Tutorial on Phylogenetic Tree Estimation

Junhyong Kim
Department of Ecology and Evolutionary Biology
Yale University
New Haven, Connecticut
e-mail: junhyong.kim@yale.edu

Tandy Warnow
Department of Computer Science
University of Texas
Austin, TX
e-mail: tandy@cs.utexas.edu

1 Tutorial Summary

All biological disciplines are united by the idea that species share a common history. The genealogical history of life - also called an "evolutionary tree" - is usually represented by a bifurcating, leaf-labeled tree. The use of evolutionary trees is a fundamental step in many biological problems, such as multiple sequence alignments, protein structure and function prediction, and drug design.

The primary scientific objective of phylogenetic studies is not to solve a given optimization problem, but rather to recover the order of speciation or gene duplication events represented by the topology of the true evolutionary tree. (Locating the root of the evolutionary tree is a scientifically difficult task, so that a method is considered to have been successful if it recovers the topology of the *unrooted* tree.) This means that good or poor performance with respect to optimization problems is only important to the degree that it guarantees good or poor performance with respect to topology estimation.

Unfortunately, inferring evolutionary trees is an enormously difficult problem for several reasons. For one, the phylogeny problem is a difficult statistical problem because its parameter space has a complicated structure, and there is no 'off the shelf' solution to the phylogeny problem that can be applied. The phylogeny problem also presents a considerable computational challenge. Typical data sets now consist of several hundred species, and presently available tree reconstruction methods are inadequate to the task of analyzing such datasets. For example, an *rbc*L DNA sequence data set of 500 plants has been analyzed for several years now, without solution. The explanation for why these analyses are so difficult is simple: the optimization problems are *NP*-hard, and the heuristics used in an attempt to solve these optimization problems use hill-climbing techniques to search through an exponentially large space of phylogenetic trees.

Statistical approaches towards phylogeny reconstruction have modeled the evolutionary process stochastically, and have studied the performance of methods for recovering phylogenetic trees in terms of the accuracy of these methods on datasets of finite length sequences generated under different model trees. These studies have shown that some methods recover the true tree topology with high probability, once the sequences are long enough, while other methods have no such guarantees. Over the last decade or so, computer scientists have also begun to design and analyze the performance of phylogenetic methods under these statistical models.

One of the results of this interest in using statistical models of evolution to explore the performance of different methods for tree reconstruction is the recent development of analytical techniques for estimating the sequence lengths that suffice for accuracy with high probability of a given phylogenetic method, when applied to data generated on a fixed model tree. These techniques have shown that a number of polynomial time methods converge to the true tree from sequences of lengths that grow exponentially in n, the number of leaves in the tree. More recently, these techniques have been extended to show how to develop methods that are *fast-converging*, so that they recover the true tree topology with high probability from sequences that grow only polynomially in n, and typically grow polylogarithmically. These advances may represent the first breakthroughs towards obtaining polynomial time methods that can handle large evolutionary datasets.

This tutorial will present the techniques for analyzing convergence rates, and describe the basics of the fast-converging methods. No background in probability, statistics, or phylogenetics will be assumed.

2 Basics

An evolutionary tree (also called a *phylogenetic tree*) models the evolution of a set of taxa (species, biomolecular sequences, languages, etc) from a common origin. Thus, an evolutionary tree is rooted at the most recent common ancestor of the taxa, and the internal nodes of the tree are each labeled by a hypothesized or known ancestor. The common practice today is to use biomolecular sequences as representatives of the species set, so that the leaves of the tree are labeled by biomolecular (DNA, RNA, or amino acid) sequences. Morphological features are also used to assist in the reconstruction of phylogenetic trees. Both morphological features and aligned biomolecular sequences define *qualitative characters*, which means that they induce a partition of the species set into distinct *character states*. Thus, for example, the morphological feature *vertebrate-invertebrate* defines a binary (two-state) character. When using biomolecular sequences, each *site* (i.e. position) within the multiple alignment defines a character, so that the sequences having the same nucleotide (or amino-acid) at that site exhibit the same state of that character.

Thus, any given set of species set S can be represented by the values each species in S attains for each of the characters in a set C of characters; hence, we can represent the input to a phylogenetic reconstruction problem by a $|S| \times |C|$ matrix such that the ij^{th} entry is the state of the i^{th} species for the j^{th} character. A *phylogenetic tree* T is a tree whose leaves are labeled by the species in the set, and are numbered by $1, 2, \ldots, n$. The objective then of a phylogenetic reconstruction algorithm is to find a tree which best fits the data.

At this point, it is important to point out again the biological assumptions of this input matrix. Our assumption is that rows of the matrix represent biological entities that have arose through a sequence of multiplications from a single common ancestor. Here, by "multiplication" we mean biological events like gene duplication, cell duplication, speciation, and so on. A graphical representation of this sequence of multiplications is the tree. The columns of the matrix represent measurements where some aspect of the measurement is shared among the row objects by common descent from the single ancestor. This shared attribute due to common ancestry is called *homology*. Therefore, in biomolecules we assume that their positional organization arose from common descent and we talk of positional homology. The importance of these concepts is to point out where a phylogenetic tree analysis is inappropriate. First, it is inappropriate to apply phylogenetic tree analysis where we do not expect a hierarchical (tree-like) data structure-e.g., expression patterns of genes. Second, it is an inappropriate tool when there is no reasonable way to establish homology of the characters-e.g., presence-absence of protein folds.

Our discussion of tree reconstruction is not concerned with the location of the root of the tree, because the location of the root is difficult to achieve with any degree of accuracy. However, rooted trees are reconstructed by systematic biologists, and the technique generally employed is to use an *outgroup*, which is a taxon which is clearly less related to the rest of the group than any two members in the group are to each other. For example, if we were interested in the phylogenetic relationship of primates, we might use some other mammal that is clearly not a primate, say a mouse, as the outgroup. Once the best unrooted tree containing the outgroup is constructed, the unrooted tree can be "rooted" on the edge separating the outgroup from the rest of the taxa. The problem with using outgroups is that what appears to be an outgroup may not in fact be an outgroup (if all "outgroup?" decisions were easy, then trees would be easy to construct using methods in [74]), and that if the taxon is definitely an outgroup, it may be difficult to locate the edge to which it should attach because it might be very different from the tingroupv. For example, an bacterium is clearly an outgroup member with respect to primates, but they are too different to even easily determine homologous characters. There is also a not-so-commonly used rooting method called *mid-point rooting*. In mid-point rooting the longest edge-weighted path between two terminal taxa are found and the tree is rooted at the mid-point of this path [107]. However, the rationale for this method comes from making a molecular-clock assumption-which is often found to be violated. Furthermore, seeking rooted versions of trees just increases the probability of error, since it adds another aspect of the tree which can be incorrectly analyzed.

Consequently, our objective in tree reconstruction is to obtain an accurate recovery of the topology of the unrooted tree, and this is in general accomplished through the use of related optimization criteria. Some criteria are based upon the sequence data, while others are based upon distances computed between sequences in the data, but unfortunately almost all of the resultant optimization problems have been shown to be NP-hard (and some even NP-hard to solve approximately!) [3, 36, 49]. Of the various sequence-based criteria used to evaluate trees, *parsimony, compatibility* and *maximum likelihood* are the most popular. Parsimony and compatibility are both NP-hard problems, so that "solutions" are generally obtained using heuristics (mostly hill-climbing), although exact algorithms based upon branch-and-bound approaches are used for small enough data sets (generally speaking of up to about 17 taxa). Solutions for the "maximum likelihood" tree are obtained through similar hill-climbing searches, but evaluating each fixed leaf-labeled topology is more computationally expensive (in principle, not even known to be solvable in polynomial time; however in practice, heuristics which find local optima exist, and these seem to scale as a square of the number of the leaves). Consequently, maximum likelihood has not been used as frequently as parsimony for tree reconstruction.

Distance-based approaches are also popular, and have a solid statistical foundation. While optimization-based approaches are desirable, almost all optimization problems relevant to distance-based reconstruction are again NP-hard (see [3, 36, 49]). Some polynomial time distance methods also have guaranteed accuracy once the sequences are long enough, and have been shown experimentally to have good performance on small data sets [70, 83, 102].

3 Stochastic models of evolution and performance criteria under these models:

3.1 The models

Many models have been proposed to describe the evolution of biomolecular sequences. Such models depend on the underlying phylogenetic tree and some randomness. Many models assume that the sites are independently and identically distributed (iid.). In the most general stochastic model that we study the sequence sites evolve iid. according to the general Markov model from the root [110]. Since the iid. condition is assumed, it is enough to consider the evolution of a single site in the sequences. Substitutions (point mutations) at a site are generally modeled by a probability distribution π on a set of $r > 1$ character states at the root ρ of the tree (an arbitrary vertex or a subdividing point on an edge), and each edge e has an associated $r \times r$ stochastic transition matrix $M(e)$. Assigning such a stochastic transition matrix to each edge gives us a Markov chain proceeding from each vertex to vertex. It is more common to assume a continuous time Markov model where each edge of the tree has some positive real value associated with it corresponding to actual evolutionary time. Along with the time variable we also assume that each edge has an associated instantaneous rate matrix $Q(e)$ such that the stochastic transition matrix for the edge is given by $M(e) = e^{Q(e)t}$.

Many popular variants of the four-state model (i.e. $r = 4$) exist because of their relevance to DNA sequence data. Different models can be seen as variations of imposing constraint relationships among the elements of the Markov transition probability matrix (or the transition rate matrix depending on discrete or continuous time parameterization). For example, the Jukes-Cantor model has the most extreme form of constraints in that all off-diagonal elements of every $M(e)$ are constrained to be identical. At least eight different forms of the four-state model is encountered in the biological biological literature [115]. The most popular of these are: Jukes-Cantor [146], Kimura 2-parameter [145], Hasegawa-Kishino-Yano (HKY) [144], and the Equal-input model [143]. In terms of performance, the use of a more complicated model (i.e. a model with more parameters) does not necessarily lead to better tree estimates since the sampling variance associated with a more complicated model may be higher at smaller data size (i.e. shorter sequences, see [147]). The determination of an appropriate model choice for molecular data is an active field of research [140-142].

Biological evolution is unlikely to follow any single Markov model from the previous discussion. Many different forms of more complex models have been proposed-usually in the form of some kind of a mixture model across the sites of the evolving molecule. Roughly, there are two kinds of mixture models: those models that result in *iid* models, and those that relax the iid assumption. The class of *iid* models corresponds to cases in which the mixture is a stochastic mixture. The best way to think of this is as drawing a character from two (or more) populations, say X and Y. In a stochastic mixture model, we draw from either the population X or population Y based on some stochastic process (say, we flip a coin). Therefore, we do not know whether the particular character has been sampled from population X or population Y. In this sense, this results in an iid sampling distribution of the characters since every character has the same mixture distribution. One popular form of stochastic mixture is the gamma distributed rates model where the rate parameter for the Poisson process is assumed to have a prior distribution from the gamma family of distributions [135-139]. The non-identical distribution models arise when the model allows the characters to be drawn from different populations and either makes an a priori distinction into different classes (e.g., different codon positions within a coding DNA fragment) or a best guess classification into different classes. The latter approach includes Hidden-Markov Models (HMM) where it is assumed that the population class of the character forms a hidden state space. Then, HMMs assume that the hidden states form a Markov chain across the sites [133] (see also [132]). More recently [131] have shown that the maximum parsimony optimization problem solves the maximum likelihood problem under a model in which every site evolves under its own evolutionary process. More precisely, if we have a model such that for each site i there is a set of symmetric transition matrices $M_i(e)$ with identical off-diagonal elements $m_{ij} \leq (r-1)/r$ where r is the number of states, then the leaf-labelled tree maximizing the maximum likelihood score under this model are identical to the leaf-labelled trees optimizing the maximum parsimony score. Other parameter rich models have also been proposed, in which the tree has a fixed set of edge weights corresponding to time, but every site i has its own scaling of the edge weights by a rate parameter λ_i (G. Olsen, pers. comm.). However, it is not clear that such parameter-rich estimates will perform satisfactorily from finite sized data. It is also the case, that since the number of parameters scale linearly with the number of sites, statistical consistency cannot be guaranteed for these models. A more interesting class of complex models involve a slight relaxation of time-homogeneity in the Markov process. For example, the covarion model in which each site is able to change only in some portions of the tree [130, 129]. Thus, sites "turn on" and "turn off", which is presumed to model the gain or loss of constraints on various regions of a given protein due to changes in its structure or function. In particular, this model (as well as HMMs) also results in a relaxation of the independence assumption across sites.

Statisticians have addressed the performance of phylogenetic reconstruction (or estimation) methods with respect to the accuracy of the unrooted leaf-labeled tree obtained by the method. Methods which will recover the true tree (i.e. leaf-labeled tree) with arbitrarily high probability, given long enough sequences, are said to be **statistically consistent** for that tree. A fair amount is known about the statistical consistency (or lack thereof) of different phylogenetic methods under the simplest *iid.* models, but the performance of estimation methods under various mixture models is poorly studied. For example, the statistical consistency of estimators based on mixture models has not been established, and Chang [127] showed that standard maximum likelihood estimators (based upon pure Markov models) become inconsistent estimators under stochastic mixture models. On a more important issue, under mixture models, it is not clear that different tree topologies are identifiable. A set of tree topologies are **identifiable** if under a suitable set of stochastic evolutionary models, the joint probability distribution of the characters is disjoint for the different tree topologies except at trivial points (points where edge weights are zero). Under standard models of Markov evolution, Chang established that different tree topologies are identifiable [128]. Steel et al. [109] also showed that if the particular mixture is known and identical for different trees, then the tree topologies are identifiable. However, under unrestricted mixture models all tree topologies can generate identical joint probability distributions for the characters. It remains to be rigorously demonstrated whether popular mixture models such as the gamma distributed models are identifiable, but more recent results seem to indicate that this is indeed the case (K. Atteson, pers. comm.).

3.2 The Cavender-Felsenstein model

We now introduce the **Cavender-Felsenstein** model [31, 32, 52] (also called the *Cavender-Farris* model, and henceforth referred to as the "CF model"). This is the simplest possible Markov model of evolution, and under this model (as well as under more general *iid.* models), we can establish bounds on the convergence rates of different methods.

Let $\{0, 1\}$ denote the two states. The root is a fixed leaf and the distribution π at the root is uniform. For each edge e of a tree T we have an associated *mutation probability* that lies strictly between 0 and 0.5. Let $p : E(T) \to (0, 0.5)$ denote the associated map. Each site evolves down the tree identically and independently according to a Markov process, so that $p(e)$ denotes the probability that the character state in site i changes at the endpoints of the edge e.

Thus, the CF model is an instance of the general Markov model with

$$M(e) = \begin{bmatrix} 1-p_e & p_e \\ p_e & 1-p_e \end{bmatrix}.$$

We now describe a nice formula which is useful for understanding the performance of methods with respect to topology estimation. Given a leaf-labeled tree T and a subset S of the leaves, we denote the subtree of T induced by this set S by $T_{|S}$. If we suppress all nodes of degree two in $T_{|S}$ (by contracting edges incident with such nodes) we obtain the tree we denote by $T^*_{|S}$. Given $T^*_{|S}$, we can define mutation probabilities on the edges of $T^*_{|S}$ so that the probability distribution on the patterns on S is the same as the marginal of the distribution on patterns provided by the original tree T. The mutation probability that we assign to an edge of $T^*_{|S}$ is just the probability p that the endpoints of the associated path in the original tree T are in different states, and p is nicely related to the mutation probabilities $p_1, p_2, \ldots, p_k$ of edges of the path of the original tree:

$$p = \frac{1}{2}\left(1 - \prod_{i=1}^{k}(1-2p_i)\right) \quad (1)$$

Formula (1) is well-known and easy to prove by induction.

An equivalent way to view a Cavender-Farris (CF) tree (that will be useful when we analyze the performance of distance-based methods) is as a pair $(T, \{\lambda_e : e \in E(T)\})$, where λ_e is the expected number of changes of a random site on edge e, and where the random variable for the number of changes on each edge is Poisson. It is not hard to see that $\lambda_e = -1/2ln(1-2p(e))$, where $p(e)$ is the probability of an observed change on the edge e.

The values of $p(e)$ or λ_e are sometimes referred to as the *edge weights*; in context, which of the two is usually clear. (But, unfortunately, not so clear in some of the literature.)

3.3 Quantifying Accuracy

While we will show that exact accuracy in the topology estimation is possible with high probability, given long enough sequences, on finite length sequences typically observed in real data, some error in the topology estimation is likely to occur. Therefore, the *degree of accuracy* has also been quantified. If T is the model tree and T' is the estimation of the model tree (so that both are leaf-labeled by the same set S of taxa), then we can compare T and T' to determine the degree of accuracy in the estimation.

Most typically, this comparison operates as follows:

- Let $e \in E(T)$ be an *internal* edge of T, and let π_e be the bipartition of S induced by deleting the edge e from T. Let $C(T) = \{\pi_e : e \in E(T)\}$. Similarly let $C(T') = \{\pi_e : e \in E(T')\}$. This set is called the **character encoding of** T.
- The **false positives** are those bipartitions in $C(T') - C(T)$, and the **false negatives** are those bipartitions in $C(T) - C(T')$.
- The **false negative rate** is $|FN|/|E_I(T)|$, where $E_I(T)$ denotes the internal edges of T. The **false positive rate** is $|FP|/|E_I(T')|$.

Note:

- $FP = FN$ if both T and T' are binary
- If T is binary, then $FP \leq FN$ for all T'.

Some authors suggest using the average of the false negative and false positive rate as a technique for comparing trees; this is called the **Robinson Foulds** score. When both trees are binary, the Robinson-Fould score is equal to the False Negative rate and the False Positive rate. However, when T is binary but T' is not, then the Robinson-Foulds score can be biased in favor of unresolved trees (for example, the Robinson-Foulds score of a star-tree is 50%, which is the same as a binary tree which gets half the edges of the true tree). Therefore, the use of the *expected Robinson-Foulds* score, which is the Robinson-Foulds score of a random binary refinement of T', has also been suggested.

4 Sequence Based Reconstruction

4.1 Parsimony

Parsimony is one of the most popular methods for phylogenetic tree inference, and yet it is a method whose applicability to phylogeny reconstruction is seriously and sometimes violently disputed in the systematic biology literature. In order to define the parsimony method, we begin with the following definitions.

The **Hamming distance** between two sequences x and y of the same length is $|\{i : x_i \neq y_i\}|$ and is denoted $H(x,y)$. The **parsimony length** of a tree in which each node v is labeled by a sequence s^v of length k over Σ is the sum of the Hamming distances of sequences labeling endpoints of edges in the tree, i.e. $\sum_{(a,b)\in E} H(a,b)$. Given a set S of sequences, a **most parsimonious tree** for S is a tree leaf-labeled by S and assigned sequences for the internal nodes, of minimum parsimony length. Thus, the **parsimony criterion** is to find a tree of minimum length.

The motivation for the parsimony criterion is the observation that if evolution is assumed to operate only through point mutations (for example, substitutions of one nucleotide for another) then the parsimony length of a tree is the minimum possible number of evolutionary events needed to obtain the set of sequences observed at the leaves through point mutations.

Given an arbitrary set of sequences, the **parsimony problem** is to find a tree of minimum parsimony cost (i.e. a "most parsimonious tree"). Unfortunately, this is an NP-hard problem, even when the sequences are binary (i.e. the alphabet size is two) [37, 60, 65]. The most typical approach (and the one that is implemented in most of the popular software packages) has been taken to attempting to find the maximum parsimony tree(s) is to examine as many different leaf-labeled topologies as possible, evaluating each one for its best possible labeling of the internal nodes, and selecting the best of all the considered trees. Given a fixed leaf-labeled tree, computing the parsimony length can be achieved in polynomial time [57,67]; consequently, this technique is polynomial in the number of considered trees. Branch-and-bound algorithms exist which reduce the number of trees that need to be evaluated, but in practice parsimony can be solved exactly only for up to about 20 to 30 sequences, depending on the dataset.

When we consider parsimony from the perspective of how well it performs for reconstructing trees under various stochastic models of evolution, we get a mixed response. It is now well known that there are very simple model conditions under which parsimony is an inconsistent estimator (see [55,82]), while under these conditions simple polynomial time distance methods are statistically consistent [39,40,41]. However, under other model conditions (that may be reasonably considered closer to biologically realistic cases), even maximum likelihood estimation can be inconsistent [109]. It is also the case that maximum parsimony method has an interesting kind of robustness - for the subset of models that it consistently estimates, it can consistently estimate any mixture of those models. By contrast, maximum likelihood does not have this property. Furthermore, experimental studies indicate that maximum parsimony performs well in practice [69,100] by comparison to polynomial time distance-based methods, which are guaranteed to be statistically consistent. For these reasons, among others, maximum parsimony remains an important problem in systematic biology.

Approximating Parsimony: The MP optimization problem can be 2-approximated in polynomial time in a very simple way. This is a well-known result in the community.

Given the set S of sequences define the weighted complete graph $G(S)$ whose node set is bijectively labeled by the species in S, and where $w(i,j)$ is the Hamming distance between the i^{th} and j^{th} sequences.

Theorem 1 *Let T be a minimum spanning tree on $G(S)$. Then the parsimony length of T is at most twice that of the most parsimonious tree.*

Proof: Consider a most parsimonious tree, T^*, and consider the result of doubling the tree T^* to create an edge-weighted graph G in which every edge in T^* appears twice. This is an Eulerian graph since every node has even degree, and consequently G has an Eulerian tour, γ. Create from γ a smaller tour, γ', which contains only the nodes of $G(S)$ ordered in the way in which they appear in γ. Let the weight of the tour γ be denoted by $w(\gamma)$, and define it to be the sum of the weights of the edges in γ. Similarly define $w(\gamma')$. It is easy to see the following:

$$w(\gamma) = w(G) = 2w(T^*)$$

since G is Eulerian, and that

$$w(\gamma') \leq w(\gamma)$$

since Hamming distances satisfy the triangle inequality. Now note that if we delete any single edge from γ' we create a path P such that

$$w(P) \leq w(\gamma').$$

Now consider T, a minimum spanning tree for the graph $G(S)$. Since P is also a spanning tree, it follows that

$$w(T) \leq w(P) \leq w(\gamma') \leq w(\gamma) = 2 * w(T^*).$$

□

The main contribution of this result is an upper bound on the parsimony length of the most parsimonious tree for a given input.

Maximum Parsimony is *not* guaranteed to be statistically consistent, even under very simple models. Felsenstein showed an example of a four-taxon CF tree with two rates on the edges, in which Maximum Parsimony is inconsistent [55], and we are beginning to establish more general conditions under which we can show that MP is inconsistent for specific model trees [82]. However, in simulation studies, MP often performs well, with respect to its false negative rates [69]. Consequently, it is possible that MP's accuracy under biologically realistic CF trees will be fairly good.

4.2 Maximum Likelihood

The use of maximum likelihood estimators in evolutionary trees were pioneered by J. Felsenstein, A. Edwards, and E. Thompson. The general idea behind maximum likelihood estimators is the observation that

$$P(Model|Data) = P(Model\ and\ Data)/P(Data)$$

$$= P(Data|Model)P(Model)/P(Data).$$

In this formulation $P(Model|Data)$ is proportional to $P(Data|Model)$; therefore we can ad hoc justify estimating a model by finding the model that maximizes the conditional probability $P(Data|Model)$, which is also called the *likelihood* of the data.

One advantage of the maximum likelihood estimator is that it allows an "algorithmic" way of incorporating model-theoretic view of sequence evolution. Namely, specify the model as a stochastic model which yields the joint probability distribution of characters. For example, let $p_i = p_i(\theta, T)$ be the probability of the ith character pattern (i.e., ith way of assigning character states to the leaves of the tree) where θ denotes the various parameters of the stochastic model (e.g., edge weights) and T is the tree topology. Then the log-likelihood of an observed data set with f_i the frequency of the i^{th} character pattern is given by $L = \sum_i f_i \log p_i(\theta, T)$

For example, the maximum likelihood estimation (MLE) method for Cavender-Farris trees is as follows:

> Given set S of sequences generated on an unknown Cavender-Farris tree, find a CF tree $(T, \{\lambda_e\})$ such that $Pr[S|(T, \{\lambda_e\})]$ is maximized.

Note that the MLE method depends upon the model; thus, the MLE method for Jukes-Cantor trees is different than the MLE method for CF trees, and both are different methods than for the rates-across-sites MLE method.

The usual MLE programs evaluate a given tree topology by numerically finding local optima for the edge weight parameters, and then searching through the different tree topologies. Remember that although the maximum parsimony problem (finding the tree of optimal parsimony score) is NP-hard, computing the parsimony score of a fixed tree is solvable in linear time, using the Fitch-Hartigan algorithm. By contrast, the MLE optimization problem on a fixed tree can be computationally expensive, even under the simplest models in which all the sites evolve identically and independently (*iid*). In this case, we have k edge weight parameters and m edges, and the maximum likelihood estimation problem is a km dimensional numerical optimization problem. Although some faster optimizing programs have been developed [5] the computational problem can be significant. Usual optimization algorithms converge to a local optimum in cubic-time for a smooth function. Popular programs such as $PAUP^*$ use heuristic procedures (in fact, the same heuristics used to estimate the maximum parsimony tree) to get close to the MLE optima, and in practice seem to converge in quadratic time. However, the proportionality constant can be very large due to the cost of evaluating the likelihood function. Furthermore, there can be multiple local optima even at the branch weight optimization step, and thus none of the currently available programs are guaranteed to find the globally optimal edge-weighting, of a fixed tree.

For many problems, maximum likelihood estimators are statistically consistent estimators. Unfortunately, consistency properties only hold for a rather restricted class of MLEs, and evolutionary tree estimators do not fall into this class. Therefore, the consistency of evolutionary tree MLE must be proved separately, for each evolutionary model class. As noted above, the sampling distribution of finite state characters follows the multinomial distribution. The convergence of a MLE to the "true" parameters (i.e., the probability of each character pattern) is easily established. The problem is that the probability of each character pattern is a function of the tree topology and edge weights. Therefore, we must also establish whether the tree topologies are identifiable in the sense previously discussed. As mentioned, we do not have a complete collection of results for identifiability.

4.3 Relationship between MLE and MP

Although MP is not statistically consistent under all CF trees, it nevertheless has provable statistical properties. Under a model in which the sites evolve independently but not identically, and there is no assumption of scaling across the sites, MP=MLE [131], in the sense that if the different leaf-labeled trees are rank ordered on the basis of the MP scores and also rank ordered on the basis of the ML scores, then the rank orderings are the same.

We now prove this.

Theorem 2 *Let k binary sites evolve randomly on a tree T with leaves labeled by S, and let the i^{th} site evolve under its own process (i.e. if we let $p(e,i)$ denote the probability of change of site i on edge e, then there is no restriction on the $p(e,i)$ other than that $0 < p(e,i) \leq 1/2$ for all e,i). Let T' and T'' be two different trees with leaves labeled by S. If T' is better than T'' with respect to its maximum likelihood score, then T' is better than T'' with respect to its maximum parsimony score.*

Proof: Let $p(e,i)$ denote the probability of change for site i on edge e. For any finite set of sequences labeling the leaves of a tree T', we show how to set $p(e,i)$ for a given tree T' to maximize the probability of observing the sequences. Let L be an optimal labeling of the internal nodes of the tree T' with respect to maximum parsimony. For a given edge e and site i, if site i changes on e (with respect to the labeling L) we set $p(e,i)$=1/2; otherwise we set $p(e,i) = 0$. It can then be proven that this setting for the substitution probabilities on the edges maximizes the probability of generating the observed sequences at the leaves, and hence solves the maximum likelihood estimation problem for this fixed tree. Furthermore, it is easy to see that the probability of observing the sequence data on T' is $(1/2)^{length(T')}$. Consequently, minimizing the parsimony length of T' is identical to maximizing the likelihood of T'. Furthermore, if T' is better than T'' with respect to maximum likelihood scores, then T' is better than T'' with respect to maximum parsimony scores. Thus, MP and MLE are identical under this model. $\square$

5 Distance-based methods

In this section we discuss some of the most promising distance-based methods that are used in systematic biology.

5.1 Basic concepts

Given a leaf-labeled tree T with positive edge weights, we can define the *path distance* between leaves i and j to be the sum of the weights of the edges in the path between i and j.

Definition 1 *A distance matrix D is* **additive** *if there exists a tree with positive edge weights such that $D_{ij} = \sum_{e \in P_{ij}} w(e)$, where P_{ij} denotes the path between leaves i and j in the tree, and $w(e)$ is the weight of edge e.*

The following theorem was proved in [125].

Theorem 3 *Given an additive $n \times n$ distance matrix d, there is a unique positive edge-weighted tree without nodes of degree two, in which n nodes in the tree are labeled $s_1, s_2, \ldots, s_n$, so that the path distance between s_i and s_j is equal to d_{ij}. Furthermore, the unique tree consistent with d is reconstructible in $O(n^2)$ time.*

We will call any symmetric matrix which is zero-diagonal and positive off-diagonal a **dissimilarity matrix**. A **distance method** d maps $n \times n$ dissimilarity matrices to $n \times n$ additive distance matrices.

There are many distance-based methods and optimization problems related to distance-based reconstruction. Because almost all optimization problems are NP-hard ([3, 36, 49]), almost all methods are based upon simple heuristics.

We now discuss two natural requirements we may wish to make of a distance method, which we will later prove will ensure that the method is a consistent estimator for inferring binary model trees. These two properties are *combinatorial consistency* and *local continuity*. We will say that a method M is **combinatorially consistent** if $M(D) = D$ whenever D is additive. This property is true of just about all distance methods that are in use today, except those that seek to reconstruct ultrametric trees (i.e. rooted trees in which the distance from the root to any leaf is the same). This property is also automatically true of any method which solves or approximates (with a performance guarantee) an optimization problem of the form "*given distance matrix d, find a nearest additive matrix D*", where by "nearest" we permit any metric between distance matrices to be used. Furthermore, if we define a metric on distance matrices, we may naturally define continuity with respect to that metric. For example, the L_∞ metric is defined by $L_\infty(d, d') = \max_{ij} |d_{ij} - d'_{ij}|$. A distance method M is then **continuous** at d (with respect to the L_∞ metric) if for all $\epsilon > 0$ there is a $\delta > 0$ such that $L_\infty(d, d') < \delta$ implies that $L_\infty(M(d), M(d')) < \epsilon$.

We will say that a distance method is **reasonable** if it is both combinatorially consistent and continuous on a neighborhood around every additive distance matrices corresponding to positively edge-weighted binary trees. Almost all methods used to reconstruct trees from distances are reasonable. The importance of being "reasonable" will be shown in Section 4.5, in which we will prove that any method which is reasonable" is guaranteed to be *consistent* for estimating binary trees.

Many of the methods used in practice are based upon *agglomerative clustering*. Agglomerative clustering is a basic technique which constructs a tree by successively deciding which pair of leaves should be siblings, thus reducing the size of the input in each step. The particular technique by which the siblinghood decision is made, and the way in which the distance matrix is then modified, distinguishes the different clustering methods. Some of the most popular methods used in practice, such as the $O(n^2)$ *Neighbor Joining* method (popularized by Saitou and Nei in [103]) and the $O(n^4)$ *Fitch-Margoliash* method [59], are based upon this technique. All of these methods, except those that reconstruct ultrametric trees, are "reasonable" and hence provably consistent estimators for binary trees.

Ultrametric trees are rooted and edge-weighted so that the distance from the root to every leaf is the same. Consequently, given an additive but not ultrametric distance matrix D, methods which reconstruct ultrametric trees will modify D, sometimes even changing its topology!

The reconstruction of ultrametric trees used to be popular among biologists when the "molecular-clock" hypothesis was accepted. This hypothesis asserts that mutations occur in a more-or-less clocklike fashion, so that differences between sequences should be proportional to the evolutionary time between the two sequences (i.e. to the time back to their most recent common ancestor). This is also expressed by saying that DNA sequences evolve at a constant rate across different lineages. The molecular-clock hypothesis has however been discredited, and there is mounting evidence that different lineages can evolve at unboundedly different rates, and that even mitochondrial DNA does not evolve at anything close to a constant rate. (See [117] for the original disproof of the molecular clock hypothesis, and [25, 63, 88, 89, 106] for other such results.)

Many new distance-based methods have been introduced, such as *BIONJ* [62], *Quartet Puzzling* [114], *the Short Quartet Method* [39, 40, 41] and *Agarwala's 3-approximation* [3] and its variant, the *Double-Pivot* [33]. However, these methods are not yet in use by the systematic biology community, and there has not yet been enough experimental performance analysis of these methods for their advantages and disadvantages on realistic data sets to be understood.

5.2 Statistical basis of distance-based methods

The idea behind distance-based methods is to compute distances between sequences so that these pairwise distances reflect the actual number of point mutations that occurred on the path between the leaves representing the two sequences. If this can be done so that the actual computed distances exactly equal the number of changes on the paths, then these distances are *additive* and hence can be used to reconstruct the evolutionary tree. Furthermore, as we have shown, reconstructing the underlying tree from additive matrices is easy to do in polynomial time.

There are two main hurdles in this basic approach. The first is that the distances must be computed appropriately, so that these distances will be proportional to the path distances in the true tree. Hamming distances clearly fail this test because of the "multiple-hits" phenomenon where two sequences are identical on a particular site, but that site has changed state on the path between the two sequences. It turns out that computing additive distances from finite length sequences is not really possible to do, but it is nevertheless possible to define distances so that as the sequence length gets longer, the computed distances more closely approximate additive distances.

Corrected distance transformations were invented for this purpose. A corrected distance transformation simultaneously:

- represents the model tree (i.e. leaf-labeled tree with information about the evolutionary process governing each edge) as an edge-weighted tree, and
- defines distances between sequences generated on the tree, so that the following holds: *as the sequence length increases, the matrix of observed distances converges to the additive distance defined by the edge weighted tree.*

Using such corrected distance transformations then ensures that a distance method can be consistent. Corrected distance transformations exist for the CF model and for the general Markov model. We now describe the corrected distance transformation for the CF model, and why it makes distance methods consistent.

Given a CF tree T and sequences of length k generated at the leaves of T, let $H(i,j)$ denote the *Hamming distance* of sequences i and j and $h^{ij} = H(i,j)/k$ denote the *dissimilarity score* of sequences i and j. The *corrected distance* between i and j is denoted by $d_{ij} = -\frac{1}{2}\log(1-2h^{ij})$ and the model probability of change of character state between the sequences i and j is denoted by E^{ij} (i.e. E^{ij} denotes the expected value of h^{ij}). We let $D_{ij} = -\frac{1}{2}\log(1-E^{ij})$ denote the *theoretical distance* between i and j, computed with E^{ij} instead of h^{ij}. If we assign to any edge e a positive weight $w(e) = -\frac{1}{2}\log(1-2p_e)$, then it follows from Equation (1) above that D_{ij} is exactly the sum of the weights along $P(i,j)$.

What we have shown is that a combinatorially consistent method applied to distances computed for *infinite length sequences* will with probability 1 reconstruct the correct topology. However, we never have infinite length sequences, so that we need to discuss whether the method attains the correct topology on some finite length sequences. For this to be true, we will need the continuity property. However, we will only be able to finish the proof after we establish conditions under which two additive matrices can be guaranteed to define the same topology. The next few sections will develop these results.

5.3 Buneman's Four-Point Condition

The following theorem of Buneman [27], called the *Four Point Condition*, provides a characterization of additive distance matrices which is of interest in its own right, and has several consequences for algorithm design. Before we give the four-point condition, we provide the following definition.

Definition 2 *Let $ij|kl$ denote the tree on leaves i,j,k,l in which the pair i,j is separated from the pair k,l by a path.*

The dyadic closure of a set X is defined by two rules.

Lemma 1 *(Four Point Condition [27]) A matrix D is additive if and only if for all i, j, k, l (not necessarily distinct), the maximum of $D_{ij} + D_{kl}$, $D_{ik} + D_{jl}$, $D_{il} + D_{jk}$ is not unique. The edge weighted tree (with positive weights on internal edges and non-negative weights on leaf edges) representing the additive distance matrix is unique among the trees without vertices of degree two.*

Proof: We only prove one direction, because it is easy and also illuminative. Suppose that D is additive so that there is a tree T with positive edge weights on the internal edges and non-negative edge weights on the edges incident with leaves, so that D_{ij} equals the path distance in T between i and j. Now consider a quartet i, j, k, l. These four nodes induce a subtree of T which is either a star or a resolved binary tree. It is easy to see that the four nodes induce a star if and only if the three pairwise sums are identical. In the case where the four nodes induce a binary tree in which i and j are separated from k and l by a path of positive weight, the smallest of the three pairwise sums will be $D_{ij} + D_{kl}$, while the other two pairwise sums will be identical. □

The sketch of the proof we have just described actually indicates that $D_{ij} + D_{kl} < D_{ik} + D_{jl} = D_{il} + D_{jk}$ if and only if the topology of the subtree of T induced by i, j, k, l is $ij|kl$. Consequently, if the distance matrix is additive, then the topology of the model tree can be obtained by simply inferring the topology of every quartet. It is then straightforward to construct the tree topology, since siblinghood of leaves (and subsequently of subtrees) can be easily inferred, and once the tree topology is reconstructed, the edge-weights realizing the distance method can also be obtained by solving linear equations. This is just one of many polynomial time methods for reconstructing the unique positively edge-weighted tree realizing the distance matrix (though this particular method uses much more time than the other methods!).

However, distances calculated and appropriately corrected from finite length sequences generated on a model tree are not actually additive, even though these distances do (with probability 1) converge to the additive distance defining the model tree. Consequently, the real issue is whether we can infer the model tree from distances that are close to but not identical to the additive matrix defining the model tree.

5.4 Topology Invariant Neighborhoods and Consistency

Since distance methods must be applied to nonadditive distance matrices, it is relevant to consider whether a method can return the topology of the model tree even when the distances are not additive. In order to answer this question, we consider the question of when two different additive matrices define the same topology. All of the results in this section are from [39, 40, 41].

Lemma 2 *Two additive distance matrices D and D' define the same topology if and only if for every quartet i, j, k, l, $D_{ij} + D_{kl}$ is the minimum of the three pairwise sums if and only if $D'_{ij} + D'_{kl}$ is the minimum of the corresponding three pairwise sums.*

Proof: First, note that the topology of a tree is defined by the topology the tree induces on every quartet of leaves in the tree. Given this observation, we note that Buneman's four-point condition shows that the topology of any quartet i, j, k, l can be inferred by examining the three pairwise sums, $D_{ij} + D_{kl}, D_{ik} + D_{jl}, D_{il} + D_{jk}$. The minimum of these three pairwise sums is $D_{ij} + D_{kl}$ if and only if the topology on i, j, k, l in the tree is $ij|kl$. Therefore, two additive distance matrices define the same topology if and only if they impose the same ordering on such pairwise sums. □

A surprising consequence of this theorem is that there is a positive neighborhood around each additive distance matrix defining a *binary* tree (i.e. all nodes of degree 3), on which *all* additive distance matrices define the same topology.

Lemma 3 *Let D be an additive distance matrix defining an edge-weighted binary tree T, and let x be the weight of the smallest edge in T. Let D' be another additive distance matrix defining a (not necessarily binary) edge-weighted tree T'. If $L_\infty(D, D') = max_{ij}|D_{ij} - D'_{ij}| < x/2$ then the topologies of T and T' are identical.*

Proof: It suffices to note that if $D_{ij} + D_{kl}$ is the minimum of the three pairwise sums, then it is less than the other two sums by at least $2x$. If $L_\infty(D, D') < x/2$, then $D'_{ij} + D'_{kl}$ is also less than both of $D'_{ik} + D'_{jl}$ and $D'_{il} + D'_{jk}$. Consequently, D and D' define the same topology (although with different edge weights). □

An immediate consequence of this theorem is the following:

Theorem 4 *All combinatorially consistent distance methods which are continuous at additive distance matrices defining binary trees are consistent methods for inferring topologies of binary model trees T.*

Proof: The proof is straightforward. Suppose that T is a binary model tree, x is its smallest edge weight, and M is a method which is both combinatorially consistent and continuous at additive matrices for binary trees (i.e. "reasonable"). Since M is continuous, then there is some δ such that if d satisfies $L_\infty(d, D) \leq \delta$, then $L_\infty(M(D), M(d)) < x/2$. Since M maps distance matrices to additive distances, under these conditions $M(d)$ is guaranteed to have the same topology as $M(D)$. Since M is combinatorially consistent, $M(D) = D$. Consequently $M(d)$ is an additive matrix which defines the topology of the model tree T. □

It is worth noting that all of the standard distance-based methods (with the exception of those that explicitly seek to reconstruct ultrametric trees) are continuous at binary trees, and combinatorially consistent, and hence are consistent methods for inferring binary evolutionary trees, but little is understood about the convergence rate of these methods.

However, the proof of this theorem establishes a mechanism by which the convergence rate of different methods can be compared: For each method M and for each binary model tree T and the additive matrix D defined by T, there is some positive δ such that given a distance matrix $d_0 \in N(D, \delta) = \{d : L_\infty(d, D) < \delta\}$, $M(d_0)$ is guaranteed to be an additive distance matrix having the same topology as T. It then follows that the larger the maximum δ for which this is true, the easier it is for a method to be guaranteed to obtain an accurate topology.

5.5 Ultrametric Tree Reconstruction

If distances computed on the basis of differences between biomolecular sequences are proportional to time since the sequences split off from a common ancestor (the "molecular clock" hypothesis), then a special kind of edge-weighted tree, called an "ultrametric tree", is an appropriate model of evolution. **Ultrametric** trees are rooted edge-weighted trees in which the distances from the root to any two leaves are the same.

However, as we have discussed earlier, the molecular clock hypothesis is now generally discredited, and the reconstruction of ultrametric trees is no longer generally considered relevant to biomolecular evolutionary studies.

However, there are two reasons to discuss ultrametric tree reconstruction: first, there are some very nice algorithms which have been developed for obtaining optimal solutions to problems related to ultrametric tree reconstruction, and second, these algorithms have been shown to be useful in *approximating* the nearest fitting additive tree. We describe the algorithms for ultrametric tree reconstruction in this section, and show in the next section how they can be used to approximate additive trees.

We have already noted that optimization problems in distance-based reconstruction are typically NP-hard. When the desired tree is constrained to be ultrametric, it is however possible that the problem's complexity could become tractable, but in almost all cases, optimization problems for reconstructing ultrametric trees are still NP-hard [3, 49]. One notable exception is the problem of finding the nearest ultrametric distance matrix (i.e. distance matrix fitting an ultrametric tree) to a given distance matrix, with respect to the L_∞-criterion.

The first result of this type is due to Gower and Ross [64], who proved the following:

Theorem 5 *Given distance matrix d, there is a unique ultrametric distance matrix D satisfying*

- *1. D is* dominated by *d (i.e. $D[ij] \leq d[ij]$ for all i, j), and*
- *2. D dominates all other ultrametric distance matrices which are dominated by d (i.e. if D' is ultrametric and $D'[ij] \leq d[ij]$ for all i, j, then $D'[ij] \leq D[ij]$).*

Their proof was constructive: given d:

- Step 1: weight the complete graph on $1, 2, \ldots, n$ by the matrix d; i.e. $w(i, j) = d_{ij}$,
- Step 2: construct a minimum spanning tree T on K_n
- Step 3: define the ultrametric matrix U by letting $U[ij]$ be the maximum weight on the edge in T between i and j.

This algorithm produces an ultrametric tree which is called the *subdominant* ultrametric of the matrix d.

Surprisingly, the same tree can be constructed in a greedy *agglomerative* fashion, through the "Single-linkage algorithm," which takes as input a distance matrix d and computes a rooted tree whose edges can be weighted appropriately to obtain the subdominant ultrametric of d.

5.6 Single-linkage algorithm:

- 1. Begin with all taxa (leaves) in their own classes, and set the distance between two classes x, y to be $d(x, y)$.
- 2. While there is more than one class, DO:
  - Choose the classes C_1 and C_2 minimizing the quantity $d(C_1, C_2)$.
  - Join the subtrees for C_1 and C_2 into one rooted subtree, by making their roots children of the same root.
  - Create the class $C = C_1 \cup C_2$, and define $d(C, C') = min(d(C_1, C'), d(C_2, C'))$ for all classes $C' \neq C_1, C_2$.
  - If $C = S$ then return tree, else delete C_1 and C_2.

This algorithm [107] can be easily modified to assign weights to the edges of the tree so as to define an ultrametric tree, and it is easy to see (see [72] for the first such statement of this observation) that the topology this method reconstructs is the same as the topology obtained by Gower and Ross's algorithm for the subdominant ultrametric.

These algorithms were rediscovered in a later work by Farach, Kannan, and Warnow [48] in the context of a more general problem:

5.7 Matrix sandwich problem:

Given two distance matrices M_l and M_h which represent lower and upper bounds respectively, determine if there is an ultrametric matrix U satisfying $M_l[ij] \leq U[ij] \leq M_h[ij]$.

This is a general class of problems since the matrix sought can be less constrained (i.e. we may seek an additive tree only, or an ultrametric satisfying additional constraints). The following then is obvious:

Theorem 6 *Let M_l and M_h be two $n \times n$ distance matrices. The following are equivalent:*

- *1. There is an ultrametric matrix U such that $U \in [M_l, M_h]$, and*
- *2. It is possible to weight the edges of the tree obtained by the Gower-Ross algorithm applied to distance matrix M_h so as to obtain an ultrametric matrix $U' \in [M_l, M_h]$.*

This theorem implies that a simple algorithm can determine whether there is an ultrametric matrix in an arbitrary sandwich, and the algorithm uses only polynomial time. This formulation has some nice properties, since in general it means that many different optimization problems related to obtaining nearest ultrametric trees can be solved in polynomial time, for various definitions of "nearest."

5.8 Approximation algorithms for nearest trees

Various approximation algorithms for obtaining "nearest" additive trees to a given distance matrix have been developed, all of which are based upon a fundamental observation relating ultrametric trees and additive trees. To explain this relationship we first define what a *centroid metric* is.

A **centroid metric** is an additive metric which can be realized by edge-weighting a star topology (i.e. a tree with exactly one non-leaf node).

The critical observation relating centroid metrics, additive metrics, and ultrametrics, was first observed by Farris, and communicated to Carroll who published it in 1976 in [29]:

Theorem 7 *Let D be an additive matrix, and let X be a centroid matrix. Then $D + X$ is an ultrametric matrix.*

This suggests a general strategy for reconstructing nearby additive metrics to given distance metrics, which we now describe.

The basic idea: If d is a matrix, D^{opt} is the nearest additive metric to d (under some optimization criterion). Let X be a centroid metric. Then $U^* = D^{opt} + X$ is an ultrametric matrix, and since D^{opt} is near to d, it may be that U^* is close to $d + X$. If we could get from $d + X$ to U^* we could then easily obtain D^{opt} from U^* by subtracting X. Hence, the problem in some sense "reduces" to finding a nearest ultrametric to $d + X$, for some suitably selected centroid metric X.

To summarize, the observation that an additive metric decomposes into the sum of a centroid (which can be arbitrarily selected) and an ultrametric suggests a *general algorithmic strategy*:

5.9 General algorithmic strategy for obtaining nearby additive metrics:

- 1. Given distance matrix d, compute a centroid metric X, and compute distance metric $d' = d + X$.
- 2. Use some method to find an ultrametric U which is close to d'.
- 3. Compute the additive metric $D = U - X$, and reconstruct the edge-weighted tree T realizing D. Return T.

This basic approach was possibly first discovered by Blanken *et al.* in 1982 [20], also used by Brossier in 1984 [24], and again used by Agarwala *et al.* in 1996 [3]. The major contribution of the Agarwala *et al.* paper was the observation that if the basic method were implemented by using the Single-Linkage algorithm and the topology obtained were correctly weighted, then the resultant additive matrix would be *guaranteed* to be no more than three times as far from the input matrix than the nearest additive matrix, *with respect to the L_∞ metric.* In other words, Agarwala and her colleagues showed that the basic approach could be implemented to produce a 3-approximation for the nearest tree, with respect to the L_∞-metric.

5.10 Reconstruction based upon combining subtrees

5.11 Introduction

One general technique that can be used to reconstruct an evolutionary tree is to reconstruct all subtrees of a given size, and then combine these subtrees into one tree.

An unrooted leaf-labeled tree can be defined by the topology it induces on the quartets of leaves in the tree. Thus, one approach to reconstructing evolutionary trees is to determine (using some technique) the topology on every quartet of leaves, and then combine these quartets if possible into one tree consistent with the entire set. It is easy to see that if all the quartets are consistent, it is easy to reconstruct the (unique) tree consistent with the constraints in polynomial time. However, quartet topologies are not always consistent, so that each quartet-based method must also specify a means for resolving inconsistencies or else permit the output of *inconsistent* to be returned on those inputs for which some topology is inaccurately reconstructed.

In essence, then, a quartet method takes as input a set Q of topologies on quartets, and determines a tree from this set. One problem with quartet-based approaches is that some quartets are simply harder to estimate than others (see [70] for a study of quartet estimation). Thus, one quartet-based approach is to take the quartets one has confidence in, and use those only to reconstruct the tree. Unfortunately, consistency of a set of quartets with a tree is in general NP-complete [111] (although the case where the set contains topologies for all quartets is solvable in polynomial time). Thus, quartet methods generally use all the possible quartets and specifically identify a heuristic step for handling inconsistencies, but have the flexibility to allow any method whatsoever for reconstructing trees on quartets.

These methods are historically popular though they have been replaced by the faster and possibly more powerful methods (such as neighbor-joining) introduced in recent years. Recently however there have been new quartet-based methods introduced which have very nice properties and interesting performance in experimental studies. Before we discuss the more sophisticated quartet based methods, we begin with the simplest of all possible methods, which we call the *Naive Quartet Method.*

5.11.1 The Naive Quartet Method

Consider the following quartet based method. For every set i, j, k, l, select the topology $ij|kl$ if and only if $D_{ij} + D_{kl} < \min(D_{ik} + D_{jl}, D_{il} + D_{jk})$. If all the quartet topologies can be simultaneously realized in a single tree, then that tree can be reconstructed in polynomial time (simply determine siblinghood of pairs of leaves, and then of subtrees, and hence reconstruct the tree "from the outside-in").

The problem with this approach is that it may happen that one of the $\Omega(n^4)$ quartets may be incorrectly inferred, especially if the tree contains widely separated pairs of taxa, or very short edges. Nevertheless, this method is combinatorially consistent (i.e. it satisfies $M(D) = D$, when D is additive) and continuous at binary trees, and hence it is a consistent method for reconstructing binary evolutionary trees.

Most (but not all) quartet-based methods begin in essentially the same way as the Naive Method, in that they infer the topology of every quartet (using some method, typically a distance-based reconstruction, but sometimes other techniques are used) and then reconstruct the tree from the set of quartets. Since typically some quartets will be incorrectly estimated, most quartet-based methods must provide a mechanism for handling incompatible sets of quartets.

5.11.2 The Buneman Tree

One of the classical quartet-based approaches is the Buneman tree, suggested by Buneman in [27], as follows:

The topology of every quartet is inferred using the same basic approach as the Naive Method. This defines a set of quartet topologies, Q. If all the topologies in Q are simultaneously realizable with one tree, we return that tree. Otherwise, we seek a tree in which "every edge is supported by Q". We now define what this means.

Consider a bipartition of the leaves S into two sets A and B defined by an edge of a tree T. We will say that this bipartition is completely supported by a set Q of quartet topologies if for every $\{a, a'\} \subseteq A$ and $\{b, b'\} \subseteq B$, the topology on a, a', b, b' defined by Q separates a, a' from b, b'.

Buneman's approach [27] was to reconstruct the tree which contained *every* bipartition that was supported by Q. Although there are exponentially many possible bipartitions, the set of bipartitions that are completely supported by a set of topologies for all of the possible quartets is compatible, and hence defines a unique tree (see [18]). Furthermore, reconstructing the unique tree containing all the completely supported bipartitions can be accomplished in polynomial time; an $O(n^5)$ algorithm to reconstruct the **Buneman Tree** has been implemented in the SplitsTree phylogenetic software package (available at `ftp://ftp.uni-bielefeld.de/pub/math/splits`). A faster $O(n^4)$ algorithm has also been obtained by Berry and Gascuel [18].

5.11.3 Split Decomposition

An interesting quartet-based method was developed by Bandelt and Dress, which has a different objective than other methods. This method is the *split decomposition* method [12, 13] to construct networks of relationships. The basic idea in this method is the observation that observed sequence or distance data may not support a single tree, but rather may support a *set* of trees. Thus, for example, Buneman showed that additive distance matrices D are characterized as having the four-point condition: *for all* i, j, k, l, *the maximum and median of* $D_{ij} + D_{kl}, D_{ik} + D_{jl}, D_{il} + D_{jk}$ *are the same*; consequently the smallest of the three pairwise sums indicates the topology. But what happens if the distance matrix is not additive? Perhaps the three numbers are distinct, and the smallest is so close to the median that it is hard to determine with any confidence the actual topology of the quartet? In that case, the most accurate representation of the topology on the quartet i, j, k, l is that it is *either* of one of two trees, but not the third. Bandelt and Dress showed that it was possible to effectively represent such ambiguities using *"networks"* (which are really just graphs), and they developed software for representing these networks as planar graphs so that identifying sets of parallel lines would yield the different trees that are consistent with the data. This method and its variants has been implemented and constitutes the *SplitsTree* phylogenetic software package at `ftp://ftp.uni-bielefeld.de/pub/math/splits`.

6 Convergence proof techniques

As we have seen, there are many methods for reconstructing trees. In this section we will describe the techniques used to predict their performance under different model trees in the Cavender-Farris model of evolution. Until recently, very little was known about performance of estimators under the CF model, except whether each converged or did not converge to the true tree under these models, as the sequence length increases. However, in the last few years, we now have established very general techniques for proving convergence and bounding the convergence rates of distance-based methods.

6.1 Techniques for proving convergence to CF trees

We begin with some basic results, whose proofs we omit.

Lemma 4 *Let $(T, \{\lambda_e\})$ be a CF tree, with $0 < f \leq \lambda_e \leq g$. Define*

$$L_\infty^{(q)}(d, \lambda) = \max\{|d_{i,j} - \lambda_{i,j}| : \min\{\lambda_{i,j}, d_{i,j}\} < q\}.$$

For every $\epsilon, \delta > 0$, there exists a constant C (depending upon ϵ and δ) such that if $k \geq C \log n e^{O(q)}$, then $Pr[L_\infty^{(q)}(d^k, \lambda) < \delta] > 1 - \epsilon$.

This suggests that the estimates of small distances converge faster to their true values than the estimates of large distances. It may also suggest that the error in the estimations of small distances may be smaller then the error in the estimation of a large distance.

6.1.1 General Technique for Proving Convergence of distance methods

The technique we use is the following:

Let $(T, \{\lambda_e\})$ be a CF tree with $0 < f \leq \lambda_e \leq g$ for all edges e. Let M be a fixed phylogenetic method.

- Show $\exists \delta > 0$ so that if d satisfies $L_\infty(d, \lambda) = \max_{i,j} |d_{i,j} - \lambda_{i,j}| < \delta$, then $M(d) = T$.

Then Lemma 4 implies there exists a constant C dependent upon ϵ and δ such that if $k \geq C \log n e^{O(\max \lambda_{ij})}$ then $Pr[M(d^k) = T] \geq 1 - \epsilon$. Hence M converges. However, this does not prove fast convergence. Then $\max \lambda_{i,j} = O(g \cdot diam(T))$, where $diam(T)$ is the number of edges in the longest path in the tree. Erdös, Steel, Szekely, and Warnow then showed:

- $diam(T) \leq n - 1$, and
- $diam(T) = \Omega(\sqrt{n})$ for random trees under the uniform distribution.

This technique has been used to prove convergence, although not fast convergence, for several distance methods:

1. The Buneman and Naive Quartet Methods are accurate when $\delta = f/2$ [39,40,41].
2. The Agarwala *et al.* (SODA '96) algorithm is accurate when $\delta = x/8$ [48].
3. The Neighbor-Joining Method is accurate when $\delta = x/2$ [11].

The largest δ can be is $f/2$, and generally δ is of the form f/p for some constant $p \geq 2$. Hence, the convergence rate has a dependence upon f. We present this analysis technique on the simplest of the methods, the Naive Quartet Method (see Section 5.11.1).

Lemma 5 *Let M be an additive matrix associated to tree T with edge-weighting w and $f = \min w(e)$. If d is an $n \times n$ dissimilarity matrix and $L_\infty(d, M) < f/2$, then the Four Point Method is correct on every quartet, and hence the Naive Quartet Method applied to d reconstructs T.*

Proof: By the four-point condition, when $L_\infty(d, M) < f/2$, then for all i, j, k, l, both d and M induce the same ordering on the three pairwise sums. The proof then follows since a tree is defined by its induced quartet subtrees. □ Thus, the Naive Quartet Method is a statistically consistent method for CF tree reconstruction, by Lemma 4.

6.2 Extension to other models

These results also extend to other models, for which there is a statistically consistent technique for estimating leaf-to-leaf distances. For example, consider the rates-across sites assumption. Under these models, if the distribution of rates across sites is known, then the same positive results hold, as these depend only upon the ability to obtain arbitrarily good estimates of the path distances in the tree with high probability, given long enough sequences. Thus, models with gamma-distributed rates across sites, where the shape parameter is given, are just as easily inferred as the Jukes-Cantor model trees.

For some other models of evolution, it is also possible to estimate leaf-to-leaf distances arbitrarily well, given long enough sequences, and so the same results can be established for a larger range of models. However, it is not the case that these distance estimations exist for every model. See [115] for a discussion of the existence of such distance estimation techniques under a wider class of evolutionary models.

7 Fast Converging Methods

7.1 Fast Convergence

We now describe the concept of "fast convergence" under the CF reconstruction problem. Although our description will be based upon the assumption that the data are generated by a fixed but unknown CF tree, similar statements can be made about the primary objective of tree reconstruction under other models.

Given a set S of sequences that have been generated by an unknown CF tree $(T, \{\lambda_e\})$, and $\epsilon > 0$, recover the unrooted leaf-labeled tree T with probability at least $1 - \epsilon$

- in time that is polynomial in n (the number of leaves) and k (the length of the sequences),
- from sequences of lengths that are polynomial in n (for fixed $f = \min_e \lambda_e$ and $g = \max_e \lambda_e$), and
- without knowing the $\{\lambda_e\}$.

A method which converges from polynomial length sequences is said to be **fast converging**.

Note that the input to the method does not include any information about the values of the λ_e, other than that they are positive; nor any constraints about the tree T other than that it is binary. At first glance recovering T without additional knowledge about the parameters of T might seem impossible to do with any degree of confidence; however, there are only a finite number of different unrooted tree topologies, and hence there is a potential to be able to recover this tree exactly, with probability going to 1 as the sequence length increases. By contrast, exactly recovering the values λ_e as well cannot be established with high probability except from infinite length sequences, unless the set of permitted λ_e is discretized.

As we will show, it is possible, even using simple polynomial time methods, to recover all CF trees with arbitrarily high probability, given long enough sequences. Recovering them from polynomial length sequences is harder, and requires more complex algorithms, but here too polynomial time methods can be developed.

7.2 Introduction

The challenge we have is how to prove (or obtain) fast convergence. The previous technique cannot prove fast convergence, because it relies upon *all* distance estimates being sufficiently accurate, and the convergence rate for this is likely to be tight. However, this does not mean that the particular methods *are not fast converging!* They may be, but this technique will not establish it. To prove fast convergence, we may need to show that a *different* condition suffices to guarantee accuracy for a method, and that this condition will hold with high probability from short sequences.

Consider Lemma 4: Estimates of small distances converge faster to their true values than estimates of large distances. This suggests that if we recover the tree without using the large entries in the distance matrix, then we may be able to define a fast converging method. As we will show, the fast converging methods that have been developed exploit this observation.

7.2.1 Short Quartets

We begin with the basics underlying the first fast converging method, the Dyadic Closure Method; see [39,40,41] for this exposition.

Definition 3 *Let D be the additive matrix for the tree T. The* **D-width** *of a quartet i, j, k, l is* $\max\{D_{i,j}, D_{i,k}, D_{i,l}, D_{j,k}, D_{j,l}, D_{k,l}\}$. *A* **short quartet** *around an edge e is a quartet with leaves in each of the four subtrees around that edge of minimum D-width. The set of all quartets that could have been selected in this manner is the set of short quartets of a tree, and the set of trees induced in T by the short quartets is denoted* **SQT(T)**. **D-width***(T) is the maximum D-width of any short quartet in T. Given d, a dissimilarity matrix,* **d-width***(T) is the maximum d-width of any short quartet in T.*

We now show that $SQT(T)$ defines T. The proof is constructive, First note that the set $Q(T)$ of four-leaf induced subtrees of any tree T defines the tree T. We now describe an algorithm (the Dyadic Closure Method) which produces $Q(T)$ when applied to the set $SQT(T)$.

Dyadic Closure

- **Input:** Set X of trees on four-leaves
- **Output:** $cl(X)$, the *dyadic closure of* X

Recall the definition of the four-taxon tree $ij|kl$. Then the dyadic closure of the set X is defined on the basis of two rules, which we now describe.

- **Rule 1:** Given two quartet trees $ij|kl$ and $jk|lm$, infer the quartet trees $ij|km, ij|lm$ and $ik|lm$.
- **Rule 2:** Given two quartet trees $ij|kl$ and $ij|km$, infer the quartet tree $ij|lm$.

Given these two rules, the dyadic closure $cl(X)$ of a set X is the minimal set of quartet trees which contains X and is closed under Rule 1 and Rule 2. It follows that $cl(X)$ is unique and can be reconstructed in $O(n^5)$ time.

Theorem 8 *Let T be a fixed edge-weighted tree, and let $SQT(T)$ denote the set of trees induced by the short quartets of T. Let $Q(T)$ denote the set of four-leaf trees in T.* If $SQT(T) \subseteq X \subseteq Q(T)$ then $cl(X) = Q(T)$.

Proof: The proof is by induction on the number of leaves in T. If T has four leaves, the statement is true. Now assume the statement is true for all trees on $n-1$ leaves. Let a and b be a sibling pair of leaves, and let a be the nearest of the two leaves to their common parent. (If they are equidistant from their common parent, then select a at random.) It is easy to verify that $SQT(T - \{b\}) \subset SQT(T)$. Also, a and b are in some short quartet together, and hence there exists $ab|cd \in SQT(T)$. Therefore,

$$cl[cl(SQT(T - \{b\})) \cup \{ab|cd\}] \subseteq cl(X).$$

By induction, $cl(SQT(T - \{b\})) = Q(T - \{b\})$. The proof then follows from the observation that $cl(Q(T - \{b\}) \cup \{ab|cd\}) = Q(T)$. □

Corollary 1 *T can be reconstructed from $SQT(T)$ in polynomial time.*

7.2.2 The Dyadic Closure Method

We can now define the Dyadic Closure Method. The input to the Dyadic Closure Method is a dissimilarity matrix d, and the output is either a tree or *Fail*.

Algorithm

Binary search over the $q \in d_{ij}$:

1. Use the Four-Point Method to compute a tree for each set of four leaves with d-width bounded by q. Let A_q denote the set of trees (computed using the Four-Point Method) on each quartet of $d-$width $\leq q$.

2. Compute $cl(A_q)$, the *Dyadic Closure* of A_q. There are three cases to consider:

   - If $cl(A_q)$ contains exactly one tree on *every* quartet, then return T such that $Q(T) = cl(A_q)$ (polytime and unique tree T exists).
   - If $cl(A_q)$ does not contain a tree on every quartet, increase q.
   - If $cl(A_q)$ contains two trees on some quartet, then decrease q.

If all q fail to return a tree, then return *fail.*

Theorem 9 *Let d be a fixed $n \times n$ dissimilarity matrix, and let $\{\lambda_{i,j}\}$ be an $n \times n$ additive matrix associated to the tree T with edge-weighting λ_e. Let $f = \min \lambda_e$. Then the Dyadic Closure Method returns T if $L_\infty^{(d-width(T))}(d, \lambda) < f/2$.*

Proof: Let $q' = d - width(T)$. If $L_\infty^{(q')}(d, \lambda) < f/2$, then $SQT(T) \subseteq A_{q'} \subseteq Q(T)$. By Theorem 7, $cl(A_q) = Q(T)$. Also, it is easy to see that when $X \subset Y$ then $cl(X) \subseteq cl(Y)$. Consequently, the set $[0, \max d_{ij}]$ can be divided into three intervals:

- An initial interval, A, in which $cl(A_q) \subset\neq Q(T)$ for all $q \in A$.
- A middle interval,B, in which $cl(A_q) = Q(T)$ for all $q \in B$, and
- A final interval, C, in which $Q(T) \subset\neq cl(A_q)$ for all $q \in C$.

By assumption $q' \in B$, so that $B \neq \emptyset$. Therefore, the binary search is guaranteed to examine some entry $q \in B$, and hence to construct the tree T. □

Theorem 10 *The Dyadic Closure Method is $O(n^5 \log n)$ time and fast-converging for CF tree reconstruction. Furthermore, polylogarithmic length sequences suffice for accuracy with high probability for random CF trees.*

Proof: We provide a brief sketch. Let $(T, \{\lambda_e\})$ be a Cavender-Farris tree on n leaves, $0 < f \leq \lambda_e \leq g$, and $\epsilon > 0$ be given.

Establishing the running time is easy. To prove that the Dyadic Closure Method is fast converging, let $q = \lambda$-width(T)+$f/2$. We show that Dyadic Closure(d) $= T$ if $L_\infty^{(q)}(d, \lambda) < f/2$. We then show that $q = O(g \cdot \log n)$ for all trees, so that by Lemma 4, the Dyadic Closure Method is fast converging. Finally, we show that random trees have $q = O(g \cdot \log \log n)$, so that by Lemma 4, the Dyadic Closure Method converges from *polylogarithmic* length sequences on random CF trees. □

7.3 General technique for proving fast convergence

Given a method M and a CF tree, $(T, \{\lambda_e\})$, with $0 < f \leq \lambda_e \leq g$ for all edges e, we wish to show that M obtains the tree T with probability at least $1 - \epsilon$ if the sequence length k is as large as a polynomial in n. In the previous section we proved that the Dyadic Closure Method is fast converging. In this section, we generalize the technique used to prove the Dyadic Closure Method is fast converging.

- Show $\exists \delta > 0$ and $q = O(g \log n)$ so that if a dissimilarity matrix d satisfies $L_\infty^{(q)}(d, \lambda) < \delta$, then $M(d) = T$.
- Lemma 4 proves M is fast converging.

7.4 The Fast Converging Methods

There are now four methods known to be fast converging and whose proofs use this technique. These methods are:

1. The *Dyadic Closure Method* [40] is $O(n^5 \log n)$ time
2. The *Witness-Antiwitness Method* [41] is $O(n^4 \log n)$ time, but on most trees will be faster

3. The *Harmonic Greedy Triplets* method [149] is $O(n^3)$ time

4. The *Disk-Covering Method* [148] is a general phylogenetic method *booster*, so that it is used with other phylogenetic methods. It is not polynomial time, but performs very well in practice. The DCM-Buneman and DCM-Naive Quartet methods are both fast converging, although not polynomial time.

7.5 The Disk-Covering Method

All four of the fast converging methods have the same theoretical performance guarantees, but the fourth of these methods has a different flavor, and is of interest in its own right because it can be used for a variety of purposes. This technique [148] is a *meta-method* for phylogenetic tree reconstruction.
Given fixed phylogenetic method, M, which will be called the "base method".

- **Phase I:** For every $q \in \{d_{ij}\}$, DO:
  1. compute *threshold graph* $d^w = (S, E_q)$, where $(i,j) \in E_q$ iff $d_{ij} \leq q$
  2. let G^q be a triangulation of d^q (minimizing, if possible, the weight of the largest edge added). Furthermore, we can require, without loss of generality, that G^q is a supergraph of $G^{q'}$ if $q' \leq q$.
  3. Compute all maximal cliques $C_1, C_2, \ldots, C_t$ in G^w, and compute trees $T_i = M(C_i), i = 1, 2, \ldots, t$ (note $t \leq n$).
  4. Combine trees $T_1, T_2, \ldots, T_t$ into one tree, T^w.
- **Phase II:** Compute the *consensus tree* T of the trees T^w, and return T.

Several portions of the technique are flexible and can be modified to obtain improved performance for different phylogenetic methods. In particular: the triangulation of the threshold graph, the technique used for combining subtrees, and the consensus tree technique, can each be modified.

7.5.1 DCM-Naive Quartet

Some DCM-boosted methods can be proven to be fast converging. One such example is the DCM-Naive Quartet method. For fixed q, construct and triangulate the threshold graph, minimizing the weight of the

heaviest edge added.

1. The base method is the Naive Quartet Method. If the Naive Quartet Method fails to return a tree on any subproblem, then return *fail*.

2. Merge subtrees using the strict consensus merger (see Figure 1).

3. Having computed all T_q, select the most resolved tree (the tree with the most internal edges). If more than one maximally resolved tree, then select the highest indiced such tree.

Theorem 11 *DCM-Naive Quartet Method is a fast converging method.*

8 Open Problems

We now describe some of the many open problems that remain, and where progress might reasonably be made in the short term:

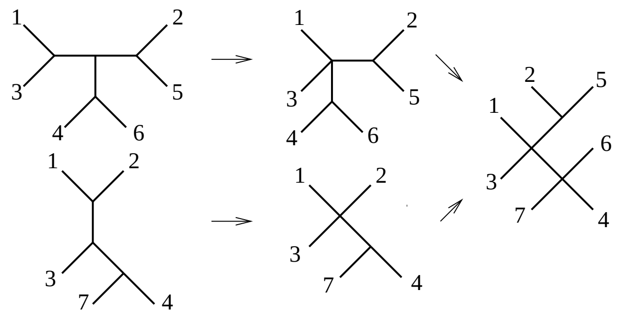

Figure 1: Merging two trees together, by first transforming them (through a minimal set of edge contractions) so that they induce the same subtrees on their shared leaves.

1. We have mathematical theory establishing upper bounds on the sequence lengths that suffice for accurate tree reconstruction with high probability, for a number of methods under a number of stochastic models of sequence evolution. These bounds are, in some cases, quite loose, as experimental studies indicate much faster convergence rates than are predicted by theory. Furthermore, we have very little theory establishing the rate at which the errors decrease to 0 for many statistically consistent methods, so that we cannot predict, mathematically, what the performance will be on sequence lengths that are too short to guarantee 100% accuracy.

   - Can we establish tighter bounds on the sequence lengths that suffice for topological accuracy for statistically consistent distance methods, under standard models of evolution? In particular, the experimental performance of the neighbor-joining (NJ) method is dramatically better than that of the naive quartet method, yet both have the same theoretical guarantees. Is NJ fast converging?
   - Can we establish bounds on the degree of topological error obtained by various phylogenetic methods, when the sequences are too short for 100% accuracy to be guaranteed?
   - Are there other techniques for establishing fast convergence, or other techniques for bounding the convergence rate, which could help us obtain better bounds on the convergence rates of important phylogenetic methods? In particular, can we establish a technique for analyzing the convergence rate of methods other than distance-based methods?

2. We have very little mathematical theory about maximum parsimony, except for the theory which indicates the conditions under which maximum parsimony will be statistically inconsistent. Yet simulation studies indicate that maximum parsimony often has very low error rates, by comparison to many statistically consistent methods, under realistic conditions. Can we establish any mathematical theory about maximum parsimony?

3. The techniques used for establishing fast convergence suggest optimization problems based upon fitting dissimilarity matrices to additive matrices, but considering only the entries below fixed constants q. For example:

   Given $n \times n$ dissimilarity matrix d, $q \geq 0$, find additive matrix D minimizing $L_\infty^{(q)}(d, D)$.

   For q unbounded, this is equivalent to the NP-hard L_∞-nearest tree problem, but it is possible that for q small, the problem may be polynomial. Even if the problems of this sort are NP-hard, it may be possible to obtain good approximations of optimal solutions in polynomial time.

References

1. Adams, E. III (1972). *Consensus techniques and the comparison of taxonomic trees*, Syst. Zool., 21: 390-397.

2. Adams, E. III (1986). *N-trees as Nestings : Complexity, Similarity, and Consensus*, Journal of Class., 3: 299-317.

3. Agarwala, R., Bafna, V., Farach, M., Narayanan, B., Paterson, M., and M. Thorup (1996). *On the approximability of numerical taxonomy: fitting distances by tree metrics.* Proceedings of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 365-372.

4. Agarwala, R. and D. Fernández-Baca (1996). *Simple Algorithms for Perfect Phylogeny and Triangulating Colored Graphs.* In the special issue on Algorithmic Aspects of Computational Biology, International Journal of Foundations of Computer Science, Vol. 7, No. 1, pp. 11–21.

5. Agarwala, R. and D. Fernández-Baca (1993). *A Polynomial-Time Algorithm for the Perfect Phylogeny Problem when the Number of Character States is Fixed.* SIAM Journal on Computing, Vol. 23, No. 6, pp. 1216–1224, December 1994. Also available in Proceedings of the 34th Annual Symposium on Fundamentals of Computer Science, pp. 140–147.

6. Agarwala, R., D. Fernández-Baca, and G. Slutzki (1995). *Fast Algorithms for Inferring Evolutionary Trees.* Journal of Computational Biology, Vol. 2, No. 3, pp. 397–408. An earlier version of this paper is available in Proceedings of the 30th Allerton Conference on Communication, Control, and Computing, pp. 594–603, 1992.

7. Aldous, D. (1995). *Probability distributions on cladograms*, in: Discrete Random Structures, eds. D. J. Aldous and R. Permantle, Springer-Verlag, IMA Vol. in Mathematics and its Applications. Vol. 76, 1-18.

8. Ambainis, A., Desper, R., Farach, M., and S. Kannan (1997), *Nearly tight bounds on the learnability of evolution,* Proceedings of the 1997 ACM Foundations of Computer Science, pp. 524-533.

9. Amir, A. and Keselman, D. *Maximum Agreement Subtree of a set of Evolutionary Trees - Metrics and Efficient Algorithms,* SIAM J. Computing, 1997, Vol. 26, No. 6, pp. 1656-1669, (preliminary version appeared in FOCS 94).

10. Arnborg, A., Corneil, D., and A. Proskurowski (1987). *Complexity of finding embeddings in a k-tree.* SIAM J. Algebraic Discrete Meth., 8, pp. 277-284.

11. Atteson, K. (1997). *The Performance of Neighbor-Joining Methods of Phylogeny Reconstruction*, to appear, Algorithmica, special issue on computational biology. A preliminary version appeared in Third Annual International Computing and Combinatorics Conference (COCOON), Shanghai, China.

12. Bandelt, H-J., and A. Dress (1986), *Reconstructing the shape of a tree from observed dissimilarity data,* Advances in Applied Mathematics, 7 pp. 309-343.

13. Bandelt, H-J., and A. Dress (1992), *A canonical decomposition theory for metrics on a finite set,* Advances in Mathematics, Vol. 92, No. 1, pp. 47-105.

14. Barthélemy, J. and Janowitz, F. (1991). *A formal theory of consensus*, SIAM J. Disc. Math., 3:305-322.

15. Barthélemy, J. and A. Guénoche, Trees and Proximity Representations. 1991, John Wiley and Sons Ltd. West Sussex, England.

16. Barthélemy, J. and McMorris, F. *The median procedure for n-Trees*, Journal of Classification, 3:329-334 (1986).

17. Bellare, M. and M. Sudan, *Improved non-approximability results*, Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing, (Montreal), ACM, pp. 184-193.

18. Berry, V. and O. Gascuel (1997). *Inferring evolutionary trees with strong combinatorial evidence.* Proceedings of COCOON.

19. Berry, V. and O. Gascuel (1996), *On the interpretation of bootstrap trees: appropriate threshold of clade selection and induced gain,* Mol. Biol. Evol. 13(7): 999-1011.

20. Blanken, R., Klotz, L., and A. Hinnebush. (1982) *Computer comparisons of new and existing criteria for constructing evolutionary trees from sequence data,* J. Mol. Evol. 19, 9-19.

21. H.L. Bodlaender, M.R. Fellows, Michael T. Hallett, H. Todd Wareham, and T. Warnow. On the complexity of problems on colored graphs of small width. To appear, Theoretical Computer Science. A preliminary version of some of the results in this paper appear in *Two strikes against perfect phylogeny,* Proceedings, International Congress on Automata and Language Processing, 1992.

22. Bodlaender, H. and Kloks, T. (1993). *A simple linear time algorithm for triangulating three-colored graphs,* Journal of algorithms, 15, pp. 160-172.

23. Bonet, M., Phillips, C.A., Warnow, T., and S. Yooseph (1996). *Constructing evolutionary trees in the presence of polymorphic characters,* ACM Symposium on the Theory of Computing. To appear, SIAM J. Computing.

24. Brossier, G. (1985) *Approximation des dimmilarités par des arbres additifs,* Math. Sci. Hum. 91, 5-21.

25. Bruns, T.D. and T.M. Szaro (1992). *Rate and mode differences between nuclear and mitochondrial small-subunit rRNA genes in mushrooms.* Mol. Biol. Evol. 9 (5): 836-855.

26. Bryant, D. (1995). *Hunting for binary trees in binary character sets: efficient algorithms for extraction, enumeration, and optimization,* Research Report #124, Department of Mathematics and Statistics, Canterbury University, Christchurch, New Zealand.

27. Buneman, P. Mathematics in the archaeological and historical sciences (F.R. Hobson, D.G. Kendal, and P. Tautu, eds.) University Press, Edinburgh, p. 387.

28. Buneman, P. (1974). *A characterization of rigid circuit graphs,* Discrete Mathematics, 9:205-212.

29. Carroll, J.D. (1976). *Spatial, non-spatial, and hybrid models for scaling,* Psychometrika, 41 (4) 439-463.

30. Carroll, J.D., and S. Pruzansky. (1980) Discrete and hybrid scaling models. In *Similarity and Choice,* E.B. Lanterman and H. Freger, eds. Hans Huber, Berne.

31. Cavender, J.A. (1978). *Taxonomy with confidence,* Math. Biosci., **40**, 271–280.

32. Cavender, J.A. and J. Felsenstein (1987). *Invariants of phylogenies: simple case with discrete states,* J. Classification, **4**, 57–71.

33. Cohen, J. and M. Farach (1997). *Numerical Taxonomy on Data: Experimental Results.* ACM-SIAM Symposium on Discrete Algorithms, pp. 410-417.

34. Day, W.H.E. (1985), *Optimal algorithms for comparing trees with labelled leaves*, Journal of Classification 2, pp. 7-28.

35. Day, W.H.E., and Sankoff, D. (1986), *Computational complexity of inferring phylogenies by compatibility,* Systematic Zoology, 35, pp. 224-229.

36. Day, W.H.E. (1987). *Computational complexity of inferring phylogenies from dissimilarity matrices,* Bull. of Math. Biol, Vol 49, No. 4, pp. 461-467.

37. Day, W.H.E. (1983). *Computationally difficult parsimony problems in phylogenetic systematics,* Journal of Theoretical Biology, 103: 429-438.

38. Dress, A. and Steel, M.A. (1992). *Convex Tree Realizations of Partitions,* Applied Math Letters, 5(3): 3-6.

39. Erdös, P.L., Steel, M., Szekeley, L. and T. Warnow (1997). *Inferring big trees from short sequences.* Proceedings of International Congress on Automata, Languages, and Programming 1997.

40. Erdös, P.L., Steel, M., Szekeley, L. and T. Warnow (1997). *A few logs suffice to build (almost) all trees - I.* DIMACS Technical Report 97-71. Also appears in Random Struct. Alg., 14, 153-184, 1999.

41. Erdös, P.L., Steel, M., Szekeley, L. and T. Warnow (1997). *A few logs suffice to build (almost) all trees - 2.* DIMACS Technical Report 97-72. To appear in Theoretical Computer Science, special issue for selected papers from ICALP 1997.

42. Estabrook, G.F. and Landrum, L. (1975). *A simple test for the possible simultaneous evolutionary divergence of two aminoacid positions,* Taxon. 24: 609-613.

43. Estabrook, G.F. and McMorris, F.R. (1977). *When are two qualitative taxonomic characters compatible?,* J. Math. Biol., 4: 195-200.

44. Estabrook, G.R., Johnson, C.S., Jr., and F.R. McMorris (1976), *A mathematical foundation for the analysis of cladistic character compatibility,* Math. Biosci. 29, pp. 181-187.

45. Estabrook, G.F., and McMorris, F.R. (1980), *When is one estimate of evolutionary relationships a refinement of another?,* J. Math. Biosci. 10, pp. 327-373.

46. Estabrook, G.F., Johnson, C.S. Jr. and F.R. McMorris (1975). *An idealized concept of the true cladistic character,* Math. Biosci. 23, pp. 263-272.

47. Estabrook, G.F., Johnson, C.S. Jr. and F.R. McMorris (1976). *An algebraic analysis of cladistic characters,* Discrete Math., 16, pp. 141-147.

48. Farach, M. and S. Kannan (1996). *Efficient algorithms for inverting evolution, Proceedings of the ACM Symposium on the Foundations of Computer Science,* 230–236.

49. Farach, M., Kannan, S. and T. Warnow (1996). *A Robust Model for Finding Optimal Evolutionary Trees,* Algorithmica, special issue on Computational Biology, Vol. 13, No. 1, pp. 155-179. (A preliminary version of this paper appeared at STOC 1993.)

50. Farach, M. Przytycka, T. and M. Thorup (1995). *On the agreement of many trees,* Information Processing Letters, Vol. 55, No. 6, pp. 297-301.

51. Farach, M. and Thorup, M. (1994), *Optimal evolutionary tree comparisons by sparse dynamic programming,* Proceedings of the 35th annual IEEE Foundations of Computer Science, 1994, pp. 770–779. Also, SIAM Journal on Computing, Vol. 26, No. 1, pp. 210-230.

52. Farris, J.S. (1973). *A probability model for inferring evolutionary trees,* Syst. Zool., **22**, 250–256.

53. Felsenstein, J. (1982). *Numerical methods for inferring evolutionary trees,* The Quarterly Review of Biology, Vol. 57, No. 4.

54. Felsenstein, J. (1988) *Phylogenies from molecular sequences: inferences and reliability,* Ann. Rev. Genetics, 22:521-565.

55. Felsenstein, J. (1978). *Cases in which parsimony or compatibility methods will be positively misleading,* Syst. Zoology, 27:401-410.

56. Fernández-Baca, D., and J. Lagergren (1996). *A polynomial time algorithm for near-perfect phylogeny,* Proceedings ICALP, pp. 670-680.

57. Fitch, Wm. (1971). *Toward defining the course of evolution: minimum change for a specified tree topology,* Syst. Zool., 20:406-416.

58. Fitch, W.M. (1975). *Toward finding the tree of maximum parsimony,* Proc. Eighth International Conference on Numerical Taxonomy, G.F. Estabrook, ed., pp. 189-230, W.H. Freeman, San Francisco.

59. Fitch, W.M., and E. Margolaish (1967), *Construction of phylogenetic trees,* Science, 155, 279-284.

60. Foulds, L.R., and R.L. Graham. (1982). *The Steiner Problem in Phylogeny is NP-Complete,* Adv. Appl. Math. 3, 43-49.

61. Garey, M.R. and Johnson, D.S (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Company, NY.

62. Gascuel, O. (1997). *BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data,* Cahier du GERAD G-97-18, to appear, Molecular Biology and Evolution.

63. Gillespie, J.H. (1984). *The molecular clock may be an episodic clock.* Proc. Natl. Acad. Sci. 81 (24): 8009-8013.

64. Gower, J., and G. Ross (1969). *Minimum spanning trees and single linkage cluster analysis, Appl. Stat.* 18, 54-64.

65. Gusfield, D. (1984). *The steiner tree problem in phylogeny,* Technical Report 332, Yale University, Department of Computer Science.

66. Gusfield, D. (1991), *Efficient algorithms for inferring evolutionary trees*, Networks 21, pp. 19-28.

67. Hartigan, J.A. (1973). *Minimum mutation fits to a given tree,* Biometrics 29, pp. 53-65.

68. Hein, J. (1990) *Unified approach to alignment and phylogenies,* in Methods in Enzymology, Vol 183.

69. Hillis, D. (1997). *Inferring complex phylogenies*, Nature, Vol. 383, pp. 130-131.

70. Huelsenbeck, J. and D. Hillis (1993). *Success of phylogenetic methods in the four-taxon case.* Syst. Bio. 42(3):247-264.

71. Idury, R. and Schaffer, A. (1993). *Triangulating three-colored graphs in linear time and linear space,* SIAM J. of Discrete Mathematics.

72. Jardine, C., Jardine, N., and R. Sibson. (1967) *The structure and construction of taxonomic hierarchies,* Math. Biosci. 1, 173-179.

73. Kannan, S. and Warnow, T. (1992). *Triangulating three-colored graphs,* SIAM J. Discrete Mathematics, May 1992, pp. 249-258.

74. Kannan, S. Lawler, E. and T. Warnow (1996). *Determining the evolutionary tree*, J. of Algorithms, 21, pp. 26-50.

75. Kannan, S. and T. Warnow. (1994). *Inferring Evolutionary History from DNA Sequences,* SIAM J. on Computing, Vol. 23, No. 4, pp. 713-737. (A preliminary version of this paper appeared at FOCS 1990.)

76. Kannan, S., Warnow, T. and S. Yooseph (1995). *Computing the local consensus of trees,* The Association for Computing Machinery and the Society of Industrial Applied Mathematics, Proceedings, ACM/SIAM Symposium on Discrete Algorithms, 1995, pp. 68-77. To appear, SIAM J. Computing.

77. Kannan, S. and Warnow, T. (1997). *A fast algorithm for the computation and enumeration of perfect phylogenies when the number of character states is fixed,* SIAM J. Computing, Vol. 26, No. 6, pp. 1749-1763. A preliminary version appeared in the Proceedings, ACM/SIAM Symposium on Discrete Algorithms, 1995.

78. Kannan, S. and T. Warnow (1995). *Tree reconstruction from partial orders.* SIAM J. Computing. Vol. 24, No. 3, pp. 511-519.

79. Kao, M. (1997). *Tree contractions and evolutionary trees,* Proceedings of the Third Italian Conference on Algorithms and Complexity.

80. Kearney, P. (1997). *A six-point condition for ordinal matrices,* Journal of Computational Biology.

81. Kearney, P., Hayward, R.B., and H. Meijer, (1997) *Inferring evolutionary trees from ordinal data,* Proceedings of the 8th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 418-426.

82. Kim, J. (1996). *General inconsistency conditions for maximum parsimony: effects of branch lengths and increasing numbers of taxa.* Syst. Biol. 45(3): 363-374.

83. Kuhner, M. and J. Felsenstein, (1994). *A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates.* Mol. Biol. Evol. 11:459-468.

84. LeQuesne, W.J. (1969). *A method of selection of characters in numerical taxonomy,* Syst. Zool., 18: 201-205.

85. LeQuesne, W.J. (1972). *Further studies on the uniquely derived character concept,* Syst. Zool., 21: 281-288.

86. LeQuesne, W.J. (1974). *The uniquely evolved character concept and its cladistic application,* Syst. Zool., 23: 513-517.

87. LeQuesne, W.J. (1977). *The uniquely evolved character concept,* Syst. Zool., 26, pp. 218-223.

88. Li, W.H. and M. Tanimura (1987), *The molecular clock runs more slowly in man than in apes and monkeys.* Nature 326 (6108): 93-96.

89. Li, W.H., Tanimura M., and P.M. Sharp (1987), *An evaluation of the molecular clock hypothesis using mammalian DNA sequences.* J Mol Evol 25 (4): 330-342.

90. McMorris, F.R. (1977). *On the compatibility of binary qualitative taxonomic characters,* Bull. Math. Biol. 39: 133-138.

91. McMorris, F.R. and C.A. Meacham, *Partition intersection graphs*, Ars Combinatorica, 16-B, pp. 135-138.

92. McMorris, F.R. and Steel, M. (1994), *The complexity of the median procedure for binary trees.* Proceedings of the 4th Conference of the International Federation of Classification Societies, Paris 1993, to be published in the series "Studies in Classification, Data Analysis, and Knowledge Organization" by Springer Verlag.

93. McMorris, F.R., Warnow, T. and Wimer, T. (1993). *Triangulating vertex colored graphs,* SIAM J. of Discrete Mathematics, Vol. 7, No. 2, pp. 296-306.

94. Nelson, G. (1979). *Cladistic analysis and synthesis: Principles and definitions, with a historical note on Adanson's* Familles des Plantes *(1763-1764).* Syst. Zool., 28:1-21.

95. Nelson, G. and N.I. Platnick (1981). *Systematics and biogeography : Cladistics and vicariance,* Columbia Univ. Press, New York.

96. Page, R.D.M. (1990), *Tracks and Trees in the Antipodes: a reply to Humphries and Seberg.* Syst. Zool. 39(3):288-299, 1990.

97. Page, R. D. M. (1993). *Genes, Organisms, and Areas: The Problem of Multiple Lineages*, Systematic Biology, 42(1), 77-84.

98. Page, R. D. M. (1993), *Reconciled Trees and Cladistic Analysis of Historical Associations Between Genes, Organisms, and Areas*, manuscript.

99. Phillips, C.A. and T. Warnow (1996). *The Asymmetric Median Tree: a new model for building consensus trees.* Discrete Applied Mathematics, Special Issue on Computational Molecular Biology, 71, pp. 311-335.

100. Rice, K.A. and T. Warnow (1997). *Parsimony is hard to beat!*, COCOON Proceedings.

101. Ringe, D., Warnow, T., Taylor, A., Michailov, A., and L. Levison. (1997). Computational cladistics and the position of Tocharian. In V. Mair (Ed.), *The Bronze Age and Early Iron Age Peoples of Eastern Central Asia*, a special volume of the Journal of Indoeuropean Studies.

102. Saitou, N. and T. Imanishi. (1989). *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.* Mol. Biol. Evol. 6:514-525.

103. Saitou, N., and M. Nei. (1987). *The neighbor-joining method: A new method for reconstructing phylogenetic trees.* Mol. Biol. Evol. 4:406-425.

104. Sankoff, D. (1975). *Minimum mutation trees of sequences,* SIAM J. on Applied Mathematics. 28(1):35-42.

105. Schoniger, M. and A. von Haeseler (1995). *Performance of maximum likelihood, neighbor joining, and maximum parsimony methods when sequence sites are not independent.* Syst. Biol. **(44)**4: 533-547.

106. Sharp, P. and W.H. Li (1989). *On the rate of DNA sequence evolution in Drosophila*, J. Mol. Evol. 28 (5): 398-402.

107. Sokal, R., and P. Sneath, (1963), *Principles of Numerical Taxonomy*, Freeman, San Francisco.

108. Sourdis, J. and M. Nei, (1996). *Relative efficiencies of the maximum parsimony and distance-matrix methods in obtaining the correct phylogenetic tree.* Mol. Biol. Evol. 5(3): 293-311.

109. Steel, M.A., L.A. Székely, and M.D. Hendy (1994). *Reconstructing trees when sequence sites evolve at variable rates,* Journal of Computational Biology, Volume 1, No. 2, pp. 153-163.

110. Steel, M.A. (1994). *Recovering a tree from the leaf colourations it generates under a Markov model,* Appl. Math. Lett., **7**, 19–24.

111. Steel, M.A. (1992). *The complexity of reconstructing trees from qualitative characters and subtrees,* Journal of Classification, Vol. 9:91-116.

112. Steel, M.A., and Warnow, T.J. (1993), *Kaikoura tree theorems: Computing the maximum agreement subtree,* Information Processing Letters 48, pp. 72-82.

113. Strimmer, K. and A. von Haeseler. (1996). *Accuracy of neighbor joining for n-taxon trees,* Syst. Biol. 45 (4): 516-523.

114. Strimmer, K. and A. von Haeseler (1996). *Quartet Puzzling: a quartet maximum likelihood method for reconstructing tree topologies,* Mol. Biol. Evol., 964–969.

115. Swofford, D.L., Olsen, G.J., Waddell, P., and D. M. Hillis (1996). Chapter 11: Phylogenetic inference, in: *Molecular Systematics*, D. M. Hillis, C. Moritz, B. K. Mable, eds., 2nd edition, Sinauer Associates, Inc., Sunderland, pp. 407–514.

116. Templeton, A. (1992). *Human origins and analysis of mitochondrial DNA sequences.* Science, Vol. 255, 737-739.

117. Vawter, L, and Wm. Brown (1986). *Nuclear and mitochondrial DNA comparisons reveal extreme rate variation in the molecular clock,* Science 234 (4773): 194-196.

118. Wang, L. and T. Jiang, (1994) *On the complexity of multiple sequence alignment,* J. Computational Biology, 1:337-348.

119. Wang, L., Jiang, T., and E. Lawler (1996). *Approximation algorithms for tree alignment with a given phylogeny,* Algorithmica 16, pp. 302-315.

120. Wang, L. and D. Gusfield (1998). *Improved approximation algorithms for tree alignment,* Journal of Algorithms, to appear.

121. Wareham, H. T. (1985) "An Efficient Algorithm for Computing Ml Consensus Trees", Honors Dissertation, Department of Computer Science, Memorial University of Newfoundland, St. John's, Newfoundland.

122. Warnow, T.J. (1994), *Tree compatibility and inferring evolutionary history,* Journal of Algorithms, 16, pp. 388-407.

123. Warnow, T. (1997). *Mathematical approaches to comparative linguistics.* Proceedings of the National Academy of Sciences, Vol. 94, pp. 6585-6590.

124. Warnow, T., Ringe, D. and A. Taylor (1996). *Reconstructing the evolutionary history of natural languages,* Association for Computing Machinery and the Society of Industrial and Applied Mathematics, Proceedings of ACM-SIAM Symposium on Discrete Algorithms (SODA), 1996, pp. 314-322.

125. Waterman, M.S., Smith, T.F., Singh, M. and W.A. Beyer (1977). *Additive evolutionary trees,* Journal Theoretical Biol. 64:199-213.

126. Wilson, A. C. and R. L. Cann, (1992). *The recent African genesis of humans,* Scientific American, April, 1992, pp. 68–73.

127. Chang, J. Inconsistency of evolutionary tree topology reconstruction methods when substitution rates vary across characters. Mathematical Biosciences 134:189-215 (1996).

128. Chang, J. Full reconstruction of markov models on evolutionary trees: identifiability and consistency, Math. Biosci., vol. 137, pp. 51-73, 1996.

129. Shoemaker, J.S¿ and W. M. Fitch, "Evidence from nuclear sequences that invariable sites should be considered when sequence divergence is calculated," Molecular Biology and Evolution, vol. 6, pp. 270-289, 1989.

130. Tuffley, C. and M. Steel, "Modeling the covarion hypothesis of nucleotide substitution," Math. Biosci., vol. 147, pp. 63-19, 1998.

131. Tuffley, C. and M. Steel, "Links between maximum likelihood and maximum parsimony under a simple model of site substitution," Bull. Math. Biol., vol. 59, pp. 581-607, 1997.

132. Krogh, A. M. Brown, I. S. Mian, K. Sjolander, and D. Haussler, "Hidden Markov Models in computational biology: Applications to protein modeling," J. Mol. Biol., vol. 235, pp. 1501-1531, 1994.

133. Felsenstein, J. and G. A. Churchill, "A hidden Markov model approach to variation among sites in rate of evolution," Mol. Biol. Evol., vol. 13, pp. 93-104, 1996.

134. Steel, M.A., Szekely, L.A., and M.D. Hendy. Reconstructing trees whwn sequence sites evolve at variable rates. J. Computational Biology. 1:153-163 (1994).

135. X. Gu and W. H. Li, "Estimation of evolutionary distances under stationary and nonstationary models of nucleoide substitution," Proc. Natl. Acad. Sci. USA, vol. 95, pp. 5899-5905, 1998.

136. Gojobori, T., K. Ishii, and M. Nei, "Estimation of average number of nucleotide substitutions when the rate of substitution varies with nucleotide," J. Mol. Evol., vol. 18, pp. 414-423, 1982.

137. Z. Yang, "Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: approximate methods," J. Mol. Evol., vol. 39, pp. 306-314, 1994.

138. L. Jin and M. Nei, "Limitations of the evolutionary parsimony method of phylogenetic analysis," Mol. Biol. Evol., vol. 7, pp. 82-102, 1990.

139. G. B. Golding, "Estimates of DNA and protein sequence divergence: an examination of some assumptions," Mol. Biol. Evol., vol. 1, pp. 125-142,

140. Z. Yang, N. Goldman, and A. Friday, "Comparison of models for nucleotide substitution used in maximum-likelihood phylogenetic estimation," Mol. Biol. Evol., vol. 11, pp. 316-324, 1994.

141. Z. Yang, "How often do wrong models produce better phylogenies," Mol. Biol. Evol., vol. 14, pp. 105-108, 1997.

142. N. Goldman, "Simple diagnostic statistical tests of models for DNA substitution," J. Mol. Evol., vol. 37, pp. 650-661, 1993.

143. J. Felsenstein, "Evolutionary trees from DNA sequences: A maximum likelihood approach," Journal of Molecular Evolution, vol. 17, pp. 368-376, 1981.

144. M. Hasegawa, H. Kishino, and T. Yano, "Dating of the human-ape splitting by a molecular clock of mitochondrial DNA," J. Mol. Evol., vol. 22, pp. 160-174, 1985.

145. M. Kimura, "Estimation of evolutionary distances between homologous nuceotide sequences," Proc. Natl. Acad. Sci. U.S.A., vol. 78, pp. 454-458, 1981.

146. T. H. Jukes and C. R. Cantor, "Evolution of protein molecules," in Mammalian Protein Metabolism, vol. 3, H. N. Munro, Ed. New York: Academic Press, 1969, pp. 21-132.

147. Kumar, S. K. Tamura, and M. Nei. 1993. MEGA: Molecular Evolutionary Genetics Analysis. Pennsylvania State University, University Park, PA.

148. Huson, D., Nettles, S. , and T. Warnow. Obtaining accurate topology estimates of evolutionary trees from very short sequences. Proceedings RECOMB 1999, pp. 198-207.

149. Csuros, M., and Kao, Ming-Yang. Recovering evolutionary trees through harmonic greedy triplets. In *Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 261-270, 1999.