SIAM J. COMPUT.
Vol. 30, No. 6, pp. 1942–1961

A POLYNOMIAL TIME APPROXIMATION SCHEME FOR INFERRING EVOLUTIONARY TREES FROM QUARTET TOPOLOGIES AND ITS APPLICATION*

TAO JIANG†, PAUL KEARNEY‡, AND MING LI‡

Abstract. Inferring evolutionary trees has long been a challenging problem for both biologists and computer scientists. In recent years research has concentrated on the *quartet method* paradigm for inferring evolutionary trees. Quartet methods proceed by first inferring the evolutionary history for every set of four species (resulting in a set Q of inferred quartet topologies) and then recombining these inferred quartet topologies to form an evolutionary tree. This paper presents two results on the quartet method paradigm. The first is a polynomial time approximation scheme (PTAS) for recombining the inferred quartet topologies optimally. This is an important result since, to date, there have been no polynomial time algorithms with performance guarantees for quartet methods. To achieve this result the natural *denseness* of the set Q is exploited. The second result is a new technique, called *quartet cleaning*, that detects and corrects errors in the set Q with performance guarantees. This result has particular significance since quartet methods are usually very sensitive to errors in the data. It is shown how quartet cleaning can dramatically increase the accuracy of quartet methods.

Key words. dense instance, evolutionary tree, approximation algorithm, quartet method, smooth polynomial

AMS subject classifications. 68Q25, 92B99

PII. S0097539799361683

1. Introduction.

A fundamental problem in computational biology is inferring an evolutionary tree from biological data. Virtually all formulations of this problem (maximum likelihood, maximum parsimony, minimum distance, etc. [10, 15]) are NP-hard, and so, methods tend to be either heuristic (and seldom with performance guarantees) or prohibitively exhaustive. This is a real conundrum for evolutionary biologists as datasets can be very large forcing them to use heuristic methods that can lead to erroneous results.[1] The difficulties of inferring evolutionary trees that hamper biologists have also catalyzed a surge of algorithmic research in the computer science community. Despite this attention, efficient algorithms with performance guarantees have been slow in coming.

In recent years the computational biology community has focused on the *quartet method* paradigm for inferring evolutionary trees [3, 5, 8, 11, 14]. Quartet methods utilize topological information on sets of four labels[2] to infer an evolutionary tree. To illustrate, consider the four possible trees labeled by $\{a, b, c, d\}$ as depicted in Figure

*Received by the editors September 27, 1999; accepted for publication (in revised form) October 12, 2000; published electronically March 20, 2001. A preliminary version of this paper appeared in *Proceedings of the 39th Annual Symposium on Foundations of Computer Science*, IEEE, 1998.
http://www.siam.org/journals/sicomp/30-6/36168.html

†Department of Computer Science, University of California, Riverside, CA 92521 (jiang@cs.ucr.edu). This author was supported in part by NSERC research grant OGP0046613, a CITO grant, and a UCR startup grant. He is on leave from McMaster University.

‡Department of Computer Science, University of Waterloo, Waterloo, ON, N2L 3G1, Canada (pkearney@math.uwaterloo.ca, mli@math.uwaterloo.ca). The work of the second author was supported in part by a CITO grant and by NSERC research grant 160321. The work of the third author was supported in part by NSERC research grant OGP0046506, CITO, a CGAT grant, and the Steacie Fellowship.

[1] As exemplified by the *Out of Africa* fiasco [16, 17].

[2] A label may represent a species or, more generally, a DNA or protein sequence.

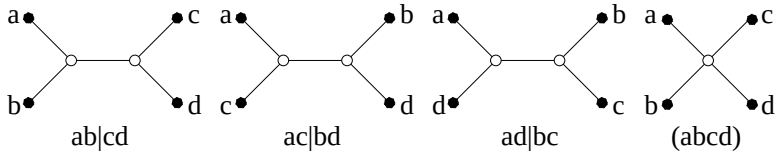

FIG. 1.1. *The four possible trees labeled by* $\{a, b, c, d\}$.

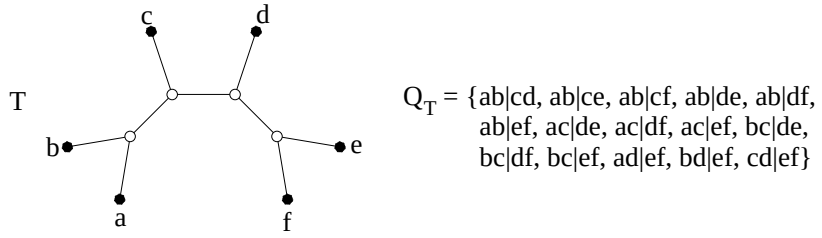

FIG. 1.2. *A tree* T *labeled by* $\{a, b, c, d, e, f\}$ *and* Q_T.

FIG. 1.3. *The quartet method paradigm.*

1.1. These labeled trees are denoted $ab|cd$, $ac|bd$, $ad|bc$, and $(abcd)$, respectively. The topology induced by the quartet $\{a, b, c, d\} \subseteq S$ in a tree T labeled by S is $ab|cd$ if a, b and c, d reside in disjoint subtrees of T. Q_T is defined to be the set of all topologies induced in T by quartets taken from S. For example, Figure 1.2 depicts a labeled degree-3 (i.e., fully resolved) tree T and its set of induced quartet topologies Q_T. We note that an evolutionary tree is usually represented by a labeled degree-3 tree with weighted edges. However, often the edge weights are determined after the shape of the tree is determined, as is the case for quartet methods. For the duration of the paper let evolutionary tree be synonymous with degree-3 labeled tree. Note that the set Q_T does not contain topologies of the form $(abcd)$ if T is a degree-3 tree.

Quartet methods are motivated by the following theorem that reveals the strong relationship between Q_T and T [7].

THEOREM 1.1. *Let* T *be an evolutionary tree.* Q_T *is unique to* T*, and furthermore,* T *can be recovered from* Q_T *in polynomial time.*

In other words, induced quartet topology provides much information about an evolutionary tree. This motivates the quartet method paradigm which is the following two step approach to inferring evolutionary trees (see Figure 1.3). Let T denote the evolutionary tree being estimated whose leaves are labeled by the elements of S:

1. A *quartet topology inference method* is used to infer the topology of each quartet in S from the data (typically DNA or protein sequence data). This results in a set Q of inferred quartet topologies.
2. A *quartet recombination method* is used to recombine the quartet topologies in Q to form an estimate T' of T.

There are several quartet topology inference methods including neighbor joining [12], the ordinal quartet method [11], maximum likelihood [9], and maximum

parsimony [15]. Quartet recombination methods attempt to solve variations of the following optimization problem.

MAXIMUM QUARTET CONSISTENCY (MQC).
Instance: A set Q of quartet topologies over label set S.
Goal: Find an evolutionary tree T labeled by S that maximizes $|Q_T \cap Q|$.

At this point we make an important distinction between two versions of MQC: *complete* MQC and *incomplete* MQC. A set Q of quartet topologies is *complete* if Q contains a quartet topology for each quartet over label set S. Incomplete MQC permits the input Q to be incomplete whereas the input to complete MQC is complete. Incomplete MQC is NP-hard [13]. In section 4 a proof of the NP-completeness of complete MQC is presented. Due to these results, most quartet recombination methods are heuristic or solve weaker optimization requirements. Examples are the Q^* method [5], the short quartet method [8], a semidefinite programming method [3], and quartet puzzling [14]. None of these methods give a performance guarantee. Despite the popularity of quartet methods, the absence of performance guarantees has been a legitimate criticism of the quartet method approach.

The distinction between complete and incomplete MQC is important for two reasons. First, in practice one almost always can obtain complete Q. Second, in this paper we present a polynomial time approximation scheme (PTAS) for complete MQC. In fact, this is the first PTAS for inferring an evolutionary tree under the quartet method paradigm and thus is a significant advancement in the development of algorithms for inferring evolutionary trees. A PTAS is desirable since it allows the approximation of an optimal solution with arbitrary precision (at the cost of efficiency). In contrast, Steel's proof that incomplete MQC is NP-hard can be adapted to show that incomplete MQC is MAX-SNP–hard; hence there is no PTAS for incomplete MQC unless NP=P, by the results of [2]. It should be pointed out that one may wish to weight quartets in practice. However, the weighted version of MQC is MAX-SNP–hard, as it generalizes the incomplete MQC, and thus has no PTAS.

Instances of complete MQC are *dense* relative to instances of incomplete MQC. Recently, the examination of dense versions of such MAX-SNP problems as Max–Cut, Betweenness, and Max–k–Sat has yielded PTASs for these problems [1, 2]. Dense instances of problems such as Max–Cut are graphs with $\Omega(n^2)$ edges whereas dense instances of Max–k–Sat are boolean k–Sat formulae with $\Omega(n^k)$ clauses. MQC is an example of an applied problem that motivates the examination of dense problems. How the natural denseness of an instance of MQC can be exploited to obtain a PTAS is explored further in section 2. In section 5 it is shown that the MQC PTAS can be extended to an important weighted variation of the problem and that this weighted version of the PTAS can be utilized to solve a consensus problem. For the duration of the paper MQC can be assumed to mean complete MQC.

Our second result is a new technique, called *quartet cleaning*, that can detect and correct quartet errors in the set Q of inferred quartet topologies. The quartet topology $ab|cd \in Q$ is called a quartet error if $ab|cd \notin Q_T$, where T is the evolutionary tree being estimated.

The practical motivation for quartet cleaning is that the accuracy of most quartet recombination methods depends critically upon the quality of the set Q. To illustrate, consider the sensitivity of the Q^* method and the short quartet method to quartet errors in Q. In particular, if e is an edge of T then $\{a, b, c, d\}$ is a quartet *across* e if a and b are in a separate component of $T - \{e\}$ than c and d (see Figure 1.4(i)). Both the Q^* method and the short quartet method have the property that a single

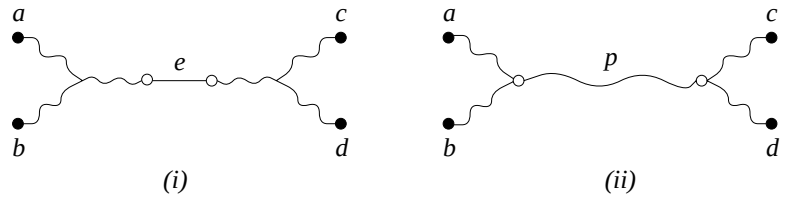

FIG. 1.4. (i) *ab|cd is a quartet across edge e.* (ii) *ab|cd is a quartet across path p.*

quartet error in Q involving a quartet across e can result in the edge e of T not being recovered. (These methods return a contraction of T at e.)

If there are m quartets across e in T and Q contains no more than $\alpha\sqrt{m}$ quartet errors involving quartets across e, where α is a constant, then quartet cleaning applied to Q returns a set Q' of quartet topologies where all of these quartet errors across e have been corrected. This results in a dramatic decrease in the sensitivity of quartet recombination methods such as Q^* method and the short quartet method to quartet errors; they can now tolerate up to $\alpha\sqrt{m}$ quartet errors across edge e, where before they could tolerate none. The bound $\alpha\sqrt{m}$ is shown to be asymptotically optimal in that no algorithm can correct more than $\alpha\sqrt{m}$ quartet errors across an edge.

A polynomial algorithm for quartet cleaning is presented in section 3. It makes nontrivial use of the PTAS for the MQC problem described above and thus serves as another motivation for this PTAS. This suggests that the ideas and techniques developed here are powerful and may find wider use for inferring evolutionary trees.

2. A PTAS for MQC. Let Q be an instance of MQC with label set S. Our discussion begins with the observation that $|Q_{T_{OPT}} \cap Q| \geq \binom{n}{4}/3$ where T_{OPT} is an optimal solution [3, 4]. This follows since a randomly selected tree has a 1/3 chance of inducing $ab|cd \in Q$, for each quartet $\{a, b, c, d\}$. Hence, for some constant c, $|Q_{T_{OPT}} \cap Q| \geq cn^4$. Our goal is then to find an approximation algorithm such that

$$|Q_{T_{APX}} \cap Q| \geq |Q_{T_{OPT}} \cap Q| - \epsilon n^4,$$

where T_{APX} is the result of the approximation algorithm.

The approximation algorithm is founded upon two concepts: a k–bin decomposition of T_{OPT} and smooth integer polynomial programs.

DEFINITION 2.1. *T_k is a k-bin decomposition of T_{OPT} if there is a partition of S into* bins *S_1, S_2, ..., S_k such that the following hold.*

- *For each S_i, $|S_i| \leq 6n/k$. Furthermore, there is a vertex v_i of degree $|S_i| + 1$, called the* bin root*, that is adjacent to each vertex in S_i.*
- *For all quartets $\{a, b, c, d\}$ where a, b, c, and d are in different bins of T_k, $ab|cd \in Q_{T_{OPT}}$ if and only if $ab|cd \in Q_{T_k}$.*

An example of a k-bin decomposition appears in Figure 2.1.

Section 2.1 will discuss k-bin decompositions in detail. In particular, it is shown that there is a k-bin decomposition T_k of T_{OPT} that is a good approximation of T_{OPT}, i.e., $|Q_{T_k} \cap Q| \geq |Q_{T_{OPT}} \cap Q| - (c'/k)n^4$, for some constant c'. Our approach is to approximate T_{OPT} indirectly by approximating T_k.

Consider a fixed k. Let K be T_k with all leaves removed (and thus the leaves of K are the bin roots of T_k). K is called the *kernel* of T_k and T_k is called a *completion* of K. K is completed to T_k by providing a label-to-bin assignment.

If the kernel K of T_k is known, then, to approximate T_k, it suffices to determine an approximately optimal label-to-bin assignment for K. This problem is formalized as follows.

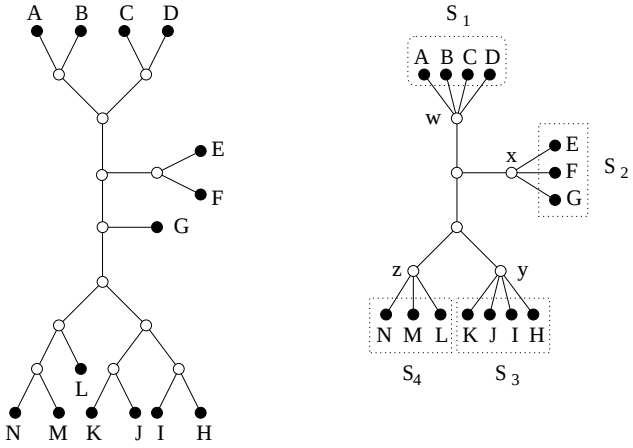

FIG. 2.1. *A tree (left) and a 4-bin decomposition of the tree (right) with bin roots w, x, y, and z.*

LABEL-TO-BIN ASSIGNMENT (LBA).
Instance: Set Q of quartet topologies and (degree-3) kernel K with k leaves.
Goal: Find a completion T' of K that maximizes $|Q_{T'} \cap Q|$.

It follows from the result in [6] that LBA is NP-hard. In section 2.2, LBA is formulated as a smooth integer polynomial problem and a PTAS for LBA is presented. In particular, it is shown that, for any $\epsilon > 0$, $|Q_{T'} \cap Q| \geq |Q_{\hat{T}} \cap Q| - \epsilon n^4$, where Q and K denote the instance of LBA, T' is the completion of K produced by our PTAS and $\hat{T}$ is an optimal completion of K.

The approximation algorithm solves (approximately) an instance of LBA for every tree with k leaves. Since k is a constant, this can be done in polynomial time using the PTAS for LBA. Let T_{APX} be the completed tree obtained that maximizes $|Q_{T_{APX}} \cap Q|$. Since the kernel K of T_k is one of the trees completed, it follows that $|Q_{T_{APX}} \cap Q| \geq |Q_{T'} \cap Q|$ where T' is the completion of K.

2.1. Decomposing the optimal tree. First we demonstrate that there is a k-bin decompositions that is a good approximation of the optimal evolutionary tree.

THEOREM 2.2. *There is a k-bin decomposition T_k of T_{OPT} such that $|Q_{T_k} \cap Q| \geq |Q_{T_{OPT}} \cap Q| - (c'/k)n^4$ for some constant c'.*

Theorem 2.2 is proven algorithmically. For convenience, let $T = T_{OPT}$ and assume that T is degree-3, without loss of generality. Let T_k denote the result of the following algorithm.

ALGORITHM k-BIN DECOMPOSITION(T).

1. Root T at an arbitrary internal vertex. Let $T(v)$ denote the subtree of T rooted at v.
2. Traverse T, beginning at the root, such that for each vertex v visited:
   If $|T(v)| \leq 6n/k$ then
   - contract all internal edges of $T(v)$,
   - label v as a bin root if v is not a leaf, and
   - continue traversal at v's parent.

   Otherwise, continue traversal at an unvisited child of v.
3. For each bin root v with parent v' and sibling u':
   If $|T(v)| \leq 3n/k$ and u' has a child u with $|T(u)| \leq 3n/k$ then
   - contract $\{u', v'\}$,

- transfer the leaves in $T(u)$ to the bin of v, and delete u.

4. For each leaf u of T not assigned to a bin, bisect the edge between u and its parent with a new vertex v, and mark v as a bin root.

Note that the last step of the algorithm is necessary since a leaf cannot be a bin root. Call a bin of T_k *small* if it has size less than $3n/k$. A bin root is *small* if its bin is small. Call a bin of T_k *large* if it has size between $3n/k$ and $6n/k$, inclusive. A bin root is *large* if its bin is large. Let s denote the number of small bins in T_k and l the number of large bins in T_k.

LEMMA 2.3. $s < 2l$.

Proof. We prove the following by induction: For every h, if u is a vertex of height h and is not a bin root then the lemma holds for the subtree $T(u)$.

For the base case, assume that u has children p and q. It cannot be that both p and q are small otherwise $|T(u)| < 6n/k$ and the algorithm would not have visited p and q. Hence, the lemma is true for $T(u)$.

In general, assume that u has children p and q. If both p and q are bin roots then the argument for the base case applies. If neither p nor q are bin roots then the inductive hypothesis applies to $T(p)$ and $T(q)$, and hence the lemma holds for $T(u)$. Otherwise, suppose that p is not a bin root but q is.

If q is large then the induction can be applied to $T(p)$ and the claim follows. Otherwise, suppose that q is small. Let p_1 and p_2 be the children of p. Neither p_1 nor p_2 are small, since otherwise the algorithm would have merged one of $T(p_1)$ and $T(p_2)$ with $T(q)$ in step 3. By the inductive hypothesis, $s_1 < 2l_1$ and $s_2 < 2l_2$ where s_1, l_1, s_2, and l_2 are the numbers of small and large bins in $T(p_1)$ and $T(p_1)$, respectively. It follows that $s_1 + s_2 + 1 < 2(l_1 + l_2)$, and hence the claim holds for $T(u)$. ∎

Since each large bin of T_k has size at least $3n/k$, $l \leq k/3$. T_k has $l + s$ bins, and so, $l + s < l + 2l = 3l \leq k$. We conclude that T_k has less than k bins each with size bounded by $6n/k$. Furthermore, since T_k was obtained from T by contracting edges and transferring leaves to neighboring bins, it follows that, for all quartets $\{a, b, c, d\}$ where a, b, c, and d are in different bins of T_k, $ab|cd \in Q_T$ if and only if $ab|cd \in Q_{T_k}$. The following lemma completes the proof of Theorem 2.2.

LEMMA 2.4. $|Q_{T_k} \cap Q| \geq |Q_{T_{OPT}} \cap Q| - (c'/k)n^4$ *for some constant* c'.

Proof. Observe that $|Q_{T_k} \cap Q| \geq |Q_{T_{OPT}} \cap Q| - |Q'|$ where Q' is the set of all quartets with an induced topology in T_{OPT} that is not induced in T_k. Let $\{a, b, c, d\} \in Q'$. It follows that at least two of these labels are in the same bin of T_k.

There are at most $k(6n/k)^2 n^2 = 36n^4/k$ quartets with two labels in the same bin. Similarly, there are at most $36n^4/k^2$ and $36n^4/k^3$ quartets with three and four labels in the same bin, respectively. It follows that

$$\begin{aligned} |Q_{T_k} \cap Q| &\geq |Q_{T_{OPT}} \cap Q| - 3(36n^4/k) \\ &\geq |Q_{T_{OPT}} \cap Q| - (108/k)n^4. \end{aligned}$$ ∎

2.2. A PTAS for LBA. Let Q and K be an instance of the LBA problem where K has k leaves. $\hat{T}$ is an optimal completion of K if $\hat{T}$ maximizes $|Q_{\hat{T}} \cap Q|$ over all completions of K. To formalize this optimization problem *smooth polynomials* are used. A degree d polynomial $p(x)$ is *t-smooth*, where t is a constant, if the coefficient of each degree i term is in the interval $[-tn^{d-i}, tn^{d-i}]$ for $0 \leq i \leq d$.

Define the 0-1 label-to-bin assignment $x = (x_{sb})$ as follows: $x_{sb} = 1$ if label s is assigned to bin b; otherwise x_{sb} is 0. For each quartet $\{a, b, c, d\} \in Q$, create a degree

4 polynomial

$$p_{ab|cd}(x) = \sum_{ij|kl \in Q_K} x_{ai} x_{bj} x_{ck} x_{dl},$$

and define

$$p(x) = \sum_{ab|cd \in Q} p_{ab|cd}(x).$$

Observe that $p(x)$ is 1-smooth. To ensure that each label is assigned to exactly one bin, the following constraints are added: for each label s, $\sum_{b=1}^{k} x_{sb} = 1$. We also require that none of the bins are too large: for each bin b, $\sum_{s=1}^{n} x_{sb} \leq 6n/k$.

Clearly, if x satisfies these constraints then $p_{ab|cd}(x) = 1$ if $ab|cd \in Q_{T'}$, and $p_{ab|cd}(x) = 0$ if $ab|cd \notin Q_{T'}$, where T' is the completion of K specified by the label-to-bin assignment x. Hence, our optimization problem is to find a 0-1 label-to-bin assignment $x = (x_{sb})$ so that

$$\begin{aligned} &p(x) \text{ is maximized,} \\ &\sum_{b=1}^{k} x_{sb} = 1 \text{ for each label } s, \\ &\sum_{s=1}^{n} x_{sb} \leq 6n/k \text{ for each bin } b. \end{aligned}$$

Arora, Frieze, and Kaplan [1] present a PTAS for solving t-smooth integer polynomial programs.

THEOREM 2.5. *For each $\epsilon > 0$ there is a polynomial time algorithm that produces a 0-1 assignment x such that $p(x) \geq m - \epsilon n^d$ where $p(x)$ is a t-smooth polynomial of degree d with maximum value m.*

The PTAS of Theorem 2.5 first solves the fractional version of the problem to obtain a solution $(\hat{x}_{sb})$. Randomized rounding is then used to obtain a 0-1 solution (x_{sb}). However, the rounding procedure used rounds each $\hat{x}_{sb}$ individually. This does not quite work here because of the additional constraints $\sum_{b=1}^{k} x_{sb} = 1$ for each label s. Hence, the following randomized rounding procedure is used instead: with probability $\hat{x}_{sb}$, $x_{sb} = 1$ and $x_{sj} = 0$ for all $j \neq b$. This ensures that exactly one of $x_{s1}, \ldots, x_{sk}$ is assigned 1 and the rest are assigned 0. This modification can be easily incorporated so that Theorem 2.5 holds. Following from the above discussion and Theorem 2.5, we have the following theorem.

THEOREM 2.6. *For each $\epsilon > 0$, there is a polynomial time algorithm that, for each instance Q and K of LBA, produces a completion T' of K such that $|Q_{T'} \cap Q| \geq |Q_{\hat{T}} \cap Q| - \epsilon n^4$ where $\hat{T}$ is an optimal completion of K.*

Combining the above results we can establish the following approximation result.

THEOREM 2.7. *For each $\epsilon > 0$, there is a polynomial time algorithm that, for each instance Q of* MQC*, produces a tree T_{APX} such that $|Q_{T_{APX}} \cap Q| \geq (1-\epsilon)|Q_{T_{OPT}} \cap Q|$.*

Proof. Let T_{OPT} and T_{APX} be defined as before and T_k be a k-bin decomposition of T_{OPT} that satisfies Theorem 2.2 for some constant k to be determined. Combining Theorems 2.6 and 2.2, we have that

$$\begin{aligned} |Q_{T_{APX}} \cap Q| &\geq |Q_{T_{OPT}} \cap Q| - (c'/k + \epsilon_1)n^4 \\ &\geq (1 - c'/(ck) - \epsilon_1/c)|Q_{T_{OPT}} \cap Q| \end{aligned}$$

for any constant $\epsilon_1 > 0$, since $|Q_{T_{OPT}} \cap Q| \geq cn^4$. The theorem result follows by choosing ϵ_1 sufficiently small and k sufficiently large. □

3. Quartet cleaning. Let T be an evolutionary tree and Q an estimate of Q_T. In order to correct quartet errors in Q we assume the following quartet error model: For each edge e of T there are at most $\alpha\sqrt{|Q_T(e)|}$ quartet errors in Q involving quartets across e where α is a constant to be determined and $Q_T(e)$ denotes the set of quartet topologies across the edge e of T (see Figure 1.4(i)). In this section we present a polynomial algorithm for correcting all quartet errors in Q under this error model. It is also shown that the above upper bound on quartet errors is asymptotically tight by proving a matching lower bound. More precisely, we prove that no algorithm can correctly infer the tree T when the set Q contains more than $\sqrt{|Q_T(e)|}$ errors across some edge e. Therefore, our algorithm is (asymptotically) optimal in terms of its power to correct quartet errors across an edge of T.

The section is organized as follows. We first define a variant of MQC, called the *minimum inconsistent balanced bipartition* (MIBB) problem, and devise a polynomial time approximation algorithm for MIBB with an additive error of ϵn^4 for any constant $\epsilon > 0$, using the same technique utilized by the PTAS for MQC. This approximation algorithm is then used recursively to clean quartets. The lower bound on quartet errors is given in section 3.3.

3.1. MIBB and its approximation. Let $S = \{1, \ldots, n\}$ denote the set of leaf labels. Each edge e of the evolutionary tree T induces a bipartition $X|Y$ of the labels. The quartets across the edge e are also referred to as the quartets induced by the bipartition $X|Y$. The bipartition $X|Y$ is called a *balanced bipartition* if $|X| \leq 2n/3$ and $|Y| \leq 2n/3$. An *edge separator* of the tree T is any edge that induces a bipartition $X|Y$ with the property that $|X| \leq 8n/9$ and $|Y| \leq 8n/9$. It is easy to see that T has at least one edge separator. We consider the following variant of MQC.

Minimum Inconsistent Balanced Bipartition (MIBB).
Instance: Set Q containing a quartet topology for each quartet of labels in S.
Goal: Find a balanced bipartition $A|B$ that induces the minimum number of quartet topologies inconsistent with the set Q. That is, we want to minimize the number of quartets $\{a, b, c, d\} \subseteq S$ such that $a, b \in A$, $c, d \in B$, and $ac|bd \in Q$.

MIBB is known to be NP-hard [6]. By formulating MIBB as a 2-bin variant of LBA, an approximation algorithm for MIBB with additive error ϵn^4 can be derived for any constant $\epsilon > 0$. This results in the following theorem.

Theorem 3.1. *For each $\epsilon > 0$, there is a polynomial time algorithm that produces a balanced bipartition $A|B$ that induces at most ϵn^4 more quartet topologies inconsistent with the set Q than an optimal balanced bipartition.*

3.2. A recursive algorithm for cleaning quartets. In this section we prove the following.

Theorem 3.2. *For some $\alpha > 0$, there is a polynomial time algorithm that produces the correct evolutionary tree T given a (complete) set Q of quartet topologies which contains at most $\alpha\sqrt{|Q_T(e)|}$ errors across any edge e of T.*

Before describing the algorithm in detail, we sketch its basic idea. First, we observe that the bipartition $A|B$ obtained by the approximation algorithm for MIBB on input Q is also a good approximation of a minimum inconsistent balanced bipartition for the set Q_T, since Q contains at most a total of $\alpha n^3 = o(n^4)$ erroneous quartet topologies. Moreover, we show that the bipartition $A|B$ in fact comes very close to the bipartition $X|Y$ induced by some edge separator of T, i.e., the symmetric differences

$A \oplus X$ and $B \oplus Y$ are very small. We *bootstrap* the bipartition $A|B$ by repeatedly swapping and joining incorrectly placed pairs of labels until it actually becomes the correct bipartition $X|Y$. Then we recursively bipartition sets X and Y independently, taking into account the labels in the set Y and X, respectively. Consider the case of bipartitioning X with the presence of Y. Let T_X denote the subtree of T induced by X. Observe that we can still approximately bipartition the set X as long as the errors in Q across an edge separator of T_X is significantly smaller than $|X|^4$. That is, we should have $|X|^4 \gg \alpha\sqrt{n^2|X|^2} = \alpha n|X|$, i.e., $|X| \gg n^{1/3}$. Hence, we stop the recursion at $|X| = \sqrt{n}$ and switch to a different algorithm which attempts to reconstruct the subtree T_X by directly taking advantage of the existence of a large "reference set" $Y = S - X$.[3] The details of the three main parts of the quartet cleaning algorithm are given below.

• **Inferring the first bipartition of *T*.** Fix a sufficiently small constant $\epsilon > 0$ so that all the inequalities below involving ϵ will hold. We run the approximation algorithm for MIBB on set Q to get a balanced bipartition $A|B$ of S. By Theorem 3.1, $A|B$ induces at most ϵn^4 quartet topologies inconsistent with Q and thus at most $\epsilon n^4 + O(n^3)$ quartet topologies inconsistent with Q_T. The following lemma shows that $A|B$ is "almost correct" in the sense that it is actually very close to the bipartition induced by some edge separator of T.

LEMMA 3.3. *Let e_0 be an edge separator of T inducing a bipartition $X|Y$ such that* (i) $|A \cap X| \geq |A|/3 \geq n/9$, (ii) $|B \cap Y| \geq |B|/3 \geq n/9$, *and* (iii) $\max\{|A \cap Y|, |B \cap X|\}$ *is minimized. (It is easy to see that such an edge e_0 exists.) Then* $\max\{|A \cap Y|, |B \cap X|\} \leq 11\epsilon^{1/3}n$.

Proof. Without loss of generality, assume that $|A \cap Y| \geq |B \cap X|$. Let $\epsilon_1 = |B \cap X|/n$ and $\epsilon_2 = |A \cap Y|/n$. Then

$$|Y| \geq |B \cap Y| = |B| - |B \cap X| \geq (1 - \epsilon_1)|B|.$$

Since each quartet $\{a, b, c, d\}$, where $a \in A \cap X$, $b \in A \cap Y$, $c \in B \cap X$, and $d \in B \cap Y$, yields a topology $ac|bd$ in T which is inconsistent with the bipartition $A|B$, we have

$$(n/9) \cdot (\epsilon_1 n) \cdot (\epsilon_1 n) \cdot (n/9) \leq \epsilon n^4 + O(n^3),$$

which implies that

$$\epsilon_1 \leq 9\sqrt{\epsilon}. \tag{3.1}$$

Suppose that the set Y is further partitioned into subsets Y_1 and Y_2, as illustrated in Figure 3.1. Without loss of generality, assume that $|B \cap Y_1| \leq |B \cap Y_2|$. Suppose that the constant ϵ (and thus the constant ϵ_1) is so small that

$$|B \cap Y_2| \geq (1 - \epsilon_1)|B| - |B \cap Y_1| \geq (1 - \epsilon_1)|B| - |B|/2 \geq |B|/3.$$

Thus, the edge e_1 inducing the partition $X \cup Y_1|Y_2$, as illustrated in Figure 3.1, is also an edge separator satisfying the conditions (i) and (ii) of the lemma. Then, we must have

$$|B \cap X| + |B \cap Y_1| \geq |A \cap Y|,$$

[3] There is certain analogy between this algorithm and the idea of using an *out-group* to root an evolutionary tree in biology.

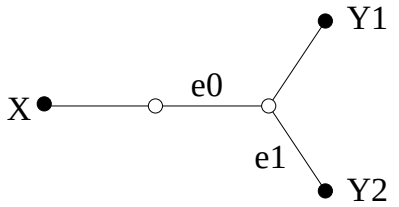

FIG. 3.1. *The edge separator e_0 and its induced bipartition.*

because otherwise the edge e_1 would have been a better choice than the edge e_0, violating the condition (iii) of the lemma. Since $|B \cap X| \leq \epsilon_1 n$,

$$|B \cap Y_1| \geq |A \cap Y| - \epsilon_1 n.$$

Since each quartet $\{a, b, c, d\}$, where $a \in A \cap X$, $b \in A \cap Y_1$, $c \in B \cap Y_1$, and $d \in B \cap Y_2$, yields a topology $ad|bc$ in T which is inconsistent with the bipartition $A|B$, we have

$$(n/9) \cdot |A \cap Y_1| \cdot (|A \cap Y| - \epsilon_1 n) \cdot (|A \cap Y| - \epsilon_1 n) \leq \epsilon n^4 + O(n^3).$$

Similarly, since each quartet $\{a, b, c, d\}$, where $a \in A \cap X$, $b \in A \cap Y_2$, $c \in B \cap Y_1$, and $d \in B \cap Y_2$, yields a topology $ac|bd$ in T which is inconsistent with the bipartition $A|B$,

$$(n/9) \cdot |A \cap Y_2| \cdot (|A \cap Y| - \epsilon_1 n) \cdot (|A \cap Y| - \epsilon_1 n) \leq \epsilon n^4 + O(n^3).$$

Summing up the above two inequalities, we have

$$(n/9) \cdot |A \cap Y| \cdot (|A \cap Y| - \epsilon_1 n)^2 \leq 2\epsilon n^4 + O(n^3).$$

Therefore,

$$\epsilon_2(\epsilon_2 - \epsilon_1)^2 \leq 18\epsilon.$$

It follows from (3.1) and the fact $\epsilon < 1$ that

$$\epsilon_2 < 11\epsilon^{1/3}. \qquad \square \tag{3.2}$$

Since the bipartition $A|B$ may still be incorrect (i.e., it may not be the bipartition induced by any edge of T), we try to revise it so it becomes eventually correct. From now on, let $\epsilon' = 11\epsilon^{1/3}$. We first try to detect the small number of pairs of labels that are "reversed" in the bipartition $A|B$. For any pair (a, b) of leaves, where $a \in A$ and $b \in B$, let us analyze how many quartet topologies $ax|by$ in the set Q, where $x \in A - \{a\}$ and $y \in B - \{b\}$, would "support" (i.e., be consistent with) the placement that the label a is in the set A and the label b is in the set B if (i) $a \in X$ and $b \in Y$ or (ii) $a \in Y$ and $b \in X$.

LEMMA 3.4. *Q has at least $((1-3\epsilon')^2 - \alpha/2)|X| \cdot |Y|$ quartet topologies supporting the placement $a \in A$ and $b \in B$ in case* (i) *(correctly), and at most $(12\epsilon' + \alpha/2)|X| \cdot |Y|$ quartet topologies supporting the placement $a \in A$ and $b \in B$ in case* (ii) *(incorrectly).*

Proof. Suppose that $a \in X$ and $b \in Y$. Let us first analyze how many quartet topologies $ax|by$ in the set Q_T, where $x \in A - \{a\}$ and $y \in B - \{b\}$, across the edge e_0 support the placement $a \in A$ and $b \in B$. Clearly, for any $x \in A \cap X - \{a\}$ and

$y \in B \cap Y - \{b\}$, the quartet topology $ax|by \in Q_T$ supports the placement. Hence the number of such supportive quartet topologies in Q_T across the edge e_0 is at least (roughly, omitting minor terms)

$$|A \cap X| \cdot |B \cap Y| \geq (1 - 3\epsilon')^2 |X| \cdot |Y|.$$

Since the set Q contains at most $\alpha|X| \cdot |Y|/2$ erroneous quartet topologies across the edge e_0, Q has at least

$$((1 - 3\epsilon')^2 - \alpha/2)|X| \cdot |Y|$$

quartet topologies supporting the placement $a \in A$ and $b \in B$.

The quartet topologies in Q supporting the placement $a \in A$ and $b \in B$ in case (ii) is at most

$$\begin{aligned} &|A| \cdot |B| - ((1 - 3\epsilon')^2 - \alpha/2)|X| \cdot |Y| \\ &\leq ((1 + 3\epsilon')^2 - (1 - 3\epsilon')^2 + \alpha/2)|X| \cdot |Y| \\ &= (12\epsilon' + \alpha/2)|X| \cdot |Y| \qquad \square \end{aligned}$$

Supposing that α is small enough and choosing ϵ sufficiently small, we can guarantee that the following inequality

$$\frac{12\epsilon' + \alpha/2}{(1 - 3\epsilon')^2 - \alpha/2} \leq \beta \tag{3.3}$$

holds for some small (but not too small) threshold β (exact value to be determined). Thus, from the set Q, we can decide if we should keep placing a in set A and b in set B, or switch them by checking the ratio between the supportive quartet topologies for each case.

We repeat the above test and correction until we cannot find any pair of labels to swap. Note that, in this process we may also swap pairs (a, b) with the property that $a \in A$, $b \in B$, and $a, b \in Y$ (or $a, b \in X$). That is, for such pairs we should (correctly) join them in the set Y (or X, respectively), but the quartet topologies in Q_T (and thus Q) tell us that they are reversed and we should swap them according to the above separation ratio. When this (e.g., $a, b \in Y$) happens, it must be the case that Y is bipartitioned into subsets Y_1 and Y_2 in the tree T, $a \in Y_1$, $b \in Y_2$, and $|Y_2| < \beta'|Y_1|$ for some constant β' depending on β, ϵ, and α. So, if we make sure that β is so small that $\beta' < 1/2$, then we won't switch such a pair (a, b) back and forth. Hence the process will converge in $O(n)$ swaps.

When the above process terminates, the bipartition $A|B$ may still not be consistent with any edge (separator) of T. But we must now have the property that either $X \subseteq A$ or $Y \subseteq B$, although we do not know which situation holds. So, in the following we try to further improve the bipartition $A|B$ assuming each situation separately.

Consider the case $X \subseteq A$ (the other case is symmetric). Let e be the edge of T whose induced bipartition, denoted $X'|Y'$, has the largest set X' that is completely contained in A. It is easy to see that $(1-3\epsilon')|Y'| \leq (1-3\epsilon')|Y| \leq |B \cap Y| \leq |B| \leq |Y'|$. Observe that e is in fact an edge separator. We will try to modify $A|B$ so it becomes the bipartition $X'|Y'$. Suppose that Y' is further bipartitioned into subsets Y_1' and Y_2' in the tree T, where $|Y_1'| \leq |Y_2'|$. By the choice of the edge e, $Y_1' \not\subseteq A$ and $Y_2' \not\subseteq A$. Moreover, from the above discussion on convergence, we know that if $|Y_1'| < \beta'|Y_2'|$ then $Y_2' \subseteq B$. Again, we take pairs of labels (a, b), where $a \in A$ and $b \in B$, and

analyze the support from Q for joining a and b in the set B if (i) $a \in X'$, or (ii) $a \in Y_i'$ and $b \in Y_i'$, for some i.

LEMMA 3.5. *Q contains at least* $(\min\{1/2 - 3\epsilon', \beta'/(1+\beta')\} - \alpha/2)|X'| \cdot |Y'|$ *supportive quartet topologies in case* (i) *and at most* $(6\epsilon' + \alpha/2)|X'| \cdot |Y'|$ *supportive quartet topologies in case* (ii).

Proof. Suppose that $a \in X'$. Q_T may contain a supportive quartet topology $ab|xy$, where $x \in A$ and $y \in B$, only if $x \in A \cap Y'$. Thus, the number of supportive quartet topologies in this case is at most

$$|Y' - Y' \cap B| \cdot |B| \leq 3\epsilon'|Y'|^2 \leq 6\epsilon'|X'| \cdot |Y'|.$$

Observe that, out of the $|X'|^2 \cdot |Y'|^2/4$ quartet topologies in Q_T across the edge e, only $\alpha|X'| \cdot |Y'|/2$ of them can go wrong in Q and become supportive; we claim Q contains at most

$$(6\epsilon' + \alpha/2)|X'| \cdot |Y'|$$

supportive quartet topologies.

Now suppose that $a \in Y_i'$ and $b \in Y_i'$ for some i. We consider two subcases. If $a \in Y_1'$ and $b \in Y_1'$, then all quartet topologies $ab|xy$ in Q_T, where $x \in A \cap X'$ and $y \in B \cap Y_2'$, would certainly be supportive. Hence, the number of supportive quartet topologies (across the edge connecting the sets $X \cup Y_2'$ and Y_1' in T) is at least

$$|X'| \cdot |Y_2'| \geq |X'| \cdot |Y'| \cdot (1/2 - 3\epsilon')$$

If $a \in Y_2'$ and $b \in Y_2'$, then $Y_2' \not\subseteq B$ and thus $|Y_1'| \geq \beta'|Y_2'|$. Since all quartet topologies $ab|xy$ in Q_T, where $x \in A \cap X'$ and $y \in B \cap Y_1'$, are supportive in this case, we have at least

$$|X'| \cdot |Y_1'| \geq |X'| \cdot \beta'|Y'|/(1+\beta')$$

supportive quartet topologies across the edge connecting the sets $X \cup Y_1'$ and Y_2' in T. Hence, Q contains at least

$$(\min\{1/2 - 3\epsilon', \beta'/(1+\beta')\} - \alpha/2)|X'| \cdot |Y'|$$

supportive quartet topologies in either subcase. ∎

Therefore, if we assume that α is sufficiently small, choose ϵ small enough and keep β' sufficiently large relative to α, then we can ensure that the inequality

$$\frac{6\epsilon' + \alpha/2}{\min\{1/2 - 3\epsilon', \beta'/(1+\beta')\} - \alpha/2} < \gamma \tag{3.4}$$

for some small threshold $\gamma < 1$. In other words, the set Q would contain sufficient information for us to tell if we should move a from the set A to the set B or not.

So we repeat the above step until (i) we cannot find any label to move or (ii) the size of A is getting below $|X| \geq n/9$. Observe that if we do not move anything at all in the whole process, then $A|B = X'|Y'$. In case (ii), we know that we have been moving in the wrong direction (i.e., $Y \subseteq B$ before the process started). In case (i), we could either get a correct bipartition $A|B = X'|Y'$ if $X \subseteq A$, or possibly an incorrect bipartition if $Y \subseteq B$. To check if the resulting bipartition is indeed one induced by some edge separator of T, all we need is to check if any pair (a, b), where $a \in A$ and

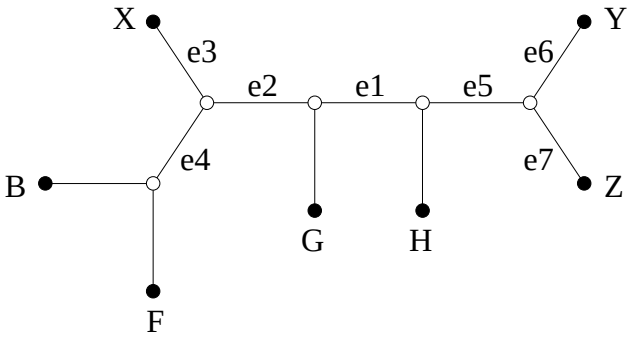

FIG. 3.2. *The placement of the labels* $b \in B, x \in X, y \in Y$, *and* $z \in Z$ *in* T.

$b \in B$, should be reversed under the criterion in inequality (3.3). Therefore, we can obtain at least one correct bipartition induced by some edge separator e' (which may or may not be e) of T, and clean up the erroneous quartets in Q across the edge e' safely.

• **The recursive step.** To continue cleaning up all errors in Q (or, equivalently, constructing the full T), we proceed in two stages. In the first stage, we recursively bipartition sets of sizes larger than $\sqrt{n}$. The general form of the problem is, given a correct bipartition $A|B$ of S, where $\sqrt{n} \leq |A| \leq 2n/3$, we try to find a bipartition of the set A consistent with T. The basic idea is the same as before, but we have to take into account the set B. Observe that since in the subtree of T induced by A, denoted as T_A, there can be at most $\alpha\sqrt{|A|^2 \cdot |B|^2} = \alpha|A| \cdot |B| = o(|A|^4)$ erroneous quartets across any edge of T_A, the approximation algorithm for MIBB will work on input Q_A (i.e., Q restricted to A), and will give us a good approximation of the bipartition induced by some edge separator of T_A as claimed in Lemma 3.3.

Let this approximate bipartition be $A_1|A_2$. We "hypothesize" if B should be placed in A_1 or A_2 by considering two bipartitions $A_1 \cup B|A_2$ and $A_1|A_2 \cup B$. For the bipartition $A_1 \cup B|A_2$, we swap and join pairs (a, b), where $a \in A_1$ and $b \in A_2$, as before, but by considering quartets $\{a, b, x, y\}$ of the special form $x \in B$ and $y \in A_2$. Similarly, for the bipartition $A_1|A_2 \cup B$, we swap and join pairs (a, b), where $a \in A_1$ and $b \in A_2$, by considering quartets $\{a, b, x, y\}$ of the form $x \in A_1$ and $y \in B$. It is straightforward to show that Lemmas 3.4 and 3.5 can be easily extended to work for these bipartitions. At the end we can again check which of the bipartitions yields a correct bipartition consistent with some edge separator of T_A as before.

• **Terminating the recursion.** We use the above recursive step to produce correct bipartitions $A|B$, where $|A| \leq \sqrt{n}$. Now we cannot continue the recursion since the errors in Q may jeopardize the performance of the approximation algorithm for MIBB when $|A|$ gets below $n^{1/3}$. So, we turn to a more direct cleaning approach which makes essential use of the large size of the set B.

Consider a quartet $\{b, x, y, z\}$, where $x, y, z \in A$ and $b \in B$. Suppose that the labels are placed in the tree T as illustrated in Figure 3.2.

Let's consider all quartets consisting of a label $b' \in B$, two of the labels x, y, z, and a label $a \in A - \{x, y, z\}$, and analyze the supports from the sets Q_T and Q for each of the three possible topologies $bx|yz$, $by|xz$, and $bz|xy$. For each edge e, let $f_Q(e)$ denote the number of errors in Q across edge e.

LEMMA 3.6. *The set Q contains at least*

$$\begin{aligned}&(1-6\alpha/2)|B|\cdot(|H|+|G|+|Y|+|Z|)-f_Q(e_3)-f_Q(e_4)\\&\leq(1-7\alpha/2)|B|\cdot(|H|+|G|+|Y|+|Z|)-\alpha|B|\cdot|X|\end{aligned}$$

more supportive quartet topologies for $bx|yz$ than for $by|xz$ or for $bz|xy$.

Proof. We first calculate the number of such quartet topologies in Q_T across an (arbitrarily chosen) edge e_1 as illustrated in the figure that support the correct topology $bx|yz$. A quartet $\{b', x, y, a\}$ would yield a supportive topology across e_1 if $a \in H \cup Z \cup Y - \{y\}$. So Q_T has $|B|\cdot(|H|+|Y|+|Z|-1)$ such supportive quartet topologies. Similarly, we know that Q_T has $|B|\cdot(|H|+|Y|+|Z|-1)$ supportive quartet topologies across e_1 of the form $b'x|az$ and $|B|\cdot(|F|+|G|+|X|)$ supportive quartet topologies across e_1 of the form $b'a|yz$. Hence, Q_T has a total of

$$|B|\cdot(2|H|+2|Y|+2|Z|+|F|+|G|+|X|-2)$$

supportive quartet topologies across the edge e_1. This implies that Q has at least

$$\begin{aligned}&|B|\cdot(2|H|+2|Y|+2|Z|+|F|+|G|+|X|-2)\\&-(\alpha/4)(|B|+|X|+|F|+|G|)\cdot(|Y|+|Z|+|H|)\\&\geq\ |B|\cdot(2|H|+2|Y|+2|Z|+|F|+|G|+|X|-\alpha(|Y|+|Z|+|H|)/4)\end{aligned}$$

supportive quartet topologies.

Let's calculate the number of quartet topologies in Q_T supporting the topology $by|xz$. A quartet $\{b', x, y, a\}$ would yield a supportive topology only if $a \in X - \{x\}$, and thus Q_T has $|B|\cdot(|X|-1)$ such supportive quartet topologies. Similarly, Q_T has $|B|\cdot|F|$ supportive quartet topologies of the form $b'a|xz$ and $|B|\cdot(|Z|-1)$ supportive quartet topologies of the form $b'y|az$. Hence, Q_T has a total of

$$|B|\cdot(|X|+|Z|+|F|-2)$$

supportive quartet topologies. This implies that Q has at most

$$|B|\cdot(|X|+|Z|+|F|-2)+\sum_{i=1}^{7}f_Q(e_i)$$

supportive quartet topologies, where $f_Q(e_i) \leq \alpha\sqrt{|Q_T(e_i)|}$ denotes the number of errors across the edge e_i in the current set Q.

Using the same idea, we claim that the set Q has at most

$$|B|\cdot(|X|+|Y|+|F|-2)+\sum_{i=1}^{7}f_Q(e_i)$$

members supporting the topology $bz|xy$.

So the difference between the support for the correct topology $bx|yz$ and that for incorrect ones $by|xz$ or $bz|xy$ is at least

$$\begin{aligned}&(1-\alpha/4)|B|\cdot(2|H|+|G|+|Y|+|Z|)-\sum_{i=1}^{7}f_Q(e_i)\\&\leq(1-\alpha/4)|B|\cdot(|H|+|G|+|Y|+|Z|)\end{aligned}$$

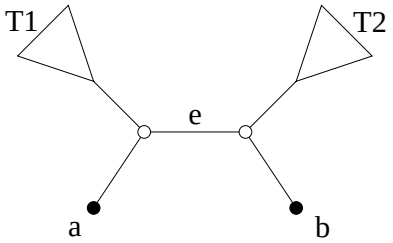

FIG. 3.3. *A hard case for quartet cleaning.*

$$\begin{aligned}
&- 5 \cdot (\alpha/4)|B| \cdot (|H| + |G| + |Y| + |Z|) - f_Q(e_3) - f_Q(e_4) \\
&\leq (1 - 6\alpha/4)|B| \cdot (|H| + |G| + |Y| + |Z|) - f_Q(e_3) - f_Q(e_4) \\
&\leq (1 - 6\alpha/4)|B| \cdot (|H| + |G| + |Y| + |Z|) \\
&\quad - \alpha|B| \cdot |X|/4 - \alpha|B| \cdot (|X| + |H| + |G| + |Y| + |Z|)/4 \\
&\leq (1 - 7\alpha/4)|B| \cdot (|H| + |G| + |Y| + |Z|) - \alpha|B| \cdot |X|/2. \qquad \square
\end{aligned}$$

The above difference between the supports is at least $-\alpha|B| \cdot |X|/2$ and would in fact be at least $(1 - 9\alpha/2)|B| \cdot (|H| + |G| + |Y| + |Z|)$ if (i) $|X| \leq |H| + |G| + |Y| + |Z|$ or (ii) the erroneous quartet topologies in Q across the edges e_3 and e_4 have already been fixed. This suggests that we should first work on the quartet $\{b, x, y, z\}$ which yields the largest difference.

More precisely, our algorithm finds the quartet $\{b, x, y, z\}$, where $x, y, z \in A$ and $b \in B$, with the largest margin in the supports from Q for each of its three possible topologies, and correct Q according to the topology with the highest support. We then consider the remaining quartets $\{b, x, y, z\}$ of the form $x, y, z \in A$ and $b \in B$, and repeat the same operation until correct topologies for all such quartets have been found.

3.3. An upper bound for quartet cleaning. The following theorem establishes an upper bound on the number of quartet errors across an edge that can be corrected.

THEOREM 3.7. *No algorithm can correctly reconstruct the evolutionary tree T if Q contains $\sqrt{|Q_T(e)|}$ or more erroneous quartet topologies across some edge e of T.*

Proof. Consider the tree in Figure 3.3 and quartets of the form $\{a, b, x, y\}$ where $x \in T_1$ and $y \in T_2$. If half of these quartets have topology $ax|by$ in Q whereas the other half have topology $ay|bx$ in Q then it cannot be decided which of these quartet topologies are erroneous. It follows that there must be no more that $|T_1||T_2|/2$ quartet errors across e. The theorem follows from

$$|T_1||T_2|/2 \leq \sqrt{|Q_T(e)|}. \qquad \square$$

4. Complete MQC is NP-complete. In this section we demonstrate that the following problem is NP-complete.

MAXIMUM QUARTET CONSISTENCY (DECISION) (MQCD).
Instance: A complete set Q of quartet topologies over label set S and nonnegative integer k.
Question: Is there an evolutionary tree T labeled by S such that $|Q_T \cap Q| \geq k$?

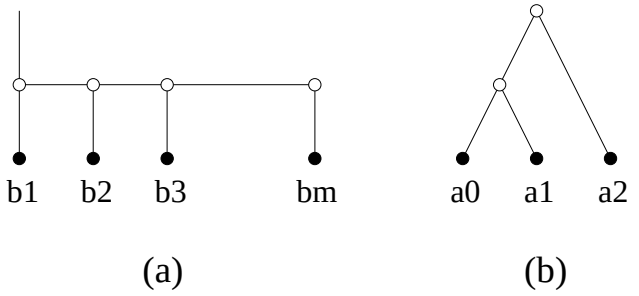

FIG. 4.1. (a) *The caterpillar tree M with $m \gg n^5$ feet (leaves).* (b) *Each label $a \in S$ is replaced by three $a_0, a_1, a_2 \in S'$.*

It is clear that MQC is in NP. To complete the proof we reduce Quartet Compatibility to MQC.

QUARTET COMPATIBILITY.
Instance: A (possibly incomplete) set Q of quartet topologies on label set S.
Question: Is Q compatible, i.e., does there exist an evolutionary tree T labelled by S such that $Q \subseteq Q_T$?

This problem is known to be NP-hard [13]. Given an instance of **Quartet Compatibility** with a quartet topology set Q defined on a label set S, where $n = |S|$, we will construct an instance of MQC with a *complete* quartet topology set Q' defined on a new label set S' such that there is a evolutionary tree T realizing Q with no error if and only if there is a evolutionary tree T' realizing Q' with at most $g(n)m + f(n)$ errors, where $f(n) = O(n^4)$, $g(n) = O(n^3)$ and $m \gg n^5$ will be specified later.

The basic idea behind this proof is to add topologies to Q for quartets that are not specified in Q to create a complete set Q' such that with respect to any optimal evolutionary tree T' for Q', precisely one third of the added quartet topologies are correct. In order to do this, we will need a large evolutionary tree M on m leaves with a fixed (e.g., caterpillar) structure (see Figure 4.1 (a)), where m is a number that is both divisible by three and much larger than the number of missing quartet topologies in Q, e.g., $m \gg n^5$. M will be embedded as a subtree in the optimal evolutionary tree T' and will be used to enforce certain useful structures in T'.

We construct the sets S' and Q' as follows. Add the m leaf-labels $b_1, \ldots, b_m$ in M to S', and for each $a \in S$, create three new labels a_0, a_1, a_2 and add them to S'. Note that we do not add the original labels in S to S'. We want to specify Q' such that (i) each triplet of labels a_0, a_1, a_2 appear together in a subtree of T' as in Figure 4.1(b) and (ii) the optimal evolutionary tree T' for Q' is formed by attaching M to some branch of a evolutionary tree $\hat{T}$ that is obtained from an optimal evolutionary tree T for Q by replacing every leaf of T with a subtree containing three leaves as shown in Figure 4.1(b).

Intuitively, Q' must be constructed to enforce the following conditions:

- M appears intact in T' as in Figure 4.1(a).
- Each created triplet of labels a_0, a_1, a_2 appear together in a subtree of T' as in Figure 4.1(b), with nothing else inserted between them.
- The quartet topologies in Q extend naturally to Q'. That is, if $ab|cd \in Q$, then $a_i b_j | c_k d_l$ is included in Q' for all $0 \le i, j, k, l \le 2$.
- For each quartet involving one label from M and three labels corresponding to three distinct elements of S, its topology is related to the (unknown) structure

of T (or $\hat{T}$) and the branch of $\hat{T}$ where M is attached. Hence, we should make sure that the number of erroneous topologies induced from such quartets is independent of the structure of T and the location where M is attached.

- For all quartets on S' that correspond to labels of Q with missing topologies in Q, we add quartet topologies in Q' such that precisely one third of these new quartet topologies are satisfied in T'. This is the difficult part since we do not know the structure of T'.

Here are the details of the construction. Let $w, x, y, z \in S'$ be four distinct labels.

1. If the labels are all in M, specify the quartet topology according to the structure of M as shown in Figure 4.1(a).
2. If $w = a_i$ for some $a \in S$, $x = b_j, y = b_k$, and $z = b_l$ with $j < k < l$, specify the topology as $wx|yz$.
3. If $w, x \in M$ and $y, z \notin M$, specify the topology as $wx|yz$.
4. If $w = b_i \in M$ and $x, y, z \notin M$, we consider two subcases.
   (a) If at least two of x, y, z correspond to the same label in S, then the quartet topology can be specified according to the required structure of T' described above.
   (b) If x, y, z correspond to distinct labels in S, then specify the topology as $b_i x|yz$ if $i \le m/3$, or as $b_i y|xz$ if $m/3 < i \le 2m/3$, or as $b_i z|xy$ if $i > 2m/3$. Intuitively, here we are partitioning M into three subsets $\{b_1, \ldots, b_{m/3}\}$, $\{b_{m/3+1}, \ldots, b_{2m/3}\}$, and $\{b_{2m/3+1}, \ldots, b_m\}$ so that quartets of the above form will introduce the same number of errors no matter how T looks and where M is attached in $\hat{T}$.
5. Finally, if none of the labels are from M, we consider three subcases.
   (a) If at least two of them correspond to the same label in S, then this quartet topology can be specified as before.
   (b) If they correspond to a quartet on S that has a resolved topology in Q, then specify the same topology in Q'.
   (c) If they correspond to a quartet on S whose topology is missing in Q, we take care of all such quartets collectively. Recall that each such quartet of labels $w, x, y, z \in S$ corresponds to $3^4 = 81$ different quartets on S'. We divide them into $81/3 = 27$ disjoint *groups*. Each group contains three quartets:

$$w_0, x_i, y_j, z_k,$$
$$w_1, x_{i+1 \bmod 3}, y_{j+1 \bmod 3}, z_{k+1 \bmod 3},$$
$$w_2, x_{i+2 \bmod 3}, y_{j+2 \bmod 3}, z_{k+2 \bmod 3}.$$

   For each group, we specify quartet topologies as follows:

$$w_0 x_i | y_j z_k,$$
$$w_1 y_{j+1 \bmod 3} | x_{i+1 \bmod 3} z_{k+1 \bmod 3},$$
$$w_2 z_{k+2 \bmod 3} | x_{i+2 \bmod 3} y_{j+2 \bmod 3}.$$

Obviously, the quartet topologies defined in items 1, 2, 3, 4(a), 5(a), and 5(b) do not introduce any errors in T' if M stays as shown in Figure 4.1(a) and for each label $a \in S$, its associated triplet of labels $a_0, a_1, a_2 \in S'$ appear together in a subtree of T' as shown in Figure 4.1(b). The following lemmas give exact error bounds for quartet topologies defined in items 4(b) and 5(c).

LEMMA 4.1. *If M does not split, item* 4(b) *introduces precisely $g(n)m$ quartet errors, where $g(n) = 18\binom{n}{3}$.*

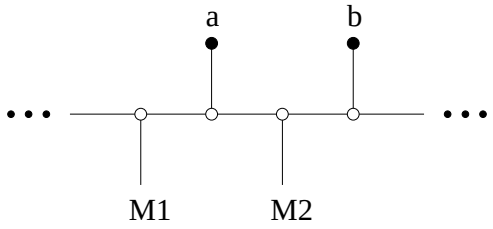

FIG. 4.2. *M is split by label a.*

Proof. The number of quartets considered in item 4(b) is $\binom{n}{3} \cdot 3^3 \cdot \frac{2m}{3} = 18\binom{n}{3}m = g(n)m$. ∎

LEMMA 4.2. *Let $q(n) = \binom{n}{4} - |Q|$ be the number of missing quartet topologies in Q. If M does not split, then for each $a \in S$, its associated triplet of labels $a_0, a_1, a_2 \in S'$ appear together in a subtree of T' as shown in Figure* 4.1(b) *and item* 5(c) *introduces precisely $f(n) = 54q(n)$ quartet errors.*

Proof. Suppose that for some $a \in S$, the triplet $a_0, a_1, a_2 \in S'$ do not appear together in a subtree as shown in Figure 4.1(b). Assume for the worst case that there is a single leaf x separating these three elements in T' as shown below: Then, for any label $y \in M$, the quartet topology $a_0x|a_1y$ in Q' would be an error. As there are at least m such y, this gives rise to at least m quartet errors. Since $m \gg n^5$, we can improve T' by moving x away from the triplet a_0, a_1, a_2, which contradicts to the fact that T' is optimal for Q'.

Given that each triplet of labels corresponding to a label in L appear together in a subtree in T' as shown in Figure 4.1(b), precisely one third of the 81 quartet topologies specified in item 5(c) for each missing quartet in Q are correct. Hence, item 5(c) introduces precisely $81 \cdot \frac{2}{3}q(n) = 54q(n) = f(n)$ quartet errors. ∎

Finally, we have to show that M does not split in the optimal evolutionary tree T'.

LEMMA 4.3. *In the optimal evolutionary tree T' realizing Q', M stays intact as the caterpillar shown in Figure* 4.1(a).

Proof. Splitting M may reduce the error terms in Lemma 4.1 and Lemma 4.2. We argue that such splittings are not worthwhile because they introduce more quartet errors to T' than they can save.

To illustrate the idea of the argument, suppose that some labels $a, b \notin M$ split M into two subsets M_1 and M_2, as shown in Figure 4.2.

In this case, we may (or may not) save at most $g(n) \cdot \min\{|M_1|, |M_2|\}$ errors from item 4 that. But this splitting creates at least $\Omega(|M_1| \cdot |M_2|)$ new errors because all quartet topologies of the form

$$m_1a|m_2b,$$

where $m_1 \in M_1, m_2 \in M_2$, are erroneous. Since $m \gg n^5$ and $g(n) = O(n^3)$, it is easy to see that $|M_1| \cdot |M_2| \gg g(n) \cdot \min\{|M_1|, |M_2|\}$. Hence, the splitting in fact creates more errors than it can save.

Therefore, in the optimal evolutionary tree T', M stays in one piece as a caterpillar subtree. ∎

From the above lemmas, we conclude that Q is compatible if and only if there exists an evolutionary tree T' that is inconsistent with Q' on at most (in fact, exactly) $g(n)m + f(n)$ quartets.

5. Discussion. In practice, the inference of quartet topology is not reliable, and so, confidence levels are assigned to quartet topologies. For example, for quartet $\{a, b, c, d\}$ the quartet topologies $ab|cd$, $ac|bd$, and $ad|bc$ may be assigned confidence levels 80%, 15%, and 5% indicating that we have the most confidence in the inference $ab|cd$ but that this confidence is not 100%. Given this information, the weighted MQC problem is to obtain an evolutionary tree T that maximizes

$$\sum_{ab|cd \in Q_T} w(ab|cd),$$

where $w(ab|cd)$ denotes the confidence level of quartet topology $ab|cd$. The MQC PTAS can be extended to solve this weighted variation on MQC as long as the weights are drawn from some interval of positive integers of constant range to preserve the smoothness of the polynomial integer programs. On the other hand, when the weights are allowed to be 0-1, weighted MQC becomes the incomplete MQC problem which is MAX-SNP–hard.

The PTAS for weighted MQC can also be used to solve the quartet consensus problem [6]. In the quartet consensus problem, several evolutionary trees T_1, T_2, ..., T_k compete as alternate hypotheses for the evolutionary history of a label set S. The goal is to produce an evolutionary tree T that maximizes the sum

$$\sum_{i=1}^{k} |Q_T \cap Q_{T_i}|.$$

When k is a constant this can be solved by defining $w(ab|cd)$ to be the number of evolutionary trees T_i in which $ab|cd$ is induced, for each quartet topology $ab|cd$, and then applying the weighted MQC PTAS.

In an error model that restricts the number of quartet errors across an edge, each quartet error may be "charged" to many edges. For example, in Figure 1.4(ii), a quartet error involving the labels a, b, c, and d would be charged to all edges on the path p connecting a, b with c, d. Hence, it is also natural to associate quartet errors with *paths* instead of edges. If p is a path in T then $\{a, b, c, d\}$ is a quartet across p if p contains all the edges crossed by the quartet $\{a, b, c, d\}$, as illustrated in Figure 1.4(ii). Error models that restrict the number of quartet errors across paths and those that restrict the number of quartet errors across edges are incomparable. In general, the former are good at capturing uniformly distributed errors while the latter are better suited for describing localized errors. Let $Q_T(p)$ denote the set of quartets across a path p of T. It is not hard to extend our bipartition-based quartet cleaning algorithm to work under the assumption that Q contains at most $\alpha\sqrt{|Q_T(p)|}$ quartet errors across any path p of T, for some constant $\alpha > 0$.

Several open problems present themselves. In particular, the quartet cleaning technique presented here is based upon an error model that bounds the number of quartet errors across every edge of the evolutionary tree. A method that could clean quartet errors across edges independently would be an improvement. It is hopeless to obtain a PTAS for the sparse MQC problem since it is MAX-SNP–hard. Can we obtain a better than 1/3 approximation for the sparse MQC problem?

Acknowledgment. We would like to thank the anonymous referees for their detailed comments.

REFERENCES

[1] S. Arora, A. Frieze, and H. Kaplan, *A new rounding procedure for the assignment problem with applications to dense graph arrangement problems*, in 37th Annual Symposium on Foundations of Computer Science, IEEE Computer Society Press, 1996, pp. 21–30.

[2] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy, *Proof verification and hardness of approximation problems*, in Proceedings of the 33rd IEEE Symposium on the Foundations of Computer Science, IEEE Computer Society Press, 1992, pp. 14–23.

[3] A. Ben-Dor, B. Chor, D. Graur, R. Ophir, and D. Pelleg, *From four–taxon trees to phylogenies: The case of mammalian evolution*, in Proceedings of the 2nd Annual International Conference on Computational Molecular Biology, ACM Press, 1998, pp. 9–19.

[4] V. Berry, *private communication*, LIRM, Montpellier, France, 1998.

[5] V. Berry and O. Gascuel, *Inferring evolutionary trees with strong combinatorial evidence*, in Proceedings of the 3rd Annual International Computing and Combinatorics Conference, Springer, 1997, pp. 111–123.

[6] D. Bryant, *Structures in Biological Classification*, Ph.D. thesis, Department of Mathematics and Statistics, University of Canterbury, Canterbury, UK, 1997.

[7] P. Buneman, *The recovery of trees from measures of dissimilarity*, in Mathematics in the Archaeological and Historical Sciences, F. R. Hodson, D. G. Kendall, and P. Tautu, eds., Edinburgh University Press, Edinburgh, 1971, pp. 387–395.

[8] P. Erdös, M. Steel, L. Székely, and T. Warnow, *Constructing big trees from short sequences*, in Proceedings of the 24th International Colloquium on Automata, Languages, and Programming, Springer, 1997.

[9] J. Felsenstein, *Evolutionary trees from DNA sequences: A maximum likelihood approach*, J. Molecular Evolution, 17 (1981), pp. 368–376.

[10] J. Felsenstein, *Numerical methods for inferring evolutionary trees*, Quarterly Review of Biology, 57 (1982), pp. 379–404.

[11] P. E. Kearney, *The ordinal quartet method*, in Proceedings of the 2nd Annual International Conference on Computational Molecular Biology, ACM Press, 1998, pp. 125–134.

[12] N. Saitou and M. Nei, *The neighbor–joining method: A new method for reconstructing phylogenetic trees*, Molecular Biology and Evolution, 4 (1987), pp. 406–425.

[13] M. Steel, *The complexity of reconstructing trees from qualitative characters and subtrees*, J. Classification, 9 (1992), pp. 91–116.

[14] K. Strimmer and A. von Haeseler, *Quartet puzzling: A quartet maximum-likelihood method for reconstructing tree topologies*, Molecular Biology and Evolution, 13 (1996), pp. 964–969.

[15] D. L. Swofford, G. J. Olsen, P. J. Waddell, and D. M Hillis, *Phylogenetic inference*, in Molecular Systematics, 2nd ed., D. M. Hillis, C. Moritz, and B. K. Mable, eds., Sinauer Associates, Sunderland, MA, 1996, pp. 407–514.

[16] A. Templeton, *Human origins and analysis of mitochondrial DNA sequences*, Science, 255 (1991), p. 737.

[17] L. Vigilant, M. Stoneking, H. Harpending, H. Hawkes, and A. C. Wilson, *African populations and the evolution of human mitochondrial DNA*, Science, 253 (1991), pp. 1503–1507.