Median-Joining Networks for Inferring Intraspecific Phylogenies

Hans-Jürgen Bandelt, Peter Forster, and Arne Röhl
Mathematisches Seminar, Universität Hamburg, Hamburg, Germany

Reconstructing phylogenies from intraspecific data (such as human mitochondrial DNA variation) is often a challenging task because of large sample sizes and small genetic distances between individuals. The resulting multitude of plausible trees is best expressed by a network which displays alternative potential evolutionary paths in the form of cycles. We present a method (‘‘median joining’’ [MJ]) for constructing networks from recombination-free population data that combines features of Kruskal's algorithm for finding minimum spanning trees by favoring short connections, and Farris's maximum-parsimony (MP) heuristic algorithm, which sequentially adds new vertices called ‘‘median vectors’’, except that our MJ method does not resolve ties. The MJ method is hence closely related to the earlier approach of Foulds, Hendy, and Penny for estimating MP trees but can be adjusted to the level of homoplasy by setting a parameter ε. Unlike our earlier reduced median (RM) network method, MJ is applicable to multistate characters (e.g., amino acid sequences). An additional feature is the speed of the implemented algorithm: a sample of 800 worldwide mtDNA hypervariable segment I sequences requires less than 3 h on a Pentium 120 PC. The MJ method is demonstrated on a Tibetan mitochondrial DNA RFLP data set.

Introduction

The phylogenetic median-joining (MJ) network algorithm which we present here offers new features compared with our previous reduced median (RM) network algorithm (Bandelt et al. 1995) in that it can handle larger sets of genetic data, as well as multistate data such as amino acid sequences. The MJ method begins with the minimum spanning trees, all combined within a single (reticulate) network. Aiming at parsimony, we subsequently add a few consensus sequences (i.e., median vectors, or Steiner points) of three mutually close sequences at a time. These median vectors can be biologically interpreted as possibly extant unsampled sequences or extinct ancestral sequences. The median operation, also referred to as ‘‘Steinerization’’ in mathematics (in which the most parsimonious realizations of MP trees are called Steiner trees; see Hwang, Richards, and Winter 1992), is basic to all fast MP heuristic algorithms, although it is typically applied in a very restricted (‘‘greedy’’) manner in order to arrive at a single tree (Farris 1970). In contrast, the unconstrained use of the median operation eventually generates the so-called full quasimedian network (known as the full median network in the case of binary data), which normally harbors all optimal trees, as well as numerous suboptimal trees. This quasimedian network is in general too complex for visualization or even too large for storage in a computer. With MJ, we take care that at each stage only those median vectors which have a good chance of appearing as branching nodes in an MP tree are generated by considering only triplets of sequences for which one sequence is linked to the other two in the network under processing. An additional ranking of these candidate triplets according to a distance score (as proposed by Tateno 1990) allows further refinement of the triplet selection. After each round of median generation, the process restarts with the thus enlarged set of sequences.

This approach, then, is quite similar to that of Foulds, Hendy, and Penny (1979), which, unfortunately, seems to have been rather forgotten in the field of biology after tree-building program packages became widely available. The major differences between the two methods are as follows: the criterion of selecting triplets for median generation is different; MJ stops earlier (with a postprocessing phase being optional, see below); and MJ is more generous and flexible in that it uses an explicit parameter, ε, fuzzifying the employed distance measure, with the effect that by increasing ε, MJ produces more median vectors simultaneously at each stage. In the illustrative example using Tibetan mtDNA, we compare the different network methods and show how the network analysis guides the informed choice of a single tree estimate in this particular case.

Minimum Spanning Networks

A minimum spanning tree for a set of sequence types connects all given types without creating any cycles or inferring additional (ancestral) nodes, such that the total length (i.e., the sum of distances between linked sequence types) is minimal. Kruskal's (1956) algorithm quickly finds one minimum spanning tree: in a preliminary step the pairs of sequence types are listed in increasing order of their distances (ordering of the pairs with the same distance is arbitrary and serves as a tie-breaking rule); then, the tree is built up by successively selecting the first link from the preference list which does not create a cycle together with the already chosen links.

A simple modification of this algorithm (namely, dropping the tie-breaking rule), allows one to construct the union of all minimum spanning trees, which we will call (by a slight abuse of language) the minimum spanning network. (This construction is completely analogous to that proposed by Excoffier and Smouse [1994], who constructed this network by departing from the algorithm of Prim [1957]). Assume that there are k distinct distance values, $\delta_1 < \delta_2 < \ldots < \delta_k$, between the se-

Key words: phylogeny construction, networks, parsimony, human mitochondrial DNA.

Address for correspondence and reprints: Hans-Jürgen Bandelt, Mathematisches Seminar, Universität Hamburg, Bundesstrasse 55, D-20146 Hamburg, Germany.

Mol. Biol. Evol. 16(1):37–48. 1999

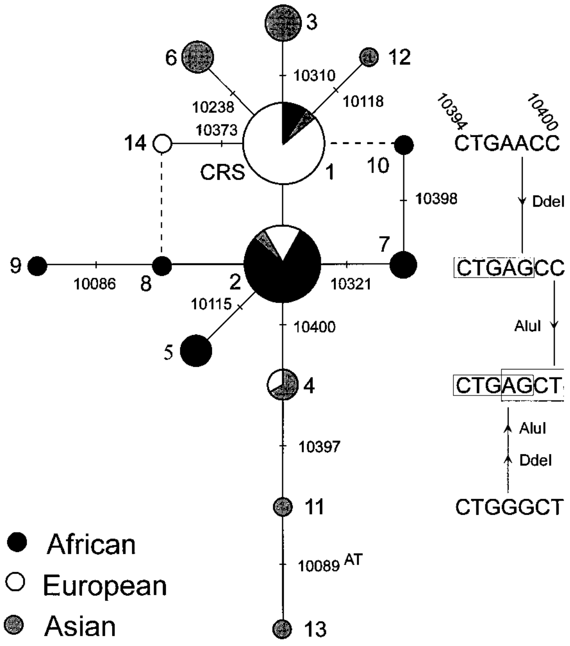

FIG. 1.—Minimum spanning network displaying the mitochondrial *ND3* variation in 61 Africans, Asians, and Europeans (data from Nachman et al. 1996, table 1). This one-step network is identical to the MJ and RM networks and contains all 15 MP trees. Circles (nodes) represent sequence types, numbered from 1 to 14; areas of (and within) circles are proportional to the number of sampled individuals. Transitions are referred to by the nucleotide positions (numbered as in Anderson et al. 1981); parallel lines in a reticulation represent mutations at identical nucleotide positions and are thus labeled only once; superscript refers to transversions. Sequence type 1 represents the Cambridge reference sequence. At the right of the figure, gains and losses (represented by arrows) of restriction sites 10394*Dde*I and 10397*Alu*I are explained. There is a single candidate for a most likely tree (indicated by unbroken lines) when geographical information is taken into account. First, it is likely that the link between type 8 (African) and type 14 (European) can be dropped, as suggested by Nachman et al. (1996). Second, the African mtDNA tree of Chen et al. (1995) includes several restriction sites within *ND3*, which indicates that sequence types 7 and 10 from Nachman et al. (1996) occur among the Biaka (West Pygmies), and thus favors the paths from type 2 to type 1 and from type 2 via type 7 to type 10. The branching and frequency pattern of the resulting *ND3* tree suggests that sequence type 2 is the root of the tree.

quences under consideration. We proceed in increasing order of these values. At the beginning, no pair of sequence types is linked. For the recursive step, assume that the next value to be processed is δ_i, and the network constructed so far is not yet connected, that is, it comprises several connected subnetworks (its components). Then, add links between all sequence types from different connected components that are at distance δ_i. If the resulting network is connected, the algorithm stops; if not, it continues with the next value, δ_{i+1}. The proof that the minimum spanning network is constructed in this way is deferred to appendix 1. Easy to compute, the minimum spanning network is of little direct use for representing genetic data, since in general a minimum spanning tree is far from being most parsimonious. It serves, however, as a good point of departure in each recursive step of our MJ network construction for generating additional inferred sequence types which reduce tree length.

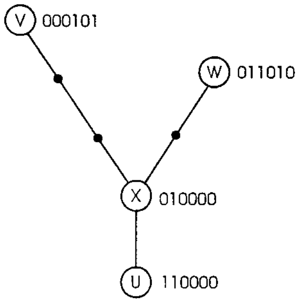

FIG. 2.—The median vector *X* of binary sequences *U*, *V*, and *W*.

There are, however, rare cases in which the minimum spanning network is a connected one-step network and therefore provably contains all MP trees. In such a case, the connected network consists of exactly those links between sequence types which differ in only one character (e.g., nucleotide position or restriction site). This case can arise when the resolution of the employed characters is fairly low and the sampling of the population is sufficiently exhaustive; for example, the human *ND3* data set published by Nachman et al. (1996) can be represented by a connected one-step network (fig. 1). This network has two cycles sharing a link which give rise to 15 MP trees (rather than 27, as claimed in Nachman et al. 1996).

We can conceive of obtaining the minimum spanning network by departing from the complete network in which any two sequence types are connected by a link with a length equal to the respective distance and then deleting links sequentially as follows: First, order the links from maximal to minimal length. Processing links in this order, check for each step whether the link under consideration joins two nodes which are connected by a path comprising only shorter links; whenever this is the case, the processed link is deleted. This procedure may be shortened in updating a network. For example, if a minimum spanning network is enlarged by adding a single node with incident links, then one needs to screen only the subnetwork formed by the new cycles containing the added node (cf. Foulds, Hendy, and Penny 1979).

Median Vectors and Quasimedian Networks

For three aligned sequences, *U*, *V*, and *W*, there is just one tree (the ‘‘star’’) connecting them. If all characters corresponding to the sequence positions are binary, then this tree has a single most parsimonious reconstruction (MPR; see Swofford et al. 1996); namely, the sequence that is assigned to the interior node in order to realize minimum tree length is necessarily the median vector *X* obtained by majority consensus (fig. 2). If, however, *m* of the characters have three different states in *U*, *V*, and *W*, then there exist 3^m distinct MPRs. Three of them are distinguished by minimizing the length of one link (at the expense of the two others); for instance, the sequence assigned to the interior node in an MPR

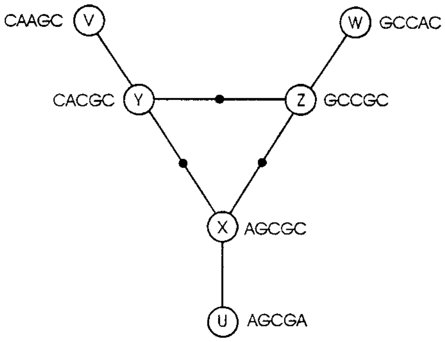

FIG. 3.—The median vectors *X, Y,* and *Z* of sequences *U, V,* and *W.*

which is closest to *U* is sequence *X*, where for each two-state character, the majority state is taken, and for each three-state character, the state of *U* is taken. Similarly, sequences *Y* and *Z* which minimize the distances to *V* and *W*, respectively, are defined (see fig. 3 for an example). The three sequences *X, Y,* and *Z* are called the ''median vectors'' of *U, V,* and *W*. The network composed of the equilateral triangle with nodes *X, Y,* and *Z* and three further links joining it with the sampled sequences *U, V,* and *W* then represents the distances between *U, V,* and *W.*

Given any sample comprising *n* sequences, one may successively add median vectors as follows: At each step, select any three sequences *U, V,* and *W* from the pool of sequences sampled or created so far, and add the median vector(s) of *U, V,* and *W* to the pool. Continue until no further new sequences can be generated. The terminal pool of sequences can be organized as a network in which two sequences *U* and *V* are linked exactly when there is no third sequence *X* between them (i.e., *X* is the median vector of *U, V,* and *X*). This network is called the (full) quasimedian network generated by the sampled sequences. When all characters are binary, this network coincides with the (full) median network described by Bandelt et al. (1995). Mathematical properties of quasimedian networks have been studied and surveyed by Bandelt, Mulder, and Wilkeit (1994). In practice, the quasimedian network generated by the given data may be somewhat large due to homoplasy, such that only a portion should be heuristically constructed by carefully selecting triplets of sequences for median generation.

Median Joining

Prerequisites

The input data for our network algorithm comprise correctly aligned sequences of a population sample. It is stipulated that ambiguous states are infrequent and recombination is absent. These requirements are met for published human mtDNA RFLP and control region sequences as well as Y-chromosomal short tandem repeat variation when a single-repeat mutation model is assumed.

Distances

The simplest way to obtain a distance measure between two sequences is to count the number of character differences (the ''Hamming distance''). As a refinement, we may also weight character changes, albeit only in a symmetrical fashion (i.e., giving both directions of change between two states the same weight). The ''weighted'' Hamming distance between two sequence types is then the sum of weighted differences.

Ambiguous States

Prior to network construction, missing or ambiguous states in the sample are treated in the calculation of distances as follows. An ambiguous state *X* is equated with the set of states that specify this ambiguous state. For instance, $X = \mathrm{R}$ (purine) would be equated with the {A, G} pair of nucleotides, whereas $X = \mathrm{N}$ would mean the set of all nucleotides, {A, G, C, T}. The weighted distance between such sets of character states is equated with the minimum (weighted) difference between the states from those sets; for example, in the unweighted case, the difference between R and N is 0, but that between R and T (thymine) is 1. Ambiguities in the character string of a sequence type are then specified at the initial stage of the network construction, which is greedily realized by comparing the ambiguous states in each sequence with the definite states of the other minimally distant sequences (with respect to the above distance measure). An ambiguous state will be assigned to the setting of the most common definite state of these sequences (ties being broken arbitrarily).

The Algorithm

For the algorithm, we specify a tolerance ε up to which we wish not to distinguish between distances. Increasing the parameter ε widens the search for potential new median vectors (incurred by the choice of links in step 2, below) and also relaxes a distance criterion (step 4). (One could replace the single parameter ε by a pair of parameters governing the two steps separately, but we choose not to do this here.) At each stage, the algorithm constructs an initial part of the minimum spanning network (described by the ''feasible'' links) for the current sequence types (i.e., sequence types under processing), or, in the case in which ε is set >0, the ''ε-relaxed'' minimum spanning network, which contains additional feasible links (step 2). Triplets of sequence types are admitted to median generation only if there are at least two feasible links among them, but only those median vectors are actually generated and added to the current pool of sequence types for which the total distance to the corresponding triplet attains the minimum value plus ε at most (step 4). The whole process is iterated until no further median vectors can be generated following these rules. Some intermediate purging of obsolete sequence types may be necessary (step 3). The final network is then the minimum spanning network of the expanded set of sequence types (step 5).

Phase I: Successive selection of median vectors.

Initialization: Specify $\varepsilon \geq 0$. The current sequence types comprise the sampled sequence types.

Step 1: Determine the distance matrix d for the current sequence types, pool identical sequence types, and order the different distance values as $\delta_1 < \delta_2 < \ldots < \delta_k$.

Step 2: Determine the links between sequence types which describe the (ε-relaxed) minimum spanning network, i.e., which are feasible with regard to ε in the following sense: two sequence types V and W are feasibly linked if there is no path from V to W consisting of sequence types $V = U_0, U_1, \ldots, U_k = W$ which fulfil the inequality $d(U_i, U_{i+1}) < d(V, W) - \varepsilon$ for all $i = 0, \ldots, k - 1$. Thus, V and W with $d(V, W) = \delta_j$ form a feasible link if either $\delta_j - \varepsilon \leq \delta_1$ or V and W belong to different connected components of the ‘‘threshold’’ subnetwork in which sequence types are linked exactly when their distance does not exceed δ_i, where i is the largest index with $\delta_i < \delta_j - \varepsilon$.

Step 3: Iteratively remove from the set of current sequence types those (obsolete) sequence types which are not among the sampled sequences but are feasibly linked to at most two current sequence types. If obsolete types were detected, go back to step 2; else continue.

Step 4: Determine the feasible triplets U, V, and W of sequence types, which are defined as follows: at least two pairs from U, V, and W are feasibly linked, and at least one median vector X of U, V, and W is not yet a current sequence type. If there are no feasible triplets at all, then continue with step 5. Otherwise, compute the connection cost $d(U, X) + d(V, X) + d(W, X)$ of the median vectors X for each feasible triplet U, V, and W. This value constitutes the length of MP trees connecting U, V, and W. Compute the minimum connection cost λ for all feasible triplets U, V, and W. Now, generate all median vectors X of feasible triplets for which the connection costs do not exceed $\lambda + \varepsilon$. Expand the set of current sequence types with these new median vectors. Go back to step 1.

Phase II: Construction of the final network.

Step 5: Calculate the minimum spanning network for the new set of current sequence types. This can be accomplished by performing a pass through step 3 with parameter ε set to zero (so that only minimum length connections are taken into account); then, the feasible links with regard to $\varepsilon = 0$ yield the minimum spanning network. If obsolete median vectors are present, remove these and repeat step 5. If not, these feasible links describe the final network.

The construction ensures that every link between two sequence types V and W in the final network has the same length as any shortest path between V and W in the sequence space endowed with the (weighted) Hamming distance. The segment bounded by V and W in this space consists of all possible sequences Z between V and W, that is, lying on shortest paths between V and W in the space or, equivalently, satisfying $d(V, Z) + d(W, Z) = d(V, W)$. Two segments in the sequence space bounded by pairs V, W and X, Y of sequence types that are linked in the final network never intersect whenever V and W are distinct from X and Y, that is, there is no possible sequence type Z between V and W as well as between X and Y (see appendix 2 for a proof). We can therefore select any of the shortest paths from the sequence space in order to connect linked sequence types without creating any internal node twice. A simple consequence of this observation is that for binary data free of homoplasy, the unique most parsimonious tree representing them is reconstructed by this network method.

Table 1
Four Sequence Types, *A*, *B*, *C*, and *D*, Defined by Five Weighted Binary Characters

Sequence type	Characters	Distances			
		A	*B*	*C*	*D*
A	0 0 0 0 0		4	4	7
B	1 1 0 0 0			6	5
C	1 0 1 1 0				7
D	0 1 1 0 1				
Weights . . .	1 3 2 1 2				

Illustrations

Example 1

In order to show the MJ algorithm at work, we consider an artificial data set consisting of four sequence types, A, B, C, and D, defined by five weighted binary characters (see table 1, which also displays the weighted Hamming distances). There are four different distances within the initial data set: $\delta_1 = 4$, $\delta_2 = 5$, $\delta_3 = 6$, and $\delta_4 = 7$. We now determine the δ-step components, that is, the connected components of the subnetwork in which pairs of sequences are linked when their distance does not exceed δ, for each choice of δ from δ_1, δ_2, and δ_3. At $\delta = \delta_1 = 4$, we can link sequence A to both sequence B and sequence C, thus obtaining the two four-step components $\{A, B, C\}$ and $\{D\}$. At $\delta = 5$, we can also link sequences B and D so that a single five-step component $\{A, B, C, D\}$ arises, thus ending the search for δ-step components. Therefore, the three pairs A, B and A, C and B, D are feasibly linked for every choice of ε. Thus, A, B, C and A, B, D constitute feasible triplets, from which median vectors $U = 10000$ and $V = 01000$ can be generated at connection costs 7 and 8. Now, the process depends on the actual setting of the parameter ε.

We begin with $\varepsilon = 0$. Then, only the median vector U is generated at minimum cost $\lambda = 7$ in the first round. The distances from U to A, B, C, and D equal 1, 3, 3, and 8, respectively. The new distance values of the expanded set of sequences are thus $\delta_1 = 1$, $\delta_2 = 3$, $\delta_3 = 4$, $\delta_4 = 5$, $\delta_5 = 6$, $\delta_6 = 7$, and $\delta_7 = 8$. The one-step components are $\{A, U\}$, $\{B\}$, $\{C\}$, and $\{D\}$, and the three-step components are $\{A, B, C, U\}$ and $\{D\}$, which are also the four-step components. These are then merged into the single five-step component. We thus have feasible links from A to U, B to D, B to U, and C to U. Notice that the pairs A, B and A, C are not joined by a feasible link because they are at distance 4 from each other and belong to a common δ-step component

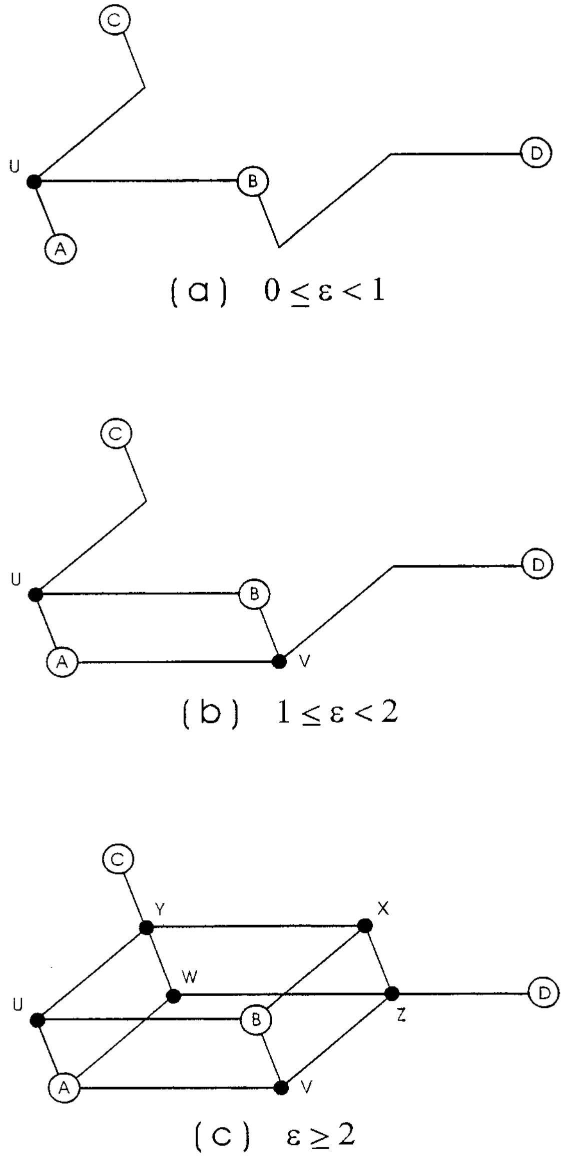

FIG. 4.—MJ networks (drawn to scale) constructed from the data of table 1 with three different settings (*a–c*) of the parameter ε; inferred sequence types *U, V, W, X, Y,* and *Z* are added to the growing network as median vectors.

Table 2
Six Sequence Types, A_1, A_2, B_1, B_2, C, and D, Defined by Six Binary Characters and One Ternary Character

SEQUENCE TYPES	CHARACTERS							DISTANCES					
								A_1	A_2	B_1	B_2	C	D
A_1	G	A	A	A	A	A	A		3	4	5	5	5
A_2	A	G	A	A	A	A	A			5	6	6	6
B_1	A	A	G	A	A	A	G				3	5	7
B_2	A	A	A	G	A	A	G					6	8
C	A	A	A	A	A	G	C						4
D	A	A	A	A	G	G	A						
Weights ..	1	2	1	2	2	2	2						

at δ < 4. Obtaining no further feasible links, the algorithm stops (as triplets *A, B, U* and *A, C, U* and *B, C, U* and *B, D, U* generate only median vectors which are among the current sequence types). The former feasible links yield a connected network which describes an MP tree.

Setting the parameter ε to 1, the starting situation is exactly the same as with ε = 0, except that we must now check all pairs of sequences which do not exceed distance 5 + ε = 6 for feasible linkage. Pair *B, C* must be checked also, but it does not constitute an additional feasible link, since sequences *B* and *C* belong to a common δ-step component at δ < 6 − ε = 5. We then obtain the same feasible links, feasible triplets, and minimum value for λ as for ε = 0. Now, we have to generate all median vectors of feasible triplets not exceeding connection cost λ + ε = 7 + 1 = 8. Thus, for ε = 1, both median vectors *U* and *V* are generated. The distances from *V* to *A, B, C, D,* and *U* equal 3, 1, 7, 4, and 4, respectively. The one-step components are now {*A, U*}, {*B, V*}, {*C*}, and {*D*}; the three-step components are {*A, B, C, U, V*} and {*D*}; and for δ ≥ 4, all six sequences are within one δ-step component {*A, B, C, D, U, V*}. In the current set of six sequence types, feasible links connect *A, B, U,* and *V* among each other; furthermore, they connect *C* to *A* and *U* as well as *D* to *B* and *V*. Since no feasible triplets arise, the algorithm terminates with the network of figure 4*b*.

The final setting, ε = 2, will eventually yield the full median network generated by the data matrix, thus displaying the full homoplasy of this data set. At the outset, all links between the given sequence types are feasible. Then, all triplets are feasible. The median vectors *U, V, W* (00100), and *X* (11100) are generated at connection costs not exceeding 7 + 2 = 9. In the expanded set of sequence types, the triplets *C, U, W* and *D, V, X* are feasible, producing median vectors *Y* (10100) and *Z* (01100) at connection costs 4 and 5, respectively. The resulting 10 sequence types yield the final network shown in figure 4*c*. This constitutes a median network, since the median vectors for all 120 triplets of distinct sequence types are already found among the 10 sequence types.

Example 2

To demonstrate the MJ method with multistate characters, consider the following artificial data set comprising six sequences, A_1, A_2, B_1, B_2, *C,* and *D,* with seven weighted positions (see table 2). These sequences generate the quasimedian network displayed in figure 5, in which the prism signifies the incompatibility of the single ternary character with one binary character. MJ with ε = 2 or larger retrieves this network. When setting ε = 1 instead, three median vectors are generated in the first round: *V* from the triplet A_1, A_2, B_1, and *W* from A_1, B_1, B_2, both at connection cost 6, as well as *X* from A_1, *C, D* at cost 7; no further median vectors are generated, so MJ terminates with the unique MP tree for these data. In contrast, MJ with ε = 0 first generates *V* and *W,* and then, in the second round, it adds both *X* and *Y* at connection cost 6. We thus have the seemingly paradoxical situation (although it is rarely seen with real data) that the MJ network may shrink when passing from ε = 0 to ε = 1.

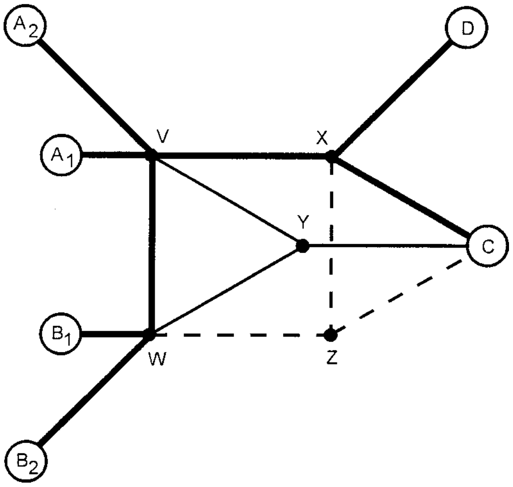

FIG. 5.—MJ network with $\varepsilon = 2$ constructed from the data of table 2. The unique MP tree is indicated with bold lines and coincides with the MJ network for $\varepsilon = 1$. Unbroken lines constitute the MJ network for $\varepsilon = 0$.

Comparison with the Method of Foulds, Hendy, and Penny (1979)

The approach taken by Foulds, Hendy, and Penny (1979) to exactly determine the MP trees for data sets (of small size) entails a heuristic network method, which is comparable to our MJ algorithm with parameter $\varepsilon = 0$, albeit with some differences. Steps 1 and 2 are performed the same, whereas step 3 (elimination of obsolete sequence types) is not explicitly mentioned by those authors but would clearly fit their strategy. The selection of median vectors in step 4 to be added to the growing network is quite different. To describe this, consider pairs *U, V* and *U, W* of feasibly linked sequence types such that the median vector *X* (for the triplet *U, V, W*) nearest to *U* is different from *U* (as in figs. 2 and 3); say that *X* is within the search radius max($d(U, V)$, $d(U, W)$) and yields the positive profit $d(U, X)$. Now, create those median vectors *X*, maximizing the profit within the smallest possible search radius in which new median vectors would arise, expand the set of current sequence types by these created median vectors, and otherwise proceed with step 5 of MJ. We refer to the resulting network as the ‘‘greedy FHP network’’ (prior to further processing).

The complete approach of Foulds, Hendy, and Penny (1979) is somewhat difficult to compare with the MJ algorithm for $\varepsilon = 0$, since the former employs the explicit comparison of the length of the shortest trees connecting the sampled sequences within the network with a calculated lower bound for the length of MP trees. This comparison, however, cannot effectively be realized for very large data sets. We deliberately interpret their approach as consisting of two phases: an initial phase, essentially yielding the greedy FHP network, and a final phase at which the network may further grow in order to capture additional putative MPRs. Specifically, each pair *U, V* of sampled sequences for which the length of a shortest path in the current network exceeds the input distance is processed separately: add an artificial feasible link between *U* and *V*, and seek to create new median vectors by reiterating the previous phase. By way of illustration, consider the greedy FHP network in example 1, given by the MP tree of figure 4*a*. In this tree, the pairs *A, D* and *C, D* do not have their input distances realized because of homoplasy. When executing *A, D* first, we would create the extra link between *A* and *D* and thereby force the triplet *A, B, D* to become feasible, from which *V* is obtained as a median vector. This leads to the network shown in figure 4*b*. Introducing the extra link between *C* and *D* to this network gives rise to the feasible triplets *C, D, U* and *C, D, V*, from which *Y* and *Z* are generated as median vectors. Both vectors, however, are subsequently removed as being obsolete, at which point the procedure terminates. On the other hand, when executing the pair *C, D* before *A, D*, the forced link between *C* and *D* yields the feasible triplets *C, D, U* and *B, C, D*, from which *Y* and *X* are now generated. The two median vectors become linked in the minimum spanning network for *A, B, C, D, U, X*, and *Y* and thus survive the purging of obsolete sequences. The pair *A, D* then still needs treatment: adding an extra link between *A* and *D* produces the median vector *Z* from *A, D, X*, which, however, remains obsolete and is removed. We thus see that one may obtain different networks in the final phase, depending on the order in which the pairs are processed. Even if we took the subnetwork comprising all temporarily constructed sequences *V, X, Y*, and *Z* together with *A, B, C, D*, and *U*, we would still dismiss those most-parsimonious reconstructions which include the node *W*.

Case Study

The effectiveness of network analyses (along with appropriate weighting) for refining our understanding of human mtDNA evolution will now be demonstrated with the Tibetan RFLP data from Torroni et al. (1994). Variation at the two restriction sites 10394*Dde*I and 10397*Alu*I in the human mitochondrial genome defines the deepest currently known phylogenetic split within non-African mtDNA, distinguishing supergroup M (both sites present; rare in Europeans but frequent in Asians) from the other supergroup (both sites absent; frequent in both Europeans and Asians). However, because the two recognition sites overlap, it is conceivable that a mutation in the overlap of the two sites may occasionally cause a secondary loss of both sites. This occurrence has already been postulated by Torroni et al. (1993) and can now be confirmed with the *ND3* data from Nachman et al. (1996) (see fig. 1; the double loss is induced by a mutation at nucleotide position [np] 10397). Two out of five Eurasians with T at np 10400 (characterizing the supergroup M; see Torroni et al. 1996) have the np 10397 mutation, which indicates that sequencing of np’s 10397, 10398, and 10400 (cf. fig. 1) rather than RFLP analysis is required to identify M membership reliably.

Single-hit losses of the overlapping sites 10394*Dde*I/10397*Alu*I would be most disturbing for phylogenetic analyses of Asian mtDNA RFLP data if we did not downweight these sites. In the Tibetan RFLP data from Torroni et al. (1994), which we reanalyze ignoring the unreliable site 16517*Hae*III (Chen et al. 1995), no MP tree exists which would recognize this double loss as one event in sequence types 131 and 135 within mtDNA group D. There are, however, such MP trees once sites 10394*Dde*I and 10397*Alu*I are each weighted by 1/2; employing this weighting, the greedy FHP network comprises exactly the 4 × 4 × 15 = 240 different most-parsimonious realizations of all MP trees for this data. MJ with $\varepsilon = 0$ offers one additional median vector, namely that for the triplet 146, 147, 62, along with three additional links (dotted in fig. 6). The RM network, in contrast, does not incorporate all MP trees, as a result of the hypothesis of a parallel event at 5259*Ava*II/5261*Hae*III rather than at 12406*Hpa*I/*Hinc*II. The MJ network with ε set to 1 (not shown) would embrace this alternative as well.

A most plausible MP solution (highlighted in fig. 6) is obtained by taking into account mtDNA type frequencies and downweighting highly mutable sites: double losses of sites 10394*Dde*I/10397*Alu*I are quite likely, and all MP trees suggest that both 15925*Hpa*II and 1667*Dde*I/1670*Alu*I have experienced at least two independent mutations. This tree agrees quite well with the tree presented by Torroni et al. (1994), even though they incorporated the unstable site 16517*Hae*III in their analysis. Note that, departing from our tree, it requires no fewer than six mutations to explain the distribution of site 16517*Hae*III, whereas all other sites are estimated to have experienced three or fewer mutations (see also Forster et al. 1997). A notable improvement to the tree presented by Torroni et al. (1994) is our proposed phylogeny for mtDNA group F, which suggests that 16303*Rsa*I (corresponding to np 16304) is a useful control region marker for group F.

Setting of the Parameter ε

Generalizing from the Tibetan case study and others not recorded here, we recommend running MJ with $\varepsilon = 0$ for human mtDNA RFLP data, provided that notoriously noisy sites in the control region (e.g., 16310*Rsa*I, 16517*Hae*III) are downweighted. In any case, postprocessing (described below) is encouraged, as is a trial with $\varepsilon = 1$ to explore the homoplasy of the data. For the more homoplasious human mtDNA control region sequences, differential weighting of sites is even more important, and it is advisable to compare the $\varepsilon = 0$ network with the $\varepsilon = 1$ network, to employ only frequent sequence types, or even to resort to a hybrid approach (see below). In general, the longer the maximum length of links and the sparser the sampling, the higher the setting of ε should be. These recommendations are valid for a data set with a weight of 1 for most characters. For example, a choice of $\varepsilon = 4$ has the same effect as $\varepsilon = 0$ if all weights of characters are multiples of 5. Therefore, increments of ε are only effective when scaled to increments in the distance matrix of sequences.

Implementation

Data Reduction

In view of the time the algorithm will take to calculate the network, it is advisable to reduce the data set to a minimum. It is routine to first group identical sequences into sequence types for which the sample frequency is recorded. Moreover, all sequence positions which are unvaried in the data set are eliminated. Second, we can take a look ahead and predict which small ‘‘peripheral’’ one-step subnetworks (such as pendant one-step subtrees) will inevitably show up in the resulting MJ network, no matter how the parameter ε is chosen. These peripheral subnetworks can be constructed separately, so that the algorithm will run on a smaller data set, producing a smaller MJ network to which the separately erected ‘‘extremities’’ are attached later. For instance, every sequence type linked to exactly one other sequence type in the full quasimedian network may be eliminated. To identify such a type Z in the data set, we check whether there exists some type W such that the characters distinguishing Z from W are binary (uninformative) characters for which all sequence types different from Z share the same state. We then remove Z from the further-processed sample of types and continue the search. Moreover, we will erase any four-cycle of sampled sequences, which would be peripheral in the full quasimedian network, by deleting its linked pair of types not linked to any type outside of this four-cycle. To this end, we check whether there exist four sequence types W, X, Y, and Z, distinguished by only two binary characters, α and ω, such that Y and Z have one state with respect to ω, and all remaining types (in the processed sample) have the other state, while α distinguishes W, Z from X, Y. Then, W, X, Y, and Z form a peripheral four-cycle attached to the linked types W and X; remove Y and Z from the processed sample, and continue until no further peeling is possible. Note that the calculation of median vectors in the processed sample for the algorithm is not affected by the removal of these peripheral parts. All eliminated sequences are, of course, resurrected in the final network display.

Running Times

We have implemented the MJ algorithm with the data reduction as described, accepting character weights and any choice of ε. The program Network 1.5 (Röhl 1997) accepts up to 9,000 different sequences distinguished at up to 252 characters. The resulting network is graphically displayed and is also described by a list of links.

To demonstrate the efficiency of the MJ method, we tested a worldwide sample comprising 2,055 human mtDNA control region sequences (hypervariable segment I, with 199 varied positions). After pooling identical sequences, there are 1,272 different sequences in this sample. The data reduction further eliminates 14 sequences and 14 positions before executing the core algorithm. Positions are weighted uniformly and the parameter ε is set to 0. The algorithm takes 25 rounds for median generation. The final network has 548 median vectors (which are not among the sampled sequences). The running time on a personal computer (IBM Cyrix 6X86 150+, similar in speed to a Pentium 120) was 26 h and 21 min. If we limit the reduction process to the pooling of identical sequences (so that we have 1,272

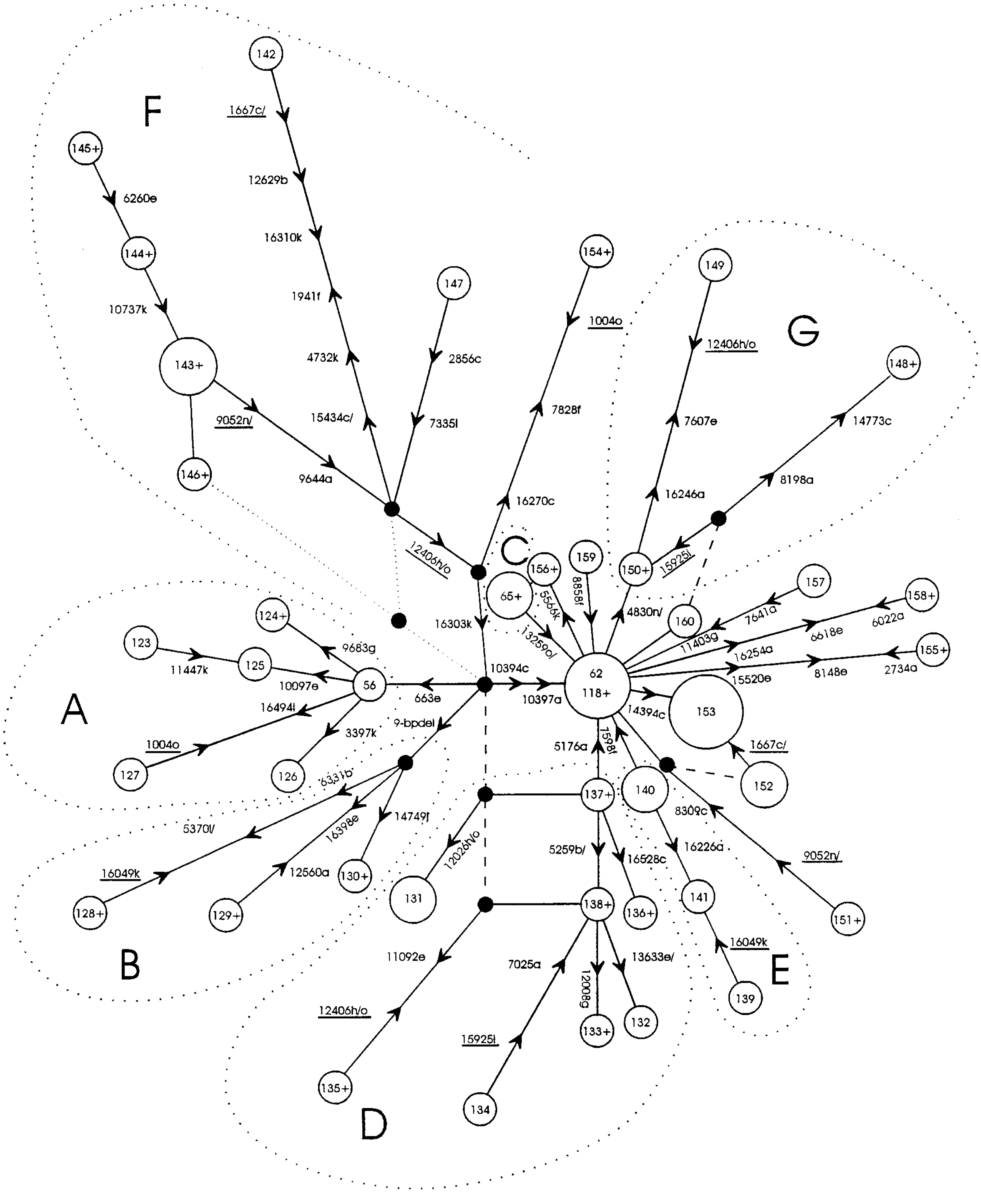

FIG. 6.—MJ network (ε = 0) for Tibetans, based on mtDNA restriction sites (Torroni et al. 1994; sites 10394*Dde*I and 10397*Alu*I are weighted 1/2; site 16517*Hae*III is disregarded, but its presence is indicated by "+" accompanying the sequence types; an error in the original table concerning the status of 10394*Dde*I/10397*Alu*I in mtDNA type 118 has been corrected). Designation of sequence types, restriction sites, and mtDNA groups A to G accords with Torroni et al (1994). A slash indicates that a site is recognized by more than one restriction enzyme, and underlining denotes resolved recurrent mutations. Arrows point to presence of restriction sites; areas of circles are proportional to the numbers of sampled individuals. The dotted links do not appear in any MP trees; the unbroken links indicate a plausible MP tree (see text).

Table 3
Results with Data Reduction for Sequences Randomly Drawn from 2,055 mtDNA Control Region Sequences (hypervariable segment I)

Number of Input Sequences	Number of Eliminated Sequences	Running Time for Data Reduction (s)
300	67	34
350	79	45
400	106	51
450	103	57
500	124	82
550	150	80
600	166	101
650	192	155
700	289	137
750	233	219
800	229	183

Table 4
Results for the MJ Algorithm Applied to the Reduced Data Sets of Table 3

Number of Input Sequences	Number of Varied Positions	Number of Generated Median Vectors	Running Time (s)
223	128	144	943
271	120	138	1,587
294	127	195	1,240
347	137	182	3,221
376	130	232	4,652
400	143	181	4,508
434	145	207	6,361
458	144	230	7,193
511	152	238	8,812
517	146	195	7,833
571	153	260	10,420

different sequences of length 199), then we get a running time of 28 h—an increase of 6.3%. In contrast, the data reduction takes only 0.7% of the total running time. It thus pays to perform the reduction.

To obtain a rough estimate of the average complexity of the algorithm, we randomly drew 300–800 sequences from the above data set and subjected them to the algorithm. Table 3 lists the numbers of sequences employed, along with the numbers of varied sequence positions, the numbers of sequences eliminated in the data reduction phase, and the time needed for this. The reduced data sets in each case are then executed by the algorithm, and the running times are recorded (see table 4). For estimating the time complexity of the algorithm, we measure the amount of the input (reduced) data by counting the total number m of entries in the aligned sequences (i.e., multiplying the numbers of input sequences and of varied positions), and then assume a functional relationship of computing time t and size m of the form $t \approx \alpha m^{\beta}$. With a least-squares approach applied after taking logarithms, we estimate from table 4 that $\alpha \approx 1.6 \times 10^{-7}$ s and $\beta \approx 2.2$. Applying this formula to the complete data set after data reduction (i.e., 1,258 sequences of length 185), we would expect a running time of 28 h and 30 min. Compared with the actual running time, this is an overestimate of more than 8%.

Since the number of varied sites in HVS-I increases very slowly with sample size and is bounded by the segment length, we may express the expected total running time (including data reduction) in terms of the number of sampled sequences. For larger data sets (with sizes above 500), we still observe an average case complexity on the order of 2.2.

Postprocessing

After having run MJ with different settings of ε (and possibly alternative weighting schemes for the sequence positions), one should focus on certain parts of the MJ network, namely nontrivial blocks and large cells, in order to enhance a parsimony search. A block of a network N is any maximal connected subnetwork which cannot be disconnected by deleting any one of its nodes. The blocks of a tree are all trivial; i.e., they are formed by the pairs of linked nodes. Every cycle in N extends to a nontrivial block (located within the ‘‘torso,’’ which is the smallest connected subnetwork of N containing all nontrivial blocks). Two distinct blocks can intersect in at most one node. Nodes occurring in more than one block are called cut nodes. Deletion of a cut node then disconnects N. The network N is therefore built up in a cactuslike fashion from its blocks. For example, the network of figure 4*c* has one nontrivial block, the cube, and two trivial blocks, the pendant links. As long as there is no ‘‘obsolete’’ block (i.e., a block with exactly one cut node and harboring no sampled sequence different from the cut sequence), every tree subnetwork of an MJ network N which connects the sampled sequences necessarily includes all cut nodes of N. In contrast, one may remove any single node which neither is a cut node nor is represented by a sampled sequence and still retain a subnetwork connecting the whole sample. Since all tree estimates realized within N are unanimous about the inclusion of the sequences labeling the cut nodes, we may treat these sequences as if they were actually sampled and rerun MJ with the thus expanded data set, this time applying it to each block separately. For example, consider the three superimposed MJ networks ($\varepsilon = 0, 1, 2$) in figure 5. The MJ network for $\varepsilon = 2$ has V, W, and X as its cut nodes, giving rise to six blocks, the prism, and five pendant links. MJ applied to C, V, W, and X recovers the prism whenever $\varepsilon \geq 1$ is chosen, but with $\varepsilon = 0$, no median vector is generated, as C, V, W, and X form a one-step path. Consequently, the MJ network for $\varepsilon = 0$ shrinks to the MP tree, whereas the MJ network for $\varepsilon = 1$ expands to the full quasimedian network, thus eliminating the nonmonotonicity anomaly in this case. Another instance in which this postprocessing of blocks comes into play is in the MJ network (with $\varepsilon = 0$) of figure 6. The torso of this network comprises four (nontrivial) blocks, namely two four-cycles and two ‘‘dominoes’’ (each composed of two four-cycles sharing a link). The domino harboring types 143 and 146 includes three unlabeled cut nodes; applying MJ with $\varepsilon = 0$ to these five sequences does not generate the sixth node of the domino anymore. Therefore, postprocessing the blocks here

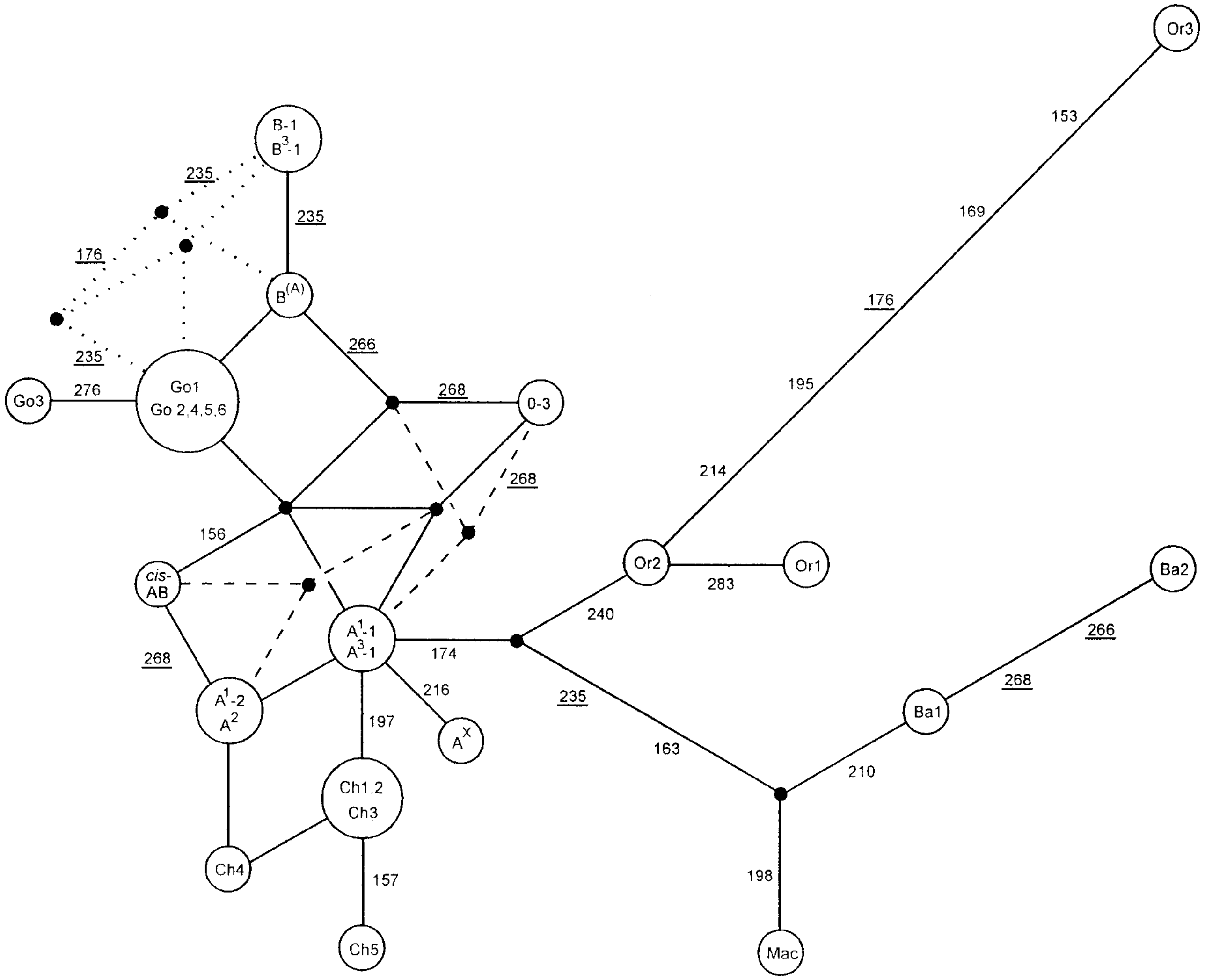

FIG. 7.—MJ networks for amino acid sequences of primate AB0 blood group enzymes, based on examined positions 153–283 (Saitou and Yamamoto 1997, table 3, but with amino acid P at position 157 transferred from Ch3 to Ch5). Prefixes Ch, Go, Or, Mac, and Ba stand for chimpanzee, gorilla, orang-utan, macaque, and baboon, respectively, whereas the unprefixed labels refer to human AB0 enzymes. The full network represents the raw MJ network for $\varepsilon = 1$; the postprocessed network for $\varepsilon = 1$ is obtained by deleting the obsolete nodes and links indicated by dotted lines; the MJ network for $\varepsilon = 0$ is described by the unbroken lines. Underlining indicates resolved multiple hits at a position.

results in the network exactly composed of the MP trees (which coincides with the greedy FHP network).

The MJ algorithm cannot guarantee that all obsolete intermediate sequence types will be eliminated in step 3, since the quick test for the number of incident links cannot detect obsolete groups of tightly clustered sequences. To give an illustration, consider the amino acid sequences of primate AB0 blood group enzymes compiled by Saitou and Yamamoto (1997, table 3). Setting ε to 1 (and disregarding four partially unexamined, uninformative positions), we obtain the MJ network shown in figure 7 for these data; the three nodes incident with the dotted links are remnants of preliminary connections abandoned later in the course of the algorithm. (For $\varepsilon = 0$, these nodes are no longer present, but for $\varepsilon \geq 2$ they become functional in that they provide alternative connections.) Postprocessing the single nontrivial block for $\varepsilon = 1$ eliminates these three sequence types along with the incident links. The postprocessed network decomposes along the cut node represented by the human sequence A^1-1/A^3-1 into the full quasimedian network for the AB0 enzymes from humans, chimpanzees, and gorillas on one side and the perfect tree linking the AB0 enzymes from orang-utans, baboons, and macaques with A^1-1/A^3-1 on the other side. According to this network, positions 176 and 235 each have experienced two changes (but caused by mutations at different nucleotide positions at the DNA level). Strikingly, amino acid positions 266 and 268 have mutated in concert at least twice and possibly three times during the divergence of humans, gorillas, and orang-utans. These two amino acid changes are necessary and sufficient to convert a blood group A enzyme to a functional blood group B enzyme, suggesting that their recurrent appearance in primate evolution may have been selected for (Saitou and Yamamoto 1997).

The most alarming substructures in MJ networks that need closer investigation are large cells. The cells of a network N are, intuitively speaking, kinds of minimal cycles from which the nontrivial blocks of N are built up. More precisely, using mathematical jargon, a cell is any cycle which cannot be obtained as the ''modulo 2'' sum of cycles with fewer links in the linear space (over the prime field) associated with the set of links of N. The cells in a full quasimedian network (and incidentally, in all the networks shown in this article) are three- or four-cycles. We then speak of a large cell if its number of links exceeds four. Arbitrarily large cells can easily be generated from artificial data sets; consider, for instance, $n \geq 6$ binary sequences $A_1, A_2, \ldots, A_n$ of length n, where the ith position is 1 at A_i and A_{i+1}, but 0 otherwise (indices read modulo n). The MJ network of this data set for $\varepsilon \geq 2$ is the full median network, which resembles a rosette or bouquet of four-cycles. When MJ is applied with $\varepsilon \leq 1$ instead, a two-step cycle is obtained which coincides with the minimum spanning network. The length of a minimum spanning tree (path) equals $2n - 2$ here. In contrast, the MP trees have length $(3/2)n$ if n is even and $(3/2)n + (1/2)$ if n is odd. Thus, the length differences between spanning path and MP tree grow linearly with n. In practice, large cells may occur for several reasons: either homoplasy is generally high and the choice of ε was too low (the result of a desire to produce a treelike or at least drawable network), or recombination (gene conversion, etc.) has partially acted on the data, or ambiguities of states in the sampled sequences are too frequent, or artifacts, such as contamination or documentation errors, are present. It is then recommended to generate the full quasimedian network of the sequences (whether sampled or reconstructed) representing the nodes of a particular large cell and to investigate the causes for its appearance in the MJ network under study.

Hybrid Approaches

The construction of the MJ network may be enhanced by running the RM method beforehand. RM operates in a fashion complementary to MJ in that it first resolves some character conflicts and eventually returns an extended data matrix with more characters but a reduced level of homoplasy. RM may also be regarded as parameter-driven: the crucial parameter r (the ''reduction threshold''), set equal to 2 by default, expresses, in equation (5) of Bandelt et al. (1995), how much larger the total weight of compatible characters compared with a conflicting character must be in order to postulate a parallelism for the latter. Thus, lowering r toward 1 leads to further ''reduction'' of network reticulations into more treelike networks, thereby increasing the risk of discarding the true evolutionary paths, whereas a reduction threshold $r \geq 2$ (as we would recommend when RM is combined with MJ for $\varepsilon \leq 1$), postulates fewer, but more obvious, recurrent events beforehand.

MJ and RM alone may not be ideally suited to data supporting potential phylogenies with rather long branches. In these cases, MJ may also be combined with other tree-building methods, since it explores a restricted solution space in the (joint) neighborhood of postulated trees. To this end, one would apply MJ to a set of sequences which comprises the original data set plus ancestral sequences hypothesized either from tree-building methods applied to the original data or from network analyses of smaller subsets, thus allowing for a hierarchical approach.

Acknowledgments

We thank N. Saitou, V. Macaulay, T. Warnow, and an anonymous reviewer for helpful comments. This work was performed by A.R. in partial fulfillment of the requirements for the doctoral degree.

APPENDIX 1

We verify that the constructed network N comprises exactly the links occurring in minimum spanning trees. Consider any link from N between sequence types X and Y of length δ_i. Select a preference ordering for the pairs of sequence types at distance δ_i where the pair X, Y comes first. Since the link X, Y connects different components of the partial network constructed up to distance δ_{i-1}, there cannot be any path between X and Y for which all links have lengths δ_{i-1} or smaller. Kruskal's algorithm (after having processed the pairs of sequence types at distances $\leq\delta_{i-1}$) would then select the link X, Y, given the prescribed preference ordering. On the other hand, the network N must include all links of an arbitrary minimum spanning tree T. Suppose the contrary; namely, let the pair U, V constitute the shortest link of T that would not be found in N. When this link is removed, T falls to two connected components. Necessarily (as T is minimal), all links of N connecting these two components have lengths of at least the distance δ_i of U and V. Therefore, in the construction of N, the sequence types U and V belong to different components of the partial network erected right after distance value δ_{i-1} has been processed. The link U, V would then be added to the growing network at the next stage (for distance values δ_i), contrary to our assumption.

APPENDIX 2

It remains to verify that the segments bounded by disjoint pairs V, W and X, Y of linked sequence types (having no ambiguous states) from the constructed network do not intersect. Suppose the contrary; then there is some vector Z (from the sequence space) satisfying

$$d(V, Z) + d(W, Z) = d(V, W)$$

and

$$d(X, Z) + d(Y, Z) = d(X, Y) \tag{1}$$

where d is the (possibly weighted) distance in sequence space. Then,

$$\begin{aligned} &d(V, X) + d(W, Y) \\ &\quad \leq d(V, Z) + d(X, Z) + d(W, Z) + d(Y, Z) \\ &\quad = d(V, W) + d(X, Y) \end{aligned} \tag{2}$$

by virtue of the triangle inequality and equality (1). Similarly,

$$d(V,\ Y) + d(W,\ X) \leq d(V,\ W) + d(X,\ Y). \qquad (3)$$

Assume

$$d(V,\ W) \leq d(X,\ Y). \qquad (4)$$

Let δ be the largest distance strictly smaller than $d(V, W)$ between sequence types from the network. Since the pairs V, W and X, Y constitute feasible links (with regard to $\varepsilon = 0$), we can state—after interchanging the roles of X and Y if necessary—that neither V and X nor W and Y are within a δ-step component. In particular, the distances for these pairs exceed δ and thus are bounded by $d(V, W)$ from below. Assuming

$$d(V,\ X) \leq d(W,\ Y) \qquad (5)$$

without loss of generality, the preceding fact may be expressed by the single inequality

$$d(V,\ W) \leq d(V,\ X). \qquad (6)$$

Moreover, as X and Y do not belong to the same δ-step component,

$$d(X,\ Y) \leq \min(\max(d(V,X), d(V,Y)), \max(d(W,X), d(W,Y))). \qquad (7)$$

In view of inequality (5), this may be simplified to

$$d(X,\ Y) \leq \max(d(W,\ Y),\ \min(d(V,\ Y),\ d(W,\ X))). \qquad (8)$$

If $d(X, Y) \leq d(W, Y)$ is true, then combining this inequality with inequalities (6) and (2) implies that equality must hold throughout, and in particular,

$$d(V,\ W) = d(V,\ X)$$

and

$$d(X,\ Y) = d(W,\ Y) \qquad (9)$$

would be true. If, however, $d(X, Y) > d(W, Y)$, then inequality (3), together with inequality (8), would yield the equalities $d(V, W) = d(X, Y) = d(V, Y) = d(W, X)$, which are then in conflict with $d(V, W) \leq d(W, Y) < d(X, Y)$ (employing inequalities 5 and 6). We conclude that equality (9) is indeed the only alternative. Since inequality (2) is thus an equality, it follows that

$$d(V,\ Z) + d(X,\ Z) = d(V,\ X). \qquad (10)$$

Hence, Z belongs to both segments from V to W and from V to X, respectively. We conclude that there would be a median vector for the triplet V, W, X which is necessarily different from V, W, X. This, however, would contradict the status of feasible links (such as the pairs V, W and V, X) in the final network. This settles the claim.

LITERATURE CITED

ANDERSON, S., B. G. BANKIER, M. H. BARRELL et al. (14 coauthors). 1981. Sequence and organisation of the human mitochondrial genome. Nature **290**:457–465.

BANDELT, H.-J., P. FORSTER, B. C. SYKES, and M. B. RICHARDS. 1995. Mitochondrial portraits of human populations using median networks. Genetics **141**:743–753.

BANDELT, H.-J., H. M. MULDER, and E. WILKEIT. 1994. Quasi-median graphs and algebras. J. Graph Theory **18**:681–703.

CHEN, Y.-C., A. TORRONI, L. EXCOFFIER, A. S. SANTACHIARA-BENERECETTI, and D. C. WALLACE. 1995. Analysis of mtDNA variation in African populations reveals the most ancient of all human continent-specific haplogroups. Am. J. Hum. Genet. **57**:133–149.

EXCOFFIER, L., and P. E. SMOUSE. 1994. Using allele frequencies and geographic subdivision to reconstruct gene trees within a species: molecular variance parsimony. Genetics **136**:343–359.

FARRIS, J. S. 1970. Methods for computing Wagner trees. Syst. Zool. **19**:83–92.

FORSTER, P., R. HARDING, A. TORRONI, and H.-J. BANDELT. 1997. Reply to Bianchi and Bailliet. Am. J. Hum. Genet. **61**:245–247.

FOULDS, L. R., M. D. HENDY, and D. PENNY. 1979. A graph theoretic approach to the development of minimal phylogenetic trees. J. Mol. Evol. **13**:127–149.

HWANG, F. K., D. S. RICHARDS, and P. WINTER. 1992. The Steiner tree problem. North-Holland, Amsterdam.

KRUSKAL, J. B. 1956. On the shortest spanning subtree of the graph and the travelling salesman problem. Proc. Amer. Math. Soc. **7**:48–57.

NACHMAN, M. W., W. M. BROWN, M. STONEKING, and C. F. AQUADRO. 1996. Nonneutral mitochondrial DNA variation in humans and chimpanzees. Genetics **142**:953–963.

PRIM, R. C. 1957. Shortest connection networks and some generalizations. Bell Syst. Technol. J. **36**:1389–1401.

RÖHL, A. 1997. Network. A program package for phylogenetic networks. Mathematisches Seminar, Universität Hamburg (available on request).

SAITOU, N., and F. YAMAMOTO. 1997. Evolution of primate AB0 blood group genes and their homologous genes. Mol. Biol. Evol. **14**:399–411.

SWOFFORD, D. L., G. J. OLSEN, P. J. WADDELL, and D. M. HILLIS. 1996. Phylogenetic inference. *In* D. M. HILLIS, C. MORITZ, and B. K. MABLE, eds. Molecular systematics. Sinauer, Sunderland, Mass.

TATENO, Y. 1990. A method for molecular phylogeny construction by direct use of nucleotide sequence data. J. Mol. Evol. **30**:85–93.

TORRONI, A., K. HUOPONEN, P. FRANCALACCI, M. PETROZZI, L. MORELLI, R. SCOZZARI, D. OBINU, M. L. SAVONTAUS, and D. C. WALLACE. 1996. Classification of European mtDNAs from an analysis of three European populations. Genetics **144**:1835–1850.

TORRONI, A., J. A. MILLER, L. G. MOORE, S. ZAMUDIO, J. ZHUANG, T. DROMA, and D. C. WALLACE. 1994. Mitochondrial DNA analysis in Tibet: implications for the origin of the Tibetan population and its adaption to high altitude. Am. J. Phys. Anthropol. **93**:189–199.

TORRONI, A., R. I. SUKERNIK, T. G. SCHURR, Y. B. STARIKOVSKAYA, M. F. CABELL, M. H. CRAWFORD, A. S. G. COMUZZIE, and D. C. WALLACE. 1993. mtDNA variation of aboriginal Siberians reveals distinct genetic affinities with Native Americans. Am. J. Hum. Genet. **53**:591–608.

NARUYA SAITOU, reviewing editor

Accepted August 31, 1998