

Implementing Parallel Hmm-pfam on the EARTH Multithreaded Architecture

Weirong Zhu, Yanwei Niu, Jizhu Lu, Guang R. Gao
Department of Electrical & Computer Engineering, University of Delaware
{weirong, niu, jlu, ggao}@capsl.udel.edu

Abstract

Hmmpfam is a widely used computation-intensive bioinformatics software for sequence classification. This poster describes a new parallel implementation of hmmpfam on EARTH, which is an event-driven fine-grain multi-threaded programming execution model. The comparison results of the original PVM implementation and our implementation shows notable improvements on absolute speedup and scalability. On a cluster of 128 dual-CPU nodes, the execution time of a representative testbench is reduced from 15.9 hours to 4.3 minutes.

1. Introduction

HMMER is an implementation of profile hidden Markov Models [1][2] (profile HMMs) for sensitive database searching. Hmmpfam is one program in the HMMER 2.2g package, it is a widely used tool for searching a single sequence against an HMM database. In real situations, this program may need a few months to finish processing large amounts of sequence data. Thus parallelization of the Hmmpfam is an urgent demand from bioinformatics researchers.

In this poster we will show a detailed analysis of the hmmpfam program and different parallel implementation of HMM-pfam on EARTH [3] - an event-driven fine-grain multi-threaded program execution model. Then we will show our test results on various computing environments.

2. PVM Implementation in HMMER 2.2g

HMMer 2.2g provides a parallel hmmpfam program based on PVM [4] (Parallel Virtual Machine). In this implementation, the computation for one sequence is executed concurrently, the master node dynamically assigns one profile to a specific slave node for comparison. Upon finishing its job, the slave node reports the results to the master, which will respond by assigning a new profile. When all the comparison regarding this sequence is completed, the master node sorts and ranks all the results it collects, and outputs the top hits. Then the computation on the next sequence begins. However, the experimental results show that this implementation does not achieve good scalability.

3. The Parallel Implementation on EARTH

Two parallel schemes of the hmmpfam algorithm are implemented on EARTH developed by CAPSL group at the University of Delaware.

3.1. The EARTH Execution Model

EARTH (Efficient Architecture for Running Threads) is an event-driven fine-grain multi-threaded programming execution model. In its current implementations, the EARTH runtime system (version 2.5) performs fiber scheduling, inter-node communication, inter-fiber synchronization, global memory management, dynamic load balancing and SMP node support.

3.2. Task Decomposition Scheme

The basic idea of hmmpfam algorithm is to read a single sequence from seqfile each time and to compare it against all the HMMs in the hmmfile looking for significantly similar sequence matches. In our new scheme, we consider the computation of one sequence against the whole database as a single job. Because normally the number of sequences in a seq file is much larger than the number of computing nodes available, this decomposition scheme still can achieve ideal parallelism. Moreover, since all the computation of one single sequence will be performed on one fixed node, there is no need to send back the result to master node, the sorting and ranking can be done locally, thus there is no barrier and no more communication needed between the master and this slave.

3.3. Two Parallel Schemes on EARTH

For parallelizing hmmpfam, we develop two different schemes. In the static load balancing scheme, the programmer pre-determines job distribution on all computing nodes by a round-robin algorithm. The other scheme takes advantage of the dynamic load balancing support of EARTH Runtime system, which simplifies the programmer's coding work by making the job distribution completely transparent. In this scheme, once a slave node finishes a job, it sends a request to the master process.

Master responds by sending back a new job to satisfy the slave's work requirement thus to keep it busy. The job-request and job-assignment are determined by EARTH RTS dynamically and transparently.

4. Experiment Result

The experiment was conducted on two clusters. One is "COMET", which consists of 18 nodes, each containing two 1.4 GHz AMD Athlon processors. The second one – "Chiba City" [5], which locates at Argonne National Laboratory, is comprised of 256 computational servers, each with two 500MHz Pentium III processors. The experiments are done by using two data sets. Data set 1 includes a HMM database containing 585 profile families, and a sequence file with 250 sequences; the data set 2 includes a HMM database containing 50 profile families, and a sequence file containing 38192 sequences.

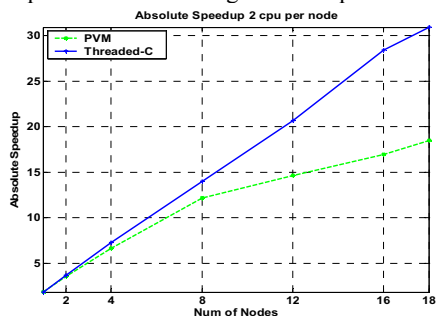


Figure 1. Comparison of PVM version and EARTH version on COMET

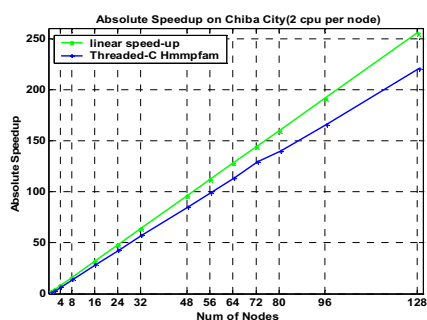


Figure 2. Static Load Balancing Scheme on Chiba City

From figure 1, it is easily seen that our new version has much better scalability than the PVM implementation. From figure 2 and figure 3, the near linear speedup curves on supercomputing cluster are achieved for both static and dynamic load balancing scheme. An absolute speedup of 222.8 on 128 dual-CPU nodes is obtained for data set 2, which means that it could reduce the total execution time from 15.9 hours (serial program) to only 4.3 minutes.

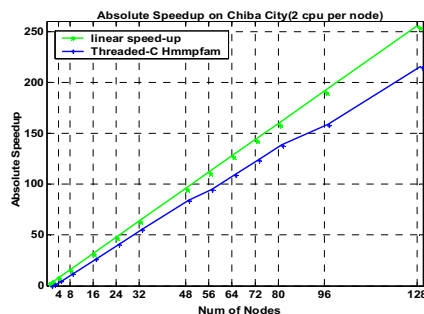


Figure 3. Dynamic Load Balancing Scheme on Chiba City

4. Conclusion

Sequence family classification and HMM database searching is very important to the biological research community. With the help of supercomputing resources, researchers can now save research time and get new discoveries more quickly than ever. Porting of hmmpfam to EARTH model provides very promising result, in further research, it is worthwhile to port other bioinformatics applications such as multiple alignments to EARTH platform.

5. Reference

- [1] Eddy, S.R. "Profile hidden Markov models", *Bioinformatics*, Oxford University Press, Oxford, UK, 14, 1998, pp.755–763.
- [2] Btman A, Birney E, Cerruti L, Durbin R, Etwiller L, Eddy SR, Griffiths-Jones S, Howe KL, Marshall M, Sonnhammer EL. "The Pfam Protein Families Database", *Nucleic Acids Research* 30(1), 2002, pp. 276-280.
- [3] Kevin B. Theobald, "EARTH: An Efficient Architecture for Running Threads", *PhD thesis*, McGill University, Montreal, Quebec, May 1999.
- [4] A. Geist, A. Beguelin et al., "PVM: Parallel Virtual Machine", the MIT press, Cambridge, Massachusetts, 1994.
- [5] Chiba City, the Argonne Scalable Cluster, Argonne, Illinois, <http://www-unix.mcs.anl.gov/chiba>.