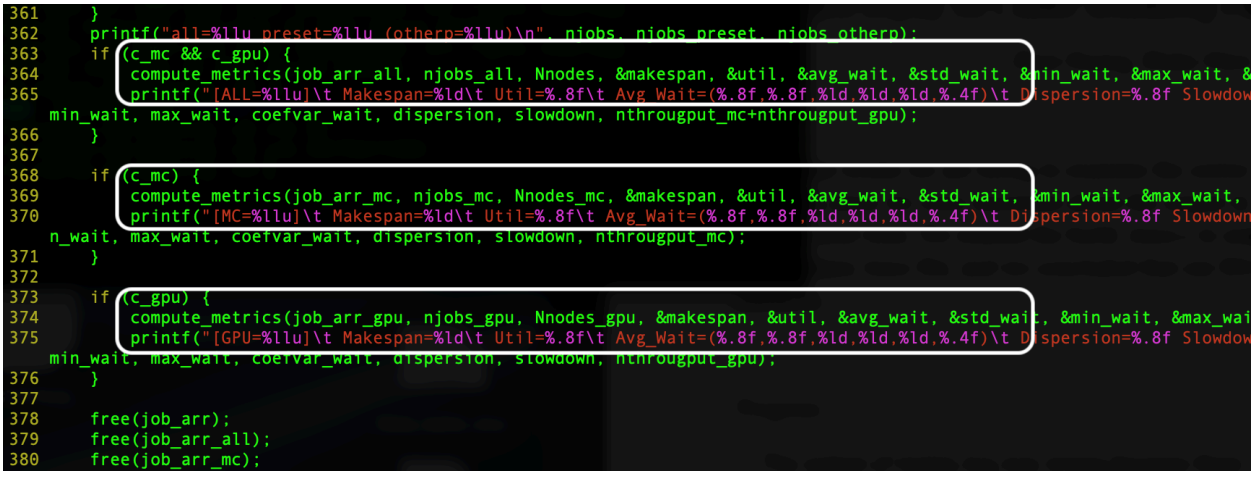
RM-Replay for Cluster Scheduling Project Report 3/15-3/27

Zhen Huang, Blake Ehrenbeck

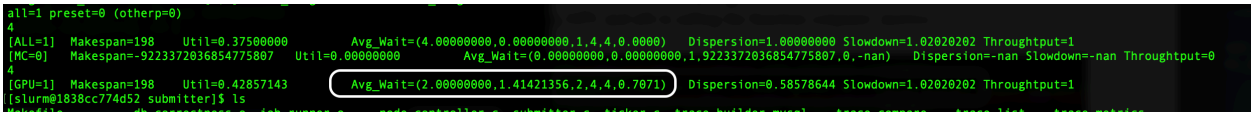
These two weeks we are trying to examine the correctness of the RM-Replay that we installed, since we were uncertain about the output of metric.log in regard to utility and average wait time as well as work on running some Theta jobs in RM-Replay. We studied the trace_metric.c to understand the output and compared to the binary trace file, and we concluded that our RM-Replay should be working correctly. We are also nearly completion of importing the data from Theta to our new database.

Progress:

1. The output of metric.log was wrong because there is a bug in the trace_metrics.c, we found that in the *compute_metrics* method starts from line 362 each categories (mc, GPU and all) is calling the function *computer_metrics.*



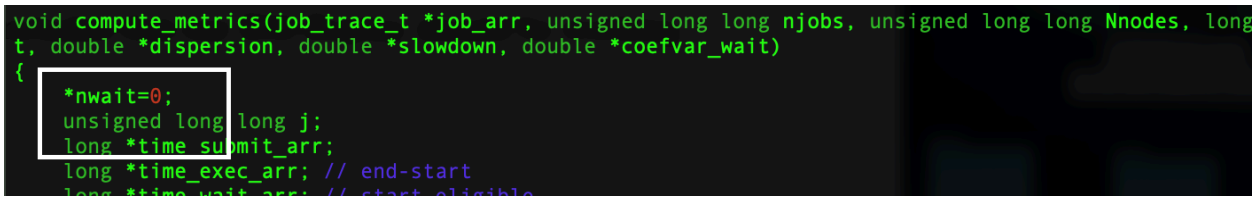
However, the function *compute_metrics* doesn't reset the variable *nwait* to 0*,* so the output in metrics was off by one when the metrics output get printed out, and we get the wrong number of waits (should be 1 but 2):

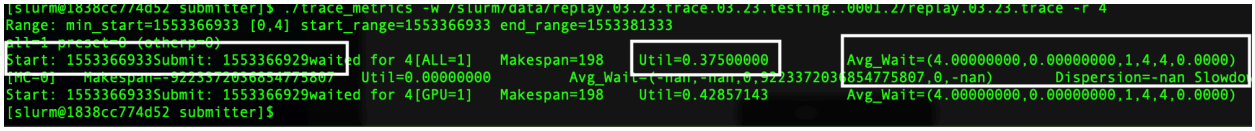


The format of Avg_Wait(…) is as follows:

(Average Wait, Standard Deviation Wait, Number of Waits, Min Wait, Max Wait, Coef Var Wait)

After we change the code by reseting *nwait* to 0 every time it gets called, we get the correct output:



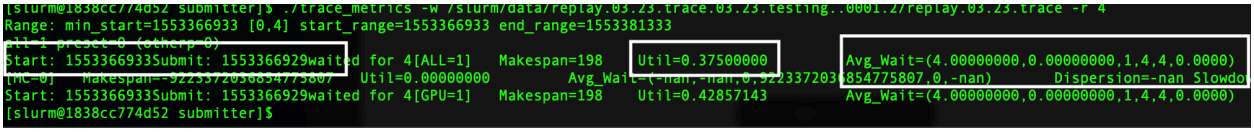


(MC output here is nonsensical since the job we submitted was using GPU nodes)

The wait time seen above we found was actually in *seconds* not minutes as we previously thought. We figured this out by printing the start and submit timestamps and converting them from a UNIX timestamp to a more readable date-time format. This few second long wait checks out with logs from Slurm.

2. By studying the configuration of Jarvis provided by Yao, we were able to calculate the theoretical utility, and we check with the output of RM-Replay which shown to be correct:

Jarvis has 7 node with GPU and 1 without. So total number of nodes is 8. Our testing job use 3 nodes to run a MPI job, so the overall utilization is 3/8 = 0.375. GPU utilization is also correct: We used 3/7 GPU enabled nodes, which is about 42% utilization.



We are now continuing to work on feeding Theta logs into RM-Replay. We have nearly completed recreating the slurm_acct_db. Both the jarvis_job_table and resv_table schemas are created.

Also we have created the CSV file mapping the Cobalt logs from Theta over to something we can import into the slurm_acct_db we built.

We are now working on importing the CSV file into the database we built. We've just run into a minor issue with the datatypes not lining up completely, but it just requires a minor fix.