Molecular Phylogenetics and Evolution
Vol. 16, No. 1, July, pp. 8–28, 2000
doi:10.1006/mpev.2000.0792, available online at http://www.idealibrary.com on IDEAL®

Median Networks: Speedy Construction and Greedy Reduction, One Simulation, and Two Case Studies from Human mtDNA

Hans-Jürgen Bandelt,* Vincent Macaulay,† and Martin Richards‡

*Fachbereich Mathematik, Universität Hamburg, Bundesstraße 55, D-20146 Hamburg, Germany; †Department of Statistics, University of Oxford, 1 South Parks Road, Oxford OX1 3TG, United Kingdom; and ‡Galton Laboratory, Department of Biology, University College London, Wolfson House, 4 Stephenson Way, London NW1 2HE, United Kingdom

Received May 20, 1999; revised January 19, 2000

Molecular data sets characterized by few phylogenetically informative characters with a broad spectrum of mutation rates, such as intraspecific control-region sequence variation of human mitochondrial DNA (mtDNA), can be usefully visualized in the form of median networks. Here we provide a step-by-step guide to the construction of such networks by hand. We improve upon a previously implemented algorithm by outlining an efficient parametrized strategy amenable to large data sets, greedy reduction, which makes it possible to reconstruct some of the confounding recurrent mutations. This entails some postprocessing as well, which assists in capturing more parsimonious solutions. To simplify the creation of the resulting network by hand, we describe a speedy approach to network construction, based on a careful planning of the processing order. A coalescent simulation tailored to human mtDNA variation in Eurasia testifies to the usefulness of reduced median networks, while highlighting notorious problems faced by all phylogenetic methods in this context. Finally, we discuss two case studies involving the comparison of characters in the two hypervariable segments of the human mtDNA control region in the light of the worldwide control-region sequence database, as well as additional restriction fragment length polymorphism information. We conclude that only a minority of the mutations that hit the second segment occur at sites that would have a mutation rate comparable to those at most sites in the first segment. Discarding the known "noisy" sites of the second segment enhances the analysis.

Key Words: **median networks; compatibility; human mtDNA; hypervariable segments; heterogeneity of mutation rates.**

INTRODUCTION

It is common practice to represent phylogenetic data by means of a tree; yet, there are several cases in which a network representation may be more appropriate. Obvious examples are instances of reticulate evolution, such as gene transfer, hybridization, and recombination, but networks may also be appropriate when a dendritic model applies (Bandelt, 1994; Bandelt and Dress, 1992; Bandelt *et al.*, 1995; Fitch, 1996, 1997). In this case, they can be used to indicate ambiguity in the tree topology that is inherent in the data set. This is especially relevant to intraspecific data, in which a number of the informative phylogenetic characters may have undergone recurrent mutation. In such instances, the presence of cycles in the phylogeny indicates uncertainty as to which characters have evolved more than once.

A phylogenetic diagram, whether it be a tree or a more general network, is, mathematically speaking, a connected graph consisting of nodes and links. The nodes represent taxa (e.g., haplotypes) that are either sampled or inferred as hypothetical intermediate taxa. The links are associated with evolutionary changes in the characters, which describe the genotypes or phenotypes of the taxa. A phylogenetic tree is simply a network without cycles, so that every pair of nodes is joined by a unique path. Networks are always understood to be unrooted unless a node is explicitly specified as the root (by drawing upon external information). The "median" networks that we use in this paper will have the additional property that all cycles have an even number of links since we assume that all employed characters are binary, that is, have only two states (a requirement that is no real limitation for mitochondrial DNA (mtDNA) data, as will be argued below). More specifically, the median networks are built up from cubes of any dimension. Compatibility analysis of the binary characters will tell us which cubes we have to expect and where to locate them in the final network.

Mitochondrial DNA data usually comprise either restriction fragment length polymorphisms (RFLPs) distributed throughout the mitochondrial molecule or sequence variation in the noncoding control region. The

1055-7903/00 $35.00

TABLE 1

Binary Matrix for the HVS I Data of Vigilant *et al.* (1989) with a Possible Shelling

			Nucleotide position							
	1111	11111	1	1	1	11	1	1	1	1
	6666	66666	6	6	6	66	6	6	6	6
	0222	11222	1	2	2	22	2	2	2	2
	9236	27079	5	1	1	34	3	4	6	9
	3346	92981	3	2	4	65	9	3	0	4
Sequence	t									
1, 2, 3, 4	0	0	0	0	0	0	1	0	0	1
5	1	0	0	0	1	0	0	0	0	0
6, 13, 15	0	0	1	0	0	0	0	0	0	1
7	1	0	0	0	0	0	0	0	1	0
8	0	0	0	0	0	0	1	1	0	1
9, 10	0	0	0	0	0	0	0	0	0	0
11	0	0	0	1	0	0	0	0	0	0
12	0	1	0	0	0	0	0	1	0	0
14	0	0	1	0	0	1	0	0	0	1
Periphery	2	1	2	1	1	1	*	*	1	*
Torso	*	*	*	*	*	*	4	3	*	4

variation in the control region is concentrated in humans in two hypervariable segments, HVS I and HVS II (Stoneking *et al.,* 1991), but the majority of sequencing studies have utilized data only from HVS I, although some positions in HVS II have been shown to be informative in certain instances (Aris-Brosou and Excoffier, 1996; Torroni *et al.,* 1996; Wilkinson-Herbots *et al.,* 1996). However, several workers have published data on both segments, and here we take a phylogenetic approach using median networks to investigate the variation in the two segments independently to assess the value of incorporating HVS II into the analysis.

COMPATIBILITY ANALYSIS

Preprocessing mtDNA Data

Before compatibility analysis and network construction begins, some preprocessing of the data is necessary to generate the binary data matrices. Examples are shown in Tables 1 and 2, taken from the sequence data in Fig. 2 of Vigilant *et al.* (1989). In that study, 15 !Kung individuals (from a Khoisan-speaking hunter–gatherer population in southern Africa) were analyzed for positions 16090 to 16365 in the first hypervariable segment (HVS I) and for approximately

TABLE 2

Binary Matrix for the HVS II Data of Vigilant *et al.* (1989) with a Possible Shelling

				Nucleotide position					
	00	0	0	0	0	0	0	0	0
	00	0	0	0	0	0	0	0	0
	11	1	1	1	1	1	2	2	3
	22	5	5	8	9	9	0	4	0
	57	0	2	9	8	9	7	7	9
Sequence								del	+ C
1, 2, 3, 4, 8, 13	0	0	1	0	0	0	0	0	0
5, 7	0	0	0	0	0	1	0	0	0
6	0	0	0	0	0	0	0	0	0
9, 10	0	0	0	0	1	0	0	0	1
11	1	1	0	0	0	0	0	0	0
12	0	1	0	1	1	0	1	0	0
14	0	0	0	0	0	0	0	1	1
15	0	0	0	0	0	0	1	0	0
Periphery	1	*	1	1	*	1	*	1	*
Torso	*	3	*	*	4	*	2	*	2

positions 00125 to 00310 in the second hypervariable segment (HVS II) of the mtDNA control region. Here we use the standard numbering system of the Cambridge reference sequence (CRS) (Anderson *et al.*, 1981) for the variant positions, which requires adding 16023 to the positions in HVS I and subtracting 546 from the positions in HVS II when numbered as in Vigilant *et al.* (1989) and some subsequent publications. Note that insertions in the cytosine tracts of HVS II may be scored in different ways. We refer to a sequence by listing the position numbers for which there is a mismatch with the CRS.

The matrices in Tables 1 and 2 comprise zeros at positions where the sequence haplotype in question matches the consensus type and ones where there is a (usually transitional) variant. It is important to note that this convention does not impose any polarity on the characters, but rather conveniently helps to refer to majority states (0) and minority states (1) of the haplotypes (not individuals); in case of ties we arbitrarily use the CRS to determine the 0 state. Within the binary framework, transversions may require further processing, as may insertions and deletions (indels). If there is a single transversional variant but no transitional variant at a position, then this position constitutes a binary character. If three or four bases occur, the transition–transversion ratio, which is of the order of 30:1 for mtDNA, is taken into account to break the character into a single transversion and one or two transitions. To decide between the two or four possibilities, respectively, for the reconstruction of the recurrent changes at such a position, a parsimony argument is invoked either by direct inspection or by running a parsimony program. In any case, such instances are rare, and moreover may sometimes be the result of sequencing or documentation errors. Further, there may be occasional alignment difficulties, especially in the cytosine tracts surrounding positions 16189 and 00310, where heteroplasmy is often involved (Bendall and Sykes, 1995) and the detection of length polymorphism might depend on the sequencing procedure (and thus be lab dependent). It is then best to disregard such unstable information (e.g., variants often scored as 16182t and 16183t, with suffix "t" indicating transversions).

Some further preprocessing (actually partly performed already in the data set of Vigilant *et al.*, 1989) simplifies the data matrix. For example, although there are 15 individuals, there are only nine distinct haplotypes across the first segment. We eliminate such multiple rows from the data and record the associated individuals or the frequency of each haplotype separately. In addition, the matrix needs only to represent varied positions; unvaried positions are thus removed. If no position survives, the data are trivial and we stop here. Multiple columns are pooled and considered as a single character with a weight equal to the number pooled (or their sum of weights, in the case in which mutations are weighted differentially). Thus, in the first !Kung matrix, positions 16236 and 16245 form one character of weight two transitions.

Trees from Cliques

The first stage of the analysis is to decide which pairs of characters could individually fit on trees without any additional recurrent changes. For instance, positions 16239 and 16294 from Table 1 have only three character states, 00, 01, and 11, in combination. Such characters are said to be *compatible;* the evolutionary path $00 \leftrightarrow 01 \leftrightarrow 11$ describes a unique change at either position. In contrast, the presence of all four combinations, 00, 01, 10, and 11, as is the case for positions 16239 and 16243, makes it impossible to have both characters uniquely derived on any tree; such characters are said to be *incompatible.*

At the second stage, these pairwise comparisons are aggregated in the following way: a *clique* of characters is a set of pairwise compatible characters which is maximal with respect to inclusion—that is, it cannot be extended further by finding yet another character from the table that would be compatible with all characters in this set. (Contrast Meacham and Estabrook, 1985, who do not require maximality with respect to inclusion.) The complementary notion is that of an *anticlique,* which is a set of pairwise incompatible characters that cannot further be extended. Every character may belong to more than one clique as well as to more than one anticlique.

It is well known (cf. Barthélemy, 1989; Meacham and Estabrook, 1985) that every clique supports a tree, the links of which correspond to the characters in the clique. For instance, Table 1 has exactly two cliques: one comprising all positions except 16243 and another, smaller, one consisting of all positions except 16239 and 16294. The tree representing a clique can be built up simply by processing the characters one after another in any order ("tree popping" sensu Meacham, 1981). The character states of the branching nodes in the tree constructed from a clique can be inferred from suitably chosen triplets of sampled haplotypes in median fashion—that is to say, by taking the majority consensus. Given such a node x, select a haplotype from any of three distinct branches emanating from x. The character states of x (the consensus) are then obtained by applying the majority rule to the corresponding states of the three selected haplotypes (e.g., 0, 0, 1 would give 0). Thus, every branching node which is not occupied by a sampled haplotype can be generated as a median type from at least one triplet of sampled haplotypes. Conversely, the median types generated from triplets of any nodes from the tree are all nodes of the tree.

By definition, every set of pairwise compatible characters can be extended to at least one clique. The

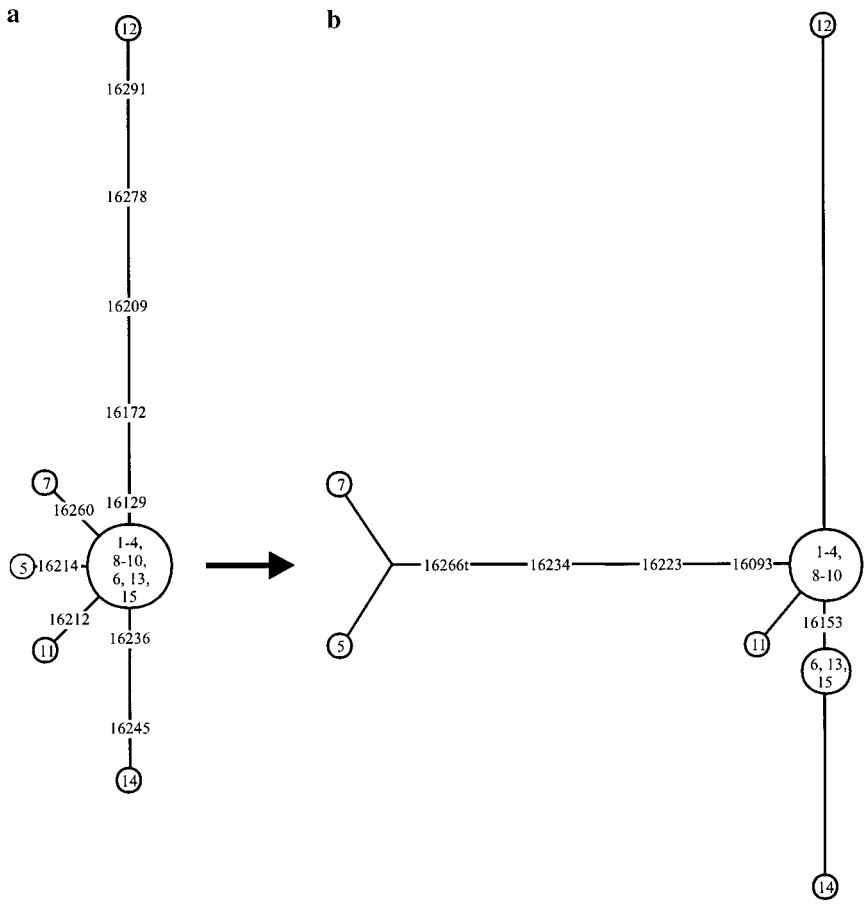

FIG. 1. (a) Tree representing all parsimoniously uninformative characters (which determine the terminal links of the median network) and (b) tree representing the intersection of the two cliques in the data set of Table 1 (which corresponds to the composition of all subtrees pendant from the median network). Nodes are labeled by the numbered individuals which they represent; labeled links correspond to newly entered characters.

characters common to all cliques are exactly those which are compatible with every character from the given data set. Among these characters we distinguish the peripheral characters, which are determined iteratively by the following straightforward procedure (cf. Saitou, 1998). Given the preprocessed data matrix, one searches for a character for which one of its two states, 0 or 1, occurs in only one sequence type. In the network representation that we envision, this type is connected to the rest of the network by a terminal link, along which this character changes, the tip of the link being occupied by the unique type with the variant state. We then erase the character from the processed data matrix and put it into *shell* 1 (i.e., it receives label 1). The same procedure applies to all other characters with a unique 1 or a unique 0. All these characters will make up shell 1. Then, to iterate this procedure, one needs to update the data matrix by collapsing types that have become identical; that is, one passes through a rudimentary preprocessing phase. Note that the characters transferred to shell 1 describe a star on their own, that is, a tree with a unique center and $k \geq 2$ tips linked to the center or simply a single link with its two end nodes. After preprocessing the truncated matrix, the iteration proceeds by identifying the characters that delineate the tip nodes for this new matrix and putting them into shell 2. This shelling procedure terminates at the point where one cannot discern any characters in the processed data matrix with a unique appearance of one of the two states. This final truncated matrix is then referred to as the *torso* data matrix (Bandelt *et al.*, 1999), the characters of which are said to be *nonperipheral*. The characters assigned to the shells 1 to p are then referred to as the *peripheral* characters.

To give an example, consider Tables 1 and 2. In the data matrix of Table 1, all characters except three are peripheral and fall into two shells. The star described by the characters of the first shell is shown in Fig. 1a, whereas the tree of Fig. 1b displays all peripheral characters. In Table 2, all peripheral characters are assigned to shell 1. The nonperipheral characters in either case will be assigned to the subsequent shells, in

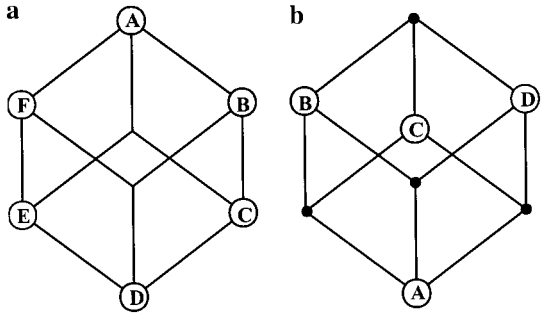

FIG. 2. The generic situations for the location of sampled sequences in a cube with respect to three pairwise incompatible characters: labeled nodes are nonempty (that is, represent sampled sequences) and nodes indicated by a filled circle may or may not be empty.

a natural extension of the process for trees; see below ("Shelling").

In the simplest situation the torso would be empty. This is exactly the case when the original characters form a clique and thus support a tree. This tree is then easily constructed in a streamlined form of tree popping. Start with the characters of the deepest shell p. From the nodes of this star one then pops out the characters of shell $p - 1$, and iterates. This "speedy" construction of the tree thus proceeds from its center or bicenter through the shells toward the tips.

Cubes from Anticliques

Two incompatible characters partition the haplotypes into four groups that can be represented by a cycle in which two nodes are linked when they are distinguished by just one of the characters. Sampled haplotypes compared at three pairwise incompatible characters occupy part of a cube, either in the fashion of Fig. 2a, in which two opposite nodes are empty intermediate nodes, or in the fashion of Fig. 2b, in which at least four nodes are occupied that are pairwise distinguished by exactly two characters. For example, with positions 00150, 00198, and 00207 from Table 2, we produce a cube since all eight combinations of character states—if not already present in a sampled haplotype—can be generated as median types: the three unobserved combinations 110, 011, and 101 are obtained from the haplotype triplets (9, 11, 12), (9, 12, 15), or (11, 12, 15), respectively (cf. Figs. 3e and 5). If we were to pool the data from Tables 1 and 2, then a four-dimensional cube would emerge, reflecting the pairwise incompatibility between the former three positions in HVS II and the position 16243 in HVS I.

One expects that, more generally, k pairwise incompatible characters generate a k-dimensional cube; that is, all character state combinations will eventually be generated by successively producing median types. This is indeed the case (following from a universal algebraic fact; Bergman, 1977) and is readily established by mathematical induction. For $k = 2$, we have a cycle with four nodes (i.e., a two-dimensional "cube") occupied by sampled haplotypes. In the induction step, assume that we have $k \geq 3$ pairwise incompatible characters, numbered 1 through k, and suppose that the assertion is already verified up to $k - 1$ characters. Then either set of $k - 1$ characters $2, 3, \ldots, k$ and $1, 3, 4, \ldots, k$ generate a $(k - 1)$-dimensional cube. Consider any node x of the k-dimensional cube that does not represent a sampled type. We need to show that x can be obtained by taking median types iteratively. If we disregard the first and the second character separately, then either truncated type associated with x can be generated as an iterated median type by virtue of the induction hypothesis. Performing the median process with the omitted characters now included, we obtain either the node x (at least once) or the two nodes u and v, linked to x in the k-dimensional cube, that differ from x in character 1 or 2, respectively. The desired combination of states for x with respect to characters 1 and 2 alone must be present in some sampled haplotype w since these two characters are incompatible, but then x is exactly the median type produced from u, v, and w. This finishes the induction step.

We can also build up the k-dimensional cube in question by processing character by character: in step j we double the already constructed $(j - 1)$-dimensional cube, link the pairs of corresponding nodes, and sort the haplotypes to the two parts according to their state at the jth character. It suffices to label only one of the 2^{j-1} links corresponding to the jth character.

The cubes (with dimensions exceeding one) representing the anticliques (with more than one character) thus focus on the conflicts in the data due to recurrent mutations: any most parsimonious tree would require at least $k - 1$ additional steps for k characters forming an anticlique.

Median Networks

So far, we have encountered here two kinds of median networks: trees and cubes (of any dimension). The median network generated from given binary data will normally fall in between these two extremes. Formally, the median network may be defined as follows: the set of nodes is given by the smallest set of sequence types which includes the sampled types and the consensus sequence for each triplet of (sampled or hypothetical) types; two nodes are linked whenever there is no other node which is intermediate between them. This definition is neither conceptually very illuminating nor adequate for computational purposes. More economically, the construction proceeds in a character-by-character fashion. In the process of construction, compatibilities between characters become manifest in simple branching, whereas incompatibilities increase dimensionality by doubling parts of the network. For instance, if an-

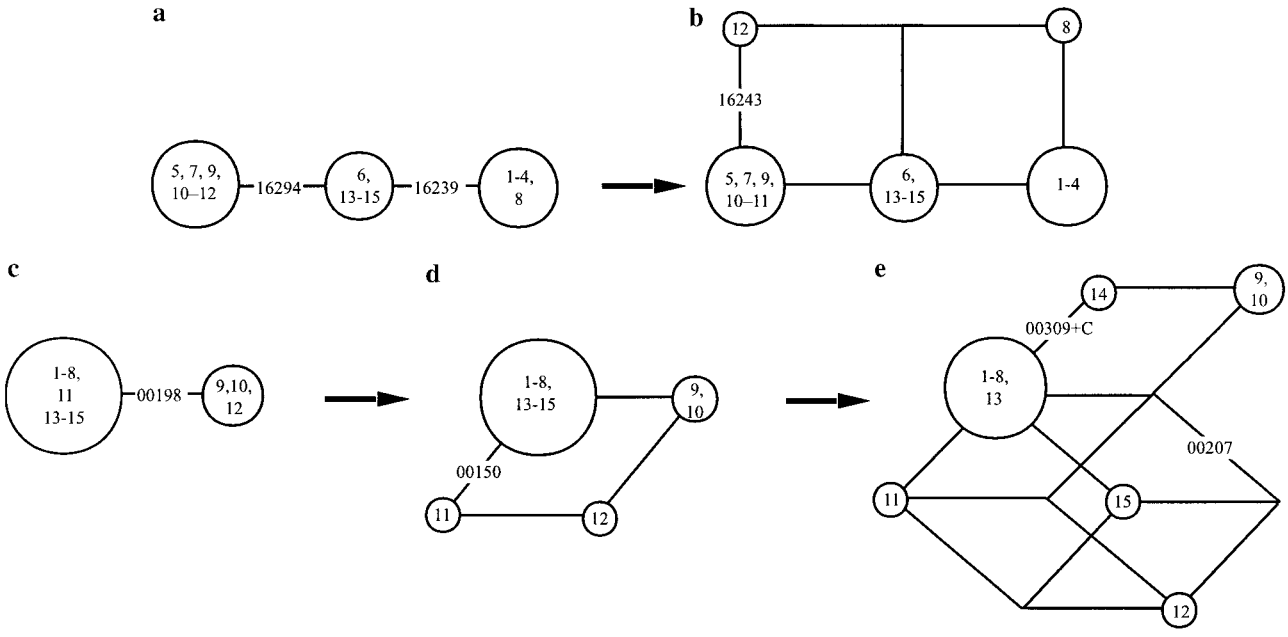

FIG. 3. Stepwise constructions of the torsos, (b) and (e), of the median networks representing Tables 1 and 2, respectively. The characters entering the torsos are (shell-) numbered in the last row of Tables 1 and 2. Labeling of nodes and links as in Fig. 1.

other character is entered into a set of characters which are all incompatible with it, then the median network is doubled, so that each of the previous nodes is linked to its duplicate. This can be shown rigorously in a fashion similar to that in the case of cubes. An example is provided by the torso of the median network representing the data of Table 1: after positions 16294 and 16239 have been processed (Fig. 3a), the third position 16243 is incompatible with both of them, and thus the path is doubled and a "domino" (Fig. 3b) results.

The generic situation for a character α in a data set is that the other characters are neither all compatible nor all incompatible with it. The two states, 0 and 1, of α sort the sequence types into the *sides* of α: the 0-*side* A comprises all types with state 0 for α, whereas the complement A' of A consists of all types with state 1. Since each character compatible with α has one side in which all types have the same state at α, every character $\beta \neq \alpha$ from the data set belongs to exactly one of the following categories: (1) β is incompatible with α, (2) the 0-side of α includes one side of β, or (3) the 1-side of α includes one side of β. An extremal choice of α would simplify the picture in that one side of α is then fully included in this set of sequences: this happens precisely when one of the categories 2 or 3 is empty, that is, when one side of α does not contain any side of any other character of the data set. The status of β does not change when we extend the data matrix by the (hypothetical) sequences labeling the unsampled nodes of the median network, because the (majority) consensus rule respects compatibility of characters. Therefore, every link corresponding to α connects two nodes at which each character β, compatible with α, has the state $e \in \{0, 1\}$ for which the e-side of β includes one side of α (specifically, the 1-side of α if β is of category 2 and the 0-side of α if β is of category 3). When, in all sampled sequences, the state at each character β compatible with α is made equal to the state e, then the thus-modified sequences constitute a subnetwork of the median network for the original data set and furthermore form a copy of the median network associated with the truncated data matrix comprising only α and the characters β incompatible with α. The structure of the latter network has been analyzed in the preceding paragraph: it is obtained from the median network associated with characters incompatible with α by duplication (see Fig. 3 of Bandelt *et al.*, 1995 for an illustration).

The order in which the characters are processed during network construction does not affect the resulting network. Nevertheless, the construction process can be streamlined by a careful choice of the order, as described next.

SPEEDY CONSTRUCTION

Extremal Characters

The characters which correspond to the terminal links of a tree (or more generally a median network) and the characters determining a cube all constitute instances of what we perceive as extremal characters. A character α is said to be *extremal* for a data matrix if

removal of all characters incompatible with it renders it a peripheral character corresponding to a terminal link in the median network associated with the truncated data matrix. Equivalently, α is extremal exactly when there are no distinct characters β and γ in the data set such that β, α, and γ together on their own are represented by a path with three links, where the interior link corresponds to α. This, in turn, can be expressed in terms of the sides of the characters involved: the set of sides of all characters is ordered by inclusion—precisely the minimal members (not necessarily of the same cardinality) of this ordered set yield sides of extremal characters. Indeed, if both sides A, A' of a character α are nonminimal, then there are characters β and γ with B and C, respectively, as one of their sides such that $B \subset A$ and $C \subset A'$ (thus obtaining a forbidden path as described above). An extremal character can be found by a single pass through the characters (in any order): beginning with one of the two sides of the first character, we compare the currently held candidate for a minimal side with the two sides of the character under processing and substitute the former by one of the latter if we obtain a proper subset this way. The extreme situation in which every character is extremal has been investigated by Bandelt and van de Vel (1991): exactly in this case the maximal cubes in the associated median network would share a common node.

Shelling

Recall that the shelling of a clique and its associated tree amounts to collapsing successively terminal links and thereby moving tip types stepwise toward a central node. When one aims at generalizing this to arbitrary data matrices and their median networks, one typically faces the obstacle that the extremal characters are not necessarily pairwise compatible and hence do not conform to a star. We therefore need to take care to select at each stage a collection of pairwise compatible extremal characters. No further requirement is imposed, but we would normally prefer a collection (maximal with respect to inclusion) to which no further extremal character could be added without violating compatibility. The collection however selected is then called the *outer shell* or *shell* 1. We iterate this procedure by removing all shell 1 characters from the data matrix and updating the list of extremal characters; those that were extremal and not put into shell 1 stay extremal, but the truncated matrix will typically admit new extremal characters (unless all shell 1 characters were incompatible with the remaining characters). The next collection of pairwise compatible extremal characters is then the shell with the next number, and so forth, until all characters have been assigned to shells. This ordered partition of the characters is then called a *shelling*. Mathematically speaking, a shelling is any mapping of the characters to natural numbers such that (1) all characters with the same shell number are pairwise compatible and (2) for any three distinct characters β, α, and γ determining a path with interior link corresponding to α, at least one of β and γ has a smaller shell number than α.

The effect of the shelling process can be visualized in the median network representing the full data matrix as a kind of organized trip of the sampled types toward a single node. Whenever an extremal character α exits to a shell, one replaces all coordinates of the sampled sequences at α by the state shared by the side A' complementary to a (chosen) minimal side A of α, so that henceforth this character becomes unvaried in the thus-modified data matrix. After all characters have been processed, the final modified data matrix comprises a single sequence which labels a node of the median network; we refer to this sequence/node as the *center* of the shelling. The potential centers of a shelling form some cube of dimension ≥ 0, because both sides of a character α in shell i are minimal with respect to the truncated data matrix of shells $\geq i$ exactly when shell i comprises only α and α is incompatible with all characters of shells $\geq i+1$: one could shift these singleton shells to the end of the shelling, where they determine the cube of potential shelling centers.

To give an example, consider the numbering of the characters in the last two rows of Tables 1 and 2: in both cases a shelling is indicated that entails shellings of periphery and torso separately. For Table 1 one can devise another ("greedy") shelling which no longer separates periphery from torso but requires only three shells, by assigning character 16239 to shell 1, character 16243 to shell 2, and character 16294 to shell 3 instead.

For network construction, it would be desirable to minimize the total number of necessary shells. To find an optimal shelling (i.e., one for which this minimum number of shells is attained), however, turns out to be a computationally hard problem. For the mathematically inclined reader we briefly indicate why. Any 2-connected, nonbipartite, triangle-free graph G can be turned into a binary data matrix (viz., its edge–vertex incidence matrix), where the characters correspond to the vertices and the individuals are the edges: in each character, state 1 appears at those individuals which as edges of G share the vertex associated with this character. The nodes of the median network generated from this data matrix are the incidence vectors of the edges of G plus all unit vectors (i.e., the vectors with a single 1 coordinate) and the zero vector. The network has no 3-cubes (because G is triangle-free) and is its own torso (because G is 2-connected). In this particular case, two characters are compatible precisely when the combination 1,1 of states does not occur. Hence, every partition of the character set into sets of pairwise compatible characters is a shelling centered at the zero vector and vice versa. The shellings centered at the

zero vector are therefore in one-to-one correspondence with the colorings of the graph G (Bandelt and van de Vel, 1989). For any shelling with another center, the deepest shell consists of a single character, which could as well become the outer shell instead, so that this shifted shelling would now be centered at the zero vector. Then, as G is nonbipartite, the data matrix does not admit any shelling with just two shells. The decision problem of whether three shells suffice ("3-shellability") for this data matrix is thus equivalent to the problem of whether the graph G is 3-chromatic. Since 3-colorability is NP-complete (even for the particular class of graphs described here), so is 3-shellability. Therefore, one cannot hope to design an efficient algorithm for optimal shelling. In practice, though, the following "greedy" approach would do the job.

Procedure Greedy Shelling

- Input: Preprocessed binary data matrix.
- Iteration: Beginning with the input matrix as current matrix and setting $i = 1$, determine the extremal characters of the current matrix and process them in their input order. Take the first extremal character, then the next extremal one that is compatible with the first, then select the next one compatible with the former two, and so on, until all extremal characters have been screened. The selected characters form shell i. The remaining characters constitute the current matrix for the next round, with i increased by 1. Continue as long as the current matrix is not empty.

Network Construction

We construct the median network shellwise from the center of a shelling. At initialization, the network (for the empty character set) consists only of the center node, which temporarily hosts all sampled types. Whenever a character α is processed, the corresponding coordinate of the center sequence determines which types are popped out, viz., exactly those which have the variant state at α compared to the center. For brevity, we refer to these types together as the variant side of α (relative to the shelling center). For the iteration assume that the median network of the truncated data matrix for shells $\geq i+1$ has been constructed, where $1 \leq i \leq s$, with s being the largest shell number. For each character α in shell i determine the characters from shells $\geq i+1$ incompatible with α. These characters are then represented by a subnetwork exactly comprising the nodes which share with the variant side of α the same (unvaried) state combination for the other characters compatible with α from shells $\geq i+1$. (Consequently, this subnetwork includes all shortest paths between any of its nodes—and is the smallest one with this property which includes the variant side of α.) This subnetwork is then matched to a copy, in which the variant side of α is moved to its matched neighbor; the matching links all correspond to character α and are drawn in parallel for easier recognition. This popping process can be carried out independently (without updating) for all the characters of shell i since the variant sides of these characters are disjoint (by compatibility). So, it can never happen that a type needs to be moved more than once per shell. After the outer shell has been finished, the median network of the data matrix is complete.

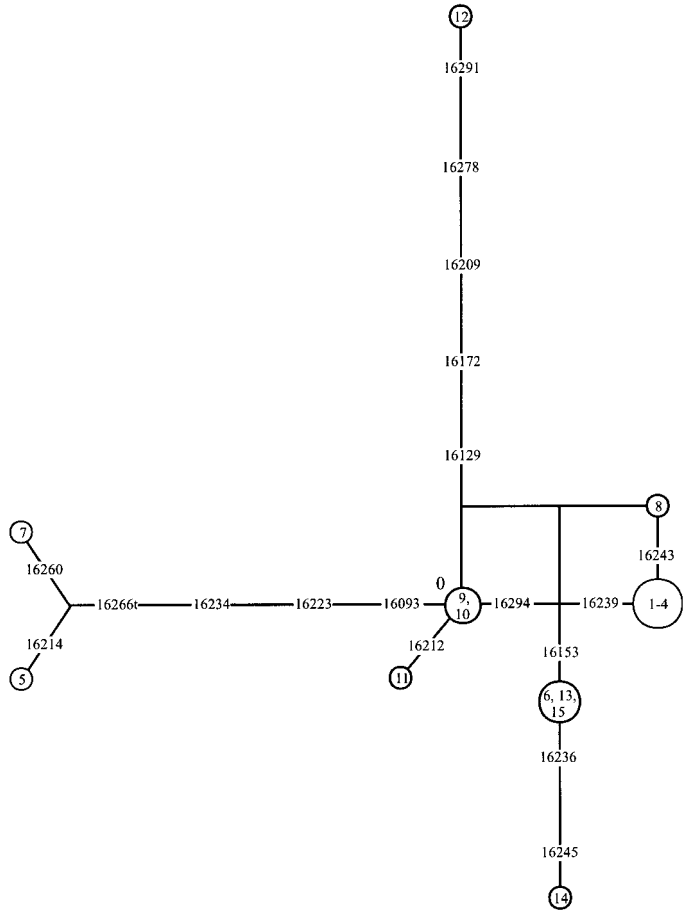

FIG. 4. Median network representing Table 1, which displays HVS I variation in the !Kung (Vigilant *et al.*, 1989). The node labeled "0" corresponds to the consensus sequence type (encoded by a string of zeros).

For the data of Tables 1 and 2 with shellings as indicated, the construction proceeds from Fig. 3a to Fig. 3b and then in two steps to Fig. 4, which thus represents Table 1, and from Fig. 3c via Fig. 3d to Fig. 3e and finally to Fig. 5, representing Table 2. It appears that the HVS I network is more tree-like.

GREEDY REDUCTION

The median network generated from the raw data matrix (of binary characters) will in general be unnecessarily large, because it includes potential evolutionary pathways which are extremely unlikely. We have

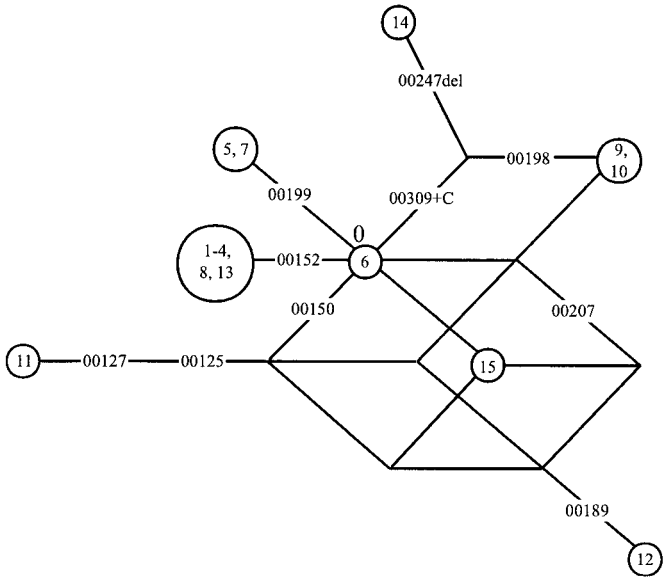

FIG. 5. Median network representing Table 2, which displays HVS II variation in the !Kung (Vigilant *et al.,* 1989). The node labeled "0" corresponds to the consensus sequence type. Indel events are indicated by suffixes to the nucleotide position concerned: "del" for a deletion and "+C" for the insertion of a cytosine.

therefore proposed a method of reconstructing the most obvious recurrent mutations by splitting characters into new characters that account for the hypothetical multiple hits (Bandelt *et al.,* 1995). The thus-transformed data matrix is then turned into a median network, which has been called the "reduced" median network to distinguish it from the ("full") median network generated from the raw data. This process has been referred to as "reduction" since both homoplasy of the data matrix and resulting reticulation in the associated network are reduced. The explicit reconstruction of recurrent mutations involves a *local* search, focusing on only a few characters at a time. Each of the reduction steps employs a parsimony criterion and additionally invokes a frequency criterion (for sampled sequence types). To speed up this process for manual analysis as well as computer search, we here propose to modify the reduction rules of Bandelt *et al.* (1995) slightly and to operate with a priority order for the characters. Reductions will then be performed in a greedy manner (rather than evaluating all potential reduction instances and choosing the best); that is, whenever an instance shows up in the processing order and conforms to the reduction rules, the reduction is executed. A final postprocessing is proposed to undo (rare) instances of excessively greedy reduction. The resulting reconstructed data matrix is then represented by what will be called the *greedily reduced median* (GRM) network. Since reductions cannot affect the periphery, greedy reduction will operate only on the nonperipheral characters. In fact, the characters belonging to all cliques of the data matrix will never be needed in the reduction queries. One could employ the decomposition strategy described in Bandelt *et al.* (1995, p. 747), but for the sake of conceptual simplicity we will consider the whole torso as the target of reduction.

Priority Order

We propose basing the priority order on estimated relative mutational rates of sites. There is by now sufficient evidence that the positions in the control region of human mtDNA vary greatly in their mutational rates, with transversions much less probable than transitions. Hasegawa *et al.* (1993) have analyzed HVS I in regard to positional mutability. Although more refined methods and additional phylogenetic information are needed to estimate the actual positional mutation rates, their list (their Table 3) gives a good impression as to which positions in HVS I may be expected to be hypervariable. The three most variable positions in HVS I are reported to be nps 16129, 16189, and 16311; this concurs with our experience that nps 16189 and 16311 are the major troublemakers in phylogenetic analyses of Eurasian and Native American HVS I data (Forster *et al.,* 1996; Richards *et al.,* 1996), whereas np 16129 appears to be extremely variable in African haplogroup L1 lineages (Watson *et al.,* 1997).

The Hasegawa *et al.* (1993) list is translated here into relative mutational rates by dividing the positional scores by the total score (i.e., the number of all recorded mutational events). Note that the positions which were not observed to vary in their compiled data set should not be regarded as absolutely invariable but instead should receive a mutational score between zero and one, say $\frac{1}{3}$. The relative mutational rate of a character, constituted by several sites, is then the product of the individual rates of the constituent sites (assuming independence of sites). To ensure that the relative mutation rates of the characters under consideration give a strict linear order, inverted to give the *priority order,* ties must be broken (arbitrarily if no further information is available). Thus, the higher the mutation rate, the lower the priority.

Reduction Rules

To motivate the reduction process, let us reverse the approach and suppose that we know the true tree with the mutations (all constituting transitions, say) recorded at the links. Consider a single instance of homoplasy: assume that two mutations hit the same site. Then we distinguish three cases: (1) the two mutations happened along one and the same link of the true tree, thus canceling each other, (2) the two mutations hit two distinct links sharing a common end node, or (3) the two mutations hit links which are connected by a shortest path of $k \geq 1$ links in the true tree. In cases (1) and (2), the recurrent events ("hidden mutations")

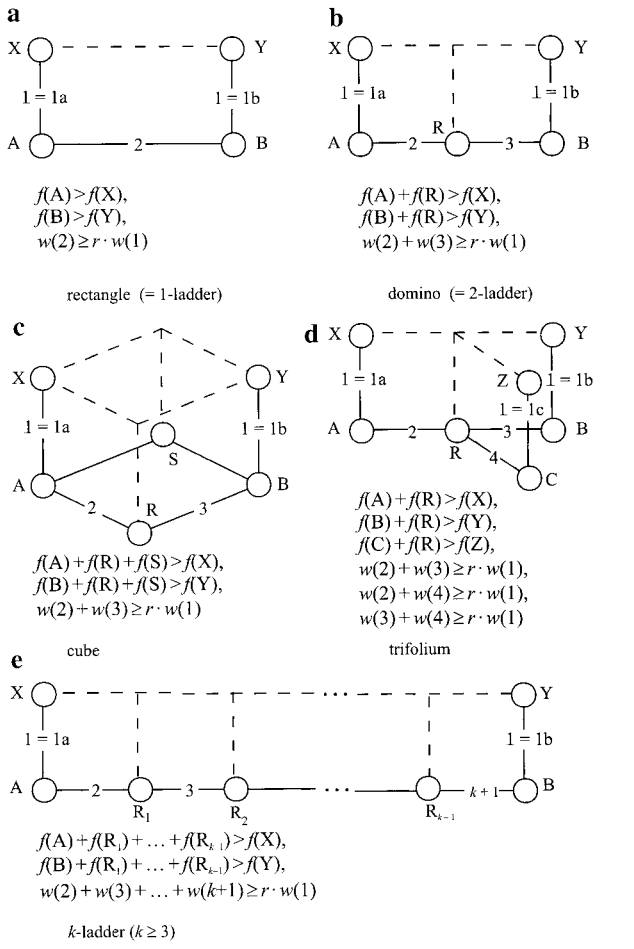

FIG. 6. The five instances of greedy reduction. Each diagram displays the median network relative to characters 1, 2, etc.; the reduced median network is represented by unbroken lines and postulates recurrent mutations at character 1 (underlined), which is replaced by characters 1a, 1b (and 1c). Labeled nodes are assumed to be nonempty, that is, at which sample frequency f is larger than zero; w refers to character weights; $r \geq 1$ is a prescribed parameter (set to 2 in all subsequent examples). Reduction is executed only when all respective inequalities are valid and an additional compatibility check for the resolution of character 1 into two or three new characters is passed (see text).

are unreconstructible from the observed data, which nevertheless perfectly conform to a (wrong) tree. In case (3), the full median network will represent the recurrent event by exhibiting some reticulation in the form of a single chain of k 4-cycles ("k-ladder"; see Figs. 6a, 6b, and 6e). This network would also be recovered in this trivial case even by distance-based methods, such as split decomposition (Bandelt and Dress, 1992) or the heuristic approach of Fitch (1997). If two homoplastic events of type (3) took place, then it may happen that the two ladders which would separately reflect each event would overlap or fold up in the full median network. Figures 6c and 6d illustrate the corresponding reticulate parts of the network in two generic instances: the trifolium (Fig. 6d) is the overlay of three mutually overlapping 2-ladders incurred by recurrent changes at the same character (No. 1), whereas the cube (Fig. 6c) arises when the ladders fold up, as a result of homoplastic events at two characters (No. 1 and one of Nos. 2 and 3).

The recurrent changes at character 1 in each instance depicted in Fig. 6 can be reconstructed by parsimony as long as the weight of character 1 exceeds the total weight of the other characters incompatible with it. In the instance shown in Fig. 6a, however, there would be two alternative pathways (cf. Fig. 4 of Bandelt *et al.*, 1995), indistinguishable by parsimony alone, which would account for two changes at character 1. In this case, the frequency with which the types were sampled can assist in choosing the pathway which is more likely. For example, if type A is sampled more often than type X, as well as B more often than Y, then there is a good chance that A and B bear the ancestral state at character 1, particularly when they are taken from an expanding population (Donnelly and Tavaré, 1986). In contrast, if character 1 underwent a back mutation, then the expected frequency pattern would not allow us to prefer one path over the other and the rectangle would thus remain unresolved. In all remaining instances (Figs. 6b–6e), we will employ frequencies to guide and reinforce the parsimony decision: if the observed frequency pattern is at odds with the expectation, then the resolution is blocked at this stage of the analysis. As the intermediate types enter the frequency comparisons twice, even back mutations have a fair chance of being identified. To lower the risk of false reconstructions we will require that the parsimony decision itself be more clear-cut, by introducing a factor $r > 1$, the *reduction threshold*, in the comparison of character weights: the multiple changes of character 1 in Fig. 6 will be recognized only when its weight is at least r times the total weight of the other characters indicated in the figure (see the weight inequalities in Fig. 6). Then each set of characters in the (torso) data set which conforms to one of the five instances (Figs. 6a–6e) constitutes a potential reduction instance. Resolution of character 1 would then entail its replacement by two or three (in instance Fig. 6d) new characters 1a, 1b (and 1c) which distinguish the recurrent changes. These new characters inherit the priority of the old character 1 but with ties broken at random. Before actually reducing the homoplasy in this way, an additional check must be passed, as motivated next.

An implicit goal of the reduction process could be seen as to reduce the level of data homoplasy, as expressed by character incompatibility. Although the weight and frequency conditions alone would normally achieve this, there are, however, potential pitfalls, which may occur with real data such as human Y-STRs (Peter Forster, pers. comm.). Figure 7 displays an artificial instance illustrating this problem: four haplo-

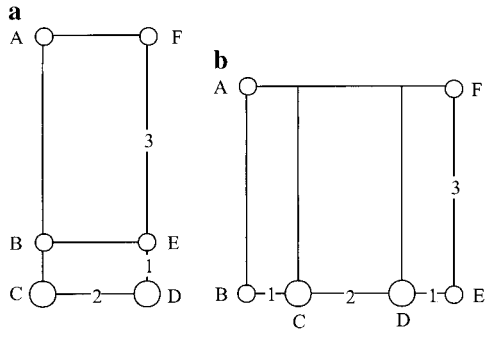

FIG. 7. Median network (a) and the reduced median (RM) network (b) for an artificial data set distinguishing haplotypes A–F relative to weighted characters 1, 2, and 3. This reduction (performed with $r = 2$), however, violates the compatibility criterion for characters 1 and 3. Lengths of links are proportional to character weights and areas of circles are proportional to haplotype frequencies.

types A, B, E, and F with frequency 1 and two haplotypes C and D with frequency 3 are distinguished by three characters 1, 2, and 3 having weights 1, r, and $r + 2$, respectively, where $r > 1$. With the reduction threshold set to r, there is only a single possibility for reduction, invoking the rectangle rule for character 1 vs character 2; no further reduction can then be carried out. This result, however, is hardly convincing: while the length of a most parsimonious tree is $3r + 4$, the shortest trees connecting the six haplotypes in the reduced median network have length $3r + 6$. To measure the amount of incompatibility, we propose taking the sum of the products of weights for all incompatible pairs of characters. The total weight of incompatibilities in the original data set is thus $r \cdot (1 + r + 2) = r(r + 3)$, whereas the total weight for the "reduced" data set equals $(1 + r + 1) \cdot (r + 2) = (r + 2)^2$. This increase is reflected in the grid sizes of Figs. 7a and 7b. In general, the total weight of incompatibilities is proportional to the total area of the rectangular "cells" of the median network, provided that no three characters are pairwise incompatible. To anticipate such unfavorable cases (in the presence of considerable homoplasy), an additional compatibility check is introduced before reduction is actually executed. Assume that character 1 is processed as a potentially resolvable character conforming to one of the instances in Fig. 6. The compatibility check now demands that every character of the currently processed data matrix which is compatible with character 1 should also be compatible with any of the new resolutions of character 1. Notice that the compatibility check for character 1 in the above example (Fig. 7) would block the resolution of this character: characters 1 and 3 are compatible (Fig. 7a) but the two new characters substituting character 1 (both labeled by $\underline{1}$ in Fig. 7b) would become incompatible with character 3.

Now we have all the ingredients at hand for applying the reduction rules. We organize the reduction process in rounds $i = 1, \ldots, m$, each associated with a reduction threshold r_i in decreasing order $r_1 > \ldots > r_m$. As a rough guideline, we propose to choose m from 1, 2, and 3 and reduction thresholds between 3 and 1.5. Since it is very likely that the low-priority characters underwent recurrent changes rather than the ones of high priority, we seek to resolve a character having low priority first. To this end, start in round i with the lowest priority character and name it No. 1 temporarily. Then we seek the lexicographically highest priority set of characters (designated as 2 to $k + 1$) which together with the current No. 1 conform to one of the reduction instances, thus meeting the weight and frequency requirements, such that the resolution of character 1 passes the compatibility check. After character 1 has been processed we proceed to the next character up the priority order, which now plays the role of No. 1, and continue in the same way until the highest priority character has been processed. If any reductions have been made in this first pass through the priority queue, we make a second pass, to allow for cases in which the resolution of a low-priority character depends on one of higher priority having already been resolved. We loop through the priority queue until a pass occurs in which no reductions are performed. Then round i terminates and we enter the next round with the new reduction threshold (as long as $i < m$).

Postprocessing

Often one does not have to worry about the order in which the reduction steps are executed. Where it does matter, the priority order assists in performing the most promising reductions first. Inevitably, the deliberately myopic view of just a few characters at a time may lead to reconstructions that are not fully justifiable in some cases. To compensate partially for unfortunate decisions taken at an early stage of the reduction algorithm, we propose performing an a posteriori search.

Again we employ the given priority order of characters, but now start with the highest priority character of the torso. If this character changes more than once in the reduced median network, then we collapse all links corresponding to the predicted resolutions of this character and then apply the reduction criteria to determine whether we can restore the initial decision. It may happen that some of the hypothesized recurrent mutations can now no longer be separated, and instead the network grows by adding further cycles or cubes to express the ambiguity: see Fig. 8a for a generic instance. To give an example from a real data set, consider the network of "isolated African lineages" from Watson *et al.* (1997, Fig. 3), which was obtained by an early implementation of the reduction procedure as described by Bandelt *et al.* (1995). In this network three mutations are postulated at np 16274: two of

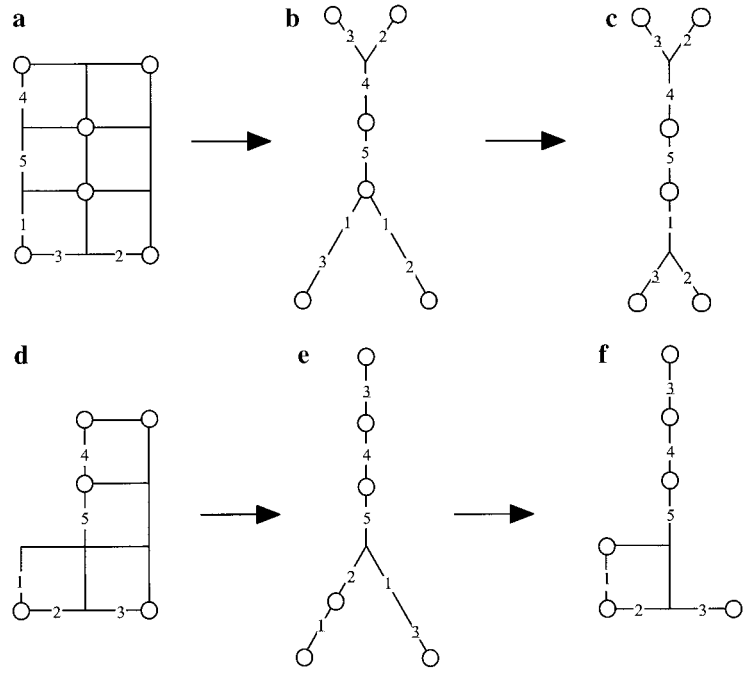

FIG. 8. Two median networks (a) and (d), their greedy reductions with respect to priority orderings given by character numbering (b) and (e), and the resulting networks (c) and (f), after postprocessing.

those, labeled as 16274a and 16274c, are attached to links both incident with a single link corresponding to 16293a. If we focus only on nps 16274a, 16274c, and 16293a, we have a path of length 3. We would thus merge 16274a and 16274c back into one character, which would then create a cube in the African network (since 16360a is incompatible with both 16293a and the merged character 16273a/c). This may better represent the uncertainty that we have about the internal branching of the Western/Central African cluster.

After the highest priority character has been processed and the network possibly expanded, we proceed to the next character in the priority order and continue until the lowest priority character has been processed. If any expansion has been performed along the way, then we pass again through the list of nonperipheral characters and so on until we arrive at a stable network.

Algorithm GRM

- Input: Binary sequences without ambiguities and with weighted positions.
- Preprocessing: Group individuals into haplotypes (recording frequencies) and varied sequence positions into characters (adding up weights).
- Priority: Turn relative mutation rates into a priority order.
- Rounds: Select a sequence of decreasing reduction thresholds $r_1 > r_2 > \ldots > r_m$ at which greedy reduction is executed sequentially.
- Reduction: Seek to resolve the lowest priority character in the torso with the help of the highest priority characters in the torso according to the rules of Fig. 6 (including the compatibility check)—and so forth through the priority order, until no further reductions occur.
- Postprocessing: Check the hypothesized resolution of recurrent mutations for every character of the original torso data matrix against the other resolved characters.
- Shelling: We make three calls to the procedure Greedy Shelling. In the first, the peripheral characters of the original data matrix are sorted into the outer shells 1 to p first. The next shells, $p+1$ to q, are reserved for the peripheral hypothesized characters of the final data matrix, which was obtained from the original torso data matrix in the reduction process. Finally, the remaining hypothesized characters, determining the torso of the GRM network, are placed into the deepest shells $\geq q+1$.
- Network construction: Characters are processed in reverse shelling order (beginning with the deepest shell). Duplicated subnetworks are popped out in parallel for the characters of each shell in turn. After the GRM torso is completed, the final q steps amount to planting the pendant subtrees.

Comparison of RM and GRM

The algorithm GRM differs from the reduced median network method (RM: Bandelt *et al.*, 1995) in five aspects: (i) the reduction threshold r is not fixed to 2, (ii) additional reduction rules (cube and trifolium) are

used, (iii) a compatibility check for the new characters in the updated character set is performed as part of each reduction rule, (iv) reduction instances are executed "on-line" by employing the priority order of characters (thus avoiding the storage and comparison of all possible reduction instances as with RM), and (v) post-processing is performed to weed out obsolete reductions.

SIMULATION STUDY

To assess the performance of the reduction rules, a simulation was undertaken to generate data sets in which the true evolutionary pathways were known a priori; these data sets were subsequently analyzed as described above. Parameters were chosen to mimic in a Mickey Mouse fashion the pattern of HVS I variation in Eurasia. A demography consisting of a sudden expansion of effective population size t generations ago from size N_0 (applicable at all earlier times) to size N_1 (up to the present) was used. The average rate of mutation over nps 16090–16365 was taken from Forster *et al.* (1996) and the relative rates of the different sites in this mtDNA region were taken from the list of Hasegawa *et al.* (1993; their Table 3). The coalescent process (Kingman, 1982) was employed to generate a genealogy ("coalescent tree") of n individuals in this scenario. Mutations were scattered on each branch of this tree as a Poisson process with parameter equal to mutation rate $\times$ branch length, as per the model of infinite sites. Each of these distinct mutations was remapped (to a model of finite sites), by choosing to identify the mutation with a site in the list of Hasegawa *et al.* (1993), selecting the site with a probability proportional to the number of hits that it received in their table. The complete set of mutations was remapped a number of times to investigate the effect of homoplasy in various parts of the tree. For our study, we took $N_0 = 750$, $N_1 = 10{,}000$, and $t = 1000$ generations or 25,000 years. In combination with Forster *et al.*'s (1996) calibration of the HVS I mutation rate, $\mu = 0.00124$ transitions per generation, these values imply that the usual mutation–drift and expansion time parameters (Rogers and Harpending, 1992) here are $\theta_0 = 2\,\mu N_0 = 1.86$, $\theta_1 = 2\,\mu N_1 = 24.8$ and $\tau = 2\,\mu t = 2.48$. The sample size was $n = 50$. The gene tree of the infinite sites data contained 40 segregating sites and 25 haplotypes. Ten mappings to finite sites were performed and GRM networks constructed using the parameter settings $m = 1$, $r_1 = 2$, uniform weights, and a priority order based on the rank of sites in the Hasegawa list.

The realized genealogy coalesced 56,800 years ago, more than twice as far back as the expansion: the expansion occurred on an already diversified background of five founder haplotypes. If we estimate the average number of mutations from the root to every sequence, that is, the statistic ρ (Forster *et al.*, 1996), we obtain a value of 2.68, which translates into an estimated coalescence time of 54,100 years, a happy result, considering that even a lower-bound on the expected root-mean-square error of this value, obtained assuming a perfect star phylogeny (clearly not satisfied here), is $\pm$4700 years. A combined ρ for the five expansion clusters yields an age of 30,700 $\pm$ 3400 years.

Before considering the GRM networks generated from the fully remapped (finite sites) data, we investigate how the reduction rules fare when just a pair of mutations is mapped to the same site. Of all the possible mappings of the 40 mutations in our tree onto 39 sites, 18% lead to hidden mutations (2% are of type 1 and 16% of type 2, referring to the categorization at the beginning of the section "Reduction rules"), 27% to unresolvable squares (as in Fig. 6a, but with $w(1) = w(2)$), 5% to rectangles among which one tenth are unresolvable, and 50% to resolvable ladders; all resolutions were correct. When, however, the mutations are simultaneously mapped onto fewer sites (about 31 sites in our 10 remappings), some interaction takes place that sporadically results in incorrect reductions. The reduction rule which seems likely to be most vulnerable is the rectangle rule (Fig. 6a) as it solely relies on frequencies of observed types. In the GRM networks shown in Figs. 9 and 10a–10i generated from the data for the 10 remappings, we observe two topology errors, where inappropriate reductions have occurred: the branch to L in Fig. 10c and the branch to O in Fig. 10g. The first arises from the rectangle reduction rule (Fig. 6a) and the second from the domino rule (Fig. 6b). One additional topology error is made by the RM algorithm (which otherwise leads to the same networks as GRM) in the data of Fig. 10e, where U becomes detached from V and reattaches to R. A number of events are hidden in the reconstructions, typically at least two per diagram: however, these events, either reversions on a single branch (e.g., from G to L in Fig. 10a) or accidental parallelisms in neighboring branches (e.g., mutation 1 on the branches leading from G to O and H in Fig. 9), would be unreconstructed by almost any method. Disregarding these unreconstructible events, we count eight networks that contain the true tree, although it is usually not among the most parsimonious trees encompassed in the network. The cube reduction rule (Fig. 6c) was invoked four times in all, twice in Fig. 10e and once each in Figs. 9 and 10i. The incompatibility check came into operation twice to postpone otherwise permissible reduction instances (Figs. 10b and 10g), both cases in the context of 2 $\times$ 2 grids of incompatible characters. Only once was a character

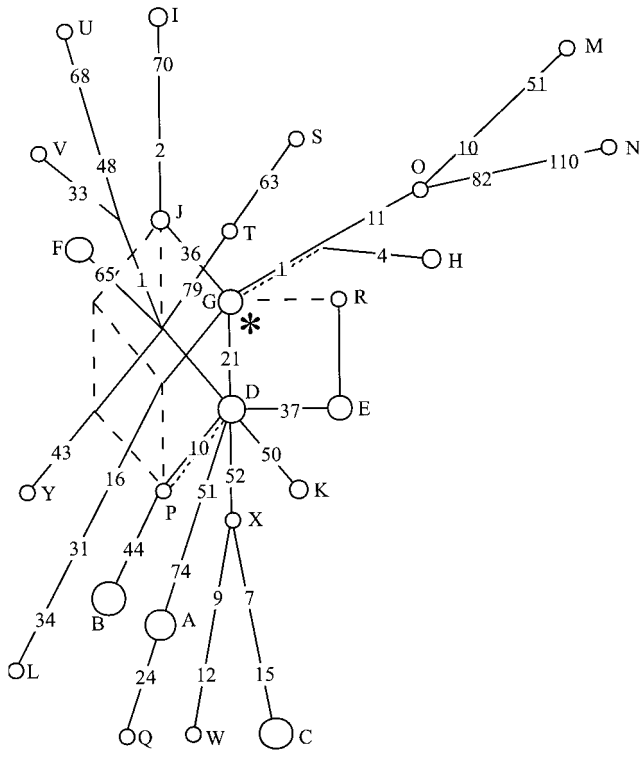

FIG. 9. The reduced median (RM as well as GRM) network for the first simulated data set. Numbers indicate positions in a hypothetical stretch of mtDNA not unlike HVS I, but ordered from fastest (1, 2) to slowest (88–143). Underlining denotes resolved parallel mutation. The circles labeled by letters denote haplotypes, with circle areas proportional to their frequency in the sample. The solid links are branches in the (known) haplotype tree. Lines composed of long dashes are links that are present in the network but not in the true tree, while stippled links are present in the true tree but not in the network and thus indicate hidden events and topology errors. The asterisk labels the root of the network.

that had been resolved refolded by the postprocessing step (Fig. 10b), and in that case, the result was the same network as that generated by RM following quite another route.

Are Cavalli-Sforza and Minch (1997) correct when they claim that "the mtDNA D-loop is probably plagued by noise"? As argued by Richards *et al.* (1997) and as demonstrated here, this is hardly the case. Indeed, the simulation probably exaggerates the lapses since it assumes that (i) the zero hit positions in the Hasegawa list are really invariable and (ii) the effective population size is constant after the sudden expansion. In practice, we are further assisted in many cases by additional knowledge of the RFLP haplogroup status of the founder lineages. If we were to assume such information available here distinguishing the founder lineages, then the situation would dramatically improve: the reduction errors are eliminated, many hidden mutations are resolved, and the three cubes disappear along with almost all cycles.

COMPARISON OF HVS I AND II

Data Recombination

With mitochondrial DNA there is no evidence for recombination *in vivo* (notwithstanding Hagelberg *et al.*, 1999, who chose to explain an enigmatic pattern at a single position by recombination)—and this is why one is willing to perform tree analyses. Recorded data, however, usually have passed a long way from the mitochondria on to the printed page or database, through several virtual cycles of replication. Recombination *in vitro* or *in silico* may very well occur in the presence of contamination or with mistyping. The most likely source for such inadvertent hybridization, however, is the mislabeling of samples whenever distinct shorter stretches need to be read (as with autoradiographs) or sequenced separately (e.g., HVS I plus II or HVS I plus RFLPs).

Potential instances of hybridization are provided by the combined HVS I and RFLP data for Amerindians from Fig. 2 of Santos *et al.* (1996), viz., in those six haplotypes which the authors called "unusual." Haplotype S18/R13 from this study has the same RFLP pattern (R13) as the genuine haplotype S26/R13 from haplogroup B, but its HVS I part (S18) is the exact overlay of the HVS I parts of S26/R13 and the genuine haplotype S32/R15 from haplogroup C. Similarly, the HVS I part of S17/R12 is the exact overlay of the HVS I parts of S22/R13 from haplogroup B and S38/R18 from haplogroup C, whereas the RFLP part comes from S22/R13, except at restriction site 13065, which may have been inherited from S38/R18.

Another good case in question is the Bulgarian data set produced by Calafell *et al.* (1996): one sequence (No. 30) combines the HVS I motif for RFLP haplogroup J with the HVS II motif for haplogroup X, while conversely sequence 31 has the hybrid X/J motif. When we focus only on nps 16069, 16126, 16189, 16223, 16278, and 16293 in HVS I and nps 00153, 00188, 00195, 00225, 00226, 00228, and 00298 in HVS II, the full median network (prior to reduction), relative to these 13 positions, displays a pronounced two-dimensional grid structure, with one dimension corresponding to the HVS I sites and the other to the HVS II sites exclusively: see Fig. 11. The most parsimonious explanation is that the HVS II sequences for individuals 30 and 31 have been interchanged by mistake. After correction, individuals 20 and 31 as well as individuals 30 and 45 would have identical subsequences (with respect to those positions), and the corresponding median network would slim down considerably.

A systematic way to detect such a pattern or other partial row shifts in data tables is to visualize the pairwise incompatibilities (Jakobsen and Easteal, 1996). Alternatively, a thorough search through all quartets of sequences could assist, as is done in statis-

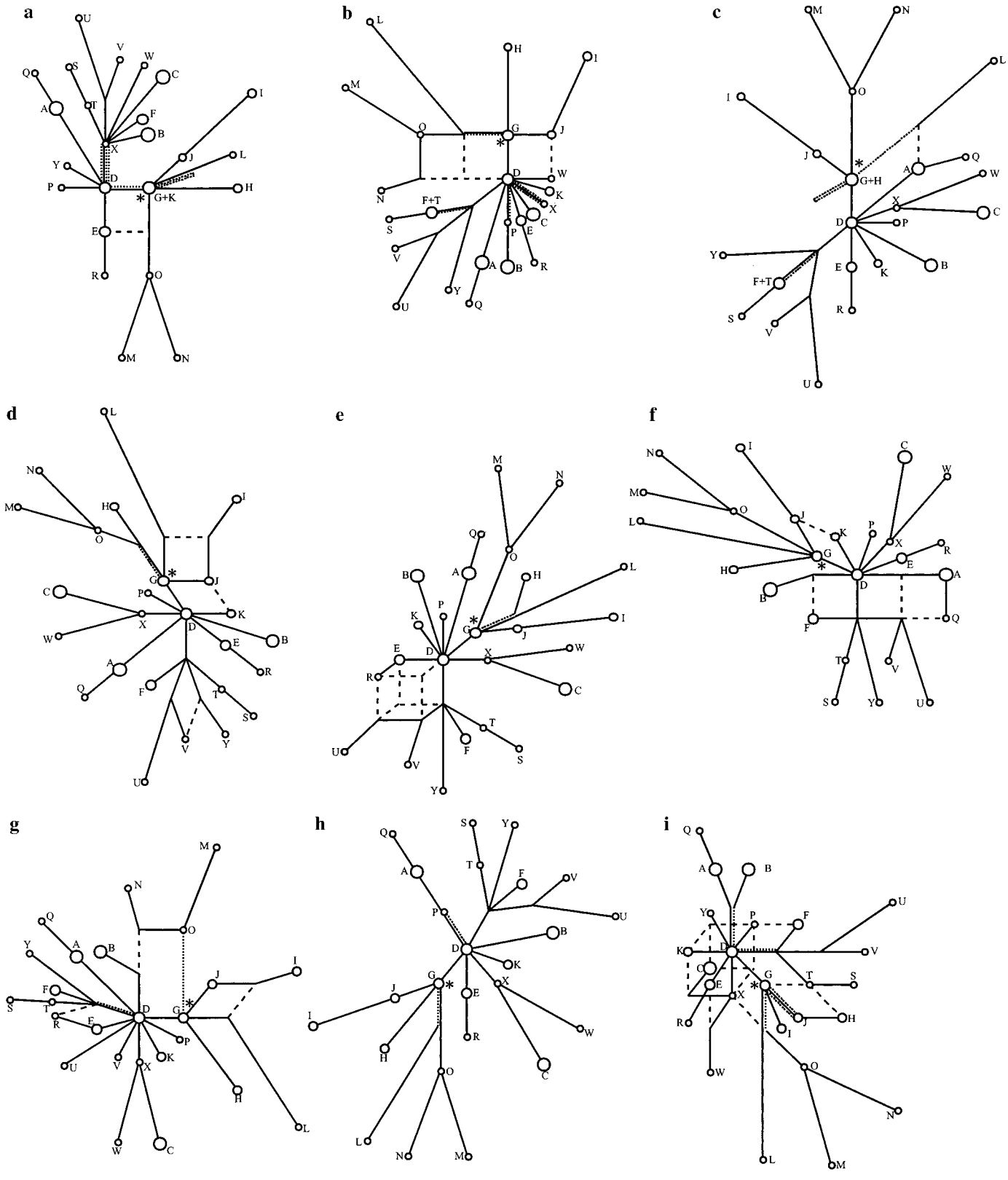

FIG. 10. Reduced median (GRM) networks generated from nine further remappings of the simulated data, as in Fig. 9, except that positions are not labeled. The RM networks are identical, except for case (e), as discussed in the text.

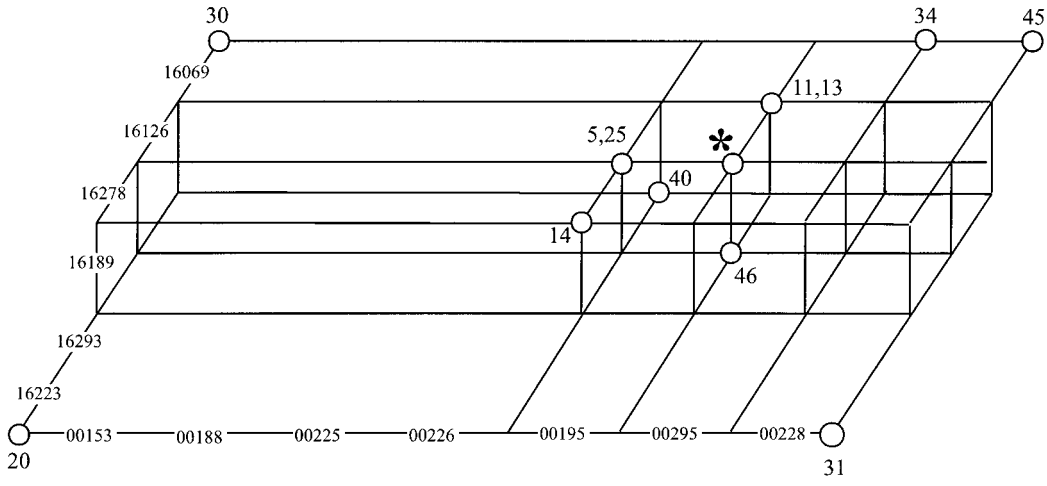

FIG. 11. Median network highlighting an artifactual recombination event between sequences 30 and 31 in the HVS I and II data for Bulgarians from Calafell *et al.* (1996; their Table 1). The node labeled by an asterisk represents the majority of the sequences. Exactly those positions in HVS I and II were selected which are informative (with respect to parsimony) for the quartet Nos. 20, 30, 31, and 45.

tical geometry (Eigen *et al.*, 1988). Four binary sequences can always be represented by an at most three-dimensional cube plus four terminal links. If two of the cube dimensions are supported by more than two sites each, then one should be alarmed and construct the full median network for all sequences truncated to those sites in the cube. In the case of the Bulgarian data, sequences 20, 30, 31, and 45 would give rise to a 6×7 rectangle, which is framing the grid-like network of Fig. 11.

We next reanalyze two small mtDNA data sets for both hypervariable segments, which display typical features of Eurasian control region data.

Bulgarian mtDNA

We subjected 30 Bulgarian sequences from Calafell *et al.* (1996, their Table 1) to greedy reduction, applied to HVS I and HVS II separately. The resulting networks are displayed in Figs. 12 and 13. For HVS I, recurrent mutations were reconstructed at four (highly variable) positions; one cube and one cycle remain unresolved in the network, in both of which highly variable positions are also involved. In view of our simulations, which were tailored to HVS I with a similar amount of variation, we would expect some hidden recurrent mutations. This turns out to be the case when we employ additional information about HVS I motifs for different RFLP haplogroups (Torroni *et al.*, 1996). Sequence 46 probably belongs to haplogroup H, whereas sequences 20 and 31 are members of haplogroup X, and therefore the seemingly shared transition at np 16189 is in fact incurred by two independent events. Similarly, the transition at np 16311 on the way to No. 25 (haplogroup HV: Macaulay *et al.*, 1999) and Nos. 5, 8, 37, and 47 (haplogroup K) has to be resolved into two events. This does not come as a surprise since we regard nps 16189 and 16311 as the two most variable positions in the first segment of Eurasian mtDNAs (cf. Forster *et al.*, 1996). Further, there may be a topology error involving the location of sequences A4, 10, and 24, which becomes evident through a thorough search of the European mtDNA database: np 16270 is characteristic for a major branch (U5: Richards *et al.*, 1998; Torroni *et al.*, 1996) of haplogroup U and should correspond to a basal link branching from the CRS (the node marked by an asterisk in Fig. 12), so that the branch leading to No. 10 must have experienced a back mutation at np 16270. Sequence 24 probably also belongs to haplogroup U5, thus suggesting yet another parallel mutation at 16311 and (oddly enough) another back mutation at np 16270.

Comparing the reduced median network of Fig. 12 with trees estimated using other phylogenetic methods, we find that, for example, this network happens to comprise exactly all most parsimonious reconstructions of the most parsimonious trees. Among those, we also find the most parsimonious reconstructions of the four distinct neighbor-joining trees (Saitou and Nei, 1987), depending on the order in which the sequences are entered. In contrast, the result offered by a complete run of quartet puzzling (Strimmer and von Haeseler, 1996) is somewhat puzzling: the link testifying to the sister group relationship of haplogroups J and T (which is well established: Macaulay *et al.*, 1999) is lost. Quartet puzzling seems to get quickly puzzled by peripheral parallel mutations, which would not present a problem for standard parsimony-based methods. This failure should not be attributed to maximum-likelihood (at which quartet puzzling aims), but rather to the inadequacy of consensus methods based on quar-

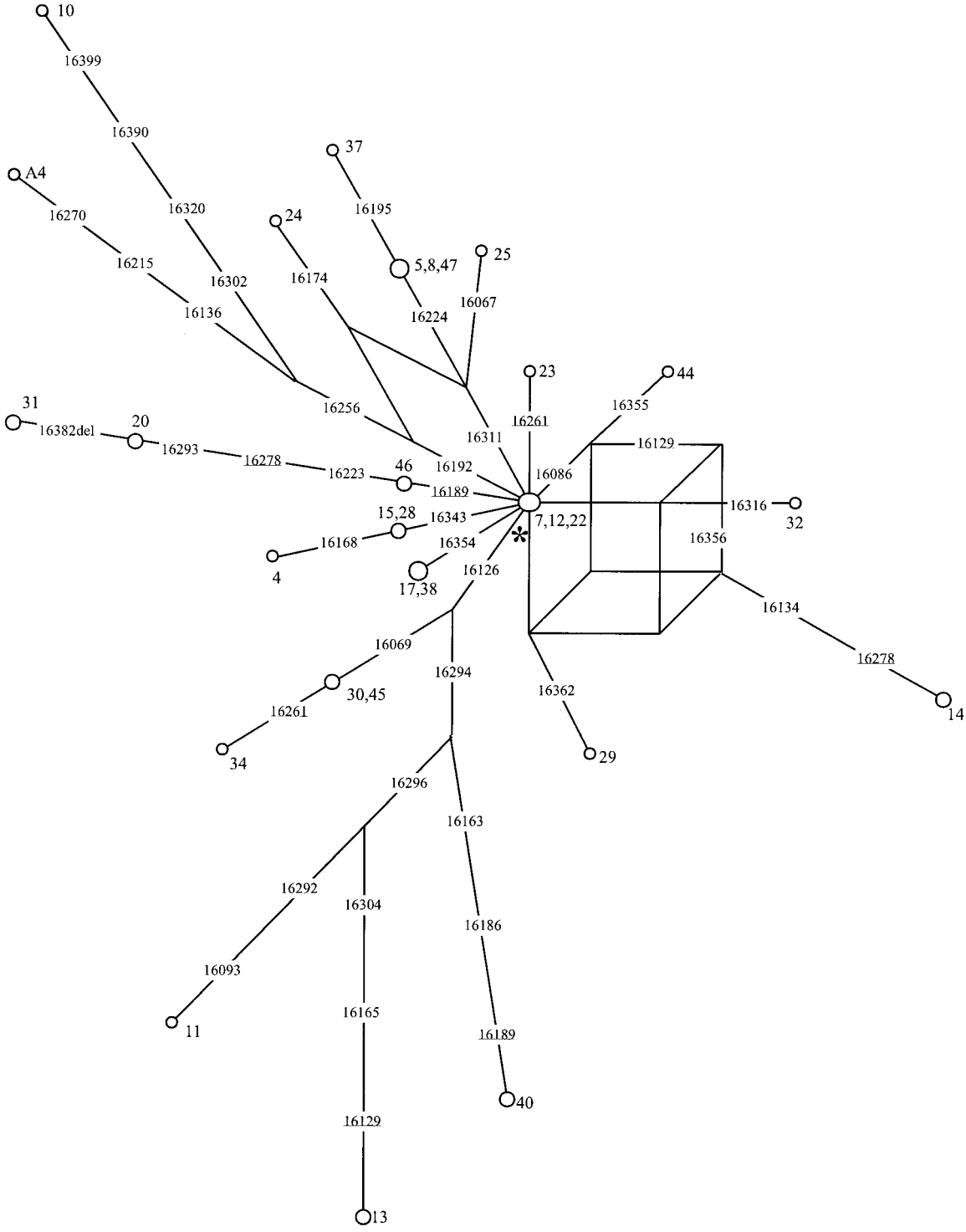

FIG. 12. The reduced median (RM as well as GRM) network for the HVS I data of Bulgarians (Calafell *et al.*, 1996). The node labeled by an asterisk corresponds to the CRS. Underlining indicates reconstructed recurrent mutations.

tets (incorporated in quartet puzzling and split decomposition) for this kind of intraspecific data. Stone and Stoneking (1998) make a similar observation in connection with Asian and Native American HVS I data.

HVS II has little to offer for further clarifying the picture that we obtain with HVS I. Only haplogroups J (Nos. 30, 34, and 45) and X (Nos. 20 and 31) can be recognized by their respective signature positions in HVS II. In contrast, haplogroups K (Nos. 5, 8, 37, and 47) and T (Nos. 11, 13, and 40) are scattered over the

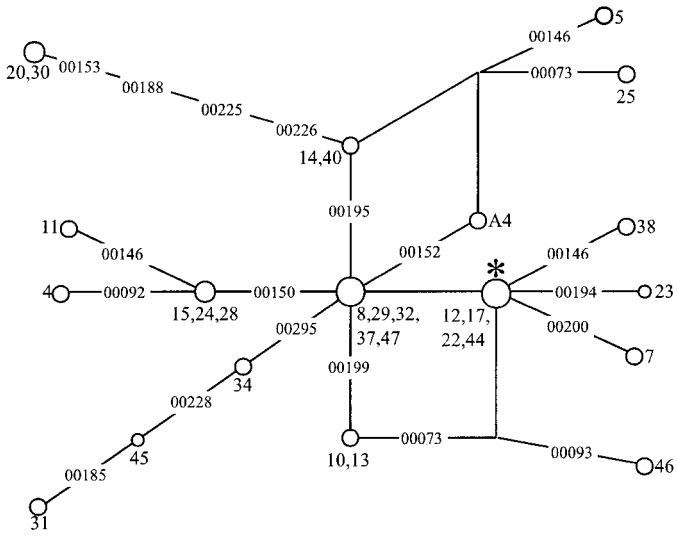

FIG. 13. The reduced median (GRM) network for the HVS II data of Bulgarians (Calafell *et al.*, 1996). The node labeled by an asterisk corresponds to the CRS except at np 00263. Indels in the cytosine tracts near 00309 and 00315 are ignored.

network of Fig. 13. This network (actually composed of the most parsimonious trees) predicts at least one recurrent event at np 00073, in sequence 25. However, this reconstruction does not conform to other information at hand, which suggests that sequences with the HVS I motif 16067T are members of haplogroup HV (e.g., Macaulay *et al.*, 1999). On the other hand, sequence 32 is expected to have undergone a back mutation at 00073, since identical HVS I sequences have been found, for example, in the Mediterranean area, with confirmed haplogroup H status (Torroni *et al.*, 1996 and unpublished data).

Finally, we estimated a tentative tree from the combined data (after correction of the recombination error described above), anticipating RFLP haplogroup status in the light of the worldwide sequence database. We estimate that two mutational hits occurred at nps 16086, 16261, 16278, and 00073, three hits at nps 16129, 16189, 16270, 16311, 00146, 00150, 00152, and 00199, and five hits at np 00195. Note that nps 16129, 16189, and 16311 appear as the three most variable positions of HVS I in the list of Hasegawa *et al.* (1993). Although on average HVS II seems to be somewhat less variable (per position) than HVS I (Harpending *et al.*, 1993), the homoplastic events are more numerous in HVS II but concentrated on fewer sites: see Fig. 14a.

Japanese mtDNA

The data provided by Oota *et al.* (1995) comprise two relatively short segments of HVS I and II, called MT1 and MT3 by these authors, covering nps 16231–16362 and 00146–00263, respectively. The reduced median (GRM) networks for HVS I and II are shown in Figs. 15 and 16. Notice that to resolve the recurrent mutations at np 16311, trifolium reduction (Fig. 6d) had to come into action, while for postulating three mutations at np 00146, cube reduction (Fig. 6c) was needed. The difference between the two networks is considerable, although MT1 and MT3 harbor an almost equal amount of variation: while the network for MT1 constitutes the unique most parsimonious tree, the web-like network for MT3 is composed of more than 2000 most parsimonious trees.

Taking RFLP information (Forster *et al.*, 1996; Torroni *et al.*, 1994) and further Asian HVS I and II data (Lee *et al.*, 1997) into consideration, we would predict that a major part of the branch described by 16362C belongs to haplogroup D, 16304C in conjunction with a deletion at np 00248 characterizes haplogroup F, and the nucleotides 16290T, 16319A, and 00235G together determine haplogroup A. When we put together the MT1 and MT3 data, we estimate a tentative tree of total length 44 steps (not shown); two mutations are inferred at nps 16249, 00195, 00199, and 00207, three mutations at nps 16311, 00146, and 00152, and four mutations at np 00150: see Fig. 14b.

Hot Spots in HVS II

The positions in HVS II which have undergone recurrent mutations in both examples are nps 00146

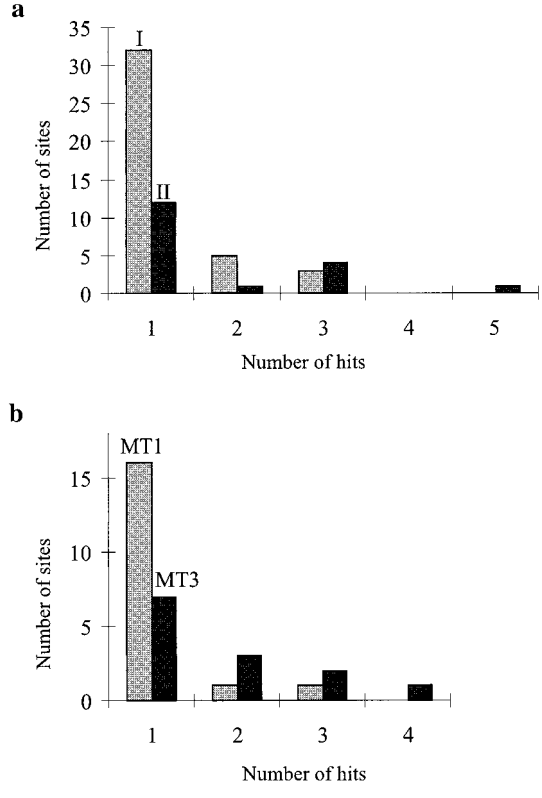

FIG. 14. Numbers of sites in the two segments of the control region with inferred number (1–5) of mutational hits: (a) contrasting HVS I and HVS II in Bulgarians (cf. Figs. 12 and 13); (b) contrasting the subsegments MT1 and MT3 in Japanese (cf. Figs. 15 and 16).

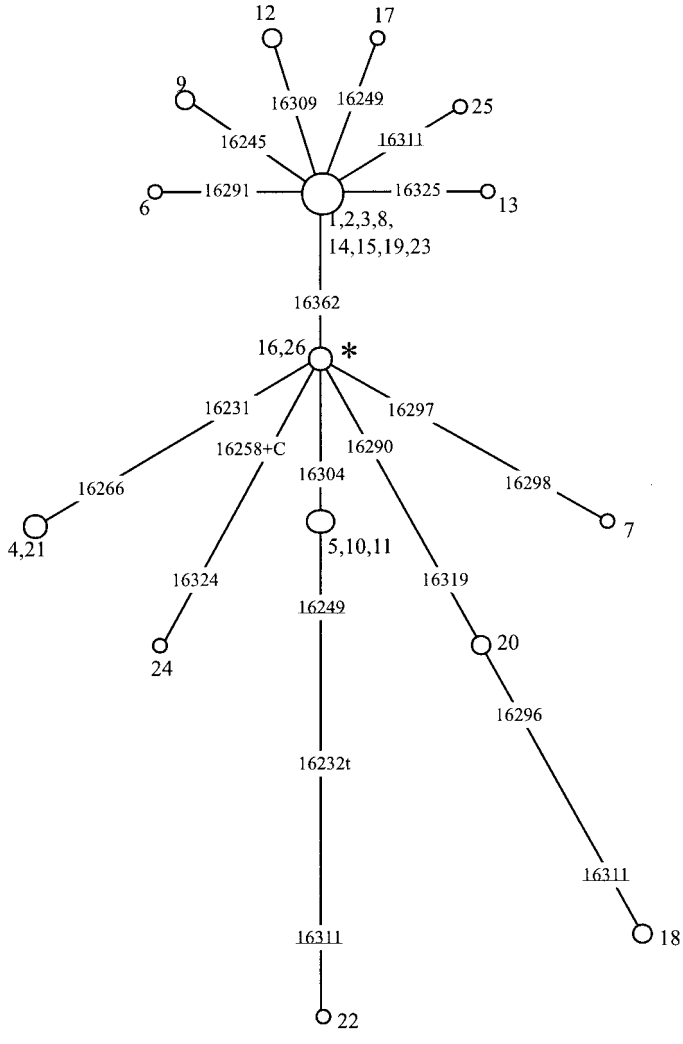

FIG. 15. The tree (GRM network) obtained via greedy reduction from the MT1 data in Japanese (Oota *et al.*, 1995). The node labeled by an asterisk corresponds to the CRS (restricted to MT1).

(with 6 hits), 00150 (7 hits), 00152 (6), 00195 (7), and 00199 (5). They are thus at least as variable as the topmost variable positions nps 16189 and 16311 in HVS I. Three of those sites (nps 00150, 00152, and 00195) have already been identified as fast sites in a European data set (Torroni *et al.*, 1996). To enhance phylogenetic analyses of control region data we tentatively recommend completely ignoring not only length polymorphisms at 16189 and 00309 but also those five hypervariable positions in HVS II and downweighting nps 16189 and 16311 by a factor of $\frac{1}{2}$, say, at least for Eurasian data. A fuller picture will emerge with the analysis of larger data sets.

Although it is not clear whether the same recommendation should be given for African mtDNAs, the removal of those HVS II positions does seem to enhance the analysis of the pooled HVS I and II data in the case of the !Kung. In fact, three of the five positions in HVS II which give rise to incompatibilities with the HVS I data would be dismissed (00150, 00152, and 00309+C). The other two (00198 and 00207) are of weight 1 and are mutually incompatible. The single most parsimonious tree with respect to this weighting scheme (which is identical to the reduced median network) postulates two hits at nps 00207 and 16243. Incidentally, the resolution of character 16243 into two events (which is also predicted on the basis just of HVS I) is even more compelling in the light of more Khoisan data (see Bandelt and Forster, 1997; their Fig. 2).

DISCUSSION

For intraspecific data, free of recombination, the inference of phylogenetic trees should preferably depart from an explicit partial reconstruction of character evolution rather than brachiating through the jungle of all dichotomous trees. The identification of the most salient parallel events may be effective enough to shrink the plausible solution space considerably, so that it can well be presented in the form of a median network. Further information can then assist in sorting out the most likely evolutionary pathways. We have chosen criteria based on (molecular) weighted parsimony which are conveniently checked in both manual and computer-assisted searches.

Simulations of data mimicking typical HVS I variation in Eurasians have demonstrated that reduced median networks present a fair picture of the phylogenetic relationships, thereby admitting that some minor details are virtually unreconstructible by whatever method is used, unless additional diagnostic characters from the coding region are incorporated.

As for phylogenetic inference, HVS II is generally inferior to HVS I as it contains too few sites with moderate mutation rates and seems to contain more hypervariable sites than HVS I. Nevertheless, a few nucleotide positions in HVS II appear to be highly informative as they concur with RFLP haplogroup signatures (see Torroni *et al.*, 1996 for European data). The message clearly is: use as much information as possible but avoid swamping the reliable signals with noisy hypervariable sites.

What goes for mtDNA will also hold for Y chromosome data, although we still lack a sufficient number of identified polymorphisms with low to moderate mutation rates—but there is a beginning (Jobling *et al.*, 1997). Data for nuclear genes (e.g., Harding *et al.*, 1997) could be treated similarly; the emphasis here, however, would be systematically to detect and partially to reconstruct any recombinations which may have hit certain stretches of the gene sequences.

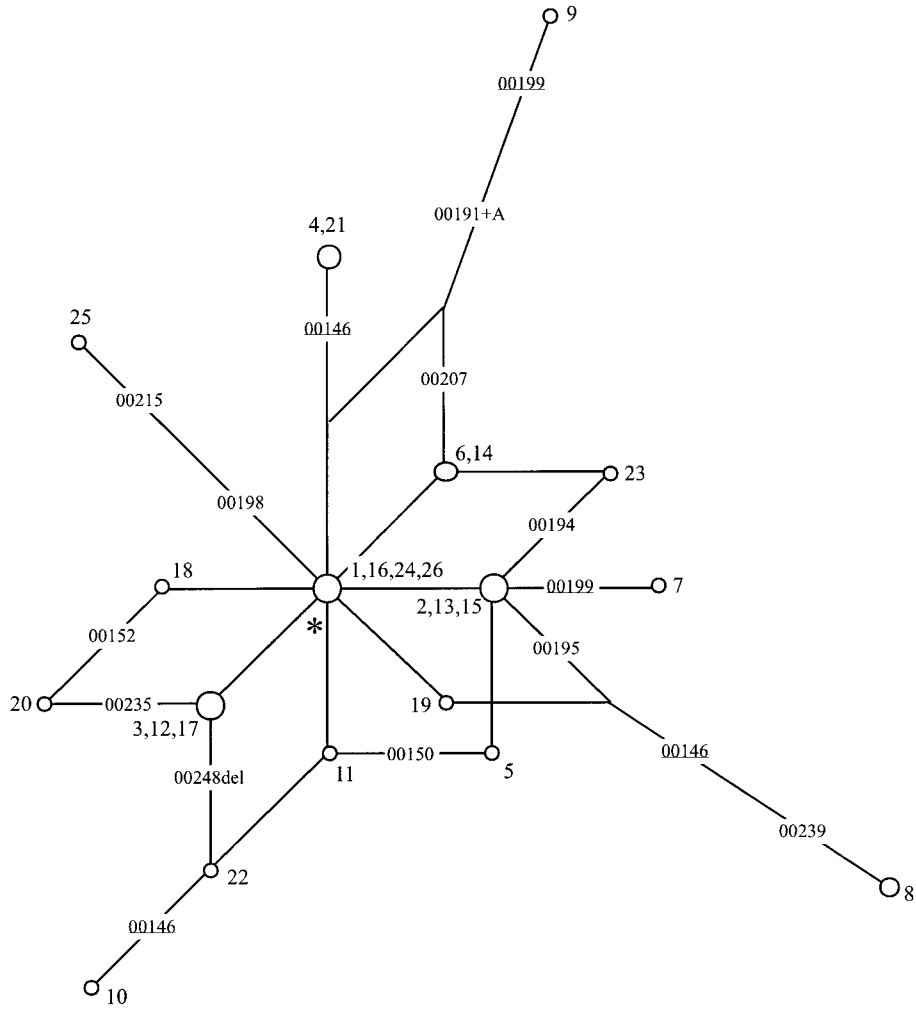

FIG. 16. The reduced median (GRM) network obtained via greedy reduction from the MT3 data in Japanese (Oota *et al.,* 1995). The node labeled by an asterisk corresponds to the CRS except at np 00263.

ACKNOWLEDGMENTS

V.M. and M.R. were supported by The Wellcome Trust and H.-J.B. by a travel grant from the DAAD. We thank Arne Röhl for critical reading of the manuscript.

REFERENCES

Anderson, S., Bankier, A. T., Barrell, B. G., de Bruijn, M. H. L., Coulson, A. R., Drouin, J., Eperon, I. C., Nierlich, D. P., Roe, B. A., Sanger, F., Schreier, P. H., Smith, A. J. H., Staden, R., and Young, I. G. (1981). Sequence and organization of the human mitochondrial genome. *Nature* **290:** 457–465.

Aris-Brosou, S., and Excoffier, L. (1996). The impact of population expansion and mutation rate heterogeneity on DNA sequence polymorphism. *Mol. Biol. Evol.* **13:** 494–504.

Bandelt, H.-J. (1994). Phylogenetic networks. *Verhandl. Naturwiss. Vereins Hamburg* **34:** 51–71.

Bandelt, H.-J., and Dress, A. W. (1992). Split decomposition: A new and useful approach to phylogenetic analysis of distance data. *Mol. Phylogenet. Evol.* **1:** 242–252.

Bandelt, H.-J., and Forster, P. (1997). The myth of bumpy hunter-gatherer mismatch distributions. *Am. J. Hum. Genet.* **61:** 980–983.

Bandelt, H.-J., Forster, P., and Röhl, A. (1999). Median-joining networks for inferring intraspecific phylogenies. *Mol. Biol. Evol.* **16:** 37–48.

Bandelt, H.-J., Forster, P., Sykes, B. C., and Richards, M. B. (1995). Mitochondrial portraits of human populations using median networks. *Genetics* **141:** 743–753.

Bandelt, H.-J., and van de Vel, M. (1989). Embedding topological median algebras in products of dendrons. *Proc. London Math. Soc. (3)* **58:** 439–453.

Bandelt, H.-J., and van de Vel, M. (1991). Superextensions and the depth of median graphs. *J. Combin. Theory Ser. A* **57:** 187–202.

Barthélemy, J.-P. (1989). From copair hypergraphs to median graphs with latent vectors. *Discrete Math.* **76:** 9–28.

Bendall, K. E., and Sykes, B. C. (1995). Length heteroplasmy in the first hypervariable segment of the human mtDNA control region. *Am. J. Hum. Genet.* **57:** 248–256.

Bergman, G. M. (1977). On the existence of subalgebras of direct

products with prescribed *d*-fold projections. *Algebra Universalis* **7:** 341–356.

Calafell, F., Underhill, P., Tolun, A., Angelicheva, D., and Kalaydjieva, L. (1996). From Asia to Europe: Mitochondrial DNA sequence variability in Bulgarians and Turks. *Ann. Hum. Genet.* **60:** 35–49.

Cavalli-Sforza, L. L., and Minch, E. (1997). Paleolithic and neolithic lineages in the European mitochondrial gene pool. *Am. J. Hum. Genet.* **61:** 247–251.

Donnelly, P., and Tavaré, S. (1986). The age of alleles and a coalescent. *Adv. Appl. Prob.* **18:** 1–19.

Eigen, M., Winkler-Oswatitsch, R., and Dress, A. (1988). Statistical geometry in sequence space—A method of quantitative sequence analysis. *Proc. Natl. Acad. Sci. USA* **85:** 5913–5917.

Fitch, W. M. (1996). The variety of human virus evolution. *Mol. Phylogenet. Evol.* **5:** 247–258.

Fitch, W. M. (1997). Networks and viral evolution. *J. Mol. Evol.* **44:** S65–S75.

Forster, P., Harding, R., Torroni, A., and Bandelt, H.-J. (1996). Origin and evolution of Native American mtDNA variation: A reappraisal. *Am. J. Hum. Genet.* **59:** 935–945.

Hagelberg, E., Goldman, N., Lió, P., Whelan, S., Schiefenhövel, W., Clegg, J. B., and Bowden, D. K. (1999). Evidence for mitochondrial DNA recombination in a human population of island Melanesia. *Proc. R. Soc. Lond. B* **266:** 485–492.

Harding, R. M., Fullerton, S. M., Griffiths, R. C., Bond, J., Cox, M. J., Schneider, J. A., Moulin, D. S., and Clegg, J. B. (1997). Archaic African *and* Asian lineages in the genetic ancestry of modern humans. *Am. J. Hum. Genet.* **60:** 772–789.

Harpending, H. C., Sherry, S. T., Rogers, A. R., and Stoneking, M. (1993). The genetic structure of ancient human populations. *Curr. Anthropol.* **34:** 483–496.

Hasegawa, M., Di Rienzo, A., Kocher, T. D., and Wilson, A. C. (1993). Toward a more accurate time scale for the human mitochondrial DNA tree. *J. Mol. Evol.* **37:** 347–354.

Jakobsen, I. B., and Easteal, S. (1996). A program for calculating and displaying compatibility matrices as an aid in determining reticulate evolution in molecular sequences. *CABIOS* **12:** 291–295.

Jobling, M. A., Pandya, A., and Tyler-Smith, C. (1997). The Y chromosome in forensic analysis and paternity testing. *Int. J. Legal Med.* **110:** 118–124.

Kingman, J. F. C. (1982). On the genealogy of large populations. *J. Appl. Prob.* **19A:** 27–43.

Lee, S. D., Shin, C. H., Kim, K. B., Lee, Y. S., and Lee, J. B. (1997). Sequence variation of mitochondrial DNA control region in Koreans. *Foren. Sci. Int.* **87:** 99–116.

Macaulay, V., Richards, M., Hickey, E., Vega, E., Cruciani, F., Guida, V., Scozzari, R., Bonné-Tamir, B., Sykes, B., and Torroni, A. (1999). The emerging tree of west Eurasian mtDNAs: A synthesis of control-region sequences and RFLPs. *Am. J. Hum. Genet.* **64:** 232–249.

Meacham, C. A. (1981). A manual method for character compatibility analysis. *Taxon* **30:** 591–600.

Meacham, C. A., and Estabrook, G. F. (1985). Compatibility methods in systematics. *Annu. Rev. Ecol. Syst.* **16:** 431–446.

Oota, H., Saitou, N., Matsushita, T., and Ueda, S. (1995). A genetic study of 2,000-year-old human remains from Japan using mitochondrial DNA sequences. *Am. J. Phys. Anthropol.* **98:** 133–145.

Richards, M., Côrte-Real, H., Forster, P., Macaulay, V., Wilkinson-Herbots, H., Demaine, A., Papiha, S., Hedges, R., Bandelt, H.-J., and Sykes, B. (1996). Paleolithic and neolithic lineages in the European mitochondrial gene pool. *Am. J. Hum. Genet.* **59:** 185–203.

Richards, M., Macaulay, V., Sykes, B., Pettitt, P., Hedges, R., Forster, P., and Bandelt, H.-J. (1997). Reply to Cavalli-Sforza and Minch. *Am. J. Hum. Genet.* **61:** 251–254.

Richards, M. B., Macaulay, V. A., Bandelt, H.-J., and Sykes, B. C. (1998). Phylogeography of mitochondrial DNA in western Europe. *Ann. Hum. Genet.* **62:** 241–260.

Rogers, A. R., and Harpending, H. (1992). Population growth makes waves in the distribution of pairwise genetic differences. *Mol. Biol. Evol.* **9:** 552–569.

Saitou, N. (1998). Simultaneous sequence joining (SSJ): A new method for construction of phylogenetic networks of related sequences. Abstract S-6-5. *Anthropol. Sci.* **106:** 41.

Saitou, N., and Nei, M. (1987). The neighbor-joining method: A new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.* **4:** 406–425.

Santos, S. E. B., Ribeiro-dos-Santos, A. K. C., Meyer, D., and Zago, M. A. (1996). Multiple founder haplotypes of mitochondrial DNA in Amerindians revealed by RFLP and sequencing. *Ann. Hum. Genet.* **60:** 305–319.

Stone, A. C., and Stoneking, M. (1998). mtDNA analysis of a prehistoric Oneota population: Implications for the peopling of the New World. *Am. J. Hum. Genet.* **62:** 1153–1170.

Stoneking, M., Hedgecock, D., Higuchi, R. G., Vigilant, L., and Erlich, H. A. (1991). Population variation of human mtDNA control region sequences detected by enzymatic amplification and sequence-specific oligonucleotide probes. *Am. J. Hum. Genet.* **48:** 370–382.

Strimmer, K., and von Haeseler, A. (1996). Quartet puzzling: A quartet maximum-likelihood method for reconstructing tree topologies. *Mol. Biol. Evol.* **13:** 964–969.

Torroni, A., Huoponen, K., Francalacci, P., Petrozzi, M., Morelli, L., Scozzari, R., Obinu, D., Savontaus, M.-L., and Wallace, D. C. (1996). Classification of European mtDNAs from an analysis of three European populations. *Genetics* **144:** 1835–1850.

Torroni, A., Miller, J. A., Moore, L. G., Zamudio, S., Zhuang, J. G., Droma, T., and Wallace, D. C. (1994). Mitochondrial DNA analysis in Tibet: Implications for the origin of the Tibetan population and its adaptation to high altitude. *Am. J. Phys. Anthropol.* **93:** 189–199.

Vigilant, L., Pennington, R., Harpending, H., Kocher, T. D., and Wilson, A. C. (1989). Mitochondria DNA sequences in single hairs from a southern African population. *Proc. Natl. Acad. Sci. USA* **86:** 9350–9354.

Watson, E., Forster, P., Richards, M., and Bandelt, H.-J. (1997). Mitochondrial footprints of human expansions in Africa. *Am. J. Hum. Genet.* **61:** 691–704.

Wilkinson-Herbots, H., Richards, M., Forster, P., and Sykes, B. (1996). Site 73 in hypervariable region II of the human mitochondrial genome and the origin of European populations. *Ann. Hum. Genet.* **60:** 499–508.