Reflection on High-Performance Computing Semester

Camren J. Khoury
ECE 4990 High-Performance Computing, Clemson University

May 3, 2025

1 Introduction

During Spring 2025, I enrolled in the ECE 4990 High-Performance Computing course at Clemson University. Over the semester, I built a small HPC cluster, learned parallel programming models and practices, and applied benchmarking tools. This report summarizes what I did, what I learned, how it reshaped my view of HPC, and the lasting impact on my remaining time (and beyond) at Clemson.

2 Semester Activities

2.1 Cluster Setup

- Procured and prepared four Raspberry Pi 4 B boards running Raspberry Pi OS Lite.
- Configured static IPs and password-less SSH across nodes.
- Set up NFS for shared home directories.
- Installed MPI libraries (OpenMPI) and tested communication with simple `mpirun` "hello world" jobs.

2.2 Software Stack and Tools

- Used Spack for managing HPC software: MKL, HPL benchmarks.
- Compiled and ran the High-Performance Linpack (HPL) to measure FLOPS.
- Experimented with job scheduling using simple shell scripts (no Slurm).

2.3 Benchmarking and Optimization

- Collected performance data under varying process counts.
- Profiled code for communication vs. computation bottlenecks.
- Tuned HPL parameters (`NB`, `P x Q` grid sizes) for optimal throughput.

2.4 SBCC Competition

As part of the annual SBCC high-performance challenge, our team architected and deployed an 11-node Raspberry Pi cluster under real-world constraints:

- Provisioned 11 Raspberry Pi 4 B boards with static IPs, password-less SSH, and a shared NFS filesystem.
- Integrated MPI for parallel task distribution and ran the custom `DLLAMA` communication benchmark.
- Deployed and optimized `ParFEMWARP`, our proprietary finite-element solver, achieving scalable speedup across all nodes.
- Coordinated load-balancing and fault-tolerance strategies to ensure uninterrupted multi-hour execution.

2.5 Palmetto Supercomputer Access

Alongside our Raspberry Pi clusters, I obtained undergraduate access to Clemson's Palmetto supercomputer:

- **Compute Infrastructure:** Over 1,138 compute nodes (46,736 CPU cores), with 700+ GPU-equipped nodes and 12 large-memory (1 TB+) nodes.
- **Storage:** 250 GB home directories per user, 2.2 PB scratch space, and high-performance InfiniBand interconnects.
- **Scheduler:** Slurm workload manager for job submission, monitoring, and resource allocation.
- **Software Environment:** Centrally managed modules for MPI, compilers (Intel, GCC), and scientific libraries, with support for custom builds via Spack or Singularity.
- **Access & Support:** SSH login via `slogin.palmetto.clemson.edu`, Duo two-factor authentication, and dedicated RCD workshops for onboarding.

3 Challenges and Mitigation

During the semester I encountered a few recurring issues with the Raspberry Pi clusters:

- **SD Card Failures:** Several cards became corrupted. I replaced each faulty card and reflashed the OS image repeatedly until all nodes booted reliably.
- **Library Version Conflicts:** Dependency mismatches between system packages, Snap installs, and HPC libraries caused errors. I resolved this by iteratively trying different versions until I found a combination that installed and ran correctly.
- **Static IP Configuration:** Some Pis failed to appear on the network. I reconfigured their static IP settings multiple times until the static ips held over multiple boots and all four were concurrently available on the host network.

4 Key Learnings

- **Parallel Programming Concepts:** Mastered the SPMD model, MPI primitives (`MPI_Send/Recv`, `MPI_Barrier`), and the trade-offs highlighted by Amdahl's Law.

- **Systems and Networking:** Understood how network topology and shared NFS performance impact communication latency and I/O throughput.
- **Large-Scale Deployment:** Gained practical insights into multi-node cluster tuning through the SBCC competition, particularly balancing MPI communication patterns and application-specific optimizations for `ParFEMWARP`.
- **Software Management:** Learned to use Spack for reproducible environment management and containerization strategies for consistent deployments.

5 Impact on My View of HPC

Before this course, I viewed HPC as remote supercomputers. Now I recognize:

- *Accessibility:* Affordable hardware (Raspberry Pis) can effectively demonstrate core HPC principles.
- *Complexity:* Managing dependencies, tuning libraries, and orchestrating distributed workflows is as critical as writing parallel code.
- *Collaboration:* Successful HPC projects require teamwork,system administration, code optimization, and data analysis.

6 Impact on My Clemson Experience

This course has significantly enriched my undergraduate journey at Clemson by providing practical skills and conceptual insights that I'll carry forward:

- **Deepened Understanding of Parallel Computing:** Hands-on work with MPI and cluster orchestration clarified how tasks are divided, synchronized, and aggregated—knowledge I can apply in future coursework and research.
- **Microcomputer as Teaching Tools:** Deploying Raspberry Pis demonstrated that even low-power devices can effectively model large-scale parallel systems, reinforcing core concepts without requiring access to massive hardware.
- **Linux Systems Proficiency:** Managing multiple distributions, resolving library conflicts, and troubleshooting installations sharpened my command of Linux package management and system configuration.
- **Networking Fundamentals:** Configuring static IPs, NFS shares, and SSH connectivity improved my practical understanding of networked storage and secure remote access.
- **Debugging Distributed Systems:** Diagnosing node failures, SD-card issues, and performance bottlenecks boosted my problem-solving toolkit for future team projects involving heterogeneous hardware.
- **Research and Collaboration:** With this foundation, I'm better prepared to contribute to faculty-led HPC research, mentor peers in CEDC, and leverage Clemson's resources—like Palmetto—for honors projects or summer internships.

7 Preparation for Beyond Clemson

Industry Readiness

This course has provided:

- Hands-on cluster design and management experience—from OS configuration to networked storage.
- Deep familiarity with parallel paradigms, performance profiling, and parameter tuning for large-scale simulations.
- Expertise in reproducible workflows and environment management (Spack, containers), skills highly valued by employers.

8 Conclusion

Over the semester, hands-on cluster assembly, competitive deployment for the SBCC challenge, benchmarking with HPL and `DLLAMA`, and software-stack management reshaped my understanding of parallel computing. I transitioned from theoretical concepts to practical mastery—appreciating how communication overhead, library configurations, and system topology collectively determine real performance. These lessons position me to leverage Clemson's HPC resources in future projects and provide a solid foundation for tackling performance challenges in industry or research after graduation.