Molecular Phylogenetic Analysis using both DNA and Amino Acid Sequence Data and Its Parallelization

Hideo Matsuda, Hiroshi Yamashita and Yukio Kaneda
matsuda@seg.kobe-u.ac.jp {momo, kaneda}@koto.seg.kobe-u.ac.jp

Department of Computer and Systems Engineering,
Faculty of Engineering, Kobe University
1-1 Rokkodai, Nada, Kobe 657 Japan

Abstract

Phylogenetc analysis of DNA sequences has played an important role in the study on evolution of life. However recent researches suggest that in some cases phylogenetic analysis of protein sequences is more important than that of DNA sequences. Thus we developed a system for phylogenetic analysis of protein sequence data. Since this system is based on our previously developed system for the analysis of DNA sequence data, one can obtain both protein-based and DNA-based trees and compare them. In the two systems, we took the same tree-construction algorithm (so called, a maximum likelihood method). Although this method has concrete models of the evolutionary process, it requires a huge amount of computational costs especially in the analysis of protein sequence data. Therefore we parallelized tree-construction steps in our method on a massively parallel machine.

1 Introduction

Biology has been rapidly becoming computational analytical science. As DNA sequencing enables researchers to achieve enormous increasing in sequence data, computing methods that allow efficient processing of those data play a crucial role in almost all fields of biological research [1]. In order to interpret these huge and complex data, extremely high computational cost is demanded. Massively parallel processing is thus raised as one of the key technologies to solve this issue.

Molecular phylogenetic analysis is one of the biological fields that require a large amount of computation [2, 3, 4]. It examines the molecular sequence data of taxa (biological entities

松田秀雄，山下浩，金田悠紀夫：神戸大学工学部情報知能工学科，〒 657 神戸市灘区六甲台町 1–1

such as genes, proteins, individuals, populations, species, or higher taxonomic units), infers their evolutionary process, and constructs their phylogenetic tree, that is, the family tree of the taxa [4]. A wide spread misconception of this analysis is that the building of phylogenetic trees simply requires the grouping of taxa according to overall similarities. This approach ignores the possibility that apparent overall similarity and true evolutionary relationship are not necessarily the same thing [5].

Phylogenetic analysis has mainly used DNA sequence data. However recent researches suggest in some cases phylogenetic trees based on the analyses of DNA sequences are misleading – especially when G+C content differs widely among lineages – and that protein-based trees from amino acid sequences may be more reliable [6, 7]. Thus we developed a system for phylogenetic analysis of protein sequence data. Since this system is based on fastDNAml [8] that is our previously developed system for phylogenetic analysis of DNA sequence data, one can construct phylogenetic trees using both of DNA sequences and their translated amino acid sequences by the same algorithm.

The tree-construction method we used is based on maximum likelihood inference developed by Felsenstein [9]. Approaches based on maximum likelihood have concrete models of the evolutionary process and are well-motivated statistically [10]. However their use has been hindered by the computational costs involved. In order to reduce the costs, we parallelized tree-construction steps in the maximum likelihood method. We will describe this later.

Similar research is reported in [11, 12]. It also takes a maximum likelihood approach and uses an original tree-construction method (called star-decomposition). Yet our system uses the same tree-construction method (stepwise addition) as Felsenstein's PHYLIP DNAML [13]. Another difference is that our system is parallelized for executing on massively parallel machines.

2 Maximum Likelihood Method

A number of methods for molecular phylogenetic analysis have been proposed [2, 3, 4, 5]. The algorithm we used is based on a maximum likelihood method proposed by Felsenstein [9]. In the algorithm, a phylogenetic tree is expressed as an unrooted tree. **Figure 1** shows a phylogenetic tree for three taxa and three possible alternative trees for four taxa.

Specifically, one seeks the tree and its branch lengths that have the greatest probability of giving rise to a given molecular sequence. The sequence data for this analysis include gaps by sequence alignment. During evolution, sequence data of taxa are changed by insertion, deletion and substitution of DNA bases. In order to compare evolutionarily related parts of taxa, several gaps are inserted corresponding to the insertion or deletion of DNA bases (or their translated amino acids). Consequently, an evolutionary change can be described by a substitution of a base (or a gap) at a base position of a molecular sequence with the base (or the gap) at the same position of another sequence.

The probability is computed on the basis of a stochastic model (Markov chain model of order one) on DNA base (or amino acid) substitutions in an evolutionary process. The model assumes a base (or an amino acid) substitution at a position of a molecular sequence takes place independently of substitution at other base (or amino acid) positions within a sequence.

As an example of the maximum likelihood method, consider a tree of size 3. Such a tree has three sequences observed from current taxa (corresponding to three tips in Figure 1) and one

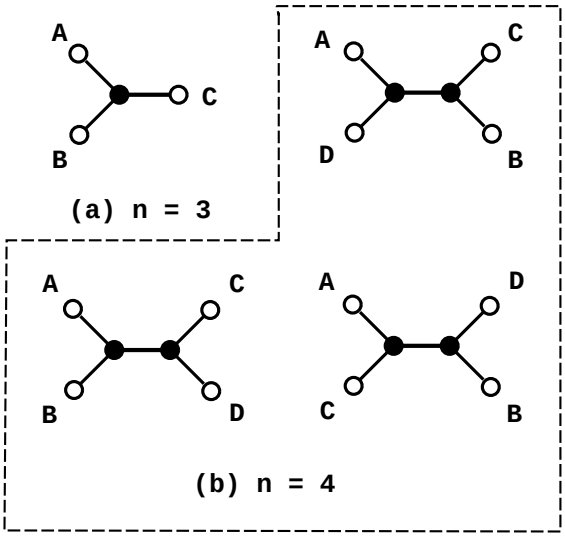


Figure 1: Phylogenetic trees expressed as unrooted trees.

unobserved sequence (the center node) which denotes the common ancestor taxa of the three current taxa. A tree of size 4 can be built from this tree of size 3 by adding one more sequence in all possible locations (of which there are 3, since the new sequence's branch can intersect any of the branches in a tree with three tips as shown in Figure 1). The objective is to choose the lengths of the three branches (v_1, v_2 and v_3) with maximum likelihood.

For a position j in the four molecular sequences, let $t_1(j), t_2(j), t_3(j)$ and $c(j)$ be the DNA bases (or the amino acids) in the three observed sequences and the base (or the amino acid) in the unobserved sequence, respectively. The likelihood of the position j is expressed:

$$L(j) = \sum_{c(j)=1}^{l} \pi_{c(j)} P_{c(j)t_1(j)}(v_1) P_{c(j)t_2(j)}(v_2) P_{c(j)t_3(j)}(v_3), \tag{1}$$

by adding over all possible values (in DNA sequences $l = 4$, or in amino acid sequences $l = 20$) of $c(j)$. Here the symbols in Equation 1 denote as follows:

π_x: the initial probability that a base (or an amino acid) in the unobserved sequence has a value x.

$P_{xy}(v)$: the transition probability that a base (or an amino acid) x in a sequence is substituted with a base (or an amino acid) y in another sequence within a branch length v.

v_1, v_2 and v_3: the three (unknown) branch lengths in a tree of size 3.

The transition probability depends on DNA base (or amino acid) substitution model. In fastDNAml, we had used a model taken by Felsenstein's DNAML program (a generalized two-parameter model [14]) for DNA base substitutions. In our system, we took a model proposed in [11] based on an empirical transition matrix compiled by Dayhoff and her coworkers [15].

Then the whole likelihood is expressed over all positions of sequences:

$$L = \prod_{j=1}^{m} L(j). \tag{2}$$

Here m is the length of sequences (note that after the sequence alignment, the lengths of all sequence data are arranged to the same size).

The three possible branch lengths v_1, v_2 and v_3 in a tree of size 3 are computed by solving equations to maximize likelihood L:

$$\frac{\partial L}{\partial v_1} = 0, \qquad \frac{\partial L}{\partial v_2} = 0, \qquad \frac{\partial L}{\partial v_3} = 0. \tag{3}$$

One can build a tree of size 4 from a tree of size 3 by adding one more sequence in all possible locations. Then one can build a tree of size 5 by adding another sequence to the most likely tree of size 4. In general, one can build a tree of size i from a tree of size $i-1$, until all n taxa have been added. Since there are $2i-5$ branches into which the i-th sequence's branch point can be inserted, there are $2i-5$ alternative trees to be evaluated and compared.

If the number of possible trees for a given set of taxa is not too large, one could generate all unrooted trees containing the given taxa, and compute the branch lengths for each that maximize the likelihood of the tree giving rise to the observed sequences. One then retains the best tree.

However, when the number of taxa becomes larger, the number of bifurcating unrooted trees is,

$$\prod_{i=3}^{n}(2i-5) = \frac{(2n-5)!}{2^{n-3}(n-3)!} \tag{4}$$

which rapidly leads to numbers that are well beyond what can be examined practically. Thus, some type of heuristic search is required to choose a subset of the possible alternative trees to examine.

Felsenstein develops a search algorithm in his software package [13]. It performs successive tree expansion by iterating steps constructing a tree of size i from a tree of size $i-1$ until all n taxa have been added. For each step, only the best tree is selected and the others are discarded.

3 Implementation of Parallel Processing

To reduce the computational time of the maximum likelihood method, we took a kind of functional parallel approach [16] that splits the computation into two processes: a *master* process that generates alternative tree topologies, a *slave* process that computes the branch lengths of the trees and their likelihood.

Then a general-purpose *dispatcher* is introduced between the master and the slave. This process initiates an arbitrary number of copies of the slave and distributes alternative trees to them. A *merger* process, which collects the results (their branch lengths and likelihood) to be sent back to the master, is also introduced for reducing the overhead to collect the results from the slaves. Creation of these processes and communication among them are implemented using a portable parallel processing system *p4* developed at Argonne National Laboratory [19].

By measuring execution time in this type of parallelism on a massively parallel machine, the Intel Touchstone DELTA, we realized its speedup factor was saturated on more than several tens of processors [16]. Also it required a whole set of sequence data in every process computing

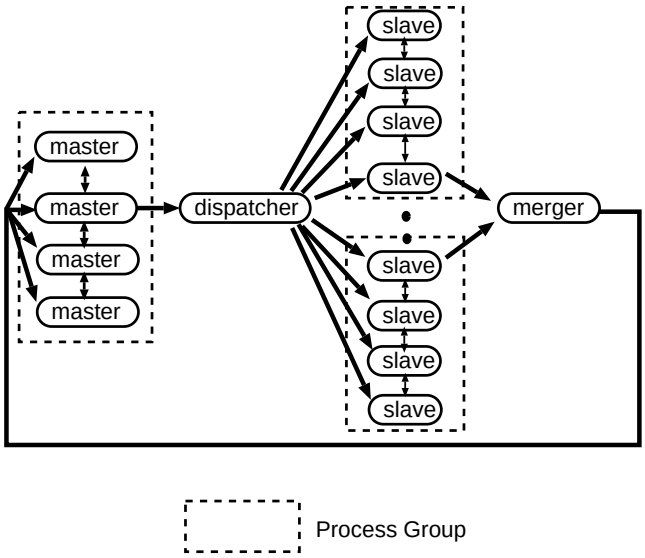


Figure 2: Process model for combining functional and data parallelisms.

branch lengths. It is not practical that each processor has a whole set if a large size of sequence data are to be examined.

In order to overcome these issues, we developed a parallel method for computing likelihood based on a data-parallel approach [17]. The method briefly consists of two steps, computing partial likelihood at each position of sequences in parallel (see Equation (1)) and combining those partial likelihoods into a complete one as described in Equation (2) [18]. It utilizes that the maximum likelihood method assumes a substitution in a sequence is carried out independently on the other sites of the sequence.

To combine these two types of parallelism, we divided a master (and a slave) process into a group of master (and slave) subprocesses corresponding to the number of the partition of sequence data (see **Figure 2**).

In a process group, each subprocess performs SPMD (Single Program Multiple Data Stream) execution that computes partial likelihood with a different part of sequence data and exchanges its partial likelihood with the other subprocesses to combine them into a complete one. Thus only one type of communication (a global multiply of likelihood) is required for this computation. We also implement it using global operation facility provided by the p4 system.

As shown in Figure 2, dispatcher and merger send input data to each slave and master subprocess, respectively. In a process group, each subprocess receives same data. But only one of subprocesses in a group needs to send its output since the output of the subprocesses is the same (i.e., they select the same tree topology).

4 Performance Results

To measure the performance of our system, we constructed phylogenetic trees using the RNA sequence data of small-subunit ribosomal RNA (16S and 18S rRNA) and the protein sequence

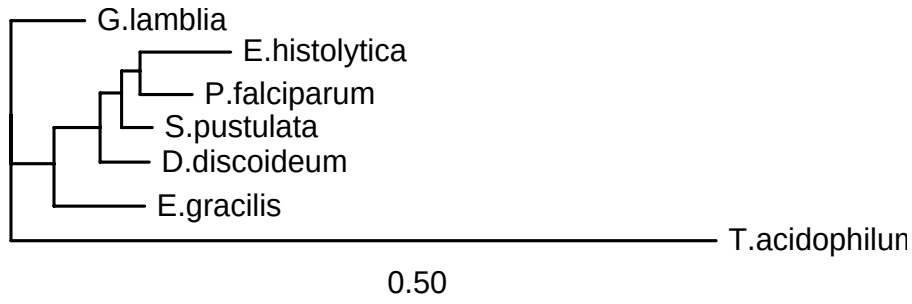


Figure 3: A phylogenetic tree inferred from ribosomal RNA sequence data.

data of elongation factor-1α (EF-1α) on the basis of a work [7].

Ribosomal RNA sequence has been widely used for constructing phylogenetic trees since it can be seen in all organisms. EF-1α is useful protein in tracing the early evolution of life [20] because it can be also seen in all organisms and the substitution rate of its sequence is relatively slow; for example, more than 50% identity is retained between eukariotic and archaebacterial sequences [7].

The data of small-subunit ribosomal RNA and EF-1α protein sequences were obtained from the Ribosomal Database Project [21] and Entrez [22], respectively. The EF-1α sequences include six eukariotes *Dictyostelium discoideum* (EMBL Accession No. X55972), *Entamoeba histolytica* (GenBank Accession No. M92073), *Euglena gracilis* (EMBL Accession No. X16890), *Giardia lamblia* (DDBJ Accession No. D14342), *Plasmodium falciparum* (EMBL Accession No. X60488) and *Stylonychia lemnae* (EMBL Accession No. X57926), and an archaebacterium *Thermoplasma acidophilum* (EMBL Accession No. X53866). The archaebacterium was used as an outgroup of the other organisms. We made the alignments of sequences using ClustalV [23].

The ribosomal RNA sequences are aligned by the Ribosomal Database Project. From the sequence database, We selected the same organisms described above except *Stylonychia pustulata* instead of *Stylonychia lemnae*.

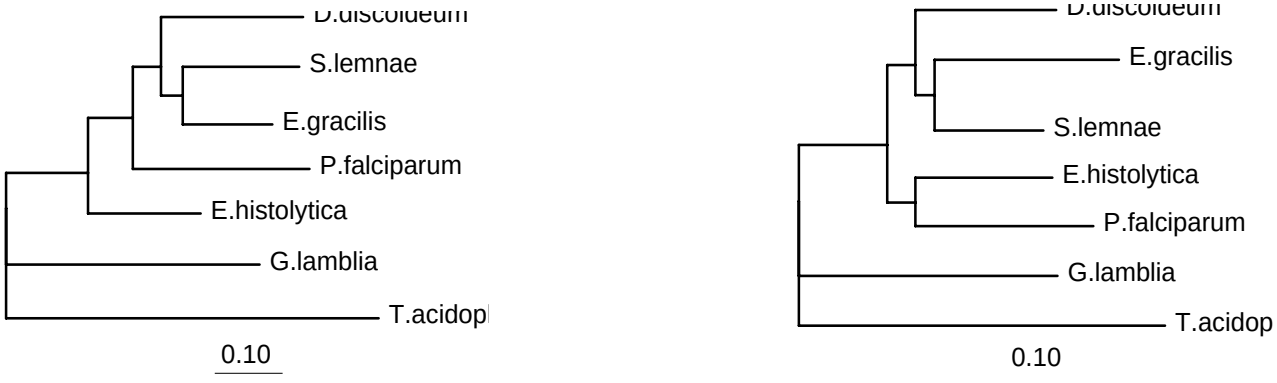
The tree-construction algorithm developed by Felsenstein (so called, stepwise addition) depends on the input order of sequences. To avoid this problem, we adapt rearrangement operations in that algorithm [8]. These rearrangements can move any subtree to a neighboring branch (often called nearest neighbor interchanges).

Adding to the rearrangements, we performed bootstrap (random sampling of sequence sites) [24] and jumble (random sampling of sequences, i.e., random interchanges of input order) sampling. We made 100 boostrap and 100 jumble sampling sets.

Figure 3 shows a phylogenetic tree inferred from ribosomal RNA sequence data. By bootstrap and jumble analyses, this tree was 65% conserved in 200 sampling sets. The scale bar in Figure 3 shows the length corresponding to 0.5 base substitutions per position.

Figure 4 (a) and **(b)** show phylogenetic trees inferred from EF-1α protein sequence data. These trees were also obtained by bootstrap and jumble analyses. The tree (a) and (b) are 30% and 17.5% conserved (60 trees and 35 trees in 200 sampling sets) respectively. Each scale bar in Figure 4 shows the branch length corresponding to 0.1 amino acid substitutions per position.

The results in these figures are well fitted to the result in [7]. In the DNA-based tree



(a) 60 (b) 35

Figure 4: Phylogenetic trees inferred from EF-1α protein sequence data.

(Figure 3), *Entamoeba histolytica*, which lacks mitochondria, is placed at a farther position from the outgroup *Thermoplasma acidophilum* and surrounded by organisms that has mitochondria (*Dictyostelium discoideum*, *Plasmodium falciparum*, *Stylonychia lemnae* and *Euglena gracilis*). This misleading of the position of *Entamoeba histolytica* is considered as the difference of G+C contents [7]. Although the DNA-based tree shown in Figure 3 is only 65% conserved in bootstrap and jumble analyses, it is 97% conserved (194 trees of 200 sets) that the position of *Entamoeba histolytica* is surrounded by organisms that has mitochondria.

On the other hand, it is 84.5% (169 trees) conserved that the position of *Entamoeba histolytica* is placed near organisms that lack mitochondria (*Giardia lamblia* and *Thermoplasma acidophilum*) in the 200 sampling sets of protein-based trees.. Then the organisms that have mitochondria are clearly separated from those lacking them in this tree.

The execution times for constructing a DNA-based tree with fastDNAml [8] and a protein-based tree with our newly developed system are 149.4 and 123.8 seconds, respectively, on Sun SPARCstation 10 model 41.

The reason why the execution time for DNA-based tree is larger than that for a protein-based tree is that the length of ribosomal RNA sequence alignments is much larger that that of EF-1α protein alignments (4071 bases in ribosomal RNA and 454 amino acids in EF-1α) and the number of generated alternative trees for ribosomal RNA is much larger than that for EF-1α due to iterative search in rearrangement step (67 trees for ribosomal RNA and 25 trees for EF-1α).

In order to reduce this computational time, we are going to implement a parallel processing system for protein phylogeny by the method described in Section 3. The target machine we will use is the Intel Touchstone DELTA comprising 512 i860 processors (33 MIPS and 60M flops peak performance per processor).

On the DELTA system, we already implemented an enhanced version of fastDNAml based on the method in Section 3 [18]. Using 16S ribosomal RNA sequence data distributed from the Ribosomal Database Project [21], we constructed the phylogenetic trees of prokaryotic microorganisms (archaebacteria and mycoplasma).

Figure 5 shows speedup in the combination of functional and data parallelisms compared to

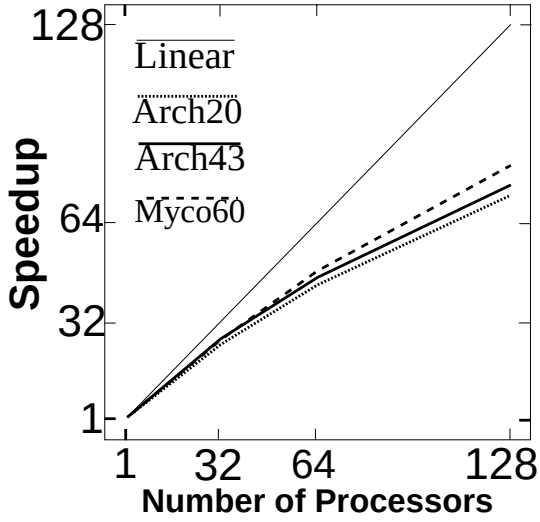


Figure 5: Speedup in the combination of functional and data parallelisms.

sequential execution. In this figure, `Arch20, Arch43,` and `Myco60` denote the results of 20 taxa of archaea, 43 taxa of archaea, and 60 taxa of mycoplasma, respectively. The maximum speedup is about 82 on 128 processors (`Myco60` in Figure 5). We will also measure the performance of protein phylogenetic analysis.

5 Conclusions

Phylogenetic analysis of DNA sequences has played an important role in the study on evolution of life. However recent researches suggest in some cases phylogenetic trees based on the analyses of DNA sequences are misleading and that protein-based trees from amino acid sequences may be more reliable. Therefore, we developed a system for this analysis of protein sequence data based on our previously developed system for the analysis of DNA sequence data.

Since both of these systems use the same tree-construction algorithm except handling different types of sequence data, one can compare a DNA-based tree and a protein-based tree of the same organisms. For example, we constructed phylogenetic trees from both DNA and protein sequence data of an elongation factor (EF-1α) and examined these trees using bootstrap analysis.

The tree-construction algorithm we used is a maximum likelihood method which requires a large amount of computational cost. To reduce the cost, we developed a parallel processing method that combines a functional parallel approach (simultaneous evaluation of possible alternative trees) and a data parallel approach (parallel computation of likelihood). For phylogenetic analysis of DNA sequence data, we achieved a maximum of 82 fold using 128 processors on a massively parallel machine, the Intel Touchstone DELTA.

Acknowledgement

We are grateful to the Concurrent Supercomputing Consortium for access to the Intel Touchstone DELTA System. This work was supported in part by the Japan Ministry of Education, Science and Culture under Grant in Aid for Scientific Research 04235103 and 06249205.

References

[1] Lander, E. S., Langridge, R. and Saccocio, D. M: Mapping and Interpreting Biological Information, *Commun. ACM,* Vol. 34, No. 11, pp.33–39 (1991).

[2] Nei, M.: *Molecular Evolutionary Genetics,* Columbia University Press, New York, Chap.11 (1987).

[3] Weir, B. S.: *Genetic Data Analysis,* Sinauer Associates, Sunderland, Mass. (1990).

[4] Swofford, D. L. and Olsen, G. J.: Phylogeny Reconstruction, In *Molecular Systematics,* ed. Hillis, D. M. and Moritz, C., pp.411–501, Sinauer Associates, Sunderland, Mass. (1990).

[5] Stewart, C.-B.: The Powers and Pitfalls of Parsimony, *Nature,* Vol. 361, No. 6413, pp.603–607 (1993).

[6] Hasegawa, M. and Hashimoto, T.: Ribosomal RNA Trees Misleading?, *Nature*, Vol. 361, p. 23 (1993).

[7] Hasegawa, M., Hashimoto, T., Adachi, J., Iwabe, N. and Miyata, T.: Early Branchings in the Evolution of Eukaryotes: Ancient Divergence of Entamoeba that Lacks Mitochondria Revealed by Protein Sequence Data, *J. of Molecular Evolution,* Vol. 36, pp.380–388 (1993).

[8] Olsen, G. J., Matsuda, H., Hagstrom, R. and Overbeek, R.: fastDNAml: A Tool for Construction of Phylogenetic Trees of DNA Sequences Using Maximum Likelihood, *Computer Applications in Biosciences*, Vol. 10, No. 1, pp.41–48 (1994).

[9] Felsenstein, J.: Evolutionary Trees from DNA Sequences: A Maximum Likelihood Approach, *J. of Molecular Evolution,* Vol. 17, pp.368–376 (1981).

[10] Hasegawa, M., Kishino, H. and Saitou, N.: On the Maximum Likelihood Method in Molecular Phylogenetics, *J. of Molecular Evolution,* Vol. 32, pp.443–445 (1991).

[11] Kishino, H., Miyata, T. and Hasegawa, M.: Maximum Likelihood Inference of Protein Phylogeny and the Origin of Chloroplasts, *J. of Molecular Evolution,* Vol. 31, pp.151–160 (1990).

[12] Adachi, J., Hasegawa, M.: MOLPHY: Programs for Molecular Phylogenetics I – PROTML: Maximum Likelihood Inference of Protein Phylogeny, Computer Science Monographs, No. 27, Institute of Statistical Mathematics, Tokyo (1992).

[13] Felsenstein, J.: *PHYLIP Manual Version 3.4,* University of Washington, Seattle (1991).

[14] Kishino, H. and Hasegawa, M.: Evaluation of the maximum likelihood estimate of the evolutionary tree topologies from DNA sequence data, and the branching order in Hominoidea, *J. of Molecular Evolution* Vol. 29, pp.170–179 (1989).

[15] Dayhoff, M. O., Schwartz, R. M. and Orcutt, B. C.: A Model of Evolutionary Change in Proteins, *Atlas of Protein Sequence and Structure*, ed. Dayhoff, M. O., National Biomedical Research Foundation, Washington DC, Vol. 5, No. 3, pp.345–352 (1978).

[16] Matsuda, H., Olsen, G. J., Hagstrom, R., Overbeek, R. and Kaneda, Y., Implementation of a Parallel Processing System for Inference of Phylogenetic Trees, *Proc. of IEEE Pacific Rim Conference on Communications, Computers and Signal Processing,* pp.280–283 (1993).

[17] Hillis, W. D. and Steele, G. L.: Data Parallel Algorithms, *CACM,* Vol. 29, No. 12, pp.1170–1183 (1986).

[18] Matsuda, H., Olsen, G. J., Overbeek, R. and Kaneda, Y.: Fast Phylogenetic Analysis on a Massively Parallel Machine, Proceedings of 8th ACM International Conference on Supercomputing, pp.297–302 (1994).

[19] Butler, R and Lusk, E.: User's Guide to the p4 Programming System, Tech. Report ANL-92/17, Mathematics and Computer Science Division, Argonne National Laboratory (1992).

[20] Iwabe, N., Kuma, K., Hasegawa, M., Osawa, S. and Miyata, T.: Evolutionary Relationship of Archaebacteria, Eubacteria, and Eukaryotes Inferred from Phylogenetic Trees of Duplicated Genes, *Proc. Nat'l Acad. Sci.,* USA, Vol. 86, pp.9355–9359 (1989).

[21] Larsen, N., Olsen, G. J., Maidak, B. L., McCaughey, M. J., Overbeek, R., Macke, T. J., Marsh, T. L. and Woese,C. R.: The Ribosomal Database Project, *Nucleic Acids Research,* Vol. 21, No. 13, pp.3021–3023 (1993).

[22] Entrez User's Guide, National Center for Biotechnology Information, National Institutes of Health (1993) (On-line document at ncbi.nlm.nih.gov).

[23] Higgins, D. G., Bleasby, A. J. and Fuchs, R.: Clustal V: Improved Software for Multiple Sequence Alignment, Computer Applications in Biosciences, Vol. 8, No. 2, pp.189–191 (1992).

[24] Felsenstein, J.: Confidence Limits on Phylogenies: An Approach Using The Bootstrap, *Evolution,* Vol.39, No.4, pp.783–791 (1985).