SIAM J. COMPUT.
Vol. 32, No. 5, pp. 1115–1127


# A POLYNOMIAL-TIME ALGORITHM FOR NEAR-PERFECT PHYLOGENY*

DAVID FERNÁNDEZ-BACA† AND JENS LAGERGREN‡

**Abstract.** A parameterized version of the Steiner tree problem in phylogeny is defined, where the parameter measures the amount by which a phylogeny differs from "perfection." This problem is shown to be solvable in polynomial time for any fixed value of the parameter.

**Key words.** algorithms, computational biology, character-based methods, evolutionary trees, parsimony, perfect phylogeny, phylogeny, Steiner tree

**AMS subject classifications.** 68Q25, 68R05, 68R10, 68W40, 92B99

**DOI.** 10.1137/S0097539799350839

## 1. Introduction.

A fundamental problem in biology and linguistics is that of inferring the evolutionary history of a set of taxa, each of which is specified by the set of *traits* or *characters* that it exhibits [4, 6, 15]. Formally, let $C$ be a set of *characters*, and for every $c \in C$ let $\mathcal{A}_c$ be the set of allowable *states* for character $c$. Let $m = |C|$ and $r_c = |\mathcal{A}_c|$. A *species* $s$ is an element of $\mathcal{A}_1 \times \cdots \times \mathcal{A}_m$; $c(s)$ is referred to as the *state of character* $c$ *for* $s$. A *phylogeny* for a set of $n$ distinct species $S$ is tree $T$ with the following properties:

(C1) $S \subseteq V(T) \subseteq \mathcal{A}_1 \times \cdots \times \mathcal{A}_m$,

(C2) every leaf in $T$ is in $S$.

Define the *length* of a phylogeny $T$ for $S$ as

$$\text{length}(T) = \sum_{(u,v)\in E(T)} \text{dist}(u, v),$$

where, for any two species $u$, $v$, $\text{dist}(u, v)$ denotes the number of character states in which $u$ and $v$ differ (that is, $\text{dist}(u, v)$ is the Hamming distance between $u$ and $v$). The *Steiner tree problem in phylogeny* (STP) is to find a phylogeny $T$ of minimum length for a given set of species $S$.

STP and many of its variants are known to be NP-hard [7, 3]. While polynomial-time approximation algorithms with constant ratio bound are known for this problem (for a recent example, see [11]), there are limits to the approximability of STP [5].

STP is related to the problem of determining whether $S$ has a *perfect* phylogeny, i.e., one that satisfies (C1), (C2), and the following:

(C3) For every $c \in C$ and every $\sigma \in \mathcal{A}_c$, the set of all $u \in V(T)$ such that $c(u) = \sigma$ induces a subtree of $T$.

*Received by the editors January 27, 1999; accepted for publication (in revised form) March 13, 2003; published electronically August 6, 2003. A preliminary version of this paper was presented at the 23rd International Conference on Automata, Languages, and Programming, Paderborn, Germany, 1996.

http://www.siam.org/journals/sicomp/32-5/35083.html

†Department of Computer Science, Iowa State University, Ames, IA 50011-1041 (fernande@cs.iastate.edu). The work of this author was supported in part by the National Science Foundation under grants CCR-9211262, CCR-9520946, and CCR-9988348.

‡Department of Numerical Analysis and Computer Science, Royal Institute of Technology, Stockholm, Sweden (jensl@nada.kth.se). The work of this author was supported by grants from the NFR and TFR.

The perfect phylogeny problem was shown to be NP-complete by Bodlaender, Fellows, and Warnow [2] and, independently, by Steel [14]. This motivated the study of the fixed-parameter versions of the problem, where either $m$ or $r$ is fixed. Both versions have been shown to be polynomially solvable, the first by McMorris, Warnow, and Wimer [13], and the second by Agarwala and Fernández-Baca [1]. The time bound of the latter's algorithm was improved by Kannan and Warnow [12].

If a set of species admits a perfect phylogeny, the underlying set of characters $C$ is *compatible*; thus, the perfect phylogeny problem is often called the *character compatibility problem*. In practice most sets of characters are incompatible, and thus a natural problem is to find a maximum-cardinality subset of $C$ that is compatible. This problem is, unfortunately, equivalent to CLIQUE [8] and hence not only NP-hard, but also extremely hard to solve approximately [10].

The difference between $m$ and the maximum-cardinality compatible subset of $C$ is one measure of the degree of compatibility of a set of species. Here we study a measure of incompatibility that we believe is equally natural, which is motivated by the following result [1, 9].

THEOREM 1. *Let $T^*$ be a phylogeny for $S$. Then* $\text{length}(T^*) \geq \sum_{c \in C}(r_c - 1)$ *and $T^*$ is a perfect phylogeny if and only if* $\text{length}(T^*) = \sum_{c \in C}(r_c - 1)$.

Thus, the length of a perfect phylogeny (assuming one exists) gives a tight lower bound on the length of any phylogeny for $S$. Motivated by this observation, let us define the *penalty* of a phylogeny $T$ as

$$\text{penalty}(T) = \text{length}(T) - \sum_{c \in C}(r_c - 1).$$

Obviously, STP can be rephrased as the problem of finding a phylogeny $T$ such that $\text{penalty}(T)$ is minimum. We are interested in the fixed-parameter version of the problem, namely, given a set of species $S$ and an integer $q$, does $S$ have a phylogeny with penalty at most $q$? We show that for each fixed $q$ and $r$, the resulting "near-perfect" phylogeny problem can be solved in polynomial time. The running time of our algorithm is a polynomial whose degree depends on the parameters, making the algorithm practical only for small values of the parameters. On the other hand, the flexibility of allowing one or more characters to violate condition (C3) by some fixed amount may extend the range of applicability of character-based methods.

Our near-perfect phylogeny algorithm shares several ideas with earlier work on the perfect phylogeny problem [1, 12]. As in the algorithms for the latter problem, we rely on the observation that there is a polynomially bounded number of ways in which species can be partitioned into subfamilies that respect state boundaries for some character. (See section 2 for a precise definition.) The approach is to build subphylogenies for these subfamilies, proceeding by increasing cardinality. Subphylogenies are joined through their roots to form subphylogenies for larger subfamilies.

The construction of subphylogenies is complicated by issues that do not arise in the perfect phylogeny problem. Each edge in a perfect phylogeny corresponds to a *character partition*, i.e., a partition of $S$ into subfamilies such that there exists a character $c$ on which no state is shared between species of different subfamilies. This property and the fact that the number of character partitions is polynomially bounded when $r$ is fixed are keys to the efficient solution of the perfect phylogeny problem. Unfortunately, it can easily be seen that imperfect phylogenies may have bad edges, i.e., edges not inducing character partitions. We show, however, that the number of bad edges is polynomially bounded when the penalty is bounded.

Our strategy to build a subphylogeny for a subset of species is therefore to generate different candidate trees consisting only of bad edges and use them as skeletons from which to hang subphylogenies for character subfamilies. From among all of the trees thus enumerated, we select the one that results in the least penalty. It is nontrivial to determine which subphylogenies to connect to a candidate bad tree, because there is no a priori bound on the degree of a vertex. (Such a bound would imply that the subphylogenies could be found in polynomial time by simply trying all combinations of character subfamilies.) We show that it suffices to enumerate a polynomially bounded number of labeled candidate trees.

The rest of the paper is organized as follows. Section 2 gives definitions and notation. Section 3 explains how to compute perfect phylogenies. The properties of low-penalty phylogenies and subphylogenies are studied in section 4. In particular, bounds are derived there on the number of bad edges in a near-perfect phylogeny and on the amount of information that must be enumerated to construct such a phylogeny. Our near-perfect phylogeny algorithm is presented in section 5. Section 6 concludes the paper.

**2. Basic definitions and notation.** The vertex sets of all trees are assumed to be subsets of $\mathcal{A}_1 \times \cdots \times \mathcal{A}_m$. Note that this implies that every two adjacent nodes are distinct. No generality is lost, since a tree that does not satisfy this condition can be transformed into one that does and that has at most the same length.

Throughout the paper, $r$ denotes $\max_{c \in C} r_c$.

Let $c$ be a character. We assume that each state in $\mathcal{A}_c$ is exhibited by some element of $S$. Obviously, any state that is not exhibited by any species can be deleted from $\mathcal{A}_c$. We assume that $\mathcal{A}_i \cap \mathcal{A}_j = \emptyset$ for $i \neq j$. No generality is lost by making this assumption, since it can always be enforced by replacing each state $\sigma$ on every character $c$ by the state $(\sigma, c)$.

Let $T$ be a tree, and let $\sigma \in \mathcal{A}_c$. Then $T[\sigma]$ denotes the subgraph of $T$ induced by all nodes $v$ such that $c(v) = \sigma$. For any edge $(u, v) \in T$, $T_u$ and $T_v$ denote the components of $T - \{(u, v)\}$ containing $u$ and $v$, respectively.

DEFINITION 1. *Let $T$ be a phylogeny for $S$. Character $c$ is* convex *in $T$ if for every $\sigma \in \mathcal{A}_c$, $T[\sigma]$ is connected. If $T[\sigma]$ is not connected, $\sigma$ is a* penalty state *in $T$ and $c$ is* nonconvex *in $T$.*

In what follows, $C_p \subseteq C$ denotes a set of characters that are required to be convex.

DEFINITION 2. *Let $T$ be a phylogeny for $S$. $T$ is $q$-*near-perfect *if* penalty$(T) \leq q$. *$T$ is $C_p$-*perfect *if every $c \in C_p$ is convex in $T$. If $C_p = C$, $T$ is simply called* perfect. *A* minimum $C_p$-perfect phylogeny *is a $C_p$-perfect phylogeny of minimum length.*

An example of a $C_p$-perfect phylogeny is shown in Figure 1. For clarity, the set of states for every character is written as $\{0, 1, 2, 3, 4\}$; in reality, $\mathcal{A}_{c_i} = \{(j, c_i)\}_{j=0}^{4}$ for $i = 1, 2, 3, 4$. We shall refer to Figure 1 throughout the paper to illustrate various concepts.

Clearly, a $C_p$-perfect phylogeny may not exist for a given set $C_p$. Note also that, by Theorem 1, all perfect phylogenies have the same (minimum) length, so it is redundant to talk about minimum perfect phylogenies.

DEFINITION 3. *Two subsets $X$, $Y$ of $S$* share a state *on $c \in C$ if there exists a state $\sigma$ of $c$ such that $c(x) = c(y) = \sigma$ for some $x \in X$, $y \in Y$. State $\sigma$ is referred to as a* shared state.

DEFINITION 4. *A* character partition *with respect to a character $c$ is a partition $(S_1, S_2)$ of $S$ such that no species in $S_1$ shares a state of $c$ with any species of $S_2$. The subsets $S_1$ and $S_2$ are* character subfamilies. *A character subfamily $Q$ is* proper *if at*

| | $c_1$ | $c_2$ | $c_3$ | $c_4$ |
|---|---|---|---|---|
| $s_1$ | 0 | 2 | 1 | 2 |
| $s_2$ | 4 | 2 | 1 | 1 |
| $s_3$ | 0 | 2 | 3 | 1 |
| $s_4$ | 0 | 4 | 4 | 0 |
| $s_5$ | 0 | 1 | 0 | 0 |
| $s_6$ | 3 | 2 | 1 | 0 |
| $s_7$ | 2 | 0 | 1 | 3 |
| $s_8$ | 1 | 0 | 1 | 2 |
| $s_9$ | 0 | 2 | 2 | 4 |
| $s_{10}$ | 0 | 3 | 2 | 2 |

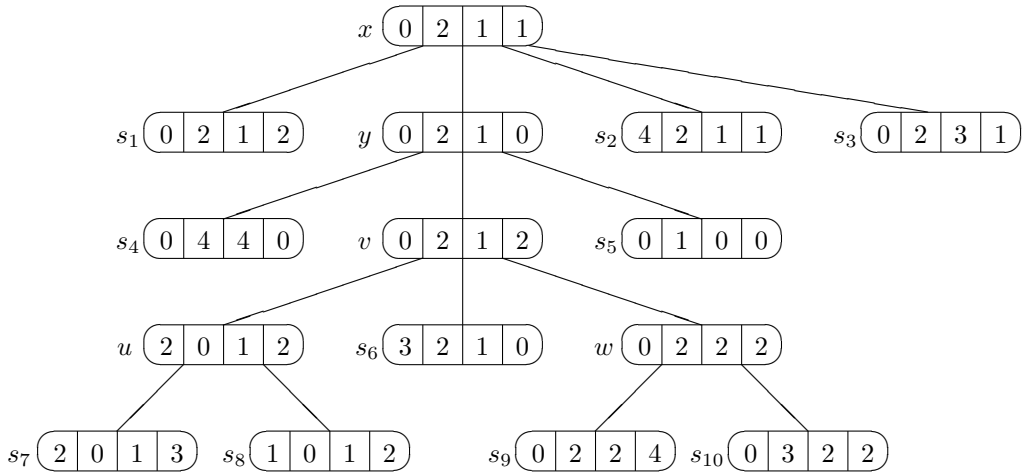


FIG. 1. *Top: A set of ten species described by their states on a set of characters* $C = \{c_1, c_2, c_3, c_4\}$*, each with five states. Bottom: A* $C_p$*-perfect phylogeny for the set of species, where* $C_p = \{c_1, c_2, c_3\}$*. The length of the tree is* 18*; its penalty is* 2.

*most one state is shared between* $Q$ *and* $S - Q$ *for every* $c \in C_p$*.*

From this point forward, all character subfamilies that we consider are assumed to be proper.

The number of character subfamilies is $O(2^r m)$, since at most $2^r$ are defined by the states of any given character [1]. This bound is polynomial when $r$ is fixed, which motivates the following approach to computing minimum $C_p$-perfect phylogenies: Enumerate the subfamilies by increasing cardinality, and for each subfamily find a minimum-length rooted $C_p$-perfect phylogeny made up of phylogenies for smaller subfamilies. Since our goal is to compose the phylogenies by linking roots through edges, the permissible states for the roots are partially determined by convexity. To formalize these ideas, we need some definitions. In what follows, $*$ denotes an unspecified state, which is in none of the $\mathcal{A}_i$'s.

DEFINITION 5. *Let* $Q \subseteq S$ *be a character subfamily. The* splitting vector *of* $Q$ *is the species* $Sv(Q)$ *where, for each character* $c$*, if* $c \in C_p$ *and* $Q$ *and* $S - Q$ *share state* $\sigma$ *on character* $c$*, then* $c(Sv(Q)) = \sigma$*; otherwise,* $c(Sv(Q)) = *$*.*

DEFINITION 6. *Let* $Q$*,* $Q_1$ *be character subfamilies such that* $Q_1 \subset Q$*.* $Q$ *and* $Q_1$ *are* compatible *if for every* $c \in C_p$ *such that* $c(Sv(Q)), c(Sv(Q_1)) \neq *$*,* $c(Sv(Q)) =$

$c(Sv(Q_1))$.

Intuitively, if $Q$ and $Q_1$ are compatible, there conceivably exists a $C_p$-perfect phylogeny for $Q \cup \{Sv(Q)\}$ such that one of the subtrees of $Sv(Q)$ is a phylogeny for $Q_1 \cup \{Sv(Q_1)\}$. In such a phylogeny, the states of the root on some characters $c \in C_p$ such that $c(Sv(Q)) = *$ may have to take on specific values, because a state on $c$ may be shared between $Q_1$ and $S - Q_1$ that is not shared between $Q$ and $S - Q$. This motivates the following definition.

DEFINITION 7. *Let $Q, Q_1$ be compatible character subfamilies such that $Q_1 \subset Q$. The* splitting vector *for $(Q, Q_1)$ is the species $Sv(Q, Q_1)$ where for each character $c$, if $c \in C_p$ and state $\sigma$ is shared between $Q$ and $S - Q$ or between $Q_1$ and $S - Q_1$, then $c(Sv(Q, Q_1)) = \sigma$; otherwise, $c(Sv(Q, Q_1)) = *$.*

DEFINITION 8. *Let $Q \subseteq S$ and $x$ be a species. Then $\sim_x$ denotes the equivalence relation on $Q$ defined as the transitive closure of the following relation $R_x$: For $s, t \in Q$, $(s, t) \in R_x$ if there exists a character $c$ such that $c(s) = c(t) \neq c(x) \neq *$. Denote by $Q/x$ the collection of equivalence classes of $\sim_x$.*

Observe that each of the sets in $Q/x$ must be in the same connected component of $T - \{x\}$ for any perfect (not just $C_p$-perfect) phylogeny $T$ for $Q \cup \{x\}$.

DEFINITION 9. *Let $T$ be a phylogeny for $S$ and let $e = (u, v)$ be an edge of $T$. Then $(S \cap V(T_u), S \cap V(T_v))$ is an* edge partition *of $S$ (with respect to $T$). The subsets $S \cap V(T_u)$ and $S \cap V(T_v)$ are* edge subfamilies. *Edge $(u, v)$ is* good *if the partition $(S \cap V(T_u), S \cap V(T_v))$ induced by $e$ is a character partition; otherwise, $e$ is* bad.

To close this section, we illustrate some of the concepts introduced here, making reference to Figure 1. Let $Q_1 = \{s_9, s_{10}\}$ and $Q = \{s_7, s_8, s_9, s_{10}\}$. Then, $Sv(Q_1) = (0, 2, *, *)$ and $Sv(Q) = (0, 2, 1, *)$; thus, $Q_1$ and $Q$ are compatible, and $Sv(Q, Q_1) = (0, 2, 1, *)$. In the phylogeny shown, edges $(s_1, x)$, $(x, y)$, and $(y, v)$ are bad; all other edges are good.

**3. Finding perfect phylogenies.** Before studying near-perfect phylogenies, we review the perfect phylogeny algorithm of Agarwala and Fernández-Baca and the improvements devised by Kannan and Warnow. The algorithm relies on two facts. The first is the aforementioned polynomial bound (for fixed $r$) on the number of character subfamilies. The second is that perfect phylogenies can be assembled from phylogenies for character subfamilies, because, as shown in [1], every edge in a perfect phylogeny for $S$ is good.

DEFINITION 10. *Let $Q$ be a character subfamily. A* perfect subphylogeny *for $Q$ is a rooted perfect phylogeny for $Q$, whose root $x$ satisfies $c(x) = c(Sv(Q))$ for all $c$ such that $c(Sv(Q)) \neq *$, and $c(x) = c(s)$ for some $s \in Q$ otherwise.*

It is straightforward to verify that if $Q_1$ and $Q_2 = S - Q_1$ have perfect subphylogenies $T_1$ and $T_2$, respectively, then the tree obtained by connecting the roots of $T_1$ and $T_2$ by an edge is a perfect phylogeny for $S$.

DEFINITION 11. *Let $Q, Q_1$ be compatible character subfamilies such that $Q_1 \subset Q$. A* perfect subphylogeny *for $(Q, Q_1)$ is a rooted perfect phylogeny for $Q$, whose root $x$ is such that* (i) *$c(x) = c(Sv(Q, Q_1))$ for all $c$ such that $c(Sv(Q)) \neq *$, and $c(x) = c(s)$ for some $s \in Q$ otherwise, and* (ii) *the removal of $x$ partitions $Q$ into subsets some of which union to $Q_1$.*

The following result is proved in [12, 1].

LEMMA 2. *Suppose that $Q$ is a character subfamily and that $Q_1 \subset Q$ has a subphylogeny. Let $Q_2 = Q - Q_1$. Then, $(Q, Q_1)$ has a subphylogeny if and only if* (i) *$Q_2$ has a subphylogeny or* (ii) *every element of $Q_2/Sv(Q, Q_1)$ has a subphylogeny. In case* (ii)*, $c(Sv(Q, Q_1)) \neq *$ for every character $c$.*

SUBPHYLOGENY($Q$).
For each subfamily $Q_1 \subset Q$ compatible with $Q$ do the following:

1. Let $T_{Q_1} = \mathcal{N}(Q_1)$.
2. If $T_{Q_1} \neq \emptyset$, then do the following:
   (a) Let $Q_2 = Q - Q_1$ and $T_{Q_2} = \mathcal{N}(Q_2)$.
   (b) If $T_{Q_2} \neq \emptyset$, then let $T_Q$ be the subphylogeny for $Q$ whose root is a node $x_Q$ satisfying $c(x_Q) = c(Sv(Q, Q_1))$ for every $c \in C$ such that $c(Sv(Q, Q_1)) \neq *$, and $c(x_Q) = c(x_{Q_1})$ for every other $c$, where $x_{Q_1}$ is the root of $T_{Q_1}$. Set $\mathcal{N}(Q) = T_Q$ and return.
   (c) Otherwise, let $\{P_i\}_{i=1}^{k}$ be the set of equivalence classes $Q_2/Sv(Q, Q_1)$. If $T_{P_i} = \mathcal{N}(P_i) \neq \emptyset$ for every $i \in \{1, \ldots, k\}$, then let $T_Q$ be the subphylogeny for $Q$ whose root $x_Q$ satisfies $c(x_Q) = c(Sv(Q, Q_1))$ for every $c \in C$ and whose subtrees are $T_{Q_1}$ and $T_{P_1}, \ldots, T_{P_k}$. Set $\mathcal{N}(Q) = T_Q$ and return.

PERFECT-PHYLOGENY($S, C$).

1. Allocate a table $\mathcal{N}$ with one entry for each character subfamily $Q$. Set $\mathcal{N}(Q) = \emptyset$ for each such $Q$.
2. Enumerate, by increasing cardinality, each character subfamily $Q$, and run SUBPHYLOGENY($Q$).
3. If there exists a pair of subfamilies $Q_1, Q_2$ such that $Q_2 = S - Q_1$ and $\mathcal{N}(Q_1), \mathcal{N}(Q_2) \neq \emptyset$, then return the tree $T$ obtained by linking the roots of $T_{Q_1} = \mathcal{N}(Q_1)$ and $T_{Q_2} = \mathcal{N}(Q_2)$ by an edge. Otherwise, return $\emptyset$.

FIG. 2. *The perfect phylogeny algorithm.*

This leads to the algorithm of Figure 2. The main procedure, PERFECT-PHYLOGENY, considers character subfamilies by increasing cardinality; it attempts to build a subphylogeny for each one using procedure SUBPHYLOGENY, inserting the result into a table $\mathcal{N}$.

PERFECT-PHYLOGENY iterates over all $O(2^r m)$ character subfamilies $Q$. For each of these, SUBPHYLOGENY considers $O(2^r m)$ choices of $Q_1$. Kannan and Warnow show how to find the equivalence classes of $Q_2/Sv(Q, Q_1)$ in $O(n)$ time at the expense of precomputing, in $O(2^r nm^2)$ time, the equivalence classes of $S/Sv(G)$ for every subfamily $G$ (see [12]). An $O(2^{2r} nm^2)$ bound follows.

**4. Near-perfect phylogenies.** The algorithm of Figure 2 relies heavily on the fact that perfect phylogenies have no bad edges, a property that may not hold for near-perfect phylogenies. In this section, we show that near-perfect phylogenies can be decomposed into perfect and imperfect parts. The former can be handled by the techniques described in the previous section. We prove that the latter can be generated by examining an amount of information that is polynomial for each fixed $q$. As before, $C_p$ denotes a set of characters required to be convex. Before proceeding, we need some definitions.

DEFINITION 12. *A* penalty state assignment *is a function $\alpha$ that maps each $c \in C - C_p$ to an element $\alpha(c)$ of $\mathcal{A}_c$. The* penalty state assignment *of a species $s$ is the penalty state assignment $\alpha_s$, where $\alpha_s(c) = c(s)$ for each $c \in C - C_p$.*

DEFINITION 13. *Let $Q$ be a character subfamily and $\alpha$ be a penalty state assignment. A* subphylogeny *for $(Q, \alpha)$ is a rooted $C_p$-perfect phylogeny $T$ for $Q$ whose root $x$ satisfies the following properties:*

(i) *For every* $c \in C_p$, $c(x) = c(Sv(Q))$ *if* $c(Sv(Q)) \neq *$; *otherwise,* $c(x) = c(u)$ *for some* $u \in Q$.
(ii) *For every* $c \in C - C_p$, $c(x) = \alpha(c)$.

$T$ *is a* minimum-length subphylogeny *if it has the smallest length among all subphylogenies for* $(Q, \alpha)$.

DEFINITION 14. *Let* $T$ *be a subphylogeny with root* $x$ *for* $(Q, \alpha)$. *Let* $(u, v)$ *be an edge of* $T$, *where* $u$ *is the parent of* $v$. *Then,* $(u, v)$ *is* good *if* $S \cap V(T_v)$ *is a character subfamily; otherwise* $(u, v)$ *is* bad. *The maximal subtree* $T$ *that contains* $x$ *and only bad edges is the* bad tree *of* $T$ *and is denoted* $B(T)$. $T$ *is in* normal form *if, for every good edge* $(u, v)$ *in* $T$ *such that* $u$ *is in* $B(T)$, $T_v$ *is a subphylogeny for some pair* $(Q_v, \alpha_v)$, *where* $Q_v \subseteq Q$.

Note that, by the maximality of $B = B(T)$, if an edge of $T$ is not in $B$ but is adjacent to an edge of $B$, then it is good; i.e., the associated edge subfamily is a character subfamily as well.

DEFINITION 15. *Let* $T$ *be a subphylogeny,* $v$ *be a node of* $B = B(T)$, *and* $u$ *be a child of* $v$ *not in* $B$. *Then the subset of* $S$ *contained in* $T_u$ *is an* edge subfamily at $v$. *An edge subfamily* $Q$ *at* $v$ *is* perfect *if no state is shared between* $Q$ *and* $S - Q$ *on a character* $c$ *except (possibly)* $c(v)$, *and* $Q \cup \{v\}$ *has a perfect phylogeny. All other edge subfamilies at* $Q$ *are* imperfect.

For a node $v$ in $B$, $Pe(v)$ and $Im(v)$ stand for the sets of perfect and imperfect edge subfamilies at $v$, respectively. $\mathcal{P}(v)$ is the union of all perfect edge subfamilies at $v$, and $\mathcal{F}(v)$ is the union of all edge subfamilies at $v$.

Let $T$ be a subphylogeny for $(Q, \alpha)$, and let $v$ be a node in $T$. Then, by definition, the subtree of $T$ consisting of $v$, together with all $T_u$ such that $S \cap V(T_u)$ is a perfect edge subfamily at $v$, is a subphylogeny for $\mathcal{P}(v)$.

We now illustrate some of the notions introduced so far, making reference to Figure 1. The subtree rooted at $y$ in that diagram is a subphylogeny for $(Q, \alpha)$, where $Q = \{s_4, s_5, s_6, s_7, s_8, s_9, s_{10}\}$ and $\alpha(c_4) = 0$; its bad tree consists of edge $(y, v)$. Indeed, if the set $P = \{s_1, \ldots, s_{10}\}$ is a character subfamily within a larger set $S$ such that $P$ and $S - P$ share (say) state $(1, c_3)$, then the whole tree at $x$ is a subphylogeny for $(P, \alpha)$, where $\alpha(c_4) = 1$. The bad tree in this case consists of edges $(s_1, x)$, $(x, y)$, and $(y, v)$. Both of the subphylogenies we have described are in normal form. Observe that $Pe(v) = \{\{s_7, s_8\}, \{s_9, s_{10}\}\}$ and $Im(v) = \{\{s_6\}\}$.

The results that follow characterize the structure of near-perfect phylogenies and subphylogenies.

LEMMA 3. *Let* $T$ *be a* $C_p$*-perfect phylogeny, and let* $(u, v)$ *be a bad edge in* $T$. *Then, for each* $c \in C_p$, $c(u) = c(v)$. *Furthermore, there is some* $c \in C - C_p$ *such that* $c(u) \neq c(v)$.

*Proof.* We first show that for each $c \in C_p$ there exists a shared state on $c$ between $Q_u = S \cap V(T_u)$ and $Q_v = S \cap V(T_v)$. Suppose that this is false. Then $(Q_u, Q_v)$ is a character partition and $(u, v)$ is, by definition, a good edge, a contradiction.

Because of the state shared between $Q_u$ and $Q_v$ on $c \in C_p$ and the fact that $T$ is $C_p$-perfect, we must have $c(u) = c(v)$. We must have $c(u) \neq c(v)$ for some $c \in C - C_p$ because we are dealing with phylogenies where every two nodes differ in at least one state. □

LEMMA 4. *Let* $T$ *be a* $C_p$*-perfect phylogeny such that* penalty$(T) \leq q$. *Then* $T$ *has at most* $qr$ *bad edges.*

*Proof.* Let $C' = C - C_p$. For each $c \in C'$ let $l_c$ be the number of edges $(u, v)$ in $T$ such that $c(u) \neq c(v)$. Since penalty$(T) \leq q$, $|C'| \leq q$. By Lemma 3, for every

bad edge $(u, v)$ there must be some $c \in C'$ such that $c(u) \neq c(v)$. Thus, the number of bad edges is at most $\sum_{c \in C'} l_c$. Moreover, $\sum_{c \in C'}(l_c - (r_c - 1)) \leq q$. Hence, the number of bad edges is bounded by $q + \sum_{c \in C'}(r_c - 1) \leq qr$. ∎

LEMMA 5. *Suppose that $Q$ is a character subfamily having a rooted $C_p$-perfect phylogeny $T$, with root $x$ such that, for every $c \in C_p$, $c(x) = c(Sv(Q))$ if $c(Sv(Q)) \neq *$. Let $\alpha$ be the penalty state assignment of $x$. Then, $(Q, \alpha)$ has a subphylogeny of length at most* length$(T)$.

*Proof.* $T$ is a subphylogeny for $(Q, \alpha)$, except that there might be some $c \in C_p$ such that $c(x) \neq c(u)$ for any $u \in Q$. For each such $c$, carry out the following step until it no longer applies:

> Let $c(x) = \sigma$ be such that $\sigma \neq c(u)$ for any $u \in Q$. Let $A$ be the connected component of $T[\sigma]$ containing $x$, and let $(u, v)$ be any edge of $T$ such that $u \in A$ and $v \notin A$. Then, $c(u) = \beta \neq \sigma$. Set $c(w)$ equal to $\beta$ for all $w$ in $A$.

Each application of the above step preserves perfection with respect to $C_p$; furthermore, it does not affect the contribution of the nonconvex characters to the length. When the step no longer applies, $T$ is a subphylogeny. ∎

LEMMA 6. *Suppose that the pair $(Q, \alpha)$ has a subphylogeny. Then, $(Q, \alpha)$ has a minimum-length subphylogeny in normal form.*

*Proof.* Let $T$ be a minimum-length subphylogeny for $(Q, \alpha)$, and let $B = B(T)$. If $T$ is in normal form, we are done, so suppose it is not. Successively consider each good edge $(u, v)$ in $T$ such that $u$ is in $B$ and $T_v$ is not a subphylogeny for $(Q_v, \alpha_v)$, where $Q_v = S \cap V(T_v)$ and $\alpha_v$ is the penalty state assignment of $v$. For each such edge, apply the following transformation to $T$: Let $T'_v$ be a subphylogeny for $(Q_v, \alpha_v)$ such that length$(T'_v) \leq$ length$(T_v)$; such a tree $T'_v$ exists by Lemma 5, since $T_v$ is a $C_p$-perfect phylogeny for $Q_v$ where, for every $c \in C_p$, $c(v) = c(Sv(Q_v))$ if $c(Sv(Q_v)) \neq *$. Replace $T_v$ by $T'_v$ by deleting $T_v$ and making the root of $T'_v$ a child of $u$.

Each application of the transformation preserves the properties that $T$ is of minimum length and that $T$ is a subphylogeny for $(Q, \alpha)$. After the final application, $T$ is in normal form. ∎

LEMMA 7. *Let $T$ be a subphylogeny for $(Q, \alpha)$, let $B = B(T)$, and let*

$$U = (S - Q) \cup \{u \in P : P \in Im(v), v \in B\}.$$

*Then, for each $v \in V(B)$, $\mathcal{P}(v) = G(v) - U$ for some set $G(v) = \bigcap_{i=1}^{l} Q_i$, where $\{Q_i\}_{i=1}^{l}$ is a set of character subfamilies for different characters $c \in C - C_p$.*

*Proof.* Pick $G(v)$ as follows. For each node $v$ of $B$ and each nonconvex character $c$, let $Q(v, c)$ be the character subfamily consisting of all species $s \in S$ such that $c(s) = c(x)$ for some $x \in \mathcal{P}(v)$. The set $G(v)$ is the intersection of $Q(v, c)$ over all characters $c \in C - C_p$.

To prove the claim, we show containment in both directions:

- Suppose $s \in \mathcal{P}(v)$. Then, $s \in Q(v, c)$ for each $c \in C - C_p$. Hence, $s \in G(v) - U$.
- Suppose $s \in G(v) - U$. By definition of $G(v)$, for each $c \in C - C_p$ there is a species $x \in \mathcal{P}(v)$ such that $c(x) = c(s)$. Also, there must exist a node $u \in B$ such that $s \in \mathcal{F}(u)$. Since $s \notin U$, $c(s) = c(v) = c(u)$ for every $c \in C - C_p$. But then, by Lemma 3 we must have $u = v$. Hence, $s \in \mathcal{P}(v)$. ∎

LEMMA 8. *Let $T$ be a subphylogeny for $(Q, \alpha)$ such that* penalty$(T) \leq q$ *and $B$ be the bad tree of $T$. Then, $|\bigcup_{v \in B} Im(v)| \leq 4q$.*

*Proof.* An edge subfamily $P$ at $v$ is imperfect if either (a) $P$ shares a state $\sigma$ with $S - P$ on character $c$ and $c(v) \neq \sigma$ or (b) $P \cup \{v\}$ does not have a perfect

NEAR-PERFECT-PHYLOGENY$(S, C, q)$.

1. Let $T =$ PERFECT-PHYLOGENY$(S, C)$. If $T \neq$ NIL, then return $T$.
2. If $|S| \leq qr + 1$, then use exhaustive enumeration to search for a minimum-length $q$-near-perfect phylogeny for $(S, C)$. Return NIL if no such phylogeny exists. Otherwise, return any such phylogeny.
3. For each $C_p \subseteq C$ such that $|C_p| \geq m - q$, find a minimum-length $C_p$-perfect phylogeny $T_{C_p}$ of penalty at most $q$ for $S$, if one exists, as follows:
   (a) Allocate a table $\mathcal{N}$ with one entry for each possible pair $(Q, \alpha)$, where $\alpha$ is a penalty state assignment and $Q$ is a subfamily. Initialize $\mathcal{N}(Q, \alpha)$ to NIL for every pair $(Q, \alpha)$.
   (b) Enumerate, by increasing cardinality of $Q$, the pairs $(Q, \alpha)$ such that $\alpha$ is a penalty state assignment and $Q$ is a subfamily. For each $(Q, \alpha)$, attempt (using Lemma 9) to find a minimum-length $C_p$-perfect subphylogeny of penalty at most $q$. If such a subphylogeny $T_Q$ exists, set $\mathcal{N}(Q, \alpha) = T_Q$.
   (c) Let $T_{C_p}$ be the minimum-length tree from among those that can be obtained by putting an edge between the roots of subphylogenies for $(Q_1, \alpha_1)$ and $(Q_2, \alpha_2)$ such that $Q_2 = S - Q_1$ and $\mathcal{N}(Q_1, \alpha_1), \mathcal{N}(Q_2, \alpha_2) \neq$ NIL.
4. Return the tree $T_{C_p}$ that minimizes length$(T_{C_p})$ over all sets $C_p$ enumerated in the previous step. If no tree exists, return NIL.

FIG. 3. *The near-perfect phylogeny algorithm.*

phylogeny. The number of subfamilies of the latter sort is at most $q$, since each of them contributes at least 1 to the total penalty. Let $K$ be the set of subfamilies $P$ that satisfy (a). Let $K_0$ be the subset of $K$ consisting of all subfamilies $P$ such that there is a character $c$ and species $s \in P$ satisfying the requirements that $P$ is a subfamily at $v \in V(B)$, $c(v) \neq c(s)$, and $c(s) = c(s')$ for some $s' \in S - Q$. Since each $P \in K_0$ contributes at least 1 to the total penalty, $|K_0| \leq q$. Let $J$ be the graph whose vertex set is $K - K_0$ and whose edge set is defined as follows. Let $Q_u \in K - K_0$ and $Q_v \in K - K_0$ be imperfect subfamilies at $u$ and $v$, respectively. There is an edge between $Q_u$ and $Q_v$ in $J$ if and only if there are a character $c$ and species $s_u \in Q_u$ and $s_v \in Q_v$ such that $c(u) \neq c(s_u) = c(s_v) \neq c(v)$. Let $\mu$ be the size of a maximum matching in $J$. One can verify that

$$q \geq \mu + |K - K_0| - 2\mu \geq |K - K_0|/2.$$

Therefore, $|K - K_0| \leq 2q$, and the lemma follows. □

**5. The algorithm.** Our near-perfect phylogeny algorithm is shown in Figure 3. Its analysis relies on the result below, proved in the next subsection.

LEMMA 9. *A minimum-length subphylogeny for a pair $(Q, \alpha)$ can be found in $|Q|m^{O(q)}2^{O(q^2r^2)}$ time and $O(q(r + \log m))$ space.*

We now have the main result of this paper.

THEOREM 10. *The algorithm* NEAR-PERFECT-PHYLOGENY *runs in time $|S|m^{O(q)}2^{O(q^2r^2)}$. That is, for fixed $q$ and $r$, the problem of determining whether $S$ has a $q$-near-perfect phylogeny, and, if so, finding such a tree of minimum length, can be solved in polynomial time.*

*Proof.* Step 1 takes $O(2^{2r}nm^2)$ time, as explained in section 3. By Theorem 1, if

$S$ has a perfect phylogeny, this tree must also be an optimum near-perfect phylogeny. It can be shown that step 2 can be completed within the claimed time bound.

We now argue that step 3 of NEAR-PERFECT-PHYLOGENY finds an optimal phylogeny for each choice of $C_p$. Assume that step 3(b) correctly computes a minimum-length subphylogeny for each pair $(Q, \alpha)$ it considers (or determines that no such tree exists). Let $T$ be any minimum-length $C_p$-perfect phylogeny for $S$. It suffices to prove that in step 3(c) the algorithm encounters a $C_p$-perfect phylogeny $T'$ for $S$ such that $\text{length}(T') = \text{length}(T)$.

By Lemma 4 and the fact that $|S| > qr + 1$, $T$ must have at least one good edge $e = (u_1, u_2)$. For $i = 1, 2$, let $\alpha_i$ be the penalty state assignment of $u_i$, and let $Q_i = S \cap V(T_{u_i})$. Then, for $i = 1, 2$, $Q_i$ is a character subfamily and, by Lemma 5, $(Q_i, \alpha_i)$ has a subphylogeny $T_i'$ of length at most $\text{length}(T_i)$. Without loss of generality, assume that this $T_i'$ is generated in step 3(b). Then, step 3(c) generates a tree $T'$ by putting an edge between the roots of $T_1$ and $T_2$. By the minimality of $T$, $\text{length}(T') = \text{length}(T)$, as claimed.

Note that NEAR-PERFECT-PHYLOGENY enumerates only sets $C_p$ of size at least $m - q$, because a $q$-near-perfect phylogeny has at most $q$ nonconvex characters. Thus, the result returned by step 4 is a minimum-length $q$-near-perfect phylogeny for $S$, if one exists.

The total number of sets $C_p$ considered in step 3 is

$$\sum_{i=m-q}^{m} \binom{m}{i} = O(qm^q),$$

and step 3(b) enumerates $O(m2^r r^q)$ $(Q, \alpha)$ pairs. By Lemma 9, this leads to a total running time of $|S| m^{O(q)} 2^{O(q^2 r^2)}$. □

**5.1. Computing a subphylogeny.** We now prove Lemma 9 by giving an algorithm to find a minimum-length $C_p$-perfect subphylogeny $T$ for $(Q, \alpha)$. The key idea is given by Lemma 6, which suggests that, to find a $C_p$-perfect subphylogeny $T$ of minimum penalty, it suffices to guess $B = B(T)$ and, for each node $v$ of $B$, the perfect and imperfect edge subfamilies at $v$. Our procedure enumerates a sequence of *candidates*, each of which is used to generate a potential tree. A candidate consists of four pieces of information:

- $\tilde{B}$, a guess as to the bad tree of $T$.
- For each node $v$ of $\tilde{B}$, a penalty state assignment $\alpha_v$ such that $\alpha_v = \alpha$ if $v$ is the root of $\tilde{B}$.
- $\tilde{\mathcal{P}}$, a mapping from each vertex $v$ of $\tilde{B}$ to a subset of $S$ representing a guess as to the union of perfect edge subfamilies at $v$.
- $\tilde{Im}$, a mapping from each vertex $v$ of $\tilde{B}$ to a collection of subsets of $S$ representing a guess as to the collection of imperfect edge subfamilies at $v$.

Assume that the candidate is a correct guess as to the various components of a subphylogeny for $(Q, \alpha)$. We now describe how to construct such a subphylogeny from this information.

We first find, for each $v \in \tilde{B}$, the decomposition $\tilde{Pe}(v)$ of $\tilde{\mathcal{P}}(v)$ into perfect edge subfamilies. As in algorithm SUBPHYLOGENY (Figure 2), we rely on Lemma 2, which states that if we know one of the subfamilies $R$ such that $R \subseteq \tilde{\mathcal{P}}(v)$, we have one of two possibilities:

(i) $\tilde{Pe}(v) = \{R, \tilde{\mathcal{P}}(v) - R\}$ or
(ii) $\tilde{Pe}(v) = R \cup (\tilde{\mathcal{P}}(v) - R)/v$, where $c(v) = c(Sv(\tilde{\mathcal{P}}(v), R))$ for every character $c \in C$.

In the latter case, $c(Sv(\tilde{\mathcal{P}}(v), R)) \neq *$ for every character $c$. There are polynomially many (for fixed $r$) choices for $R$; one of these must enable us to make the appropriate decomposition of $\mathcal{P}(v)$ if the candidate is a correct guess.

The distribution of perfect and imperfect edge subfamilies across the vertices of $\tilde{B}$ forces the states of some its nodes to assume certain values in order to maintain the convexity of the corresponding characters. For a vertex $v$ in $\tilde{B}$, let $Q_v$ be the set of all species in the subtree of $T$ rooted at $v$. The state of vertex $v$ in $\tilde{B}$ on character $c \in C_p$ is forced to equal $\sigma$ if either $Q_v$ and $S - Q_v$ share a state on character $c$ or there are distinct subtrees $T_1$, $T_2$ at $v$, where $T_1$, $T_2$ contain species $x_1$, $x_2$, respectively, such that $c(x_1) = c(x_2)$. The remaining unforced states are set in any way that is consistent with convexity of the characters in $C_p$. This can be done in time polynomial in $n$, $m$, and $r$.

We now produce a subphylogeny for $(Q, \alpha)$ by doing the following for each $v \in \tilde{B}$:

(a) For each $R \in \tilde{Im}(v)$, enumerate all penalty state assignments $\gamma$ to find the pair $(R, \gamma)$ such that $T_R = \mathcal{N}(R, \gamma) \neq \emptyset$ and $\text{length}(T_R) + \text{dist}(u, v)$ is minimum, where $u$ is the root of $T_R$. Connect the root of $T_R$ to $v$.
(b) For every $R \in \tilde{Pe}(v)$, enumerate all penalty state assignments $\gamma$ to find a pair $(R, \gamma)$ such that $T_R = \mathcal{N}(R, \gamma) \neq \emptyset$ and the tree obtained by linking $v$ to the root of $T_R$ is a perfect phylogeny for $R \cup \{v\}$. Connect the root of $T_R$ to $v$.

There is, of course, no guarantee that a candidate is a correct guess from which a minimum-length subphylogeny can be constructed. Indeed, it is possible that a candidate simply cannot be used to produce a subphylogeny for $(Q, \alpha)$. For instance, a candidate is invalid if some $s \in Q$ is neither a vertex in $\tilde{B}$ nor contained in some set in either $\tilde{Im}(v)$ or $\tilde{Pe}(v)$ for some $v \in V(\tilde{B})$. A candidate is also invalid if it is impossible to make a state assignment for $\tilde{B}$ on the characters in $C_p$ in any consistent way. Finally, in either of steps (a) or (b) above, it may be impossible to find the required subtrees of a node $v \in \tilde{B}$. In any case, if an invalid candidate is encountered, we dismiss it. If no valid candidate can be generated for a $(Q, \alpha)$ pair, we set $\mathcal{N}(Q, \alpha) = \emptyset$.

It is also possible that a candidate allows us to generate a subphylogeny but not a minimum-length one. This issue is resolved by enumerating *all* potential candidates. The tree $T$ stored in $\mathcal{N}(Q, \alpha)$ is the one that minimizes the length among all trees generated from valid candidates.

**5.2. Generating candidates.** We now describe how candidates are generated and derive a bound on their number. First, observe that, by Lemma 4, we need to consider only trees $\tilde{B}$ with at most $qr$ edges. Thus, $qr^{O(qr)}$ distinct tree topologies $\tilde{B}$ are generated. Enumerating them takes time $qr^{O(qr)}$ and space $O(qr \log qr)$. The number of penalty state assignments enumerated for the nodes in $\tilde{B}$ is $r^{O(q^2 r)}$. These can be generated in time $2^{O(q^2 r^2)}$ and space $O(q^2 r^2)$.

Suppose that $\tilde{B}$ is indeed the bad tree of a subphylogeny $T$ for $(Q, \alpha_Q)$. By Lemma 8, there is some set of at most $4q$ character subfamilies containing all imperfect edge subfamilies at $v$; each of these is a potential choice for $\tilde{Im}(v)$. There are $m^{O(q)} 2^{O(qr)}$ choices of subfamilies and $(qr)^{O(q)}$ ways in which these subfamilies can be distributed among the vertices of $\tilde{B}$.

By Lemma 7, for every $v \in \tilde{B}$ we can restrict our attention to $\tilde{\mathcal{P}}(v)$ of the form

$\tilde{\mathcal{P}}(v) = G(v) - U$, where

$$U = (S - Q) \cup \{u \in P \colon P \in \tilde{Im}(v), v \in \tilde{B}\} \qquad \text{and} \qquad G(v) = \bigcap_{i=1}^{k} Q_i$$

such that $\{Q_i\}_{i=1}^{k}$ is a set of character subfamilies for different characters $c \in C - C_p$. For every node $v$ in $\tilde{B}$, $G(v)$ is the intersection induced by $\tilde{Im}(v)$.

Thus, the total number of candidates is $m^{O(q)}2^{O(q^2r^2)}$. These can be generated in time $m^{O(q)}2^{O(q^2r^2)}$ and space $O(q(\log m + r))$.

It can be verified that processing a candidate takes time $O(|Q|r^q)$. The total time to find a minimum-penalty subphylogeny for $(Q, \alpha)$ is therefore $|Q|m^{O(q)}2^{O(q^2r^2)}$, and the space used is $O(q(r + \log m))$. This concludes the proof of Lemma 9.

**6. Conclusions and open questions.** We have shown that a relaxed version of the perfect phylogeny problem, parameterized by the degree $q$ to which the resulting phylogeny deviates from perfection, can be solved in polynomial time. Since the perfect phylogeny model is too restrictive, our algorithm may have some practical use. Unfortunately, its practicality is limited by its running time, which is bounded by a polynomial whose degree depends on $q$. We note, however, that the time bound is based on the perhaps overly pessimistic assumption that all bad edges can occur together in a bad tree. One may ask whether this is likely to happen in practice. Also, is there a parameter that is smaller than $q$ in practice, in terms of which to express the time bound? One candidate is the maximum size of a bad tree.

Perhaps the most important open question raised by our algorithm is whether it is possible to make the degree of the polynomial describing the running time independent of the parameters; that is, whether there is an algorithm with running time $O(f(q)p(|S|, m))$, where $p$ is a polynomial whose degree does not depend on $q$. Alternatively, one could try to show that the case where $r$ is fixed and $q$ is the only parameter is hard for $W[1]$.

**Acknowledgments.** We thank the referee and editor for their suggestions, which substantially improved the presentation.

REFERENCES

[1] R. Agarwala and D. Fernández-Baca, *A polynomial-time algorithm for the perfect phylogeny problem when the number of character states is fixed*, SIAM J. Comput., 23 (1994), pp. 1216–1224.

[2] H. Bodlaender, M. Fellows, and T. Warnow, *Two strikes against perfect phylogeny*, in Proceedings of the 19th International Colloquium on Automata, Languages, and Programming, Lecture Notes in Comput. Sci., Springer-Verlag, 1992, pp. 273–283.

[3] W. H. E. Day, D. S. Johnson, and D. Sankoff, *The computational complexity of inferring rooted phylogenies by parsimony*, Math. Biosci., 81 (1986), pp. 33–42.

[4] G. F. Estabrook, *Cladistic methodology: A discussion of the theoretical basis for the induction of evolutionary history*, Annu. Rev. Ecology and Systematics, 3 (1972), pp. 427–456.

[5] D. Fernández-Baca and J. Lagergren, *A polynomial-time algorithm for near-perfect phylogeny*, in Proceedings of the 23rd International Conference on Automata, Languages, and Programming, Lecture Notes in Comput. Sci., Springer-Verlag, 1996, pp. 670–680.

[6] W. M. Fitch, *Aspects of molecular evolution*, Annu. Rev. Genet., 7 (1973), pp. 343–380.

[7] L. R. Foulds and R. L. Graham, *The Steiner problem in phylogeny is NP-complete*, Adv. Appl. Math., 3 (1982), pp. 43–49.

[8] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, 1979.

[9] D. Gusfield, *The Steiner Tree Problem in Phylogeny*, Technical report 334, Computer Science Department, Yale University, New Haven, CT, 1984.
[10] J. Håstad, *Clique is hard to approximate within* $n^{1-\epsilon}$, Acta Math., 182 (1999), pp. 105–142.
[11] S. Hougardy and H. J. Prömel, *A* 1.598 *approximation algorithm for the Steiner problem in graphs*, in Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms, Baltimore, MD, SIAM, Philadelphia, 1999, pp. 448–453.
[12] S. Kannan and T. Warnow, *A fast algorithm for the computation and enumeration of perfect phylogenies*, SIAM J. Comput., 26 (1997), pp. 1749–1763.
[13] F. R. McMorris, T. J. Warnow, and T. Wimer, *Triangulating vertex-colored graphs*, SIAM J. Discrete Math., 7 (1994), pp. 296–306.
[14] M. A. Steel, *The complexity of reconstructing trees from qualitative characters and subtrees*, J. Classification, 9 (1992), pp. 91–116.
[15] T. Warnow, D. Ringe, and A. Taylor, *Reconstructing the evolutionary history of natural languages*, in Proceedings of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms, Atlanta, GA, SIAM, Philadelphia, 1996, pp. 314–322.