Performance Analysis of MPI and Hybrid MrBayes Computational Biology Code on Multicore Systems

Holly Williams hlly.williams@yahoo.com
Dr. Valerie Taylor taylor@cse.tamu.edu
Dr. Xingfu Wu wuxf@cs.tamu.edu
Department of Computer Science and Engineering, Texas A&M University

Motivation

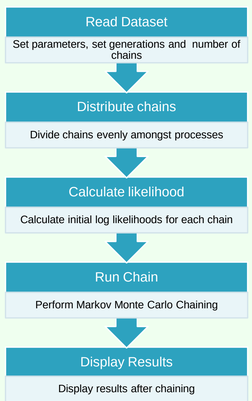
- The current trend in parallel computing has shifted towards multicores
- MPI and OpenMP are two popular parallel programming paradigms
- Combining OpenMP and MPI to construct a hybrid program for an application allows users to explore multiple levels of parallelism on multicore systems

Goals

Analyze and compare performance for MPI and hybrid versions of MrBayes on multicore systems and analyze energy consumption

Application

MrBayes is a computational biology program for Bayesian estimation of phylogeny. MrBayes uses a simulation technique called Markov chain Monte Carlo to approximate the posterior probabilities of trees.

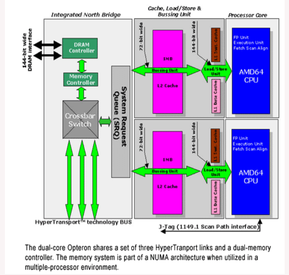


- MPI MrBayes
  - Written in C with MPI
  - Seven benchmark datasets
  - Developed by Fredrik Ronquist, John P. Huelsenbeck, and Paul van der Mark
- Hybrid MrBayes
  - Based on MPI MrBayes
  - OpenMP is applied to parallelize for loops in the mcmc source file where the Monte Carlo chaining is handled
  - Developed by Wayne Pfeiffer, Alexandros Stamatakis

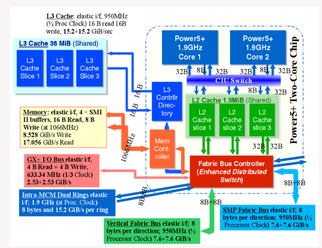
Multicore Systems

	Dori	Hydra
Number of Nodes	8	52
CPUs per node	4	16
Cores per chip	2	2
Total number of cores	32	832
CPU type	1.8 GHz AMD Opteron	1.9GHz IBM Power5+
Memory per node	6GB	32GB/49 nodes 64GB/3nodes

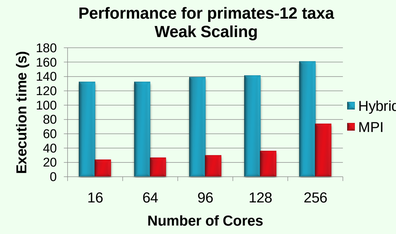
Data Flow view of the AMD Opteron™ Processor Dual core Model 100

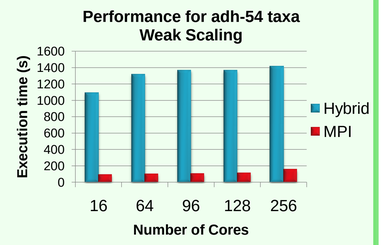


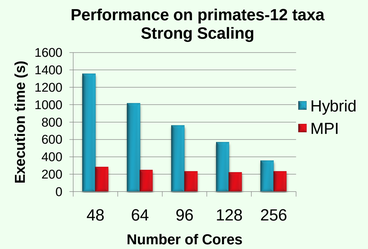
The dual-core Opteron shares a set of three HyperTranport links and a dual-memory controller. The memory system is part of a NUMA architecture when utilized in a multiple-processor environment.

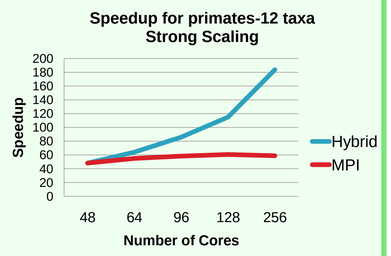


Experimental Results on Hydra









- There was a memory problem when running with Hydra
  - Only the two smallest datasets could be run using strong scaling
  - Weak scaling was used for all other datasets
- Results show that MPI MrBayes has lower execution time than hybrid MrBayes.
- Strong scaling for primates also shows that the MPI-only version has a lower execution time.
  - Speedup data shows that the hybrid version experienced more speedup than the MPI version.

Methodology

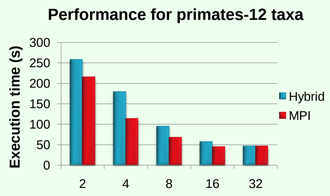
Performance of MrBayes on Multicore Systems

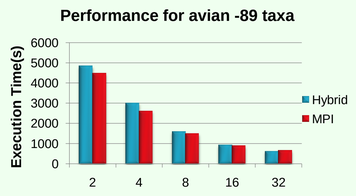
- Seven different datasets were run on two multicore systems, Hydra and Dori
- Each dataset was run for 10,000 generations on a different number of cores
- A different number of chains was used
  - On Dori, 32 chains were used
  - On Hydra, up to 512 chains were used
    - Weak scaling - 16 chains per node
    - Strong scaling - 512 chains
- Analyze and compare the performance of the hybrid version and the MPI-only version using execution time and speedup

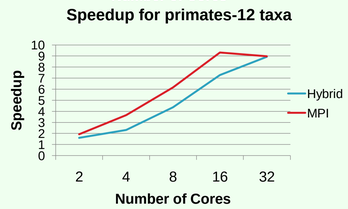
Energy Consumption of MrBayes

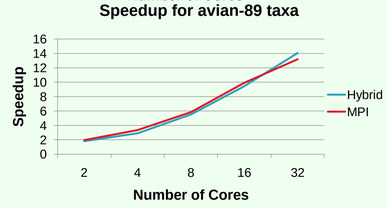
- PowerPack was used on Dori to collect energy data
- MPI-only and hybrid versions of MrBayes were run on one dataset
- We compared the energy consumption of the hybrid version with that of the MPI-only version

Experimental Results on Dori









Energy Consumption of MrBayes for primates-12 taxa

No. of Cores	Time (s)	System (kJ)	CPU (kJ)	Memory (kJ)	Hard Drive (kJ)	Motherboard (kJ)
2-hybrid	254.154	45.963065	27.30582	5.256297	2.103262	3.481821
2-mpi	214.611	39.263947	23.852611	3.997544	1.809655	2.966753
4-hybrid	178.418	37.648057	22.565793	3.875879	1.518767	2.486724
4-mpi	113.77	24.91126	14.459266	2.647337	0.972663	1.587298
8-hybrid	94.565	39.304558	22.842574	3.83381	1.545488	2.580222
8-mpi	67.301	28.043112	15.750636	2.833562	1.154268	1.855912
16-hybrid	56.887	45.89634	25.943368	4.370968	1.896912	3.085648
16-mpi	43.887	34.622904	18.566292	3.34392	1.378972	2.38708
32-hybrid	46.099	72.668664	40.61384	6.896368	3.023776	5.036632
32-mpi	46.651	71.06504	39.91648	7.145648	3.3108	5.181408

Conclusions and Future Work

- The results of this experiment show that the MPI version of MrBayes outperforms hybrid MrBayes.
  - Hybrid MrBayes only uses OpenMP in the mcmc source file
  - The program benefits from more MPI processes than OpenMP threads because the hybrid program is not fully parallelized.
  - MrBayes divides the chains amongst the MPI processes. This caused a memory issue for the hybrid version.
  - Hybrid MrBayes scaled better than the MPI version.
- The hybrid version also has higher energy consumption than the MPI-only version.
- For future work, the hybrid version can be improved by using OpenMP to parallelize the loops in other functions
- Through this project, I learned about parallel programming using MPI and OpenMP and how to do performance analysis

Acknowledgments

- I would like to thank my mentors, Dr. Valerie Taylor, Dr. Xingfu Wu, and Charles Lively for all the their help and guidance.

- This project was supported by CRA-W DREU Program and the REU program at Texas A&M University