Reconstruction of Maximum Likelihood Phylogenetic Trees in Parallel Environment Using Logic Programming

Satoshi OOta ootа@thinker.lab.nig.ac.jp

Naruya Saitou nsaitou@genes.nig.ac.jp

Department of Genetics, School of Life Science, Graduate University of Advanced Studies
and
Laboratory of Evolutionary Genetics, National Institute of Genetics, Mishima, 411 Japan

Abstract

With rapid increase of nucleotide and amino acid sequence data, it is required to develop reliable and flexible application programs to infer molecular phylogenetic trees. The maximum likelihood method is known to be robust among many methods for reconstruction of molecular phylogenetic trees, however, this method requires extremely high computational cost. Although parallel computation is a good solution to realize complicated inference such as the maximum likelihood method, generally speaking, it is not so easy to implement parallel programs. In actual data analyses, furthermore, it is often needed to modify or expand application programs. In other words, there is no perfect application program for all data analyses. Logic programming is a good candidate to implement such data analysis programs in natural science fields, because programs in logic programming are easy to write, easy to modify, and easy to implement in parallel environment. We thus have developed an experimental system for reconstruction of phylogenetic trees in parallel environment in logic programming as a part of molecular evolutionary analysis system DeepForest. We propose the core algorithm for parallel execution of the maximum likelihood and show its verification according to simulation using amino acid data.

1 Objectives

Our objectives in this study are as follows. (1) Efficient iterative computation. (2) Robustness for data size. (3) Portability of the system for various environments. (4) Flexible parallel computation depending on data.

2 Methods

The most apparent computation which can be carried out in parallel is the topology search. Search of one topology is completely independent from that of another topology. The search space will increase almost exponentially if we execute the exhaustive search. Thus it is necessary to introduce some heuristics [4], however, it may violate the independency among subproblems. The evaluation for each site on the sequences can also be executed in parallel when we assume that mutations occur at each site independently. This strategy is powerful in terms of independency of the subproblems. To realize portability of the system for various environments and flexible parallel computation depending on data, our programs have been written in KL1 (KLIC 2.002) [1, 5], which is parallel logic programming. We used the Cray CS-6400 (shared memory system) of NIG.

3 Results and Discussion

Parallel execution. Using 20 CPUs, about 11 times speed-up was obtained for 5 actin amino acid sequences (376 a.a. sites). Interestingly, when we used larger data (10 MRF family's amino acid sequence data of 90 a.a. sites), the speed-up effect was about 17 times using 20 CPUs. It suggests *DeepForest* is robust for large data. However, actual computation time is not enough for large data analyses. More sophisticated algorithm is needed. Furthermore, *pragma* [1] was not so adjusted for appropriate data stream. It might make the speed-up effect unstable (Figure 1).

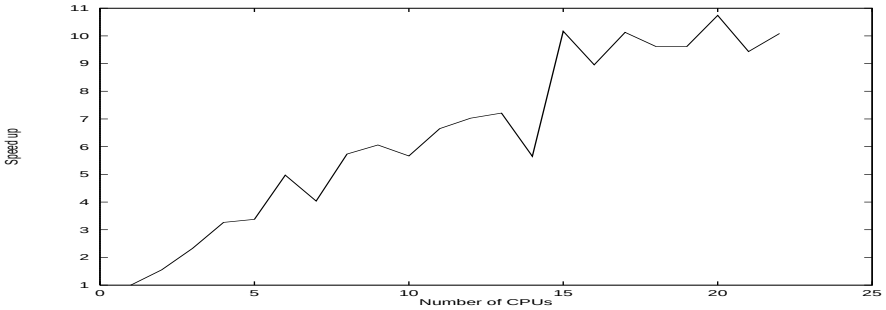

Figure 1: The speed up effect of *DeepForest* on shared memory system (Cray CS-6400).

Simulation. In the model tree, the branch lengths is always integer, while the reconstructed tree has expected branch lengths, which may not be integer. However, inferred branch lengths are quite similar to integers. It suggests that this method gave good inference. Every branch length of the reconstructed tree is longer than one of the model tree. It is because the branch length optimization program *traverse/4* uses a simple Poisson model [3], while the simulation program *sim/3* uses Dayhoff's PAM 250 matrix [2]. Although such difference exists, the inference was sufficiently good. It suggests usefulness and the robustness of our maximum likelihood program.

4 Acknowledgments

We appreciate kind advices of Dr. Takashi Chikayama of University of Tokyo and Dr. Masato Ishikawa of Meiji university. Especially, Dr. Chikayama provided us patches for parallel computation on shared memory system. We thank the KLIC users group for providing us various informations on KLIC/KL1. This work was supported by grants of AITEC, Japan and grants-in-aid of MESSC, Japan.

References

[1] Takashi Chikayama, Hiroyuki Sato, and Toshihiko Miyazaki. Overview of the parallel inference machine operating system (PIMOS). In *Proceedings of FGCS'88*, pages 230–251, Tokyo, Japan, 1988.

[2] M. O. Dayhoff. Survey of new data and computer methods of analysis. In M. O. Dayhoff, editor, *Atlas of protein sequence and structure*, pages 2–8. Silver Springs, 1978.

[3] Joseph Felsenstein. Evolutionary trees from DNA sequences: a maximum likelihood approach. *Journal of Molecular Evolution*, 17:368–376, 1981.

[4] Naruya Saitou. Reconstruction of gene trees from sequence data; computer methods for macromolecular sequence analysis. In Russell F. Doolittle, editor, *Methods in enzymology*, volume 266, pages 428–449. Academic Press, Inc., 1996.

[5] Kazunori Ueda. Guarded horn clauses. Technical report, ICOT, 1985.