Northeastern University

Advisors: Kurt Keville, Yash Ukidave, Xiang Gong, Professor David Kaeli

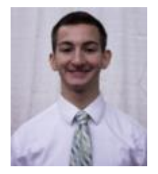
Students: Colleen Finnegan, Trevor Gale, Spencer Hance, Maddy Leger, Zachary Marcus, Carter McCardwell, Tristan Sweeney

Meet the Team

Our team is made up of 7 undergraduate students in Northeastern's Electrical and Computer Engineering Department. All of the students are part of NUCAR, a computer architecture research group led by Professor David Kaeli, and all have research interests in HPC, big data, and parallelization.

Colleen Finnegan
Junior, Electrical/Computer Engineering

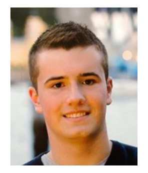
Trevor Gale
Sophomore, Computer Engineering



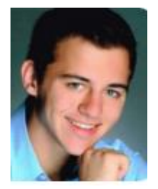
Spencer Hance
Sophomore, Computer Engineering

Maddy Leger
Sophomore, Computer Engineering

Zachary Marcus
Sophomore, Computer Engineering/Science



Carter McCardwell
Sophomore, Computer Engineering



Tristan Sweeney
Sophomore, Computer Engineering/Science

Our System

Our cluster is a homogeneous 3 node systems that employs kernel extensions for distributed symmetric multiprocessing (DSMP) and a large pool of shared physical memory to achieve peak performance on memory intensive applications. Each node features 4 16-core AMD Opteron 6380 processors and 32 DIMM's each with 16 GB 1600 MHz ECC DDR3 for a total of 1.5TB of shared memory across the entire system. Because of the simple shared memory system and remote thread handling kernel extensions, our system forgoes the use of a message-passing interface (MPI) in favor of a system specific implementation of OpenMP and Pthreads so each application can take advantage of all 192 cores and 1.5TB of shared memory.

Applications & Strategies

Our Strategy: One team member is the "system administrator", and each remaining team member focuses on one specific app. This way, we can effectively focus on each app and how it interacts with our unique hardware setup.

LINPACK

Our shared-memory system allows us to perform HPL on a massive matrix without the overhead of paging or message passing. This enables us to score a high N_max score, and with the speed of accessing shared memory, we should have a consistently high R_max.

Trinity

With Trinity's built in commands to change the amount of memory and threads it utilizes, we can take full advantage of the DSMP system. We can also utilize all of the cores on the system to greatly increase the speed of the second module, Chrysalis, which has support for multithreading.

HPC Repast

Since HPC Repast relies heavily on MPI, we will be running it on a standard Linux kernel instead of our DSMP version. Even without DSMP, our system is still competitive and boasts a large number of cores.

MILC

MILC benefits from shared memory because it uses a structures of lattices which are memory expensive. Quicker accesses to these mean a stronger performance.

WRF

WRF will be able to take advantage of the large amount of RAM on our system, enabling it to run faster. Because of this, it is likely that the whole process will be able to be done in memory.

Mystery App

Our strategy for the mystery app is the same as for all of our other apps – use our large amount of memory to store data and thus make the memory accesses quicker.

Why We'll Win

HPC is at a transition point. Moore's Law is coming to an end and Borkar's Law implies that we will not get energy efficiency out of "embarrassingly parallel" for much longer. This has created an opportunity for shared memory approaches to reenter code development efforts that are currently dominated by message passing methods. The hardware improvements of the next decade will come from boutique board and chip design solutions that were not affordable, whatever your value metric is, until recently. There is a shared memory solution out there for most production codes, including the ones we are running here, and now it is competitive with MPI on raw FLOPs, FLOPs per Watt, and FLOPs per Watt per sq. inch.

Acknowledgements

We would like to acknowledge our advisors, Kurt Keville, Yash Ukidave, Xiang Gong and Professor David Kaeli for the support they've given us during our preparation for SCC. We would also like to thank Richard Anderson and the team at Symmetric Computing, as well as AMD, for the design of the DSMP system.

NUCAR

AMD

SYMMETRIC COMPUTING