A Fast Method for Approximating Maximum Likelihoods of Phylogenetic Trees from Nucleotide Sequences

JAMES S. ROGERS¹ and DAVID L. SWOFFORD²

¹Department of Biological Sciences, University of New Orleans, New Orleans, Louisiana 70148, USA; E-mail: jsrbs@uno.edu
²Laboratory of Molecular Systematics, MRC-534, Smithsonian Institution, Washington, D.C. 20560, USA; E-mail: swofford@onyx.si.edu

Abstract.—We have developed a rapid parsimony method for reconstructing ancestral nucleotide states that allows calculation of initial branch lengths that are good approximations to optimal maximum-likelihood estimates under several commonly used substitution models. Use of these approximate branch lengths (rather than fixed arbitrary values) as starting points significantly reduces the time required for iteration to a solution that maximizes the likelihood of a tree. These branch lengths are close enough to the optimal values that they can be used without further iteration to calculate approximate maximum-likelihood scores that are very close to the “exact” scores found by iteration. Several strategies are described for using these approximate scores to substantially reduce times needed for maximum-likelihood tree searches. [Approximations; maximum likelihood; parsimony; tree searching.]

In the past decade major advances in the analysis of nucleotide sequences have led to widespread use of data from DNA and RNA to study the phylogenetic relationships of genes and organisms (Hillis et al., 1996). One difficulty facing workers in this field is choosing among the great number of methods that may be used to estimate phylogenies from sequences (Swofford et al., 1996). In recent years several researchers have compared the efficiencies of many of these methods in recovering correct phylogenetic tree topologies and in estimating branch lengths of trees (e.g., Huelsenbeck and Hillis, 1993; Hillis et al., 1994; Kuhner and Felsenstein, 1994; Tateno et al., 1994; Yang, 1994; Huelsenbeck, 1995). In several of these studies the maximum-likelihood method (Felsenstein, 1981) has been found to be equal or superior in efficiency to most other methods under a number of different models of nucleotide substitution. However, this method has been handicapped by the slowness of the algorithms for implementing it (Kuhner and Felsenstein, 1994; Tateno et al., 1994).

A major component of the computational effort involves the estimation of branch lengths (expected amounts of change between each pair of adjacent nodes) that allow maximization of the likelihood for a specific tree topology. A common strategy for optimizing branch lengths is to initialize them to arbitrary values and then use an iterative procedure to improve these starting values. This strategy is used in DNAML from the PHYLIP package (Felsenstein, 1993) and in fastDNAml (Olsen et al., 1994) for optimizing user-defined trees. Examination of the amount of time required for the arbitrary branch lengths to converge to values that are reasonably close to the final values suggests that beginning the iterations from a more advanced starting point might lead to a speeding up in the total computation time. We describe here a method for estimating initial branch lengths that significantly reduces the amount of time required for finding the maximum-likelihood branch lengths of most trees and for conducting maximum-likelihood tree searches. Likelihood scores approximated by our method can be used to quickly reject trees of very low likelihood, focusing the search on those that have a reasonable chance of achieving maximal likelihood when branch lengths are iterated to their optimal values.

MAXIMUM-PARSIMONY APPROXIMATION TO THE MAXIMUM-LIKELIHOOD SOLUTION

Evaluating the likelihood of a tree with known sequences at the terminal nodes
and unknown sequences at the internal, or ancestral, nodes involves, in effect, summing the likelihoods of the tree over all possible reconstructions of the ancestral sequences, given some model of nucleotide substitution and a set of branch lengths of the tree (Felsenstein, 1981). Thus, the total likelihood \( L \) of a tree can be represented as

\[
L = \sum_{i=1}^{r} L_i \quad (1)
\]

where \( r \) is the number of possible reconstructions. For an unrooted, bifurcating tree with \( s \) terminal sequences, \( s - 2 \) ancestral sequences, and sequences \( n \) nucleotides long, the number of possible ancestral reconstructions is given by

\[
r = 4^{n(s-2)} - 2^{2n(s-2)} \quad (2)
\]

This number is astronomical for even very small data sets. For example, for four terminal sequences of 50 nucleotides \( r = 1.6 \times 10^{60} \). (Fortunately, by assuming independence of sites and employing Felsenstein’s [1973, 1981] “pruning” algorithm, it is possible to evaluate the total \( L \) of a tree without calculating the separate likelihoods of the \( r \) reconstructions.) The nearly infinite number of possible ancestral reconstructions suggests that a very large proportion of them, particularly those that require very many nucleotide substitutions, contribute little to \( L \), unless all of the branches of the tree are very long, in which case all reconstructions contribute equally.

This intuitive notion can be substantiated by considering the evaluation of \( L \) for an unrooted, bifurcating tree under Jukes and Cantor’s (1969) model of nucleotide substitution in the absence of a molecular clock (i.e., substitution rates may vary over branches). According to this model the likelihood of a single reconstruction is given by

\[
L_i = \frac{1}{4^n} \prod_{j=1}^{2s-3} \left( \frac{1 + 3z_{ij}}{4} \right)^{n(1-d_{ij})} \left( \frac{1 - z_{ij}}{4} \right)^{nd_{ij}} \quad (3)
\]

where \( 2s - 3 \) is the number of branches of an unrooted, bifurcating tree with \( s \) terminal sequences, \( d_{ij} \) is the proportion of sites that differ between the ends of branch \( j \) in reconstruction \( i \) and \( z_{ij} = \exp(-4\alpha t_j) \), \( \alpha \) being the rate of substitution from any one nucleotide to any other and \( t_j \) the time over branch \( j \). Taking the first partial derivative of expression 3 for some branch \( k \) gives

\[
\frac{\partial L_i}{\partial z_k} = nL_i \left[ \frac{3(1-d_{ik})}{1 + 3z_k} - \frac{d_{ik}}{1 - z_k} \right] \quad (4)
\]

Combining Equations 1 and 4 results in

\[
\frac{\partial L}{\partial z_k} = n \left[ \frac{3 \sum (1-d_{ik})L_i}{1 + 3z_k} - \sum d_{ik}L_i \right] \quad (5)
\]

Then

\[
\frac{\partial \log L}{\partial z_k} = \frac{1}{L} \frac{\partial L}{\partial z_k} = n \left[ \frac{3(1-\bar{d}_k)}{1 + 3z_k} - \frac{\bar{d}_k}{1 - z_k} \right] \quad (6)
\]

where

\[
\bar{d}_k = \sum_i d_{ik}L_i/L \quad (7)
\]

is the average over all reconstructions of the proportion of differences on branch \( k \), weighted by the contribution of each reconstruction to \( L \). Setting expression 6 equal to zero and solving for \( z_k \) results in

\[
\hat{z}_k = 1 - \frac{4}{3}\bar{d}_k \quad (8)
\]

and

\[
\hat{D}_k = -\frac{3}{4}\ln \left( 1 - \frac{4}{3}\bar{d}_k \right) \quad (9)
\]

where \( \hat{D}_k \) is the maximum-likelihood Jukes–Cantor length for branch \( k \). Thus, the maximum-likelihood branch length is a function of the mean observed difference over all reconstructions, weighted by the contributions of the reconstructions to the total likelihood. The reconstructions with highest individual likelihoods will therefore contribute most heavily to the determination of branch lengths. Which reconstructions will contribute most heavily? On substituting expression 8 into Equation 3, taking the log of both sides, and rearranging terms,
\[
\frac{1}{n} \log L_i = -\log 4 + \sum_j \log(1 - \tilde{d}_j) \\
- \sum_j d_{ij} \log \left[ \frac{3(1 - \tilde{d}_j)}{\tilde{d}} \right] 
\] (10)

Because, according to the Jukes–Cantor model, it is expected that \( 0 \leq \tilde{d}_j < \frac{3}{4} \), the log of the term in square brackets in Equation 10 is expected always to be positive. Thus, \( L_i \) is a decreasing function of the \( d_{ij} \) values, resulting in low likelihood for reconstructions that require many changes. Conversely, reconstructions that require few changes will have relatively high likelihoods and will dominate the determination of \( L \) and the \( d_{ij} \) values. In fact, if all branches of the tree have the same expected length, such that \( \tilde{d}_j = \tilde{d} \) for all \( j \), then Equation 10 becomes

\[
\frac{1}{n} \log \hat{L}_i = -\log 4 + (2s - 3)\log(1 - \tilde{d}) \\
- \log \left[ \frac{3(1 - \tilde{d})}{\tilde{d}} \right] \sum_j d_{ij} 
\] (11)

from which it is obvious that in this case, but only in this case, the reconstructions of highest individual likelihood are always those that minimize the total number of observed substitutions over the whole tree, that is, the most parsimonious reconstructions (MPRs). It can also be seen from Equation 11 that this effect will be much stronger for very short trees than for very long trees. As \( \tilde{d} \) approaches its maximum, \( \frac{3}{4} \) the last term in Equation 11 approaches zero for all reconstructions, giving them all approximately the same likelihood. Conversely, as \( \tilde{d} \) approaches zero the last term becomes so large that the likelihoods of reconstructions other than the maximum-parsimony ones become negligible by comparison, making the maximum-likelihood and maximum-parsimony trees equivalent. The equivalence of maximum likelihood and parsimony under very small expectation of evolution has been noted by other workers, such as Felsenstein (1973) and Goldman (1990).

Even when the branches are of unequal lengths it is clear from Equation 10 that the MPRs will have relatively high likelihoods, although not necessarily the highest likelihoods, as demonstrated by Yang et al. (1995). However, not all MPRs will have the same likelihood. Those that require few changes on branches with small \( \tilde{d}_j \) will have higher likelihoods than others. They, in turn, contribute more to the total likelihood of the tree and to the determination of the \( d_{ij} \) values. Such a reconstruction may be used to get “close” approximations of the maximum-likelihood estimates of the branch lengths of the tree, which may, in turn, be used to approximate the total maximum likelihood of the tree. Other workers (e.g., Farris, 1973, 1977; Felsenstein, 1973; Barry and Hartigan, 1987; Yang et al., 1995) have explored the relationship between MPRs and maximum likelihood but have not suggested using them to approximate maximum-likelihood branch lengths. However, Yang and Kumar (1996) used all MPRs weighted equally to estimate patterns of nucleotide substitution and of rate variation among sites. Adachi and Hasegawa (1995) and Waddell (1995) calculated approximate likelihoods (called “non-iterated” likelihoods by Waddell) from least-squares estimates of branch lengths. Adachi and Hasegawa (1995) recommended using these approximate likelihoods to quickly eliminate unpromising trees during tree searches.

A data set for given sequences and an unrooted tree may permit several to many MPRs. For example, the four sequences and the tree of Figures 1a and 1b permit 24 MPRs. These reconstructions and the likelihoods calculated from their branch lengths according to the Jukes–Cantor model are shown in Table 1. We see from this table that MPR 13 has the highest log likelihood (−122.344), making it the best approximator of the exact likelihood (−122.237) for this example. For a very small data set such as this, it would be feasible to examine all of the MPRs to find the best one. However, for larger data sets this can be prohibitively time-consuming, making it desirable to have a more direct search procedure. The following algorithm was designed to accomplish this goal. Although
Table 1. All 24 most parsimonious reconstructions (MPRs) and their approximate likelihoods for the four terminal sequences and unrooted tree of Figure 1. MPR 13 has the highest approximate likelihood and is also the MPR selected by the algorithm illustrated in Figure 1.

<table>
<thead>
<tr>
<th>MPR no.</th>
<th>Node 5</th>
<th>Node 6</th>
<th>−ln L</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AACAAAA</td>
<td>GACAAAA</td>
<td>123.104</td>
</tr>
<tr>
<td>2</td>
<td>AACAAAG</td>
<td>GACAAAG</td>
<td>123.133</td>
</tr>
<tr>
<td>3</td>
<td>AACAAAT</td>
<td>GACAAAT</td>
<td>122.835</td>
</tr>
<tr>
<td>4</td>
<td>AACAAAC</td>
<td>GACAAAC</td>
<td>123.073</td>
</tr>
<tr>
<td>5</td>
<td>AACAAAA</td>
<td>GACAAAT</td>
<td>122.463</td>
</tr>
<tr>
<td>6</td>
<td>AACAAAA</td>
<td>GACAAAC</td>
<td>122.794</td>
</tr>
<tr>
<td>7</td>
<td>AACAAAG</td>
<td>GACAAAT</td>
<td>122.516</td>
</tr>
<tr>
<td>8</td>
<td>AACAAAG</td>
<td>GACAAAC</td>
<td>122.843</td>
</tr>
<tr>
<td>9</td>
<td>AACAAAA</td>
<td>GTCAAAC</td>
<td>122.463</td>
</tr>
<tr>
<td>10</td>
<td>AACAAAG</td>
<td>GTCAAAG</td>
<td>122.516</td>
</tr>
<tr>
<td>11</td>
<td>AACAAAT</td>
<td>GTCAAAT</td>
<td>122.416</td>
</tr>
<tr>
<td>12</td>
<td>AACAAAC</td>
<td>GTCAAAC</td>
<td>122.477</td>
</tr>
<tr>
<td>13</td>
<td>AACAAAA</td>
<td>GTCAAAT</td>
<td>122.344</td>
</tr>
<tr>
<td>14</td>
<td>AACAAAA</td>
<td>GTCAAAC</td>
<td>122.532</td>
</tr>
<tr>
<td>15</td>
<td>AACAAAG</td>
<td>GTCAAAT</td>
<td>122.416</td>
</tr>
<tr>
<td>16</td>
<td>AACAAAG</td>
<td>GTCAAAC</td>
<td>122.603</td>
</tr>
<tr>
<td>17</td>
<td>AACAAAA</td>
<td>GCCAAAA</td>
<td>122.794</td>
</tr>
<tr>
<td>18</td>
<td>AACAAAG</td>
<td>GCCAAAG</td>
<td>122.843</td>
</tr>
<tr>
<td>19</td>
<td>AACAAAT</td>
<td>GCCAAAT</td>
<td>122.477</td>
</tr>
<tr>
<td>20</td>
<td>AACAAAC</td>
<td>GCCAAC</td>
<td>123.134</td>
</tr>
<tr>
<td>21</td>
<td>AACAAAA</td>
<td>GCCAAAT</td>
<td>122.532</td>
</tr>
<tr>
<td>22</td>
<td>AACAAAA</td>
<td>GCCAAC</td>
<td>123.219</td>
</tr>
<tr>
<td>23</td>
<td>AACAAAG</td>
<td>GCCAAAT</td>
<td>122.603</td>
</tr>
<tr>
<td>24</td>
<td>AACAAAG</td>
<td>GCCAAC</td>
<td>123.277</td>
</tr>
</tbody>
</table>

a Reconstructions are shown for only the variable sites, 1–7.

it is not guaranteed to find the best reconstruction in every case, it appears to work well in practice.

Step 1. For each ancestral sequence and each site, find all most parsimonious reconstruction sets (MPR sets) by some method (e.g., Fitch, 1971; Hartigan, 1973; Swofford and Maddison, 1992), holding fixed (unambiguous MPR set) all sequences and sites that were fixed in earlier cycles of the algorithm. If all sequences and sites are now fixed, STOP; otherwise, go to Step 2.

Step 2. For each branch of the tree, count the number of sites at which the two connected sequences are fixed for the same nucleotide (S) or potentially have the same nucleotide (PS), that is, at least one of the two sequences has an ambiguous MPR set and their MPR sets intersect.

Step 3. Rank all branches in decreasing order of S + PS. If two branches have the same value for this sum, rank the one with larger PS ahead of the other.

Step 4. For each site proceed through the branches in rank order until a branch is found for which one of the two connected sequences is unambiguous, the other sequence is ambiguous, and the MPRs of the two intersect. Set the ambiguous sequence equal to the fixed sequence at that site. When one such branch has been found, or none are found, go to the next site. (No more than one branch may be fixed in this step at any site in a single cycle of the algorithm. Ambiguous sites, that is, gaps or polymorphisms, in terminal sequences are also fixed by this step.)

Step 5. If all sequences have now been fixed at all sites, STOP; otherwise, go to Step 1.

The goal of this algorithm is to find a maximum-parsimony reconstruction that makes necessarily short branches (i.e., those that will be short under any MPR) as short as possible, thus forcing as many changes as possible onto necessarily longer branches.

Figures 1c–j illustrate the operation of the algorithm for the example of Figure 1 and Table 1. In Step 1 the MPR sets of nodes 5 and 6 for the seven variable sites are found. The MPR set of node 6 at site 2 and the MPR sets of nodes 5 and 6 at site 7 are ambiguous; all other MPR sets are unambiguous (fixed).

Because there are ambiguous MPR sets we go to Step 2 where S and PS values are determined for each branch. Taking branch 1, which connects nodes 1 and 5, for example, the two connected nodes are fixed for different states at site 5 and fixed for the same state at 48 sites, resulting in S = 48. At site 7 node 1 is fixed for A and node
**Figure 1.** Illustration of the procedure for finding a maximum parsimony reconstruction with branch lengths that are close to maximum-likelihood branch lengths. (a) Four nucleotide sequences of 50 sites each, 7 variable and 43 invariant. (b) Given unrooted, dichotomous tree with four terminal nodes, two ancestral nodes, and five branches. (c–j) Steps as discussed in the text.
5 is ambiguous, but includes A in its MPR set, resulting in $PS = 1$.

In Step 3 the branches are ranked, first in descending order of their $S + PS$ values and then in descending order of $PS$ values. For example, branch 3 is ranked first because it has the highest value of $S + PS$, and branch 5 is ranked second because it has a lower $S + PS$ than branch 3 but a higher $PS$ than branch 1.

In Step 4 each site with an ambiguous MPR set (2 and 7) is visited. For each of these sites the branches of the tree are visited in the order determined in Step 3 until a branch is found for which one connected node has a fixed MPR set, the other node has an ambiguous MPR set, and the two MPR sets intersect, that is, have a state in common. In this example this is branch 3 for both sites. So for branch 3 the state of node 6 is set equal to the state of node 3 at both sites. Because no more than one such assignment is permitted at each site in each cycle of the algorithm, site 7 is left ambiguous at node 5.

In Step 5 it is determined that at least one ambiguous MPR set (node 5, site 7) remains, so the procedure returns to Step 1.

In Step 1' site 7 is revisited, because it is the only remaining site with an ambiguous MPR set. The ambiguous MPR set at node 5 of this site is reconstructed, holding fixed the state of node 6, which was fixed in Step 4 given earlier. With this constraint, the MPR set of node 5 changes from $(A, G, T, C)$ to $(A, G, T)$. In this case the MPR set has changed but remains ambiguous. In other cases a previously ambiguous MPR set may be fixed at this step.

Because there is still at least one ambiguous MPR set on the tree, the procedure goes to Step 2' where the $S$ and $PS$ values are recalculated and to Step 3' where the branches are ranked again. In the latter step branches 3 and 4 are not ranked because in both cases the two connected nodes have unambiguous MPRs at all sites, as indicated by their $PS$ values of zero. At Step 4' the state of node 5 is set equal to the state of node 1 at site 7, because branch 1 was ranked highest in Step 3'. Now that all MPR sets on the tree have been unambiguously determined the search stops. The "best" reconstruction that has been found at this point is number 13 of Table 1, which is indeed the reconstruction of highest likelihood.

Although the theoretical justification for this algorithm was couched in terms of the Jukes–Cantor model of nucleotide substitution, the description and the example make clear that the implementation of the algorithm does not require any particular substitution model. The reconstruction that it produces permits the frequencies of all possible classes of substitutions to be estimated for each branch of a tree. These can then be used to calculate branch lengths and likelihoods according to any model of substitution. This algorithm and all other procedures described herein have been implemented in the PAUP* computer program, version 4.0 (currently available as a test version from D.L.S.).

**Examples**

We illustrate the approximate likelihood and exact likelihood procedures described earlier using four data sets taken from the literature: primate mtDNA (12 sequences, 898 sites; Hayasaka et al., 1988), corvine cytochrome-\textit{b} genes (11 sequences, 1,143 sites; Helm-Bychowski and Cracraft, 1993), pocket gopher cytochrome-\textit{b} (14 sequences, 399 sites; Patton and Smith, 1994), and chordate ribosomal RNA (28 sequences, 2,046 sites; extracted from the RDP database, Maidak et al., 1994). In each case we used the alignments of the original authors. For each data set we selected 100 or more optimal or nearly optimal ("short") trees under the maximum-parsimony criterion and 100 trees generated randomly under a model in which each possible tree is equally probable. Note that the set of "short" trees is also likely to contain trees that fit well according to the maximum-likelihood criterion, due to the high correlation between parsimony and likelihood scores in many data sets (e.g., DeBry and Abele, 1995). For each set of trees we then computed the approximate likelihoods and exact likelihoods under four different substitution models: JC (Jukes and Cantor,
Figure 2. Comparison of approximate negative log likelihoods (−ln L) from the parsimony approximation procedure to "exact" −ln L derived by iteration for 100 random trees and four data sets. All likelihoods were computed with the HKY (Hasegawa et al., 1985) model of nucleotide substitution. (a) Primate mtDNA (Hayasaka et al., 1988). (b) Corvine cytochrome-b (Helm-Bychowski and Cracraft, 1993). (c) Pocket gopher cytochrome-b (Patton and Smith, 1994). (d) Chordate ribosomal RNA (Maidak et al., 1994).

1969), K2P (Kimura, 1980), F84 (from J. Felsenstein’s DNAML program, described by Kishino and Hasegawa, 1989; see also Tateno et al., 1994), and HKY (Hasegawa et al., 1985). In each case the approximate likelihoods were compared to the exact likelihoods to determine the accuracy of the approximations. The results are shown in Figures 2–4 and Table 2.

For three substitution models, JC, K2P, and F84, we compared the exact likelihoods found by PAUP* to those found by fastDNAml (Olsen et al., 1994) as an independent check on the accuracy of the PAUP* values. We also recorded the computer run times for each combination of data set, tree set, substitution model, computer program, and approximate versus exact likelihood. The run times are shown in Figure 5.

For all runs with fastDNAml we began iteration from the program’s default value of 0.9 for the parameter $z = \exp(-4\alpha t)$, where $\alpha$ is the substitution rate in the JC model. We used two different starting points for finding exact likelihoods with PAUP*, the parsimony approximation branch lengths and an arbitrary branch length of $0.079 = -0.75 \ln(0.9)$, which is equivalent to the default $z$ value of fastDNAml for the JC model.

The parsimony approximation branch lengths, whether used directly for calculating approximate likelihoods or as starting
points for iteration to the "exact" likelihoods, were calculated with the appropriate distance formula, that is, JC distance for the JC model, K2P distance for the K2P model, and F84 distance for the F84 and HKY models. In the latter case, the F84 distance was used with the HKY model because there is not a closed-form expression for HKY distances, but the model is rather similar to the F84 model.

All of the computations discussed in this paper were done on a 133-MHz DEC Alpha 3000/400 workstation running the OSF/1 (Digital Unix) operating system.

RESULTS AND DISCUSSION

It is clear from Figures 2-4 and Table 2 that there is a strong correlation between the exact likelihoods found by iteration and the approximate likelihoods calculated from the branch lengths of the parsimony reconstructions. The deviations of approximate from exact values are generally small, but increase somewhat with increasing complexity of the substitution model (HKY ≈ F84 > K2P > JC), with decreasing likelihood of trees, and with increasing size of the data sets. Note that the apparently poorer fit for the "short" trees
(Fig. 3) is an artifact of the axis scaling; combining results from the “short” and random trees into a single scattergram (Fig. 4) highlights the tightness of the correlation over a wide range of tree likelihoods.

Due to the tightness of the correlations and the small deviations for the more likely trees, we envision several possible strategies for using approximate likelihoods to reduce the computation time required for maximum-likelihood tree searches. First, starting the iteration for optimal branch lengths from the lengths implied by the parsimony reconstruction significantly reduces the amount of time needed to calculate the “exact” likelihoods of individual trees (Fig. 5). Although the amount of this improvement depends on the data sets and tree sets, calculation of exact likelihoods is always faster with the parsimony-derived branch lengths, sometimes by a factor of 3 or more.

An additional strategy is suggested by the deviation scores presented in Table 2. Because the deviations were never greater than 2.8% for random trees or 0.6% for short trees, the approximate likelihoods represent a potentially effective criterion for quickly rejecting trees that have little chance of becoming optimal upon further branch-length optimization. One way to accomplish this is to set a “threshold” score based on a percentage of the best likelihood score found so far at any point during a search (the “target” score). If the approximate likelihood is worse than this threshold value, the tree is discarded with-
out wasting time on further branch-length optimization, under the assumption that the exact likelihood of this tree would still be worse than the target score. To test this strategy, we performed heuristic searches under the HKY substitution model on the 28-taxon chordate rRNA data set with PAUP*, using stepwise addition and TBR branch swapping. A search in which exact likelihoods were calculated for every tree evaluated required 165 min of CPU time. Using PAUP*'s default of rejecting trees exceeding the target score by more than 5%, the search time was reduced to 145 min. Based on the deviations apparent from Table 2, however, this default is unnecessarily conservative. Searches with 2%, 1%, and 0.5% thresholds required 92, 58, and 48 min, respectively. In every case, the same optimal tree was selected.

The differences in run times with different threshold values raise the issue of how to choose a good threshold value. If the threshold is set too low, trees that would have been retained had full branch-length optimization been performed may be rejected based on the approximate scores. In the chordate rRNA example, use of a 0.1% threshold yielded a final tree whose likelihood score was worse than that of the tree found using higher threshold values. This result is not surprising given that the differences in approximate versus exact likelihood scores for the “short” tree set ranged from 0.283% to 0.488% (Table 2). In practice, a user can establish a reasonable threshold value by evaluating the approximate and exact likelihood scores for a selection of “good” trees (e.g., the best 100 trees under the parsimony criterion); this information is available from the “iteration log” output of PAUP*.

A final strategy is worth mentioning, although it has not been fully evaluated.
This method involves a “fast” heuristic search using approximate likelihood scores, saving the $n$ best trees encountered during this search (with, say, $n = 100$). The “exact” likelihoods of each of these $n$ trees are then calculated, and the best tree is selected according to these scores. Preliminary results indicate that this method is effective in finding optimal trees, but that it is not necessarily faster than the threshold approach already described. The problem is that for many data sets, although it takes less time to evaluate the likelihood of each tree tested during the search, more trees are calculated.
are evaluated because the search does not converge upon the best tree as quickly using the approximate scores. Of course, there are still other ways in which the speed of maximum-likelihood searches might be improved. One obvious approach is simply to begin iteration from arbitrary branch lengths but exert less effort in estimating the optimal branch lengths for any topology being evaluated, either by limiting the number of passes over the tree to a very small number or by relaxing the convergence criterion so that a smaller decrease in likelihood will terminate the iteration process. Again, preliminary results indicate that this approach is less effective than the threshold strategy, but further evaluation is needed.

Because the approximate likelihood does not always rank the best trees in the same order as the exact likelihood (Fig. 3), optimal trees should not be selected solely by the approximate likelihood if the goal is to identify the tree(s) of highest likelihood. However, in some situations, the tree of highest approximate likelihood may have a better chance of being the correct tree than an optimal tree under a less desirable criterion.

PAUP* and fastDNAml find identical likelihood scores in the great majority of cases (tested using the primate mtDNA set only), although PAUP* is somewhat faster (Fig. 5). In the few cases where the programs estimated different likelihoods, these differences were small, usually involving trees of very low likelihood. Such trees are prone to have multiple likelihood peaks (in the sense of Steel, 1994; Rogers and Swofford, unpubl. data), resulting in the two programs occasionally climbing different peaks for the same tree.

The accuracy of the estimates of branch length from the approximate likelihood method is also of interest. For the hypothetical data and tree of Figures 1a and b the Jukes–Cantor branch lengths from the approximate method are 0.0203, 0.0625, 0.0000, 0.0846, and 0.0625 for branches 1, 2, 3, 4, and 5, respectively. The corresponding “exact” maximum-likelihood branch lengths are 0.0312, 0.0577, 0.0050, 0.0823, and 0.0550. In this case the approximate method underestimated the shorter branch lengths and overestimated the longer lengths, as might be expected from an algorithm that forces as many changes as possible onto longer branches. However, this effect is not general. In analyses of real data sets, for example, the “short” trees for the pocket gopher cytochrome-\(b\) data (Table 2, Figs. 2–5), longer branch lengths are often underestimated.

ACKNOWLEDGMENTS

We thank Paul Lewis and John Huelsenbeck for comments on the manuscript and for helpful discussion. Joe Felsenstein and an anonymous referee reviewed the paper thoroughly and made several very helpful suggestions.

REFERENCES


Helm-Bychowski, K., and J. Cracraft. 1993. Recov-
ering phylogenetic signal from DNA sequences: Relationships within the corvine assemblage (class Aves) as inferred from complete sequences of the mitochondrial cytochrome-b gene. Mol. Biol. Evol. 10:1196–1214.

Hillis, D. M., J. P. Huelsenbeck, and C. W. Cun-


Received 13 November 1996; accepted 22 March 1997

Associate Editor: D. Cannatella