The Performance of Phylogenetic Methods on Trees of Bounded Diameter

Luay Nakhleh[1], Usman Roshan[1], Katherine St. John[1,2], Jerry Sun[1], and Tandy Warnow[1,3]

[1] Department of Computer Sciences, University of Texas, Austin, TX 78712;
{nakhleh,usman,jsun,stjohn,tandy}@cs.utexas.edu
[2] Lehman College & the Graduate Center, City U. of New York
Supported in part by NSF Award 99-73874 and TICAM.
[3] Supported in part by the David and Lucile Packard Foundation.

Abstract. We study the convergence rates of neighbor joining and several new phylogenetic reconstruction methods on families of trees of bounded diameter. Our study presents theoretically obtained convergence rates, as well as an empirical study based upon simulation of evolution on random birth-death trees. We find that the new phylogenetic methods offer an advantage over the neighbor joining method, except at low rates of evolution where they have comparable performance. The improvement in performance of the new methods over neighbor joining increases with the number of taxa and the rate of evolution.

1 Introduction

Phylogenetic trees (that is, evolutionary trees) form an important part of biological research. As such, there are many algorithms for inferring phylogenetic trees. The majority of these methods are designed to be used on biomolecular (i.e. DNA, RNA, or amino-acid) sequences. Methods for inferring phylogenies from biomolecular sequence data are studied (both theoretically and empirically) with respect to the topological accuracy of the inferred trees. Such studies evaluate the effects of various model conditions (such as the sequence length, the rates of evolution on the tree, and the tree "shape") on the performance of various methods.

The *sequence length requirement* of a method is the sequence length needed by the method in order to obtain (with high probability) the true tree topology. Earlier studies established analytical upper bounds on the sequence length requirements of various methods (including the popular neighbor joining [18] method). These studies showed that standard methods, such as neighbor joining, recover the true tree (with high probability) from sequences of lengths that are exponential in the evolutionary diameter of the true tree. Based upon these studies, in [5, 6] we defined a parameterization of model trees in which the longest and shortest edge lengths are fixed, so that the sequence length requirement of a

method can be expressed as a function of the number of taxa, n. This parameterization leads to the definition of "fast-converging" methods, which are methods that recover the true tree from sequences of lengths bounded by a polynomial in n once f, the minimum edge length, and g, the maximum edge length, are bounded. Several fast-converging methods were developed [3, 4, 8, 21]. We and others analyzed the sequence length requirement of standard methods, such as neighbor joining (NJ), under the assumptions that f and g are fixed. These studies [1, 6] showed that neighbor joining and many other methods can be proven to be "exponentially-converging", that is, they recover the true tree with high probability from sequences of lengths bounded by a function that grows exponentially in n. So far, none of these standard methods are known to be "fast-converging."

In this paper, we consider a different parameterization of the model tree space, where we fix the evolutionary diameter of the tree, and let the number of taxa vary. This parameterization, suggested by John Huelsenbeck [personal communication], allows us to examine the differential performance of methods with respect to "taxon sampling" strategies [7]. In this case, the shortest edges can be arbitrarily short, forcing the method to require unboundedly long sequences in order to recover these shortest edges. Hence, the sequence length requirements of all methods cannot be bounded. However, for a natural class of model trees, it can be assumed that $f = \Theta(1/n)$ (for example, random birth-death trees fall into this class). In this case even very simple polynomial time methods converge to the true tree from sequences whose lengths are bounded by a polynomial in n. Furthermore, the degrees of the polynomials bounding the convergence rates of neighbor joining and the "fast-converging" methods are identical – they differ only with respect to the leading constants. Therefore, with respect to this parameterization, there is no significant theoretical advantage between standard methods and the "fast-converging" methods. We then evaluate two methods, neighbor joining and DCM-NJ+MP (a method introduced in [14]) with respect to their performance on simulated data, obtained on random birth-death trees with bounded deviation from ultrametricity. We find that DCM-NJ+MP obtains an advantage over neighbor joining throughout most of the parameter space we examine, and is never worse. That advantage increases as the deviation from ultrametricity increases or as the number of taxa increases.

The rest of the paper is organized as follows. In Section 2, we present the basic definitions, models of evolution, methods, and terms, upon which the rest of the paper is based. In Section 3, we present the theory behind convergence rate bounds for both neighbor joining and "fast-converging" methods. We derive bounds on the convergence rates of various methods for trees in which the

evolutionary diameter (but not the shortest edge lengths) is fixed. We then derive bounds on the convergence rates of these methods for random trees drawn from the distribution on birth-death trees described above. In Section 5, we describe our experimental study comparing the performance of neighbor joining and DCM-NJ+MP. In Section 6, we conclude with a discussion and open problems.

2 Basics

In this section, we present the basic definitions, models of evolution, methods, and terms, upon which the rest of the paper is based.

2.1 Model Trees

The first step of every simulation study for phylogenetic reconstruction methods is to generate *model trees*. Sequences are then evolved down these trees, and these sequences are used, by the methods in question, to estimate the model tree. The accuracy of the method is determined by how well the method reproduces the model tree. Model trees are often taken from some underlying distribution on all rooted binary trees with n leaves. Some possible distributions include the uniform (all binary trees on n leaves are equiprobable) and the Yule-Harding distribution (a distribution based upon a model of speciation).

In this paper, we use random birth-death trees with n leaves as our underlying distribution. To generate these trees, we view speciation and extinction events occurring over a continuous interval. During a short time interval, Δt, a species can split into two with probability $b(t)\Delta t$, and a species can become extinct with probability $d(t)\Delta t$. The values of $b(t)$ and $d(t)$ depend on how much time has passed in the model. To generate a tree with n taxa, we begin this process with a single node and continue until we have a tree with n taxa (with some non-zero probability some processes will not produce a tree of the desired size since all nodes could go "extinct" before n species are generated; if this happens, we repeat the process, until a tree of the desired size is generated). Under this distribution, trees have a natural length assigned to each edge– that is the time t between the speciation event that began that edge and the event (which could be either speciation or extinction) that ended that edge.

Birth-death trees are inherently ultrametric, that is, the branch lengths are proportional to time. In all of our experiments we modified each edge length to deviate from this assumption that sites evolve under the strong molecular clock. To do this, we multiplied each edge by a random number within a range $[1/c, c]$, where we set c to be some small constant. We call this constant the *deviation factor*.

2.2 Models of Evolution

Under the *Kimura 2-Parameter* (K2P) model [10], each site evolves down the tree under the Markov assumption, but there are two different types of nucleotide substitutions: transitions and transversions. A transition is a substitution of a purine (an adenine or guanine nucleotide) for a purine, or a pyrimidine (a cytosine or thymidine nucleotide) for a pyrimidine; a transversion is a substitution of a purine for a pyrimidine or vice versa. The probability of a given nucleotide substitution depends on the edge and upon the type of substitution. A K2P tree is defined by the triplet $(T, \{\lambda(e)\}, ts/tv)$, where $\lambda(e)$ is the expected number of times a random site will change its nucleotide on e, and ts/tv is the transition/transversion ratio. In our experiments, we fix this ratio to 2, one of the standard settings.

It is sometimes assumed that the sites evolve identically and independently down the tree. However, we can also assume that the sites have different rates of evolution, and that these rates are drawn from a known distribution. One popular assumption is that the rates are drawn from a gamma distribution with shape parameter α, which is the inverse of the coefficient of variation of the substitution rate. We use $\alpha = 1$ for our experiments under K2P+Gamma. With these assumptions, we can specify a K2P+Gamma tree just by the pair $(T, \{\lambda(e)\})$.

2.3 Statistical Performance Issues

A phylogenetic reconstruction method is *statistically consistent* under a model of evolution if for every tree in that model the probability that the method reconstructs the tree tends to 1 as the sequence length increases. Under the assumption of a K2P+Gamma evolutionary process, if the transition/transversion ratio and shape parameter are known, it is possible to define pairwise distances between taxa so that distance-based methods (such as neighbor joining) are statistically consistent [11]. Real biomolecular sequences are of limited length. Therefore, the length k of the sequences affects the performance of the method M significantly. The *convergence rate* of a method M is the rate at which it converges to 100% accuracy as a function of the sequence length.

2.4 Phylogenetic Reconstruction Methods

We briefly discuss the two phylogenetic methods we use in our empirical studies: neighbor joining and DCM-NJ+MP. Both methods have polynomial running time.

Neighbor Joining: Neighbor joining [18] is one of the most popular distance based methods. Neighbor joining takes a distance matrix as input and outputs a tree. For every two taxa, it determines a score, based on the distance matrix. At each step, the algorithm joins the pair with the minimum score, making a subtree whose root replaces the two chosen taxa in the matrix. The distances are recalculated to this new node, and the "joining" is repeated until only three nodes remain. These are joined to form an unrooted binary tree.

DCM-NJ+MP: The DCM-NJ+MP method is a variant of a provably fast-converging method that has performed very well in previous studies [14]. In these simulation studies, DCM-NJ+MP outperforms, in terms of topological accuracy, the methods *DCM**-NJ (of which it is a variant) and neighbor joining.

The method works as follows: let d_{ij} be the distance between taxa i and j.

- *Phase 1:* For each $q \in \{d_{ij}\}$, compute a binary tree T_q, by using the Disk-Covering Method from [6], followed by a heuristic for refining the resultant tree into a binary tree. Let $\mathcal{T} = \{T_q : q \in \{d_{ij}\}\}$. (Readers interested in more details of how Phase I is handled should see [6].)
- *Phase 2:* Select the tree from $\mathcal{T}$ which optimizes the parsimony criterion.

If we consider all $\binom{n}{2}$ thresholds in Phase 1, DCM-NJ+MP takes $O(n^6)$ time. However, if we consider only a fixed number p of thresholds, DCM-NJ+MP takes $O(pn^4)$.

2.5 Measures of Accuracy

There are many ways of measuring error between trees. We use the *Robinson-Foulds* (RF) distance [16] which is defined as follows. Every edge e in a leaf-labeled tree T defines a bipartition π_e on the leaves (induced by the deletion of e), and hence the tree T is uniquely encoded by the set $C(T) = \{\pi_e : e \in E(T)\}$, where $E(T)$ is the set of all internal edges of T. If T is a model tree and T' is the tree obtained by a phylogenetic reconstruction method, then the error in the topology can be calculated as follows:

- *False Positives:* $C(T') - C(T)$.
- *False Negatives:* $C(T) - C(T')$.

The RF distance is $\frac{|C(T) \triangle C(T')|}{2(n-3)}$, i.e., the average of the false positive and the false negative rates.

3 Theoretical Results on Convergence Rates

In [1], the sequence length requirement for the neighbor joining method under the Cavender-Farris model was bounded from above, and extended to the General Markov model in [5]. We state the result here:

Theorem 1. *([1, 5]) Let (T,M) be a model tree in the General Markov model. Let*

$$\lambda(e) = -\log|det(M_e)|, \text{ and set } \lambda_{ij} = \sum_{e \in P_{ij}} \lambda(e).$$

Assume that f is fixed with $0 < f \leq \lambda(e)$ for all edges $e \in T$. Let $\varepsilon > 0$ be given. Then, there are constants C and C' (that do not depend upon f) such that, for

$$k = \frac{C}{f^2} \log n e^{C'(\max \lambda_{ij})}$$

then with probability at least $1-\varepsilon$, neighbor joining on S returns the true tree, where S is a set of sequences of length k generated on T. The same sequence length requirement applies to the Q^* method of [2].

From Theorem 1 we can see that as the edge length gets smaller, the sequence length has to be larger in order for neighbor joining to return the true tree with high probability. Note that the diameter of the tree and the sequence length are "exponentially" related.

3.1 Fixed-parameter Analyses of the Convergence Rate

Analysis when both f and g are fixed: In [8, 21], the convergence rate of neighbor joining was analyzed when both f and g are fixed (recall that f is the smallest edge length, and g is the largest edge length). In this setting, by Theorem 1 and because $\max \lambda_{ij} = O(gn)$, we see that neighbor joining recovers the true tree, with probability $1-\varepsilon$, from sequences that grow exponentially in n. An average case analysis of tree topologies under various distributions shows that $\max \lambda_{ij} = \Theta(g\sqrt{n})$ for the uniform distribution and $\Theta(g \log n)$ for the Yule-Harding distribution. Hence, neighbor joining has an average case convergence rate which is polynomial in n under the Yule-Harding distribution, but not under the uniform distribution.

By definition, "fast-converging" methods are required to converge to the true tree from polynomial length sequences, when f and g are fixed. The convergence rates of fast-converging methods have a somewhat different form. We show the analysis for the *DCM**-NJ method (see [21]):

Theorem 2. *([21]) Let (T,M) be a model tree in the General Markov model. Let*

$$\lambda(e) = -\log|det(M_e)|, \text{ and set } \lambda_{ij} = \sum_{e \in P_{ij}} \lambda(e).$$

Assume that f is fixed with $0 < f \leq \lambda(e)$ for all edges $e \in T$. Let $\varepsilon > 0$ be given. Then, there are constants C and C' (that do not depend upon f) such that, for

$$k = \frac{C}{f^2} \log n e^{C'(width(T))}$$

then with probability at least $1-\varepsilon$, DCM-NJ on S returns the true tree, where S is a set of sequences of length k generated on T, and $width(T)$ is a topologically defined function which is bounded from above by $\max \lambda_{ij}$ and is also $O(g \log n)$.*

Consequently, fast-converging methods recover the true tree from polynomial length sequences when both f and g are fixed.

Analysis when $\max \lambda_{ij}$ is fixed: Suppose now that we fix $\max \lambda_{ij}$ but not f. In this case, neither neighbor joining nor the "fast-converging" methods will recover the true tree from sequences whose lengths grow polynomially in n, because as $f \to 0$, the sequence length requirement increases without bound. However, for "random" birth-death trees, the expected minimum edge length is $\Theta(1/n)$. Hence, suppose that in addition to fixing $\max \lambda_{ij}$ we also require that $f = \Theta(1/n)$. In this case, application of Theorem 1 and Theorem 2 shows that neighbor joining and the "fast-converging" methods all recover the true tree with high probability from $O(n^2 \log n)$-length sequences. The theoretically obtained convergence rates differ only in the leading constant, which in neighbor joining's case depends exponentially on $\max \lambda_{ij}$, while in the case of *DCM**-NJ's this rate depends exponentially on $width(T)$. Thus, the performance advantage of a fast-converging method– from a theoretical perspective– depends upon the difference between these two values. We know that $width(T) \leq \max \lambda_{ij}$ for all trees. Furthermore, the two values are essentially equal only when the strong molecular clock assumption holds. Note also that when the tree has a low evolutionary diameter (i.e. when $\max \lambda_{ij}$ is small), then the predicted performance of these methods suggests that they will be approximately identical. Only for large evolutionary diameters should we obtain a performance advantage by using the fast-converging methods instead of neighbor joining.

In the next section we discuss the empirical performance of these methods.

4 Earlier Performance Studies Comparing DCM-NJ+MP to NJ on Random Trees

In an earlier study [14], we studied the performance of the neighbor joining (NJ) method, and several new variants of the disk-covering method. The DCM-

NJ+MP method was one of these new variants we tested. Our experiments (some of which we present here) showed that for random trees (from the uniform distribution on binary tree topologies) with random branch lengths (also drawn from the uniform distribution within some specified range), the DCM-NJ+MP method was a clear improvement upon the NJ method with respect to topological accuracy. The DCM-NJ+MP method was also more accurate in many of our experiments than the other variants we tested, leading us to conclude that the improved performance on random trees might extend to other distributions on model trees.

Later in this paper we will present new experiments, testing this conclusion on random birth-death trees with a moderate deviation from ultrametricity. Here we present a small sample of our earlier experiments, which shows the improved performance and indicates how DCM-NJ+MP obtains this improved performance.

Recall that the DCM-NJ+MP method has two phases. In the first phase, a collection of trees is obtained, one for each setting of the parameter q. This inference is based upon dividing the input set into overlapping subsets, each of diameter bounded from above by q. The NJ method is then used on each subset to get a subtree for the subset, and these subtrees are merged into a single supertree. These trees are constructed to be binary trees, and hence do not need to be further resolved. This first phase is the "DCM-NJ" portion of the method. In the second phase, we select a single tree from the collection of trees $\{T_q : q \in d_{ij}\}$, by selecting the tree which has the optimal parsimony score (i.e. the fewest changes on the tree).

The accuracy of this two-phase method depends upon two properties: first, the first phase must produce a set of trees so that at least some of these trees are better than the NJ tree, and second, the technique (in our case, maximum parsimony) used in the second phase must be capable of selecting a better tree than the NJ tree. Thus, the first property depends upon the DCM-NJ method providing an improvement, and the second property depends upon the performance of the maximum parsimony criterion as a technique for selecting from the set $\{T_q\}$. In the following figures we show that both properties hold for random trees under the uniform distribution on tree topologies and branch lengths.

In Figure 1, we show the results of an experiment in which we scored each of the different trees T_q for topological accuracy. This experiment is based upon random trees from the uniform distribution. Note that the best trees are significantly better than the NJ tree. Thus, the DCM-NJ method itself is providing an advantage over the NJ method.

In Figure 2 we show the result of a similar experiment in which we compared several different techniques for the second phase (i.e. for selecting a tree

from the set $\{T_q\}$). This figure shows that the Maximum Parsimony (MP) technique obtains better trees than the Short Quartet Support Method, which is the technique used in the second phase of the *DCM**-NJ method. Furthermore, both DCM-NJ+MP and *DCM**-NJ improve upon NJ, and this improvement increases with the number of taxa.

Thus, for random trees from the uniform distribution on tree topologies and branch lengths, DCM-NJ+MP improves upon NJ, and this improvement is due to both the decomposition strategy used in Phase 1, and the selection criterion used in Phase 2.

Note however that DCM-NJ+MP is not statistically consistent, even under the simplest models, since the maximum parsimony criterion can select the wrong tree with probability going to 1 as the sequence length increases.

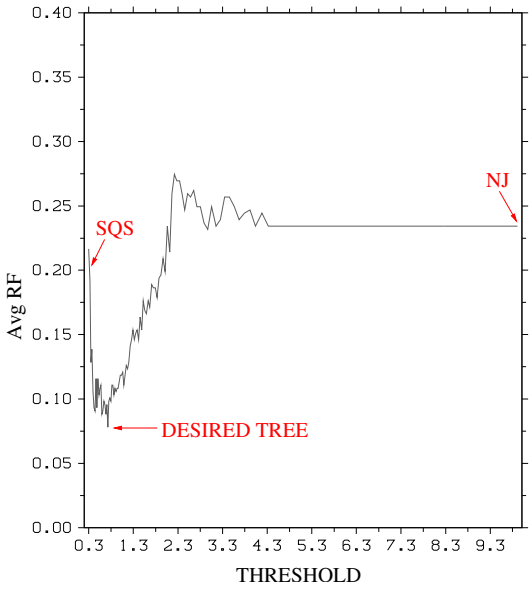

Fig. 1. The accuracy of the T_q's for different values of q on a randomly generated tree with 100 taxa, sequence length 1000, and an average branch length of 0.05.

5 New Performance Studies under Birth-Death Trees

5.1 Introduction

In this paper we focused upon the question of whether the improvement in performance over NJ that we saw in DCM-NJ+MP was a function of the distribu-

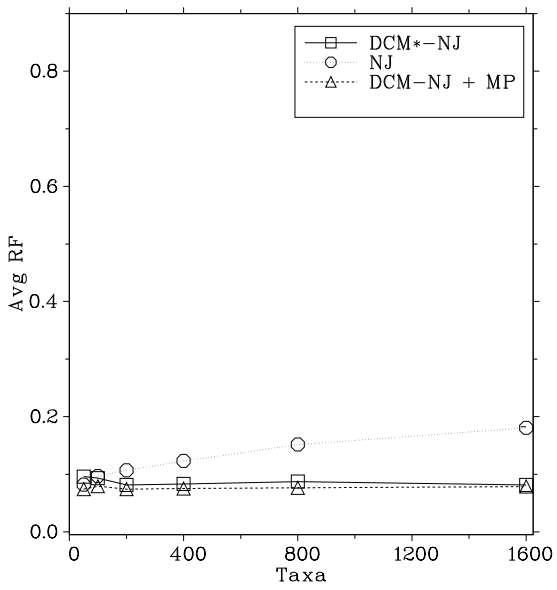

Fig. 2. DCM-NJ+MP vs. *DCM**-NJ vs. NJ on random trees (uniform distribution on tree topologies and branch lengths) with sequence evolution under the K2P+Gamma model. Sequence length is 1000. Average branch length is 0.05.

tion on tree topologies and branch lengths (both uniform), or whether we would continue to see an improvement in performance, by comparison to NJ, when we restrict our attention to a more biologically based distribution on model trees. Hence we focus on random birth-death trees, with some deviation from ultrametricity added (so that the strong molecular clock does not hold). As we will show, the improvement in performance is still visible, and our earlier claims extend to this case.

5.2 Experimental Platform

Machines: The experiments were run on the SCOUT cluster at University of Texas, which contains approximately 130 different processors running the Debian Linux operating system. We also had nighttime use of approximately 150 Pentium III processors located in public undergraduate laboratories.

Software: We used Sanderson's `r8s` package for generating birth-death trees [17] and the program `Seq-Gen` [15] to randomly generate a DNA sequence for the root and evolve it through the tree under K2P+Gamma model of evolution. We calculated evolutionary distances appropriately for the model (see [11]). In

the presence of saturation (that is, datasets in which some distances could not be calculated because the formula did not apply), we used the "fix-factor 1" technique, as defined in [9]. In this technique, the distances that cannot be set using the standard technique are all assigned the largest corrected distance in the matrix.

The software for DCM-NJ was written by Daniel Huson. To calculate the maximum parsimony scores of the trees we used PAUP* 4.0 [19]. For job management across the cluster and public laboratory machines, we used the Condor software package [20]. We generated the rest of this software (a combination of C++ programs and Perl scripts) explicitly for these experiments.

5.3 Bounded Diameter Trees

We performed experiments on bounded diameter trees, and observed how the error rates increase as the number of taxa increases. The birth-death trees that we generated using `r8s` have diameter 2. In order to obtain trees with other diameters, we multiplied the edge lengths by factors of 0.01, 0.1, and 0.5, thus obtaining trees of diameters 0.02, 0.2, and 1.0, respectively. Then, to deviate these trees from ultrametricity, we modified the edge lengths using deviation factor 4. The resulting trees have diameters bounded from above by 4 times the original diameter, but have expected diameters of approximately twice the original diameters. Thus, the final model trees have expected diameters that are 0.04, 0.4, and 2.0. In this way we generated random model trees with 10, 25, 50, 100, 200, 400, and 800 leaves. For each number of taxa and diameter, we generated 30 random birth-death trees (using `r8s`).

5.4 Experimental Design

For each model tree we generated sequences of length 500 using `seq-gen`, computed trees using NJ and DCM-NJ+MP. We then computed the Robinson-Foulds error rate for each of the inferred trees, by comparing it to the model tree that generated the data.

5.5 Results and Discussion

In order to obtain statistically robust results, we followed the advice of McGeoch [12] and Moret [13] and used a number of *runs*, each composed of a number of *trials* (a trial is a single comparison), computed the mean and standard deviation over the runs of these events. This approach is preferable to using the same total number of samples in a single run, because each of the runs is an independent pseudorandom stream. With this method, one can obtain estimates

of the mean that are closely clustered around the true value, even if the pseudo-random generator is not perfect.

The standard deviation of the mean outcomes in our studies varied depending on the number of taxa. The standard deviation of the mean on 10-taxon trees is 0.2 (which is 20 percent, since the possible values of the outcomes range from 0 to 1), on 25-taxon trees is 0.1 (which is 10 percent), whereas on 200, 400 and 800-taxon trees the standard deviation ranged from 0.02 to 0.04 (which is between 2 and 4 percent). We graph the average of the mean outcomes for the runs, but omit the standard deviations from the graphs.

In Figure 3, we show how neighbor joining and DCM-NJ+MP are affected by increasing the rate of evolution (i.e. the height). The *x*-axis is the maximum expected number of changes of a random site across the tree, and the *y*-axis is the RF rate. We provide a curve for each number of taxa we explored, from 10 up to 800. The sequence length is fixed in this experiment to 500. Note that both neighbor joining and DCM-NJ+MP have high errors for the lowest rates of evolution, and that at these low rates of evolution the error rates increase as n increases. This is because for these low rates of evolution, increasing the number of taxa makes the smallest edge length (i.e. f) decrease, and thus increases the sequence length needed to have enough changes on the short edges for them to be recoverable. As the rate of evolution increases, the error rates initially decrease for both methods, but eventually the error rates begin to increase again. This increase in error occurs where the exponential portion of the convergence rate (i.e. where the sequence length depends exponentially on $\max \lambda_{ij}$) becomes significant. Note that where this happens is essentially the same for both methods– and that they perform equally well until that point. However, after this point, neighbor joining's performance is worse, compared to DCM-NJ+MP; furthermore, the error rate increases for neighbor joining at each of the "large" diameters, as n increases, while DCM-NJ+MP's error rate does not reflect the number of taxa nearly as much.

In Figure 4, we present a different way of looking at the data. In this figure, the *x*-axis is the number of taxa, the *y*-axis is the RF rate, and there is a curve for each of the methods. We show thus how increasing n (the number of taxa) while fixing the diameter of the tree affects the accuracy of the trees reconstructed. Note that at low rates of evolution (the left figure), the error rates for both methods increase with the number of taxa. At moderate rates of evolution (the middle figure), error rates increase for both methods but more so for neighbor joining than for DCM-NJ+MP. Finally, at the higher rate of evolution (the right figure), this trend continues, but the gap is even larger – in fact, DCM-NJ+MP's error increase looks almost flat.

These experiments suggest strongly that except for low diameter situations, the DCM-NJ+MP method (and probably the other "fast-converging" methods) will outperform the neighbor joining method, especially for large numbers of taxa and high evolutionary rates.

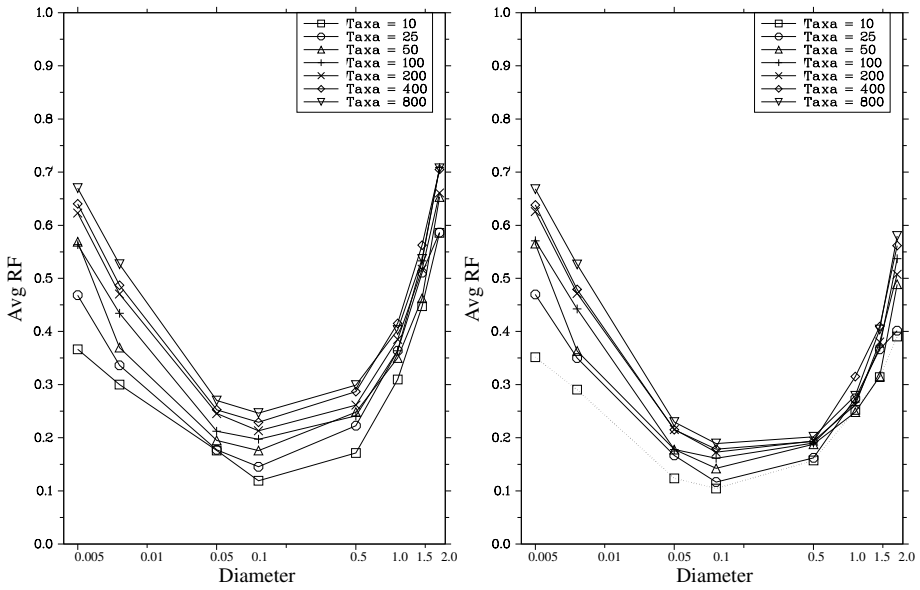

Fig. 3. NJ (left graph) and DCM-NJ+MP (right graph) error rates on random birth-death trees as the diameter (x-axis) grows. Sequence length fixed at 500, and deviation factor fixed at 4.

Table 1 shows the average running times of neighbor joining and DCM-NJ+MP on the trees that we used in the experiments. The DCM-NJ+MP version that we ran looked at 10 thresholds in Phase 1 instead of looking at all the $\binom{n}{2}$ thresholds.

6 Conclusion

In an earlier study we presented the DCM-NJ+MP method and showed that it outperformed the NJ method for random trees drawn from the uniform distribution on tree topologies and branch lengths. In this study we show that this improvement extends to the case where the trees are drawn from a more biologically realistic distribution, in which the trees are birth-death trees with a moderate deviation from ultrametricity. This study has consequences for large phylogenetic analyses, because it shows that the accuracy of the NJ method

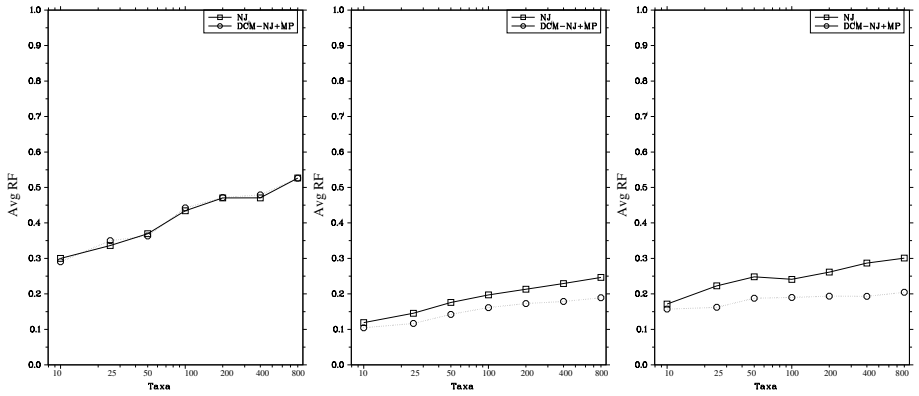

Fig. 4. NJ and DCM-NJ+MP: Error rates on random birth-death trees as the number of taxa (x-axis) grows. Sequence length fixed at 500 and the deviation factor at 4. The expected diameter of the resultant trees are 0.02 (for the left graph), 0.2 (for the middle graph), and 1.0 (for the right graph).

Taxa	NJ	DCM-NJ+MP
10	0.01	1.94
25	0.02	9.12
50	0.06	24.99
100	0.35	132.46
200	2.5	653.27
400	20.08	4991.11
800	160.4	62279.3

Table 1. The running times of NJ and DCM-NJ+MP in seconds.

may suffer significantly on large datasets. Furthermore, since the DCM-NJ+MP method has good accuracy, even on large datasets, our study suggests that other polynomial time methods may be able to handle the large dataset problem without significant error.

7 Acknowledgments

We would like to thank the David and Lucile Packard Foundation (for a fellowship to Tandy Warnow), the National Science Foundation (for a POWRE grant to Katherine St. John), the Texas Institute for Computational and Applied Mathematics and the Center for Computational Biology at UT-Austin (for support of Katherine St. John), Doug Burger and Steve Keckler for the use of the SCOUT cluster at UT-Austin, and Patti Spencer and her staff for their help.

References

1. K. Atteson. The performance of the neighbor-joining methods of phylogenetic reconstruction. *Algorithmica*, 25:251–278, 1999.
2. V. Berry and O. Gascuel. Inferring evolutionary trees with strong combinatorial evidence. In *Proc. 3rd Ann. Int'l Conf. Computing and Combinatorics (COCOON 97)*, pages 111–123. Springer Verlag, 1997. in *LNCS 1276*.
3. M. Csűrös. Fast recovery of evolutionary trees with thousands of nodes. To appear in RECOMB 01, 2001.
4. M. Csűrös and M. Y. Kao. Recovering evolutionary trees through harmonic greedy triplets. *Proceedings of ACM-SIAM Symposium on Discrete Algorithms (SODA 99)*, pages 261–270, 1999.
5. P. L. Erdos, M. Steel, L. Székély, and T. Warnow. A few logs suffice to build almost all trees– I. *Random Structures and Algorithms*, 14:153–184, 1997.
6. P. L. Erdos, M. Steel, L. Székély, and T. Warnow. A few logs suffice to build almost all trees– II. *Theor. Comp. Sci.*, 221:77–118, 1999.
7. J. Huelsenbeck and D. Hillis. Success of phylogenetic methods in the four-taxon case. *Syst. Biol.*, 42:247–264, 1993.
8. D. Huson, S. Nettles, and T. Warnow. Disk-covering, a fast-converging method for phylogenetic tree reconstruction. *Comput. Biol.*, 6:369–386, 1999.
9. D. Huson, K. A. Smith, and T. Warnow. Correcting large distances for phylogenetic reconstruction. In *Proceedings of the 3rd Workshop on Algorithms Engineering (WAE)*, 1999. London, England.
10. M. Kimura. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *J. Mol. Evol.*, 16:111–120, 1980.
11. W. H. Li. *Molecular Evolution*. Sinauer, Massachuesetts, 1997.
12. C. McGeoch. Analyzing algorithms by simulation: variance reduction techniques and simulation speedups. *ACM Comp. Surveys*, 24:195–212, 1992.
13. B. Moret. Towards a discipline of experimental algorithmics, 2001. To appear in Monograph in Discrete Mathematics and Theoretical Computer Science; Also see http://www.cs.unm.edu/ moret/dimacs.ps.
14. L. Nakhleh, U. Roshan, K. St. John, J. Sun, and T. Warnow. Designing fast converging phylogenetic methods. Oxford U. Press, 2001. To appear in *Bioinformatics*: Proc. 9th Int'l Conf. on Intelligent Systems for Mol. Biol. (ISMB 01).
15. A. Rambaut and N. C. Grassly. Seq-gen: An application for the Monte Carlo simulation of dna sequence evolution along phylogenetic trees. *Comp. Appl. Biosci.*, 13:235–238, 1997.
16. D. F. Robinson and L. R. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53:131–147, 1981.
17. M. Sanderson. `r8s` software package. Available from http://loco.ucdavis.edu/r8s/r8s.html.
18. N. Sautou and M. Nei. The neighbor-joining method: A new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.*, 4:406–425, 1987.
19. D. L. Swofford. PAUP*: Phylogenetic analysis using parsimony (and other methods), 1996. Sinauer Associates, Underland, Massachusetts, Version 4.0.
20. Condor Development Team. Condor high throughput computing program, Copyright 1990-2001. Developed at the Computer Sciences Department of the University of Wisconsin; `http://www.cs.wisc.edu/condor`.
21. T. Warnow, B. Moret, and K. St. John. Absolute convergence: true trees from short sequences. *Proceedings of ACM-SIAM Symposium on Discrete Algorithms (SODA 01)*, pages 186–195, 2001.