A Fast Algorithm for Joint Reconstruction of Ancestral Amino Acid Sequences

Tal Pupko, Itsik Pe'er,† Ron Shamir,† and Dan Graur**

*Department of Zoology, George S. Wise Faculty of Life Sciences, and †Department of Computer Science, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Ramat Aviv, Israel

A dynamic programming algorithm is developed for maximum-likelihood reconstruction of the set of all ancestral amino acid sequences in a phylogenetic tree. To date, exhaustive algorithms that find the most likely set of ancestral states (joint reconstruction) have running times that scale exponentially with the number of sequences and are thus limited to very few taxa. The time requirement of our new algorithm scales linearly with the number of sequences and is therefore applicable to practically any number of taxa. A detailed description of the new algorithm and an example of its application to cytochrome *b* sequences are provided.

Introduction

By using extant sequences and the phylogenetic relationships among them, it is possible to infer the most plausible ancestral sequences from which they have been derived. Ancestral reconstruction has been applied in several contexts. For instance, the parsimony method (Fitch 1971; Swofford 1993) has been used to infer ancestral amino acid sequences of lysozyme. The inferred ancestral proteins were then synthesized in vitro and used in studies of adaptive evolution (Malcolm et al. 1990; Stewart 1995). The parsimony method, however, has many faults. For instance, it is inherently biased toward overestimating the number of ''common to rare'' changes (Walker 1998). Furthermore, this method does not supply the means for discriminating among equally parsimonious reconstructions (Yang, Kumar, and Nei 1995).

Maximum likelihood (ML) is a general estimation paradigm which has been widely utilized in evolutionary studies, overcoming many shortcomings of parsimony (Felsenstein 1981; Kishino, Miyata, and Hasegawa 1990). ML-based methods for inference of ancestral sequences were devised by Yang, Kumar, and Nei (1995) and Koshi and Goldstein (1996). A widely used variant of ML (the Bayesian approach) finds the most probable parameter set given the data. Applying this ML variant to ancestral-sequence reconstruction, one maximizes P(ancient amino acid sequences | contemporary sequences). Indeed, the method developed by Yang, Kumar, and Nei (1995) is Bayesian. By using the most likely set of sequences at all the internal nodes to evaluate the number of synonymous versus nonsynonymous substitutions along branches, Zhang, Rosenberg, and Nei (1998) inferred positive Darwinian selection after gene duplication in primate ribonuclease genes.

Yang (1995) distinguished between two variants of ancestral ML reconstruction, termed ''joint'' and ''marginal.'' To illustrate the difference between the two methods, let us consider the multifurcated tree in figure 1. Suppose character state *A* can change to either *B* or *C*, and then to *D* . . . *I*, according to the probabilities listed in figure 1. If we are interested in the most likely pathway, then clearly the answer is set{*A, C, I*}, i.e., $A \rightarrow C \rightarrow I$, which has a probability of $0.45 \times 0.9 = 0.405$. If, on the other hand, we are interested in the most likely character state after one step, then *B* is the winner (although *B* does not even feature in the most likely set). As far as ancestral-sequence inference is concerned, we have an analogous situation. We may be interested either in a the set of all the hypothetical taxonomic unit (HTU) sequences (joint reconstruction) or in a specific HTU whose sequence we would like to estimate (marginal reconstruction). As our examples demonstrate, the results are not necessarily the same under the two methods of ML reconstruction.

The two ML reconstruction methods have been implemented in the phylogenetic software packages PAML (Yang 1995) and ANCESTOR (Zhang and Nei 1997). However, joint reconstruction is very inefficiently implemented in these applications, which employ inherently slow algorithms. These algorithms evaluate many possible reconstructions one by one, and there are c^n such reconstructions per site, where c is the number of character states observed in the site and n is the number of reconstructed ancestors. Thus, the running time of existing reconstruction programs is exponential, i.e., such programs are inapplicable when the number of taxa is large.

Koshi and Goldstein (1996) have developed a fast dynamic programming algorithm for marginal reconstruction, whose variants are implemented in existing software (Yang 1995; Zhang and Nei 1997). To date, no fast algorithm exists for joint reconstruction.

Here, we provide a new efficient algorithm for joint ML ancestral reconstruction. The running time of our algorithm scales linearly with the number of sequences and thus can be applied to a practically unlimited number of sequences. The new algorithm is based on the dynamic programming scheme (for a general description of this scheme, see, e.g., Cormen, Leiserson, and Rivest 1990). Finally, we compare the performance of our method to those of extant algorithms by applying it to a data set of cytochrome *b* sequences.

Key words: ancestral sequences, fast algorithm, joint reconstruction, maximum likelihood, dynamic programming, molecular evolution.

Address for correspondence and reprints: Dan Graur, Department of Zoology, George S. Wise Faculty of Life Sciences, Tel Aviv University, Ramat Aviv 69978, Israel. E-mail: graur@post.tau.ac.il.

Mol. Biol. Evol. 17(6):890–896. 2000

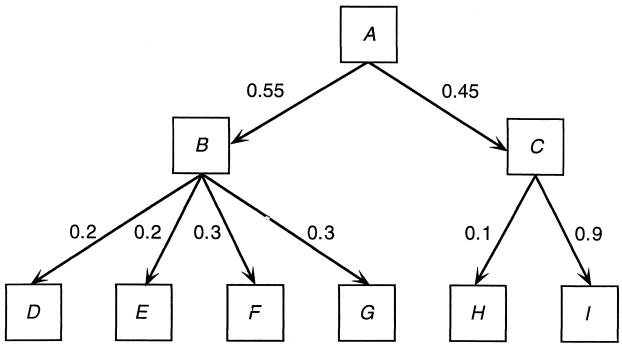

FIG. 1.—A numerical illustration of the difference between the joint maximum-likelihood and the marginal maximum-likelihood methods. The probability of change from any character state to any other is shown on the arrows. The most likely joint route is $A \rightarrow C \rightarrow I$, with a probability of $0.45 \times 0.9 = 0.405$. On the other hand, the most likely first step is $A \rightarrow B$, with a probability of 0.55 as compared with a probability of only 0.45 for $A \rightarrow C$, which is part of the most likely joint pathway.

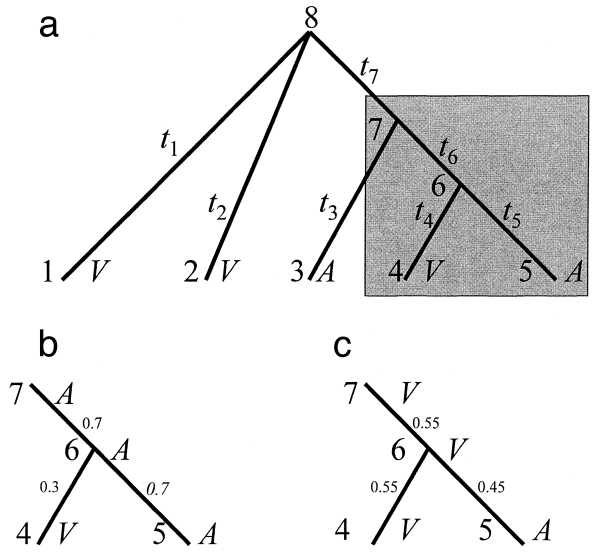

FIG. 2.—*a*, Unrooted phylogenetic tree for five taxa. Letters denote amino acids: *A*, alanine; *V*, valine. The digit to the left of each node is an arbitrary node label. t_x is the branch connecting node *x* and its father. (We use the symbol t_x to represent both the branch and its length.) The subtree supported by the branch t_6 appears in a shaded rectangle. *b*, Under the assumption that the character state at node 7 is *A*, there are two possible reconstructions for node 6. If we assign *A* to node 6, then the likelihood of the shaded subtree is $0.7 \times 0.7 \times 0.3 = 0.147$. (Replacement probabilities along the branches are taken from table 1.) On the other hand, if *V* is assigned to node 6, then the likelihood of the shaded subtree is $0.3 \times 0.55 \times 0.45 = 0.074$. Hence, given that node 7 is assigned *A*, the best reconstruction for node 6 is *A*. Thus, $L_6(A) = 0.147$ and $C_6(A) = A$, where *L* and *C* denote likelihood and most likely character state, respectively. *c*, A similar computation of $L_6(V)$ and $C_6(V)$.

Materials and Methods

ML Ancestral Reconstruction

Following Yang, Kumar, and Nei (1995), we assumed that different sites evolve independently. We therefore restricted the subsequent description to a single site. We further assumed that sequence evolution is governed by a probabilistic reversible model. As far as amino acid sequences are concerned, this model is described by a 20×20 matrix M, indicating the relative replacement rates of amino acids, and a vector $(P_A, \ldots, P_Y)$ of amino acid frequencies. For each branch of length t, the $i \rightarrow j$ replacement probability, denoted as $P_{ij}(t)$, can be calculated from the eigenvalue decomposition of M (Kishino, Miyata, and Hasegawa 1990). The unrooted tree topology and branch lengths are assumed to be known a priori.

Consider, for example, an unrooted tree with five operational taxonomic units (OTUs), as in figure 2. At a given sequence position, there are 20^3 possible reconstructions of the amino acids at the three internal nodes (*A* in node 6, *A* in node 7, *A* in node 8; or *A* in node 6, *A* in node 7, *C* in node 8; . . . or *Y* in node 6, *Y* in node 7, *Y* in node 8). The aim of joint ML is to identify the triplet v, maximizing $P(v \mid \text{data})$. That is, we want to find, from among all possible triplets, the one that maximizes

$$\frac{P(\text{data} \mid v) \times P(v)}{P(\text{data})}. \quad (1)$$

Since $P(\text{data})$ is the same for all triplets, it suffices to maximize $P(\text{data} \mid v) \times P(v)$. For the tree in figure 2, we solve

$$\max_{(y_6,y_7,y_8)} [P_{y_8}P_{y_8y_1}(t_1)P_{y_8y_2}(t_2)P_{y_8y_7}(t_7)P_{y_7y_3}(t_3)P_{y_7y_6}(t_6) \times P_{y_6y_4}(t_4)P_{y_6y_5}(t_5)]. \quad (2)$$

The choice of node 8 as the root is arbitrary, because the model assumed is time-reversible (Felsenstein 1981; Yang, Kumar, and Nei 1995).

Complexity

The complexity issue is central to ancestral reconstruction. The solution sought in equation (2) is the maximum over all possible triplets. However, for larger trees, say, with h HTUs, one needs to maximize over the set of all possible combinations of h ancestral character states. As explained in the introduction, this set is very large, including 20^h such combinations. Even if one considers only the c character states observed in the site, there are c^h combinations of such states that are likely to appear (Zhang and Nei 1997). Naive implementation, examining each such combination separately, is impractical for all but very modest values of h.

In the following, we present a fast new algorithm for ancestral reconstruction. This dynamic programming algorithm guarantees finding the set of sequences, one sequence per node, most likely to have been the progenitors of the extant sequences. The complexity of the algorithm is linear. That is, its running time per site is proportional to the number of internal nodes, and its efficiency enables its employment for any number of OTUs. Note that our algorithm maximizes the likelihood over all 20^h possible combinations. For very small numbers of OTUs, this implies longer running times than those for existing programs, which check only c^h combinations. However, since our algorithm is efficient, it does not require more than a few seconds (see *Results*).

Table 1
Replacement Probabilities for the Toy Model (See Text)

	To Alanine	To Valine
From alanine	0.70	0.30
From valine	0.45	0.55

NOTE.—We assume that there are only two possible character states: alanine and valine. We also assume that the frequencies of alanine and valine are 0.6 and 0.4, respectively.

Terminology

We root the tree at an arbitrary internal node. If node x is the direct descendant of node y, we say that y is the father of node x, and x is the son of node y. Thus, each nonroot node has a father. Each internal node has two sons, except for the root, which has three sons. OTUs have no sons. Also, each tree branch supports a subtree, which includes the father and son that it connects, together with all of the descendants of the son. For a demonstration of the terminology, consider the tree in figure 2. Node 8 is the arbitrary root. Node 7 is the father of nodes 3 and 6. The branch connecting nodes 7 and 6 supports the subtree in the shaded region, including nodes 7, 6, 5, and 4.

A Linear-Time Algorithm for Joint Ancestral Reconstruction

Only alanine (A) and valine (V) are observed at OTUs of the tree in figure 2. Thus, the chances of any other amino acid occurring at internal nodes are quite negligible. For the sake of clarity and compactness, in this section we shall only deal with the values of $P_{ij}(t)$ for i and j which are either alanine or valine. For simplicity, we further assume that all the branches of the phylogenetic tree in figure 2 are of the same length t and that the replacement probabilities $P_{ij}(t)$ for this value of t are given in table 1. Note, however, that this table is merely a toy model for the sole purpose of demonstrating the algorithm. In practice, we use $P_{ij}(t)$ values from matrices published in standard literature (Dayhoff, Schwartz, and Orcutt 1978; Jones, Taylor, and Thornton 1992; Adachi 1995) and use trees whose branch lengths have been estimated.

Like many phylogenetic algorithms, our algorithm first traverses the tree from the OTUs toward the root. Upon visiting a nonroot node x, we compute for each character state i a quantity $L_x(i)$ and a character state $C_x(i)$. $L_x(i)$ and $C_x(i)$ are to be interpreted as follows: If the father of node x is assigned character state i, then finding the best reconstruction of the subtree supported by the branch between x and its father is a separate problem from reconstructing the rest of the tree. $L_x(i)$ is the likelihood of the best reconstruction of this subtree on the condition that the father of node x is assigned character state i. $C_x(i)$ is the character state assigned to node x in this optimal conditional reconstruction.

These concepts are best understood with an example. Consider node $x = 6$ in the tree in figure 2. If one assumes its father, i.e., node 7, is assigned A, then the best reconstruction of the shaded subtree is given in figure 2b. In this case, $L_6(A) = 0.7 \times 0.7 \times 0.3 = 0.147$, and $C_6(A) = A$. By a similar argument, $L_6(V) = 0.55 \times 0.55 \times 0.45 = 0.136$, and $C_6(V) = V$, as demonstrated in the legend of figure 2.

We now give a full description of the algorithm.

1. For each OTU y perform the following:
   1a. Let j be the amino acid at y. Set, for each amino acid i: $C_y(i) = j$. This implies that no matter what is the amino acid in the father of y, j is assigned to node y.
   1b. Set for each amino acid i: $L_y(i) = P_{ij}(t_y)$, where t_y is the branch length between y and its father.
2. Visit a nonroot internal node, z, which has not been visited yet, but both of whose sons, nodes x and y, have already been visited, i.e., $L_x(j)$, $C_x(j)$, $L_y(j)$, and $C_y(j)$ have already been defined for each j. Let t_z be the length of the branch connecting node z and its father. For each amino acid i, compute $L_z(i)$ and $C_z(i)$ according to the following formulae:
   2a. $L_z(i) = \max_j P_{ij}(t_z) \times L_x(j) \times L_y(j)$.
   2b. $C_z(i)$ = the value of j attaining the above maximum.
3. If all nonroot nodes have been visited, proceed to step 4. Otherwise, return to step 2.
4. Denote the three sons of the root by x, y, and z. For each amino acid k, compute the expression $P_k \times L_x(k) \times L_y(k) \times L_z(k)$. Reconstruct r by choosing the amino acid k maximizing this expression. The maximum value found is the likelihood of the best reconstruction.
5. Traverse the tree from the root in the direction of the OTUs, assigning to each node its most likely ancestral character as follows:
   5a. Visit an unreconstructed internal node x whose father y has already been reconstructed. Denote by i the reconstructed amino acid at node y.
   5b. Reconstruct node x by choosing $C_x(i)$.
   5c. Return to step 5a until all internal nodes have already been reconstructed.

To elucidate the algorithm, we now present an example of its application to the tree in figure 2. The computation of L_6 and C_6 has been presented above. The values of L_x and C_x, computed in steps 1 and 2 for all nonroot nodes x, appear in figure 3a. The expressions used to infer the amino acid at the root (step 4) are shaded in figure 3a. As can be seen, the most likely reconstruction assigns valine to the root. The most likely reconstruction of the whole tree appears in figure 3b.

Computer Program

A computer program that implements the algorithm presented above, called FastML, was written in C++ for PC architecture. It is available at http://kimura.tau.ac.il/~tal. The program allows the user to obtain both joint and marginal reconstructions of amino acid sequences at all the internal nodes of a given phylogenetic

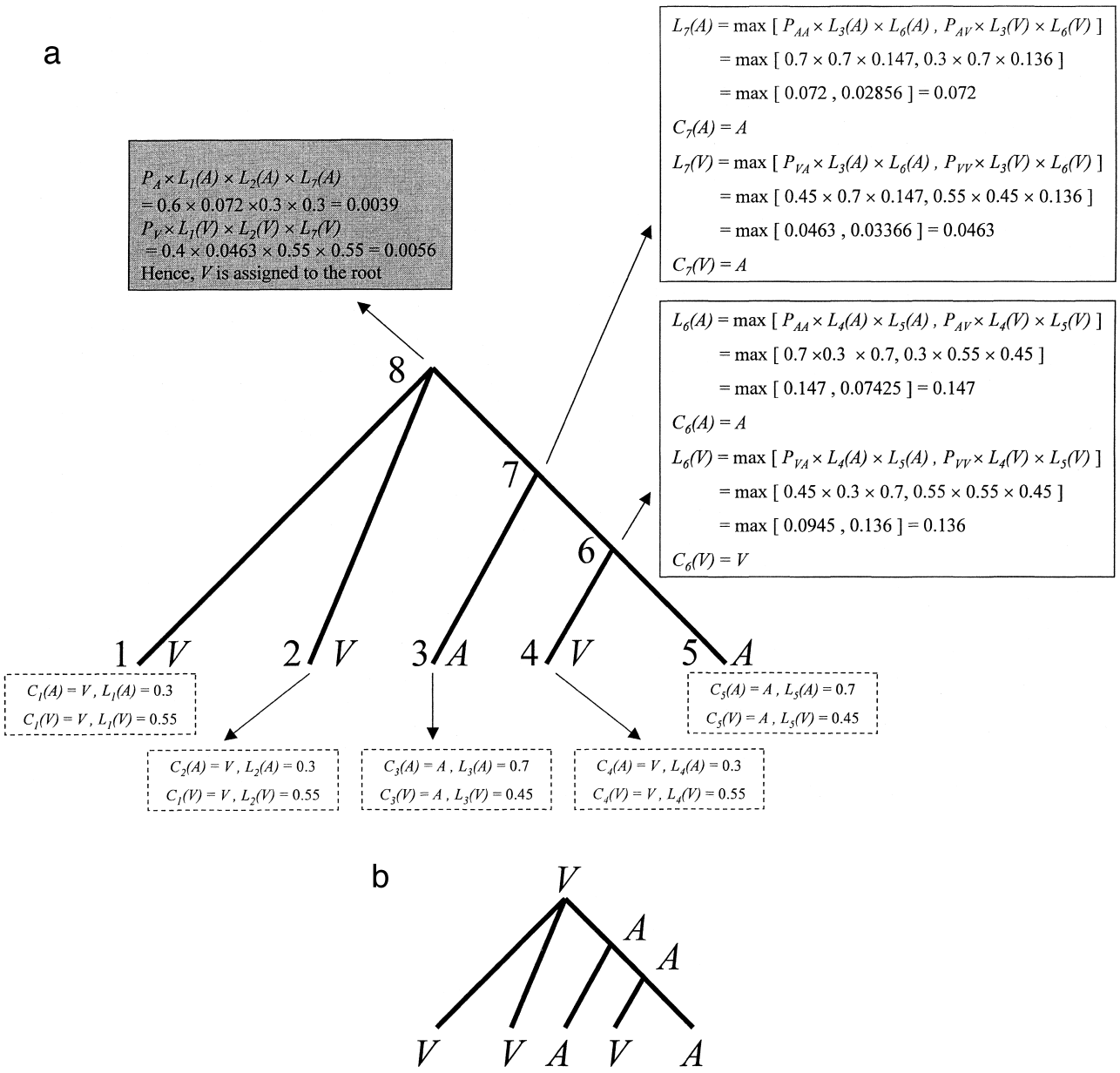

FIG. 3.—*a*, The bottom-up traversal of the algorithm (see *Materials and Methods*) is presented in dashed rectangles (step 1), solid rectangles (steps 2 and 3) and a shaded rectangle (step 4). *b*, The reconstructed ancestral character states (steps 4 and 5). After assigning *V* to the root, we assign the character states to all hypothetical taxonomic units. Since *V* is assigned to node 8 and $C_7(V) = A$, we assign *A* to node 7. Since *A* is assigned to node 7 and $C_6(A) = A$, we assign *A* to node 6.

tree. It also calculates the likelihoods of the reconstructed sequences and the posterior probability at each sequence position.

Empirical Example

Aligned cytochrome *b* amino acid sequences from a sample of mammals were taken from the data of Takezaki and Gojobori (1999). Phylogenetic trees with 4 to 24 sequences (from 21 taxa) were prepared by taking a subset from these sequences and using the neighbor-joining algorithm (Saitou and Nei 1987) to reconstruct the tree. For each such tree, we compared the running time required to estimate the ancestral sequences by the three programs: FastML, PAML, and ANCESTOR.

We also compared joint versus marginal reconstruction for cytochrome *b* sequences of the 21 taxa in the data set. The phylogenetic tree obtained for these sequences (fig. 5) was in agreement with the tree obtained by Takezaki and Gojobori (1999). Calculations of the replacement probabilities were done with the REV model (Adachi 1995).

Results

Complexity

The program ANCESTOR was unable to reconstruct the best joint ancestral sequences for more than nine amino acid sequences. In PAML, exact search for joint reconstruction is limited to six amino acid sequenc-

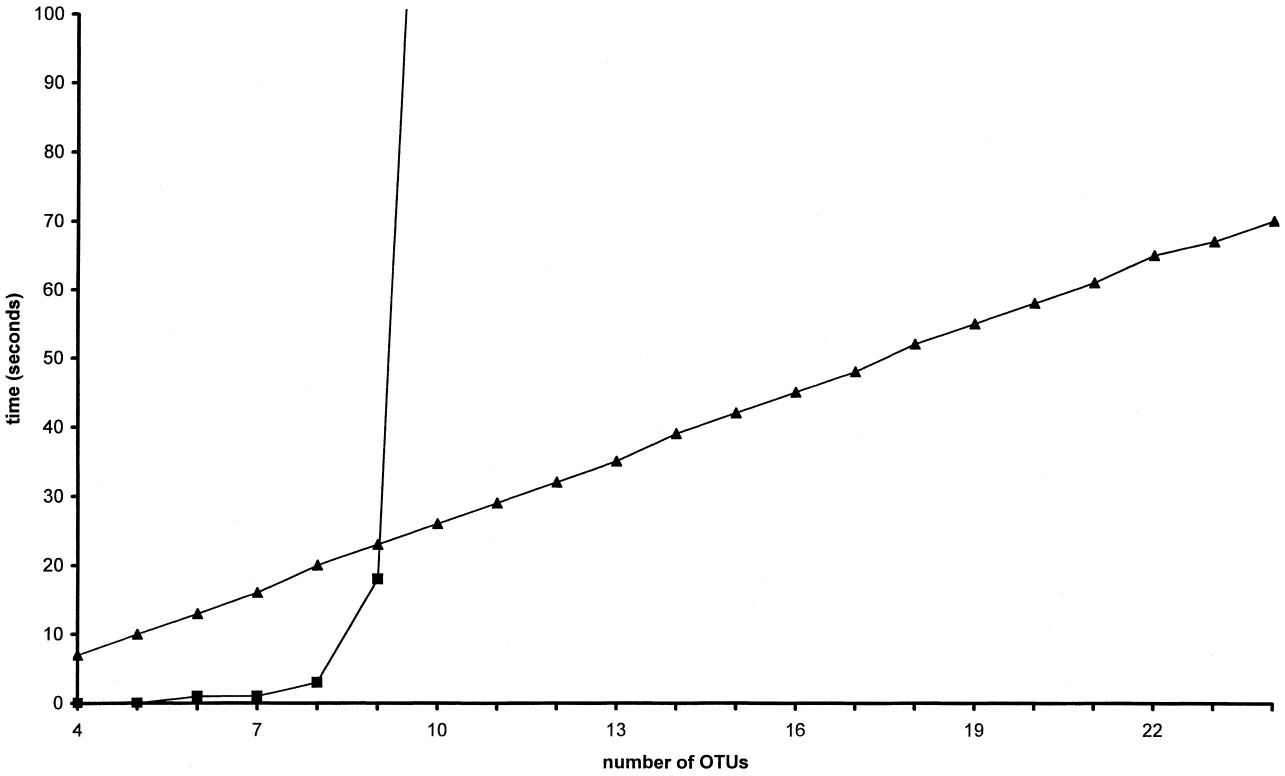

FIG. 4.—Comparison of the running time of FastML, a program implementing the new algorithm (triangles), versus that of the ANCESTOR program (squares). We were unable to find the exact reconstruction for 10 OTUs with the ANCESTOR program on our Pentium 450-MHz computer. PAML was not included in this chart, since for $n > 6$ sequences, it automatically employs a heuristic algorithm that does not necessarily produce the most likely set of ancestral sequences.

es. For a larger number of taxa, both the ANCESTOR and PAML programs resort to heuristic approaches, which are not guaranteed to find the optimal reconstruction. The running time of our program (FastML) versus the running time of ANCESTOR for the exact search is given in figure 4.

Table 2
Difference Between Joint Maximum-Likelihood (ML) Reconstruction and Marginal ML Reconstruction

Node	Position	Joint Reconstruction	Marginal Reconstruction
23	209	T	S
24	209	T	S
25	263	S	N
26	263	S	N
27	263	S	N
28	263	S	N
29	263	S	N
32	238	I	V
33	241	M	L
35	23	A	T
36	23	A	T
37	23	A	T
38	23	A	T

NOTE.—The numbers in the first column refer to the internal-node labels in figure 5. Amino acids are shown as one-letter abbreviations. For all other nodes and positions, the two methods yield the same ancestral amino acid reconstruction.

Joint Versus Marginal Reconstruction

Results of joint and marginal ML reconstruction of the 19 internal nodes for the tree in figure 5 are available at http://kimura.tau.ac.il/~tal. Differences between joint reconstruction and marginal reconstruction at each of the nodes are given in table 2. Thirteen differences were found at five positions, each in different HTUs. One such example, involving the inference of sequence position 241, is shown in figure 5.

Discussion

Previous joint reconstruction algorithms are of exponential time complexity, which severely limits their applicability to only a small number of sequences. Our algorithm overcomes this problem, reducing the complexity of the exhaustive search to linear time. Thus, the algorithm guarantees the identification of the most probable amino acid ancestral pathway of amino acid replacements even when the number of taxa is very large. Linear time algorithms are implemented in PAML and ANCESTOR only for marginal reconstruction. Nevertheless, marginal reconstruction optimizes a different criterion and can only be considered an approximation to joint reconstruction, as demonstrated in our results.

As shown in table 2, in all but five positions, there was no difference between joint and marginal reconstruction. However, depending on the intention of the

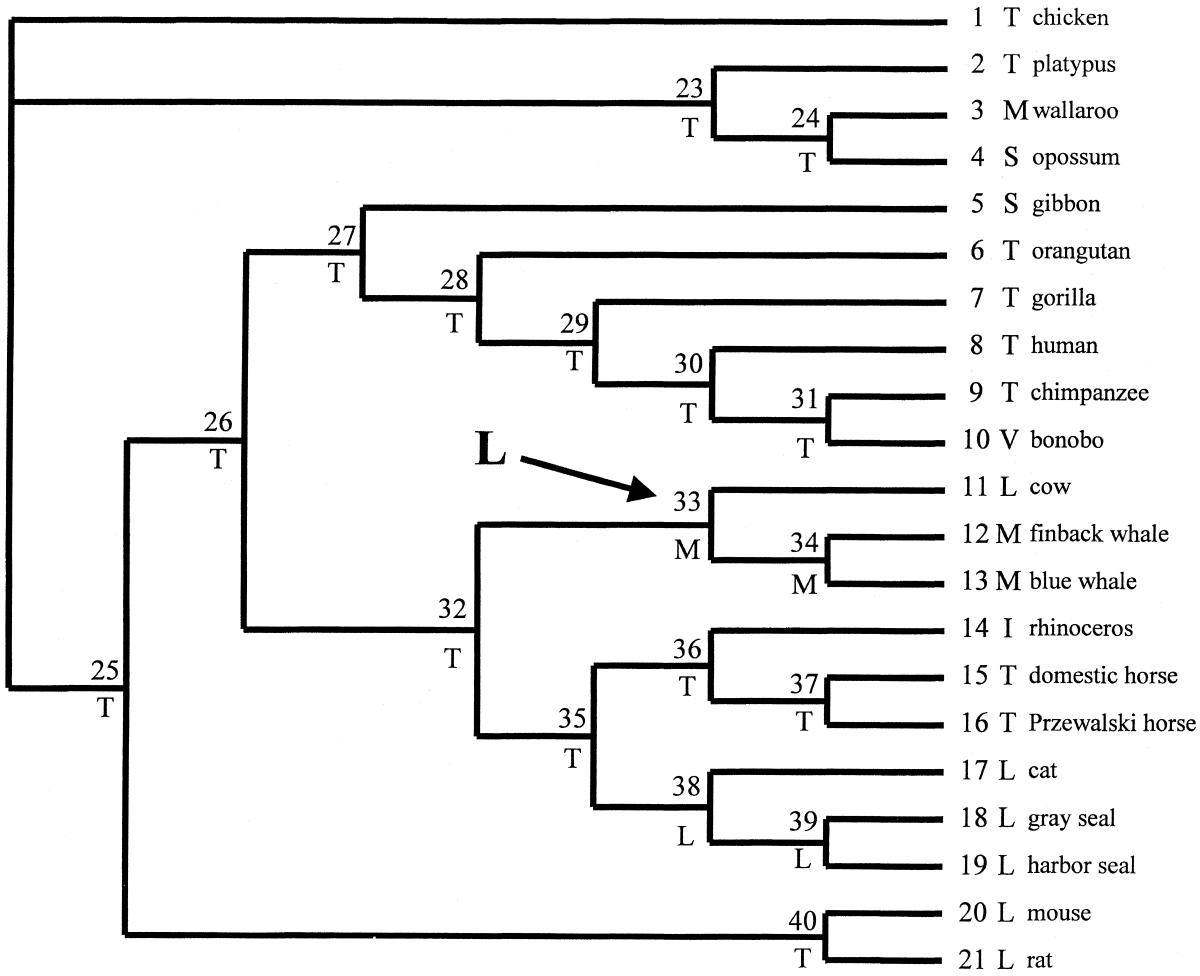

FIG. 5.—Marginal versus joint reconstruction at position 241. Marginal reconstruction of the ancestor of all Cetartiodactyla (node 33) predicts leucine. In contrast, when reconstructing the most likely set of ancestral amino acids in this position, methionine is assigned to this hypothetical taxonomic unit (arrow).

study, the differences may be important. For example, consider node 33 (fig. 5). If one seeks the most likely amino acid in this HTU, then it is leucine. However, when jointly reconstructing the whole tree, the most likely member of the set containing all the internal-node amino acid assignments is methionine. Deciding which is ''more correct'' depends on the question asked. For instance, if one wishes to count the number of threonine-to-methionine replacements over the entire tree, then the joint reconstruction should be used to obtain this number (2, in our case, on the branch connecting node 24 to node 3 and the branch connecting node 32 to node 33). However, if one wishes to synthesize the hypothetical cytochrome *b* sequence of the ancestor of Cetartiodactyla, then one should use the marginal reconstruction approach. We emphasize that both methods compute optimal reconstructions by using all of the available data. Discrepancies originate not from misuse of information, but from the difference in the nature of the probabilistic questions asked.

The rate of amino acid replacement is usually not constant among sites. Our algorithm finds the most likely ancestral sequence only for the case of constant rate only. There are as yet no programs for joint reconstruction that take rate variation among sites into account. In PAML, a marginal reconstruction of ancestral sequence that assumes gamma distribution among sites is available. We were unable to apply our linear algorithm to the gamma model because of the different expressions that have to be maximized.

The algorithm described above is presented in terms of amino acids and bifurcating trees. However, the algorithm can be easily adapted for nucleotide sequences and multifurcating trees.

Acknowledgments

This study was supported in part by a grant from the Israel Ministry of Science and a fellowship of the Clore Foundation. This study was also supported by the Magnet Da'at Consortium of the Israel Ministry of Industry and Trade and a grant from Tel Aviv University (689/96).

LITERATURE CITED

ADACHI, J. 1995. Modeling of molecular evolution and maximum likelihood inference of molecular phylogeny. Ph.D.

dissertation, Graduate University for Advanced Studies, Hayama, Japan.

CORMEN, T. H., C. E. LEISERSON, and R. L. RIVEST. 1990. Introduction to algorithms. MIT Press, Cambridge, Mass.

DAYHOFF, M. O., R. M. SCHWARTZ, and B. C. ORCUTT. 1978. A model of evolutionary change in proteins. Pp. 345–352 *in* M. O. DAYHOFF, ed. Atlas of protein sequences and structure. Vol. 5, Suppl. 3. National Biomedical Research Foundation, Washington, D.C.

FELSENSTEIN, J. 1981. Evolutionary trees from DNA sequences: a maximum likelihood approach. J. Mol. Evol. **17**:368–376.

FITCH, W. M. 1971. Toward defining the course of evolution: minimum change for a specific tree topology. Syst. Zool. **20**:406–416.

JONES, D. T., W. R. TAYLOR, and J. M. THORNTON. 1992. The rapid generation of mutation data matrices from protein sequences. Comput. Appl. Biosci. **8**:275–282.

KISHINO, H., T. MIYATA, and M. HASEGAWA. 1990. Maximum likelihood inference of protein phylogeny and the origin of chloroplasts. J. Mol. Evol. **31**:151–180.

KOSHI, J. M., and R. A. GOLDSTEIN. 1996. Probabilistic reconstruction of ancestral protein sequences. J. Mol. Evol. **42**: 313–320.

MALCOLM, B. A., K. P. WILSON, B. W. MATTHEWS, J. F. KIRSCH, and A. C. WILSON. 1990. Ancestral lysozymes reconstructed, neutrality tested, and thermostability linked to hydrocarbon packing. Nature **345**:86–89.

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

STEWART, C.-B. 1995. Active ancestral molecules. Nature **374**: 12–13.

SWOFFORD, D. L. 1993. PAUP: phylogenetic analysis using parsimony. Version 3.1. Illinois Natural History Survey, Champaign.

TAKEZAKI, N., and T. GOJOBORI. 1999. Correct and incorrect phylogenies obtained by the entire mitochondrial DNA sequences. Mol. Biol. Evol. **16**:590–601.

WALKER, A. E. 1998. Problems with parsimony in sequences of biased base composition. J. Mol. Evol. **47**:686–690.

YANG, Z. 1995. PAML: a phylogenetic analysis by maximum likelihood. Version 2.0e. Pennsylvania State University, University Park.

YANG, Z., S. KUMAR, and M. NEI. 1995. A new method of inference of ancestral nucleotide and amino acid sequences. Genetics **141**:1641–1650.

ZHANG, J., and M. NEI. 1997. Accuracies of ancestral amino acid sequences inferred by the parsimony, likelihood, and distance methods. J. Mol. Evol. **44**(Suppl. 1):s139–s146.

ZHANG, J., H. F. ROSENBERG, and M. NEI. 1998. Positive Darwinian selection after gene duplication in primate ribonuclease genes. Proc. Natl. Acad. Sci. USA **95**:3708–3713.

SHOZO YOKOYAMA, reviewing editor

Accepted February 7, 2000