



Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

High-Performance Computing Lab for CSE

2024

Student: Noah Gigler

Discussed with: Felicia Scharitzer, Luis Wirth

Solution for Project 1a

Due date: 11 March 2024, 23:59

HPC Lab for CSE 2024 — Submission Instructions
(Please, notice that following instructions are mandatory:
submissions that don't comply with, won't be considered)

- Assignments must be submitted to Moodle (i.e. in electronic format).
- Provide both executable package and sources (e.g. C/C++ files, Matlab). If you are using libraries, please add them in the file. Sources must be organized in directories called:
Project_number_lastname_firstname
and the file must be called:
project_number_lastname_firstname.zip
project_number_lastname_firstname.pdf
- The TAs will grade your project by reviewing your project write-up, and looking at the implementation you attempted, and benchmarking your code's performance.
- You are allowed to discuss all questions with anyone you like; however: (i) your submission must list anyone you discussed problems with and (ii) you must write up your submission independently.

1. Euler warm-up [10 points]

1. The module system is a tool used to manage software environments on a Euler. It allows us to configure their environment by dynamically loading or unloading software modules. These modules adjust system variables to ensure that the necessary binaries and libraries are accessible. You use it by loading specific software versions with module load and unloading them with module unload when done.

2. Slurm is a tool used in big computer clusters to help manage who gets to use the computers and when. It schedules tasks and makes sure everything runs smoothly by allocating resources like processors and memory. It's like the traffic controller for a cluster of computers.

3. see hostname.cpp

4. see slurm_job_one.sh

5. see slurm_job_two.sh

2. Performance characteristics [50 points]

2.1. Peak performance

Source: https://scicomp.ethz.ch/wiki/Euler#Euler_VII_.E2.80.94_phase_1

Table 1: Euler VII Phase 1 and Phase 2 Specifications

Phase	Compute Nodes	CPUs per Node	CPU	Clock Speed (GHz)
Phase 1	292	2	AMD EPYC 7H12	2.6
Phase 2	248	2	AMD EPYC 7763	2.45

$$n_{\text{super}} = \frac{1}{TP} = 2$$

$$n_{\text{FMA}} = 2$$

$$n_{\text{SMID}} = 4$$

Values are the same for both Euler VII Phase 1 and 2.

Source for FMA, TP and : <https://uops.info/table.html>

Source for the SIMD values: "Software Optimization Guide for AMD EPYC™ 7002 Processors" and "Software Optimization Guide for AMD EPYC™ 7003 Processors"

$$P_{\text{core}} = n_{\text{super}} \cdot n_{\text{FMA}} \cdot n_{\text{SMID}} \cdot f$$

$$P_{\text{CPU}} = P_{\text{core}} \cdot \#\text{Cores}$$

$$P_{\text{node}} = P_{\text{core}} \cdot \#\text{CPUs}$$

$$P_{\text{EulerVII}} = P_{\text{node}} \cdot \#\text{Nodes}$$

Table 2: Peak Performance Comparison

Metric	Phase 1	Phase 2
P_{core}	41.6 GFLOP/s	39.2 GFLOP/s
P_{CPU}	2.66 TFLOP/s	2.51 TFLOP/s
P_{node}	5.32 TFLOP/s	5.02 TFLOP/s
P_{EulerVII}	1.55 PFLOP/s	1.24 PFLOP/s

2.2. Memory Hierarchies

2.2.1. Cache and main memory size

Table 3: Cache and Main Memory Sizes

Phase	L1 (KB)	L2 (KB)	L3 (MB)	Main Memory (GB)
Phase 1	32	512	16	256
Phase 2	32	512	32	256

The only difference between the cache sizes of Phase 1 and Phase 2 is the L3 cache size. Phase 2 has twice the L3 cache size of Phase 1.

2.3. Bandwidth: STREAM benchmark

As we can see, the bandwidth of the AMD EPYC 7763 is higher than the AMD EPYC 7H12 for all functions. The Copy function generally has the highest bandwidth. The other functions have similar bandwidths. We can assume that the bandwidth of the AMD EPYC 7763 is 25Gb/s and the bandwidth of the AMD EPYC 7H12 is 20Gb/s.

Table 4: Memory Bandwidth Comparison

Function	AMD EPYC 7H12	AMD EPYC 7763
Copy	30804.4 MB/s	34873.5 MB/s
Scale	19134.3 MB/s	24681.9 MB/s
Add	21820.0 MB/s	25473.4 MB/s
Triad	21891.7 MB/s	25710.9 MB/s

2.4. Performance model: A simple roofline model

The formula for the naive roofline model is:

$$P = \begin{cases} \pi & \pi < \beta \cdot I \\ \beta \cdot I & \text{else} \end{cases}$$

Where π is the peak performance and I is the operational intensity and β is the peak bandwidth. Here we are looking at the performance of a single core so we can use the peak performance we calculated in task 2.1. The peak bandwidth was calculated in task 2.3.

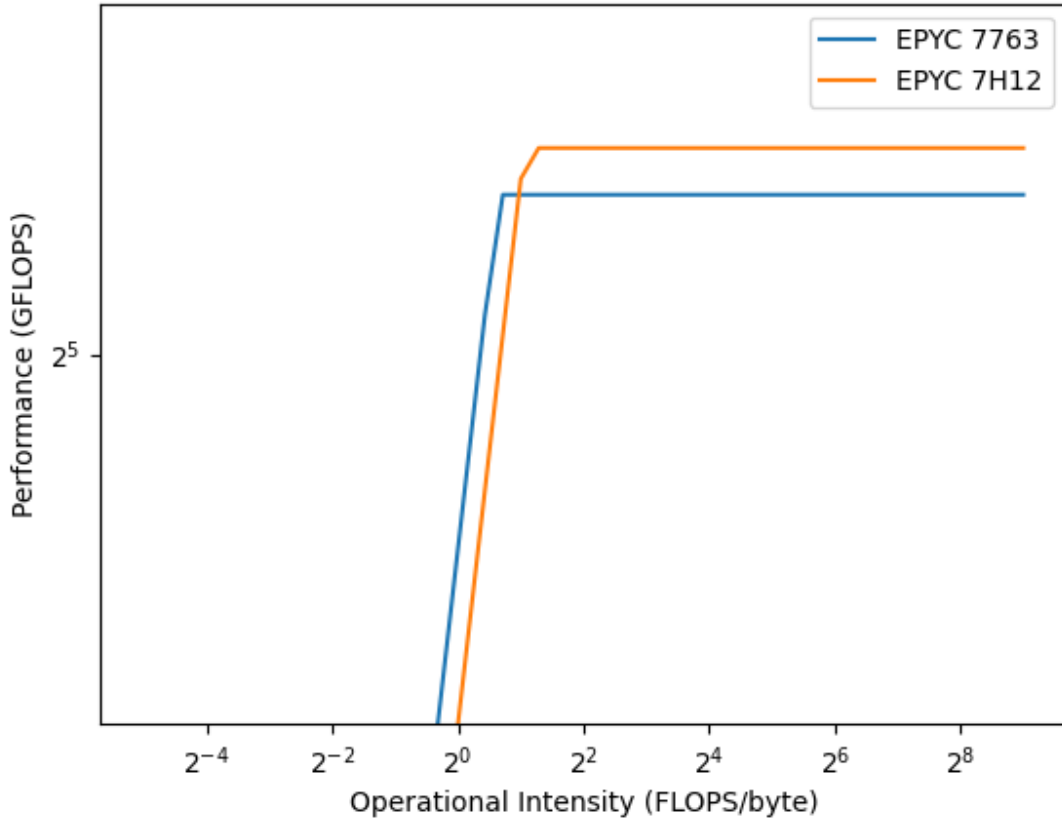


Figure 1: Roofline Model

As we can see from the formula, the performance is limited by the bandwidth when $\pi < \beta \cdot I$ and by the peak performance when $\pi \geq \beta \cdot I$. In the graph this can be seen visually. As long as the performance is still increasing we are limited by the bandwidth and once the performance plateaus we are limited by the peak performance.