A GLOBAL MAXIMUM LIKELIHOOD SUPER-QUARTET PHYLOGENY METHOD

P. WANG, B. B. ZHOU, M. TARAENEH, D. CHU, C. WANG and A. Y. ZOMAYA

School of Information Technologies, University of Sydney
NSW 2006, Australia

R. P. BRENT

Mathematical Science Institute, Australian National University
Canberra, ACT 0200, Australia

Extending the idea of our previous algorithm [17, 18] we developed a new sequential quartet-based phylogenetic tree construction method. This new algorithm reconstructs the phylogenetic tree iteratively by examining at each merge step every possible super-quartet which is formed by four subtrees instead of simple quartet in our previous algorithm. Because our new algorithm evaluates super-quartet trees, each of which may consist of more than four molecular sequences, it can effectively alleviate a traditional, but important problem of quartet errors encountered in the quartet-based methods. Experiment results show that our newly proposed algorithm is capable of achieving very high accuracy and solid consistency in reconstructing the phylogenetic trees on different sets of synthetic DNA data under various evolution circumstances.

1 Introduction

For systematic biology, evolutionary history is one of the most important topics. Therefore reconstruction of phylogenetic trees from molecular sequences has strong research significance. The quartet-based approach is one of the primary methods for phylogeny reconstruction. The basic idea is to construct a tree based on the topological properties of a set of four molecular sequences (or quartets). The advantages of the quartet-based method are that theoretically it guarantees a one-to-one correspondence between a tree topology and a set of quartet trees and if the tree topology for each individual quartet can be correctly identified, the entire evolutionary tree for a given problem can be reconstructed in polynomial time. The main disadvantage, however, is that it can be very difficult to obtain correctly resolved quartet trees using any existing methods [1, 8]. This quartet error problem greatly hinders the quartet-based approach from a wide application.

Previously we developed a quartet-based algorithm for the reconstruction of evolutionary trees [17, 18, 19]. Instead of constructing only one tree as output, this algorithm constructs a limited number of trees for a given set of DNA or protein sequences. Experimental results showed that the probability for the correct phylogenetic tree to be included in this small number of trees is very high. When we selected just a few best ones (say three) from these trees under maximum likelihood (ML) criterion, extensive tests using synthetic test data sets showed that the algorithm outperforms many known algorithms for phylogeny reconstruction in terms of tree topology [19]. One problem associated with the original form of this algorithm is that under certain circumstances it does not perform well when reconstructing only a single tree as output without the selection stage using ML. Though the method can tolerate quartet errors better than many other quartet-based algorithms, the quality of the generated tree still depends heavily on the quality of quartet trees.

In this paper we introduce a new algorithm. This algorithm is similar to our previous quartet-based algorithm. However, it reconstructs a tree based on an idea of taking more than four taxa into quartet topological estimates, which is called “super-quartet” in this paper. The main reason of using super-quartets is to alleviate the problem of quartet errors. Inherited from previous algorithm, this new algorithm maintains the theoretical advantage of one-to-one mapping of the tree and corresponding (super-) quartet weights. The main difference is that this new method iteratively merges taxa on super-quartet weights calculated by log ML values (with a probabilistic normalization). Since a super-quartet consists of more than four molecular sequences, it is expected that the quality of super-quartet weights are higher than that of basic quartet (four sequences), hence improving the accuracy of the generated tree. Our experiments confirm this and the results demonstrate that this new algorithm is able to achieve better accuracy, when reconstructing only a single tree, than many ML-based algorithms including the very popular PHYML [7] in the terms of phylogenetic tree topology.

The paper first reviews our previous algorithm and its associated problem in reconstructing of a single tree in Section 2. The super-quartet idea and our new phylogenetic reconstruction algorithm are then introduced in Section 3. We present some experiment results in Section 4. Finally the conclusions and the future work are discussed in Section 5.

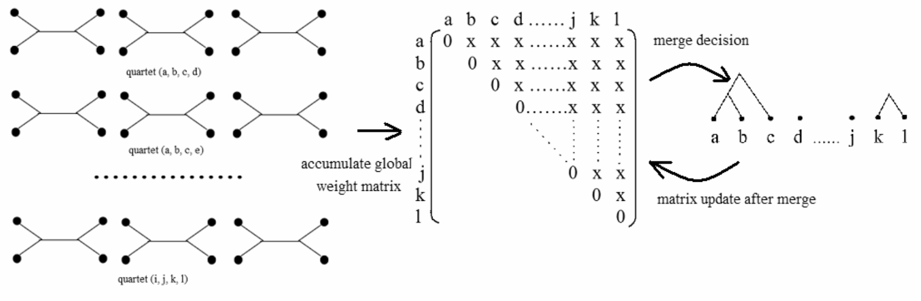


Figure 1. The three-stage procedure of previous algorithm

2 The Previous Algorithm and Problem

Our previous quartet-based algorithm rebuilds the evolutionary tree using quartet weights [17, 18]. As shown in Figure 1, the algorithm is a three-stage procedure: 1. generate all the quartets and calculate the quartet weights; 2. accumulate weights to a global quartet weight matrix; 3. iteratively merge subtrees using this matrix. The idea of quartet weights was first introduced in [5] and then extended and used in a tree-puzzling algorithm [16]. Each quartet is associated with three topologies and their normalized weights. In the ideal case when all quartets are correctly and fully resolved, there is a one-to-one correspondence between this matrix and the true tree.

In order to overcome the quartet inconsistency problem, the previous algorithm also deploys two additional mechanisms: the first is to generate more than one phylogenetic tree (a small number) when merging ambiguity occurs; the other is to update the corresponding global quartet weights to theoretical values. Theoretical values are obtained by assuming the two subtrees that have been merged in a heuristic merging step are real neighbors in the true tree. Previous experimental results [18] demonstrated that the probability for the correct tree to be included in the small set of generated trees is very high. As shown in Figure 2, the algorithm is able to achieve better results than PHYML when a limited number of trees are constructed. (The reason we use PHYML as benchmark to compare our method is because it is one of the most popular packages used by biologists, and its accuracy in building the tree is among the highest several methods up to now.) However, PHYML achieves better accuracy than our previous algorithm when the algorithm is limited to generate only a single tree. Though our previous algorithm is able to tolerate the quartet errors better than other quartet-based algorithms, the schemes used are still not sufficiently good enough to compensate the quartet error problem. It is necessary to find more vigorous mechanisms to deal with the quartet error problem.

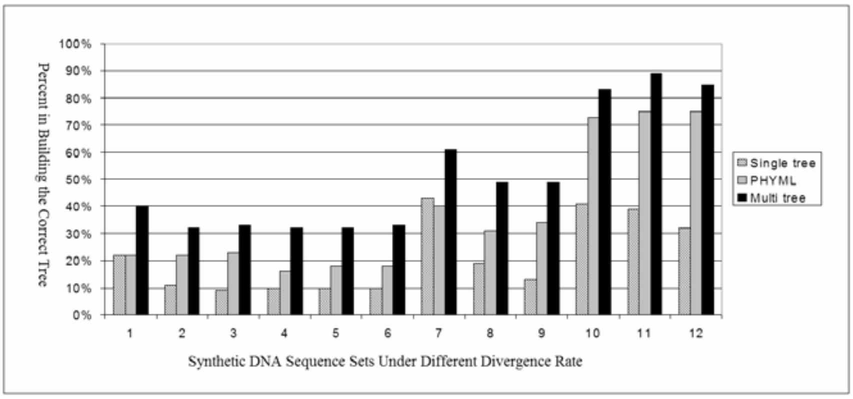


Figure 2. Experimental comparison of PHYML and previous algorithm (both on constructing single tree and multiple 5 trees) under the same condition as in experiment section.

3 The New Super-Quartet Algorithm

Phylogeny inference for only four taxa is often considered hard and the result unreliable. This is because sampling of taxa prior to phylogenetic tree reconstruction strongly influences the accuracy. If one seeks to establish the phylogenetic relationship between four groups of taxa by using a single representative for each of these groupings, the result generally depends on which representatives are selected [1, 13]. Although our previous algorithm builds the tree on a global view of quartet relationship, it is still limited to examination of the topological relationship on the basis of four taxa only.

The basic idea of quartet is to represent the topological relation of four taxa, through three possible binary (quartet) trees. In the extension of this idea we place a subtree rather than a single taxon on each vertex of a quartet binary tree, as shown in Figure 3. The weight of a super-quartet is measured using the same procedure for quartet weight calculation [9], by firstly calculating the maximum likelihood values for each possible super-quartet tree out of three possible topologies and then transforming these likelihood values into super-quartet weights. Bayes theorem with a uniform prior for all three possible trees is used for such transformation. Since each vertex may contain more than one taxon, we expect the super-quartet weights are more reliable than weights of simple quartets.

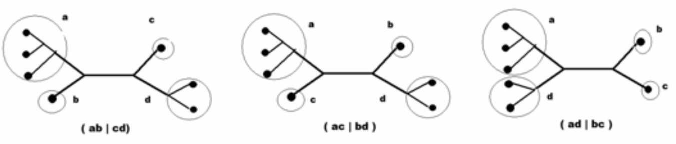


Figure 3. super-quartet trees for taxon sets of {a, b, c, d}.

After the weights for all possible super-quartets are calculated, the super-quartet weight matrix can be generated the same way as the simple quartet weight matrix [17, 18, 19] with each row or column corresponding to a subtree rather than a single taxon. Our new super-quartet method shares the same theoretical property of one-to-one tree topology and matrix mapping [17, 18].

The new algorithm is also an iterative merge algorithm; it makes decision on which two subtrees are going to be merged by selecting the pair of subtrees which shows highest probability by global super-quartet weight matrix. The entry values in the weight matrix are the agglomerated normalization weights for corresponding subtrees at a particular merge step. The metric of making the merge decision is to evaluate how close the super-quartet weight is to the theoretical value. This is easily implemented by:

$$C_{\mathrm{ij}} = M_{ij} / T_{ij} \ , \tag{1}$$

where C_{ij} is called “confidence value”, and T_{ij} is the theoretical value for the current subtrees to be merged and can be calculated by:

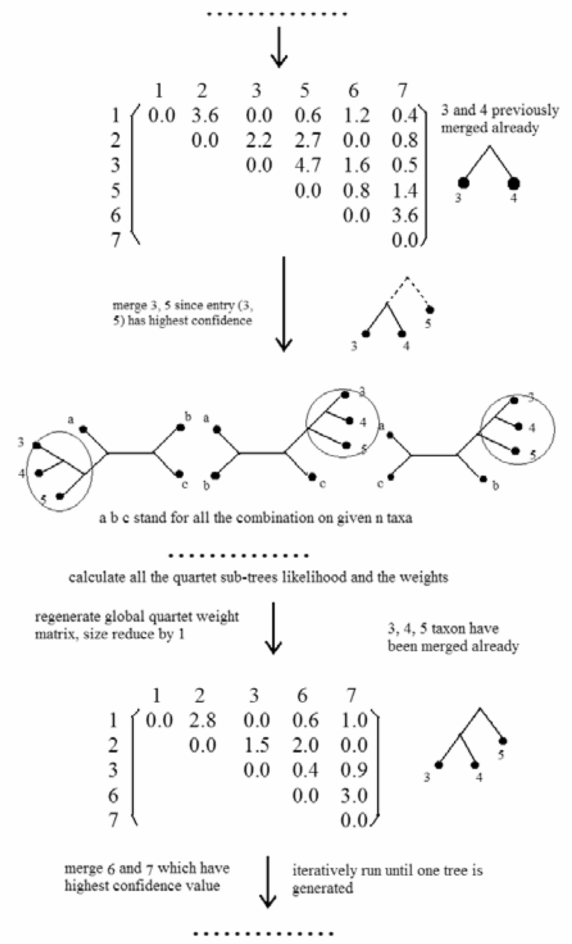
$$T_{ij} = \binom{n_k - 2}{2}. \tag{2}$$

where n_k represents the current global super-quartet weight matrix dimension.

The structure of our super-quartet algorithm is given below:

1. set $n_k = n$. (Initially every taxon represents itself as a subtree)
2. number the subtrees from 1 to n_k .

3. calculate the likelihood values of all possible super-quartet trees and the associated weights.
4. update the weight matrix. (Each subtree represented by one number corresponding to a particular row or column.)
5. for each pair of subtrees calculate the confidence value (using the entry value against a desired one) to determine how likely the pair is to be merged directly.
6. choose the pair of subtrees that has the highest confidence value and merge them into a bigger subtree.
7. reduce n_k by one.
8. if n_k >3, go back to *step 2*; otherwise merge the remaining three subtrees into one final tree.



$$\begin{array}{c|cccccc} & 1 & 2 & 3 & 5 & 6 & 7 \\ \hline 1 & 0.0 & 3.6 & 0.0 & 0.6 & 1.2 & 0.4 \\ 2 & & 0.0 & 2.2 & 2.7 & 0.0 & 0.8 \\ 3 & & & 0.0 & 4.7 & 1.6 & 0.5 \\ 5 & & & & 0.0 & 0.8 & 1.4 \\ 6 & & & & & 0.0 & 3.6 \\ 7 & & & & & & 0.0 \end{array}$$

$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 6 & 7 \\ \hline 1 & 0.0 & 2.8 & 0.0 & 0.6 & 1.0 \\ 2 & & 0.0 & 1.5 & 2.0 & 0.0 \\ 3 & & & 0.0 & 0.4 & 0.9 \\ 6 & & & & 0.0 & 3.0 \\ 7 & & & & & 0.0 \end{array}$$

....
3 and 4 previously merged already
3
4
merge 3, 5 since entry (3, 5) has highest confidence
3
4
5
a
b
c
a b c stand for all the combination on given n taxa
....
calculate all the quartet sub-trees likelihood and the weights
regenerate global quartet weight matrix, size reduce by 1
3, 4, 5 taxon have been merged already
3
4
5
merge 6 and 7 which have highest confidence value
iteratively run until one tree is generated
....

Figure 4. The algorithm iterative step procedure illustration

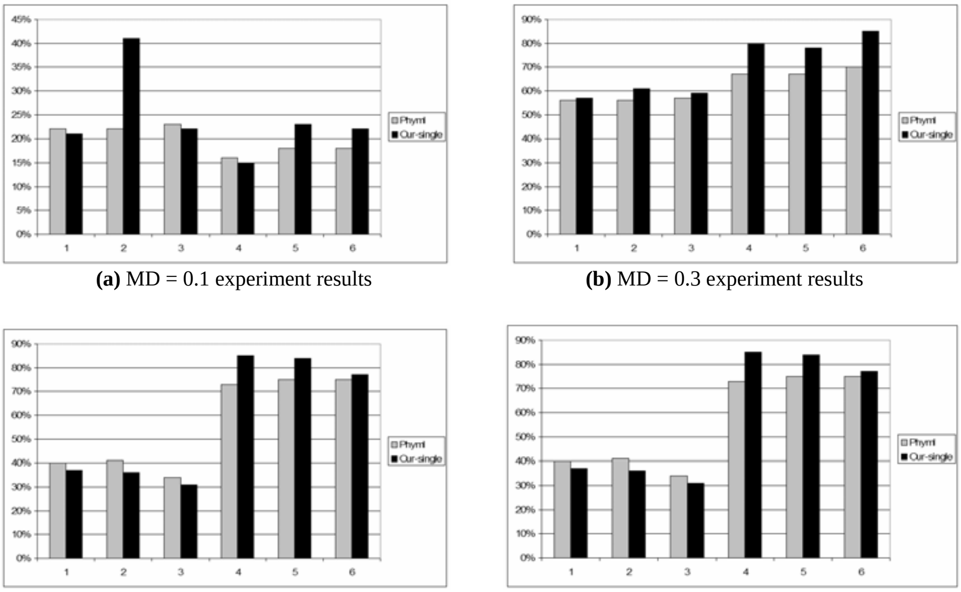
A simple example of one merge step using our algorithm is illustrated in Figure 4. In this example we assume the taxon 3 and 4 have been merged in previous step, and the current global super-quartet weight matrix is of size 6x6 since previous merging step reduced the matrix size from the original 7x7. The algorithm calculates the confidence value and find that subtree pair 3 and 5 has the highest value. These two subtrees are merged into one larger subtree. The algorithm then re-calculates the likelihood values of super-quartet trees and the size of the global super-quartet weight matrix is reduced to 5x5. This heuristic merge process continues until the whole tree is constructed.

4 Experiments

The experiments are carried out using the synthetic data sets from *Montpellier Laboratory of Computer Science, Robotics, and Microelectronics (www.lirmm.fr)*. The data sets, each consists of 12 taxa, are generated using six model trees. Three model trees are molecular clock-like, while the other three present varying substitution rates among lineages. With these model trees under various evolutionary conditions, the test data sets of DNA sequences, each being of length 300, are generated using Seq-Gen. The reason we select these synthetic data as benchmark is because for real DNA data it is very hard to clarify the correctness of the constructed tree. On the other hand, these synthetic data sets are very comprehensive in evolutionary conditions, both with molecular clock and without, both balanced and unbalanced.

We present in Figure 5 our experimental results and compare them with those obtained using PHYML in terms of the percentage of correctly constructed phylogenetic trees. For our algorithm, HKY model as evolutionary model, transition to transversion rate of one, nucleotide frequency all at 25%, uniform model of rate heterogeneity are selected as parameters to perform the experiments. We run PHYML (version 2.44) on the same data set using the same parameters to compare the results. The trees generated from both algorithms are compared with the true trees using the Robinson and Foulds topological distance (RF) method [14].

The results demonstrate that our new global ML super-quartet algorithm outperforms PHYML in most circumstances. First of all, our algorithm made great improvement for data sets without molecular clock: our method constructs the correct tree at much higher frequency than PHYML. This can be seen clearly from Figure 5 (b) and (c) for the experiment data sets without molecular clock (right 3 columns in the figure). These sets of data represent the most common evolution circumstances for phylogenetic study and the percentage of correctly constructed trees using our algorithm is nearly 15% higher than that of PHYML. Secondly on data set with large variation (MD around 2.0) with or without molecular clock, our algorithm nearly doubles the percentage of constructing the correct tree compared with PHYML. This is another significant improvement. Thirdly for the data sets with molecular clock, as shown in Figure 5 (left 3 columns), our super-quartet method is able to reconstruct the true phylogenetic tree with slightly better accuracy on the average than PHYML.



(a) MD = 0.1 experiment results

(b) MD = 0.3 experiment results

(c) MD = 1.0 experiment results

(d) MD = 2.0 experiment results

Figure 5. Results on synthetic data. X axis: different DNA set; Y axis: percent in building correct tree, figures are rounded to the nearest integers.

The reasons why we use PHYML as the benchmark to compare our algorithm lie in two aspects. Firstly PHYML is one of the most accurate algorithms among existing ML methods [10, 11, 12], and it is one of the most popular and used packages. Secondly PHYML works by perturbing one initial tree (generated by BIONJ). The perturbation is carried out to optimize the ML value by swapping the subtrees connected at each vertex of internal branch, i.e., it evaluates the three possible topologies of the tree by interchanging the four subtrees at vertices of an internal branch. The most important common characteristic of PHYML and our new method is that these two methods all construct the tree through examining the taxon subtree neighborhood relationship on quartet-like binary trees under maximum likelihood. The main difference of PHYML and our super-quartet method is that PHYML starts with a generated tree while our algorithm starts from single taxa and builds the tree sequentially. Obviously our method examines the subtree relations from a much more aggressive way than PHYML, i.e., PHYML uses an initial *N*-sequence tree and changes the tree topology on *N-3* branches, while our method may examine all the subtrees combinations on previous heuristic (*N* is the number of taxa). Our algorithm may have two advantages. Firstly at each merge step we calculate all the likelihood values of all possible super-quartets and take a global view by accumulating all these super-quartet weights to examine every neighborhood relationship of subtrees. The second advantage is our super-quartet approach inherits the theoretical mapping advantage of the quartet method. With these two theoretical advantages, our super-quartet method is able to converge on the global maximum with higher probability.

This can be seen from our experiment results that our super-quartet method is able to achieve higher accuracy than PHYML.

One disadvantage of our algorithm is that it takes $O(n^5)$ steps to complete and thus more expensive than PHYML which only takes $O(n^3)$ steps. The main computation cost for our super-quartet algorithm lies in two parts, i.e., the computation of confidence values for every merge step and the calculation of likelihood values of super-quartet trees. The likelihood value re-calculation is the most expensive part. To reduce the computational cost we may introduce a threshold. The likelihood value re-calculation takes place in a merge step only when the confidence value for the merged pair is below this threshold. Another feasible way in reduction of the computational cost is to incorporate the idea of out-group sequences. When there are ambiguities on which pair of subtrees should be merged, we may pick up only a few other subtrees which are "out-groups" of those which are currently considered to be merged, use one out-group subtree at a time which those considered to be merged to form a super-quartet and calculate its likelihood of three possible trees. Since we do not re-calculate likelihood values for all possible super-quartets from the total number of super-trees, the computational cost can thus be significantly reduced.

5 Conclusion and Future Work

In this paper, we proposed one super-quartet phylogenetic tree reconstruction algorithm. This new algorithm extends our previous quartet-based algorithm and employs an iterative super-quartet approach to enhance the algorithm accuracy. We presented our experiment results and compared them with those obtained using PHYML, one of the most accurate ML algorithms. The experimental results demonstrate that our new algorithm can achieve better results than PHYML. With super-quartets and global quartet weights mechanism, our new algorithm is able to effectively alleviate the problem of quartet errors encountered in traditional quartet-based methods. However our algorithm is computationally more expensive than other methods due to super-quartet weight re-calculation. In the paper we proposed several methods to reduce the total computational cost.

Even though our super-quartets approach is able to achieve very high accuracy, there is still no guarantee it can avoid local maxima. Possible extensions are to develop several mechanisms to deal with those critical merge steps when the ambiguity occurs. One possible extension is to build multiple trees as output in case of possible local maxima convergence.

References

1. J. Adachi and M. Hasegawa, Instability of quartet analyses of molecular sequence data by the maximum likelihood method: the cetacean/artiodactyla relationships. *Cladistics,* Vol. 5, pp.164-166, 1999.

2. J. Felsenstein, Evolutionary trees from DNA sequences: a maximum likelihood approach. *J. Mol Evol.,* 17(6):368-76, 1981.
3. J. Felsenstein, The evolutionary advantage of recombination. II. Individual selection for recombination. *Genetics,* 83(4):845-59, 1976.
4. J. Felsenstein, PHYLIP (phylogeny inference package), *version 3.6a2. Distributed by the author, Department of Genetics, Univ. Washington, Seattle,* 1993.
5. W. M. Fitch, A non-sequential method for constructing trees and hierarchical classifications. *J. Mol. Evol.,* 18, pp. 30-37, 1981.
6. O. Gascuel, BIONJ: An improved version of the NJ algorithm based on a simple model of sequence data. *Mol. Biol. Evol.,* 14:685– 695, 1997.
7. S. Guindon and O. Gascuel, A simple, fast and accurate algorithm to estimate large phylogenies by maximum likelihood. *Syst. Biol.,* 52, pp. 696-704, 2003.
8. T. Jiang, P. E. Kearney and M. Li, Orchestrating quartets: Approximation and data correction. *Proceedings of the 39th IEEE Symposium on Foundations of Computer Science,* pp.416-425, 1998.
9. K. Nieselt-Struwe and A. von Haeseler, Quartet-mapping, a generalization of the likelihood-mapping procedure. *Mol. Biol. Evol.,* 18(7), pp.1204-1219, 2001.
10. G. Olsen, H. Matsuda, R. Hagstrom, and R. Overbeek. FastDNAml: A tool for construction of phylogenetic trees of DNA sequences using maximum likelihood. *Comput. Appl. Biosci.,* 10:41– 48, 1994.
11. S. Ota and W.-H. Li. NJML: A hybrid algorithm for the neighbor-joining and maximum-likelihood methods. *Mol. Biol. Evol.,* 17:1401–1409, 2000.
12. S. Ota and W.-H. Li. NJML+: An extension of the NJML method to handle protein sequence data and computer software implementation. *Mol. Biol. Evol.,* 18:1983–1992, 2001.
13. H. Philippe and E. Douzery, The pitfalls of molecular phylogeny based on four species, as illustrated by the Cetacea/ArtiodacKuhner. *J. Mammal. Evol,.* 2: 133–152, 1994.
14. D. R. Robinson and L. R. Foulds, An optimal way to compare additive trees using circular orders. *J. Comp. Biol.,* pp.731-744, 1981.
15. V. Ranwez and O. Gascuel, Quartet-based phylogenetic inference: Improvements and limits. *Mol. Biol. Evol.,* 18(6), pp.1103-111, 2001.
16. K. Strimmer and A. von Haeseler, Quartet puzzling: A quartet maximum-likelihood method for reconstructing tree topologies. *Mol. Biol. Evol.,* 13(7), pp.964-969, 1996.
17. B. B. Zhou, M. Tarawneh, C. Wang, D. Chu, A. Y. Zomaya and R. P. Brent. A novel quartet-based method for phylogenetic inference. *Proceedings of IEEE 5th Symposium on bioinformatics and bioengineering,* pp.32-39, 2005.
18. B. B. Zhou, M. Tarawneh, D. Chu, P. Wang, C. Wang, A. Zomaya, R. Brent, Evidence of Multiple Maximum Likelihood Points for a Phylogenetic Tree. *Proceedings of IEEE 6th Symposium on bioinformatics and bioengineering, Wasington D.C,* 2006.
19. B. B. Zhou, M. Tarawneh, D. Chu, P. Wang, C. Wang, A. Y. Zomaya, and R.P. Brent, On a New Quartet-based Phylogeny Reconstruction Algorithm. *Proceedings of the 2006 International Conference on Bioinformatics and Computational Biology, Las Vegas,* 2006.