



Computational Advances in Maximum Likelihood Methods for Molecular Phylogeny

Eric E. Schadt, Janet S. Sinsheimer and Kenneth Lange

Genome Res. 1998 8: 222-233

Access the most recent version at doi:[10.1101/gr.8.3.222](https://doi.org/10.1101/gr.8.3.222)

References This article cites 21 articles, 3 of which can be accessed free at:
<http://genome.cshlp.org/content/8/3/222.full.html#ref-list-1>

License

Email Alerting Service Receive free email alerts when new articles cite this article - sign up in the box at the top right corner of the article or [click here](#).



To subscribe to *Genome Research* go to:
<https://genome.cshlp.org/subscriptions>

Cold Spring Harbor Laboratory Press

RESEARCH

Computational Advances in Maximum Likelihood Methods for Molecular Phylogeny

Eric E. Schadt,¹ Janet S. Sinsheimer,² and Kenneth Lange^{3,4}

Departments of ^{1,2}Biomathematics and ²Biostatistics, The University of California, Los Angeles, California 90095 USA; ³Departments of Biostatistics and Mathematics, The University of Michigan, Ann Arbor, Michigan 48109-2029 USA

We have developed a generalization of Kimura's Markov chain model for base substitution at a single nucleotide site. This generalized model incorporates more flexible transition rates and consequently allows irreversible as well as reversible chains. Because the model embodies just the right amount of symmetry, it permits explicit calculation of finite-time transition probabilities and equilibrium distributions. The model also meshes well with maximum likelihood methods for phylogenetic analysis. Quick calculation of likelihoods and their derivatives can be carried out by adapting Baum's forward and backward algorithms from the theory of hidden Markov chains. Analysis of HIV sequence data illustrates the speed of the algorithms on trees with many contemporary taxa. Analysis of some of Lake's data on the origin of the eukaryotic nucleus contrasts the reversible and irreversible versions of the model.

The object of phylogenetic analysis is to infer the correct evolutionary relationships among three or more contemporary species from amino acid or nucleotide sequence data on a representative member of each species. As more nucleotide sequence data accumulate from a multitude of contemporary species and as computers become ever faster, phylogenetic methods should evolve accordingly. In our opinion, properly implemented maximum likelihood methods are the best vehicles for statistical inference. First, maximum likelihood methods use all of the available data. Second, they can incorporate realistic evolutionary models. (They can also incorporate naive models.) Third, they permit estimation of parameters and their associated standard errors. Fourth, and finally, they allow comparison of different evolutionary trees through their maximum likelihood statistics. Competing methods of analysis, such as maximum parsimony, involve much less computation but neglect relevant portions of the data. They also ordinarily do not yield reliable parameter estimates.

In this paper we present a generalization of Kimura's model for base substitution at a single nucleotide site. Our generalization strikes a good compromise between parameter richness and analytic tractability. By including just enough symmetry, we can extend Kimura's arguments for deducing

finite-time transition probabilities from his two-parameter model to our eight-parameter model. We also describe an algorithm that simultaneously calculates the likelihood and its derivatives quickly. These calculations can be implemented on a parallel computer for extremely fast analysis of many evolutionary trees. We demonstrate the feasibility of the model and the new likelihood algorithms by analyzing HIV sequence data and RNA sequence data bearing on the origin of the eukaryotic nucleus.

The model adopted in this paper specifies the expected branch length separating a mother and a daughter node of an evolutionary tree as the average number of substitutions per nucleotide position (site) between the nodes. This expectation is a function of the conditional probability $p_{ij}(t) = \Pr(Z_t = j | Z_0 = i)$ that the daughter node occupies state j at time t given that the mother node occupies state i at time 0. Here i and j are any of the four possible nucleotides, namely, the purines A (adenine) and G (guanine) and the pyrimidines C (cytosine) and T (thymine). The probabilities $p_{ij}(t)$ are entries of the finite-time transition matrix $P(t)$ characterizing the Markov chain Z_t of successive nucleotides occurring along the branch at the given site. Under the Markov chain model, the probability of the observed nucleotides at the site depends on (1) the rooted tree connecting the contemporary taxa, (2) the distribution of nucleotides at the root, and (3) the matrix function $P(t)$. Phylogenetic reconstruction using maximum likelihood (Felsenstein 1981)

⁴Corresponding author.
E-MAIL klange@umich.edu; FAX (650) 855-6115.

attempts to infer the correct tree by maximizing the likelihood of the observed data with respect to entries of the various matrices $P(t)$ or parameters determining these entries. The tree yielding the largest maximum likelihood is declared to be the best tree.

For the purposes of this paper, we make the simplifying assumption that the nucleotide patterns at different sites are independently and identically distributed. Many models for the 4×4 matrix $P(t)$ have been proposed. Barry and Hartigan (1987) place no restrictions on $P(t)$. Because all row sums of $P(t)$ equal 1, this entails 12 parameters per branch. In practice, this many parameters can not be reliably estimated, even though, in some cases, the tree topology and branch lengths can be. Beginning with Kimura (1980) and Felsenstein (1981), various researchers have constructed models that capture the changes occurring at each site as a continuous-time Markov chain involving the four states A , G , C , and T . These time-homogeneous models are nicely summarized by Yang (1994). If in this context ω_{ij} is the infinitesimal transition rate of moving from state i to state $j \neq i$ and if $\omega_{ii} = -\sum_{j \neq i} \omega_{ij}$, then $P(t)$ can be expressed as the matrix exponential

$$P(t) = e^{t\Omega} = \sum_{k=0}^{\infty} \frac{t^k}{k!} \Omega^k \quad (1)$$

of the infinitesimal generator (or infinitesimal transition matrix) $\Omega = (\omega_{ij})$. Formula 1 permits, in principle, estimation of the off-diagonal entries of Ω and a branch time t_b for each branch b of an evolutionary tree. Because the branch times are confounded with the infinitesimal transition rates, it is necessary to set one arbitrary branch time equal to 1 and to determine all other branch times relative to this reference branch time.

In practice, numerical evaluation of matrix exponential 1 is a major computational bottleneck hindering maximum likelihood estimation (Yang 1994). Thus, models such as Kimura's (1980) that yield analytic expressions for the entries of $P(t)$ are extremely valuable. Kimura's highly symmetric two-parameter model incorporates a common infinitesimal rate α for all evolutionary transitions (purine to purine or pyrimidine to pyrimidine) and a common infinitesimal rate λ for all evolutionary transversions (purine to pyrimidine or pyrimidine to purine). the underlying Markov chain along any branch is reversible.

The Generalized Kimura Model

The infinitesimal generator under our generalization of Kimura's model is

MAXIMUM LIKELIHOOD IN MOLECULAR PHYLOGENY

$$\Lambda = \begin{pmatrix} A & A & G & C & T \\ G & -(\alpha + \gamma + \lambda) & \alpha & \gamma & \lambda \\ C & \epsilon & -(\epsilon + \gamma + \lambda) & \gamma & \lambda \\ T & \delta & \kappa & -(\delta + \kappa + \beta) & \beta \\ & \delta & \kappa & \sigma & -(\delta + \kappa + \sigma) \end{pmatrix} \quad (2)$$

This model obviously makes fewer biological assumptions than Kimura's model. In essence, we impose the constraint that the infinitesimal rate for a transversion depends only on the destination state and not on the initial state. In contrast, Lake's model (Lake 1987) of evolutionary parsimony imposes the opposite constraint that the infinitesimal rate for a transversion depends only on the initial state and not on the destination state. Oddly enough, Lake's model does not lead to analytic expressions for the entries of $P(t)$, whereas ours does.

Kimura's two-parameter model is a special case of the model determined by Equation 2. In view of Kolmogorov's circulation criterion (Kelly 1979), it is easy to verify that Λ in Equation 2 leads to a reversible Markov chain if and only if

$$\beta\gamma = \lambda\sigma, \quad \alpha\delta = \epsilon\kappa \quad (3)$$

When these reversibility conditions hold, Λ parameterizes the special case of Tavaré's (1986) general reversible model discussed by Tamura and Nei (1993).

Other special cases of the infinitesimal generator Λ appear in the dnaml program of Felsenstein's computer package PHYLIP (Kishino and Hasegawa 1989; Felsenstein and Churchill 1996) and in the nucml program of the MOLPHY package (Hasegawa et al. 1985; Adachi et al. 1996). PHYLIP depends on two parameters k and u and the root probabilities $(\pi_A, \pi_G, \pi_C, \pi_T)$. These five independent parameters define the entries of Λ as follows:

$$\begin{aligned} \alpha &= \left(\frac{k}{\pi_A + \pi_G} + 1 \right) \pi_G u \\ \beta &= \left(\frac{k}{\pi_C + \pi_T} + 1 \right) \pi_T u \\ \epsilon &= \left(\frac{k}{\pi_A + \pi_G} + 1 \right) \pi_A u \\ \sigma &= \left(\frac{k}{\pi_C + \pi_T} + 1 \right) \pi_C u \end{aligned} \quad (4)$$

$$(\gamma, \lambda, \delta, \kappa) = u(\pi_C, \pi_T, \pi_A, \pi_G)$$

SCHADT ET AL.

Equivalently,

$$u = \delta + \kappa + \lambda + \gamma$$

$$k = \frac{(\alpha - \kappa) + (\epsilon - \delta) + (\beta - \lambda) + (\sigma - \gamma)}{2(\delta + \kappa + \gamma + \lambda)} \quad (5)$$

$$(\pi_A, \pi_C, \pi_G, \pi_T) = u^{-1}(\delta, \kappa, \gamma, \lambda)$$

In practice, PHYLIP either requires the user to specify the components of the stationary distribution π or estimates these by the average nucleotide frequencies observed in the existing organisms. The two remaining parameters u and k are subject to the constraints $u > 0$, $k + \pi_A + \pi_G > 0$, and $k + \pi_C + \pi_T > 0$. Note that the PHYLIP model satisfies the reversibility criteria (Equation 3).

If branch lengths are unconstrained and equilibrium has been reached, then phylogenies with underlying reversible Markov chains can not be rooted (Felsenstein 1981). This pulley principle of Felsenstein fails if any of the three necessary conditions mentioned are violated. Thus, in principle, methods based on the irreversible matrix Λ can single out a best rooted tree when sufficient data are available. We will comment further on this point in our application to Lake's data.

In calculating the finite-time transition matrix $P(t)$ for the infinitesimal generator Λ , we pattern our arguments after Kimura (1980), who exploits symmetry and derives two coupled linear differential equations for the entries of $P(t)$. In our calculations, it is convenient to let Y denote either pyrimidine and R either purine. We define $q_{AY}(t)$ to be the probability the chain is in either of the two pyrimidines C or T at time t given it starts in A at time 0. We likewise define $q_{YA}(t)$ to be the probability the chain is in A at time t given it starts in either pyrimidine at time 0 and $q_{YY}(t)$ to be the probability the chain is in either pyrimidine at time t given it started in either pyrimidine at time 0. These probabilities all make sense owing to the symmetry assumptions incorporated in Λ . Similar probabilities can be defined for the purines instead of the pyrimidines.

The probability $q_{AY}(t+h)$ satisfies the Chapman-Kolmogorov relation

$$q_{AY}(t+h) = q_{AY}(t)q_{YY}(h) + \sum_{k \in Y} p_{Ak}(t)q_{kY}(h)$$

$$= q_{AY}(t)(1 - q_{YA}(h) - q_{YG}(h))$$

$$+ p_{AA}(t)q_{AY}(h) + p_{AG}(t)q_{GY}(h) \quad (6)$$

Here $p_{AA}(t)$ and $p_{AG}(t)$ are entries of the matrix $P(t)$. For small h , equation 6 becomes

$$q_{AY}(t+h) = q_{AY}(t)(1 - \delta h - \kappa h) + p_{AA}(t)(\gamma + \lambda)h$$

$$+ p_{AG}(t)(\gamma + \lambda)h + o(h)$$

$$= q_{AY}(t) + (\lambda + \gamma)h - (\delta + \kappa + \lambda + \gamma)q_{AY}(t)h + o(h)$$

Rearranging terms and taking the limit as $h \rightarrow 0$ in the resulting difference quotient yields the forward differential equation

$$q'_{AY}(t) = c_2 - c_1 q_{AY}(t), \quad (7)$$

where

$$c_1 = \delta + \kappa + \gamma + \lambda$$

$$c_2 = \gamma + \lambda$$

In view of the initial condition $q_{AY}(0) = 0$, equation (7) has solution

$$q_{AY}(t) = \frac{c_2}{c_1} (1 - e^{-c_1 t}) \quad (8)$$

We next solve for $p_{AG}(t)$ by writing the forward approximation

$$p_{AG}(t+h) = p_{AG}(t)(1 - \epsilon h - \gamma h - \lambda h) + q_{AY}(t)\kappa h$$

$$+ [1 - q_{AY}(t) - p_{AG}(t)]\alpha h + o(h)$$

Repetition of our previous arguments then gives the forward differential equation

$$p'_{AG}(t) = -c_3 p_{AG}(t) + c_4 q_{AY}(t) + \alpha \quad (9)$$

where

$$c_3 = \alpha + \gamma + \epsilon + \lambda$$

$$c_4 = \kappa - \alpha$$

Substituting the solution (Equation 8) for $q_{AY}(t)$ in Equation 9 and noting the initial condition $p_{AG}(0) = 0$, we find after straightforward calculus that

$$p_{AG}(t) = \frac{c_2 c_4 + \alpha c_1}{c_1 c_3} - \frac{c_2 c_4}{c_1 (c_3 - c_1)} e^{-c_1 t}$$

$$+ \frac{c_2 c_4 - \alpha (c_3 - c_1)}{c_3 (c_3 - c_1)} e^{-c_3 t}$$

From the identity $p_{AG}(t) + p_{AA}(t) + q_{AY}(t) = 1$, we finally harvest the entry $p_{AA}(t)$ of $P(t)$.

In summary, we have calculated the entries $p_{AG}(t)$ and $p_{AA}(t)$ and the sum $p_{AY}(t) = p_{AC}(t) + p_{AT}(t)$ from the top row $[p_{AA}(t), p_{AG}(t), p_{AC}(t), p_{AT}(t)]$ of $P(t)$. To fill out the top row, we briefly indicate how to calculate similar expressions for $q_{RR}(t)$ and $q_{RC}(t) = p_{AC}(t)$, where R stands for either purine. The forward differential equations for $q_{RR}(t)$ and for $q_{RC}(t)$ turn out to be

$$q'_{RR}(t) = -c_1 q_{RR}(t) + c_5 \quad (10)$$

and

$$q'_{RC}(t) = -c_6 q_{RC}(t) + c_7 q_{RR}(t) + \sigma \quad (11)$$

where c_1 is defined as above and

$$\begin{aligned}c_5 &= \delta + \kappa \\c_6 &= \delta + \kappa + \sigma + \beta \\c_7 &= \gamma - \sigma\end{aligned}$$

The solution of Equation 10 satisfying the initial condition $q_{RR}(0) = 1$ is

$$q_{RR}(t) = \frac{c_5 + (c_1 - c_5)e^{-c_1 t}}{c_1} \quad (12)$$

and the solution of Equation 11 satisfying the initial condition $q_{RC}(0) = 0$ is

$$\begin{aligned}q_{RC}(t) &= \frac{c_5 c_7 + \sigma c_1}{c_1 c_6} + \frac{(c_1 - c_5) c_7}{c_1 (c_6 - c_1)} e^{-c_1 t} \\&\quad - \frac{c_7 (c_6 - c_5) + \sigma (c_6 - c_1)}{c_6 (c_6 - c_1)} e^{-c_6 t}\end{aligned} \quad (13)$$

This completes the top row of $P(t)$.

The remaining rows of $P(t)$ can be similarly calculated and are provided in the Appendix. In the limit as $t \rightarrow \infty$, the top row of $P(t)$, along with all other rows, tends to the equilibrium distribution π with components

$$\begin{aligned}\pi_A &= \frac{\epsilon(\delta + \kappa) + \delta(\gamma + \lambda)}{(\alpha + \gamma + \epsilon + \lambda)(\gamma + \delta + \kappa + \lambda)} \\ \pi_G &= \frac{\alpha(\delta + \kappa) + \kappa(\gamma + \lambda)}{(\alpha + \gamma + \epsilon + \lambda)(\gamma + \delta + \kappa + \lambda)} \\ \pi_C &= \frac{\gamma(\delta + \kappa) + \sigma(\gamma + \lambda)}{(\beta + \delta + \kappa + \sigma)(\gamma + \delta + \kappa + \lambda)} \\ \pi_T &= \frac{\lambda(\delta + \kappa) + \beta(\gamma + \lambda)}{(\beta + \delta + \kappa + \sigma)(\gamma + \delta + \kappa + \lambda)}\end{aligned} \quad (14)$$

It is straightforward to check that π satisfies the balance condition $\pi \Lambda = 0$ characterizing equilibrium. When reversibility holds, the equilibrium distribution (equation 14) reduces to the equilibrium distribution of the PHYLIP model as listed in equation 5.

Likelihood Computation

Maximum likelihood methods are notoriously slow compared to competing methods such as maximum parsimony (Yang 1994). However, every advance in computing speed makes maximum likelihood methods more attractive and the excuses for preferring other methods less compelling. Of course, fast hardware is only one of the reasons for the improved position of maximum likelihood. Better algorithms also play a role. In this section we sketch an algorithm for the fast computation of the deriva-

MAXIMUM LIKELIHOOD IN MOLECULAR PHYLOGENY

tives of the likelihood of an evolutionary tree. This algorithm extends the peeling method already in place for likelihood computation (Felsenstein 1981; Waterman 1996) and builds on an extension (Lange et al. 1995) of Baum's algorithms from the theory of hidden Markov chains (Baum 1972; Devijver 1985). While derivative-free methods of optimization are viable, quick optimization almost always depends on the use of derivatives, computed either exactly or numerically. The decisive advantage of derivative based methods is already clear in timed comparisons of dnaml with its sister program fastDNaml (Olsen 1994); dnaml employs a derivative-free method and fastDNaml Newton's method.

Before introducing the peeling algorithm and describing how to modify it to compute derivatives, it is helpful to introduce several definitions. Imagine an evolutionary tree drawn with its root at the top and its tips at the bottom. Every internal node n of the tree, including the root, generates three subtrees rooted at n . One of these is simply the subtree descending from n and ultimately reaching a subset of descendant tips. This subtree can be divided into a left subtree and a right subtree by taking either the left or right branch descending from n . There is also a complementary subtree starting at the original root and containing all of the branches not included in the subtree rooted at n . Figure 1 illustrates the four subtrees at node 5 of a four-taxon tree. Subfigure a is the whole tree, c is the subtree rooted at 5, d is the left subtree, e is the right subtree, and b is the complementary subtree.

In the peeling algorithm, several probabilities enter into the calculation of the likelihood P of the

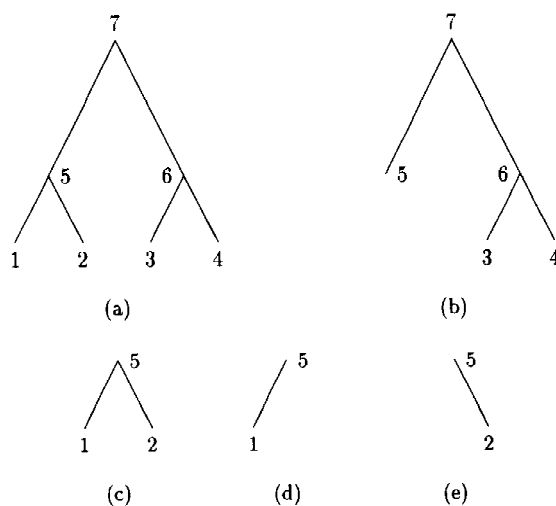


Figure 1 A four-taxon tree with subtrees at node 5.

SCHADT ET AL.

tree at a single site. These are (1) the conditional probability $T_{m \rightarrow n}(i, j)$ that the daughter node n of a branch $m \rightarrow n$ is in state j given that the mother node m is in state i ; (2) the conditional probability $D_n(i)$ of the subtree rooted at node n given node n is in state i ; (3) the conditional probability $L_n(i)$ of the left subtree rooted at node n given node n is in state i ; (4) the conditional probability $R_n(i)$ of the right subtree rooted at node n given node n is in state i ; and (5) the joint probability $U_n(i)$ of node n being in state i and the complementary subtree showing its observed bases at its tips. In definition 1, note that $T_{m \rightarrow n}(i, j)$ is nothing more than an entry of the finite-time transition matrix $P(t)$ pertaining to the branch $m \rightarrow n$. The letters D and U in $D_n(i)$ and $U_n(i)$ of definitions (2) and (5) are chosen to indicate downward and upward directions in the tree.

In the modified peeling algorithm, we make two passes through the tree. In the first pass, we traverse the tree in a postorder (upward) fashion and compute and store the conditional probabilities $[L_n(i), R_n(i), D_n(i)]$ as we visit interior node n . When we visit the root node at the end of the first pass, we recover the likelihood P of the tree. In the second pass, we traverse the tree in a preorder (downward) fashion and compute the derivatives of the likelihood with respect to each of the model parameters using the stored quantities $[L_n(i), R_n(i), D_n(i)]$ as we visit node n .

More specifically in the postorder phase of the algorithm, we employ the recurrence relations

$$L_m(i) = \sum_j T_{m \rightarrow l}(i, j) D_l(j)$$

$$R_m(i) = \sum_j T_{m \rightarrow r}(i, j) D_r(j)$$

to compute the conditional probabilities $L_m(i)$ and $R_m(i)$, where l is the left daughter of m , and r is the right daughter of m . When r and l are tips with observed states s_l and s_r , the above recurrences reduce to

$$L_m(i) = T_{m \rightarrow l}(i, s_l)$$

$$R_m(i) = T_{m \rightarrow r}(i, s_r)$$

Given the conditional probabilities $L_m(i)$ and $R_m(i)$, we compute

$$D_m(i) = L_m(i) R_m(i)$$

Derivation of these more or less obvious recurrences is left to the reader. Clearly, the computations on node m can only be carried out after m 's two daughter nodes have been processed. A postorder trans-

versal obeys this convention. Once we reach the root, we recover the likelihood via

$$P = \sum_i \pi_i D_{root}(i) \quad (15)$$

where π_i is the prior probability of the root being in state i .

In the preorder phase of the algorithm, we exploit the product rule of differentiation. Any given model parameter θ may impinge on the root distribution π_i and one or more of the branch transition probabilities $T_{m \rightarrow n}(i, j)$. The product rule implies that the partial derivative of P with respect to θ decomposes as

$$\frac{\partial}{\partial \theta} P = \frac{\partial}{\partial \theta} P_{root} + \sum_{m \rightarrow n} \frac{\partial}{\partial \theta} P_{m \rightarrow n} \quad (16)$$

in notation that will be clear momentarily. The branch contribution $(\partial/\partial\theta)P_{m \rightarrow n}$ can be computed by replacing the function $T_{m \rightarrow n}(i, j)$ by its partial derivative $(\partial/\partial\theta)T_{m \rightarrow n}(i, j)$ in the likelihood representations

$$P = \sum_i \sum_j U_m(i) R_m(i) T_{m \rightarrow n}(i, j) D_n(j)$$

involving a left branch and

$$P = \sum_i \sum_j U_m(i) L_m(i) T_{m \rightarrow n}(i, j) D_n(j)$$

a right branch, respectively. In view of Equation 15, we collect the first contribution

$$\frac{\partial}{\partial \theta} P_{root} = \sum_i \frac{\partial}{\partial \theta} \pi_i D_{root}(i)$$

when we commence the preorder traversal at the root.

Readers can rightfully object that the recipes for computing $(\partial/\partial\theta)P_{m \rightarrow n}$ invoke the undefined vectors $U_m(i)$ and, when n is a tip, $D_n(j)$. The second vector, $D_n(j)$, can be defined as $D_n(j) = 1_{(j=s_n)}$, where s_n is the base at the tip. Specification of $U_m(i)$ is more problematic. By definition $U_{root}(i) = \pi_i$. In general, if $m \rightarrow n$ is a left branch, then

$$U_n(j) = \sum_i U_m(i) R_m(i) T_{m \rightarrow n}(i, j)$$

and if $m \rightarrow n$ is a right branch, then

$$U_n(j) = \sum_i U_m(i) L_m(i) T_{m \rightarrow n}(i, j)$$

These recurrences require that mother nodes be vis-

Table 1. Likelihood Computation for Fig. 1a

Step	Node	Action
1	5	$L_5(i) = T_{5 \rightarrow 1}(i, S_1)$
2	5	$R_5(i) = R_{5 \rightarrow 2}(i, S_2)$
3	5	$D_5(i) = L_5(i) R_5(i)$
4	6	$L_6(i) = T_{6 \rightarrow 3}(i, S_3)$
5	6	$R_6(i) = T_{6 \rightarrow 4}(i, S_4)$
6	6	$D_6(i) = L_6(i) R_6(i)$
7	7	$L_7(i) = \sum_j T_{7 \rightarrow 5}(i, j) D_5(j)$
8	7	$R_7(i) = \sum_j T_{7 \rightarrow 6}(i, j) D_6(j)$
9	7	$D_7(i) = L_7(i) R_7(i)$
10	7	$P = \sum_i \pi_i D_7(i)$

ited before daughter nodes. A preorder traversal of a phylogenetic tree achieves this node visiting strategy.

As an example of the two-pass algorithm, consider the four-taxon tree shown in Figure 1a. Pass one and two consist of the steps listed in Tables 1 and 2. The partial derivatives $(\partial/\partial\theta)\pi_i$ and $(\partial/\partial\theta)T_{m \rightarrow n}(i, j)$ are calculated exactly and stored in a table prior to the start of pass two. When node n is visited during pass two, $U_n(j)$ is computed first, then $(\partial/\partial\theta)P_{m \rightarrow n}$ is computed for each parameter θ in the model. Because the $(\partial/\partial\theta)T_{m \rightarrow n}(i, j)$ are already computed and stored in a table, computing $(\partial/\partial\theta)P_{m \rightarrow n}$ for a right branch requires only that the terms $U_m(i)$, $R_m(i)$, $(\partial/\partial\theta)T_{m \rightarrow n}(i, j)$, and $D_n(j)$ be multiplied and summed over i and j . For a left branch, we substitute $L_m(i)$ for $R_m(i)$ in this calculation. When all of the nodes have been visited, the partial derivatives of P are recovered via Equation 16.

Likelihood Maximization and Simulated Annealing

We have implemented the extension of Kimura's model described above and the likelihood algorithms sketched in the previous section in a C++

program named LINNEAUS. Maximum likelihood estimation is performed within LINNEAUS by a C++ version of the optimization program SEARCH (Lange 1997). SEARCH minimizes a nonlinear function by the method of recursive quadratic programming, with quasi-Newton updates of the approximate second differential (Hessian) of the objective function based on successive values of the first differential (gradient) of the objective function (Polak 1997). It is possible to implement Newton's method using SEARCH, but the second derivatives in our model are cumbersome to compute, and LINNEAUS avoids explicit calculation of them. Exploiting the ability of SEARCH to perform numerical differentiation, we estimate that the modified peeling algorithm described in the previous section approximately doubles the speed of LINNEAUS (data not shown). These timing trials have the beneficial side effect of validating our code for the derivatives of $P(t)$. Finally, SEARCH permits upper and lower bounds and linear equality constraints to be imposed on parameters. Linear constraints are convenient because the reversibility conditions (Equation 3) can be linearized by taking logarithms. Reparam-

Table 2. Computation of Derivatives for Fig. 1a

Step	Node	Action
1	7	$U_7(i) = \pi_i$
2	7	$\frac{\partial}{\partial\theta} P_7 = \sum_i \frac{\partial}{\partial\theta} \pi_i D_7(i)$
3	5	$U_5(j) = \sum_i U_7(i) R_7(j) T_{7 \rightarrow 5}(i, j)$
4	5	$\frac{\partial}{\partial\theta} P_{7 \rightarrow 5} = \sum_i \sum_j U_7(i) R_7(i) \frac{\partial}{\partial\theta} T_{7 \rightarrow 5}(i, j) D_5(j)$
5	1	$\frac{\partial}{\partial\theta} P_{5 \rightarrow 1} = \sum_i U_5(i) R_5(i) \frac{\partial}{\partial\theta} T_{5 \rightarrow 1}(i, S_1)$
6	2	$\frac{\partial}{\partial\theta} P_{5 \rightarrow 2} = \sum_i U_5(i) L_5(i) \frac{\partial}{\partial\theta} T_{5 \rightarrow 2}(i, S_2)$
7	6	$U_6(j) = \sum_i U_7(i) L_7(j) T_{7 \rightarrow 6}(i, j)$
8	6	$\frac{\partial}{\partial\theta} P_{7 \rightarrow 6} = \sum_i \sum_j U_7(i) L_7(i) \frac{\partial}{\partial\theta} T_{7 \rightarrow 6}(i, j) D_6(j)$
9	3	$\frac{\partial}{\partial\theta} P_{6 \rightarrow 3} = \sum_i U_6(i) R_6(i) \frac{\partial}{\partial\theta} T_{6 \rightarrow 3}(i, S_3)$
10	4	$\frac{\partial}{\partial\theta} P_{6 \rightarrow 4} = \sum_i U_6(i) L_6(i) \frac{\partial}{\partial\theta} T_{6 \rightarrow 4}(i, S_4)$

SCHADT ET AL.

eterizing with log parameters has the added virtue of eliminating nonnegativity constraints.

SEARCH needs initial parameter values and an initial approximate Hessian. If maximum likelihood estimation is performed for a sequence of similar trees, then the parameter estimates from one tree can be used as initial parameter values for the next tree. The identity matrix serves as the default initial Hessian in SEARCH. Within LINNEAUS, the subprogram SEARCH minimizes the negative loglikelihood

$$-L = \sum_{i=1}^m L_i$$

where L_i is the loglikelihood of the data at site i of m sites. The sum of cross products of scores

$$H = \sum_{i=1}^m dL_i^t dL_i$$

provides an alternative to the identity matrix in approximating the Hessian

$$-d^2 L = -\sum_{i=1}^m d^2 L_i$$

In this regard, note that H is non-negative definite and has the same expectation as $-d^2 L$ relative to the data (Rao 1973).

Maximum likelihood estimation for a single tree must be fast enough to accommodate comparison of large numbers of different trees through their maximum likelihoods. With n taxa, there are $(2n - 3)!/[2^{n-2}(n - 2)!]$ rooted trees and $(2n - 5)/[2^{n-3}(n - 3)!]$ unrooted trees. These numbers are manageable for $n \leq 7$, but exhaustive consideration of all trees is out of the question for $n > 10$, and we must turn to heuristic methods of searching tree space. For example, Felsenstein (1981) suggests building an n -taxon tree by starting with any two-taxon subtree and successively adding one taxon at a time. At each stage the new taxon is attached to the existing branch that gives the subtree with largest maximum likelihood. Because the initial pair of taxa is arbitrary, Felsenstein further recommends repeating the construction using different rearrangement orders. Unfortunately, even further refinements of this greedy strategy still examine only a small fraction of tree space (Olsen 1994; Yang 1994).

Because the combinatorial optimization method of simulated annealing permits a wider search, Lundy (1985) suggested its use in phylogenetic reconstruction. For the sake of brevity, we omit a description of simulated annealing and refer readers to the references (Kirkpatrick 1983; Press

1988) for motivation and most details of implementation. The unique elements in the current setting are the energy function, the proposal mechanism, and the initial temperature. It is natural to define the energy of a tree to be its negative maximum loglikelihood. In the proposal stage, we randomly select any node other than the root node, delete the subtree rooted at this node, and reattach the subtree to a randomly selected branch among the remaining branches. The new tree is accepted with the Metropolis probability

$$\min\{1, e^{-[f(\text{tree}_{\text{new}}) - f(\text{tree}_{\text{cur}})]/t_{\text{cur}}}\}$$

where f is the energy function (the negative maximum loglikelihood) and t_{cur} is the current temperature. To obtain a good starting temperature, we randomly generate 10 trees and set the initial temperature equal to a constant times the largest absolute difference between the corresponding energies. The constant 2 has proved to be satisfactory for up to 10 taxa. Because simulated annealing gives no guarantee of convergence to the best tree under realistic cooling schedules, we typically conduct several independent runs.

Program Performance

One obvious way of improving program speed is to exploit parallel processing. The fastDNAMl program of Olsen et al. (1994) achieves superior performance by splitting likelihood computation for a phylogenetic tree into several processes that take advantage of massively parallel and clustered computing architectures. In writing LINNEAUS in Microsoft Visual C++ 5.0, we chose the most natural synchronous multithreading approach to program parallelization. LINNEAUS simply assigns each available processor to a different site of an aligned sequence. We achieve a further gain in efficiency by amalgamating those sites with a common pattern of observations and multiplying the loglikelihood and score computed for the pattern by the number of participating sites. Note that most sites in an alignment show complete agreement over all taxa; the exceptions (or segregating sites) generate most of the patterns. Under a symmetric multiprocessing operating system such as Windows NT 4.0, our current code detects the number of processors available at runtime and assigns a different observational pattern to each processor. When a processor finishes its thread, that is, computing the corresponding loglikelihood and score, it is reassigned to a new pattern unless none remains. Because most of the time

MAXIMUM LIKELIHOOD IN MOLECULAR PHYLOGENY

Table 3. Per Tree Timing Results for Maximum Likelihood

Number of taxa	Number of segregating sites	Single-processor times (sec)	Multiprocessor times (sec)
4	114–180	0.282 ± 0.062	0.149 ± .0336
5	195–641	0.706 ± 0.250	0.365 ± 0.156
6	307–931	1.54 ± 0.45	0.756 ± 0.248
10	902–2163	4.08 ± 0.96	2.02 ± 0.43

in likelihood maximization is spent in these threads, our computing times per likelihood maximization are nearly inversely proportional to the number of available processors. Linear scalability of this sort is one of the hallmarks of good parallel code.

To test the performance of LINNEAUS, we examined 12 full-length HIV genome sequences available from the Los Alamos HIV web site (Myers 1996). These taxa were obtained from 10 individuals and also include one clade B consensus sequence. The size (~10,000 nucleotide sites per sequence) and the number of segregating sites (~15% of the sites per sequence) represent a challenging computational problem. We selected 20 groups of 4, 5, and 6 taxa each and 7 groups of 10 taxa each from the original 12 taxa. Table 3 lists the average time, plus or minus one standard deviation, necessary to maximize the likelihood of one tree for the different size trees. The ALR Revolution 2× desktop computer used in these timing studies has two 300 MHz Intel Pentium II processors, each with a 512-kb L2 cache and 512 Mb of RAM. We have replicated the linear scalability results evident in Table 3 on an older four-processor machine. These four-processor results are comparable to the two-processor results and are omitted. We have also partially ported our code to a 14-processor Sun Ultra Enterprise 5000 with 1GB of RAM running under the Solaris operating system. Our preliminary results on this high-end computer demonstrate that in excess of 100 likelihoods can be maximized per second for 10-taxon trees.

Data Example

In this section, we apply our generalization of Kimura's model to a portion

of the four-taxon data used by Lake (1988) to infer the closest prokaryotic relative of the eukaryotes. Lake aligned ribosomal RNA from a eukaryote (*Aretemia salina*) and from three prokaryotes, the eocyte (*Desulfurococcus mobilis*), the halobacterium (*Halococcus morrhuae*), and the eubacterium (*Bacillus subtilis*). Here we examine a subset of 1092 aligned nucleotide quartets

from Lake's data. Adopting his designation, let E designate the topology with *A. salina* and *D. mobilis* as sister taxa, let F designate the topology with *A. salina* and *H. morrhuae* as sister taxa, and let G designate the topology with *A. salina* and *B. subtilis* as sister taxa (see Fig. 2).

Table 4 gives parameter estimates and maximum loglikelihoods (base e) for the most probable trees under four different models. Model 1 is the equilibrium version of F84 model described in Kishino and Hasegawa (1989). Model 2 is our model (Equation 2) with the reversibility restrictions (Equation 3) and equilibrium at the root imposed. Both models 1 and 2 satisfy the pulley principle and therefore, cannot be used to root trees. Model 3 coincides with model 2 except that the equilibrium assumption is relaxed. This entails adding three additional parameters to determine the root distribution. Model 4 coincides with model 3 except that the reversibility assumption is relaxed. Simulated data that we omit show that models 3 and 4 can be used to root trees.

Under all four models, the best tree has the E topology. For models 1 and 2, the maximum loglikelihoods of the next best topology, the F topology

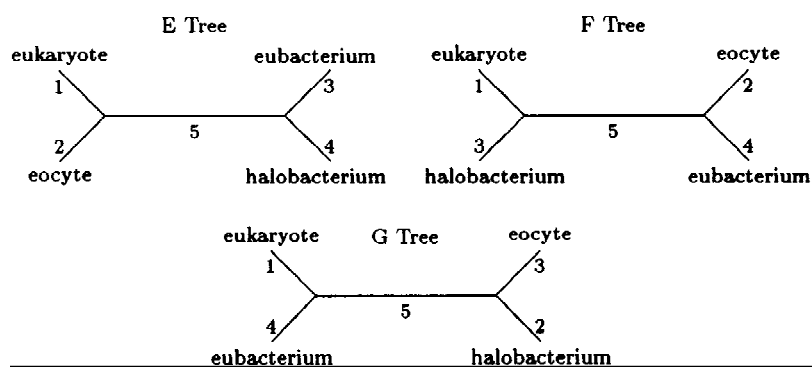


Figure 2 The E, F, and G topologies represent all possible topologies for the four-taxon case.

SCHADT ET AL.

Table 4. Parameter Estimates for the Best Trees Under Four Sets of Modeling Assumptions

Parameter	Model 1	Model 2	Model 3	Model 4
α		0.155 ± 0.016	0.125 ± 0.015	0.051 ± 0.013
β		0.174 ± 0.019	0.200 ± 0.021	0.217 ± 0.023
γ		0.084 ± 0.007	0.071 ± 0.007	0.078 ± 0.009
δ		0.080 ± 0.007	0.092 ± 0.009	0.060 ± 0.009
ϵ				0.174 ± 0.018
κ		0.107 ± 0.009	0.083 ± 0.008	0.117 ± 0.013
λ		0.067 ± 0.006	0.086 ± 0.008	0.078 ± 0.009
σ				0.134 ± 0.025
U	0.333 ± 0.023			
κ	0.464 ± 0.077			
π_A	0.233 ± 0.013		0.202 ± 0.012	0.193 ± 0.012
π_G	0.316 ± 0.014		0.362 ± 0.015	0.373 ± 0.015
π_C	0.245 ± 0.013		0.290 ± 0.014	0.297 ± 0.015
π_T	0.206 ± 0.012		0.141 ± 0.011	0.137 ± 0.011
Branch 1	1 ± 0	1 ± 0	1 ± 0	1 ± 0
Branch 2	0.232 ± 0.047	0.234 ± 0.047	.001	0.110 ± 0.044
Branch 3	0.538 ± 0.065	0.539 ± 0.065	0.544 ± 0.065	0.568 ± 0.065
Branch 4	1.650 ± 0.154	1.642 ± 0.153	1.671 ± 0.155	1.690 ± 0.154
Branch 5	0.194 ± 0.052	0.188 ± 0.051	0.196 ± 0.051	0.190 ± 0.050
Branch 6			0.215 ± 0.046	0.082 ± 0.050
Best tree	topology E	topology E	tree 5	tree 5
Iterations	11	11	11	11
Max log likelihood	-4606.6	-4598.2	-4555.7	-4536.2
χ^2 (df)	140.8 (7)	124.0 (6)	39.0 (2)	

in both cases, are -4614.2 and -4605.2 , respectively. Tree 5 in Figure 3 depicts the best rooted tree for both models 3 and 4. Unfortunately, we have little confidence in this rooting. In the case of model 4, the two remaining trees in the E topology have maximum loglikelihoods of -4537.3 and -4537.4 , respectively. Similarly inconclusive results hold for model 3. Under models 3 and 4, the best maximum loglikelihoods for trees with the F or G topology (-4564.1 and -4544.2 , respectively) are substantially worse than the maximum likelihoods listed in Table 4. LINNEAUS reached the maximum likelihood estimates for all four models listed in Table 4 in just 11 iterations.

Models 1, 2, and 3 are nested within model 4. Within the context of the best topology, or where applicable the best rooted tree, we can therefore conduct likelihood ratio tests of the three submodels versus model 4. The χ^2 statistics given at the bottom of Table 4 overwhelmingly reject each of the submodels. Clearly, assuming that the four nucleotides are at equilibrium is a poor idea in these data. This impression is reinforced by the empirical frequencies shown in Table 5. Relaxing the reversibility restrictions also substantially improves model

fit. Finally, the fact that we estimate the length of branch 2 in model 3 at the somewhat artificial lower bound of 0.001 is another indication of the superiority of model 4.

Because all four models represent restrictions of the multinomial model with $4^4 - 1 = 255$ degrees of freedom (Navidi et al. 1991), goodness of fit can be roughly assessed by comparing the maximum loglikelihood (-4536.2) under the best tree of model 4 with the maximum loglikelihood (-4361.3) under the unrestricted multinomial model. The resulting χ^2 test statistic $\chi^2_{238} = 349.8$ has asymptotic P value ≈ 0.0001 . This apparent lack of fit may spring from the failure of any of three different assumptions: (1) the transversion symmetries in our matrix Λ , (2) the applicability of large sample approximations to sparse data, and (3) the assumption that the various sites act independently and identically. Although it is impossible to make a firm judgment in the current case, it is worth noting that the phenomenon of heterogeneity of substitution rates across sites is well-documented (Gojobori 1982; Yang 1993). In particular, Rzhetsky (1995) shows substantial variation in the substitution rates between the stem and

MAXIMUM LIKELIHOOD IN MOLECULAR PHYLOGENY

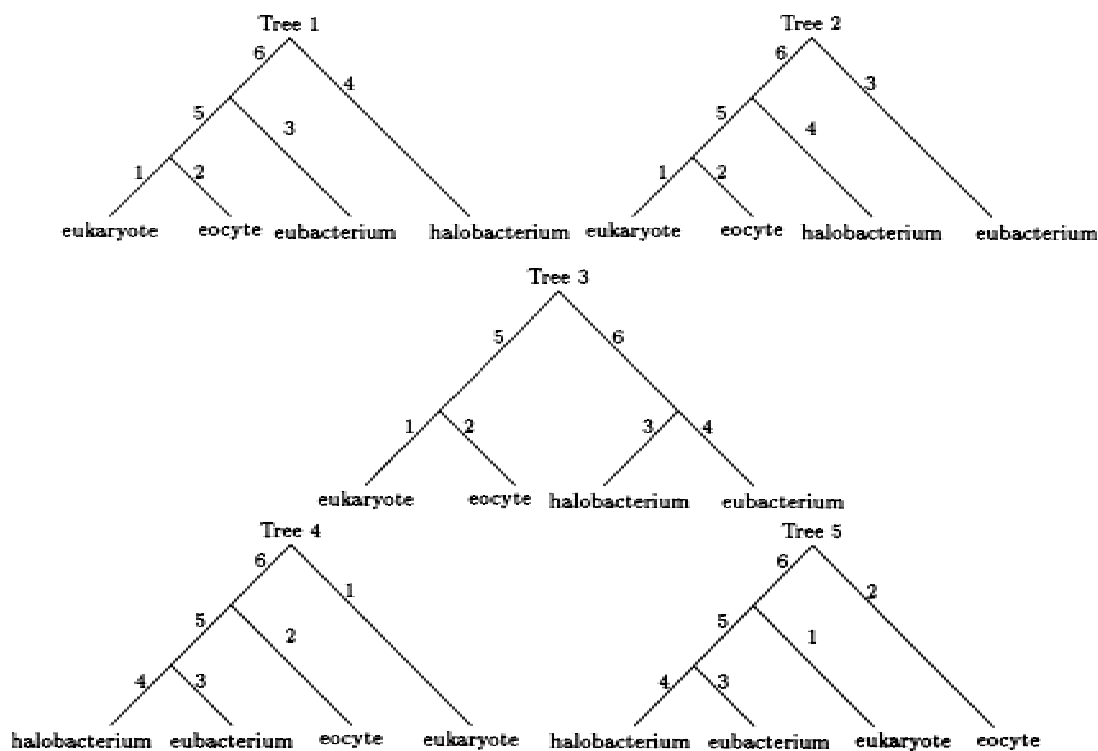


Figure 3 The five rooted trees corresponding to the unrooted tree with the E topology. These rooted trees are obtained by placing the root in each of the five branches of the unrooted tree.

loop regions of 16S-like ribosomal RNA genes from higher eukaryotes.

Conclusions

Maximum likelihood is at an obvious computational disadvantage in molecular phylogeny. To a lesser extent, it is also handicapped by the simplicity of the models it traditionally invokes. The thrust of the current paper is that advances in model construction and algorithm design can go hand in hand. Although our generalization of Kimura's base-substitution model involves a more complicated infinitesimal generator, the new generator Λ retains just enough symmetry to permit exact calculation of the matrix exponential $e^{t\Lambda}$ and its corresponding equilibrium distribution π . Furthermore, this more flexible model can be coupled with better algorithms for maximum likelihood estimation and search of tree space.

A key ingredient in the rapid optimization of any function is the ability to eval-

uate it and its derivatives quickly. Our new algorithm for likelihood differentiation should be viewed in this light. Combining this algorithm with a good, general purpose optimization subprogram and parallelizing the resulting code leads to very fast maximum likelihood estimation per tree. In fact, we now consider our likelihood optimization methods fast enough to harness simulated annealing in a search of tree space for the best tree. While simulated annealing is slower than competing methods in phylogenetic inference, it is slower precisely because it analyzes a greater number of trees, and so, tends to the best tree with greater probability.

Our infinitesimal generator Λ , which is irrevers-

Table 5. Observed Base Proportions for the Four Contemporary Taxa

Base	<i>A. salina</i> (eukaryote)	<i>D. mobilis</i> (eocyte)	<i>H. morrhuae</i> (halobacterium)	<i>B. subtilis</i> (eubacterium)
A	0.250	0.232	0.202	0.255
G	0.319	0.328	0.367	0.279
C	0.230	0.263	0.290	0.234
T	0.202	0.178	0.141	0.232

SCHADT ET AL.

ible in general, opens up the possibility of rooting trees. As Felsenstein's celebrated pulley principle shows, rooting is impossible in a reversible model when the root frequencies are at equilibrium and no constraints are imposed on branch lengths. Abandoning any of these three assumptions makes it possible in principle to root a tree. For instance, if constraints are imposed on branch lengths by assuming a molecular clock, it is possible to root a tree. However, the practical difficulty of rooting in the presence of limited data has led many researchers to dismiss the possibility altogether. Yang (1994) raises the additional numerical concern that the eigenvalues of the infinitesimal generator may be complex when reversibility fails. It is interesting to note that all eigenvalues of our matrix Λ are real.

Our experience only partially bears out the pessimism about rooting. Although topologically equivalent trees usually have similar maximum likelihoods and the root position within the best tree is usually poorly determined, we do see substantial increases in loglikelihoods when passing from simple reversible models to our more complex irreversible model. In our opinion, the massive amounts of data coming out of the various genome sequencing projects will fundamentally change the course of phylogenetic inference. Instead of aligned sequences of 10^3 sites, we may well have aligned sequences of 10^6 – 10^8 sites. Of course, this quantitative shift merely emphasizes the necessity of further gains in computing efficiency if maximum likelihood is to be the phylogenetic inference engine of choice.

In any event, the results presented in this article are meant to demonstrate the feasibility of major improvements in computing speed in maximum likelihood methods. Researchers interested in a free copy of LINNEAUS should contact one of the authors.

ACKNOWLEDGMENTS

We thank Professor James A. Lake for providing the aligned DNA sequences for our Section 6 example. The authors would also like to thank the reviewers and the editor for their helpful comments and clarifications. Research was supported in part by U.S. Public Health Service grant GM53275 (to K.L.) and CA16042 (to J.S.S.) and National Institute of General Medical Sciences (NIGMS) Systems and Integrative Biology Training grant GM08185-11 (to E.E.S.).

The publication costs of this article were defrayed in part by payment of page charges. This article must therefore be hereby marked "advertisement" in accordance with 18 USC section 1734 solely to indicate this fact.

REFERENCES

- Adachi, J., and M. Hasegawa. 1996. *MOLPHY*, Version 2.3. The Institute of Statistical Mathematics, Tokyo, Japan.
- Barry, D., and J.A. Hartigan. 1987. Statistical analysis of hominoid molecular evolution. *Stat. Sci.* 2: 191–210.
- Baum, L.E. 1972. An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. *Inequalities* 3: 1–8.
- Devijver, P.A. 1985. Baum's forward-backward algorithm revisited. *Pattern Recognition Lett.* 3: 369–373.
- Felsenstein, J. 1981. Evolutionary trees from DNA sequences: A maximum likelihood approach. *J. Mol. Evol.* 17: 368–376.
- Felsenstein, J. and G.A. Churchill. 1996. A Hidden Markov Model approach to variation among sites in rate of evolution. *Mol. Biol. Evol.* 13: 93–104.
- Gojobori, T., K. Ishii, and M. Nei. 1982. Estimation of average number of nucleotide substitutions when the rate of substitution varies with nucleotides. *J. Mol. Evol.* 18: 414–423.
- Hasegawa, M., H. Kishino, and T. Yano. 1985. Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *J. Mol. Evol.* 22: 160–174.
- Kelly, F.P. 1979. *Reversibility and stochastic networks*. Wiley, New York, NY.
- 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.
- Kirkpatrick, S., C.D. Gelatt, and M.P. Vecchi. 1983. Optimization by simulated annealing. *Science* 220: 671–680.
- Kishino, H., and M. Hasegawa. 1989. Evaluation of the maximum likelihood estimates of the evolutionary tree topologies from DNA sequence data and the branching order in Hominoidea. *J. Mol. Evol.* 29: 170–179.
- Lake, J.A. 1987. A rate-independent technique for analysis of nucleic acid sequences: Evolutionary parsimony. *Mol. Biol. Evol.* 4: 167–191.
- . 1988. Origin of the eukaryotic nucleus determined by rate-invariant analysis of rRNA sequences. *Nature* 331: 184–186.
- Lange, K. 1997. *SEARCH*, Version 3.0, University of Michigan, Ann Arbor, MI.
- Lange, K., M. Boehnke, D.R. Cox, and K.L. Lunetta. 1995. Statistical methods for polyploid radiation hybrid mapping. *Genome Res.* 5: 136–150.

Lundy, M. 1985. Applications of the annealing algorithm to combinatorial problems in statistics. *Biometrika* 72: 191–198.

Myers, G., B. Korber, S. Wain-Hobson, K.T. Jeang, L.E. Henderson, and G.N. Pavlakis. 1996. *Human retroviruses and AIDS*. Los Alamos National Laboratory, Los Alamos, NM.

Navidi, W.C., G.A. Churchill, and A. von Haeseler. 1991. Methods for inferring phylogenies from nucleic acid sequence data by using maximum likelihood and linear invariants. *Mol. Biol. Evol.* 8: 128–143.

Olsen, G.J., H. Matsuda, R. Hagstrom, and R. Overbeek. 1994. fastDNAm1: A tool for construction of phylogenetic trees of DNA sequences using maximum likelihood. *Comput. Appl. Biosci.* 10: 41–48.

Polak, E. 1997. *Optimization: Algorithms and consistent approximations*. Springer-Verlag, New York, NY.

Press, W.H., B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. 1988. *Numerical recipes in C*. Cambridge University Press, New York, NY.

Rao, C.R. 1973. *Linear statistical inference and its applications*, 2nd ed. Wiley, New York, NY.

Rzhetsky, A. 1995. Estimating substitution rates in ribosomal RNA genes. *Genetics* 141: 771–783.

Tamura, K. and M. Nei. 1993. Estimation of the number of nucleotide substitutions in the control region of mitochondrial DNA in humans and chimpanzees. *Mol. Biol. Evol.* 10: 512–526.

Tavare, S. 1986. Some probabilistic and statistical problems in the analysis of DNA sequences. *Lect. Math. Sci.* 17: 57–86.

Waterman, M.S. 1996. *Introduction to computational biology: Maps, sequences and genomes*. Chapman & Hall, New York, NY.

Yang, Z. 1993. Maximum-likelihood estimation of phylogeny from DNA sequences when substitution rates differ over sites. *J. Mol. Evol.* 10: 1396–1401.

———. 1994. Estimating the pattern of nucleotide substitution. *J. Mol. Evol.* 39: 105–111.

Received December 10, 1997; accepted in revised form January 30, 1998.

APPENDIX

If the constants c_1 through c_7 are defined as in the text, and if

$$c_8 = \delta - \epsilon$$

$$c_9 = \lambda - \beta,$$

then the 16 entries of $P(t)$ are

MAXIMUM LIKELIHOOD IN MOLECULAR PHYLOGENY

$$p_{AA}(t) = \pi_A - \frac{c_2 c_8}{c_1(c_3 - c_1)} e^{-c_1 t} + \frac{\alpha(c_3 - c_1) - c_2 c_4}{c_3(c_3 - c_1)} e^{-c_3 t}$$

$$p_{AG}(t) = \pi_G - \frac{c_2 c_4}{c_1(c_3 - c_1)} e^{-c_1 t} + \frac{c_2 c_4 - \alpha(c_3 - c_1)}{c_3(c_3 - c_1)} e^{-c_3 t}$$

$$p_{AC}(t) = \pi_C + \frac{c_2 c_7}{c_1(c_6 - c_1)} e^{-c_1 t} + \frac{\lambda\sigma - \gamma\beta}{c_6(c_6 - c_1)} e^{-c_6 t}$$

$$p_{AT}(t) = \pi_T + \frac{c_2 c_9}{c_1(c_6 - c_1)} e^{-c_1 t} + \frac{\beta\gamma - \lambda\sigma}{c_6(c_6 - c_1)} e^{-c_6 t}$$

$$p_{GA}(t) = \pi_A - \frac{c_2 c_8}{c_1(c_3 - c_1)} e^{-c_1 t} + \frac{c_2 c_8 - \epsilon(c_3 - c_1)}{c_3(c_3 - c_1)} e^{-c_3 t}$$

$$p_{GG}(t) = \pi_G - \frac{c_2 c_4}{c_1(c_3 - c_1)} e^{-c_1 t} + \frac{\epsilon(c_3 - c_1) - c_2 c_8}{c_3(c_3 - c_1)} e^{-c_3 t}$$

$$p_{GC}(t) = p_{AC}(t)$$

$$p_{GT}(t) = p_{AT}(t)$$

$$p_{CA}(t) = \pi_A + \frac{c_5 c_8}{c_1(c_3 - c_1)} e^{-c_1 t} + \frac{\epsilon\kappa - \delta\alpha}{c_3(c_3 - c_1)} e^{-c_3 t}$$

$$p_{CG}(t) = \pi_G + \frac{c_5 c_4}{c_1(c_3 - c_1)} e^{-c_1 t} + \frac{\delta\alpha - \epsilon\kappa}{c_3(c_3 - c_1)} e^{-c_3 t}$$

$$p_{CC}(t) = \pi_C - \frac{c_5 c_7}{c_1(c_6 - c_1)} e^{-c_1 t} + \frac{\beta(c_6 - c_1) - c_5 c_9}{c_6(c_6 - c_1)} e^{-c_6 t}$$

$$p_{CT}(t) = \pi_T - \frac{c_5 c_9}{c_1(c_6 - c_1)} e^{-c_1 t} + \frac{c_5 c_9 - \beta(c_7 - c_1)}{c_6(c_6 - c_1)} e^{-c_6 t}$$

$$p_{TA}(t) = p_{CA}(t)$$

$$p_{TG}(t) = p_{CG}(t)$$

$$p_{TC}(t) = \pi_C - \frac{c_5 c_7}{c_1(c_6 - c_1)} e^{-c_1 t} + \frac{c_5 c_7 - \sigma(c_6 - c_1)}{c_6(c_6 - c_1)} e^{-c_6 t}$$

$$p_{TT}(t) = \pi_T - \frac{c_5 c_9}{c_1(c_6 - c_1)} e^{-c_1 t} + \frac{\sigma(c_6 - c_1) - c_5 c_7}{c_6(c_6 - c_1)} e^{-c_6 t}$$

The infinitesimal generator Λ has eigenvalues 0, $-c_1$, $-c_3$, and $-c_6$ and corresponding right eigenvectors

$$v_1 = (1, 1, 1, 1)^t$$

$$v_2 = \left(1, 1, -\frac{c_5}{c_2}, -\frac{c_5}{c_2} \right)^t$$

$$v_3 = \left(\frac{\alpha(c_5 - c_3) + \kappa c_2}{\delta(c_3 - c_2) - \epsilon c_5}, \frac{-\epsilon(c_5 - c_3) - \delta c_2}{\delta(c_3 - c_2) - \epsilon c_5}, 1, 1 \right)^t$$

$$v_4 = \left(1, 1, \frac{\beta(c_2 - c_6) + \lambda c_5}{\gamma(c_6 - c_5) - \sigma c_2}, \frac{-\sigma(c_2 - c_6) - \gamma c_5}{\gamma(c_6 - c_5) - \sigma c_2} \right)^t$$

These can be checked by straightforward substitution.