Lab 05

High Performance Scientific Computing

Varun Hoskere

Alpine Hardware Description:

Alpine is the University of Colorado Boulder Research Computing's third-generation high performance computing (HPC) cluster.

The Alpine cluster is made up of different types of nodes outlined below:

- **CPU nodes**: 376 AMD Milan compute nodes
- **GPU nodes**:
  - 8 GPU-enabled (3x AMD MI100) atop AMD Milan CPU
  - 12 GPU-enabled (3x NVIDIA A100) atop AMD Milan CPU
- **High-memory nodes**: 24 AMD Milan nodes with 1TB of memory

These nodes are grouped into partitions based on hardware.

A brief description of the available partitions is given in the below table:

Name	**Description**	**# of nodes**	**# cores / node**	**RAM / core (GB)**
amilan	AMD Milan	376	32/48/64	3.75
ami100	GPU-enabled (3x AMD MI100)	8	64	3.75
aa100	GPU-enabled (3x NVIDIA A100)	12	64	3.75
amem	High-memory	24	48/64	16*16
csu	Nodes contributed by CSU	77	32/48	3.75

Some of the available partitions come equipped with GPU nodes. All partitions can be accessed using SLURM directives '--partition=' followed by the name of the partition.

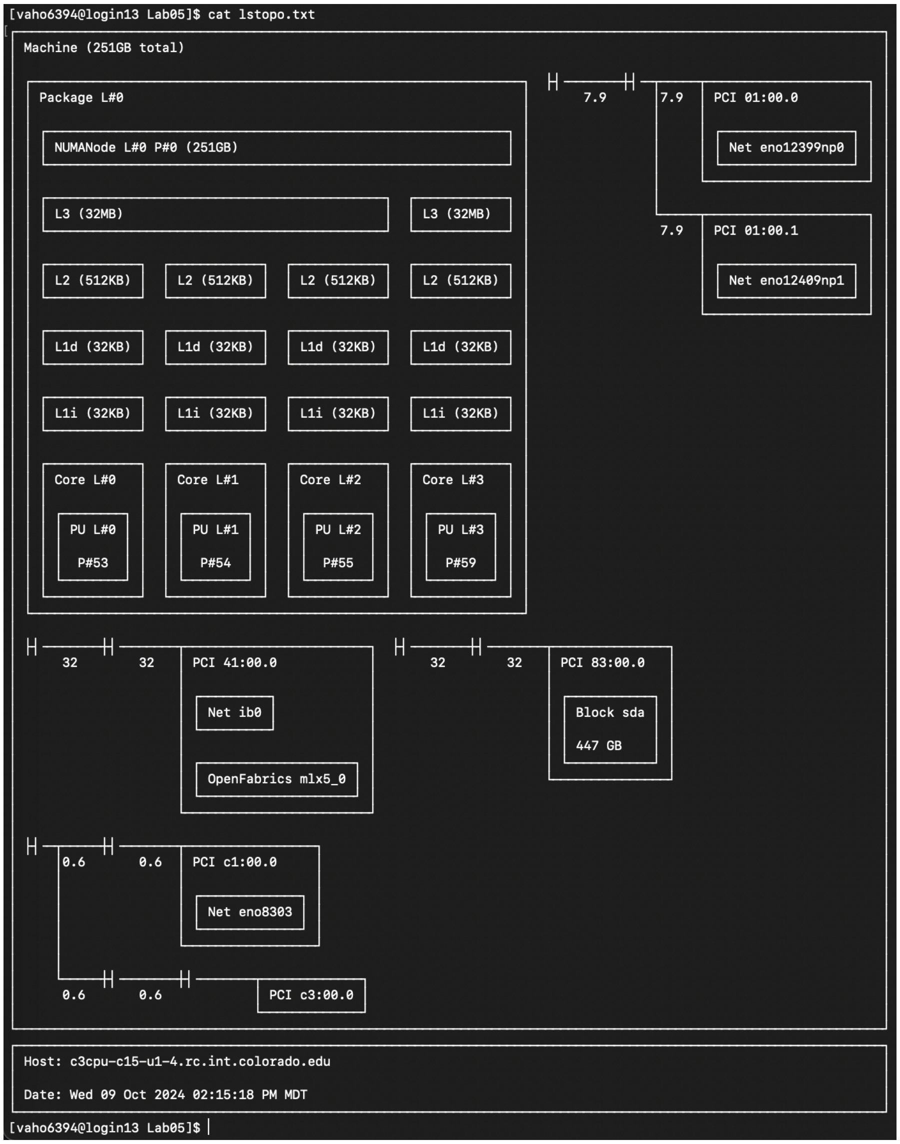
There is a special purpose partition exist – called the 'atesting' partition. It provides access to limited resources for the purpose of verifying workflows and MPI jobs. Users can request up to 2 CPU nodes for a maximum runtime of 3 hours and 16 CPUs. These also, come with GPU nodes, if requested.

There are three types of nodes available on Alpine - login nodes, compile nodes and compute nodes. To access compile nodes, users should run the command 'acompile' from the login node terminal and to access compute nodes, users should run the command 'sinteractive'. These commands can also be followed up optional tags which specify further configurations of the hardware that the user requests for.

Compute node:

Running lstop --of asci, we get information of the node that has been assigned to us.

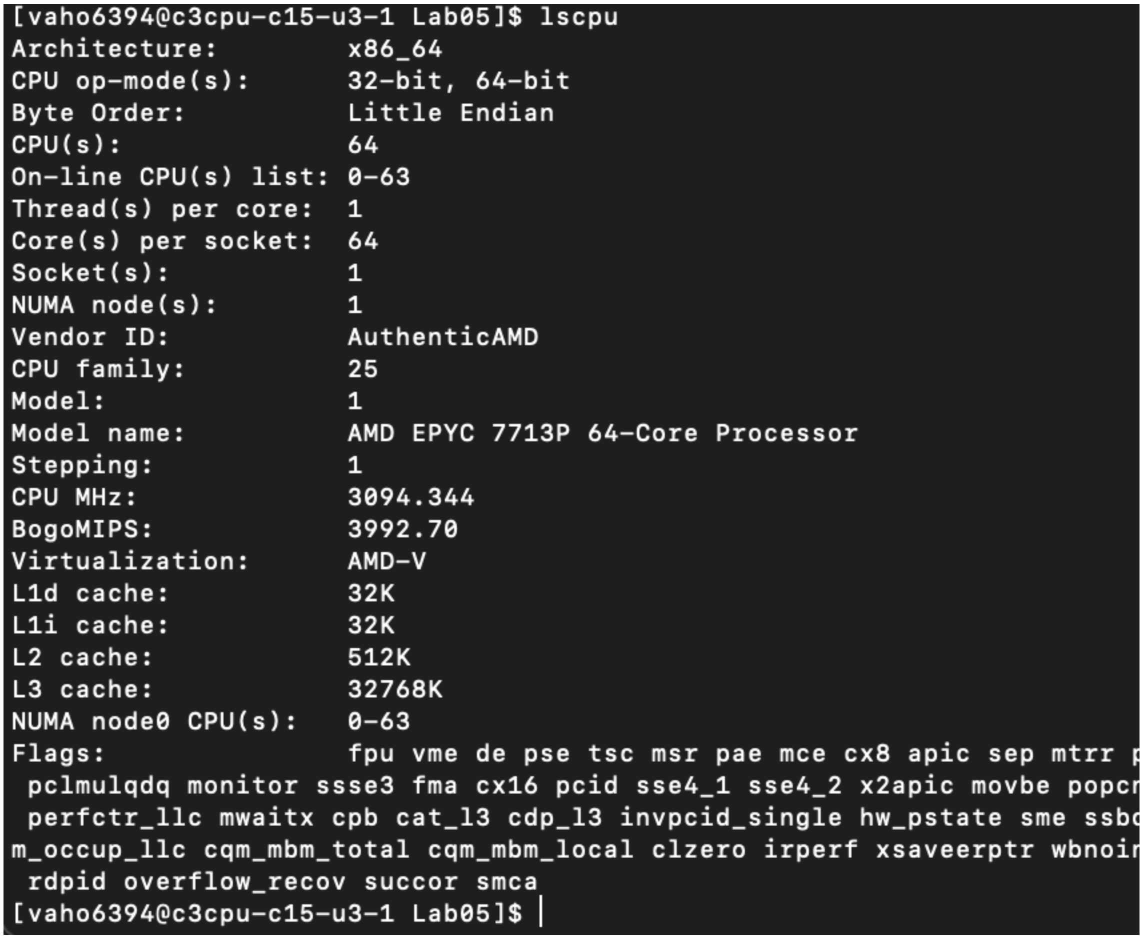
Depending on the type of partition that was requested, the number of processing units varies. Since I requested for 4 tasks, I was given 4 processing units. Each is coupled with 3 levels of cache - an L1d cache of size 32 KB, an L1i cache of size 32KB, an L2 cache of size 512 KB and an L3 cache of size 32 MB. Each PCI device is a peripheral device that is connected to the main package. There are 4 networking PCI devices, one block of storage and another unlabeled device.



```
[vaho6394@login13 Lab05]$ cat lstopo.txt
┌─────────────────────────────────────────────────────────────────────────────────────────────────────────────────┐
│ Machine (251GB total)                                                                                           │
│                                                                                                                 │
│ ┌────────────────────────────────────────────────────────────────┐   ├┤────────├┤─┬─────┌─────────────────────┐ │
│ │ Package L#0                                                    │       7.9      │7.9  │ PCI 01:00.0         │ │
│ │                                                                │                │     │                     │ │
│ │ ┌────────────────────────────────────────────────────────────┐ │                │     │ ┌─────────────────┐ │ │
│ │ │ NUMANode L#0 P#0 (251GB)                                   │ │                │     │ │ Net eno12399np0 │ │ │
│ │ └────────────────────────────────────────────────────────────┘ │                │     │ └─────────────────┘ │ │
│ │                                                                │                │     └─────────────────────┘ │
│ │ ┌────────────────────────────────────────────┐  ┌────────────┐ │                │                             │
│ │ │ L3 (32MB)                                  │  │ L3 (32MB)  │ │                └─────┌─────────────────────┐ │
│ │ └────────────────────────────────────────────┘  └────────────┘ │                 7.9  │ PCI 01:00.1         │ │
│ │                                                                │                      │                     │ │
│ │ ┌────────────┐  ┌────────────┐  ┌────────────┐  ┌────────────┐ │                      │ ┌─────────────────┐ │ │
│ │ │ L2 (512KB) │  │ L2 (512KB) │  │ L2 (512KB) │  │ L2 (512KB) │ │                      │ │ Net eno12409np1 │ │ │
│ │ └────────────┘  └────────────┘  └────────────┘  └────────────┘ │                      │ └─────────────────┘ │ │
│ │                                                                │                      └─────────────────────┘ │
│ │ ┌────────────┐  ┌────────────┐  ┌────────────┐  ┌────────────┐ │                                              │
│ │ │ L1d (32KB) │  │ L1d (32KB) │  │ L1d (32KB) │  │ L1d (32KB) │ │                                              │
│ │ └────────────┘  └────────────┘  └────────────┘  └────────────┘ │                                              │
│ │                                                                │                                              │
│ │ ┌────────────┐  ┌────────────┐  ┌────────────┐  ┌────────────┐ │                                              │
│ │ │ L1i (32KB) │  │ L1i (32KB) │  │ L1i (32KB) │  │ L1i (32KB) │ │                                              │
│ │ └────────────┘  └────────────┘  └────────────┘  └────────────┘ │                                              │
│ │                                                                │                                              │
│ │ ┌────────────┐  ┌────────────┐  ┌────────────┐  ┌────────────┐ │                                              │
│ │ │ Core L#0   │  │ Core L#1   │  │ Core L#2   │  │ Core L#3   │ │                                              │
│ │ │            │  │            │  │            │  │            │ │                                              │
│ │ │ ┌────────┐ │  │ ┌────────┐ │  │ ┌────────┐ │  │ ┌────────┐ │ │                                              │
│ │ │ │ PU L#0 │ │  │ │ PU L#1 │ │  │ │ PU L#2 │ │  │ │ PU L#3 │ │ │                                              │
│ │ │ │        │ │  │ │        │ │  │ │        │ │  │ │        │ │ │                                              │
│ │ │ │  P#53  │ │  │ │  P#54  │ │  │ │  P#55  │ │  │ │  P#59  │ │ │                                              │
│ │ │ └────────┘ │  │ └────────┘ │  │ └────────┘ │  │ └────────┘ │ │                                              │
│ │ └────────────┘  └────────────┘  └────────────┘  └────────────┘ │                                              │
│ └────────────────────────────────────────────────────────────────┘                                              │
│                                                                                                                 │
│ ├┤────────├┤────────┌─────────────────────────┐ ├┤────────├┤────────┌───────────────┐                           │
│      32        32   │ PCI 41:00.0             │      32        32   │ PCI 83:00.0   │                           │
│                     │                         │                     │               │                           │
│                     │ ┌─────────┐             │                     │ ┌───────────┐ │                           │
│                     │ │ Net ib0 │             │                     │ │ Block sda │ │                           │
│                     │ └─────────┘             │                     │ │           │ │                           │
│                     │                         │                     │ │ 447 GB    │ │                           │
│                     │ ┌────────────────────┐  │                     │ └───────────┘ │                           │
│                     │ │ OpenFabrics mlx5_0 │  │                     └───────────────┘                           │
│                     │ └────────────────────┘  │                                                                 │
│                     └─────────────────────────┘                                                                 │
│                                                                                                                 │
│ ├┤──┬─────├┤────────┌─────────────────┐                                                                         │
│     │0.6       0.6  │ PCI c1:00.0     │                                                                         │
│     │               │                 │                                                                         │
│     │               │ ┌─────────────┐ │                                                                         │
│     │               │ │ Net eno8303 │ │                                                                         │
│     │               │ └─────────────┘ │                                                                         │
│     │               └─────────────────┘                                                                         │
│     │                                                                                                           │
│     └─────├┤────────├┤────────┌─────────────┐                                                                   │
│      0.6       0.6            │ PCI c3:00.0 │                                                                   │
│                               └─────────────┘                                                                   │
└─────────────────────────────────────────────────────────────────────────────────────────────────────────────────┘
┌─────────────────────────────────────────────────────────────────────────────────────────────────────────────────┐
│ Host: c3cpu-c15-u1-4.rc.int.colorado.edu                                                                        │
│                                                                                                                 │
│ Date: Wed 09 Oct 2024 02:15:18 PM MDT                                                                           │
└─────────────────────────────────────────────────────────────────────────────────────────────────────────────────┘
[vaho6394@login13 Lab05]$ 
```

Figure 1: Results of lstopo --of ascii

After running ‘lccpu’ we can see the details about the CPU. It is an AMD EPYC 7713P 64-Core Processor. Is has one thread per core, and 64 cores per socket. Having only one socket, the total number of threads per chip amounts to 64. More information can be found at the manufacturer’s website.



```
[vaho6394@c3cpu-c15-u3-1 Lab05]$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                64
On-line CPU(s) list:   0-63
Thread(s) per core:    1
Core(s) per socket:    64
Socket(s):             1
NUMA node(s):          1
Vendor ID:             AuthenticAMD
CPU family:            25
Model:                 1
Model name:            AMD EPYC 7713P 64-Core Processor
Stepping:              1
CPU MHz:               3094.344
BogoMIPS:              3992.70
Virtualization:        AMD-V
L1d cache:             32K
L1i cache:             32K
L2 cache:              512K
L3 cache:              32768K
NUMA node0 CPU(s):     0-63
Flags:                 fpu vme de pse tsc msr pae mce cx8 apic sep mtrr p
 pclmulqdq monitor ssse3 fma cx16 pcid sse4_1 sse4_2 x2apic movbe popcn
 perfctr_llc mwaitx cpb cat_l3 cdp_l3 invpcid_single hw_pstate sme ssbd
m_occup_llc cqm_mbm_total cqm_mbm_local clzero irperf xsaveerptr wbnoin
 rdpid overflow_recov succor smca
[vaho6394@c3cpu-c15-u3-1 Lab05]$ 
```

Figure 2: Results of lscpu

The Empirical Roofline Toolkit:

The Empirical Roofline Toolkit is a performance benchmarking and analysis tool. It is used in high-performance computing environments to measure the computational capabilities of computer systems. It is designed to assist in the evaluation of hardware and identify bottlenecks. The ERT provides insights into how well a system can execute floating-point operations and utilize memory bandwidth.

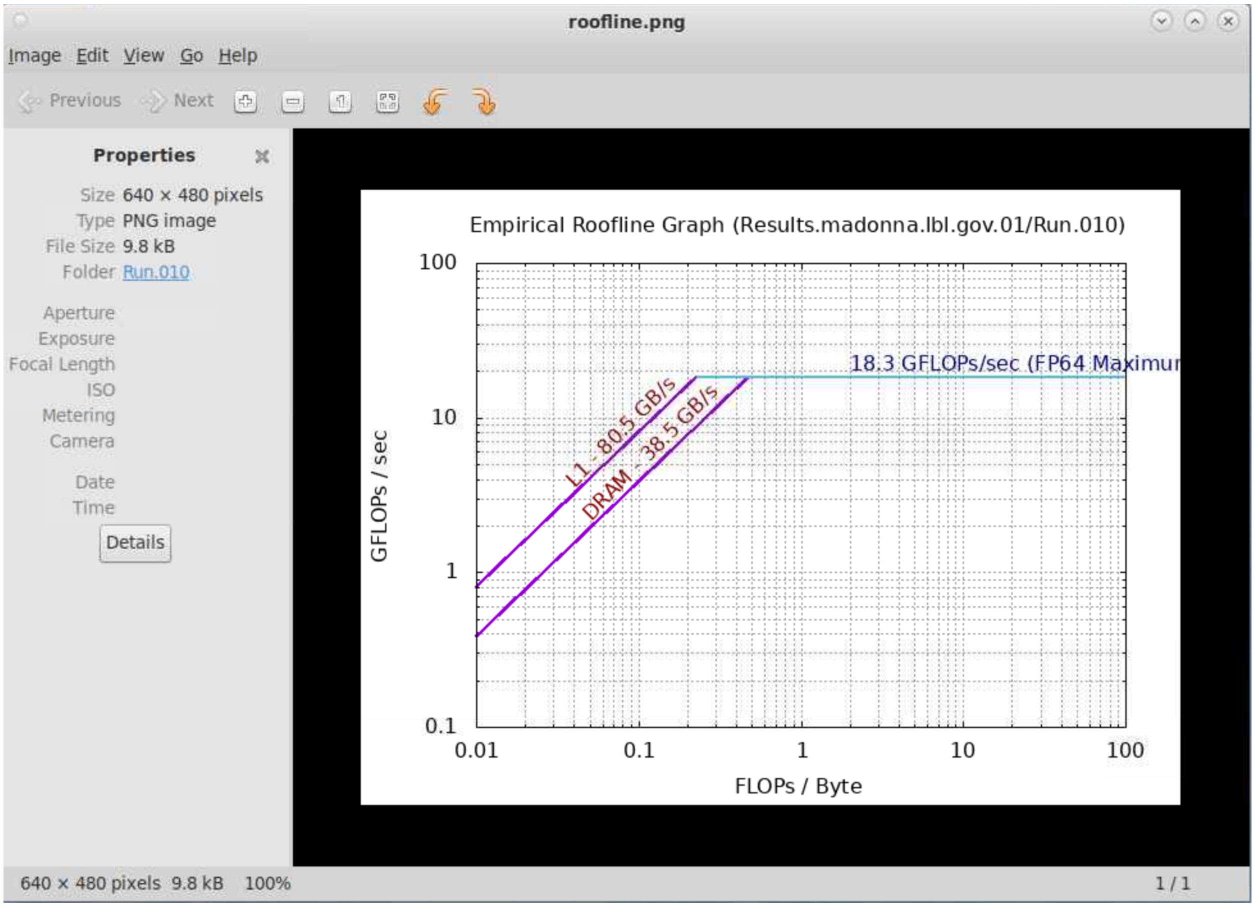
The Roofline chart has two main components:

1. Memory Bandwidth Bound: Represents the maximum data transfer rate between memory and the CPU. If an application's data transfer rate is below this limit, it is considered "memory-limited," meaning the CPU is "waiting on memory" to catch up to its speed.
2. Compute Bound (Peak FLOP Rate): Represents the maximum rate at which the CPU can perform floating-point operations (FLOPs). If an application's performance approaches this limit, it is considered "CPU-limited," meaning the CPU is operating at its limit.

These evaluations are executed based on a configuration that is specified in a .ert file. The .ert file has numerous tags that decide the configuration.

A few adjustments were needed to the configuration file that shipped with the ERT package. Alpine hardware doesn't support the sse3 instruction set, so those flags were removed in the ERT_CFLAGS. Another change is to restrict the ERT_PROCS_THREADS to 1, since the 'amilan' processor has only one thread per core.

The ERT_MPI_PROCS term defines the number of MPI processes that will be used to run the program that evaluates the hardware. The ERT_OPENMP_THREADS defines the number of OpenMP threads to be used withing each MPI process during the benchmark. The ERT_FLOPS specifies the number of floating-point operations to perform in each iteration of the benchmark. Together these parameters assist in generating a comprehensive Roofline plot that demonstrates the system's capabilities.



Size 640 × 480 pixels
Type PNG image
File Size 9.8 kB
Folder Run.010
Aperture
Exposure
Focal Length
ISO
Metering
Camera
Date
Time

640 × 480 pixels 9.8 kB 100% 1 / 1

Theoretical flop transfer rate computation:

Consider a processor having the following specifications:

- Number of cores N
- Clock speed F_c
- Number of virtual cores C_v = Number of cores * Number of threads per core
- Instructions per cycle I_c
- Fusing factor F_f

The theoretical clock speed is given by:

$$F_r = C_v * I_c * F_c * F_f$$

For the processor used by the 'amilan' partition on Alpine, we have

- $N = 1$
- $F_c = 3.094$ GHz
- $C_v = N * 1 = 64$
- $F_f = 2$
- $I_c = 4$

$$Fr = 1*3.094*2*4 = 24.7152 \text{ GB /s}$$

Theoretical memory transfer rate computation:

Consider the following specifications:

- Memory transfer rate MTR = 3200 MHz
- Number of memory channels MC = 8
- Bytes per memory access $T_n = 8$ bytes
- Number of sockets $N_s = 1$

The theoretical memory transfer rate is given by:

$$B_r = MTR * MC * T_n * N_s$$

$$B_r = 3200 * 8 * 8 * 1 = 204.8 \text{ GB/s}$$

As we can see, there is a wide discrepancy between the theoretical maximum values and the practical computational values of the memory transfer rate, and a small discrepancy in the FLOP rate values. Some potential reasons are:

1. Memory bandwidth limitations:
   a. Memory latency: Practical memory transfer rates may be limited by the latency of accessing memory.
   b. NUMA: In multi-socket systems, memory access times vary depend on whether the memory is local to a core or located in another socket. Accessing non-local memory can significantly reduce memory bandwidth.

   c. Cache misses: If the data is too large to fir in the processor's caches, frequent cache misses can hinder the rate at which data is transferred from main memory to the CPU.
2. Algorithm and data access patterns:
   a. Memory access patterns: Algorithms that randomly access memory or have poor data locality slow down the memory bandwidth.
   b. Algorithm efficiency: Some algorithms have inherently low computational intensity - such calculating the ratio of FLOPs to memory accesses - making the algorithms memory-bound and in a "waiting for memory" state, resulting in low FLOP rates.
3. Parallelization overhead: In multi-threaded applications there is an overhead for synchronizing across the parallel threads or processes. This could add to delays and affect the computed rates.

Appendix A: References

1. Alpine hardware documentation - https://curc.readthedocs.io/en/latest/clusters/alpine/alpine-hardware.html
2. AMD EPYC 7713 Documentation - https://www.amd.com/en/products/processors/server/epyc/7003-series/amd-epyc-7713p.html
3. Empirical Roofline Toolkit - https://github.com/ebugger/Empirical-Roofline-Toolkit
4. Empirical Roofline Toolkit - https://crd.lbl.gov/divisions/amcr/computer-science-amcr/par/research/roofline/software/ert/