AMB ALGORITHMS FOR MOLECULAR BIOLOGY

RESEARCH Open Access

Incompatible quartets, triplets, and characters

Brad Shutters*, Sudheer Vakati and David Fernández-Baca

Abstract

We study a long standing conjecture on the necessary and sufficient conditions for the compatibility of multi-state characters: There exists a function $f(r)$ such that, for any set C of r-state characters, C is compatible if and only if every subset of $f(r)$ characters of C is compatible. We show that for every $r \geq 2$, there exists an incompatible set C of $\Omega(r^2)$ r-state characters such that every proper subset of C is compatible. This improves the previous lower bound of $f(r) \geq r$ given by Meacham (1983), and $f(4) \geq 5$ given by Habib and To (2011). For the case when $r = 3$, Lam, Gusfield and Sridhar (2011) recently showed that $f(3) = 3$. We give an independent proof of this result and completely characterize the sets of pairwise compatible 3-state characters by a single forbidden intersection pattern.

Our lower bound on $f(r)$ is proven via a result on quartet compatibility that may be of independent interest: For every $n \geq 4$, there exists an incompatible set Q of $\Omega(n^2)$ quartets over n labels such that every proper subset of Q is compatible. We show that such a set of quartets can have size at most 3 when $n = 5$, and at most $O(n^3)$ for arbitrary n. We contrast our results on quartets with the case of rooted triplets: For every $n \geq 3$, if R is an incompatible set of more than $n - 1$ triplets over n labels, then some proper subset of R is incompatible. We show this bound is tight by exhibiting, for every $n \geq 3$, a set of $n - 1$ triplets over n taxa such that R is incompatible, but every proper subset of R is compatible.

Keywords: Phylogenetics, Quartet compatibility, Triplet compatibility, Character compatibility, Perfect phylogeny

Background

The multi-state character compatibility (or perfect phylogeny) problem is a basic question in computational phylogenetics [1]. Given a set C of characters, we are asked whether there exists a phylogenetic tree that displays every character in C; if so, C is said to be compatible, and incompatible otherwise. The problem is known to be NP-complete [2,3], but certain special cases are known to be polynomially-solvable [4-10]. See [11] for more on the perfect phylogeny problem.

In this paper we study a long standing conjecture on the necessary and sufficient conditions for the compatibility of multi-state characters.

Conjecture 1. *There exists a function $f(r)$ such that, for any set C of r-state characters, C is compatible if and only if every subset of $f(r)$ characters of C is compatible.*

If Conjecture 1 is true, it would follow that we can determine if any set C of r-state characters is compatible by testing the compatibility of each subset of $f(r)$ characters of C, and, in case of incompatibility, output a subset of at most $f(r)$ characters of C that is incompatible. This would allow us to reduce the character removal problem (i.e., finding a subset of characters to remove from C so that the remaining characters are compatible) to $f(r)$-hitting set which is fixed-parameter tractable [12].

A classic result on binary character compatibility shows that $f(2) = 2$; see [1,6,13-15]. In 1975, Fitch [16,17] gave an example of a set C of three 3-state characters such that C is incompatible, but every pair of characters in C is compatible; showing that $f(3) \geq 3$. In 1983, Meacham [15] generalized this example to r-state characters for every $r \geq 3$ demonstrating a lower bound of $f(r) \geq r$ for all r; see also [9]. For the case of $r = 3$, Lam, Gusfield, and Sridhar [9] recently established that $f(3) = 3$.

While the previous results could lead one to conjecture that $f(r) = r$ for all r, Habib and To [18] recently disproved this possibility by exhibiting a set C of five 4-state characters such that C is incompatible, but every proper subset of the characters in C are compatible, showing that $f(4) \geq 5$. They conjectured that $f(r) \geq r + 1$ for every $r \geq 4$.

*Correspondence: shutters@iastate.edu
Department of Computer Science, Iowa State University, Ames, IA 50011, USA

The main result of this paper is to prove the conjecture stated in [18] by giving a quadratic lower bound on $f(r)$. Formally, we show that for every $r \geq 2$, there exists a set C of r-state characters such that all of the following conditions hold.

1. C is incompatible.
2. Every proper subset of C is compatible.
3. $|C| = \lfloor \frac{r}{2} \rfloor \cdot \lceil \frac{r}{2} \rceil + 1$.

Therefore, $f(r) \geq \lfloor \frac{r}{2} \rfloor \cdot \lceil \frac{r}{2} \rceil + 1$ for every $r \geq 2$.

Our proof relies on a new result on quartet compatibility we believe is of independent interest. We show that for every $n \geq 4$, there exists a set Q of quartets over a set of n labels such that all of the following conditions hold.

1. Q is incompatible.
2. Every proper subset of Q is compatible.
3. $|Q| = \lfloor \frac{n-2}{2} \rfloor \cdot \lceil \frac{n-2}{2} \rceil + 1$.

This is an improvement over the previous lower bound on the maximum cardinality of such an incompatible set of quartets of $n-2$ given in [3]. We show that such a set of quartets can have size at most 3 when $n = 5$, and at most $O(n^3)$ for arbitrary n. We note here that the construction given in [18] showing that $f(4) \geq 5$ can be viewed as a special case of the construction given here when $n = 6$.

We study the compatibility of three-state characters further. The work of [9] completely characterized the sets of pairwise compatible 3-state characters by the existence of one of four forbidden intersection patterns. An alternative characterization of this result was given in [10] and was partially derived using the results of [9]. In this paper, we give a proof that $f(3) = 3$ that is independent of the results in [9], and we completely characterize the sets of pairwise compatible 3-state characters by a single forbidden intersection pattern.

We contrast our result on quartet compatibility with a result on the compatibility of rooted triplets: For every $n \geq 3$, if R is an incompatible set of triplets over n labels, and $|R| > n-1$, then some proper subset of R is incompatible. We show this bound is tight by exhibiting, for every $n \geq 3$, a set of $n-1$ triplets over n labels such that R is incompatible, but every proper subset of R is compatible.

Preliminaries

Given a graph G, we represent the vertices and edges of G by $V(G)$ and $E(G)$ respectively. We use the abbreviated notation uv for an edge $\{u, v\} \in E(G)$. For any $e \in E(G)$, $G - e$ represents the graph obtained from G by deleting edge e. For an integer i, we use $[i]$ to represent the set $\{1, 2, \cdots, i\}$.

Unrooted phylogenetic trees

An *unrooted phylogenetic tree* (or just *tree*) is a tree T whose leaves are in one to one correspondence with a label set $L(T)$, and has no vertex of degree two. See Figure 1(a) for an example. For a collection $\mathcal{T}$ of trees, the *label set* of $\mathcal{T}$, denoted $L(\mathcal{T})$, is the union of the label sets of the trees in $\mathcal{T}$. A tree is *binary* if every internal (non-leaf) vertex has degree three. A *quartet* is a binary tree with exactly four leaves. A quartet with label set $\{a, b, c, d\}$ is denoted $ab|cd$ if the path between the leaves labeled a and b does not intersect with the path between the leaves labeled c and d.

For a tree T, and a label set $L \subseteq L(T)$, the *restriction* of T to L, denoted by $T|L$, is the tree obtained from the minimal subtree of T connecting all the leaves with labels in L by suppressing vertices of degree two. See Figure 1(b) for an example. A tree T *displays* another tree T', if T' can be obtained from $T|L(T')$ by contracting edges. A tree T displays a collection of trees $\mathcal{T}$ if T displays every tree in $\mathcal{T}$. If such a tree T exists, then we say that $\mathcal{T}$ is *compatible*; otherwise, we say that $\mathcal{T}$ is *incompatible*. See Figure 1(a) for an example. Determining if a collection of unrooted trees is compatible is NP-complete [3].

Multi-state characters

There is also a notion of compatibility for sets of partitions of a label set L. A *character* χ on L is a partition of L; the parts of χ are called *states*. If χ has at most r parts, then χ is an r-state character. Given a tree T with $L = L(T)$ and a state s of χ, we denote by $T_s(\chi)$ the minimal subtree of T connecting all leaves with labels having state s for χ. We say that χ is *convex* on T, or equivalently T *displays* χ, if the subtrees $T_i(\chi)$ and $T_j(\chi)$ are vertex disjoint for all states i and j of χ where $i \neq j$. A collection C of characters is *compatible* if there exists a tree T on which every character in C is convex. If no such tree exists, then we say that C is *incompatible*. See Figure 1(a) for an example.

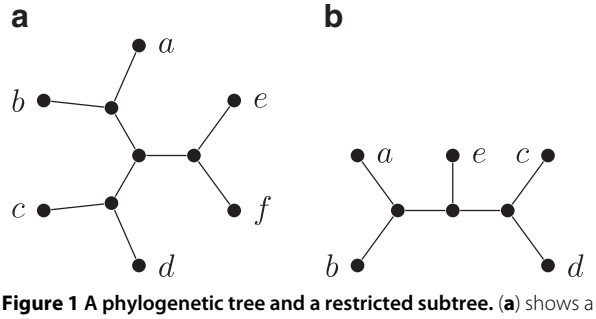

Figure 1 A phylogenetic tree and a restricted subtree. (**a**) shows a tree T witnessing that the quartets $\boldsymbol{q_1 = ab|ce}$, $\boldsymbol{q_2 = cd|bf}$, and $\boldsymbol{q_3 = ad|ef}$ are compatible; $\boldsymbol{T}$ is also a witness that the characters $\boldsymbol{\chi_{q_1} = ab|ce|d|f}$, $\boldsymbol{\chi_{q_2} = cd|bf|a|e}$, and $\boldsymbol{\chi_{q_3} = ad|ef|b|c}$ are compatible; (**b**) shows $\boldsymbol{T|\{a, b, c, d, e\}}$.

The *perfect phylogeny problem* (or *character compatibility problem*) is to determine whether a given set of characters is compatible.

For a collection C of characters, the *intersection graph* of C which we will denote by $G(C)$, is the undirected graph $G = (V, E)$ which has a vertex c_i for each character $c \in C$ and each state i of c, and an edge $c_i d_j$ precisely when there is a taxon having state i for character c and state j for character d. Note that $G(C)$ cannot have an edge between vertices associated with different states of the same character.

A graph G is *chordal* if there are no induced chordless cycles of length four or greater in H. In [19], Buneman established a fundamental connection between the perfect phylogeny problem and chordal graphs which we now describe. For a given set C of characters, suppose we color each of the vertices of $G(C)$ by assigning a unique color to each character $c \in C$, and giving each vertex of $G(C)$ corresponding to a state of c with the color assigned to the character c. A *proper triangulation* of $G(C)$ is a chordal supergraph of $G(C)$ such that every edge has endpoints with different colors.

Theorem 1. *A set C of characters is compatible if and only if $G(C)$ has a proper triangulation.*

Since there is no proper triangulation for a cycle in $G(C)$ involving only vertices from two characters, we have the following corollary.

Corollary 1. *Let C be a collection of two characters. Then C is compatible if and only if $G(C)$ is acyclic.*

Quartet rules

We now introduce *quartet (closure) rules* which were originally used in the contexts of psychology [20] and linguistics [21]. The idea is that for a collection Q of quartets, any tree that displays Q may also necessarily display another quartet $q \notin Q$, and if so we write $Q \vdash q$.

Example 1. *Let $Q = \{ab|ce, ae|cd\}$. Then the tree of Figure 1(b) displays Q, and furthermore, it is easy to see that it is the only tree that displays Q. Hence, $Q \vdash ab|de$, $Q \vdash ab|cd$, and $Q \vdash be|cd$.*

We use the following quartet rules in this paper:

$$\{ab|cd, ab|ce\} \vdash ab|de \quad \text{(R1)}$$

$$\{ab|cd, ac|de\} \vdash ab|ce \quad \text{(R2)}$$

For the purposes of this paper, we define the *closure* of an arbitrary collection Q of quartets, denoted Q^*, as the minimal set of quartets that contains Q, and has the property that if for some $q_1, q_2 \in Q^*$, $\{q_1, q_2\} \vdash q_3$ using either (R1) or (R2), then $q_3 \in Q^*$. Clearly, any tree that displays Q must also display Q^*. We will use the following lemma which follows by repeated application of (R1) and is formally proven in [22].

Lemma 1. *Let Q be an arbitrary set of quartets with $\{x, y, z_1, \ldots, z_k\} \subseteq L(Q)$. If*

$$\bigcup_{i=1}^{k-1} \{xy|z_i z_{i+1}\} \subseteq Q^* \quad ,$$

then $xy|z_1 z_k \in Q^*$.

We refer the reader to [1,23] for more on quartet rules.

Incompatible quartets

For every $s, t \geq 2$, we fix a set of labels $L_{s,t} = \{a_1, a_2, \ldots, a_s, b_1, b_2, \ldots, b_t\}$ and define the set

$$Q_{s,t} = \{a_1 b_1 | a_s b_t\} \cup \bigcup_{i=1}^{s-1} \bigcup_{j=1}^{t-1} \{a_i a_{i+1} | b_j b_{j+1}\}$$

of quartets with $L(Q_{s,t}) = L_{s,t}$. We denote the quartet $a_1 b_1 | a_s b_t$ by q_0, and a quartet of the form $a_i a_{i+1} | b_j b_{j+1}$ by $q_{i,j}$.

Observation 1. *For all $s, t \geq 2$, $|Q_{s,t}| = (s-1)(t-1) + 1$.*

Lemma 2. *For all $s, t \geq 2$, $Q_{s,t}$ is incompatible.*

Proof. For each $i \in [s-1]$,

$$\bigcup_{j=1}^{t-1} \{a_i a_{i+1} | b_j b_{j+1}\} \subseteq Q_{s,t} \subseteq Q^*_{s,t}.$$

Then, by Lemma 1, it follows that for each $i \in [s-1]$, $a_i a_{i+1} | b_1 b_t \in Q^*_{s,t}$. So,

$$\bigcup_{i=1}^{s-1} \{b_1 b_t | a_i a_{i+1}\} \subseteq Q^*_{s,t}.$$

Then, again by Lemma 1, it follows that $b_1 b_t | a_1 a_s \in Q^*_{s,t}$. But then $\{a_1 b_1 | a_s b_t, b_1 b_t | a_1 a_s\} \subseteq Q^*_{s,t}$. It follows that any tree that displays $Q_{s,t}$ must display both $a_1 b_1 | a_s b_t$ and $b_1 b_t | a_1 a_s$. However, no such tree exists. Hence, $Q_{s,t}$ is incompatible. □

Lemma 3. *For all $s, t \geq 2$, every proper subset of $Q_{s,t}$ is compatible.*

Proof. Since every subset of a compatible set of quartets is compatible, it suffices to show that for every $q \in Q_{s,t}$, $Q_{s,t} \setminus \{q\}$ is compatible. Let $q \in Q_{s,t}$. Either $q = q_0$ or

$q = q_{x,y}$ for some $1 \leq x < s$ and $1 \leq y < t$. In either case, we exhibit a tree witnessing that $Q_{s,t} \setminus \{q\}$ is compatible.

Case 1. Suppose $q = q_0$. We build the tree T as follows: There is a node ℓ for each label $\ell \in L_{s,t}$ and two additional nodes a and b along with the edge ab. There is an edge $a_x a$ for every $a_x \in L_{s,t}$, and an edge $b_x b$ for every $b_x \in L_{s,t}$. There are no other nodes or edges in T. See Figure 2(a) for an illustration. Now consider any quartet $q \in Q_{s,t} \setminus \{q_0\}$. Then $q = a_i a_{i+1}|b_j b_{j+1}$ for some $1 \leq i < s$ and $1 \leq j < t$. Then, the minimal subgraph of T connecting leaves with labels in $\{a_i, a_{i+1}, b_j, b_{j+1}\}$ is the quartet q. Hence T displays q.

Case 2. Suppose $q = q_{x,y}$ for some $1 \leq x < s$ and $1 \leq y < t$. We build the tree T as follows: There is a node ℓ for each label $\ell \in L_{s,t}$ and six additional nodes a_ℓ, b_ℓ, ℓ, h, a_h, and b_h. There are edges $a_\ell \ell$, $b_\ell \ell$, ℓh, $h a_h$, and $h b_h$. For every $a_i \in L_{s,t}$, there is an edge $a_i a_\ell$ if $i \leq x$, and an edge $a_i a_h$ if $i > x$. For every $b_j \in L_{s,t}$ there is an edge $b_j b_\ell$ if $j \leq x$, and an edge $b_j b_h$ if $j > y$. There are no other nodes or edges in T. See Figure 2(b). Now consider any quartet $q \in Q_{s,t} \setminus \{q_{x,y}\}$. Either $q = q_0$ or $q = q_{i,j}$ where $i \neq x$ or $j \neq y$. If $q = q_0$, then the minimal subgraph of T connecting leaves with labels in $\{a_1, b_1, a_s, b_t\}$ is the subtree of T induced by the nodes in $\{a_1, a_\ell, \ell, b_\ell, b_1, a_s, a_h, h, b_h, b_t\}$. Suppressing all degree two vertices results in a tree that is the same as q_0. So T displays q. So assume that $q = a_i a_{i+1}|b_j b_{j+1}$ where $i \neq x$ or $j \neq y$. We define the following subset of the nodes in T:

$$V = \begin{cases} \{a_i, a_{i+1}, a_\ell, \ell, b_\ell, b_j, b_{j+1}\} & \text{if } i < x \text{ and } j < y, \\ \{a_i, a_{i+1}, a_\ell, \ell, b_y, b_\ell, h, b_h, b_{y+1}\} & \text{if } i < x \text{ and } j = y, \\ \{a_i, a_{i+1}, a_\ell, \ell, h, b_h, b_j, b_{j+1}\} & \text{if } i < x \text{ and } j > y, \\ \{a_x, a_\ell, \ell, h, a_h, a_{x+1}, b_\ell, b_j, b_{j+1}\} & \text{if } i = x \text{ and } j < y, \\ \{a_x, a_\ell, \ell, h, a_h, a_{x+1}, b_h, b_j, b_{j+1}\} & \text{if } i = x \text{ and } j > y, \\ \{a_j, a_{j+1}, a_h, h, \ell, b_\ell, b_j, b_{j+1}\} & \text{if } i > x \text{ and } j < y, \\ \{a_j, a_{j+1}, a_h, h, b_y, b_\ell, \ell, b_h, b_{y+1}\} & \text{if } i > x \text{ and } j = y, \\ \{a_j, a_{j+1}, a_h, h, b_h, b_j, b_{j+1}\} & \text{if } i > x \text{ and } j > y. \end{cases}$$

Now, the subgraph of T induced by the nodes in V is the minimal subgraph of T connecting leaves with labels in q. Suppressing all degree two vertices gives q. Hence, T displays q.

□

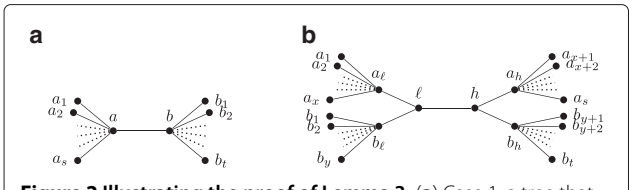

Figure 2 Illustrating the proof of Lemma 3. (**a**) Case 1: a tree that displays $Q_{s,t} \setminus \{q_0\}$. (**b**) Case 2: a tree that displays $Q_{s,t} \setminus \{q_{x,y}\}$.

With $s = \lfloor \frac{n}{2} \rfloor$ and $t = \lceil \frac{n}{2} \rceil$, Observation 1and Lemmas 2 and 3 imply the following theorem.

Theorem 2. *For every integer $n \geq 4$, there exists a set Q of quartets over n taxa such that all of the following conditions hold.*

1. *Q is incompatible.*
2. *Every proper subset of Q is compatible.*
3. *$|Q| = \lfloor \frac{n-2}{2} \rfloor \cdot \lceil \frac{n-2}{2} \rceil + 1$.*

Incompatible quartets on five taxa

When Q is a set of quartets over five taxa, we show that the set of quartets given by Theorem 2 is as large as possible. We hope that the technique used in the proof of the following theorem might be useful in proving tight bounds for $n > 5$.

Theorem 3. *If Q is an incompatible set of quartets over five taxa such that every proper subset of Q is compatible, then $|Q| \leq 3$.*

Proof. Let Q be an incompatible set of quartets with $L(Q) = \{a, b, c, d, e\}$ and $q_0 = ab|cd \in Q$. We will show that Q contains an incompatible subset of at most three quartets. If Q contains two different quartets on the same four taxa, then Q must contain an incompatible pair of quartets. So, we may assume that each quartet is on a unique subset of four of the five taxa. Hence, every pair of quartets in Q shares three taxa in common. We have the following two cases.

Case 1: Q contains at least one of the quartets $ac|be$, $ac|de$, $ad|be$, $ad|ce$, $ae|bc$, $ae|bd$, $bc|de$, or $bd|ce$. W.l.o.g. we may assume that Q contains $q_1 = ac|de$, as all other cases are symmetric. By (R2), $\{q_0, q_1\} \vdash ab|ce$. Then, by (R1), $\{q_0, q_1, ab|ce\} \vdash ab|de$. Then, again by (R1), $\{q_0, q_1, ab|ce, ab|de\} \vdash bc|de$. Now let $Q' = \{q_0, q_1, ab|ce, ab|de, bc|de\}$. Now, any quartet in Q must be either in Q' or be pairwise incompatible with a quartet in Q'. Since Q' is compatible, but by assumption, Q is incompatible, Q must contain a quartet q_2 that is pairwise incompatible with some quartet in Q'. Hence, $\{q_0, q_1, q_2\}$ is an incompatible subset of Q.

Case 2: Q contains none of the quartets $ac|be$, $ac|de$, $ad|be$, $ad|ce$, $ae|bc$, $ae|bd$, $bc|de$, or $bd|ce$. Then every quartet in Q is either of the form $ab|xy$ where $\{x, y\} \neq \{c, d\}$, or $cd|xy$ where $\{x, y\} \neq \{a, b\}$. But then

Q is compatible, contradicting our assumption that Q is incompatible.

In either case, the theorem holds. □

Incompatible quartets on arbitrarily many taxa

We say a set Q of compatible quartets is *redundant* if for some $q \in Q$, $Q \setminus \{q\} \vdash q$; otherwise, we say that Q is *irredundant*. The following lemma establishes a connection between sets of irredundant quartets and minimal sets of incompatible quartets.

Lemma 4. *If Q is incompatible, but every proper subset of Q is compatible, then every proper subset of Q is irredundant.*

Proof. Suppose that Q is incompatible and every proper subset of Q is compatible. Furthermore, suppose that some proper subset Q' of Q is redundant. Since every compatible superset of a redundant set of quartets is also redundant, we may assume w.l.o.g., that there is a unique quartet $q \in Q \setminus Q'$ (i.e., $|Q| = |Q'| + 1$). Since Q' is redundant, there exists a $q' \in Q'$ such that $Q' \setminus \{q'\} \vdash q'$. But then $(Q' \setminus \{q'\}) \cup \{q\}$ is incompatible, contradicting that every proper subset of Q is compatible. □

It follows from Lemma 4 that any upper bound on the maximum cardinality of an irredundant set of quartets can be used to place an upper bound on the maximum cardinality of a set of quartets satisfying the first two conditions of Theorem 2. The theorem follows from [22].

Theorem 4. *Let Q be a set of quartets over a set of n taxa. If Q is irredundant, then Q has cardinality at most $(n-3)(n-2)^2/3$.*

Lemma 4 together with Theorem 4 gives the following upper bound on the maximum cardinality of a set Q of quartets over $n > 5$ taxa that satisfies the first two conditions of Theorem 2.

Theorem 5. *Let Q be a set of incompatible quartets over a set of n taxa such that every proper subset of Q is compatible. Then $|Q| \leq (n-3)(n-2)^2/3 + 1$.*

Incompatible characters

There is a natural correspondence between quartet compatibility and character compatibility that we now describe. Let Q be a set of quartets, $n = |L(Q)|$, and $r = n - 2$. For each $q = ab|cd \in Q$, we define the r-state character corresponding to q, denoted χ_q, as the character where a and b have state 0 for χ_q; c and d have state 1 for χ_q; and, for each $\ell \in L(Q) \setminus \{a, b, c, d\}$, there is a state s of χ_q such that ℓ is the only label with state s for character χ_q (see Example 2). We define the set of r-state characters corresponding to Q by $C_Q = \bigcup_{q \in Q} \{\chi_q\}$.

Example 2. *Consider the quartets and characters given in Figure 1(a): χ_{q_1} is the character corresponding to q_1, χ_{q_2} is the character corresponding to q_2, and χ_{q_3} is the character corresponding to q_3.*

The following lemma relating quartet compatibility to character compatibility is well known [24], and its proof is omitted here.

Lemma 5. *A set Q of quartets is compatible if and only if C_Q is compatible.*

The next theorem allows us to use our result on quartet compatibility to establish a lower bound on $f(r)$.

Theorem 6. *Let Q be a set of incompatible quartets over n labels such that every proper subset of Q is compatible, and let $r = n - 2$. Then, there exists a set C of $|Q|$ r-state characters such that C is incompatible, but every proper subset of C is compatible.*

Proof. We claim that C_Q is such a set of incompatible r-state characters. Since for two quartets $q_1, q_2 \in Q$, $\chi_{q_1} \neq \chi_{q_2}$, it follows that $|C_Q| = |Q|$. Since Q is incompatible, it follows by Lemma 5 that C_Q is incompatible. Let C' be any proper subset of C. Then, there is a proper subset Q' of Q such that $C' = C_{Q'}$. Then, since Q' is compatible, it follows by Lemma 5 that C' is compatible. □

Theorem 2 together with Theorem 6 gives the main theorem of this paper.

Theorem 7. *For every integer $r \geq 2$, there exists a set C of r-state characters such that all of the following hold.*

1. *C is incompatible.*
2. *Every proper subset of C is compatible.*
3. *$|C| = \lfloor \frac{r}{2} \rfloor \cdot \lceil \frac{r}{2} \rceil + 1$.*

Proof. By Theorem 2 and Observation 1, there exists a set Q of $\lfloor \frac{r}{2} \rfloor \cdot \lceil \frac{r}{2} \rceil + 1$ quartets over $r + 2$ labels that that are incompatible, but every proper subset is compatible, namely $Q_{\lfloor \frac{r+2}{2} \rfloor, \lceil \frac{r+2}{2} \rceil}$. The theorem follows from Theorem 6. □

The quadratic lower bound on $f(r)$ follows from Theorem 7.

Corollary 2. $f(r) \geq \lfloor \frac{r}{2} \rfloor \cdot \lceil \frac{r}{2} \rceil + 1$.

Three-State Characters

In the remainder of this section we focus on the case when $r = 3$, and thus, fix C to be an arbitrary set of 3-state characters over a set S of taxa. Lam, Gusfield, and Sridhar [9] recently established that $f(3) = 3$, and they completely characterized the sets of pairwise compatible 3-state characters by the existence of one of four forbidden intersection patterns. We give an independent proof that $f(3) = 3$. We then completely characterize the sets of pairwise compatible 3-state characters by a single forbidden intersection pattern. Our proof uses several structural results from the algorithm for the three-state perfect phylogeny problem given by Kannan and Warnow [7].

The Algorithm of Kannan and Warnow

The algorithm of [7] takes a divide and conquer approach to determining the compatibility of a set of three-state characters. An instance is reduced to subproblems by finding a partition S_1, S_2 of the taxon set S of C with both of the following properties:

1. $2 \leq |S_i| \leq n-2, i = 1, 2$.
2. Whenever C is compatible S there is a perfect phylogeny P that contains an edge e whose removal breaks P into subtrees P_1 and P_2 with $L(P_i) = S_i, i = 1, 2$.

A partition of S satisfying both of these properties is a *legal partition*, and the following theorem shows that finding such a partition for a given set of characters is the crux of the algorithm.

Theorem 8. *[7] Given a set C of three state characters, we can in $O(nk)$ time either find a legal partition of S of determine that the set of characters is incompatible.*

Finding a legal partition

We now discuss the manner in which such a legal partition is found for a set of three-state characters C. Let T be a tree witnessing that C is compatible. The *canonical labeling* of T is the labeling where, for each internal node v of T, and each character $\alpha \in C$, if there are leaves x and y in different components of $T - \{v\}$ such that $\alpha(x) = \alpha(y)$, then $\alpha(v) = \alpha(x)$; otherwise $\alpha(v) = *$ where $*$ denotes a *dummy* state for C. Note that such a labeling of T always exists and is unique. We will assume that every compatible tree for C is canonically labeled.

The *tree-structure* for a character α in T is formed by repeatedly contracting edges of T connecting nodes that have the same state (other than $*$) for α. Note that this tree does not depend on the sequence of edge-contractions and is thus well defined. Furthermore, there is exactly one node for each state (other than the dummy state) of α, and each node labeled by $*$ has degree at least three. A tree-structure for α that is formed from some compatible tree for C is called a *realizable tree-structure* for α. There are four possible realizable tree-structures for a three-state character α which are shown in Figure 3.

To find a realizable tree structure for a character α, the algorithm examines the pairwise intersection patterns of α with every other character $\beta \in C$, and applies the following rules to rule out possible tree structures for α.

Rule 1. *Let α and β be two characters of C. If, under some relabeling of the states of α and β, we have that $\alpha_1 \subseteq \beta_1$, $\alpha_2 \cap \beta_2 \neq \emptyset$, and $\alpha_3 \cap \beta_2 \neq \emptyset$, then P^1 is not a realizable tree-structure for α. If this is the case, we say that α and β match Rule 1 with respect to α_1.*

Rule 2. [a]*Let α and β be two characters of C. If, under some relabeling of the states of α and β, we have that $\alpha_1 \cap \beta_1 \neq \emptyset$, $\alpha_2 \cap \beta_1 \neq \emptyset$, $\alpha_2 \cap \beta_2 \neq \emptyset$, and $\alpha_3 \cap \beta_2 \neq \emptyset$, then P^2 is the only possible realizable tree-structure for α. If this is the case, we say that α and β match Rule 2 with respect to α_2.*

The set Q_α^C of *candidate* tree-structures for α are all of those possible tree-structures for α that are not ruled out after comparing the intersection pattern of α with every other character in C and applying Rules 1 and 2.

The following theorem which follows from [7] shows that a legal partition is found by choosing an arbitrary $\alpha \in C$ for which $Q_\alpha^C \neq \emptyset$. Furthermore, if there is an $\alpha \in C$ for which $Q_\alpha^C = \emptyset$, then C is incompatible.

Theorem 9 ([7]). *If $Q_\alpha^C \neq \emptyset$, then we can find a legal partition of S.*

Corollary 3. *A set C of 3-state characters is compatible if and only if $Q_\alpha^C \neq \emptyset$ for every $\alpha \in C$.*

Tight bounds on three-state character compatibility

We use Corollary 3 to give upper bounds on the maximum cardinality of a minimal set of incompatible three-state characters.

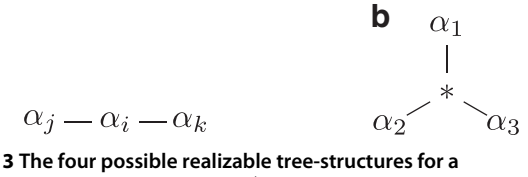

Figure 3 The four possible realizable tree-structures for a three-state character α. (**a**) A path P^i for each $i \in \{1, 2, 3\}$. (**b**) A star S^*.

Theorem 10. *Let C be a set of three-state characters on species set S. Then C is incompatible if and only if there exists a character $\alpha \in C$, and two distinct states α_i and α_j of α, such that both of the following hold:*

1. *There is a $\beta \in C$ where the intersection pattern of α and β matches Rule 2 with respect to α_i.*
2. *There is a $\gamma \in C$ where the intersection pattern of α and γ matches Rule 2 with respect to α_j.*

Proof. ($\Rightarrow$) If C is pairwise incompatible, then by Corollary 1, there is a pair $\alpha, \beta \in C$ whose intersection graph contains a cycle. Since the intersection graph is bipartite, this cycle must have length at least four and contain at least two states of each character. Let α_i and α_j be the two states of α on this cycle. Then, the intersection pattern of α and β matches Rule 2 with respect to both α_i and α_j, and so the theorem holds. So we may assume that C is incompatible but pairwise compatible.

It follows from Corollary 3 that there exists an $\alpha \in C$ such that $Q_\alpha^C = \emptyset$. Then there must exist a character $\beta \in C$ such that the intersection pattern of α and β matches Rule 2 with respect to some state α_i of α; otherwise $S^* \in Q_\alpha^C$. Hence, $Q_\alpha^C \subseteq \{P^i\}$. Then, since $Q_\alpha^C = \emptyset$, there must be a character $\gamma \in C$ such that the intersection pattern of α and γ places a constraint on Q_α^C that prevents Q_α^C from containing P^i. There are two possibilities.

Case 1: There is a state α_j of α where $j \neq i$ and the intersection pattern of α and γ matches Rule 2 with respect to α_j. In this case the theorem holds.

Case 2: The intersection pattern of α and γ matches Rule 1 with respect to α_i. W.l.o.g., we fix $i = 1$, and relabel the states of α, β, and γ so that $\alpha_1 \cap \beta_1 \neq \emptyset$, $\alpha_1 \cap \beta_2 \neq \emptyset$, $\alpha_2 \cap \beta_1 \neq \emptyset$, $\alpha_3 \cap \beta_2 \neq \emptyset$, $\alpha_1 \subseteq \gamma_1$, $\alpha_2 \cap \gamma_2 \neq \emptyset$, and $\alpha_3 \cap \gamma_2 \neq \emptyset$. Such a labeling exists since, by assumption, α and β matches Rule 2 with respect to α_1, and α and γ matches Rule 1 with respect to α_1.

If $\alpha_2 \cap \gamma_1 \neq \emptyset$, then the intersection pattern of α and γ matches Rule 2 with respect to α_2, in which case the theorem holds. If $\alpha_3 \cap \gamma_1 \neq \emptyset$, then the intersection pattern of α and γ matches Rule 2 with respect to α_3, in which case the theorem holds. So we may assume hat $\alpha_1 = \gamma_1$. Now, since $\alpha_1 \cap \beta_1 \neq \emptyset$, $\alpha_1 \cap \beta_2 \neq \emptyset$, and $\alpha_1 = \gamma_1$, we have that both $\beta_1 \cap \gamma_1 \neq \emptyset$ and $\beta_2 \cap \gamma_2 \neq \emptyset$.

γ_3 must have a nonempty intersection with at least one state of α, and since $\alpha_1 = \gamma_1$, we have that $\alpha_1 \cap \gamma_3 = \emptyset$. So γ_3 has a nonempty intersection with either α_2 or α_3. Due to the symmetry of the intersection graph of α and β, we may assume, w.l.o.g., that $\alpha_3 \cap \gamma_3 \neq \emptyset$.

By assumption, $\alpha_2 \cap \gamma_1 = \emptyset$, and if $\alpha_2 \cap \gamma_3 \neq \emptyset$, then the intersection graph of α and β contains a cycle, contradicting our assumption that C is pairwise compatible. So we may assume that $\alpha_2 \subset \gamma_2$. Then, since $\beta_1 \cap \alpha_2 \neq \emptyset$, we have that $\beta_1 \cap \gamma_2 \neq \emptyset$.

Let $s \in \alpha_3 \cap \beta_2$. Since, by assumption, $\alpha_3 \cap \gamma_1 = \emptyset$, we have that either $s \in \gamma_2$ or $s \in \gamma_3$. However, if $s \in \gamma_2$, then $\beta_2 \cap \gamma_2 \neq \emptyset$ and intersection graph of β and γ contains a cycle, contradicting our assumption that C is pairwise compatible. Hence $s \in \gamma_3$ and $\beta_2 \cap \gamma_3 \neq \emptyset$.

We have now established all of the edges of the intersection graph of α, β, and γ represented by the solid edges in Figure 4. Now, let $s_5 \in \alpha_3 \cap \gamma_2$. Now s_5 must be in some state of β. If $s_5 \in \beta_1$, then $s_5 \in \beta_1 \cap \alpha_3$ and the intersection graph of β and α contains a cycle, contradicting our assumption that C is pairwise compatible. If $s_5 \in \beta_2$, then $s_5 \in \beta_2 \cap \gamma_2$, and the intersection graph of β and γ contains a cycle, again contradicting our assumption that C is pairwise compatible. Hence $s_5 \in \beta_3$. Then, we have that $s_5 \in \beta_3 \cap \alpha_3$ and $s_5 \in \beta_3 \cap \gamma_2$, witnessing the dotted edges in Figure 4. So we have that the intersection pattern of β and α matches Rule 2 with β_2 as witness, and the intersection pattern of β and γ matches Rule 2 with β_1 as witness. Hence the theorem holds. □

Note that in the statement of Theorem 10, the characters β and γ are not necessarily distinct. In cases where they are not distinct, C contains an incompatible pair.

Corollary 4. *A set C of 3-state characters is compatible if and only if every subset of at most three characters of C is compatible.*

In [9], it was also shown that we can determine the compatibility of a pairwise compatible set C of three-state characters by testing the intersection patterns of C for the existence of one of a set of four forbidden patterns. As a corollary to Theorem 10, we have that a single forbidden pattern suffices to determine the compatibility of C.

Corollary 5. *A pairwise compatible set C of 3-state characters is compatible if and only if the partition intersection graph of C does not contain, up to relabeling of characters and states, the subgraph of Figure 5.*

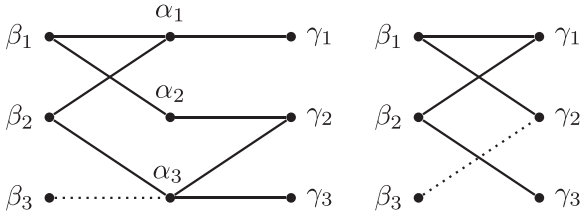

Figure 4 Illustrating the proof of Theorem 10.

Note that each edge of the graph of Figure 5 has one endpoint which is a state in α. It follows that we can find such a subgraph in the partition intersection graph of C by testing the intersection pattern of each pair of characters in C [10]. Furthermore, all p occurrences of the forbidden subgraph in the intersection graph of m characters on n taxa can be found in $O(m^2n + p)$ time. Whereas the forbidden subgraph given here is witnessed by eight taxa (or edges), each of the four forbidden subgraphs of [9] are witnessed by five taxa, making them better suited for taxon removal problems.

Incompatible Triplets

A *rooted phylogenetic tree* (or just *rooted tree*) is a tree whose leaves are in one to one correspondence with a label set $L(T)$, has a distinguished vertex called the *root*, and no vertex other than the root has degree two. See Figure 6(a) for an example. A rooted tree is *binary* if the root vertex has degree two, and every other internal (non-leaf) vertex has degree three. A *triplet* is a rooted binary tree with exactly three leaves. A triplet with label set $\{a, b, c\}$ is denoted $ab|c$ if the path between the leaves labeled a and b avoids the path between the leaf labeled c and the root vertex. For a tree T, and a label set $L \subseteq L(T)$, let T' be the minimal subtree of T connecting all the leaves with labels in L. The *restriction* of T to L, denoted by $T|L$, is the rooted tree obtained from T' by distinguishing the vertex closest to the root of T as the root of T', and suppressing every vertex other than the root having degree two. A rooted tree T *displays* another rooted tree T' if T' can be obtained from $T|L(T')$ by contracting edges. A rooted tree T displays a collection of rooted trees $\mathcal{T}$ if T displays every tree in $\mathcal{T}$. If such a tree T exists, then we say that $\mathcal{T}$ is compatible; otherwise, we say that $\mathcal{T}$ is incompatible. Given a collection of rooted trees $\mathcal{T}$, it can be determined in polynomial time if $\mathcal{T}$ is compatible [3,25].

The following theorems follow from the connection between collections of unrooted trees with at least one common label across all the trees, and collections of rooted trees [3].

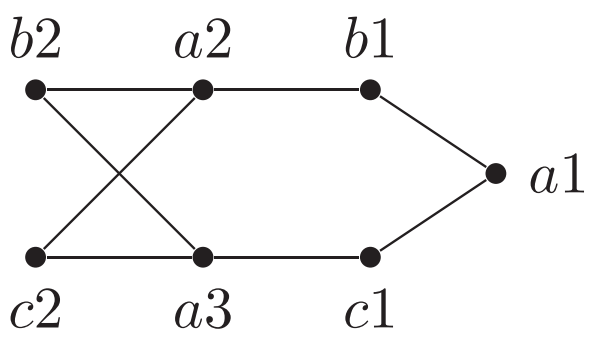

Figure 5 The forbidden subgraph for 3-state character compatibility.

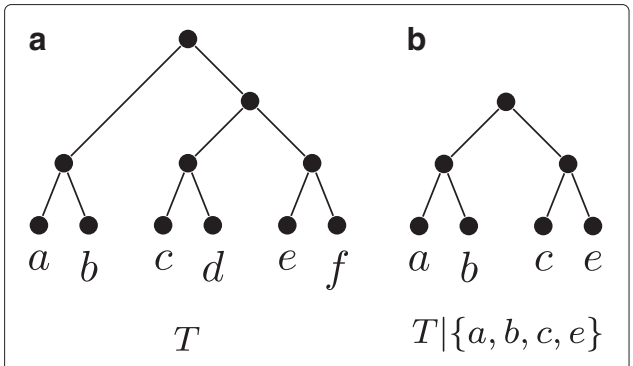

Figure 6 Example of rooted phylogenetic trees. (**a**) shows a tree T that is a witness that the triplets $ab|c$, $de|b$, $ef|c$, and $ec|b$ are compatible; (**b**) shows the tree T restricted to the label set $\{a, b, c, e\}$.

Theorem 11. *Let Q be a collection of quartets where every quartet in Q shares a common label ℓ. Let R be the set of triplets such that there exists a triplet $ab|c$ in R if and only if there exists a quartet $ab|c\ell$ in Q. Then, Q is compatible if and only if R is compatible.*

Let R be a collection of triplets. For a subset $S \subseteq L(R)$, we define the graph $[R, S]$ as the graph having a vertex for each label in S, and an edge $\{a, b\}$ if and only if $ab|c \in R$ for some $c \in S$. The following theorem is from page 439 of [26].

Theorem 12. *A collection R of rooted triplets is compatible if and only if $[R, S]$ is not connected for every $S \subseteq L(R)$ with $|S| \geq 3$.*

Corollary 6. *Let R be a set of rooted triplets such that R is incompatible but every proper subset of R is compatible. Then, $[R, L(R)]$ is connected.*

We now contrast our result on quartet compatibility with a result on triplets.

Theorem 13. *For every $n \geq 3$, if R is an incompatible set of triplets over n labels, and $|R| > n - 1$, then some proper subset of R is incompatible.*

Proof. For sake of contradiction, let R be a set of triplets such that R is incompatible, every proper subset of R is compatible, $|L(R)| = n$, and $|R| > n - 1$. The graph $[R, L(R)]$ will contain n vertices and at least n edges. Since each triplet in R is distinct, there will be a cycle C of length at least three in $[R, L(R)]$. Since R is incompatible but every proper subset of R is compatible, by Corollary 6, $[R, L(R)]$ is connected.

Consider any edge e in the cycle C. Let t be the triplet that contributed edge e in $[R, L(R)]$. Let $R' = R \setminus t$. Since the graph $[R, L(R)] - e$ is connected, $[R', L(R')]$ is connected. By Theorem 12, R' is incompatible. But $R' \subset R$, contradicting that every proper subset of R is compatible. □

To show the bound is tight, we first prove a more restricted form of Theorem 2.

Theorem 14. *For every $n \geq 4$, there exists a set of quartets Q with $|L(Q)| = n$, and a label $\ell \in L(Q)$, such that all of the following hold.*

1. *Every $q \in Q$ contains a leaf labeled by ℓ.*
2. *Q is incompatible.*
3. *Every proper subset of Q is compatible.*
4. *$|Q| = n - 2$.*

Proof. Consider the set of quartets $Q_{2,n-2}$. From Lemmas 2 and 3, $Q_{2,n-2}$ is incompatible but every proper subset of $Q_{2,n-2}$ is compatible. The set $Q_{2,n-2}$ contains exactly $n - 2$ quartets. From the construction, there are two labels in L which are present in all the quartets in $Q_{2,n-2}$. Set one of them to be ℓ. □

The following is a consequence of Theorems 14 and 11.

Corollary 7. *For every $n \geq 3$, there exists a set R of triplets with $|L(R)| = n$ such that all of the following hold.*

1. *R is incompatible.*
2. *Every proper subset of R is compatible.*
3. *$|R| = n - 1$.*

The generalization of the Fitch-Meacham examples given in [9] can also be expressed in terms of triplets. For any $r \geq 2$, let $L = \{a, b_1, b_2, \cdots, b_r\}$. Let

$$R_r = ab_r|b_1 \cup \bigcup_{i=1}^{r-1} ab_i|b_{i+1}$$

Let $Q = \{ab|c\ell : ab|c \in R_r\}$ for some label $\ell \notin L$. The set C_Q of r-state characters corresponding to the quartet set Q is exactly the set of characters built for r in [9]. In the partition intersection graph of C_Q, (following the terminology in [9]) labels ℓ and a correspond to the end cliques and the rest of the r labels $\{b_1, b_2, \cdots, b_r\}$ correspond to the r tower cliques. From Lemma 5 and Theorem 11, R_r is compatible if and only of Q is compatible.

Conclusion

We have shown that for every $r \geq 2, f(r) \geq \left\lfloor \frac{r}{2} \right\rfloor \cdot \left\lceil \frac{r}{2} \right\rceil + 1$, by showing that for every $n \geq 4$, there exists an incompatible set Q of $\left\lfloor \frac{n-2}{2} \right\rfloor \cdot \left\lceil \frac{n-2}{2} \right\rceil + 1$ quartets over a set of n labels such that every proper subset of Q is compatible. Previous results [1,6,9,13-15], along with our discussion in Section Incompatible Characters, show that our lower bound on $f(r)$ is tight for $r = 2$ and $r = 3$. For quartets, our discussion in Section Incompatible quartets gives an upper bound on the maximum cardinality of a minimal set of incompatible quartets. However, this argument does not extend to multi-state characters. Indeed, an upper bound on the maximum cardinality of a minimal set of incompatible r-state characters remains a central open question. We give the following conjecture.

Conjecture 2. $f(r) \in \Theta(r^2)$.

A less ambituous goal would be to narrow the gap between the upper bound of $O(n^3)$ and lower bound of $\Omega(n^2)$ on the maximum cardinality of a minimal incompatible set of quartets over n taxa given in Section Incompatible Quartets. Note that, due to Theorem 6, a proof of Conjecture 2 would also show that the number of incompatible quartets given in the statement of Theorem 2 is also as large as possible.

Endnote

[a]Rule 2 was state incorrectly in [7].

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
BS was responsible for the lower bounds on character compatibility, the upper and lower bounds on quartet compatibility, the characterization of three-state character compatibility, and wrote all portions of the manuscript other than the section on triplet compatibility. SV was responsible for the upper and lower bounds on triplet compatibility, contributed to the lower bounds on quartet and character compatibility, and wrote the portion of the manuscript on triplet compatibility. DFB supervised the project. All authors read and approved the final manuscript.

Acknowledgements
We thank Sylvain Guillemot, Mike Steel, and Rob Gysel for valuable comments. This work was supported in part by the National Science Foundation under grants CCF-1017189 and DEB-0829674.

Received: 20 December 2012 Accepted: 6 February 2013
Published: 1 April 2013

References

1. Semple C, Steel M: *Phylogenetics*. Oxford Lecture Series in Mathematics and its Applications. USA: Oxford University Press; 2003.
2. Bodlaender H, Fellows M, Warnow T: **Two strikes against perfect phylogeny.** In *Automata, Languages and Programming, Volume 623 of Lecture Notes in Computer Science*. Edited by Kuich W. Berlin/Heidelberg: Springer; 1992:273–283.
3. Steel M: **The complexity of reconstructing trees from qualitative characters and subtrees.** *J Classif* 1992, **9:**91–116.
4. Agarwala R, Fernández-Baca D: **A polynomial-time algorithm for the perfect phylogeny problem when the number of character states is fixed.** *SIAM J Comput* 1994, **23**(6):1216–1224.
5. Dress A, Steel M: **Convex tree realizations of partitions.** *Appl Math Lett* 1992, **5**(3):3–6.

6. Gusfield D: **Efficient algorithms for inferring evolutionary trees.** *Networks* 1991, **21:**19–28.
7. Kannan S, Warnow T: **Inferring evolutionary history from DNA sequences.** *SIAM J Comput* 1994, **23**(4):713–737.
8. Kannan S, Warnow T: **A fast algorithm for the computation and enumeration of perfect phylogenies.** *SIAM J Comput* 1997, **26**(6):1749–1763.
9. Lam F, Gusfield D, Sridhar S: **Generalizing the splits equivalence theorem and four Gamete condition: perfect phylogeny on three-state characters.** *SIAM J Discrete Math* 2011, **25**(3):1144–1175.
10. Shutters B, Fernández-Baca D: **A simple characterization of the minimal obstruction sets for three-state perfect phylogenies.** *Appl Math Lett* 2012, **25**(9):1226–1229.
11. Fernández-Baca D: **The Perfect Phylogeny Problem.** In *Steiner Trees in Industry*. Dordrecht: Kluwer; 2001:203–234.
12. Niedermeier R, Rossmanith P: **An efficient fixed-parameter algorithm for 3-Hitting Set.** *J Discrete Algorithms* 2003, **1:**89–102.
13. Buneman P: **The recovery of trees from measurements of dissimilarity.** In *Mathematics in the Archeological and Historical Sciences*. Edinburgh: Edinburgh University Press; 1971:387–395.
14. Estabrook GF, Johnson J, McMorris FR: **A mathematical foundation for the analysis of cladistic character compatibility.** *Math Biosci* 1976, **29**(1-2):181–187.
15. Meacham CA: **Theoretical and computational considerations of the compatibility of qualitative taxonomic characters.** In *Numerical Taxonomy, Volume G1 of Nato ASI series*. Heidelberg: Springer; 1983:304–314.
16. Fitch WM: **Toward finding the tree of maximum parsimony.** In *Proceedings of the 8th International Conference on Numerical Taxonomy*. San Francisco: Freeman; 1975:189–230.
17. Fitch WM: **On the problem of discovering the most parsimonious tree.** *Am Nat* 1977, **111**(978):223–257.
18. Habib M, To TH: **On a conjecture of compatibility of multi-states characters.** In *Algorithms in Bioinformatics, Volume 6833 of Lecture Notes in Computer Science*. Edited by Przytycka T, Sagot MF. Berlin/Heidelberg: Springer; 2011:116–127.
19. Buneman P: **A characterization of rigid circuit graphs.** *Discrete Math* 1974, **9:**205–212.
20. Colonius H, Schulze HH: **Tree structures for proximity data.** *Br J Math Stat Psychol* 1981, **34**(2):167–180.
21. Dekker MCH: Reconstruction methods for derivation trees. *Master's thesis*. Amsterdam: Vrije Universiteit; 1986.
22. Dietrich M, McCartin C, Semple C: **Bounding the maximum size of a minimal definitive set of quartets.** *Inf Process Lett* 2012, **112**(16):651–655.
23. Grünewald S, Huber KT: **Identifying and defining trees.** In *Reconstructing Evolution: New Mathematical and Computational Advances*. Edited by Gascuel O, Steel M: Oxford University Press; 2007.
24. Steel M: **Personal communications.** 2012.
25. Aho AV, Sagiv Y, Szymanski TG, Ullman JD: **Inferring a tree from lowest common ancestors with an application to the optimization of relational expressions.** *SIAM J Comput* 1981, **10**(3):405–421.
26. Bryant D, Steel M: **Extension operations on sets of leaf-labelled trees.** *Adv Appl Math* 1995, **16:**425–453.

doi:10.1186/1748-7188-8-11
Cite this article as: Shutters *et al.*: Incompatible quartets, triplets, and characters. *Algorithms for Molecular Biology* 2013 **8**:11.