Construction of Phylogenetic Trees from Amino Acid Sequences using a Genetic Algorithm

Hideo Matsuda
matsuda@ics.es.osaka-u.ac.jp

Department of Information and Computer Sciences,
Faculty of Engineering Science, Osaka University
1-3 Machikaneyama, Toyonaka, Osaka 560 Japan

Abstract

We have developed a novel algorithm to search for the maximum likelihood tree constructed from amino acid sequences. This algorithm is a variant of genetic algorithms which uses scores derived from the log-likelihood of trees computed by the maximum likelihood method. This algorithm is valuable since it may construct more likely tree from randomly generated trees by utilizing crossover and mutation operators. In a test of our algorithm on a data set of elongation factor-1 α sequences, we found that the performance of our algorithm is comparable to that of other tree-construction methods (UPGMA, the neighbor-joining and the maximum parsimony methods; and the maximum likelihood method with different search algorithms).

1 Introduction

Construction of phylogenetic trees is one of the most important problems in evolutionary study. The basic principle of tree construction is to infer the evolutionary process of taxa (biological entities such as genes, proteins, individuals, populations, species, or higher taxonomic units) from their molecular sequence data [1, 2].

A number of methods have been proposed for constructing phylogenetic trees. These methods can be divided into two types in terms of the type of data they use; distance matrix methods and character-state methods [3]. A distance matrix consists of a set of $n(n-1)/2$

松田秀雄：大阪大学基礎工学部情報工学科，〒560 大阪府豊中市待兼山町 1-3

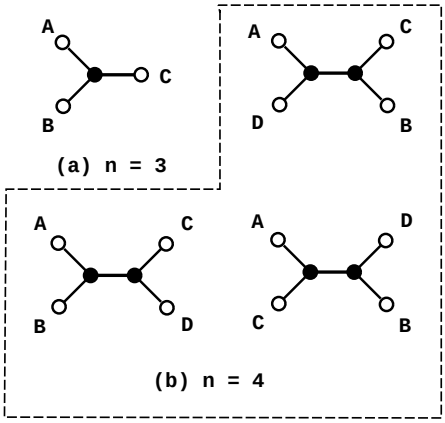


Figure 1: Phylogenetic trees expressed as unrooted trees.

distance values for n taxa, whereas an array of character states (e.g. nucleotides of DNA sequences, residues of amino acid sequences, etc.) is used for the character-state methods. The unweighted pair-group method with arithmetic mean (UPGMA) [4] and the neighbor-joining (NJ) method [5] belong to the former, whereas the maximum parsimony (MP) method [6] and the maximum likelihood (ML) method [7] belong to the latter.

In these methods, the ML method tries to make explicit and efficient use of all character-states based on stochastic models of those data (e.g. DNA base substitution models, amino acid substitution models, etc.) Also it gives the likelihood values of possible alternative trees, high resolution scores for comparing the confidence of those trees.

Recent research [8, 9] suggests that phylogenetic trees based on the analyses of DNA sequences may be misleading – especially when G+C content differs widely among lineages – and that protein-based trees from amino acid sequences may be more reliable. Adachi and Hasegawa develops their MOLPHY PROTML program [10] for constructing phylogenetic trees from amino acid sequences using the ML method.

We also have implemented a system to construct phylogenetic trees from amino acid sequences using the ML method. This implementation based on our previous program, *fastDNAml* [11], which is a speedup version of Felsenstein's PHYLIP DNAML [12] (a program for constructing trees from DNA sequences). The difference between MOLPHY PROTML and our program is their search algorithms for finding the ML tree. MOLPHY PROTML employs the star-decomposition method which can be seen as the extension of the search algorithm in the NJ method to the ML method. Whereas our system employs two algorithms: *stepwise addition*, the same algorithm [13] as fastDNAml and PHYLIP DNAML; and *a genetic algorithm*, a novel search algorithm discussed later.

2 Maximum Likelihood Method

In the ML method, 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 giving rise to given amino acid sequences. 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 amino acid residues. In order to compare evolutionarily related parts of taxa, several gaps are inserted corresponding to the insertion or deletion of residues. Consequently, an evolutionary change can be described by a substitution of a residue (or a gap) at a position of a sequence with a residue at the same position of another sequence.

The probability of a tree is computed on the basis of a stochastic model (Markov chain model of order one) on residue substitutions in an evolutionary process. The model assumes a residue substitution at a sequence position takes place independently of substitution at other positions.

The substitution probabilities are computed by an amino acid substitution model [14] based on an empirical substitution matrix newly compiled by Jones, et al. [15] as well as a widely used matrix compiled by Dayhoff, et al. [16].

One can build a tree with 4 tips from a tree with 3 tips by adding one more sequence in all possible locations. Then one can build a tree with 5 tips by adding another sequence to the most likely tree with 4 tips. In general, one can build a tree with i tips from a tree with $i-1$ tips, 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 at each step.

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, the number of bifurcating unrooted trees is,

$$\prod_{i=3}^{n}(2i-5) = \frac{(2n-5)!}{2^{n-3}(n-3)!} \tag{1}$$

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.

3 Search Algorithm for Optimal Tree

Felsenstein develops a search algorithm (so-called *stepwise addition*) in his DNAML program [12]. It performs successive tree expansion by iterating steps constructing a tree with i tips from a tree with $i-1$ tips until all n taxa have been added. Each step is carried out by putting i-th taxon on one of $2i-5$ branches in a tree with $i-1$ tips. For each step, only the best tree is retained.

At the end of each step, a partial tree check is performed to see whether minor rearrangements lead to a better tree. By default, these rearrangements can move any subtree to a neighboring branch (so-called *branch exchange*). Figure 2 shows an example of branch exchange. This operation is done by swapping two subtrees connected to an intermediate edge (an edge which is not incident with leaves). Two possible exchanges exist for each intermediate

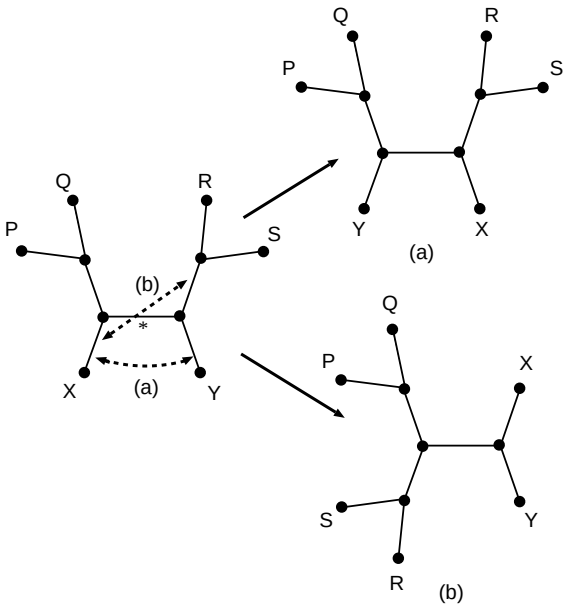


Figure 2: Branch exchange in a phylogenetic tree.

edge (such as (a) and (b) in Figure 2). This check is repeated until none of the alternatives tested is better than the starting tree [17]. The resulting tree from the stepwise addition algorithm generally depends on the order of the input taxa even though the branch exchanges are carried out at the end of each step. Hence, Felsenstein recommends performing a number of runs with different orderings of the input taxa.

Adachi and Hasegawa develop another algorithm in their MOLPHY PROTML [10] called *star decomposition.* This is similar to the algorithm employed in the NJ method using a distance matrix [5]. It starts with a star-like tree. Decomposing the star-like tree step by step, one can finally obtain an unrooted phylogenetic tree if all multifurcations can be resolved with statistical confidence. Since the information from all of the taxa under analysis is used from the beginning, the inference of the tree is likely to be stable by this procedure.

We have developed a novel algorithm based on the simple genetic algorithm [18]. The difference between our algorithm and the simple genetic algorithm is the encoding scheme (not binary strings but direct graph representation) and novel crossover and mutation operators described below. At the initial stage (i.e. the first generation), it randomly generates (a fixed number of) possible alternative trees, reproduces a fixed number of trees selected by the roulette selection proportional to their fitness values (we compute the fitness by subtracting the worst log-likelihood at the generation from each log-likelihood because log-likelihood is usually negative), then tries to improve them by using crossover and mutation operators generation by generation. Since we fix the number of trees as a constant for each generation, the tree with the best score could be removed by these operators. Thus our algorithm checks to make sure that the tree with the best score survives for each generation (i.e. elitest selection).

By the existence of the crossover operator, our algorithm is unique from the other algorithms since it generates a new tree combining any two of already generated trees. We assume that

each randomly generated tree has a good portion which contributes to improve its likelihood. If one can extract different good portions from two trees, one may construct a more likely tree which include both of those good portions. In general, it is difficult to identify the good portion of a tree. Thus we introduce a crossover operation as an approximate algorithm to do this as follows (see Figure 3). This crossover operation is based on the minimum evolution principle which is originally proposed by Cavalli-Sforza and Edwards [19] and extensively studied by Saitou and Imanishi [20].

[CROSS 1] Pick up any two of randomly generated trees (say, tree i and tree j). Compute their branch lengths so that their likelihood values are maximized, then construct a distance matrix among leaf nodes (taxa) of each tree.

[CROSS 2] By comparing two distance matrices of tree i and tree j, compute the relative difference which is defined as:

$$r_{ij}(x,y) = \frac{|d_i(x,y) - d_j(x,y)|}{d_i(x,y) + d_j(x,y)}, \tag{2}$$

where $d_i(x,y)$ denotes the distance between taxa x and y in tree i, whereas $d_j(x,y)$ denotes the distance between taxa x and y in tree j.

[CROSS 3] Find a pair of taxa (x_0, y_0) which gives the maximum of relative differences (i.e. $r_{ij}(x_0,y_0) = \max_{x,y}(r_{ij}(x,y))$).
If $d_i(x_0,y_0) < d_j(x_0,y_0)$, we regard tree i as having a relatively good portion (a minimum subtree which includes taxa x_0 and y_0) compared to tree j. Here we name tree i a reference tree and tree j a rugged tree.
If $d_i(x_0,y_0) > d_j(x_0,y_0)$, tree i is named a rugged tree, whereas tree j is named a reference tree.

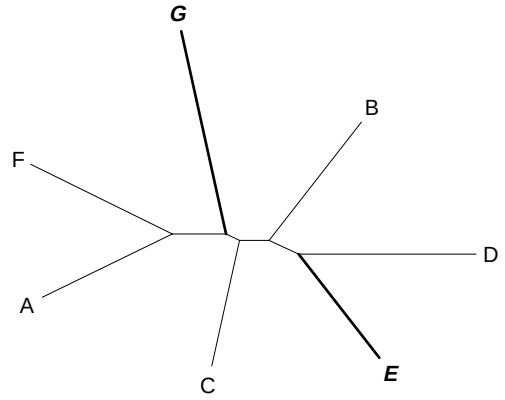
[CROSS 4] Merge those two trees into one. This procedure is carried out by three steps.

(1) From the reference tree, pick up a merge point m which is not included in a minimum subtree containing taxa x_0 and y_0. Keeping m and a minimum subtree containing taxa x_0 and y_0, remove all other taxa from the reference tree (say s for this result).

(2) From the rugged tree, remove all taxa which appear in s. If m appears in s, m is not removed from the rugged tree (say t for this result).

(3) merge s and t by overlapping m in s and t.

In Figure 3, (a) and (b) show randomly generated trees of seven taxa and their distance matrices based on the branch lengths computed by the ML method. In these trees, the pair of taxa (E,G) gives the maximum of relative differences (squarely covered in their distance matrices). Since $d_{(b)}(E,G)$ is less than $d_{(a)}(E,G)$, tree (a) is named a rugged tree and tree (b) is named a reference tree. Figure 3 (c) shows a tree obtained by the crossover operator. Here merge point m, subtree s and subtree t in **[CROSS 4]** correspond to taxon A, subtree $(A,(E,G))$ (described in the New Hampshire format used in PHYLIP) and subtree $(A,(F,(C,(B,D))))$, respectively.

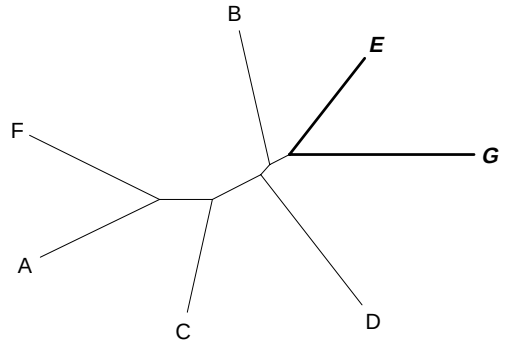


Distance Matrix

	A	B	C	D	E	F
A						
B	0.4548					
C	0.3656	0.3585				
D	0.6495	0.6053	0.5532			
E	0.4119	0.3676	0.3155	0.5232		
F	0.4435	0.5138	0.4246	0.7085	0.4709	
G	0.8621	0.8709	0.7817	1.0656	0.8280	0.9211

Ln Likelihood = -4470.4

(a) a rugged tree

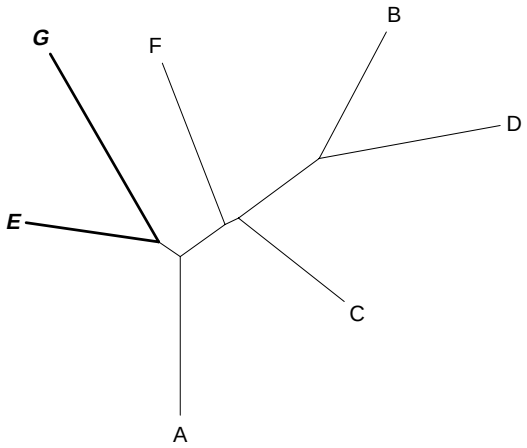


Distance Matrix

	A	B	C	D	E	F
A						
B	0.4673					
C	0.3543	0.3687				
D	0.6467	0.5936	0.5481			
E	0.4243	0.3621	0.3258	0.5507		
F	0.4431	0.5241	0.4111	0.7035	0.4811	
G	0.9037	0.8414	0.8051	1.0300	0.7758	0.9604

Ln Likelihood = -4468.9

(b) a reference tree

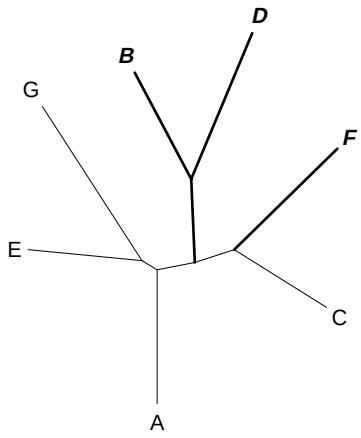


Distance Matrix

	A	B	C	D	E	F
A						
B	0.4565					
C	0.3672	0.3559				
D	0.6593	0.5096	0.5587			
E	0.3579	0.4044	0.3151	0.6072		
F	0.5048	0.4936	0.4043	0.6964	0.4527	
G	0.8520	0.8985	0.8092	1.1013	0.7672	0.9468

Ln Likelihood = -4461.9

(c) a tree constructed by the crossover operator



Distance Matrix

	A	B	C	D	E	F
A						
B	0.4443					
C	0.3743	0.3624				
D	0.6481	0.5070	0.5662			
E	0.3556	0.3924	0.3225	0.5963		
F	0.5135	0.5016	0.3797	0.7055	0.4617	
G	0.8467	0.8835	0.8135	1.0873	0.7652	0.9528

Ln Likelihood = -4457.9

(d) a tree constructed by the mutation operator

Figure 3: Phylogenetic trees constructed by generic algorithm operators.

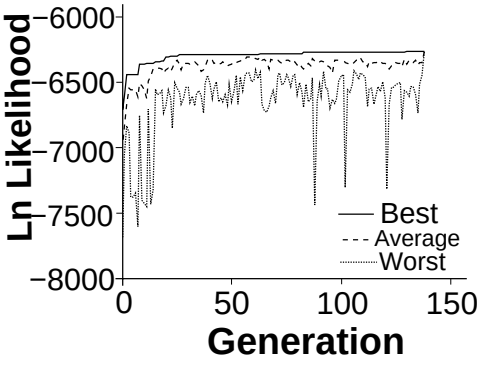
In genetic algorithms, a mutation operator is used to avoid being trapped to local optima. In our algorithm, we employed branch exchange as a mutation operator. However, it is not always guaranteed that the operator works to avoid being trapped to local optima. Figure 3 (d) shows a tree obtained by the mutation operator from (c) (taxon F and subtree (B, D) are exchanged).

4 Preliminary Performance Result

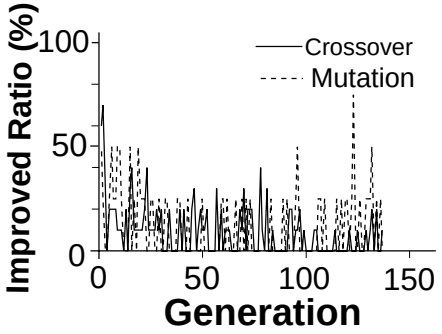
To measure the performance of our method, we used amino acid sequences of elongation factor 1 α (EF-1α). EF-1α is useful protein in tracing the early evolution of life [21] since it can be seen in all organisms and the substitution rate of its sequence is relatively slow; for example, more than 50% identity is retained between eukaryotic and archaebacterial sequences [9].

The EF-1α sequences we used are as follows: an archaebacterium *Thermoplasma acidophilum* (EMBL Accession No. X53866) and 14 eukaryotes *Arabidopsis thaliana* (EMBL Accession No. X16430), *Candida albicans* (GenBank Accession No. M29934), *Dictyostelium discoideum* (EMBL Accession No. X55972 and X55973), *Entamoeba histolytica* (GenBank Accession No. M92073), *Euglena gracilis* (EMBL Accession No. X16890), *Giardia lamblia* (DDBJ Accession No. D14342), *Homo Sapiens* (EMBL Accession No. X03558), *Lycopersicon esculentum* (EMBL Accession No. X14449), *Mus musculus* (EMBL Accession No. X13661), *Plasmodium falciparum* (EMBL Accession No. X60488), *Rattus norvegicus* (EMBL Accession No. X61043), *Saccharomyces cerevisiae* (EMBL Accession No. X00779), *Stylonychia lemnae* (EMBL Accession No. X57926) and *Xenopus laevis* (GenBank Accession No. M25504). The archaebacterium was used as an outgroup of the other organisms. We made the alignment of these sequences using CLUSTAL W [22].

Figure 4 (a) shows the best, average and worst likelihood scores observed at the end of each generation (just before roulette selection of the next generation) in the case of EF-1α tree. We set the population size of each generation to 20 trees and crossover and mutation ratios to 0.5 and 0.2 (i.e. 10 and 4 trees), respectively. At the 130th generation, the best score reached to -6260.7. Then at the 138th generation, all 20 trees were converged to the tree with the score.



(a) The improvement of log likelihood scores



(b) The improvement ratios by crossover and mutation operators

Figure 4: A performance result on constructing phylogenetic trees of 15 EF-1α sequences.

Table 1: Comparison of tree-construction methods based on the log-likelihood scores of the resulting trees.

Method	Log-likelihood score
UPGMA	-6301.2
NJ	-6301.8
MP	-6267.7 .. -6270.3
SD	-6309.6
SW/BE	-6260.6 .. -6998.8
GA	-6260.7

Table 1 shows the log-likelihood scores obtained by UPGMA, NJ and MP methods described in Section 1 and different search algorithms in the ML method (SD denotes star decomposition, SW/BE denotes stepwise addition with branch exchange and GA denotes genetic algorithm) described in Section 3.

For the measurement in Table 1, we used PHYLIP PROTDIST (computing distance matrix) and NEIGHBOR (constructing trees) for UPGMA and NJ, PHYLIP PROTPARS for MP. Then we computed the log-likelihood scores of those phylogenetic trees. The reason why the score of the MP method ranges from -6267.7 to -6270.3 is that it constructed five equally parsimonious trees (i.e. there are five trees which have the same minimum substitution count).

For different search algorithms in the ML method, we used MOLPHY PROTML for SD and our system for SW/BE and GA. Since the resulting tree from SW/BE depends on the order of the input taxa, we generated 20 randomly-shuffled orders of input taxa then construct trees using the 20 sets of sequence data. We got 16 different trees of which scores ranging from -6260.6 to -6998.8.

Although the ML method with our genetic algorithm could not reach the global maximum (but our best score -6260.7 is very close to the best of the other algorithms -6260.6), the performance of our method can be comparable to the other widely used methods. Moreover, unlike the other methods, our algorithm may be able to improve the score by increasing its population size or tuning of crossover and mutation ratios.

Figure 4 (b) shows improved ratios by the crossover and the mutation operators for each generation. For example, at the 10th generation, the ratio by the crossover operator is 10% (i.e. 10 crossover operations yielded 1 more likely trees than both of the rugged and reference trees they used), whereas the ratio by the mutation operator is 50% (i.e. 4 mutation operations yielded 2 more likely trees than the trees they used).

5 Conclusion

We developed a novel algorithm to search for the maximum likelihood tree constructed from amino acid sequences. This algorithm is a variant of genetic algorithms which uses log-likelihood scores of trees computed by the ML method. This algorithm is especially valuable since it can construct more likely tree from given trees by using crossover and mutation operators.

From a preliminary result by constructing phylogenetic trees of EF-1α sequences, we found that the performance of the ML method with our algorithm can comparable to those of the other tree-construction methods (UPGMA, NJ, MP and ML with different search algorithms). As a future work, we will develop a method to determine the parameters of our genetic algorithm (population size, the number of generations, crossover and mutation rates, etc.) which now user should decide empirically.

Acknowledgements

This work was supported in part by a Grant-in-Aid (07249203) for Scientific Research on Priority Areas from The Ministry of Education, Science, Sports and Culture in Japan.

References

[1] Nei, M., *Molecular Evolutionary Genetics,* Columbia University Press, New York, Chap.11, 1987.

[2] 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.

[3] Saitou, N., "Statistical Methods for Phylogenetic Tree Reconstruction," In *Handbook of Statistics,* eds. Rao, R. and Chakraborty, R., Vol.8, pp.317–346, Elsevier Science Publishers, 1991.

[4] Sokal, R. R. and Michener, C. D., "A Statistical Method for Evaluating Systematic Relationships," University of Kansas Sci. Bull. Vol.28, pp.1409–1438, 1958.

[5] Saitou, N. and Nei, M., "The Neighbor-Joining Method: A New Method for Reconstructing Phylogenetic Trees," Mol. Biol. Evol., Vol.4, pp.406–425, 1987.

[6] Eck, R. V. and Dayhoff, M. O., In *Atlas of Protein Sequence and Structure,* ed. Dayhoff, M. O., National Biomedical Research Foundation, Silver Springs, 1966.

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

[8] Hasegawa, M. and Hashimoto, T., "Ribosomal RNA Trees Misleading?," *Nature*, Vol. 361, p. 23, 1993.

[9] 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.

[10] 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.

[11] 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.

[12] Felsenstein, J., *PHYLIP (Phylogeny Inference Package),* (accessible by WWW at `http://evolution.genetics.washington.edu/phylip.html`), 1995.

[13] Matsuda, H., Yamashita, H. and Kaneda, Y., "Molecular Phylogenetic Analysis using both DNA and Amino Acid Sequence Data and Its Parallelization," *Proceedings of Genome Informatics Workshop V,* pp.120–129, Universal Academy Press, 1994.

[14] 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.

[15] Jones, D. T., Taylor, W. R. and Thornton, J. M., "The rapid generation of mutation data matrices from protein sequences," *Computer Applications in Biosciences,* Vol.8, pp.275–282, 1992.

[16] 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.

[17] Felsenstein, J., "Phylogenies from restriction sites: a maximum-likelihood approach," *Evolution,* Vol.46, pp.159-173, 1992.

[18] Goldberg, D. E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addisson-Wesley Publishing Company, Inc., 1989.

[19] Cavalli-Sforza, L. L. and Edwards, A. W. F., "Phylogenetic analysis: models and estimation procedures," *Am. J. Hum. Genet.,* Vol.19, pp.233–257, 1967.

[20] Saitou, N. and Imanishi, T., "Relative Efficiencies of the Fitch-Margoliash, Maximum-Parsimony, Maximum-Likelihood, Minimum-Evolution, and Neighbor-joining Methods of Phylogenetic Tree Construction in Obtaining the Correct Tree," *Molecular Biology and Evolution,* Vol.6, No.5, pp.514–525, 1989.

[21] 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.

[22] Thompson, J. D., Higgins, D. G. and Gibson, T. J., "CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice," Nucleic Acids Research, Vol.22, pp.4673–4680, 1994.