An Information-Theoretic Method for the Treatment of Plural Ancestry in Phylogenetics

Supriya Munshaw*† and Thomas B. Kepler*‡

*Department of Biostatistics and Bioinformatics, Center for Computational Immunology, Duke University Medical Center, Durham, NC; †Computational Biology and Bioinformatics PhD Program, Institute for Genome Sciences and Policy, Duke University; and ‡Department of Immunology, Department of Statistical Science, Duke University

In the presence of recombination and gene conversion, a given genomic segment may inherit information from 2 distinct immediate ancestors. Such “plural ancestry (PA)” is ubiquitous in evolution but is often neglected in the phylogenetic inference of natural histories. In the absence of PA, the topology of any resulting phylogeny is tree-like and consistent across the length of the gene. The representation of natural histories with PA requires the use of phylogenetic networks (Hudson and Kaplan 1985) or general acyclic graphs (Maynard Smith 1989). Statistical modeling on general acyclic graphs is substantially more difficult than on trees; inference of phylogenetic relationships in these cases is correspondingly more problematic. In spite of these difficulties, such modeling is critical. Schierup and Hein (2000) have shown that the neglect of PA can lead to markedly mistaken conclusions.

PA regulates the diversification of multigene families and plays a key role in the adaptive evolution of viruses within quasispecies and of alleles within populations; it is particularly important in the coevolution of higher organisms and their microbial symbionts. Gene conversion, for example, has recently been shown to have played a key role in the evolution of the interferon alpha gene family in mammals (Woelk et al. 2007). RNA viruses such as the human immunodeficiency virus (HIV) utilize virion-to-virion genetic exchange to evade host defenses and overcome susceptibility to multiple antiviral drugs (Robertson et al. 1995). An especially well-studied case of mosaicism, in vertebrates, in which antigen receptor genes undergo somatic recombination of gene segments in lymphocytes creating a large repertoire of specificities against a wide range of antigens (Tonegawa 1983). Antigen receptor genes in jawless vertebrates, unrelated to genes of similar function in jawed vertebrates, have recently been shown to be generated by gene conversion (Pancer et al. 2004), as have host defense genes in an invertebrate, Biomphalaria glabrata (Zhang et al. 2004).

Several distinct classes of methods for the treatment of mosaic sequences have been developed. Each of these infers different types of information relevant to genetic mosaicism. One includes methods that identify recombination without phylogenetic inference. For instance, Hudson and Kaplan (1985) developed a method that provides a lower bound on the number of recombination events that occurred in the history of a collection of sequences that does not require inference of a phylogeny. Sawyer (1989) developed methods for the detection of genetic mosaicism by looking at the distribution of segments of synonymous polymorphisms, again, without the use or inference of phylogenies.

Methods that generate inferred phylogenies include that of Grassly and Holmes (1997) that compares the likelihood of an evolutionary model constrained to use a single phylogenetic tree for all sites in the gene with those of models in which the tree topology is allowed to vary along the length of the gene. Although this method does not produce a single phylogenetic network, the multiple inferred trees can in principle be superimposed to produce such a network.

Hein (1990) poses the problem within the context of dynamic programming and describes an algorithm to identify the phylogenies, varying across sites, that minimize the sum of mutation and recombination costs, allowing these costs to be arbitrarily fixed. Though this algorithm is computationally infeasible, Hein (1993) also published a much simpler, fast algorithm based on a reasonable heuristic regarding the topologies that are likely to be important. These methods represent generalizations of the method of parsimony and are thus subject to the same criticisms (Felsenstein 1988).

Strimmer and Moulton (2000) addressed these perceived shortcomings by applying maximum likelihood
on directed acyclic graphs as an extension to Felsenstein’s method for trees. Felsenstein’s method is based on the conditional probability $P(y|x,t)$ of observing nucleotide $y$ given parent $x$ and intervening time $t$. Strimmer and Moulton adjust this definition to allow for PA by introducing priors $p_1$, $p_2$ for the probabilities that there are $2 \text{ parents,} \ 1 \text{ or } 2$, respectively. The graph is then constructed using the conditional probabilities.

In a more recent method, Jin et al. (2006) applied maximum likelihood methods to reticulate networks, which are obtained from trees by the addition of additional edges between edges in the original tree. Their extension is based on the decomposition of the phylogenetic network into overlapping subtrees and uses a maximum likelihood criterion with a branch-and-bound heuristic to reconstruct the phylogenetic history of putative recombinants.

The use of maximum likelihood methods in this context represents a significant step forward but faces the challenge posed by the need to choose from among several models of differing sizes. As pointed out by the authors, adding a reticulation edge always increases the likelihood. Hypothesis testing can be applied, but the hierarchy of models thus formed is not necessarily transitive. For 3 models A, B, and C, it is possible that A is rejected in favor of B, B is rejected in favor of C, yet A is not rejected in favor of C. Indeed, the authors do not actually test for the presence of recombination but use a visual heuristic to choose the size of the final model.

We have developed an information-theoretic method for phylogenetic inference in the presence of PA. The method is an outgrowth of an approach we developed for the analysis of mosaicism in host defense genes (Zhang et al. 2004) and is based on model selection by minimization of the “description length,” as we now explain.

The minimum description length (MDL) principle casts the model selection problem as that of finding the most efficient encoding of the data (Grünwald 2007). This approach arose from the theoretical work of Kolmogorov (1965), Solomonoff (1964), and Chaitin (1966) on data complexity, a concept developed to provide a definition of randomness applicable to individual data streams rather than exclusively to ensembles. The idea is to define the complexity of a data stream $D$ as the length, in bits, of the shortest computer program that produces $D$ as output. A random data stream is one that cannot be compressed at all—the shortest program essentially is PRINT $D$. The underlying intuition is that regularities allow data compression; randomness is the complete absence of regularity. Rissanen (1987) extended these abstract ideas to the practical problem of statistical data analysis by placing restrictions on the kinds of computers and programs that can be considered.

The analogy Rissanen uses is that of communication: The data stream is a message to be sent after appropriate encoding. An efficient encoding takes advantage of the data’s regularities as Morse code encodes the frequently used letters “e” as “dot” and the rarer letter “q” as “dash dash dot dash.” This strategy requires the transmission of the message in 2 parts. The first part contains the code key and the second contains the message itself, encoded using the key. In more familiar terms, the code key is analogous to the statistical model and the associated parameter estimates, and the “message itself” is analogous to the data residuals. The complexity of the model and the lack of fit are thus both evaluated using information as the common currency.

Viewed from this perspective, a tree is primarily a very efficient means of encoding data, such as DNA sequences, that contain patterns of shared features. Data compression is achieved by encoding common characteristics at the root and adding more and more specific characteristics (shared by fewer individuals) as accumulating differences as the leaves are approached. It is worth pointing out in this regard a passage from Darwin (1859) in which he describes classification by the “Natural System” as an artificial means for enunciating, as briefly as possible, general propositions, that is, by one sentence to give the characters common, for instance, to all mammals, by another those common to all carnivora, by another those common to the dog-genus, and then by adding a single sentence, a full description is given of each kind of dog. The ingenuity and utility of this system are indisputable.

It is precisely the “data compression” properties of trees that he is referring to and it is their extraordinary utility in this context that drove him to posit the “propinquity of descent” as the cause.

There is an intimate relationship between code length and probability captured by the Kraft inequalities (see, e.g., Grünwald 2007), which provides a 1-l map between coding schemes and likelihood functions. In particular, the asymptotic expansion in $N$, the size of the data set, of the MDL has minus the log of the maximum likelihood as the leading coefficient (order $N$).

The method we have developed is realized in a stochastic minimization of the description length over the space of phylogenetic networks; each elementary transaction involves random modification of the network topology. We have implemented the method in software, validated its performance on simulated data sets, and demonstrated its application on a collection of genomic sequences of the HIV-1 envelope protein isolated longitudinally from each of 8 subjects infected with HIV (Shankarappa et al. 1999).

**Materials and Methods**

**Inference Procedure**

The number of phylogenetic tree networks grows superexponentially with the size of the data (Felsenstein 1978), and the number of phylogenetic networks grows more rapidly still, making exhaustive consideration of all topologies infeasible.

We have therefore designed our procedure as a stochastic optimization on the space of phylogenetic networks. The procedure uses simulated annealing (Kirkpatrick et al. 1983) to perform the minimization of the MDL of the data plus the network as the objective function.

The process is carried out in 2 stages. During stage 1, the tree-like topology is preserved by restricting the elementary operations that are allowed. This stage is intended to find a suitable starting point for the second stage, in which an enlarged set of elementary operations is applied,
including some that do not preserve the tree topology. After each operation, the description length is evaluated and used to determine whether the result of the operation will be preserved or discarded. The procedure is explained below in greater detail.

The Scoring Function—MDL

We have cast the determination of the phylogenetic history of a set of sequences explicitly as a problem of model selection in which we seek the model that minimizes the total information required to encode the data or in other words, the description length of the data given a model.

A phylogenetic tree can be viewed as a hierarchical data structure that enables an efficient encoding of a set of similar DNA sequences through elimination of redundancies. Consider the encoding of 2 similar genes, each of length \( L \). One could use a naive model and encode the 2 genes independently, at an information cost of \( 4L \) bits (1 nt requires 2 bits to encode). Or one could encode the consensus sequence (costing 2 \( L \) bits) and then encode the changes required to recover each of the 2 genes from the consensus. Roughly speaking, each mutation requires \( \log_2 3 \) bits to specify the class of mutation and \( \log_2 L \) bits to specify the nucleotide position at which it occurs. As long as the number of mutations, \( m \), required for this encoding is small enough to satisfy \( m(\log_23 + \log_2L) < 2L \), the tree model is more efficient than the naive model.

For larger gene sets, the process is similar, though the models are more complex. Each gene must specify its parent and the changes to the parent that produce the gene. The set of pointers from genes to their parents is equivalent to the topology of the tree, and the encoding of the changes from parent to "child" is accomplished through the use of a model for mutations.

We extend this basic coding scheme by allowing any node in the network to have 2 ancestors and thus be treated as a mosaic sequence. In this case, we need to specify both ancestors, the break point where the mosaic switches from one to the other, and the mutations between this parental hybrid and the gene of interest. The point is to account accurately for the cost of allowing PA. For each sequence alignment we consider, we will use and compare 2 different models. One, designated \( M_1 \), prohibits PA, and the other, \( M_2 \), allows it.

Although the method can be used with any mutation model, we use a relatively simple model in what follows here. This model allows for different mutation rates for each of the 4 classes of pairwise relationships between nucleotides: identity, transition, transversion 1 (G,A \( \leftrightarrow \) C,T), and transversion 2 (G,A \( \leftrightarrow \) T,C). We further assume that these rates are uniform across positions in the gene and over all branches. Finally, we assume equal branch lengths throughout the tree.

Under these conditions, the mutations are distributed according to a multinomial model. The probability density function is defined as

\[
P(x|\vec{p}) = p_1^{x_1}p_2^{x_2}p_3^{x_3}p_4^{x_4},
\]

where \( x \) is the unknown parameter vector of length 4 representing the relationships between nucleotides mentioned above (the sum of the components of this vector equals the total number \( N \) of nucleotides in the network) and \( p \) contains the proportions of each relationship. Taking a uniform prior on the parameter vector, we integrate it out and obtain the total information required to encode the mutations given this model

\[
I_m = \log_2 \int dp \, P(x|\vec{p}) P(\vec{p}) = \log_2 \left( \frac{(N-1)L}{3} \right) + \log_2 \left( \frac{(N-1)L}{x} \right),
\]

where \( N \) is the total number of nodes in a given network (including the root) and \( L \) is the common length of the sequences. The arguments of the logarithms are the binomial and multinomial coefficients, respectively.

Equation (2) has a useful direct interpretation in terms of coding: The first term is the information required to specify the counts of the components of vector \( x \) given the total number, \( (N-1)L \), of nucleotides in a network (the root is excluded because there are no mutations in the root). The second term accounts for the number of ways these mutations can be assigned to the \( N \) nucleotides of the data set given the counts in vector \( x \). We ignore a term \( 2L \) that represents the amount of information required to encode the root. Because all networks in question will have a root with the same length, this term will not make any difference in the final description length.

In addition to the information required to encode the mutations, we need to encode the ancestry as well. Under a singular ancestry model, each node must specify a parent. If \( N_I \) is the number of internal nodes and \( N_O \) is the number of observed or leaf nodes, the ancestor information is given by equation 3

\[
I_A = \log_2(N_O - 1) + N_O \log_2 N_I + (N_I - 1) \log_2 (N_I - 1),
\]

where the first term represents the amount of information required to specify the number of internal nodes and hence all possible parents. The second term represents the amount of information required to specify a parent for an observed or leaf node and because an internal node cannot be its own parent, the third term represents the information to encode an ancestor for an internal node.

Under the PA model, we must first specify the number of recombinants in a given network and the number of ways of distributing these recombinants. Each recombinant will then have a secondary parent from one of the \( N_I \) internal nodes (a recombinant cannot have the same parent twice, so we have \( N_I - 1 \) choices) and the location of the crossover point, where the gene switches from similarity to parent 1 to similarity to parent 2. Because this switch cannot happen at the last nucleotide, specifying the recombination point takes \( \log_2 (L - 1) \) bits. If there are \( N_R \) nodes with dual
ancestry, the corresponding information cost is

\[ I_R = \log_2 \left( \frac{N - 1}{N_R} \right) + \log_2 \left( \frac{N - 1}{N_I} \right) + N_R \log_2 \left( \frac{N - 1}{L - 1} \right). \] (4)

Equation 4 might be a slight overestimation (of order 1) in that internal nodes are allowed to be recombinants but they cannot be their own parents. Here we do not distinguish between observed node recombinants and internal node recombinants.

The total information cost or the description length of the encoding under model M2 is simply the sum of the 3 terms, \( I_T = I_M + I_A + I_R \). The third term in the equation \( I_R \) is omitted when encoding under M1. The optimal model is the 1 that minimizes the total information. The trade-off of recombination is achieved by reducing the number of mutations and thus decreasing \( I_M \) by more than \( I_R \).

Simulated Annealing

As mentioned above, we start with a random initial topology. Figure 1A demonstrates an example of a topology with 9 species. The elementary operations we use allow any finite topology to be converted to any other finite topology in a finite number of steps. We then randomly assign our data to this topology and calculate the resulting description length. Note that each node in our network represents a DNA sequence. The leaf nodes represent the input sequences, whereas the sequences at the interior nodes, that is, nodes with children, are determined by the demand for minimizing the description length and hence contain the appropriate consensuses. We then allow the repeated application of the following elementary operations and their inverses: change parent, split a node, combine 2 nodes, or add a second parent. There are two main steps in the program, each of which is described in detail below.

Stage 1

- The operation to be applied is chosen from among the 3 possibilities with uniform probability: branch swapping, node absorption, and node fission.
- Branch swapping. We choose 2 nodes at random: an interior node (which will become the new parent) and a child node. We break the link between the chosen child and its current parent and link the child to the new parent. An example of branch swapping is shown in figure 1B, where the branch between species 6 and its parent in 1A has been swapped giving species 6 a new parent.
- Node absorption. We randomly choose an interior node and merge it with its parent. In other words, the chosen node is deleted from the tree, and all its children are made direct progeny of its parent. An example of node absorption is shown in figure 1C, where the parent of species 1 and 2 in 1A has been merged with the root.
- Node fission. We randomly choose an interior node and create a new node that is assigned as a child of the chosen node. The children of the chosen node are then distributed between the new child and itself minimizing the local contribution to the description length using a simple 2-means algorithm (MacQueen 1967). An example of node fission is shown in figure 1D, where the parent of species 6, 7, 8, and 9 has been split such that 8 and 9 have a separate parent.

The description length of the resulting tree is calculated at the end of each iteration within stage 1. The move is accepted if the description length of the resulting network is less than the existing one. If it is not, we use simulated annealing to choose whether to accept or reject the move. This is necessary in order to avoid local minima. Step 1 runs till the completion of simulated annealing.

Stage 2

Stage 2 differs from stage 1 by replacing branch swapping with “adoption.” Adoption proceeds by choosing
a node at random from all nonroot nodes, designating that the child and choosing new parent node from among all interior nodes.

- If the chosen child node has 1 parent, add a branch to the new parent while maintaining the branch to the old parent. Determine the recombination point between the 2 parents that minimizes the local description length by minimizing the number of mutations between the parent hybrid and the sequence at the child node. An example of a move of this kind is shown in figure 1E, where 6 now has 2 parents.
- If the chosen child already has 2 parents, we consider 5 operations and choose the 1 that produces the smallest description length.

1. Represent the child as a recombinant between the current primary parent and the new parent.
2. Represent the child as a recombinant between the current secondary parent and the new parent.
3. Break the link with both existing parents and represent the child with having only the new parent as a parent.
4. Break the link with the secondary parent and represent the child by the current primary parent only. This move is a check and is done to allow reversibility.
5. Break the link with the primary parent and represent the child by the current secondary parent only.

As in step 1, the description length of the tree resulting from the move is calculated and stage 2 is also iterated till the completion of simulated annealing. At the end of both steps, we have a network that minimizes the description length of the data.

### Simulated Data Sets

We generated simulated data sets using a Markov Process model to grow a phylogenetic network with point

<table>
<thead>
<tr>
<th>$k_s/k_r$</th>
<th>$k_l/k_r$</th>
<th>Mutations/Split</th>
<th>$I_1$ (without PA)</th>
<th>$I_1$ (with PA)</th>
<th>Gene-Wise Type I Error Rate*</th>
<th>Gene-Wise Type II Error Rate*</th>
<th>Alignment-Wise Type II Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>16</td>
<td>1</td>
<td>1.480</td>
<td>1.488</td>
<td>0.0</td>
<td>1</td>
<td>0.92</td>
</tr>
<tr>
<td>8</td>
<td>16</td>
<td>2</td>
<td>2.258</td>
<td>2.223</td>
<td>0.0</td>
<td>0.76</td>
<td>0.58</td>
</tr>
<tr>
<td>16</td>
<td>32</td>
<td>2</td>
<td>1.962</td>
<td>1.955</td>
<td>0.0</td>
<td>0.79</td>
<td>0.56</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
<td>4</td>
<td>3.933</td>
<td>3.412</td>
<td>0.003</td>
<td>0.36</td>
<td>0.30</td>
</tr>
<tr>
<td>8</td>
<td>32</td>
<td>4</td>
<td>3.306</td>
<td>3.073</td>
<td>0.002</td>
<td>0.36</td>
<td>0.30</td>
</tr>
<tr>
<td>16</td>
<td>64</td>
<td>4</td>
<td>3.312</td>
<td>3.105</td>
<td>0</td>
<td>0.30</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>32</td>
<td>8</td>
<td>5.823</td>
<td>5.685</td>
<td>0</td>
<td>0.28</td>
<td>0.16</td>
</tr>
<tr>
<td>8</td>
<td>64</td>
<td>8</td>
<td>5.173</td>
<td>5.085</td>
<td>0.005</td>
<td>0.20</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>128</td>
<td>8</td>
<td>4.546</td>
<td>4.506</td>
<td>0.004</td>
<td>0.18</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>64</td>
<td>16</td>
<td>9.330</td>
<td>9.122</td>
<td>0.009</td>
<td>0.18</td>
<td>0.0</td>
</tr>
<tr>
<td>8</td>
<td>128</td>
<td>16</td>
<td>9.154</td>
<td>8.754</td>
<td>0.009</td>
<td>0.49</td>
<td>0.25</td>
</tr>
<tr>
<td>16</td>
<td>256</td>
<td>16</td>
<td>7.363</td>
<td>7.324</td>
<td>0</td>
<td>0.2</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>128</td>
<td>32</td>
<td>13.276</td>
<td>12.933</td>
<td>0.002</td>
<td>0.33</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>256</td>
<td>32</td>
<td>13.340</td>
<td>13.105</td>
<td>0.003</td>
<td>0.14</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>512</td>
<td>32</td>
<td>12.225</td>
<td>12.139</td>
<td>0</td>
<td>0.14</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>512</td>
<td>64</td>
<td>20.061</td>
<td>20.903</td>
<td>0.006</td>
<td>0.40</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>512</td>
<td>64</td>
<td>19.952</td>
<td>19.602</td>
<td>0.009</td>
<td>0.31</td>
<td>0.08</td>
</tr>
<tr>
<td>16</td>
<td>1,024</td>
<td>64</td>
<td>19.673</td>
<td>19.628</td>
<td>0.002</td>
<td>0.32</td>
<td>0.2</td>
</tr>
<tr>
<td>4</td>
<td>512</td>
<td>128</td>
<td>25.946</td>
<td>25.873</td>
<td>0.002</td>
<td>0.60</td>
<td>0.17</td>
</tr>
<tr>
<td>8</td>
<td>1,024</td>
<td>128</td>
<td>26.664</td>
<td>26.644</td>
<td>0</td>
<td>0.72</td>
<td>0.50</td>
</tr>
<tr>
<td>16</td>
<td>2,048</td>
<td>128</td>
<td>26.067</td>
<td>26.064</td>
<td>0</td>
<td>0.67</td>
<td>0.57</td>
</tr>
<tr>
<td>4</td>
<td>1,024</td>
<td>256</td>
<td>29.786</td>
<td>29.791</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>2,048</td>
<td>256</td>
<td>29.830</td>
<td>29.836</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>4,096</td>
<td>256</td>
<td>29.755</td>
<td>29.760</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>2,048</td>
<td>512</td>
<td>29.992</td>
<td>29.997</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>4,096</td>
<td>512</td>
<td>30.168</td>
<td>30.173</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>4,096</td>
<td>1,024</td>
<td>30.232</td>
<td>30.238</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

* The gene-wise type I error rate is defined as the number of genes falsely identified as mosaic divided by the total number of nonmosaic genes.
* The gene-wise type II error rate is defined as the number of genes falsely identified as nonmosaic divided by the total number of mosaic genes.
* The alignment-wise type II error is the proportion of data sets judged to have no recombination ($I_1 > I_2$) but do.
mutations, recombinations, and duplications and obtain a set of sequences related as if through this network. At any given time in the process, each node is considered independently and can acquire a single mutation, split into 2 sibling nodes, become recombination receptive, or do nothing. The rates of these processes are given by $k_m$, $k_s$, and $k_r$, respectively. Mutations are random and incorporated at random locations along the gene. When a node becomes recombination receptive, a partner node is chosen from among all extant internal nodes, and a recombination crossover point is chosen at random along the common lengths of the parents. A new node with the resulting mosaic sequence is added as a shared child to the 2 parent nodes.

**HIV-I env Genes**

Shankarappa et al. (1997) studied the divergence and diversity of the C2-V5 region of the HIV-1 env gene in 9 homosexual men with moderate or slow disease progression. They obtained gene sequences at an average of 12 time points per person, covering 6–12 years of infection. These data sets were used because they provided the opportunity to study the patterns of HIV-1 evolution within single individuals and 8 of these data sets were available on National Center for Biotechnology Information (GenBank accession numbers AF137629–AF137765 [patient 1], AF137767–AF137897 [patient 2], AF137898–AF138003 [patient 3], AF138004–AF138163 [patient 5], AF138166–AF138263 [patient 8]).

**FIG. 2.—** (A) Network generated by simulation. (B) Network generated by our method.
Table 3

<table>
<thead>
<tr>
<th>Patient</th>
<th>Total Sequences</th>
<th>Length</th>
<th>Mutations/Split</th>
<th>Putative Recombinants</th>
<th>P</th>
<th>$I_1$-1 (without PA)</th>
<th>$I_1$-2 (with PA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>133</td>
<td>405</td>
<td>5.2</td>
<td>8</td>
<td>0.12</td>
<td>13,701</td>
<td>6,523</td>
</tr>
<tr>
<td>2</td>
<td>129</td>
<td>381</td>
<td>5.6</td>
<td>9</td>
<td>0.046</td>
<td>7,383</td>
<td>6,448</td>
</tr>
<tr>
<td>3</td>
<td>106</td>
<td>585</td>
<td>12.8</td>
<td>0</td>
<td>NA</td>
<td>8,006</td>
<td>8,756</td>
</tr>
<tr>
<td>4</td>
<td>159</td>
<td>578</td>
<td>6.6</td>
<td>6</td>
<td>1</td>
<td>8,458</td>
<td>7,990</td>
</tr>
<tr>
<td>5</td>
<td>98</td>
<td>627</td>
<td>3.9</td>
<td>8</td>
<td>0.06</td>
<td>15,229</td>
<td>8,282</td>
</tr>
<tr>
<td>6</td>
<td>107</td>
<td>609</td>
<td>14.3</td>
<td>24</td>
<td>0.07</td>
<td>19,366</td>
<td>10,177</td>
</tr>
<tr>
<td>7</td>
<td>113</td>
<td>296</td>
<td>5.2</td>
<td>13</td>
<td>0.019</td>
<td>11,918</td>
<td>7,055</td>
</tr>
<tr>
<td>8</td>
<td>52</td>
<td>387</td>
<td>5.5</td>
<td>0</td>
<td>NA</td>
<td>3,269</td>
<td>3,274</td>
</tr>
</tbody>
</table>

Note.—$P$. Probability that the youngest putative recombinant is as old as or older than the observed.

Results

Validation on Simulated Data

For each of the combinations of $k_d/k_i$ and $k_d/k_r$ displayed in table 1, we generated 12 replicate data sets and analyzed them using the methods described above (table 2).

Gene-wise type I and type II errors, also shown in table 2, were calculated based on the number of false positives (FPs) and false negatives (FNs) in each data set. A FP was recorded in cases where $I_1$-1 (the description length for the best network without PA) was larger than $I_1$-2 (allowing PA) and there was at least 1 gene identified as mosaic. Alignment-wise type II error was calculated when a data set with mosaic genes was identified as having no recombination. Each row of the table is a summary of the replicates, each with the stated rates of recombination and duplication.

In order to determine whether the procedure not only detects the presence of recombination but also accurately infers their evolutionary histories, we compared the simulated phylogenetic networks for each data set with networks generated by our program. Figure 2 provides 1 such visual comparison of the simulated network and the network constructed using our procedure, showing that the 2 are very similar. Nine out of 12 mosaic sequences were identified as such with the appropriate ancestors and crossover points. The mosaic sequence shown in the red dashed line could not be identified because the crossover point was at position 297 of 300 and there was too little information transferred from the secondary parent for reliable identification. The mosaic sequence shown in the blue dashed line was not identified because 1 of its ancestors lost all its non-recombinant descendants from the simulation. Hence, its mosaic descendants could not be identified as such.

Note that the inferred network is not expected to be identical to the true network because the method generates multifurcating networks when there is insufficient information to resolve the pattern of bifurcations within any multifurcating node, a condition that is the norm for data sets of this size and complexity. For example, given 3 sequences AGA, AAC, and TAC, any combination of these sequences would make an equally feasible bifurcating tree. In this case, our program simply chooses a multifurcating tree because there is no one best resolution.

To estimate the alignment-wise type I error rate (for a whole alignment), we generated 15 simulated data sets with no recombination for each of the following mutation to duplication rate ratios: 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, and 1,024. There was an average of 46.9 genes per data set. Alignment-wise FPs as defined above were counted in 11 of the 165 data sets, 9 of which had identified a single gene as mosaic, whereas the other 2 had 2 and 3 mosaic genes, respectively. These mosaic genes were generated in data sets with mutation to duplication rate ratio of 4–64. The higher mutation rates led to extremely diverse data sets resulting in poor resolution. The overall alignment-wise type I error rate is thus 6.67%. The average gene-wise error rate within each of the family of data sets where alignment-wise FPs were found is less than 0.4%. Note that where $I_1$-2 is smaller than $I_1$-1 and there are no putatively mosaic genes, we can say with certainty that the annealing in stage 1 was incomplete. This shortcoming can be reduced by improving the cooling schedule.

HIV—env Gene

We ran each data set 5 times with different random initial conditions and accepted as mosaic genes those that were identified as mosaic at least 3 out of 5 times. Table 3 shows the total number of sequences and mosaic genes found in each of the data sets as well as the average description lengths for each of the data sets. Comparison of the description lengths at stages 1 and 2 (with and without PA) suggests the presence of recombination in all cases but patient 11. Patient 11 had no putative recombinants...
at all. Each of the other patients had multiple mosaic genes and substantially smaller description length in stage 2. The mutations per split for the data sets ranged from 3.9 to 14.3. This range was covered in our simulations; the results of the simulation can be taken as relevant to the analysis of the HIV data sets. Figure 3 shows the phylogenetic network of the sequences extracted from patient 2. This particular set of sequences was shown to have approximately 5.6 mutations per split and 9 putative mosaic genes. According to our simulations, the type II error for 5.6 mutations per split could be between 0.18 and 0.36 (between 4 and 8 mutations per split). Hence, we can be confident that we have identified approximately at least 64% of the recombinants. The type I error rate in this case is between 0.0 and 0.005, which is fairly low. Figure 4 shows an example of a mosaic sequence and its inferred parents. This sequence (AF137871) was isolated 126 months after seroconversion from patient 2. Table 4 shows that putative recombinants in patient 2 were amplified starting 30 months after seroconversion.

**Discussion**

We have introduced an approach to phylogenetic inference that allows PA and validated its performance on simulated data sets. We made the following observations: 1) The gene-wise type I error rate, which is defined as the number of genes falsely identified as mosaic divided by the total number of nonmosaic genes, is fairly low at all mutation rates. This number remains less than or equal to 10% for all parameter values used in testing. 2) The gene-wise type II error rate, defined as the number of genes falsely identified as nonmosaic divided by the total number of mosaic genes, is reasonable for a range of intermediate relative mutation rates (4–64 mutations per split) but becomes intolerably large for more extreme values at very low and very high rates. Failure of this type is inevitable: mutations both provide and destroy the information necessary to identify recombination. Recombination between identical or nearly genes cannot be detected by analysis of the resulting sequence, yet very high rates corrupt the signal. 3) Similarly, the alignment-wise type II error, the proportion of data sets judged to have no recombination (1) when they in fact do, is high when the mutations per branch are extreme. 4) Finally, the overall alignment-wise type I error rate, the proportion of data sets that are falsely identified to have recombination is 0.007.

In addition to testing our method on simulations, we demonstrated its application using the C2-V5 region of the env gene of HIV-1 from 8 individuals (Shankarappa et al. 1999). It is widely known that diversity in HIV-1 is generated through point mutations as well as recombinations (Robertson et al. 1995). Because recombination is active in the HIV genome, the neglect of PA in the phylogenetic analysis of HIV-1 sequences may lead to incorrect results (Schierup and Hein 2000). Our analysis detected multiple mosaic genes in all but 2 of the patients.

Further evidence that these genes really are mosaic comes from an analysis of the times at which they were sampled. Shankarappa et al. (1999) sequenced virus...
samples at each of several times from each individual—
typically from 3 months postseroconversion to 7 or more
years (the latest sampling time was more than 11 years).

We expect to find genes arising from viral recombina-
tion to appear later in the infection because detectable re-
combination requires superinfection of target cells with
genetically distinguishable viruses, the chances of which
clearly increase over time. We therefore analyzed the sam-
ples for the sets of genes identified as mosaic by de-
termining, for each patient, the earliest time giving rise to
a sample containing a mosaic gene. For the ith patient, de-
note that sampling time ti. Take the null hypothesis to be
that the genes identified as mosaic are chosen at random
from among all genes sampled. Under this null, the prob-
ability pi that the earliest sampling time in the ith patient
sample is ti or later is

\[
p_i = \frac{(N_i - k_i)! n_i(t_i)!}{N_i!(n_i(t_i) - k_i)!},
\]

where Ni is the total number of genes sampled, ki is the size
of the subsample of putatively mosaic genes, and n_i(t_i) is the
number of genes in the complete sample with sampling time
t_i or later. The probabilities computed for the data sets are
shown in table 3.

Mosaic genes are first detected in patients at an aver-
age of 37 months after seroconversion. On analyzing the sam-
ples, we found that patient 11, the slowest progressor in the study,
took significantly longer. We also
found that patient 11, the slowest progressor in the study,
showed no presence of recombination.

Table 4: Details of Putative Recombinants Identified in Patient 2

<table>
<thead>
<tr>
<th>Recombinant</th>
<th>Sampling Time</th>
<th>Break Point</th>
<th>Mutations Saved by PA</th>
</tr>
</thead>
<tbody>
<tr>
<td>AF137802</td>
<td>10 30</td>
<td>265</td>
<td>3</td>
</tr>
<tr>
<td>AF137820</td>
<td>51 51</td>
<td>213</td>
<td>5</td>
</tr>
<tr>
<td>AF137823</td>
<td>61 51</td>
<td>213</td>
<td>5</td>
</tr>
<tr>
<td>AF137836</td>
<td>73 123</td>
<td>124</td>
<td>3</td>
</tr>
<tr>
<td>AF137864</td>
<td>103 259</td>
<td>259</td>
<td>3</td>
</tr>
<tr>
<td>AF137871</td>
<td>126 202</td>
<td>202</td>
<td>6</td>
</tr>
<tr>
<td>AF137892</td>
<td>68 158</td>
<td>158</td>
<td>4</td>
</tr>
<tr>
<td>AF137897</td>
<td>68 180</td>
<td>180</td>
<td>5</td>
</tr>
</tbody>
</table>

Note.—The break point for each gene is the putative position of re-
combination.

Our method depends entirely on the use of a single
well-behaved cost function on phylogenetic networks given
the observed sequence data so that the power of numerical
minimization can be utilized. Methods based on pairwise
hypothesis testing frequentist methods such as the likeli-
hood ratio test do not provide a cost function of this kind.
Our choice of cost function, the MDL, is intended to bal-
ance the cost of lack of model fit to the data against the cost of
model complexity in a principled and natural way, by supply-
ing a common currency—information—for both. Consistency
is essential in this context because the complexity of the
underlying network changes with the number of putatively
mosaic genes. Other methods, including those of Hein
(1990), allow the cost of recombinations relative to mutations
to be fixed arbitrarily but do not determine the appropriate
value for this ratio. MDL naturally sets this ratio adaptively:
The total cost of mutations is not simply proportional to the
number of mutations; the information cost of a single mutation
decreases as the total number of mutations increases. Similarly,
the cost of a recombination event decreases as the total number
of such events increases, and the cost of either one depends on
the total number of nucleotides in the data set.

We ran several of our simulated data sets using Hein’s
Web-based recombination tool Recpars (http://www.
daimi.au.dk/~compbio/recpars/recpars.html). Under its de-
default values, we ran 10 of our simulated data sets and ob-
served a gene-wise type I error of 2% and type II error of
43%. The same data sets with our program had a 0.2%
gene-wise type I error and 28% type II error. Out of 75 neg-
avative controls (data sets with no recombinants), we found
that Recpars erroneously identified 32 data sets (42%) as
having recombination events, compared with the 9% ob-
served using our method. Moreover, the Web version of
Repars failed to give us any results for data sets with mu-
tation to duplication ratios of 32 and higher.

In addition, MDL methods, like likelihood-based
methods, provide flexibility, allowing the use of more com-
plex evolutionary models. We have here used a simple
model for evolution, because doing so gives us access to
a simple closed form for the description length for any net-
work and any data and thus greatly facilitates numerical
minimization, but this simplification does not come without
costs of its own. For example, Crandall et al. (1999) exam-
ined sequence data from 8 patients and found several posi-
tions at which parallel mutations related to drug resistance
came prevalent. Had they been isolated from a single in-
dividual, such homoplasy would erroneously be taken as

FIG. 4.—Mosaic sequence AF137871 from patient 2 shown with its parents 1010 and 1346. The vertical line at position 202 shows the putative
point of recombination such that the first half of the gene is believed to have descended from 1010 and the second half from 1346.
contributing evidence to the hypothesis of PA. A model that accounts for mutation rate heterogeneity among positions and/or selection would be required to distinguish these cases. Similarly, a model that allows for varying branch lengths will also be useful. Even though our simplified model assumes constant time between lineages, it should be noted that our simulated data are generated with exponentially distributed branch lengths. MDL can, in principle, be applied using any model for mutation and any network topology, but models allowing differing branch lengths and heterogeneous mutation rates across alignment positions will require substantially greater computation; good approximation schemes will certainly prove helpful in this regard.

Furthermore, the stochastic search methods we use are not particularly sophisticated. Substantially improved performance could likely be achieved with greater attention paid to the numerical optimization process. The computation time depends almost entirely on the stochastic search method. Using our current search parameters, we calculated the computation time for 8 data sets ranging from 20 to 100 sequences of length 300 with an average of 8 mutations/split (the mean mutations per branch for the HIV data set). The largest data set of 100 sequences ran for 115 min (real user time on a 64-bit machine, 2.19 GHz processor, 4 GB RAM), whereas the smallest data set ran for 8 min. The average time was 48 min. A better optimization scheme could lower the computation time significantly.

In spite of the oversimplifications and relatively large computational effort required, we have shown that the method as described works well, with tolerable error rates on simulated data and biologically plausible results on clinical data from HIV-infected patients.

The executables and C++ source code for the software implementation are available upon request.

Acknowledgments

We are grateful for helpful discussions with Marcy Uyenoyama and for the financial support of the Bill and Melinda Gates Foundation through grant number 38643 to Dr Barton Haynes. We would also like to thank the 2 anonymous reviewers for their insightful comments.

Note Added during Proof Stage

We have recently discovered that Ren et al. (1995) developed a method for phylogenetic inference based on the minimum description length principle over ten years ago. Their approach is very similar to the one we have described here, though ours extends the core ideas to address the inference of more general phylogenetic networks.

Literature Cited


Naruya Saitou, Associate Editor

Accepted March 16, 2008