Multi-Core & the Hybrid Programming Model

Research Retreat 2009

Martin Chorley
Introduction

• HPC Architectures
• Hybrid Programming
• Research Questions
• Key Issues
  – Field of HPC
  – Hybrid Model (Past & Present)
  – Multi-Core Processors
HPC System Architectures

- **Distributed Memory** – Use Message Passing – MPI.
  - Explicit communication between processes.

- **Shared Memory** – Use Shared Memory threading – OpenMP.
  - Each thread has access to global shared memory – implicit comms.
Multi-Core Architectures

- Multi-Core nodes complicate system architecture.
  - Multiple levels of memory hierarchy and parallelism increase complexity over cluster of Symmetric Multi-Processing (SMP) machines.
Hybrid Programming

• Theory – shared memory accesses are faster than explicit communication.
• We can use Shared Memory programming within each node of this cluster.
• We can use Message Passing programming to communicate across all nodes of the cluster.

Shared Memory + Message Passing = Hybrid Programming
My Research – Key Question

• Does the Hybrid Model allow us to increase HPC application performance on modern multi-core clusters?

– Can a hybrid code deliver better performance than a pure MPI code?
– How do pure MPI and hybrid codes differ in their performance?
– Why do pure MPI and hybrid codes differ in their performance?
Current Progress

• Created hybrid version of Molecular Dynamics (MD) code and performance tested.
  – Hybrid version better in some situations, MPI version better in others.

• Currently creating and testing hybrid version of DLPOLY – real world HPC MD application.

• Use results to inform performance model to predict performance on next generation of hardware.
Key Issues

• High Performance Computing
  – Trends and Issues.
• Hybrid Programming
  – History and past performance.
  – Current performance.
• Multi-Core Processors
  – Application Performance.
High Performance Computing Trends

• Much literature available.
  – Many papers and articles by Dongarra et al\textsuperscript{[1][2][3]} and Strohmaier et al\textsuperscript{[4][5]}.

• Also raw data – Top500\textsuperscript{[6]} list.

• All point to increase in usage of clusters.

• Describe MPI and OpenMP as \textit{de facto} standards for HPC programming.

• Support the relevancy of this work.
Hybrid Programming Model

• Hybrid Programming proposed in the late 90’s.
• Performance studied and analysed on clusters of SMP’s by many people.\(^7\)[8][9][10][11]
• Consensus not reached on it’s effectiveness.
  – Effective in some applications, not in others.
• In general, performance is just as good using only MPI.
Hybrid Programming Model

• All past work focused on SMP clusters.
  – Clusters of large ‘fat-node’ SMP servers (8+ distinct physical processors per node)

• My research focused on multi-core clusters.
  – Nodes are ‘thinner’ – fewer cores per node.
  – Multiple levels of parallelism and memory sharing (shared cache, shared memory access).

• My work builds on prior performance studies, updating knowledge for current generation of hardware.
Hybrid Programming Model

• Some prior work is lacking in detail and analysis.
  – fails to examine the model on a cluster of SMP’s – focusing on one node only.

• Ignores communication between nodes as a factor in performance – focuses on difference in computation times only.
Hybrid Programming Model

- No current/recent performance studies of the hybrid programming model.
- Some work done on looking at the hybrid model and how threads and processes interact on multi-core machines.\(^{[12]}\)
- Works as expected – threads and processes do not interfere with one another.
- Allows research on hybrid codes to accept results are valid.
Multi-Core Processors

• Relatively new development - not as much literature available.

• Some recent studies[^13][^14][^15] looking at application performance with multi-core architectures.

• Provide pointers to potential performance problems that can be tested for and examined in my research.
Multi-Core Processors

- My research can add to this growing field of knowledge of how multi-core processors affect typical classes of HPC applications.
- Can examine both pure MPI codes (as in majority of current multi-core studies) and hybrid codes.
Conclusion

• Both Hybrid Programming and Multi-Core processors are topics of interest within the HPC community.
• My research combines these topics to examine how they affect application performance.
• Builds upon prior ideas while updating them for current and future hardware.
References

6. www.top500.org
9. Jost, G., Jin, H., an Mey, D. & Hatay, F. Comparing the OpenMP, MPI, and Hybrid Programming Paradigms on an SMP Cluster. NASA Ames Research Center, Fifth European Workshop on OpenMP (EWOMP03) in Aachen, Germany 2003