Journal of Classification 9:91-116 (1992)

The Complexity of Reconstructing Trees from Qualitative Characters and Subtrees

Michael Steel

Zentrum für interdisziplinäre Forschung der Universität Bielefeld

Abstract: In taxonomy and other branches of classification it is useful to know when tree-like classifications on overlapping sets of labels can be consistently combined into a parent tree. This paper considers the computational complexity of this problem. Recognizing when a consistent parent tree exists is shown to be intractable (NP-complete) for sets of unrooted trees, even when each tree in the set classifies just four labels. Consequently determining the compatibility of qualitative characters and partial binary characters is, in general, also NP-complete. However for sets of rooted trees an algorithm is described which constructs the "strict consensus tree" of all consistent parent trees (when they exist) in polynomial time. The related question of recognizing when a set of subtrees uniquely defines a parent tree is also considered, and a simple necessary and sufficient condition is described for rooted trees.

Keywords: Trees; Qualitative characters; Compatibility; Resolved quartets; Clusters; Strict consensus tree.

1. Introduction

Suppose we are given a "sample" collection P of trees, each having its degree-one vertices labeled by a subset of a parent label set S. The following

---

This work was supported by the Alexander von Humoldt-Stiftung. I wish to thank Andreas Dress, Hans-Jürgen Bandelt and the referees for their helpful comments.

Author's address: Michael Steel, Z.I.F der Universität Bielefeld, Wellenberg 1, D-4800, Bielefeld 1, Germany.

question is addressed: is there an efficient way to determine whether there exists a parent tree T whose degree-one vertices are labeled from S, and which induces all the branchings occuring in each tree in P? Furthermore, if so, how can T, or the (strict) consensus of all trees consistent with P, be constructed? These questions arise in a number of contexts, in particular in taxonomy (Buneman 1974, Gordon 1986) and linguistics (Dekker 1986), and they are closely related to the question of the compatibility of qualitative characters in taxonomy. They can, of course, be "solved" by checking all possible parent trees, or even just all possible parent binary trees, against each tree in P.

However such an approach quickly becomes unfeasible as the number of objects being classified increases, since the number of binary trees on $n + 2$ labels is asymptotically proportional to $\frac{n!2^n}{\sqrt{n}}$. An algorithm for testing compatibility (and building a tree if one exists) has been devised by Dekker (1986), but it has exponential complexity. An important question then is whether this bound can be substantially improved — in particular whether polynomial-time algorithms exist for this problem.

We show that in general the problem of determining compatibility belongs to the class of NP-complete problems, for which, it is believed, polynomial-time algorithms do not exist (for a discussion of NP-completeness see, for example, Garey and Johnson 1979). Indeed NP-completeness holds even for samples of binary trees on label sets of size four. This contrasts with the special case in which all the sample trees have a label in common (so that the trees can be thought of as rooted) for which a polynomial-time algorithm exists, due to Aho, Savig, Szymanski, and Ullman (1981). A direct extension of this algorithm provides a simple, polynomial-time procedure for constructing the strict consensus tree of all rooted parent trees consistent with a sample of rooted trees, answering a question posed by Gordon (1986). This problem has also been independently solved by Kant-Antonescu and Sankoff (1991).

The compatibility question for samples of trees is equivalent to that of determining the compatibility of qualitative (taxonomic) characters (also called unordered characters). Thus, this latter problem is also NP-complete, a result which has been independently established by Warnow (1991). This is a considerable strengthening of part of a well-known result by Day and Sankoff (1986), who showed that the problem of deciding whether a set of qualitative characters has a compatible subset of size k (for variable k) is NP-complete. This result does not, however, entail NP-completeness for the more basic question of the compatibility of the entire set of characters.

The question of the theoretical complexity of this latter problem has been raised by Meacham (1983) who proposed an algorithm as a partial solution. Buneman (1974) posed a related question: given a set of compatible

characters, is there a "simple" method for constructing a tree consistent with the characters? A related question, discussed in Section 5, has been raised by Colonius and Schulze (1981). The compatibility question can also be regarded as a natural extension of the characterization of tree-like "neighbours relations," discussed by Bandelt and Dress (1986), when these relations are defined only for certain quartets. Indeed this problem forms the link between the compatibility question for trees and qualitative characters. Consequently much of this paper is devoted to bringing together and summarizing different results related to these problems. Note that the problems considered in this paper, while appearing similar in spirit to those in Brossier (1990), do not really overlap since the classification problems considered here do not require or involve ultrametric distance matrices.

2. Compatibility and Consistency

We begin with some definitions — for standard graph-theoretic concepts the reader is referred to Bondy and Murty (1976).

Definitions (1). Given a label set L a *phylogenetic tree* on L is a tree without vertices of degree two, and exactly $|L|$ leaves (a leaf is a vertex of degree 1) each labeled with a distinct element of L. Vertices of degree greater than 1, and edges incident with pairs of such vertices are said to be *internal.* If the internal vertices all have degree 3, the tree is called a *binary (phylogenetic) tree* on L.

Given a phylogenetic tree T on L, and a subset S of L, let $T_{|S}$ denote the induced phylogenetic tree on S consisting of the minimal subtree of T connecting the vertices labeled from S, and having all degree-two vertices suppressed, as illustrated in Figure 1. Given two phylogenetic trees t', t on S write $t' \to t$ if t can be obtained by contracting certain edges of t'. With T and t as given we say T is *consistent with* t if $T_{|S} \to t$. Clearly if t is binary this just means $T_{|S} = t$; extending this concept to phylogenetic trees follows Constantinescu and Sankoff (1986), except here "consistent" replaces their "compatible." Informally, T is consistent with t precisely if T contains all the branching information in t. Figure 1 illustrates this notion.

For a set $P = \{t_1, \ldots, t_r\}$, where t_i is a phylogenetic tree on S_i, P is *compatible* precisely if there is a phylogenetic tree T on $S_1 \cup \cdots \cup S_r$ which is consistent with each tree in P. In this case T and P are said to be *consistent.*

Note that if P is compatible then P is consistent with a binary phylogenetic tree by a standard vertex/binary tree replacement argument; see for example, Constantinescu and Sankoff (1986). Thus we let $<P>$ denote the set of binary phylogenetic trees consistent with P. In case T is the only

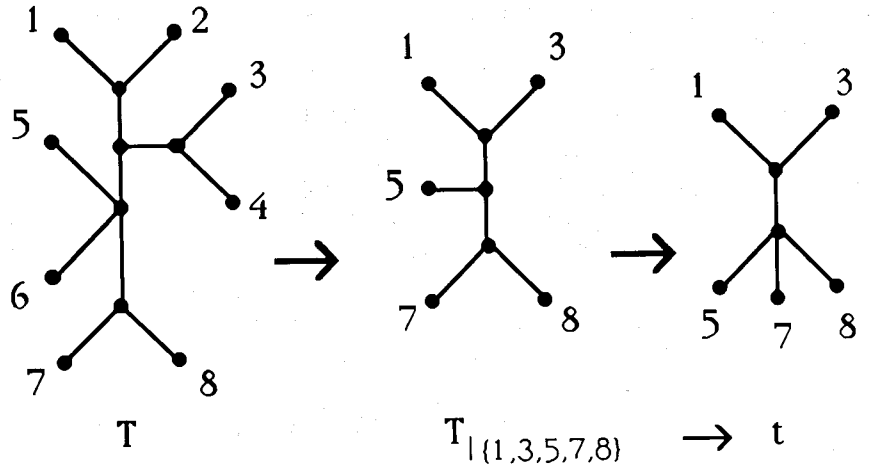

Figure 1. Transformations showing that T is consistent with t.

phylogenetic tree consistent with P, we say P *defines* T (thus T must be binary).

There is a corresponding notion of compatibility for sets of partitions of L, which we call *qualitative characters*. More generally we call a partition of a subset of L a *(partial) character* and the sets in each character are called *states*. For a set C of partial characters we let L_C denote the labels which appear in a state of at least one character in C.

To define compatibility for sets of characters, suppose we are given a tree T (not necessarily binary) with vertex set V, a labelling function $f: L \to V$ and subsets A,B of L. Write $A \mid_T B$ if the the two minimal subtrees of T connecting $f(A)$ and $f(B)$ respectively have no vertices in common. By convention a character X is said to have *convex states* on T if for *all* distinct states A,B of X, $A \mid_T B$. For example the character $\{\{1,2\}, \{3,4\}, \{5,6,7,8\}\}$ has convex states on the tree T in Figure 1. A collection C of characters is *compatible* precisely if there exists a tree T and labelling function f so that each character has convex states on T. In this case T and C are said to be *consistent*.

Again it is easily shown that if C is compatible then C is consistent with a binary phylogenetic tree (with its implicit labelling function), as observed by Dekker (1986). Thus we let $<C>$ denote the set of binary phylogenetic trees consistent with C. Also by adjoining singleton states, a set of partial characters C can trivially be extended to a set of qualitative characters C' so that $<C> = <C'>$.

Note that in taxonomy the term "qualitative character" refers not to a partition but to a function which assigns a characteristic to each taxon in the

set under study. For example, in taxonomic applications based on DNA sequences, these functions — one for each site on a sample of aligned segments of DNA — identify which of the four nucleotides, A,T,C,G, occurs for each taxon. However, such a function defines a partition of the taxa set (into states) by considering, for each characteristic, the subset of the taxa set which are assigned that characteristic. These induced partitions neglect some biologically relevant information but suffice to consider questions of "compatibility" and for this reason we have defined qualitative characters as above. Note also that the ordering of the partitions (in the DNA-case, arising from the natural linear order) is also unimportant here.

The desire for a classification tree to be consistent with a set of characters C derived in this way is motivated by parsimony: Roughly speaking, a tree which is consistent with C is one for which the characteristics which induce the partitions in C could have "evolved" along the edges of the tree in such a way that each possible change in a characteristic (genetic mutation in taxonomy) occurs at most once. For further discussion see Meacham and Duncan (1987), or McMorris (1975).

A character consisting of just two states A,B is called a *binary character* or *split* (of $A \cup B$). A split $\{A,B\}$ of L has convex states on a phylogenetic tree T, on L, precisely if $\{A,B\}$ is a split *induced by* T, in the sense that deleting an edge of T partitions the labels on the leaves of T into the sets A,B. In this way the splits of L having convex states on T are in a one-to-one correspondence with the edges of T; indeed any phylogenetic tree is characterized by the collection of splits it induces (Buneman 1971; Bandelt and Dress 1986). Thus for a binary phylogenetic tree T having n labels there are precisely $2n-3$ splits of the labels set having convex states on T. More generally the following result summarizes enumeratively how convexity confines the classes of binary trees and characters.

Proposition 1.

(1) Let $b(n)$ denote the number of binary trees on a label set of size n. Then

$$b(n+2) = \frac{(2n)!}{n!2^n} = (2n-1)\times(2n-3)\times\cdots\times 3\times 1\,.$$

(2) For a phylogenetic tree t on L' the number of binary trees T on $L \supseteq L'$ which are consistent with t is

$$\frac{b(|L|)}{b(|L'|)} \times \prod_{v} b(\partial(v))\,,$$

where v ranges over all internal vertices of t having degree $\partial(v) > 3$.

(3) For a partial character, $X = \{A_1, \ldots, A_r\}$ *the number of binary trees on L for which X has convex states, is*

$$\prod_{i=1}^{r} b(|A_i| + 1) \times \frac{b(|L|)}{b(n-r+2)}, \quad n = \sum_i |A_i| .$$

(4) For any binary tree T on L the number of qualitative characters having $s > 1$ *convex states on T is* $\begin{bmatrix} 2|L| - s - 1 \\ s - 1 \end{bmatrix}$.

Proof. Result (1) is due to Cavalli-Sforza and Edwards (1967) (for an interesting bijective proof see Erdös and Székely 1989). Result (2) is essentially due to F.J. Rohlf (see also Theorem 1 of Constantinescu and Sankoff 1986). Result (3) is Theorem 2 of Carter, Hendy, Penny, Székely and Wormald (1990), combined with (2). To establish (4), given a binary tree T let $f(T,s)$ denote the number of qualitative characters having s convex states on T. Select a vertex adjacent to two leaves u,v and let $T' = T_{|L-\{v\}}$, $T'' = T_{|L-\{u,v\}}$. Partition the set of qualitative characters having s convex states on T into three classes:

C_1: Those containing states $\{u\}$ and $\{v\}$.
C_2: Those for which v occurs in the same state as u.
C_3: Those containing states $\{u\}$ or $\{v\}$, but not both.

We have $|C_1| = f(T'',s-2)$, $|C_2| = f(T',s)$, $|C_3| = 2(f(T',s-1) - |C_1|)$. In this way $f(T,s) = f(T',s) + 2f(T',s-1) - f(T'',s-2)$. By induction on $|L|$, $f(T,s)$ depends only on $|L|$ and s, and not the tree T chosen. The result can now be derived from (3) by using Lemma 4 of Carter et al. (1990) to count pairs (T,X) where X is a qualitative character having s convex states on T. Alternatively, the result can simply be verified by inserting the claimed formula into the above recursion for $f(T,s)$. •

We now describe the relationship between the two types of compatibility described above — compatibility of trees and of characters — by considering binary trees on *quartets* (label sets of size four). Such trees are called *resolved quartets*. Given a set C of partial characters we construct a set of resolved quartets $q(C)$ with $<q(C)> = <C>$. Also, given a set P of phylogenetic trees we construct a set of partial binary characters $c(P)$ and resolved quartets $q(P)$ with $<P> = <c(P)> = <q(P)>$. Thus compatibility questions for phylogenetic trees, characters and resolved quartets are essentially equivalent. Furthermore, these constructions can be efficiently implemented,

and with $|q(C)|$ $|c(P)|$ and $|q(P)|$ having the following orders of magnitude: $O(|C| \times |L|^2)$, $O(|L| \times |P|)$ and $O(|L| \times |P|)$, respectively.

Definitions (2). For a resolved quartet t on $\{a,b,c,d\}$ we write $ab|cd$ if $\{a,b\}$ $|_t$ $\{c,d\}$. For example, for the tree T in Figure 1, the resolved quartet $T_{|\{1,2,3,4\}}$ is 12|34. Given a set of partial characters, $C = \{X_1, \ldots, X_r\}$, where $X_i = \{A_{i1}, \ldots, A_{is(i)}\}$ select $\alpha_{it} \in A_{it}$ for $1 \le t \le s(i)$, $1 \le i \le r$, and let $q(C) = \cup_{i,j,k} \{\alpha_{ij}a|\alpha_{ik}a' : a \in A_{ij}, a' \in A_{ik}\}$. For a set P of phylogenetic trees let $c(P)$ denote the set of all splits induced by trees in P.

Proposition 2.

(1) $<q(C)> = <C>$

(2) $<c(P)> = <P>$

(3) There is a set $q(P)$ of resolved quartets with $<q(P)> = <P>$ and

$$|q(P)| \le \sum_{t \in P} (-1 + \sum_{v} (\partial(v) - 2))$$

(where v ranges over all internal vertices of t). Furthermore, if any label x appears in all trees in P, $q(P)$ can be chosen so that x is a label of each tree in $q(P)$.

Proof. (1) Suppose $T \in <C>$ and $\alpha_{ij}a|\alpha_{ik}a' \in q(C)$. By definition of $q(C)$ there exist states $A_{ij}, A_{ik} \in X_i \in C$ with $\alpha_{ij}, a \in A_{ij}$, $\alpha_{ik}, a' \in A_{ik}$. Since X has convex states on T, the path in T connecting α_{ij} and a is disjoint from the path in T connecting α_{ik} and a' thus $T_{|\{\alpha_{ij},\alpha_{ik},a,a'\}} = \alpha_{ij}a|\alpha_{ik}a'$. Since this holds for all resolved quartets in $q(C)$, we have $T \in <q(C)>$. Conversely, suppose $T \in <q(C)>$. We show $T \in <C>$ by deriving a contradiction from the assumption $T \notin <C>$. For in this case there are states $A_{ij}, B_{ik} \in X_i \in C$ for which A_{ij} $|_T$ A_{ik} does not hold; that is there exists a path in T with ends labeled from A_{ij} which intersects a path having ends labeled from A_{ik}. Let a,b denote the labeled ends of the first path, and c,d the labeled ends of the second. Then $T_{|\{a,b,c,d\}} \in \{ac|bd, ad|bc\}$. Let $\alpha = \alpha_{ij}$, $\alpha' = \alpha_{ik}$. Since $T \in <q(C)>$, we have $T_{|\{\alpha,\alpha',a,c\}} = \alpha a|\alpha' c$, $T_{|\{\alpha,\alpha',a,d\}} = \alpha a|\alpha' d$, $T_{|\{\alpha,\alpha',b,c\}} = \alpha b|\alpha' c$ and $T_{|\{\alpha,\alpha',b,d\}} = \alpha b|\alpha' d$. Now it can readily be checked by case examination that the set Q^* consisting of these last four resolved quartets, together with either of the above two candidates for $T_{|\{a,b,c,d\}}$, is incompatible, and hence not consistent with T. But Q^* consists of induced resolved quartets of T, (i.e. resolved quartet of type $T_{|S}$ for a quartet S) and so, by definition, is consistent with T, giving the required contradiction.

(2) Suppose $A,B \in X \in c(P)$. Then by the construction of $c(P)$, $A \cup B$ is the

label set of some tree t in P, and $A \mid_t B$. Thus for $T \in \langle P \rangle$, since $T_{|A \cup B} \to t$, we have $A \mid_T B$. This holds for all A,B,X so $T \in \langle c(P) \rangle$. Conversely, suppose $T \in \langle c(P) \rangle$. Select $t \in P$, let L' denote its label set, and let $t' = T_{|L'}$. For each split $X = \{A,B\}$ induced by t, we have $A \mid_{t'} B$. Now collapse each edge of t' for which the induced split of t' is not an induced split of t to obtain a phylogenetic tree having the same collection of induced splits as t. Since each phylogenetic tree is characterized by its induced splits (Bandelt and Dress 1986) it follows that $T_{|L'} = t' \to t$. This holds for all $t \in P$, giving $T \in \langle P \rangle$, as required.

(3) We first note that if $\langle P_1 \rangle = \langle P_2 \rangle$ and $\langle P'_1 \rangle = \langle P'_2 \rangle$ then $\langle P_1 \cup P'_1 \rangle = \langle P_2 \cup P'_2 \rangle$, so that it suffices to prove the result when P consists of a single tree t. We use induction on $|L|$. The result holds for $|L| = 4$; suppose it holds for all $|L| < k$, and that $|L| = k > 4$. Given $x \in L$, select a vertex of t which is adjacent to a set of leaves, $U = \{u_1, \ldots, u_k\}$, $k > 1$, with $x \notin U$. This is possible since any phylogenetic tree on 4 or more labels, and having an internal edge, has at least two vertices, each adjacent to at least two leaves (if t has no internal edge the result holds trivially). Let t' be the phylogenetic tree $t_{|L'}$, where $L' = L - \{u_2, \ldots, u_k\}$. By hypothesis there exists a set Q' of at most $\lambda = -1 + \sum_v (\partial(v) - 2)$ (with v ranging over the internal vertices of t') resolved quartets, each labeled with x, and such that $\langle Q' \rangle = \langle \{t'\} \rangle$. Deleting the vertex v' of t' adjacent to u_1 and its incident edges, disconnects t' into the isolated vertex u_1 and rooted subtrees $t_1, t_2, \ldots, t_s$, $s \geq 2$, as illustrated in Figure 2.

Supposing x is a leaf of t_1, select any leaf y of t_2 and let $Q = \{xy \mid u_1u_2, \ldots, xy \mid u_1u_k\} \cup Q'$. Then it is routine to check that $\langle Q \rangle = \langle \{t\} \rangle$; thus Q is the required set of $\lambda + k - 1 = \lambda + \partial(v') - 2$ resolved quartets, each containing x, for which $\langle Q \rangle = \langle \{t\} \rangle$, completing the induction step. •

Note that the transformation $C \to c(q(C))$ transforms a set of characters into a collection of characters, each having all its states of size 2, and for which $\langle C \rangle = \langle c(q(C)) \rangle$. Also, the transformation $P \to q(c(P))$ is less efficient than the construction described in Proposition 2 (3), since $|q(P)|$ is bounded linearly by $|L|$, while $|q(c(P))|$ is only bounded by $|L|^3$.

We now give two separate characterizations of compatibility for a set of characters. We must first define two graphs.

Definitions (3). Suppose a graph G is k-partite on vertex sets $V_1, \ldots, V_k$ (that is no two vertices in V_i are adjacent for $i = 1, \ldots, k$). We say G has a

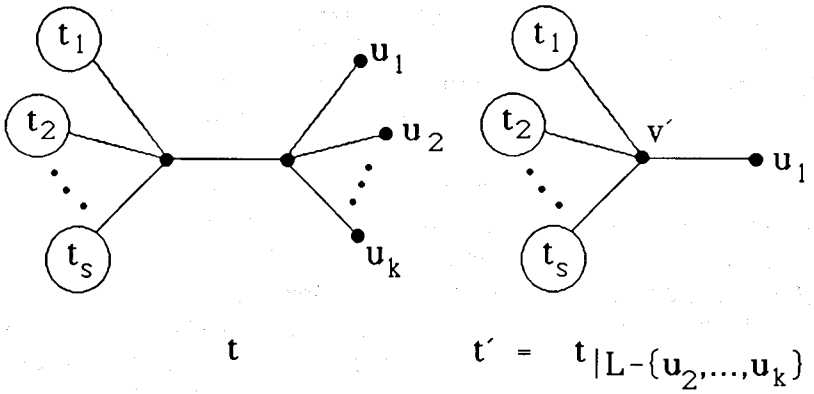

Figure 2. Pruning leaves for the proof of Proposition 2(3).

restricted chordal extension if there is a chordal graph which is k-partite on $V_1, \ldots, V_k$, and whose edges include those of G (a chordal graph has the property that every cycle of length four or more has a chord [an edge joining two vertices which are not adjacent in the cycle]). Given a collection υ of subsets of a set L, the *intersection graph* of υ is the graph with vertex set υ, and with two vertices A,B the ends of an edge precisely if $A \cap B \neq \emptyset$. For a collection of characters $C = \{X_1, \ldots, X_r\}$ the intersection graph of υ of the states of the characters is called the *character state intersection graph of* C, denoted $G(C)$. Since each character is a partition of a subset of L, $G(C)$ is r-partite on $X_1, \ldots, X_r$. Two characters X_1, X_2 are *weakly compatible* precisely if for all states $U,V \in X_1$, $U',V' \in X_2$ at least one of $U \cap U'$, $U \cap V'$, $V \cap U'$, $V \cap V'$ is empty (a pair of compatible characters are weakly compatible, but the converse does not hold, see Section 3 (1) below). Given C we define a second graph $\Omega(C)$ as follows. The vertices of $\Omega(C)$ consist of all nontrivial splits $\{A,B\}$ of L_C (nontrivial means $|A|, |B| > 1$) which are weakly compatible with every character of C. Two such splits are the ends of an edge of $\Omega(C)$ precisely if they are weakly compatible. The following results are part of the folklore; for completeness we give proofs here.

Proposition 3. *Let C be a set of partial characters. The following statements are equivalent:*

(1) *C is compatible.*
(2) *$G(C)$ has a restricted chordal extension.*
(3) *$\Omega(C)$ has a clique of size $|L_C| - 3$.*

Proof. The equivalence of (1) and (2) was essentially stated by Meacham (1983) and Buneman (1974) and follows from a characterization of chordal graphs independently due to Buneman (1974), Gavril (1974) and Walter (1972). This result states that a graph G is chordal precisely if there exists a tree T having all its vertices labeled, and a collection υ of the vertex sets of certain subtrees of T, for which G is the intersection graph of υ. Thus if C is compatible there exists a binary phylogenetic tree T, consistent with C. Assign new, distinct labels to all internal vertices of T. For each character state A occuring in C let T^A denote the vertex set of the minimum subtree of T (now fully labeled) connecting vertices labeled from A. The intersection graph of the T^A sets provides the required restricted chordal extension of $G(C)$. Conversely if $G(C)$ has a restricted chordal extension there exists a fully labeled tree T and a bijection H from the states of the characters to the vertex sets of certain subtrees of T such that

(i) if $A \cap B \neq \emptyset$ then $H(A) \cap H(B) \neq \emptyset$
(ii) if $A,B \in X \in C$ then $H(A) \cap H(B) = \emptyset$.

Define a labelling function f from L_C to the vertices of T as follows: For $y \in L_C$, let $C(y)$ denote the set of states (of the characters) containing y. For $A, A' \in C(y)$, (i) implies $H(A) \cap H(A) \neq \emptyset$. Now the vertex sets of a collection of subtrees of a tree have the Helly property (see Golumbic 1980) so T has a vertex v_y in $\cap_{A \in C(y)} H(A)$. Set $f(y) = v_y$. Then T is consistent with C under this labelling function, for if $a,a' \in A$, $b,b' \in B$, $A,B \in X \in C$, then since $f(a)$ and $f(a')$ lie in $H(A)$, and $f(b)$, $f(b')$ lie in $H(B)$, the path in T connecting a and a' is disjoint from the path connecting b and b' by (ii). Thus $A \mid_T B$ for all states $A,B \in X \in C$, as required.

(1) $\Rightarrow$ (3): If C is compatible, there exists a binary phylogenetic tree T on L_C consistent with C. Consider the set S of the $|L_C| - 3$ nontrivial splits induced by T upon deleting each internal edge of T. For $\{A,B\} \in S$ and $U,V \in X \in C$ suppose $A \cap U, A \cap V, B \cap U, B \cap V$ are all nonempty. Select one element from each set; say a, a', b, b', respectively. Then $T_{|\{a,a',b,b'\}} = aa' | bb'$, so that $U \mid_T V$ does not hold, contradicting the assumption that $T \in \langle C \rangle$. Thus S is a set of vertices of $\Omega(C)$, and S is a clique of $\Omega(C)$ by Proposition 1(b) of Bandelt and Dress (1986).

(3) $\Rightarrow$ (1): Suppose $\Omega(C)$ has a clique ω of size $|L_C| - 3$. Then by Lemma 6 of Buneman (1971), there is a binary phylogenetic tree T on L and a bijection h from the internal edges of T to ω defined by letting $h(e)$ be the split induced by T upon deleting edge e. Suppose $A,B \in X \in C$, and that $A \mid_T B$ does not hold. Then there exists $a,a' \in A$, $b,b' \in B$, $T_{|\{a,a',b,b'\}} = ab | a'b'$. Let e be an edge of T separating the path in T connecting a and b from the path connecting a' and b'. Then the states in $h(e)$ both have nonempty intersection with A and B, so $h(e) \notin \omega$, a contradiction.

Thus $A \mid_T B$, and so T is consistent with C. •

The previous proof extends to give a bijection between the cliques of $\Omega(C)$ of size $|L_C| - 3$ and $<C>$. However even if $\Omega(C)$ has a clique of size $|L_C| - 3$ a given vertex of $\Omega(C)$ need not lie in any clique of this size (i.e., if C is compatible, a split which is pairwise compatible with all characters in C may not be induced by any tree which is consistent with C). An example is provided by $C = \{\{\{1,2\},\{3,4\}\}, \{\{1,3\},\{4,5\}\}, \{\{1,4\},\{5,6\}\}\}$ and the split $\{\{2,6\},\{1,3,4,5\}\}$.

3. Special Cases

There are a number of special cases and variations on the original problem for which polynomial-time algorithms exist, which we now describe.

(1) If C has just two characters, then C is compatible if and only if the character state intersection graph $G(C)$ is acyclic. This result, due to Estabrook and McMorris (1977), follows from Proposition 3, since (i) an acyclic graph is chordal, and (ii) conversely, if $G(C)$ has a restricted chordal extension G' then G' is acyclic (otherwise G', being chordal, would have a 3-cycle, which is prohibited since G' is bipartite), and so $G(C)$ is acyclic. More generally if $|C|$ is bounded the compatibility of C can be decided in polynomial time (McMorris, Warnow and Wimer 1990).

(2) If B consists of a resolution of every quartet of L (or of every quartet containing a fixed label) then compatibility is easily decided (see for example Proposition 2 of Bandelt and Dress 1986, or Theorem 2 of Colonius and Schulze 1981). Furthermore there is at most one tree consistent with B and this can readily be constructed if it exists.

(3) More generally, given a set B of resolutions of quartets from L, B can be extended to a set B' by applying various dyadic, triadic and higher order rules, as described by Dekker (1986), which must always hold on a set of compatible quartets (see also Bandelt and Dress 1986; Bandelt, von Haeseler, Bolick and Schütte 1990). There are precisely two dyadic rules;

(i) transitivity: $ab \mid cd$ and $ab \mid de \Rightarrow ab \mid ce$,
(ii) substitution: $ab \mid cd$ and $ac \mid de \Rightarrow ab \mid ce, ab \mid de, bc \mid de$.

Then B' is constructed as the minimal set containing B which is closed under these rules. Thus if B' contains two contradictory resolutions of the same quartet, B is incompatible. The converse does not hold, at least for dyadic and triadic rules, (Dekker 1986); indeed, if $P \neq NP$ then incompatibility cannot be always detected using r-adic rules for bounded r, by Theorem 1,

below. However if B defines T this can sometimes be detected if B' has a resolution of every quartet from L, as described above in case (2).

For a set S of partial binary characters, three analogous rules for extending S have been proposed by Meacham (1983), who established their validity in case S is compatible. Meacham suggested that perhaps repeated application of these rules might suffice to identify when S is incompatible by always, in that case, generating a pair of incompatible partial binary characters (such a pair can readily be recognized, see case (1) above).

However we note here that these rules, as well as the types specifically considered by Dekker cannot achieve this. For consider a balanced incomplete block design consisting of 37 elements arranged into 111 blocks of size 4 (Hall 1967). There are 3^{111} ways to resolve all the quartets in this design, and some of these resolutions must be inconsistent, since $3^{111} > b(37)$ by Proposition 1(1), and since $<Q> \cap <Q'> = \emptyset$ if $Q \neq Q'$. Now, any two quartets in this design have at most one element in common, and so two partial binary characters $\{A,A'\}$ and $\{B,B'\}$ constructed from these quartets satisfy $|(A \cup A') \cap (B \cup B')| \leq 1$. But any application of the Dekker/Meacham rules requires a pair of partial binary characters for which $|(A \cup A') \cap (B \cup B')| \geq 2$. It follows that none of these rules can detect any of the inconsistent instances which exist in this example.

(4) If C is a set of binary qualitative characters then compatibility can be decided in polynomial time since C is compatible precisely if each pair in C is compatible, and two binary characters are compatible precisely if they are weakly compatible (see Definitions (3)). These results are essentially due to Buneman (1971) (see also McMorris 1977, and Bandelt and Dress 1986). In case C is compatible a consistent tree can be constructed in linear time by a method such as "TREE POPPING" (Meacham 1981), or by the method described by Gusfield (1991). This result does not generalize to nonbinary qualitative characters (Fitch 1975, McMorris 1975) or partial binary characters. For example, for $2 \leq i \leq n-1$ set $F_i = \{\{1,i\}, \{i+1, n+1\}\}$ and let $F_n = \{\{1,n\}, \{2, n+1\}\}$. Then $\{F_1, \ldots, F_n\}$ is an incompatible set of n partial binary characters, every subset of which is compatible.

(5) More generally if C is a set of qualitative characters, each having at most four states, then compatibility can be determined in polynomial time and a consistent tree constructed if one exists. A clever and intricate $O(|L|^2 \times |C|)$ algorithm is described by Kannan and Warnow (1990). In case each character has at most three states, a much simpler algorithm is possible (see also Dress and Steel 1991).

(6) As mentioned earlier, if each tree in some set S has all its vertices labeled, then the question of whether there is a parent tree for which each tree in S is a

subtree can be decided efficiently by determining whether the intersection graph of S is chordal. This can be determined in linear time (see, for example, Golumbic 1980); Gavril (1974) gives a polynomial-time algorithm which also constructs a tree realization of the intersection graph of S when one exists. The limitation of this useful result is that in applications to an existing set of objects derived from some evolutionary process the identity and position of particular ancestors has been lost. This is the problem of dealing with "missing objects" referred to by Buneman (1974), and is the motivation behind this present study.

(7) If additional tree-like relationships are imposed between the states in a qualitative character to give a "cladistic character," the corresponding compatibility problem can be decided in polynomial time. In this case the compatibility question reduces to pairwise compatibility, which can readily be checked (see Estabrook, Johnson and McMorris 1976; Estabrook and Meacham 1979). Recently, Gusfield (1991) has given an improved algorithm for this problem which is linear both in the number of cladistic characters and in $|L|$.

(8) For two phylogenetic trees t,t' having a label ("root") in common, an efficient way of determining the compatibility of $\{t,t'\}$ and constructing the "strict consensus tree" (defined by Sokal and Rohlf 1981) for the set $<\{t,t'\}>$ has been given by Gordon (1986). Using a different approach a general method for $k \geq 2$ rooted trees is described below (see also Kant-Antonescu and Sankoff 1991).

4. Complexity

We now show that deciding the compatibility of resolved quartets is NP-complete. By a *caterpillar* we mean a binary phylogenetic tree having at most two vertices which are each adjacent to two leaves, as in Figure 3. If α,β label two leaves which are maximally far apart in a caterpillar tree T it is convenient to call T an $\alpha\beta$-caterpiller. We write $\alpha x_1 | x_2 x_3, \ldots, x_{p-1} | x_p \beta$ to identify the caterpillar in Figure 3. Note that if T is an $\alpha\beta$-caterpillar, and $\alpha,\beta \in S$ then $T_{|S}$ is also an $\alpha\beta$-caterpillar.

Lemma *If a collection B of $\alpha\beta$-caterpillars is compatible, then B is consistent with an $\alpha\beta$-caterpillar.*

Proof. Define a relation $<$ on L as follows: $x < y$ precisely if for some $t \in B$, $t_{|\{\alpha,\beta,x,y\}} = \alpha x | y \beta$. Suppose B is consistent with T, $x < y$, and $y < z$. Then $T_{|\{\alpha,\beta,x,y\}} = \alpha x | y \beta$, and $T_{|\{\alpha,\beta,y,z\}} = \alpha y | z \beta$, so that $z \neq x$ and $T_{|\{\alpha,\beta,x,y,z\}} = \alpha x | y | z \beta$. In particular $x < z$.

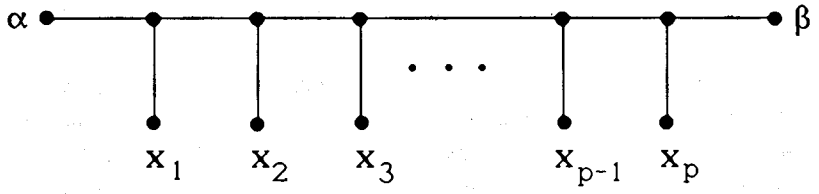

Figure 3. The αβ-caterpillar tree $\alpha x_1 | x_2, \ldots, x_{p-1} | x_p \beta$.

Thus if B is consistent, $<$ is transitive on L, and since $x < x$ cannot occur, it follows that L has no cycle $x = x_1 < \ldots < x_s = x$. Thus, by a well known result there is a linear ordering $<^*$ of L such that $x < y$ implies $x <^* y$. If $x_1 <^* x_2 <^* \cdots <^* x_{|L|}$ in this ordering let T^* denote the αβ-caterpillar tree $\alpha x_1 | x_2 \cdots | x_{|L|} \beta$. Now if $t \in B$ has label set $S \cup \{\alpha,\beta\}$ and if $t_{|\{\alpha,\beta,x,y\}} = \alpha x | y \beta$, we have $x < y$, and hence $x <^* y$ so that $T^*_{|\{\alpha,\beta,x,y\}} = \alpha x | y \beta$. Thus since $\{t_{|\{\alpha,\beta,x,y\}} : x,y \in S\}$ defines t it follows that $T^*_{|S \cup \{\alpha,\beta\}} = t$. This holds for all trees t in B, completing the proof. •

Consider then the following problem:

QUARTET COMPATIBILITY
INSTANCE: A set Q of resolved quartets on L.
QUESTION: Is Q compatible?

Theorem 1. *Quartet compatibility is NP-complete.*

Proof. Quartet compatibility is clearly in NP, for given a tree T consistent with Q this consistency can be verified by checking each resolved quartet in Q against T. We now describe a transformation from the following problem, "BETWEENNESS", which is NP-complete (Garey and Johnson 1979).

INSTANCE: Finite set A, collection C of ordered triples (a,b,c) of distinct elements from A (we may assume that each element in A occurs in at least one triple from C).
QUESTION: Is there a *betweenness ordering f of A for C*; that is a one-to-one function $f: A \to \{1,2,\ldots,|A|\}$ such that for each $(a,b,c) \in C$, either $f(a) < f(b) < f(c)$ or $f(c) < f(b) < f(a)$?

Suppose an instance of BETWEENNESS is given. We construct for each ordered triple $\pi = (a,b,c)$ in C the following set Q_π of 6 resolved quartets:

$$Q_\pi = \{pp' | ab, pa | bc, pb | cq, pc | qq', \alpha p | p'\beta, \alpha q | q'\beta\}$$

where $p = p_\pi, p' = p'_\pi, q = q_\pi, q' = q'_\pi$ are four new labels chosen for each π, while α,β are two new labels fixed for all elements of C. Thus $Q(C) = \cup_{\pi \in C} Q_\pi$ is a collection of $6|C|$ resolved quartets on $4|C| + |A| + 2$ labels.

We claim that $Q(C)$ is compatible if and only if A has a betweenness ordering for C.

First suppose $Q(C)$ is compatible. It can be checked that $<Q_\pi> = \{\alpha p | p'abcq | q'\beta, \alpha q | q'cbap | p'\beta\}$. Thus if T is consistent with $Q(C)$ let T_π denote the $\alpha\beta$-caterpillar in $<Q_\pi>$ which is $T_{|\{\alpha,\beta,p,p',q,q',a,b,c\}}$. Since T is consistent with $\{T_\pi: \pi \in C\}$, this set satisfies the hypotheses of the previous lemma so there exists an $\alpha\beta$-caterpillar tree T_1, also consistent with $\{T_\pi: \pi \in C\}$. Let T^* denote the $\alpha\beta$-caterpillar $T_{1|A}$, and define $f_{T^*} : A \to \{1,2,\ldots,|A|\}$ by letting $1 + f_{T^*}(a)$ be the number of edges on the path in T^* joining the vertices a and α. Since T is consistent with Q_π, and both trees in Q_π have b "between" a and c on the path connecting α and β, it follows that $f_{T^*}(b)$ lies between $f_{T^*}(a)$ and $f_{T^*}(c)$. This holds for all $\pi \in C$ so f_{T^*} is a betweenness ordering of A for C.

Conversely, suppose a betweenness ordering f exits. Let T denote the unique $\alpha\beta$-caterpillar tree with $f_T = f$, (where f_T is defined as above for f_{T^*}). We construct a series $T_0, \ldots, T_{|C|}$ of $\alpha\beta$-caterpillars recursively from $T_0 = T$ as follows: Index C and suppose T_{i-1}, $i > 0$, has been defined. Twice subdivide the edge of T_{i-1} incident with vertex α and make the two new vertices created by the subdivision adjacent to two new degree-one vertices v_α, v'_α by introducing two new edges. Perform a similar subdivision on the edge incident with vertex β to attach degree-one vertices v_β, v'_β. We may suppose $T_{i-1}(\{\alpha,\beta,v_\alpha,v'_\alpha,v_\beta,v'_\beta\}) = \alpha v_\alpha | v'_\alpha v_\beta | v'_\beta\beta$ For $\pi_i = (a,b,c)$ if $f(a) < f(b) < f(c)$ label $v_\alpha, v'_\alpha, v_\beta, v'_\beta$ by $p_\pi, p'_\pi, q_\pi, q'_\pi$, (respectively), while if $f(a) > f(b) > f(c)$, assign labels $q_\pi, q'_\pi, p_\pi, p'_\pi$, (respectively). In this way, $T_{|C|}$ is consistent with $Q(C)$, and in particular $Q(C)$ is compatible, which establishes the claim.

The proof of NP-completeness now follows, by the observation that the construction of $Q(C)$ from C can be carried out in polynomial time. •

One consequence of this theorem and Proposition 3 is that the problem of deciding whether a graph, which is k-partite on $V_1, \ldots, V_k$, has a restricted chordal extension is also NP-complete, even when $|V_i| = 2$ for all i. A similar question of determining whether a graph can be transformed into a chordal graph by adding $< k$ edges has been shown to be NP-complete by Yannakakis (1981).

The complexity of the compatibility question changes sharply if we consider sets of phylogenetic trees which have at least one label in common.

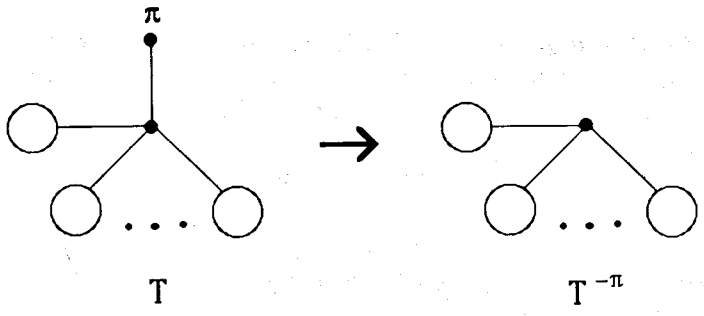

Figure 4. Pruning a leaf of a phylogenetic tree.

This is equivalent to considering sets of *rooted phylogenetic trees* (where the distinguished root vertex has degree at least two), for there is a simple bijection between these two classes defined as follows: Given a phylogenetic tree with a leaf labeled π delete this leaf and its incident edge to obtain a rooted phylogenetic tree $T^{-\pi}$ as illustrated in Figure 4. Inversely, given a rooted phylogenetic tree T on label set L, with $\pi \notin L$, we let T^{π} denote the unique phylogenetic tree on $L \cup \{\pi\}$ for which $(T^{\pi})^{-\pi} = T$.

Thus consider sets of rooted trees, which may have come directly from a hierarchical classification, or may have been derived by the above pruning process. The extension of earlier definitions is straightforward and if T is a rooted phylogenetic tree on label set $L \supseteq L'$ we let $T_{|L'}$ denote the induced rooted phylogenetic tree on L'. For completeness we first provide the analogue of Proposition 2(3), where *rooted triples* (written $a\,|\,bc$) replace resolved quartets, the proof following immediately from this earlier result (by chosing $x \notin L$, and for a set R of rooted triples applying Proposition 2(3) to $\{t^x : t \in R\}$).

Proposition 4. *Let R be a set of rooted phylogenetic trees. Then there exists a set $r(R)$ of rooted triples such that $<r(R)> = <R>$, with*

$$|r(R)| \leq \sum_{t \in R} \sum_{v:\partial(v)>2} (\partial(v) - 2)$$

We now describe, and give two simple application of, a polynomial-time algorithm for deciding the compatiblity of rooted trees, due to Aho et al. (1981).

Definitions (4). For a rooted phylogenetic tree t on L consider the partition of L obtained by deleting the root of t, and for $s \in L$ let $t(s)$ denote the set in this

partition containing s. Suppose R is a set of rooted phylogenetic trees on L. Define a graph on L by making $x,y \in L$ ends of an edge precisely if there is a tree $t \in R$ with $t(x) = t(y)$. Let $\varepsilon(R,L)$ denote the components of this graph, and let F^R denote the minimal set of subsets of L satisfying:

(1): $F^R \supseteq \varepsilon(R,L)$
(2): If $X \in F^R$ then $F^R \supseteq \varepsilon(R_{|X},X)$, where $R_{|X} = \{t_{|X}: t \in R\}$

Thus F^R is obtained by first constructing $\varepsilon(R,L)$, and for each set X in $\varepsilon(R,L)$ constructing the components of the graph (on vertex set X) induced by $R_{|X}$, and so on, until no further sets are created.

Clearly the sets in F^R are nested, in the sense that for $A,B \in F^R$ we have $A \cap B \in \{\emptyset,A,B\}$. Thus F^R represents a rooted phylogenetic tree by the well-known "n-tree" representation of Margush and McMorris (1981) as a collection of nested sets or *clusters*.

Theorem 2. *(Aho et al. 1981)*

(1) R is compatible if and only if $F^R \supseteq \{\{x\}: x \in L\}$,
(2) if $F^R \supseteq \{\{x\}: x \in L\}$ then F^R is consistent with R.
(3) if R consists of rooted triples, F^R can be constructed by an algorithm having $O(|R|^2)$ running time.

Note that if R contains trees on more than 3 labels, one can apply the construction used in the proof of Proposition 2(3) (in the context of Proposition 4) to efficiently construct a set R' of rooted triples with $|R'| = O(|L| \times |R|)$ which can then used in the above $O(|R'|^2)$ algorithm. Alternatively, if $|L| << |R|$ it may be more efficient to carry out the construction of F^R directly without such a reduction.

Corollary 1. *For each fixed k there is an algorithm of polynomial time in $|L|$ for deciding whether a sample of at most k phylogenetic trees on subsets of L is compatible.*

Proof. Select any label x in L (preferably one which occurs in a majority of trees). For each tree t in the sample P define a set $S(t)$ of rooted trees as follows: $S(t) = \{t^{-x}\}$ if x is a label of t, otherwise $S(t)$ is the set of rooted phylogenetic trees which can be obtained from t by subdividing an edge of t (and making the new vertex the root). Then P is compatible precisely if there is a compatible set R of rooted trees, one from each set $S(t)$. The number of choices for R is of order $|L|^k$ and the result now follows from the theorem, and the comment which follows it. •

The above theorem does not directly tell us whether a cluster in F^R occurs in every rooted phylogenetic tree consistent with R, or only in some. More generally given any collection of rooted phylogenetic trees it is useful to partition the clusters occuring in these trees into two types — *universal clusters* which occur in all trees in the collection, and the remainder.

Universal clusters are nested and so form a phylogenetic tree under inclusion (via the above "n-tree" representation) called the *strict consensus tree* of the collection (Sokal and Rohlf 1981).

Gordon (1986) asks whether there is an efficient method to construct this tree for the collection of all rooted phylogenetic trees (equivalently, all binary phylogenetic trees) which are consistent with R. Day (1985) has described an algorithm for constructing the strict consensus of an arbitrary set of phylogenetic trees, and this is polynomial in the number of trees, however it cannot be applied directly here as $|<R>|$ can grow exponentially with $|L|$. The following result has been independently derived by Kant-Antonescu and Sankoff (1991).

Corollary 2. *The strict consensus of all rooted phylogenetic trees consistent with R can be constructed in polynomial time.*

Proof. The following result is easily established: Suppose T is a phylogenetic tree consistent with R, S is a non-singleton cluster of T and $x \in S$. Then S is a cluster of every phylogenetic tree consistent with R if and only if, for each pair (a,b), $a \in S$, $a \neq x$, $b \notin S$, both $R(a,b) = R \cup \{ab|x\}$ and $R(b,a) = R \cup \{xb|a\}$ are incompatible. Thus to construct the strict consensus tree, first construct F^R to determine whether R is compatible. If so each cluster in F^R is tested for universality by constructing $F^{R(a,b)}$, $F^{R(b,a)}$ (for each (a,b)) to test the compatibility of $R(a,b)$, $R(b,a)$. Since F^R has at most $|L|-2$ non-singleton clusters, the strict consensus of $<R>$ can be constructed by an algorithm of running time $O(|L|^3\lambda)$ where λ is the complexity of calculating F^R. The result now follows from the comments following Theorem 2. •

Finally we mention a generalization of the compatibility question in the case of rooted trees; namely the problem of determining when a parent rooted binary tree T exists which disagrees with every tree in R (i.e., $T_{|S} \neq t$ for all $t \in R$ on label set S). This problem might arise, for example, in statistical approaches to taxonomy, in which trees which disagree with data "significantly" define a set of potential parent trees as those not containing any "unacceptable" subtrees.

However, the problem of deciding whether such a tree exists is NP-complete by a transformation from the NP-complete "Betweenness" problem

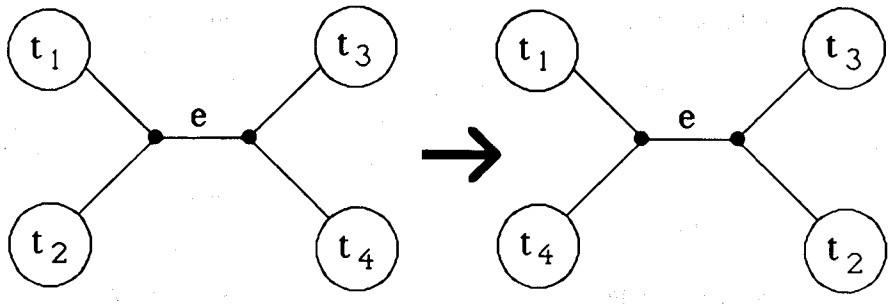

Figure 5. A nearest neighbor interchange.

described in the proof of Theorem 1. Specifically, given a set C of ordered triples from S, define two new labels, a,b, and a root leaf π and let

$$R = \cup_{(i,j,k)\in C} \{\pi j \mid ik,\ \pi i \mid k \mid jb,\ \pi k \mid i \mid jb\}$$
$$\cup \{\pi x \mid ab,\ \pi b \mid xa,\ \pi b \mid xy : x,y \in S\},$$
$$R' = \{t^{-\pi} : t \in R\}$$

Then it can be checked that there exists a rooted binary tree on S which induces none of the subtrees in R', precisely if S has a betweenness ordering for C.

5. Defining Trees by Subtrees

Finally we consider conditions for a sample of phylogenetic trees to define a parent tree. In view of Proposition 2 we can, without loss of generality, suppose that all the sample trees are resolved quartets. We first present a necessary condition. For an internal edge e of a binary tree T let $F(T,e)$ denote the four rooted subtrees of T whose roots are adjacent to e, as in Figure 5. If $\{a,b,c,d\}$ has one label in common with each tree in $F(T,e)$ we say the resolved quartet $ab \mid cd$ *distinguishes* e. This definition extends to rooted trees — if T is a rooted phylogenetic tree not labeled by π, the rooted triple $a \mid bc$ *distinguishes* an edge e of T if $a\pi \mid bc$ distinguishes e on T^{π}.

Proposition 6. *If* $\langle Q \rangle = \{T\}$ *then for each internal edge* e *of* T *there exists a resolved quartet in* Q *which distinguishes* e. *In particular the bounds given in Propositions 2 and 4 are realized.*

Proof. Suppose T has an edge e which is not distinguished by Q. Let T' be either of the two binary trees obtained from T by interchanging two of the

trees in $F(T,e)$ whose roots are separated by e. This operation, called a "nearest neighbour interchange" across e (Waterman and Smith 1978), is illustrated in Figure 5. Now for any resolved quartet $ab|cd$ in Q, considering the various ways these labels can be distributed among the trees in $F(T,e)$ so that at least one tree in $F(T,e)$ does not receive a label (required since by assumption no quartet in Q distinguishes e) it is clear that if $T_{|\{a,b,c,d\}} = ab|cd$, then $T'_{|\{a,b,c,d\}} = ab|cd$. Thus T' is also consistent with Q, and so $<Q> \neq \{T\}$, as required. •

The converse of Proposition 6 (assuming consistency of Q) does not hold; consider for example the two caterpillar trees 12|34|56, 14|52|36 which are both consistent with the set {12|36, 23|45, 14|56}. But this set has for each edge of either tree a quartet distinguishing that edge.

This raises the question of recognizing precisely when a sample of rooted or unrooted trees uniquely defines a tree. This question was posed in concluding remarks by Colonius and Schulze (1981), who conjectured that a simple characterization of such samples was unattainable. While this may be the case for unrooted trees, we give a simple characterization for rooted trees. In view of Proposition 4 we restrict attention to rooted triples.

Theorem 3. *For a set of rooted triples R, $<R> = \{T\}$ if and only if R is consistent with T, and for each internal edge e of T there is a rooted triple in R which distinguishes e.*

Proof. The "only if" part follows from Proposition 6. Conversely, we apply induction on the *height* $h(T)$ of T (the maximal number of vertices on any path from the root of T to a leaf). For $h(T) = 3$ the result holds so suppose $h(T) = k > 3$. Let T_1,T_2 be the two rooted subtrees of T, on label set L_1,L_2, obtained by deleting the root of T. For $i = 1,2$, let $R_i = \{ab|c \in R : a,b,c \in L_i\}$. Now R is consistent with T, and distinguishes every internal edge of T, *a fortiori* R_i ($i = 1,2$) is consistent with T_i, and distinguishes T_i on every internal edge of T_i. Thus since $h(T_i) < h(T)$ we may apply the inductive hypothesis and deduce that R_i defines T_i. There are now two cases to consider:

Case 1. $|L(T_i)| > 1$ for $i = 1,2$.
Case 2. $|L(T_i)| = 1$ for $i = 1$ or 2.

In case 1, represent T as in Figure 6, where $\{A_1,A_2\}$, $\{B_1,B_2\}$ denote the pairs of label sets for the two maximal subtrees of T_1 and T_2, respectively. Suppose T' is consistent with R, and let C_1,C_2 denote the labels on the two rooted subtrees of T' obtained by deleting the root of T' as in Figure 6. We show that $\{C_1,C_2\} = \{A_1 \cup A_2, B_1 \cup B_2\}$, then since R_i defines T_i, it

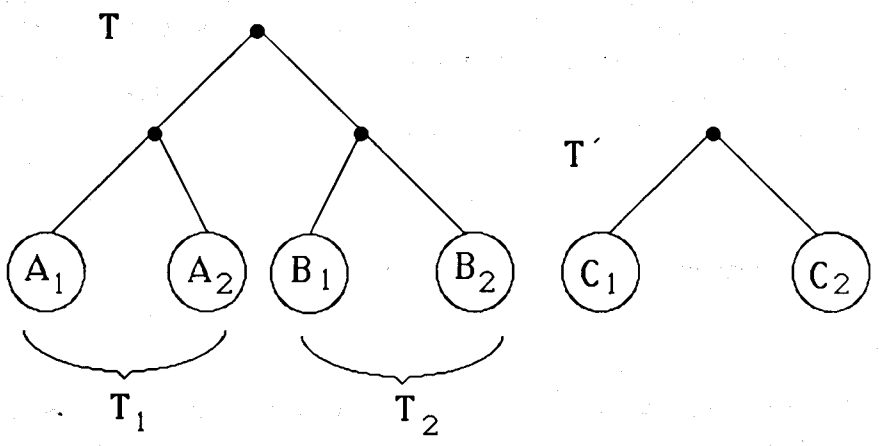

Figure 6. Tree decompositions for the proof of Theorem 3.

follows that $T' = T$. Now, since R_i defines T_i for $i = 1,2$, any set $X \in \{A_1,A_2,B_1,B_2\}$ which has nonempty intersection with C_k is contained in C_k. Now suppose A_i, B_j are both contained in C_k; without loss of generality we assume $i = j = k = 1$. Since R resolves T on the two edges incident with the root of T, $a_1a_2|b$ and $b_1b_2|a$ are both in R, where $a_i \in A_i$, $b_i \in B_i$, $a \in A_1 \cup A_2$, $b \in B_1 \cup B_2$. The conditions $a_1a_2|b \in R$ and $T' \in \langle R \rangle$ implies $T'_{|\{a_1a_2,b\}} = a_1a_2|b$, and regardless of whether $b \in B_1$ or B_2, the condition that $A_1 \cap C_1 \neq \emptyset$ then implies $A_2 \cap C_1 \neq \emptyset$. Thus A_2, and hence $A_1 \cup A_2$, is a subset of C_1. Then, by a similar argument, $b_1b_2|a \in R$ and $T' \in \langle R \rangle$ implies that $B_2 \cap C_1 \neq \emptyset$ hence B_2 is a subset of C_1, which together with the assumption that $C_1 \supseteq B_1$ implies $C_2 = \emptyset$, a contradiction. Thus, $\{C_1,C_2\} = \{A_1 \cup A_2, B_1 \cup B_2\}$, and so $T' = T$. A similar, though simpler argument in case 2 also gives $T' = T$, completing the inductive step, and so the proof. •

Since a rooted triple can resolve a rooted binary tree on at most one internal edge, this theorem gives the following result.

Corollary. *If R is a set of rooted triples which defines a rooted binary phylogenetic tree T, having n leaves, then R has a subset of cardinality $n-2$ which also defines T. Thus all minimal tree-defining sets of rooted triples have the same cardinality.*

The analogous result for unrooted trees does not apply. For although each binary phylogenetic tree having n leaves is defined by a set of $n-3$ resolved quartets (by Proposition 2(3)) there exist minimal defining sets of larger

cardinality. For example the set Q = {57|24, 15|67, 12|35, 47|13, 34|56} defines the caterpillar tree 12|345|67, but deleting the i-th element in Q as they are listed gives sets Q_i which do not define this tree. Specifically, $12|563|47 \in \langle Q_1 \rangle$, $12|346|57 \in \langle Q_2 \rangle$, $23|145|67 \in \langle Q_3 \rangle$, $67|531|24 \in \langle Q_4 \rangle$, $35|167|24 \in \langle Q_5 \rangle$.

6. Discussion

For a number of reasons, which we now list, it is often desirable to construct phylogenies from small sub-phylogenies on overlapping subsets of taxa.

(1) Recently there has been a desire to bring some degree of statistical respectability to taxonomy (Felsenstein 1988) by basing tree reconstruction methods on models of nucleotide mutation. This has led to methods based on maximum likelihood, and more recently, *invariants* (see for example, Sankoff 1990; Cavender and Felsenstein 1987; Hendy 1989). One problem in applying these methods is that for more than 4 taxa the calculations can become difficult, and in some cases the invariants are defined only on sets of 4 taxa. Thus one approach is to build up a tree from sets of resolved quartets which are, in some sense, "statistically significant."

(2) Traditional tree reconstruction methods such as parsimony and "compatibility" rely on chosing an optimal tree, and such procedures are generally NP-complete (see for example Foulds and Graham 1982; Day and Sankoff 1986). Furthermore there is no guarantee that an optimal tree is unique. However with a set of rooted triples one can efficiently decide whether they define a parent tree, and if so construct it.

(3) Subtree methods also provide an internal check on the consistency of the data and the criterion used to build the tree. The requirement that a set of resolved quartets be consistent places very stringent constraints on the ways the quartets can be resolved. In practice, any tree-building method based on resolved quartets will lead to inconsistency, however the degree of inconsistency can be measured in a variety of ways. For example, in one study Bandelt, von Haeseler, Bolick and Schütte (1990) used the number of violations to the transitivity rule (described above in Section 3(3)) as a test in this direction. Alternatively one might consider the minimal number of quartets which must be deleted to achieve consistency as a relevant, though not easily computable, measure.

(4) Quartet methods may also be useful for uncovering previously unresolved relationships between quartets. For example, applying the transitivity and

substitution rules described Section 3(3) can resolve previously unresolved quartets.

(5) In applications, quartet-based methods for tree reconstruction have been found to perform well (Dress, von Haeseler and Krueger, 1986). Also, for data in which there is a large degree of consistency amongst the quartets, such methods should be robust, in the sense that minor perturbations in the data should produce "similar" trees. This is because two trees which have a large proportion of equivalently resolved quartets in common appear "similar"; indeed it has been suggested that the appropriate metric for comparing phylogenetic trees is one based on the proportion of quartets which are equivalently resolved by the two trees (Estabrook, McMorris and Meacham 1985; Bandelt and Dress 1986; Day 1986).

These considerations motivate the question of determining when a set of trees (or characters) are compatible. In general, Theorem 1 shows that this question belongs to the class of NP-complete problems; as a consequence it is unlikely that there is a simple combinatorial characterization of the notion of compatibility. However a proof of NP-completeness is often regarded as the starting point for the development of heuristic, and branch and bound algorithms, and methods specific to the type of data given. For example, given a large number of trees or characters, the methods described in section 3(3), or based on the equivalence of (1) and (3) in Proposition 3 may work well. Conversely, if the data consists of just a few characters, a search for a restricted chordal extension of the graph $G(C)$ may be feasible (using the equivalence of (1) and (2) in Proposition 3).

Some important open problems remain; for example, determining the complexity of the compatibility question for qualitative characters when the number of states in each character is bounded (recall that qualitative characters partition the entire label set). As described earlier, an efficient algorithm has recently been obtained for characters having at most 4 states, a result which is directly relevant to taxonomy based on nucleotide sequences. Kannan and Warnow (1990) conjecture that an extension of their approach will yield an $O(r^{r-2} \times |L|^2 \times |C|)$ algorithm for the case where each character has at most r states. If so it would also be useful to know if this order of complexity, with respect to r, is best possible, since any algorithm with this complexity will be inadequate for dealing with protein sequences, where $r = 20$.

An analogous question applies to the complexity of the compatibility question for sets of trees when a bound is placed on size of the smallest subset of labels L' so that each tree has at least one label in L'. In case this bound is 1, the trees can be regarded as rooted on that label and an efficient method has been described for recognizing compatible sets. A natural

question is whether an efficient solution exists for trees in which the bound is 2. One might consider, for example sets of resolved quartets of the type $\alpha x \mid x'\beta$, $\alpha y \mid y'\gamma$, $\beta z \mid z'\gamma$, where $x,x',y,y',z,z' \in L$.

Finally, related to Buneman's question raised in the introduction is a possibly more tractable problem: Is there an efficient way of deciding, given a binary tree T and a set C of consistent characters (or resolved quartets) whether T is the only tree consistent with C?

References

AHO, A. V., SAVIG, Y., SZYMANSKI, T. G., AND ULLMAN, J.D. (1981), "Inferring a Tree from the Lowest Common Ancestors with an Application to the Optimization of Relational Expressions," *SIAM Journal on Computing, 10(3)*, 405-421.

BANDELT, H-J., and DRESS, A. (1986), "Reconstructing the Shape of a Tree from Observed Dissimilarity Data," *Advances in Applied Mathematics, 7*, 309-343.

BANDELT, H-J., VON HAESELER, A., BOLICK, J., and SCHÜTTE, H. (1990), "A Comparative Study of Sequence Dissimilarities and Evolutionary Distances Derived from Sets of Aligned RNA Sequences," preprint.

BONDY, J. A., and MURTY, U. S. R. (1976), *Graph Theory with Applications*, London: Macmillan.

BROSSIER, G. (1990), "Piecewise Heirarchical Clustering," *Journal of Classification, 7*, 197-216.

BUNEMAN, P. (1971), "The Recovery of Trees from Measures of Dissimilarity," in *Mathematics in the Archaeological and Historical Sciences*, Eds., F. R. Hodson, D. G. Kendall, and P. Tautu, Edinburgh: Edinburgh University Press, 387-395.

BUNEMAN, P. (1974), "A Characterization of Rigid Circuit Graphs," *Discrete Mathematics, 9*, 205-212.

CARTER, M., HENDY, M. D., PENNY, D., SZÉKELY, L. A., and WORMALD, N. C. (1990), "On the Distribution of Lengths of Evolutionary Trees," *SIAM Journal on Discrete Mathematics, 3*, 38-47.

CAVALLI-SFORZA, L. L., and EDWARDS, A. W. F. (1967), "Phylogenetic Analysis: Models and Estimation Procedures," *Evolution, 21*, 550-570.

CAVENDER, J. A. and FELSENSTEIN, J. (1987), "Invariants of Phylogenies: Simple Cases with Discrete States," *Journal of Classification, 4*, 57-71.

COLONIUS, H., and SCHULZE, H. H. (1981), "Tree Structures for Proximity Data," *British Journal of Mathematical and Statistical Psychology, 34*, 167-180.

CONSTANTINESCU, M., and SANKOFF, D. (1986), "Tree Enumeration Modulo a Consensus," *Journal of Classification, 3*, 349-356.

DAY, W. H. E. (1985), "Optimal Algorithms for Comparing Trees with Labeled Leaves," *Journal of Classification, 2*, 7-28.

DAY, W. H. E. (1985), "Analysis of Quartet Dissimilarity Measures between Undirected Phylogenetic Trees," *Systematic Zoology, 35(3)*, 325-333.

DAY, W. H. E., and SANKOFF, D. (1986), "Computational Complexity of Inferring Phylogenies by Compatibility," *Systematic Zoology, 35(2)*, 224-229.

DEKKER, M. C. H. (1986), *Reconstruction Methods for Derivation Trees*, Masters thesis, Vrije Universiteit, Amsterdam.

DRESS, A., VON HAESELER, A., and KRUEGER, M. (1986), "Reconstructing Phylogenetic Trees Using Variants of the 'Four-Point-Condition'," *Studien zur Klassifikation, 17*, 299-305.

DRESS, A., and STEEL, M. A. (1991) "Convex Tree Realizations of Partitions," *Applied Mathematics Letters* (in press).

DROLET, S., and SANKOFF, D. (1990), "Quadratic Tree Invariants for Multivalued Characters," *Journal of Theoretical Biology, 144*, 117-129.

ERDÖS, P., and SZÉKELY, L. A. (1989), "Applications of Antilexicographic Order 1. An Enumerative Theory of Trees," *Advances in Applied Mathematics, 10*, 488-496.

ESTABROOK, G. F., JOHNSON, C. S. Jr., and MCMORRIS, F. R. (1976), "An Algebraic Analysis of Cladistic Characters," *Discrete Mathematics, 16*, 141-147.

ESTABROOK, G. F., and MCMORRIS, F. R. (1977), "When are Two Taxonomic Characters Compatible?" *Journal of Mathematical Biology, 4*, 195-299.

ESTABROOK, G. F., and MEACHAM, C. A. (1979), "How to Determine the Compatibility of Undirected Character State Trees," *Mathematical Biosciences, 46*, 251-256.

ESTABROOK, G. F., McMORRIS, F. R., and MEACHAM, C. A. (1985), "Comparison of Undirected Phylogenetic Trees based on Subtrees of Four Evolutionary Units," *Systematic Zoology, 34(2)*, 193-200.

FELSENSTEIN, J. (1988), "Phylogenies from Molecular Sequences: Inference and Reliability," *Annual Review of Genetics, 22*, 521-565.

FITCH, W. M. (1975), "Towards Finding the Tree of Maximum Parsimony," in *The Eighth International Conference on Numerical Taxonomy*, Ed., G. F. Estabrook, San Fransisco: W.H. Freeman, 189-230

FOULDS, L. R., and GRAHAM, R. L. (1982), "The Steiner Problem in Phylogeny is NP-complete," *Advances in Applied Mathematics, 3*, 43-49.

GAREY, M. R., and JOHNSON, D.S. (1979), *Computers and Intractibility*, New Jersey: Bell Telephone Laboratories Ltd.

GAVRIL, F. (1974), "The Intersection Graphs of Subtrees in Trees are Exactly the Chordal Graphs," *Journal of Combinatorial Theory (B), 16*, 47-56.

GOLUMBIC, M. C. (1980), *Algorithmic Graph Theory and Perfect Graphs*, New York: Academic Press, 92.

GORDON, A. D. (1986), "Consensus Supertrees: The Synthesis of Rooted Trees Containing Overlapping Sets of Labeled Leaves," *Journal of Classification, 3*, 335-348.

GUSFIELD, D. (1991), "Efficient Algorithms for Inferring Evolutionary Trees," *Networks, 21*, 19-28.

KANNAN, S., and WARNOW, T. (1990), "Inferring Evolutionary Trees from DNA Sequences," in *31st Annual Symposium on Foundations of Computer Science (Proceedings)*, Los Alamitos, California: IEEE Computer Society Press, 362-371.

HALL, M. H. Jr. (1967), *Combinatorial Theory*, Waltham: Blaisdell, 175.

HENDY, M. D. (1989), "The Relationship Between Simple Evolutionary Tree Models and Observable Sequence Data," *Systematic Zoology, 38*, 310-321.

KANT-ANTONESCU, M., and SANKOFF, D. (1991), "Efficient Construction of Supertrees," manuscript.

MCMORRIS, F. R. (1975), "Compatibility Criteria for Cladistic and Qualitative Taxonomic Characters," in *The Eighth International Conference on Numerical Taxonomy*, Ed., E. A. Estabrook, San Fransisco: W. H. Freeman, 399-415.

MCMORRIS, F. R. (1977), "On the Compatibility of Binary Qualitative Taxonomic Characters," *Bulletin of Mathematical Biology, 39*, 133-138.

MCMORRIS, F. R., WARNOW, T., and WIMER, T. (1991), "Chordal Completion of Coloured Graphs for a Fixed Number of Colours", manuscript.

MARGUSH, T., and MCMORRIS, F. R. (1981), "Consensus n-trees," *Bulletin of Mathematical Biology, 43*, 239-244.

MEACHAM, C. A. (1981), "A Manual Method for Constructing Trees and Hierarchical Classifications," *Journal of Molecular Evolution, 18*, 30-37.

MEACHAM, C. A. (1983), "Theoretical and Computational Considerations of the Compatibility of Qualitative Taxonomic Characters," in *Numerical Taxonomy*, Ed., J. Felsenstein, NATO ASI Series Vol. G1, Berlin Heidelberg: Springer-Verlag, 304-314.

MEACHAM, C. A., and DUNCAN, T. (1987), "The Necessity of Convex Groups in Biological Classification," *Systematic Botany, 12*, 78-90.

SOKAL, R. R., and ROHLF, F. J. (1981), "Taxonomic Congruence in the Leptopodomorpha Re-examined," *Systematic Zoology, 30*, 309-325.

WALTER, J. R. (1972), *Representations of Rigid Cycle Graphs*, Ph.D thesis, Wayne State University, Detroit, Michigan.

WARNOW, T. (1991), *Combinatorial Algorithms for Constructing Phylogenetic Trees*, PhD thesis, University of California-Berkeley.

WATERMAN, M. S., and SMITH, T. S. (1978), "On the Similarity of Dendograms," *Journal of Theoretical Biology, 73*, 789-800.

YANNAKAKIS, M. (1981), "Computing the Minimum Fill-in is NP-complete," *SIAM Journal on Algebraic and Discrete Methods, 2*, 77-79.