BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data

Olivier Gascuel

BIONJ: An Improved Version of the NJ Algorithm Based on a Simple Model of Sequence Data

Olivier Gascuel

GERAD, Ecole des HEC, Montreal, and Département d'Informatique Fondamentale, LIRMM, Montpellier

We propose an improved version of the neighbor-joining (NJ) algorithm of Saitou and Nei. This new algorithm, BIONJ, follows the same agglomerative scheme as NJ, which consists of iteratively picking a pair of taxa, creating a new node which represents the cluster of these taxa, and reducing the distance matrix by replacing both taxa by this node. Moreover, BIONJ uses a simple first-order model of the variances and covariances of evolutionary distance estimates. This model is well adapted when these estimates are obtained from aligned sequences. At each step it permits the selection, from the class of admissible reductions, of the reduction which minimizes the variance of the new distance matrix. In this way, we obtain better estimates to choose the pair of taxa to be agglomerated during the next steps. Moreover, in comparison with NJ's estimates, these estimates become better and better as the algorithm proceeds. BIONJ retains the good properties of NJ—especially its low run time. Computer simulations have been performed with 12-taxon model trees to determine BIONJ's efficiency. When the substitution rates are low (maximum pairwise divergence ≈0.1 substitutions per site) or when they are constant among lineages, BIONJ is only slightly better than NJ. When the substitution rates are higher and vary among lineages, BIONJ clearly has better topological accuracy. In the latter case, for the model trees and the conditions of evolution tested, the topological error reduction is on the average around 20%. With highly-varying-rate trees and with high substitution rates (maximum pairwise divergence ≈1.0 substitutions per site), the error reduction may even rise above 50%, while the probability of finding the correct tree may be augmented by as much as 15%.

Introduction

The neighbor-joining (NJ) algorithm of Saitou and Nei (1987) is one of the most popular methods for reconstructing phylogenetic trees from a matrix of pairwise evolutionary distances. This algorithm follows an agglomerative scheme which was first proposed in the context of mathematical psychology by Sattath and Tversky (1977). Agglomerative algorithms iteratively pick a pair of taxa, create a new node which represents the cluster of these taxa, and reduce the distance matrix by replacing both taxa by this node. This cycle is repeated until only three taxa remain. To agglomerate pairs of nodes, NJ follows the minimum-evolution (ME) principle, which was first suggested by Kidd and Sgaramella-Zonta (1971) and which consists of choosing the tree with the smallest sum of branch lengths. Rzhetsky and Nei (1993) have shown that this principle has a sound theoretical foundation when the lengths are obtained by the ordinary least-squares method and when an unbiased estimate of evolutionary distances is used. Under these assumptions, the true tree has the smallest expected length among all possible trees. However, this result describes an expected (or average) behavior, and it is not applicable to every particular data set. Moreover, due to its greedy, agglomerative approach, NJ does not usually find the ME tree, but only a short tree whose topology is generally similar to that of the ME tree (Saitou and Imanishi 1989). This does not preclude good performance, since for any particular data set, the true tree itself is usually close to but not identical with the ME tree, and numerous computer simulations (Saitou and Nei 1987; Nei 1991; Charleston, Hendy, and Penny 1994; Kuhner and Felsenstein 1994) have shown the high relative efficiency of the NJ method in recovering the true topology. Following these authors, NJ seems to be one of the very best distance methods. It is more reliable than the maximum-parsimony approaches, which are sometimes asymptotically inconsistent, and it is just slightly weaker than the maximum-likelihood methods, especially when the molecular clock hypothesis is clearly violated, probably because it does not take adequate account of the model of sequence evolution. Moreover, the NJ algorithm, as formulated by Studier and Keppler (1988), is efficient from a computational point of view and has an $O(n^3)$ time complexity, where n is the number of taxa. Also, theoretical studies (Atteson 1996) have shown that NJ is in some sense as efficient as possible.

Several attempts have been made to improve the NJ algorithm by designing methods able to find trees very close to or identical with the ME tree. Saitou and Imanishi (1989) proposed an exhaustive search method which applies when the number of taxa is small ($n \leq 10$). Rzhetsky and Nei (1993) designed various strategies to search for the ME tree in the neighborhood of the NJ tree by conducting local rearrangements. These authors also suggested that alternative topologies could be generated using a bootstrap procedure (Rzhetsky and Nei 1994). Finally, Kumar (1996) designed efficient heuristics for searching the tree space in a more or less exhaustive manner. These methods have the ability to produce a set of short trees, which provide more information than the single NJ tree. Moreover, they usually find trees shorter than the NJ tree. But, unfortunately, computer simulations (Saitou and Imanishi 1989; Kumar

Key words: phylogeny, neighbor-joining, distance method, model of data, variances and covariances of distance estimates.

Address for correspondence and reprints before July 31, 1997: Olivier Gascuel, GERAD, Ecole des HEC, 3000 chemin de la Côte-Sainte-Catherine, Montreal, Quebec, Canada H3T 2A7. E-mail: olivierg@crt.umontreal.ca. Address as of July 31, 1997: Département d'Informatique Fondamentale, LIRMM, 161 rue Ada, 34392, Montpellier, France. E-mail: gascuel@lirmm.fr.

Mol. Biol. Evol. 14(7):685–695. 1997

1996) indicate that the ability to recover the true topology is not increased and that NJ may hardly be outstripped in this way.

This paper proposes a different approach. Instead of trying to find trees shorter than NJ trees, we reconsider the basic principle of the NJ algorithm. We show that some mathematical formulae employed by NJ may be improved by taking into account the features of biological data. This new version, which we call BIONJ, is basically intended to deal with evolutionary distances obtained from aligned sequences. In what follows, we first describe this new algorithm, then provide computer simulations to demonstrate its efficiency.

The BIONJ Algorithm

First, we provide some notation and recall the main features of NJ. Then, we show that the formula used by NJ to reduce the distance matrix belongs to a larger class of admissible formulae, and we propose selecting from this class the minimum variance reduction. In order to estimate this variance, we use a simple first-order model of the sampling variances and covariances of evolutionary distance estimates. This model leads to a simple expression of the minimum variance reduction, which is fully consistent with the agglomerative approach. These elements are combined to form BIONJ. Finally, we provide an interpretation of this algorithm, and we compare it with NJ from a theoretical perspective.

Notation and Background

In what follows, we use the simplified expression of NJ from Studier and Keppler (1988), equivalent to the original (Gascuel 1994). NJ uses an agglomerative approach. At each step, it has a distance matrix (δ_{ij}) where i and j are taxa, or clusters of original taxa agglomerated during the previous steps. The dimension of this matrix will be denoted as r, and we have $r = n - p$, where n is the number of original taxa and p is the number of steps already taken. NJ determines the next pair to be agglomerated by minimizing a criterion related, among other things, to the ordinary least-squares length of the tree under construction. In fact, this criterion admits numerous interpretations, and we have shown (Gascuel 1997) that its use is well founded even when we abandon the ordinary least-squares framework, i.e., when we abandon the assumption of independence and equivariance of the δ_{ij} estimates and when we place ourselves in the generalized least-squares framework. Let Q_{xy} be the value of this criterion for the taxa x and y, and, for the sake of simplicity, let $x = 1$ and $y = 2$. We then have

$$Q_{12} = (r - 2)\delta_{12} - S_1 - S_2, \quad \text{where } S_x = \sum_{i=1}^{r} \delta_{xi}. \quad (1)$$

Once the pair to be agglomerated has been selected, NJ creates a new node which represents the new cluster's root, and which is denoted as u in the following. For the sake of simplicity, assume that the pair {1, 2} has been selected. Then, NJ estimates the length of the branch (1, u) using the (approximate) formula

$$\delta_{1u} = \frac{1}{2}\left(\delta_{12} + \frac{S_1 - S_2}{(r - 2)}\right), \quad (2)$$

and δ_{2u} is obtained in a symmetrical way. Optimal formulae in the least-squares sense have been proposed by Vach (1989) and by Rzhetsky and Nei (1993). They may replace formula (2) without modifying the topology of the tree under construction (this point is discussed in more detail below). Moreover, once the tree structure has been constructed, it is possible to obtain a tree with only positive (or null) branch lengths using some estimation procedure based on nonnegative least-squares regression (Lawson and Hanson 1974, pp. 158–165; Barthélemy and Guénoche 1991, pp. 62–66; Swofford et al. 1996, pp. 445–450).

Finally, NJ reduces the distance matrix by deleting taxa 1 and 2 and by estimating the distance between the new taxon u and any taxon i using

$$\delta_{ui} = \frac{1}{2}\delta_{1i} + \frac{1}{2}\delta_{2i} - \frac{1}{2}\delta_{1u} - \frac{1}{2}\delta_{2u}, \quad (3)$$

where δ_{1u} and δ_{2u} are given by equation (2).

NJ satisfies a basic requirement for tree reconstruction methods: When data are additive (Barthélemy and Guénoche 1991), it necessarily finds the unique tree which perfectly represents these data. The proof (Saitou and Nei 1987; Studier and Keppler 1988; Charleston, Hendy, and Penny 1993; Gascuel 1997) is inductive and is based on the three following points:

1. If the distance matrix (δ_{ij}) is additive, then criterion (1) necessarily points out a pair of taxa which are neighbors in the tree representing (δ_{ij}).
2. When data are additive, equation (2) is exact.
3. When applied to a real pair of neighbors, reduction (3) transforms an additive matrix, represented by the valued tree **T**, into an additive matrix represented by a subtree of **T** in which both taxa and the corresponding branches have simply been deleted, the cluster's root becoming a taxon of the new tree.

The Minimum Variance Reduction

In fact, reduction (3) belongs to a large class of reduction formulae which all satisfy the third property listed above. Any of these formulae guarantees to find the correct tree with additive data, when combined with equations (1) and (2). These formulae are defined by

$$\delta_{ui} = \lambda\delta_{1i} + (1 - \lambda)\delta_{2i} - \lambda\delta_{1u} - (1 - \lambda)\delta_{2u}, \quad (4)$$

where δ_{1u} and δ_{2u} are obtained from equation (2) or by using any formula which is exact with additive data. The proof is easy. Let (d_{ij}) be an additive matrix, **T** the tree which represents this matrix, {1, 2} a pair of neighbors in **T**, and u the root of {1, 2}. When reduction (4) is applied to (d_{ij}) with the pair {1, 2}, the d_{ij} ($i, j \neq 1, 2$) distances remain unchanged, while the new distances δ_{ui} satisfy $\delta_{ui} = \lambda d_{1i} + (1 - \lambda)d_{2i} - \lambda\delta_{1u} - (1 - \lambda)\delta_{2u}$. Now, let d_{ui} be the distance in **T** between u and any leaf i. Because equation (2) is exact, we have $\delta_{1u} = d_{1u}$ and $\delta_{2u} = d_{2u}$. Moreover, because u is the root of {1, 2}, we also have $d_{1i} = d_{1u} + d_{ui}$ and $d_{2i} = d_{2u} + d_{ui}$. When

combining these equalities, we obtain $\delta_{ui} = d_{ui}$. In other words, the new matrix obtained from reduction (4) is represented by the subtree of **T**, in which 1 and 2 have been deleted, and the proof is finished.

Some remarks about definition (4):

1. The NJ's reduction (3) corresponds to $\lambda = 1/2$.
2. λ need not be constant and may vary at each step of the algorithm (but not depending on taxon i).
3. It is shown in Gascuel (1994) that when criterion (1) is used, adding a constant into the reduction formula does not change the topology of the tree under construction. The last two terms of equation (4) are constant and only depend on the branch lengths δ_{1u} and δ_{2u}. It follows that only the first two $(\lambda\delta_{1i} + (1 - \lambda)\delta_{2i})$ play a part in determining the topology. In other words, only the sampling noise affecting this sum influences the structure of the tree under construction. Moreover, this explains why changing the branch length estimate (2) does not modify the tree topology.

When combining these last two remarks, we see that it is possible at each step to adjust the value of λ in order to minimize the sampling variance of the new, reduced, matrix. More precisely, we have to minimize the variance of the topological part of this matrix defined by the sums $(\lambda\delta_{1i} + (1 - \lambda)\delta_{2i})$. In this way, more reliable estimates will be available to select the pairs of taxa to be agglomerated during the next steps. Moreover, because the process is repeated at each step, these estimates will become better and better in comparison with NJ's estimates as the algorithm proceeds.

Let δ_{ij} be an estimate of the true evolutionary distance d_{ij}; let v_{ij} be the sampling variance of this estimate, and $cov_{ij,kl}$ the covariance of δ_{ij} and δ_{kl}. Consider the center c of the cluster $\{1, 2\}$ defined by the equalities: $d_{ci} = \lambda d_{1i} + (1 - \lambda)d_{2i}$. This center depends on λ, and the goal is to determine λ_* so that the variances v_{ci} of the estimates δ_{ci} are as low as possible. Therefore, we have to minimize

$$\sum_{i=3}^{r} v_{ci} = \sum_{i=3}^{r} \mathrm{Var}[\lambda\delta_{1i} + (1 - \lambda)\delta_{2i}]$$

$$= \sum_{i=3}^{r} (\lambda^2 v_{1i} + (1 - \lambda)^2 v_{2i} + 2\lambda(1 - \lambda)cov_{1i,2i}).$$

This is a second degree polynome, and we find that

$$\lambda_* = \frac{\sum_{i=3}^{r} (v_{2i} - cov_{1i,2i})}{\sum_{i=3}^{r} (v_{1i} + v_{2i} - 2cov_{1i,2i})} \qquad (5)$$

and that

$$v_{ci}^* = \lambda_*^2 v_{1i} + (1 - \lambda_*)^2 v_{2i} + 2\lambda_*(1 - \lambda_*)cov_{1i,2i}. \qquad (6)$$

This result is very general. To use it, we need to estimate the variances and covariances of the δ_{ij} estimates. In the next section, we shall see that these quantities may be approximated in a satisfactory way when the evolutionary distances are obtained from aligned sequences.

A Simple First-Order Model of the (Co)variances of Evolutionary Distance Estimates

In phylogenetic reconstruction, a basic hypothesis is that the evolutionary distances (d_{ij}) between taxa are additive and that the valued tree **T**, which represents these distances, corresponds to the real evolution of taxa. When the evolutionary distances are obtained from aligned sequences, they are usually considered proportional to the substitution rates between sequences. Numerous models have been proposed to estimate these substitution rates from the observed differences between sequences (Zharkikh 1994). In the case of Jukes and Cantor's (1969) model, and related ones, good approximations of the sampling (co)variances of these estimates are available (Nei and Jin 1989; Bulmer 1991). Within a first order, these approximations are

$$v_{ij} \approx \frac{1}{s} d_{ij} \quad \text{and} \quad cov_{ij,kl} \approx \frac{1}{s} d_{uv}, \qquad (7)$$

where s is the sequence length, u and v represent the extremities of the intersection of the paths (i, j) and (k, l) in **T**, and d_{uv} is the evolutionary distance between these two ancestral species. When the intersection of (i, j) and (k, l) is empty, the covariance of δ_{ij} and δ_{kl} is null.

In fact, approximation (7) is valid within a first order and near 0 for almost any evolutionary distance estimate. Indeed, it is well known (Nei 1991) that when the frequency of observed substitutions between sequences is low, all estimates are practically identical and equal to this frequency. This holds whatever the model, e.g., for the two-parameter model of Kimura (1980) or for Jin and Nei's (1990) gamma estimate or even for estimates based on amino-acid sequences (Kimura 1983); the proof is obtained by considering the variances and the covariances of the frequencies of observed substitutions.

Moreover, approximation (7) remains satisfactory when departing from 0. It is easily checked with simple models such as Jukes and Cantor's (1969) or Kimura's (1980) that the behavior of v_{ij} and d_{ij} remains qualitatively the same. Let p_{ij} be the probability for a substitution to be observed in any randomly chosen site. Near 0, the variance v_{ij} and the distance d_{ij} linearly increase as a function of p_{ij}. Afterward, both grow rapidly and tend to infinity when p_{ij} becomes close to some limit value, which is equal to ¾, for example, in Jukes and Cantor's (1969) model.

Finally, our approach, as that of generalized least-squares (Bulmer 1991), does not require exact values for variances and covariances. Approximate values are sufficient. Consequently, approximation (7) is quite satisfactory for our purpose and can be applied to almost any evolutionary distance estimate obtained from aligned sequences. Moreover, it is probably applicable to other estimates. However, it does not seem to be convenient for DNA-DNA hybridization data, in which the covariances are completely different from that predicted by equation (7) (Felsenstein 1987). In this case, another model is needed.

Results Obtained with this Model

As we consider model (7), which we shall now do, we find ourselves in a very comfortable situation, for:

1. The variances (v_{ij}) may be estimated by (δ_{ij}/s).
2. The variances (v_{ij}) are tree-like, as are the evolutionary distances (d_{ij}). It follows that the covariances, which are path lengths in the variance tree, may be computed from the variances; thus, we have

$$\mathrm{cov}_{1i,2i} = \frac{1}{2}(v_{1i} + v_{2i} - v_{12}). \quad (8)$$

Therefore, we have estimates for the variances and the covariances of the distances between original taxa. These allow the algorithm to be initialized and the first agglomeration step to be taken. Let us now see what happens during the next steps, when the (δ_{ij}) matrix refers not only to original taxa, but also to clusters of these taxa. When replacing the covariances by their values (8) in expressions (5) and (6), we find that

$$\lambda_* = \frac{1}{2} + \frac{\sum_{i=3}^{r}(v_{2i} - v_{1i})}{2(r-2)v_{12}} \quad (9)$$

and that

$$v^*_{ci} = \lambda_* v_{1i} + (1 - \lambda_*)v_{2i} - \lambda_*(1 - \lambda_*)v_{12}. \quad (10)$$

First, as a consequence of the triangle inequalities $-v_{12} \leq v_{2i} - v_{1i} \leq v_{12}$, we have that λ_* necessarily belongs to [0, 1]. Moreover, formula (10) provides the new variances, introduced by the first agglomeration. Let us now consider the new covariances $\mathrm{cov}_{ci,ji}$, where $j \neq 1, 2, i$. We know from equation (8) that

$$\mathrm{cov}_{1i,ji} = \frac{1}{2}(v_{1i} + v_{ji} - v_{1j})$$

and

$$\mathrm{cov}_{2i,ji} = \frac{1}{2}(v_{2i} + v_{ji} - v_{2j}).$$

Using the center definition, equality (10), and both of the above equations, we find that

$$\begin{aligned}
\mathrm{cov}_{ci,ji} &= \mathrm{cov}(\lambda_*\delta_{1i} + (1 - \lambda_*)\delta_{2i}, \delta_{ji}) \\
&= \lambda_* \mathrm{cov}_{1i,ji} + (1 - \lambda_*)\mathrm{cov}_{2i,ji} \\
&= \frac{1}{2}(\lambda_* v_{1i} + (1 - \lambda_*)v_{2i} + v_{ji} - \lambda_* v_{1j} \\
&\qquad - (1 - \lambda_*)v_{2j}) \\
&= \frac{1}{2}(v^*_{ci} + v_{ji} - v^*_{cj}).
\end{aligned}$$

In other words, the covariances introduced by the first agglomeration can be computed from the variances using equation (8), as can the initial covariances. It follows that formulae (9) and (10) can be used in the second step. By induction, we obtain that at every step: formula (8) is valid; the explicit values of the covariances are useless; and it is possible to directly use formulae (9) and (10). This property is valid whatever the value of λ_*. Using estimated values for the variances instead of their true values makes the computation of λ_* suboptimal, but does not invalidate this property. However, according to our previous remark, this estimated value has to be maintained in [0, 1]. Let us also underline that we have not assumed that the taxa which are agglomerated are effectively neighbors in the true tree. In short, computations (9) and (10) remain consistent throughout the agglomerative procedure, even when λ_* is imperfectly estimated and when erroneous pairs of taxa have been selected. The simplicity of these computations comes from our first-order model (7), in which variances are tree-like. With a higher order model (e.g., Bulmer 1991) things would likely be more complicated.

```
    Input the distance matrix (δij) of size n×n ;
    Initialize the number of taxa: r←n ;
(a) Initialize the variance matrix: (vij)←(δij) ;
    While the number of taxa r is greater than 3:
(b)     {   Compute the sums Sx ;
(c)         Find the pair to be agglomerated by minimizing (1) ;
            Compute the branch-lengths using (2) ;
(d)         Determine λ* using (9) and the constraint λ*∈[0,1] ;
            Apply Reduction (4) to (δij) ;
(e)         Apply Reduction (10) to (vij) ;
            Decrease the number of taxa: r←r−1 }
    Compute the last three branch-lengths using (2) ;
    Output the tree found.
```

FIG. 1.—The BIONJ algorithm.

Equation (10) is a reduction formula identical to reduction (4) except for the constant term. Therefore, to compute the estimated values of variances, a very simple solution consists of iteratively reducing by equation (10) a matrix whose initial values are (δ_{ij}/s). In fact, the variance matrix (v_{ij}) may be initialized as (δ_{ij}), because the term $1/s$ does not play any part in equation (9).

The Algorithm

The algorithm is summarized in figure 1. Basically, it differs from NJ in lines (a), (d) and (e). These lines introduce an additional computational cost in terms of time and space. However, this cost is very low. Indeed, (a) has an $O(n^2)$ time complexity, while (d) and (e) require $O(r)$ time at each step and, therefore, they too have a total cost in $O(n^2)$. In fact, none of these lines introduces a time complexity comparable to that of lines (d) and (e), which are already in NJ and whose cost is $O(n^3)$. Using identical implementation for both algorithms (Allegro Common Lisp 3.0 for Windows, and a Pentium 120 PC), NJ needs 0.027 s with $n = 12$, while BIONJ needs 0.033 s. With $n = 36$, the run times become, respectively, 0.64 and 0.68 s. Therefore, there is no practical difference between NJ's and BIONJ's run time. Concerning memory space, BIONJ has to store the variance matrix (v_{ij}) and thus needs twice as much memory space as does NJ. However, this has no practical consequence with modern computers. A simple solution consists of using a unique matrix, one half occupied by the distances (δ_{ij}) and the other half by the variances (v_{ij}).

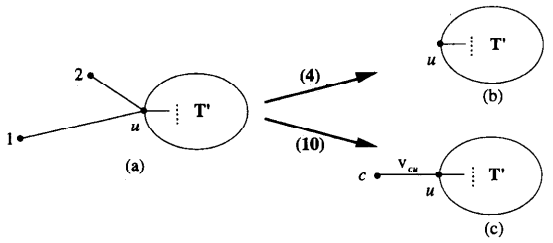

FIG. 2.—*a*, The true tree **T**. *b*, The distance tree after one reduction (4). *c*, The variance tree after one reduction (10).

Interpretation and Theoretical Comparison Between NJ and BIONJ

In order to simplify the analysis, let us assume that BIONJ is fed with the true evolutionary distances instead of their estimated values. Under this assumption, both distance and variance matrices are initially identical, and both may be represented by the true tree **T**. Once the first agglomeration has been achieved, the matrices differ. As has already been explained, the distance matrix may be represented by a subtree **T′** of **T**, which is obtained by deleting taxa 1 and 2. Reduction (10), which is identical to reduction (4) except for the constant term, also preserves the shape of **T**, but modifies the length of the branch attached to *c*, the center of the cluster {1, 2}. In fact, *c* is farther from the other taxa than is the cluster's root *u* (fig. 2). Using equations (4), (9), and (10), the expression of the additional length v_{cu} is given as

$$v_{cu} = \lambda_*^2 d_{1u} + (1 - \lambda_*)^2 d_{2u}, \qquad (11)$$

with

$$\lambda_* = \frac{1}{2} + \frac{d_{2u} - d_{1u}}{2(d_{1u} + d_{2u})}. \qquad (12)$$

During the next steps, things basically remain the same. Reduction (4) deletes pairs of neighbors, while reduction (10) replaces these neighbors by their center *c*, which is at some positive distance from their root *u*. These increments accumulate in some sense through formulae (9) and (10), and the clusters' centers become farther and farther from their root. These additional lengths, together with the rest of the tree, represent the variances of the reduced distance matrix. The situation therefore is all the more favorable when these increments are low. Let us consider equations (11) and (12) and examine the extreme cases:

1. The best possible situation brings a null increment ($v_{cu} = 0$). In this case, one of the branches d_{1u} or d_{2u} is null, and we have $\lambda_* = 1$, respectively $\lambda_* = 0$. If $d_{1u} = 0$, then 1 and *u* are identical, and reduction (10) consists of choosing for the cluster's center its root *u* (=1). The distances from taxon 2 are not taken into account in reduction (4), which may be easily interpreted, since we already have the data for an ancestor of 2 (1 = *u*). The second case is symmetrical.
2. The worst situation corresponds to $d_{1u} = d_{2u} = d_{12}/2$. Here, we have $v_{cu} = d_{12}/4$ and $\lambda_* = 1/2$. The reduction consists of the normal average between two variables which are nonindependent but have the same variance: $d_{12}/2 + d_{ui}$. Once the reduction has been achieved, the variance d_{ui} attached to the dependent part of the variables remains the same, while the rest is divided by 2, as expected.

In the latter case, the minimum variance reduction coincides with NJ's reduction (3). NJ is as good as possible, and the two algorithms are identical. In fact when systematically using $\lambda_* = 1/2$, we always obtain the same increment: $v_{cu} = d_{12}/4$, whatever the branch lengths d_{1u} and d_{2u}. It follows that in the first case, reduction (3) induces a variance much higher than the minimum variance reduction, since the increment is null for the latter. Therefore, it appears that, in contrast to NJ, BIONJ uses the fact that the neighboring branches do not necessarily have the same length, and that the shorter one induces a lower variance. This difference of branch length in the variance tree may occur for two reasons: either (the first situation, above) because both branches correspond to species which have evolved at different speeds or because one of the branches results from the agglomeration of numerous taxa, which reduces the variance, while the other does not. The first case is in contradiction with the molecular-clock hypothesis and will be the source of the most important differences between NJ and BIONJ; however, the second case may occur even when this hypothesis is satisfied. Although we have achieved a somewhat simplified analysis, the main difference between the two algorithms has been indicated.

Let us now examine the real situation where the algorithms are fed with the estimated values of the evolutionary distances, and not with their true values. Two cases can occur. In the first case, these estimated values are satisfactory, which basically means that they are not excessively biased. In this case, the value (9) of λ_* is better than the rough NJ approximation $\lambda_* = 1/2$, and BIONJ performs better than NJ. In the second case, these estimated values are inadequate and highly biased. The estimated value (9) of λ_* may then become worse than $\lambda_* = 1/2$, and NJ is better. Simulations presented in the next section show that such a situation is rare but may occur when the evolutionary distance estimate is very badly selected.

Simulation Results

Six model trees (fig. 3) were considered, each consisting of 12 taxa. The first two (A, B) satisfy the molecular-clock hypothesis, while the other four (C, D, E, F) present varying substitution rates among lineages. Trees A, B, C, and D were taken from Kumar (1996), and each consists of two copies of the same six-taxon tree, previously used by Saitou and Imanishi (1989) for similar purposes. Trees E and F are identical to trees C and D, respectively, except that the short external branches with length *b* have been replaced by longer branches having length 3*b*. Therefore, trees E and F may be seen as intermediate between the constant-rate (A, B) and the highly-varying-rate (C, D) trees. Simulations reported by Nei (1991), which relate to trees very similar

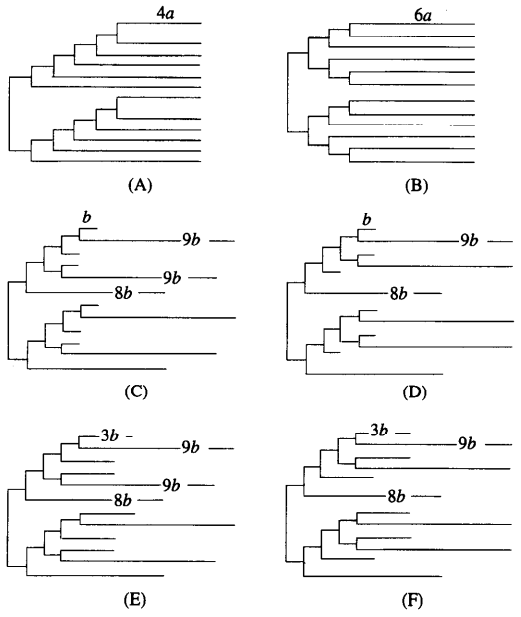

FIG. 3.—The model trees used for simulations. Each interior branch is one unit long (a for constant and b for variable-rate trees) and the lengths of external branches are given in multiples of a or b. Trees E and F are intermediate between the constant (A, B) and the highly-varying-rate (C, D) trees.

to A, B, C, and D, have shown that NJ is at least as good as all other methods with constant-rate trees, while with varying-rate trees, NJ is less efficient than the maximum-likelihood methods but remains comparable to or better than other methods.

The Kimura (1980) two-parameter model of sequence evolution was used with a transition/transversion ratio of 2. Each site evolved independently, starting from a random nucleotide sampled equiprobably from A, G, C, and T and simulating change according to the Markov chain specified by the Kimura model, with the substitution (transition + transversion) rate given by the length of that branch in the tree. Changes in different branches were independent, starting from the nucleotide that was achieved in the common ancestor of the branches. Evolutionary distances were computed using the standard Kimura (1980) estimate, except for the high/low per site condition, where we also used the two-parameter gamma-estimate (a = 1) of Jin and Nei (1990). Sequence lengths were equal to 300 or 600 sites, and four different conditions of evolution were considered:

1. Low substitution rates, which corresponded to a maximum pairwise divergence of about 0.1 substitutions per site and were obtained with a = 0.0055 and b = 0.004.
2. High substitution rates, which corresponded to a maximum pairwise divergence of about 1.0 substitutions per site and were obtained with a = 0.055 and b = 0.04.
3. Middle substitution rates, which seem more realistic than either of the previous extreme conditions and were obtained with a = 0.0275 and b = 0.02 (maximum pairwise divergence ≈0.5).
4. High/low per site substitution rates, along the lines of Kuhner and Felsenstein (1994). Within this condition, half of the sites evolved slowly in the sense previously defined (a = 0.0055 and b = 0.004), while the other half evolved quickly (a = 0.055 and b = 0.04). Then, the maximum pairwise divergence was about 0.55. This condition of evolution contradicts the usual assumption, which is made in the Kimura (1980) estimate, that all sites evolve identically. To a certain extent, the Jin and Nei (1990) two-parameter gamma estimate is more appropriate than that of Kimura, since it relaxes this hypothesis. However, the gamma distribution is clearly different from the bimodal distribution which was realized here. Therefore, this condition of evolution provides a view of the method's robustness with the Kimura estimate as well as with that of Jin and Nei, the perturbation being, however, stronger in the first case.

For each condition of evolution, each sequence length, and each model tree, 500 replications were performed. For each replicate data set, two criteria were measured to estimate and compare the performance of both algorithms:

1. The minimum-evolution (ME) criterion, using optimal estimates of branch lengths which were obtained by a method similar to that of Rzhetsky and Nei (1993). To evaluate the results, this criterion was also applied to the true tree **T** by fitting its branches with the matrix (δ_{ij}) just as was done for the estimated trees.
2. The Robinson and Foulds (1981) distance between the true tree and the estimated tree (RF). This topological distance is equal to the number of internal branches (or bipartitions) that exist in one tree but not in the other. For trees of 12 taxa it varies between 0 (identical topologies) and 18. Since both the true and the estimated trees are fully resolved, this criterion takes only even values. Thus, we display in the tables half of this criterion (i.e., RF/2), which can readily be interpreted as the number of incorrect branches in the estimated tree.

Simulations were performed on a PC, and the software (written in Allegro Common Lisp 3.0 for Windows) is available on request. The results are given in Tables 1–6. The main point, already suggested by the above theoretical analysis, is that BIONJ is much better than NJ with highly-varying-rate trees (C, D), while for constant-rate trees (A, B) the difference is slight. For example, with trees C and D the RF error reduction is, on the average 25%, while with trees A and B it is only 4%. Moreover, the probability of exactly finding the true tree is augmented up to 16% for C and D, while the gain is between −1% and 4% for A and B. In fact, when looking at the values of λ_*, we observe that they are usually very close to 0.5 in the case of trees A and B, which implies that BIONJ and NJ are almost identical, while they are much more variable with trees C and D. For example, we have observed the sequence (0.5, 0.5, 0.6, 0.6, 0.5, 0.5, 0.6, 0.6, 0.5) for one particular run of

Table 1
Results Obtained with Model Tree A

	No. of Sites		%ME<,>		RF	%RF = 0	%RF<,>	
Low	300	NJ	71	6	1.47	23	—	—
		BIONJ	73	5	1.43 (3)	22	10	6
	600	NJ	29	6	0.43	65	—	—
		BIONJ	32	3	0.43 (0)	64	3	4
Middle	300	NJ	45	12	0.76	44	—	—
		BIONJ	47	6	0.69 (8)	48	10	5
	600	NJ	14	4	0.20	82	—	—
		BIONJ	14	1	0.17 (14)	85	4	2
High/low	300	NJ	73	10	1.53	17	—	—
		BIONJ	78	4	1.46 (4)	18	16	9
	600	NJ	32	11	0.53	57	—	—
		BIONJ	37	3	0.48 (9)	60	10	6
High/low Jin and Nei estimate	300	NJ	77	10	1.81	13	—	—
		BIONJ	82	6	1.75 (4)	13	15	10
	600	NJ	39	12	0.70	49	—	—
		BIONJ	44	4	0.64 (9)	51	12	7
High	300	NJ	67	10	1.36	23	—	—
		BIONJ	73	5	1.30 (4)	22	15	10
	600	NJ	30	11	0.47	59	—	—
		BIONJ	34	3	0.43 (8)	63	11	7

NOTE.—%ME<,> provides the percentages of times where the estimated tree $\hat{\mathbf{T}}$ is shorter, respectively longer, than the true tree **T**, i.e., ME($\hat{\mathbf{T}}$) < ME(**T**), respectively ME($\hat{\mathbf{T}}$) > **ME(T)**; RF is half of the Robinson and Foulds distance between $\hat{\mathbf{T}}$ and **T**, and the number in parentheses is the percentage relative error reduction, i.e., 100(RF($\hat{\mathbf{T}}_{NJ}$, **T**) − RF($\hat{\mathbf{T}}_{BIONJ}$, **T**))/RF($\hat{\mathbf{T}}_{NJ}$, **T**); %RF = 0 is the percentage of times where the estimated and the true trees are identical; %RF<,> provides the percentages of times where BIONJ's topological accuracy is better, respectively worse, than NJ's, i.e., RF($\hat{\mathbf{T}}_{BIONJ}$, **T**) < RF($\hat{\mathbf{T}}_{NJ}$, **T**), respectively RF($\hat{\mathbf{T}}_{BIONJ}$, **T**) > RF($\hat{\mathbf{T}}_{NJ}$, **T**), and these numbers are underlined when their difference is statistically significant ($1 - \alpha \geq 95\%$).

Table 2
Results Obtained with Model Tree B

	No. of Sites		%ME<,>		RF	%RF = 0	%RF<,>	
Low	300	NJ	69	6	1.40	24	—	—
		BIONJ	70	6	1.38 (1)	24	8	6
	600	NJ	34	5	0.50	60	—	—
		BIONJ	35	4	0.49 (1)	61	3	2
Middle	300	NJ	52	11	0.92	37	—	—
		BIONJ	54	7	0.86 (7)	39	9	4
	600	NJ	15	5	0.21	80	—	—
		BIONJ	16	4	0.20 (1)	80	2	1
High/low	300	NJ	72	13	1.81	15	—	—
		BIONJ	76	10	1.80 (0)	14	10	11
	600	NJ	41	9	0.70	50	—	—
		BIONJ	43	6	0.68 (3)	52	6	4
High/low Jin and Nei estimate	300	NJ	79	8	1.97	13	—	—
		BIONJ	81	7	1.96 (1)	12	9	9
	600	NJ	47	12	0.86	41	—	—
		BIONJ	49	10	0.84 (3)	41	6	4
High	300	NJ	69	15	1.64	16	—	—
		BIONJ	71	11	1.58 (4)	18	14	9
	600	NJ	33	11	0.56	56	—	—
		BIONJ	34	9	0.55 (1)	57	6	5

NOTE.—See note to table 1.

Table 3
Results Obtained with Model Tree C

	No. of Sites		%ME<,>		RF	%RF = 0	%RF<,>	
Low	300	NJ	79	7	1.99	14	—	—
		BIONJ	74	11	1.94 (2)	15	19	14
	600	NJ	37	7	0.65	56	—	—
		BIONJ	36	8	0.61 (6)	56	9	6
Middle	300	NJ	32	8	0.57	60	—	—
		BIONJ	23	8	0.42 (26)	69	18	6
	600	NJ	4	2	0.07	94	—	—
		BIONJ	3	1	0.04 (46)	97	4	1
High/low	300	NJ	69	15	1.99	16	—	—
		BIONJ	63	16	1.88 (6)	20	26	17
	600	NJ	48	17	0.98	35	—	—
		BIONJ	39	19	0.88 (10)	42	18	10
High/low Jin and Nei estimate	300	NJ	50	14	1.18	36	—	—
		BIONJ	39	14	0.92 (22)	47	26	8
	600	NJ	16	8	0.31	76	—	—
		BIONJ	10	6	0.19 (39)	85	14	3
High	300	NJ	29	12	0.59	59	—	—
		BIONJ	18	7	0.37 (37)	75	24	5
	600	NJ	5	2	0.08	93	—	—
		BIONJ	1	1	0.03 (63)	98	6	1

NOTE.—See note to table 1.

BIONJ with tree B, and the sequence (0.1, 0.2, 0.2, 0.8, 0.4, 0.8, 0.1, 0.4, 0.7) for another run with tree D. However, the (estimated) values of λ_* rarely go outside [0, 1]. As expected, the results with moderately-varying-rate trees (E, F) are intermediate. For example, the RF error reduction for both trees is on the average 14%, while the probability of finding the correct tree is augmented up to 11%. Therefore, it appears that BIONJ preserves NJ's efficiency with constant-rate trees, while with varying-rate trees, it at least partially fills the gap, reported by Nei (1991) and others, between NJ and the maximum-likelihood methods.

Another noticeable result is that the difference between the algorithms is larger when the substitution

Table 4
Results Obtained with Model Tree D

	No. of Sites		%ME<,>		RF	%RF = 0	%RF<,>	
		NJ	79	7	2.01	14	—	—
Low	300	BIONJ	73	13	2.02 (−1)	13	14	15
	600	NJ	39	6	0.65	54	—	—
		BIONJ	33	8	0.57 (11)	59	11	4
Middle	300	NJ	26	13	0.56	61	—	—
		BIONJ	19	12	0.43 (23)	69	18	7
	600	NJ	4	4	0.08	92	—	—
		BIONJ	3	1	0.04 (56)	97	6	1
High/low	300	NJ	60	18	1.84	21	—	—
		BIONJI	49	24	1.74 (5)	27	26	19
	600	NJ	24	15	0.64	61	—	—
		BIONJ	18	24	0.66 (−2)	58	12	16
High/low Jin and Nei estimate	300	NJ	57	13	1.23	30	—	—
		BIONJ	44	15	0.99 (20)	41	30	12
	600	NJ	18	7	0.32	75	—	—
		BIONJ	9	5	0.19 (40)	86	15	3
High	300	NJ	34	13	0.77	53	—	—
		BIONJ	23	8	0.49 (36)	69	27	4
	600	NJ	7	4	0.13	89	—	—
		BIONJ	4	1	0.07 (51)	95	7	1

NOTE.—See note to table 1.

Table 5
Results Obtained with Model Tree E

	No. of Sites		%ME<,>	RF	%RF = 0	%RF<,>
Low	300	NJ	81 6	2.14	13	— —
		BIONJ	81 7	2.07 (3)	12	14 8
	600	NJ	41 9	0.70	50	— —
		BIONJ	41 8	0.70 (1)	52	7 6
Middle	300	NJ	39 15	0.78	46	— —
		BIONJ	35 10	0.62 (21)	55	20 8
	600	NJ	8 4	0.14	87	— —
		BIONJ	8 3	0.12 (12)	89	4 2
High/low	300	NJ	72 12	1.99	15	— —
		BIONJ	72 10	1.84 (8)	18	24 12
	600	NJ	46 11	0.84	42	— —
		BIONJ	42 11	0.78 (7)	47	12 8
High/low Jin and Nei estimate	300	NJ	62 14	1.65	23	— —
		BIONJ	60 12	1.54 (6)	28	23 14
	600	NJ	28 13	0.53	59	— —
		BIONJ	25 7	0.41 (23)	68	16 4
High	300	NJ	43 17	0.99	40	— —
		BIONJ	38 11	0.80 (20)	50	23 9
	600	NJ	12 7	0.22	82	— —
		BIONJ	11 3	0.16 (27)	86	8 2

NOTE.—See note to table 1.

rates increase. The most important gaps are observed with high substitution rates, while with low substitution rates, the results of the two algorithms are almost identical. With middle rates, results are intermediate. This may be understood by realizing that with low substitution rates, there are very few parallel or back substitutions. With the number of observed substitutions being practically equal to the number of realized substitutions, therefore, the distance matrix (δ_{ij}) is very close to a tree distance. The sampling variance affects the branch lengths of this tree (the number of realized substitutions may be quite different from the expected number of substitutions) but not its topology. As observed by Kumar (1996), in this case, the main difficulty the algorithms

Table 6
Results Obtained with Model Tree F

	No. of Sites		%ME<,>	RF	%RF = 0	%RF<,>
Low	300	NJ	80 6	2.12	14	— —
		BIONJ	80 6	2.10 (1)	14	12 11
	600	NJ	41 8	0.70	51	— —
		BIONJ	39 8	0.68 (2)	54	10 9
Middle	300	NJ	40 12	0.81	48	— —
		BIONJ	39 8	0.72 (11)	53	15 8
	600	NJ	10 5	0.16	86	— —
		BIONJ	7 3	0.10 (35)	90	7 1
High/low	300	NJ	72 12	1.92	16	— —
		BIONJ	72 10	1.83 (5)	19	21 14
	600	NJ	34 17	0.72	49	— —
		BIONJ	30 16	0.66 (8)	54	16 10
High/low Jin and Nei estimate	300	NJ	69 14	1.75	17	— —
		BIONJ	67 11	1.61 (8)	22	24 13
	600	NJ	33 13	0.64	54	— —
		BIONJ	29 8	0.49 (23)	64	16 4
High	300	NJ	49 14	1.04	37	— —
		BIONJ	41 13	0.83 (20)	46	23 8
	600	NJ	14 7	0.25	79	— —
		BIONJ	9 3	0.14 (44)	88	11 1

NOTE.—See note to table 1.

are faced with is that some branches are not supported by any substitution. Clearly, BIONJ may not solve this difficulty any better than does NJ. However, this phenomenon is reduced as the sequence length increases or as the substitution rates become higher.

With the high/low per site condition, the performance of both algorithms decreases, as expected. With varying-rate trees (C, D, E, F) the results are clearly better when using the Jin and Nei (1990) gamma-estimate than when using the Kimura estimate; however, the inverse holds, to a certain extent, with constant-rate trees (A, B). When using the Kimura estimate, the gain obtained by BIONJ is slight, and BIONJ is even less efficient than NJ in one case (tree D, 600 sites). This illustrates that BIONJ may sometimes be less robust than NJ. However, this is probably rare, since BIONJ is better than NJ in all other cases. Moreover, the high/low per site condition strongly violates the assumptions of the Kimura estimate. When using the Jin and Nei (1990) gamma-estimate, which is imperfect but more appropriate than that of Kimura, the gain obtained by BIONJ becomes high again, and comparable with that obtained with the middle condition. From another point of view, this result demonstrates that BIONJ is not dedicated to the Kimura estimate, and that it is profitable to use it once the evolutionary distance estimate is sufficiently well adapted to the data.

The sequence length has an influence on the type of gain which may be expected from BIONJ. The absolute error reduction is higher with short sequences than with long sequences, but the contrary holds for the relative error reduction. The probability that the two methods will differ is also much higher with short sequences than with long sequences; however, with long sequences, the probability that NJ will be better than BIONJ becomes very low, close to 0. The tree topology also has a certain influence. Few differences are observed between trees C and D, respectively E and F, because their topologies are close. However, the gain is much higher with tree A than with tree B, the topologies of which are very different. Good results obtained with tree A derive from the fact that with such a topology we frequently have to agglomerate a single original taxon with a cluster which already contains numerous taxa; such a situation induces branches with different lengths in the variance tree and is advantageous to BIONJ (see above).

BIONJ finds trees which are not shorter than NJ's in the sense of the ME criterion. Specifically, it seems that BIONJ trees are just a little shorter with constant-rate trees but longer with varying-rate trees—where BIONJ outperforms NJ. Moreover, it appears that trees found by both NJ and BIONJ are more often too short (i.e., shorter than the true tree) than too long. This explains why searching for trees shorter than NJ trees may not increase the topological accuracy.

Conclusion

We have presented an improved version of NJ which is well adapted to cases where evolutionary distances are obtained from aligned sequences. This new algorithm, BIONJ, uses a simple model of the sampling noise of evolutionary distances. Thus, it takes into account the fact that high evolutionary distances present a higher variance than do short distances. The covariances of evolutionary distances are also taken into account. Theoretical reasons, as well as the computer simulations we have performed, show that BIONJ has an expected topological accuracy greater than (or equal to) that of NJ, provided our sampling noise model is satisfactory and a reasonable estimate of the evolutionary distance has been selected. BIONJ is a very simple algorithm which requires about the same computational time as does NJ. Simulation results show that BIONJ is only slightly better than NJ when the substitution rates are low or when they are constant among lineages. When the substitution rates are higher and vary among lineages, BIONJ clearly has a better topological accuracy than NJ. Then, the error reduction may rise above 50%, and the probability of finding the correct tree may be augmented by more than 15%.

Consequently, it seems to us that BIONJ ought to be widely used when one has evolutionary distances which satisfy the algorithm's hypotheses (eq. 7). Nevertheless, interesting and important work remains to be done concerning BIONJ. This includes systematic comparison with other approaches under various conditions of evolution. It is expected that branch length estimates could be obtained which would be more consistent with the rest of the approach. Also, further exploration concerning the relationships of this theory with that of generalized least-squares is envisaged.

Acknowledgments

I thank Manolo Gouy, Alain Guénoche, Andrey Rzhetsky, and Mike Steel and for their helpful comments on earlier versions of this paper.

LITERATURE CITED

ATTESON, K. 1996. An analysis of the performance of the neighbor-joining method of phylogeny reconstruction. DIMACS Workshop on Mathematical Hierarchies and Biology, November 13–15, Rutgers University, N.J.

BARTHÉLEMY, J. P., and A. GUÉNOCHE. 1991. Trees and proximity representations. Wiley, Chichester.

BULMER, M. 1991. Use of the method of generalized least squares in reconstructing phylogenies from sequence data. Mol. Biol. Evol. **8**:868–883.

CHARLESTON, M. A., M. D. HENDY, and D. PENNY. 1993. Neighbor-joining uses the optimal weight for net divergence. Mol. Phylogenet. Evol. **2**:6–12.

———. 1994. The effect of sequence length, tree topology, and number of taxa on the performance of phylogenetic methods. J. Comput. Biol. **1**:133–151.

FELSENSTEIN, J. 1987. Estimation of hominoid phylogeny from a DNA hybridization data set. J. Mol. Evol. **26**:123–131.

GASCUEL, O. 1994. A note on Sattath and Tversky's, Saittou and Nei's and Studier and Keppler's algorithms for inferring phylogenies from evolutionary distances. Mol. Biol. Evol. **11**:961–963.

———. 1997. Concerning the NJ algorithm and its unweighted version, UNJ. In B. MIRKIN, F. R. MCMORRIS, F. S. ROB-

ERTS, and A. RZHETSKY, eds. Proceedings of the DIMACS Workshop on Mathematical Hierarchies and Biology. American Mathematical Society, Providence, R.I. (in press).

JIN, L., and M. NEI. 1990. Limitations of the evolutionary parsimony method of phylogenetic analysis. Mol. Biol. Evol. **7**:82–102.

JUKES, T. H., and C. R. CANTOR. 1969. Evolution of protein molecules. Pp. 21–132 *in* H. N. MUNRO, ed. Mammalian protein metabolism. Academic Press, New York.

KIDD, K. K., and L. A. SGARAMELLA-ZONTA. 1971. Phylogenetic analysis: concepts and methods. Am. J. Hum. Genet. **23**:235–252.

KIMURA, M. 1980. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. J. Mol. Evol. **16**:111–120.

———. 1983. The neutral theory of molecular evolution. Cambridge University Press, Cambridge.

KUHNER, M. K., and J. FELSENSTEIN. 1994. A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. Mol. Biol. Evol. **11**:459–468.

KUMAR, S. 1996. A stepwise algorithm for finding minimum evolution trees. Mol. Biol. Evol. **13**:584–593.

LAWSON, C. M., and R. J. HANSON. 1974. Solving least squares problems. Prentice Hall, Englewood Cliffs, N.J.

NEI, M. 1991. Relative efficiencies of different tree-making methods for molecular data. Pp. 90–128 *in* M. M. MIYAMOTO and J. L. CRACRAFT, eds. Phylogenetic analysis of DNA sequences. Oxford University Press, Oxford.

NEI, M., and L. JIN. 1989. Variances of the average numbers of nucleotide substitutions within and between populations. Mol. Biol. Evol. **6**:290–300.

ROBINSON, D. F., and L. R. FOULDS. 1981. Comparison of phylogenetic trees. Math. Biosci. **53**:131–147.

RZHETSKY, A., and M. NEI. 1993. Theoretical foundation of the minimum-evolution method of phylogenetic inference. Mol. Biol. Evol. **10**:1073–1095.

———. 1994. METREE: a program package for inferring and testing minimum-evolution trees. Comput. Appl. Biosci. **10**: 409–412.

SAITOU, N., and M. IMANISHI. 1989. Relative efficiencies of the Fitch-Margoliash, maximum-parsimony, maximum-likelihood, minimum-evolution, and neighbor-joining methods of phylogenetic reconstructions in obtaining the correct tree. Mol. Biol. Evol. **6**:514–525.

SAITOU, N., and M. NEI. 1987. The neighbor-joining method: a new method for reconstruction of phylogenetic trees. Mol. Biol. Evol. **4**:406–425.

SATTATH, S., and A. TVERSKY. 1977. Additive similarity trees. Psychometrika **42**:319–345.

STUDIER, J. A., and K. J. KEPPLER. 1988. A note on the neighbor-joining method of Saitou and Nei. Mol. Biol. Evol. **5**: 729–731.

SWOFFORD, D. L., G. J. OLSEN, P. J. WADDELL, and D. M. HILLIS. 1996. Phylogenetic Inference. Pp. 402–514 *in* D. M. HILLIS, C. MORITZ, and B. K. MABLE, eds. Molecular systematics. 2nd edition. Sinauer, Sunderland, Mass.

VACH, W. 1989. Least-squares approximation of additive trees. Pp. 230–238 *in* O. OPITZ, ed. Conceptual and numerical analysis of data. Springer, Heidelberg.

ZHARKIKH, A. 1994. Estimation of evolutionary distances between nucleotide sequences. J. Mol. Evol. **39**:315–329.

MANOLO GOUY, reviewing editor

Accepted April 1, 1997