Enhancing the Price/Performance of a Clustered Multiprocessor System

Joseph Myre
Dr. Daniel Ernst (Faculty Mentor)
Department of Computer Science
University of Wisconsin - Eau Claire

1. Introduction & Motivation

Historically, a researcher wishing to perform computationally demanding studies has required access to a high performance supercomputer. Typically, such systems have only been available at institutions with hefty budgets: such as government labs [2], mega-corporations, or large research centers [1].

However, with the advances in off the shelf hardware, clustered systems of commodity microprocessors have become an extremely popular alternative to the big iron. A cluster is defined as a scalable architecture based on commodity hardware, a private system network, and usually running on open-source software [4]. The benefit provided by clusters is twofold:

- Cost can be greatly mitigated.
- Clusters enable institutions with less than extraordinary budgets to establish their own modestly scalable high-performance computing (HPC) environment.

Recently, a group at Calvin College designed and built Microwulf, a personal, portable Beowulf cluster, which provides over 26Gflops of measured performance at a total cost of less than $2500 [3]. This level of performance attained by Microwulf was one of the first to break the $100 per Gflop barrier at $94 per Gflop.

It was the primary objective of this research to explore the means and metrics by which low-cost microprocessor clusters could be constructed and evaluated for academic use. This included looking at price and performance numbers for components, evaluating the system by industry standards, as well as experimenting with machine configurations to determine how various hardware and network configurations impacted performance.

2. Design

Due to our budget we were forced to be very selective with the components we chose. Our applications are particularly CPU bound with few network requirements. So we maximized CPU performance while budgeting all other items. We used Pareto analysis for CPU selection which was near trivial and allowed us to eliminate CPUs that lay outside of our optimal bounds.

Amdahl's rule of thumb: *"1Hz of processor speed, requires 1B of RAM and 1bps of network bandwidth."*

By Amdahl's rule of thumb each node needed 10GB of RAM 10Gbits of network bandwidth. This was not feasible. With our budget we could have used either 4 nodes with 4GB of RAM each or 6 nodes with 2GB of RAM each By analyzing our application space we knew that each application instance would only require 256Mb of RAM. This let us maximize processing power with the 6 node configuration.

3. Software

All of our software was free which was great for our budget.

Linux was our OS of choice:

- It currently runs 82.5% of the Top 500 supercomputers. [5].
- Specifically Ubuntu Linux, for its ease of use and fast setup time.

Inter-node communication was implemented with MPI.

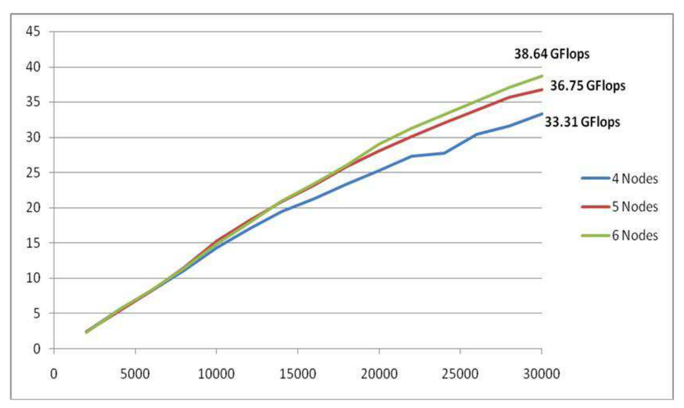
- MPI is standardized and portable.
- Users retain alarge portion of control.
- It is is a dependency of LINPACK.

We also installed two benchmarks for our evaluation:

- LINPACK: The industry standard benchmark. LINPACK primarily measures a systems double-precision floating point computational power by solving numerical linear algebra routines consisting of $N \times N$ linear systems, $Ax = B$. Matrix operations occur frequently in HPC applications so the calculation of hypothetical performance is easy. However, LINPACK results do not always indicate the real capabilities of a system since I/O capabilities are not stressed.
- SIM-MASE: SIM-MASE simulates the operation and performance of applications on a variety of microprocessor configurations. It is capable of running real applications so performance impacts can be clearly shown.

4. LINPACK

Performance vs. Problem Size



Our cluster reported 38.64Gflops for a cost efficiency of $71.92/Gflop, vs. MicroWulf which reported 26.25Gflops for a cost efficiency of $94.10/Gflop.

We found definite wisdom behind Amdahls rule of thumb since deviation from the rule had pronounced results.

- By having unbalanced memory, LINPACK performance was limited.
- By extrapolating the performance curve it is shown that the system is hypothetically capable of a problem size of 45,000 for 48.29Gflops.
- However, cost efficiency would remain fixed since increasing memory is a significant investment.

5. SIM-MASE

We ran instances of SIM-MASE over the full range of processors, 1 through 24.

- Increasing the number of processors did not impact performance.
- Each instance maintained 660 million instructions per second regardless of the number of instances running.

While memory was a primary limiting factor for LINPACK it was not a limiting factor for SIM-MASE at all. An increase in memory conceivably would have had little return. Considering this when analyzing our SIM-MASE runs to find price/performance, an interesting result was obtained.

- In the 6 node configuration (as is) the system would yield a price/performance of $0.0175/mips.
- In the 4 node configuration with more memory our system would have instead provided only $0.263/mips.
- The 4 node system would have been about 15 times worse!

5. Conclusions

Our results showed that general benchmarks do not necessarily reflect the real world performance of a system.

Had we judged our system solely based on LINPACK we likely we would have:

- Thought the same issues would be present in SIM-MASE.
- Paid for extra memory that never would have been used.

Additionally, our system shows that HPC resources can be attained by smaller universities. But more importantly:

- By understanding the application space we were able to optimize the system for a specific purpose.
- The optimizations dramatically improved real world price/performance.

References

[1] "Pittsburgh Doubles Capability of BigBen," *Pittsburgh Supercomputing Center news release,* Nov 21, 2006. http://www.psc.edu/publicinfo/news/2006/2006-11-21-bigbengrows.php

[2] Singer, Neal., "Red Storm upgrade boosts Sandia supercomputer to #2 in world," *Sandia National Laboratory news release, Nov 24, 2006.* http://www.sandia.gov/LabNews/061124.html

[3] Adams, Joel and Tim Brom., "Microwulf: A Beowulf Cluster for Every Desk," *Proceedings of the ACM Symposium on Computer Science Education,* (SIGCSE), 2008.

[4] beowulf.org., *http://www.beowulf.org/overview/index.html,* 2007.

[5] top500.org., "The Top500 Supercomputing Sites. Operating System Family Share for 11/2007.," *http://www.top500.org/stats/list/30/osfam,* 2007.

I would like to thank my faculty advisor, Dr. Dan Ernst,
as well as ORSP for funding