

EE/CprE/SE 492 STATUS REPORT 5

Start Date – End Date: 11/1/2024 - 11/14/2024

Group number: 9

Project title: Multicore Operational Analysis Tooling

Client &/Advisor: Steve Vanderleest/Joe Zambreno

Team Members/Role: Alexander Bashara – Embedded Engineer, Joseph Dicklin – Hardware Design Engineer, Hankel Haldin – OS/Tooling Engineer, Anthony Manschula – Project Coordinator/Engineer

Summary of progress:

The team has implemented the core features of our design. We are currently focused on making it easier to collect data from our test framework. These efforts go hand in hand with the user needs reported by our client in our weekly meetings.

Past report accomplishments:

The team focused on refining the initial prototype we presented to our client during this period. Our work in the previous period demonstrated that our design could collect relevant performance data. Our client expressed interest in a larger pool of test data to facilitate statistical analysis of the worst-case execution time for processes. To this end, the team developed an automated test framework that would run a preconfigured number of experiments on our target platform. The data generated by these experiments is then processed by a Python script to determine several statistical properties (e.g., max runtime, standard deviation, etc.). The framework also provides us with plots of execution time of each test case on multiple cores which facilitates the communication of our results with multiple audiences.

Milestones reached:

- Established communication between DomUs from Dom0
- Established a connection between host and target control programs
- Passed config data between host and target programs
- Exported experiment data from target platform
- Wrote Python script to extract statistical data from experiment data
- Automated creation/deletion of DomU's with XL toolchain commands and shell scripts

Pending issues:

- Create open-source GitHub repository
- Enhance automated YAML parser for experimental data
- Create documentation for automated test framework (e.g, operational manual)

Individual contributions:

<u>NAME</u>	<u>Individual Contributions</u>	<u>Hours this week</u>	<u>HOURS cumulative</u>
Alexander Bashara	Created YAML Parser and collecting data	13	100
Joseph Dicklin	Worked on final poster and presentation	11	95
Hankel Haldin	Developed documentation for project repository and user interface	12	95
Anthony Manschula	Work on stress generation frontend framework and test results collection	14	100

Plans for the upcoming reporting period:

- Develop a fully automated YAML parser to collect experimental data
- Create a GitHub repository with our work and documentation
- Create documentation in the form of an operational manual, poster and demo video

Project Work:

The following figures detail the output of our design. They present experimental data generated by our target platform and parsed by our test framework. Figures 1 and 2 present the results of our memory stressor. Figure 1 shows the statistical data we collected from our experiment and figure 2 shows how the execution time across multiple cores. Figures 3 and 4 do the same for our cache stressor.

	Base	BW 1 Core	BW 2 Cores	BW 3 Cores
WCET	7.277579	9.031392	10.686565	12.486533
Average	7.21863918	7.9937635999999985	9.764910160000001	12.00491878
Standard Deviation	0.0006476784631475986	0.04598175301888004	0.03949367584401436	0.040339238978971584

Figure 1: Memory bandwidth test results

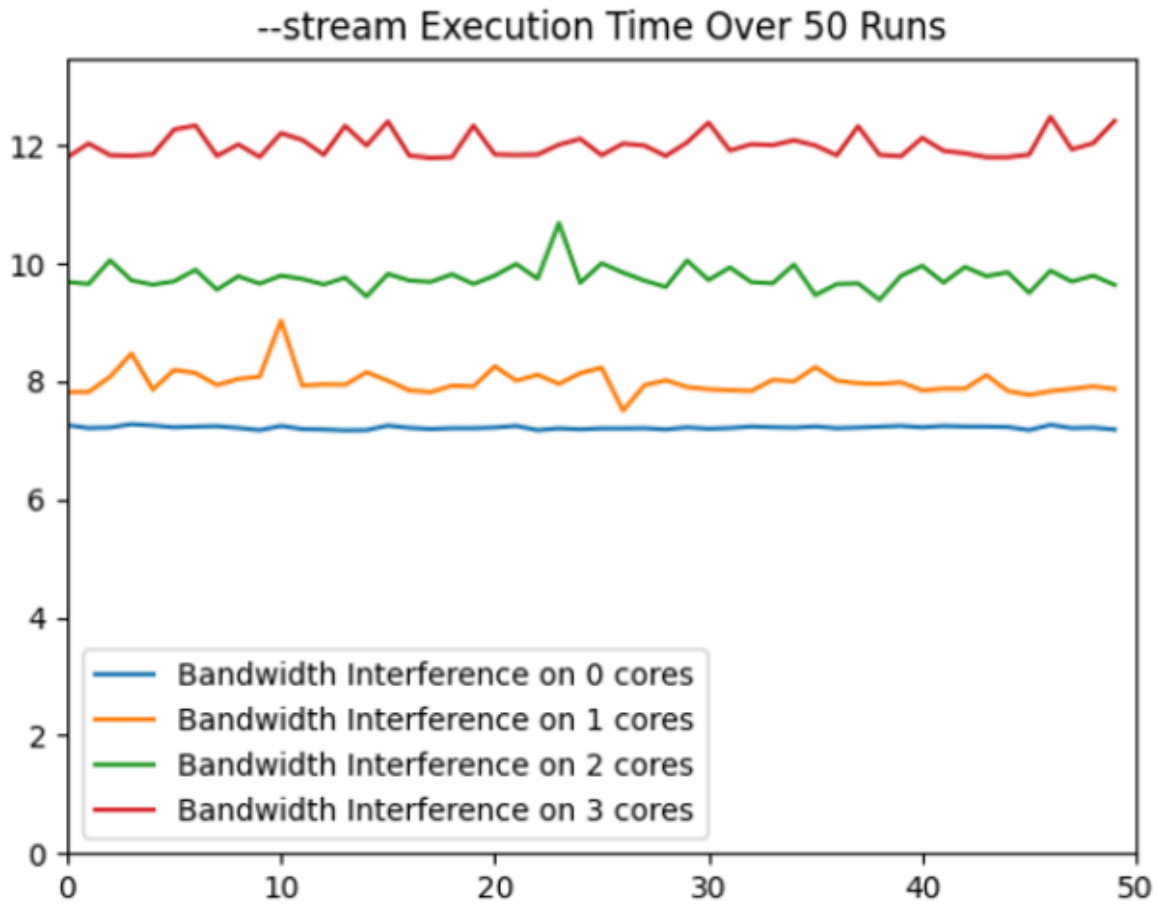


Figure 2: Memory bandwidth results across several cores

	Base Matrix Add	Cache Stress 1 Core	Cache Stress 2 Cores	Cache Stress 3 Cores
WCET	3.158273	3.208304	3.263862	3.349793
Average	2.9783544399999995	2.9777689199999999	3.24148532	3.2605565800000007
Standard Deviation	0.011232985978926404	0.015176578266473607	0.00004963228253759988	0.0038600725519636024

Figure 3: Cache results

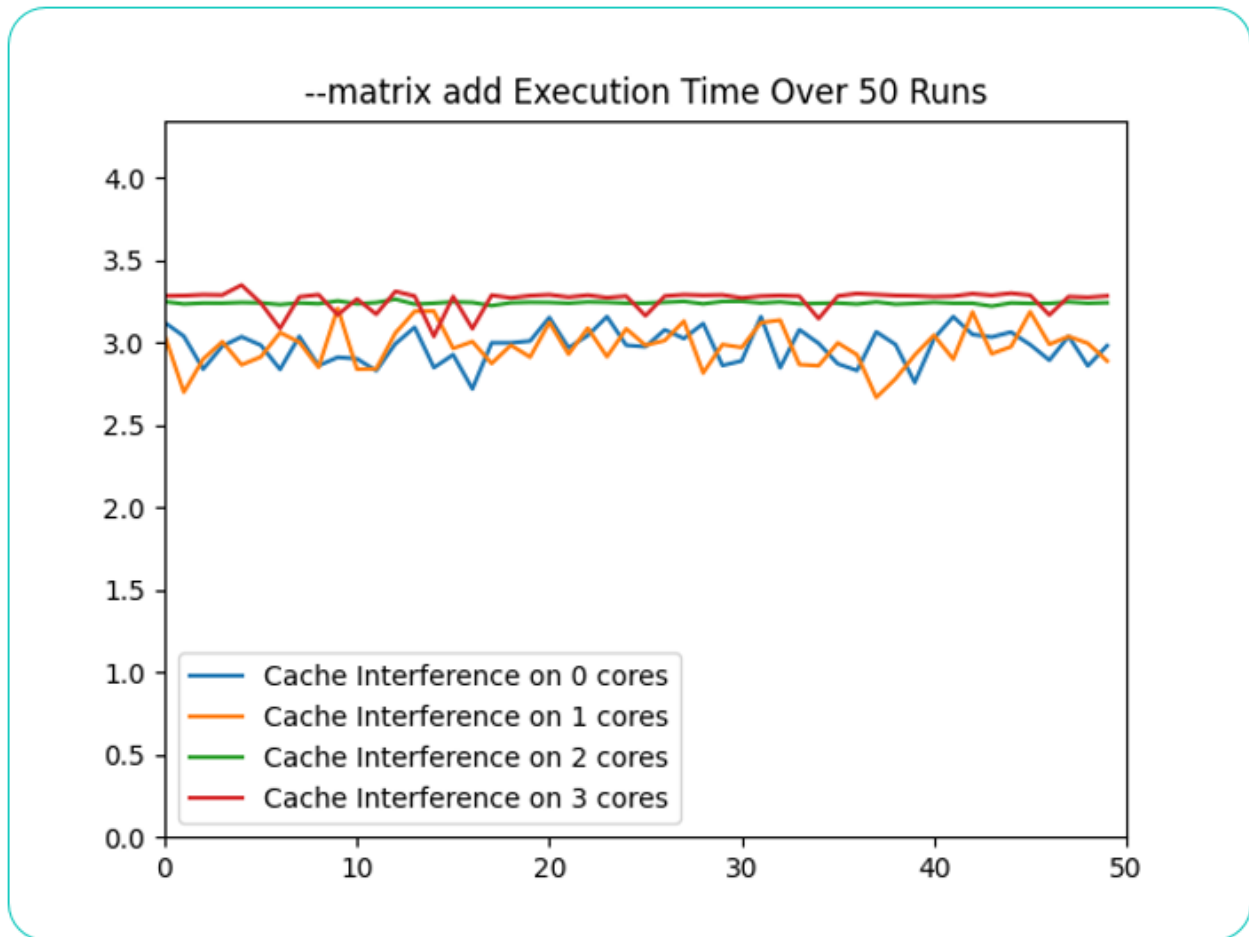


Figure 4: Cache results compare across several cores

Summary of weekly advisor meeting:

Our industry client is happy with the progress we have made in the last reporting period. Our meetings with Boeing during this reporting period focused on addressing various user needs, like the ability to pass preconfigured test scripts to our framework. This feedback guides much of the team's efforts to further develop our design.