SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

January 2016, Vol. 59 012102:1–012102:14
doi: 10.1007/s11432-015-5355-1

A fixed-parameter algorithm for the maximum agreement forest problem on multifurcating trees

Feng SHI[1], Jianxin WANG[1]*, Yufei YANG[1], Qilong FENG[1],
Weilong LI[1] & Jianer CHEN[1,2]

[1] *School of Information Science and Engineering, Central South University, Changsha* 410083*, China;*
[2] *Department of Computer Science and Engineering, Texas A&M University, College Station, Texas* 77843-3112*, USA*

Received October 8, 2014; accepted April 7, 2015; published online December 21, 2015

Abstract The Maximum Agreement Forest (MAF) problem on two given phylogenetic trees is an important NP-hard problem in the field of computational biology. In this paper, we study the parameterized version of the MAF problem: given two unrooted (multifurcating) phylogenetic trees T_1 and T_2 with the same leaf-label set L, and a parameter k, either construct an agreement forest of at most k trees for T_1 and T_2, or report that no such a forest exists. Whether there is a fixed-parameter tractable algorithm for this problem was posed as an open problem several times in the literature. In this paper, we resolve this open problem by presenting a parameterized algorithm of running time $O(4^k n^5)$ for the problem.

Keywords computational biology, multifurcating phylogenetic tree, maximum agreement forest, TBR distance, fixed-parameter algorithm

Citation Shi F, Wang J X, Yang Y F, et al. A fixed-parameter algorithm for the maximum agreement forest problem on multifurcating trees. Sci China Inf Sci, 2016, 59(1): 012102, doi: 10.1007/s11432-015-5355-1

1 Introduction

Phylogenetic trees have many applications in a variety of fields including vaccine design [1], haplotype analysis [2] and human language evolution [3]. A phylogenetic tree can be represented by a tree T, a *leaf-label set* L, and a one-to-one correspondence λ between the leaves of T and the elements in the set L. A phylogenetic tree is *rooted* if a specific vertex in the tree is designated as the root of the tree, otherwise, the tree is *unrooted*. A *phylogenetic forest* F is a collection of phylogenetic trees with disjoint leaf-label sets, whose union is called the *leaf-label set* for the forest F. Therefore, a phylogenetic tree is a special phylogenetic forest. A phylogenetic forest is *binary* if each of its vertices is of degree either 1 (i.e., a leaf) or 3 (i.e., a non-leaf). An important issue in the study of phylogenetic trees is defining a proper metric to measure the similarity between phylogenetic trees. There have been several such metrics proposed in the literature, such as Robinson-Foulds distance [4], NNI (Nearest Neighbor Interchange) distance [5], TBR (Tree Bisection and Reconnection), and SPR (Subtree Prune and Regraft) distances [6,7].

In this paper, we study the complexity of the *maximum agreement forest* problem, which is closely related to the above proposed metrics for the similarity of phylogenetic trees. We first give some necessary

*Corresponding author (email: jxwang@mail.csu.edu.cn) definitions. Two phylogentic forests F_1 and F_2 are *isomorphic* if there is an isomorphic mapping from F_1 to F_2 such that the corresponding leaves have the same label. The forests F_1 and F_2 are *homeomorphic* if after smoothing degree-2 vertices, they become isomorphic (*smoothing* a degree-2 vertex v means to replace v and its incident edges by a new edge connecting the two neighbors of v). We will simply say that a phylogenetic forest F' *is a subforest* of another phylogenetic forest F if F' is homeomorphic to a subforest of F. Note that because of the leaf labeling, if both F' and F have no degree-1 non-leaf vertices and if F' is homeomorphic to a subforest of F, then such a subforest in F is unique. Therefore, in this case, it makes sense to say that two given forests F_1 and F_2 are *disjoint* or *intersecting* in a forest F if the homeomorphic copies of F_1 and F_2 in F are disjoint or intersecting. An *agreement forest* F^* for two phylogenetic forests F_1 and F_2 with the same leaf-label set L is a phylogenetic forest whose leaf-label set is L such that F^* is a subforest of both F_1 and F_2. The agreement forest F^* for two phylogenetic forests F_1 and F_2 is a *maximum agreement forest* (MAF) if the size of F^* (i.e., the number of trees in F^*) is the smallest over all agreement forests for F_1 and F_2.

The concept of MAF was introduced by Hein et al. [8]. It has been known that the size of MAF precisely characterizes a number of similarity metrics for phylogenetic trees. Allen and Steel [9] proved that the TBR distance between two unrooted binary phylogenetic trees is equal to the size of their MAF minus 1, which is bound by, at least half of, their SPR distance. Bordewich and Semple [10] proved that the rSPR distance between two rooted binary phylogenetic trees is equal to the size of their rooted version of MAF minus 1. Hickey et al. [11] studied the SPR distance between two unrooted binary phylogenetic trees, and proved that their SPR distance is not larger than the size of their MAF minus 1. Baroni et al. [12] proved that the hybridization number of two rooted binary phylogenetic trees is equal to the size of their *maximum acyclic agreement forest* minus 1. In terms of its computational complexity, it is known that computing the size of an MAF of two phylogenetic trees is NP-hard [8].

The theory of parameterized computation and complexity is a recently developed subarea in theoretical computer science. The techniques in this theory have been applied successfully in solving a large number of NP-hard problems [13–18]. In this paper, we will focus on the parameterized version of the MAF problem of two unrooted phylogenetic trees, which is formally defined as follows:

(Parameterized) Maximum Agreement Forest (Max-AF)
Input: two unrooted phylogenetic trees T_1 and T_2 with a common leaf-label set L
Parameter: k
Output: either an agreement forest F of at most k trees for T_1 and T_2 or report
that no such an agreement forest exists

Two instances $(T_1, T_2; k)$ and $(T_1', T_2'; k')$ of the Max-AF problem are *equivalent* if the trees T_1 and T_2 have an agreement forest of size k if and only if the trees T_1' and T_2' have an agreement forest of size k'. A *kernelization algorithm* for the Max-AF problem is a polynomial-time algorithm that on an instance $(T_1, T_2; k)$ of Max-AF produces an equivalent instance $(T_1', T_2'; k')$ (the *kernel*) such that $k' \leqslant k$, and that the size of the leaf-label set for the trees T_1' and T_2' (i.e., the *kernel size*) is bounded by a function of the parameter k'. An algorithm A for the Max-AF problem is *fixed-parameter tractable* [19] if its running time is bounded by $f(k)n^{O(1)}$, where $f(k)$ is a function only depending on the parameter k but independent of the input size n.

The Max-AF problem for binary phylogenetic trees has been studied by a number of researchers in parameterized computation. Allen and Steel [9] developed a kernelization algorithm for the Max-AF problem on binary phylogenetic trees, which produces a kernel of size bounded by $c(k-1)$, where $c \leqslant 28$. This kernelization result implies a fixed-parameter tractable algorithm for the Max-AF problem. Hallett and McCartin [20] applied the two kernelization rules proposed in [9] and developed a fixed-parameter tractable algorithm of running time $O(4^k k^5) + p(n)$ for the Max-AF problem on unrooted binary phylogenetic trees, where $p(n)$ is a polynomial that is the running time of the kernelization algorithm. The algorithm in [20] was further improved by Whidden and Zeh [21], who developed a fixed-parameter tractable algorithm of running time $O(4^k k + n^3)$ or $O(4^k n)$ for the Max-AF problem on unrooted binary phylogenetic trees.

On the other hand, the computational complexity for the Max-AF problem on *multifurcating* (i.e., binary and non-binary) phylogenetic trees has not been studied as extensively as that on binary trees. A recent result by Linz and Semple [22] shows that computing the *hybridization number* of two rooted multifurcating trees is fixed-parameter tractable. However, the parameterized complexity of the Max-AF problem on multifurcating trees remained unknown. Whether the Max-AF problem on unrooted multifurcating trees is fixed-parameter tractable was posed specifically as an open problem by Hallett and McCartin [20] in 2007, and was posed again by Whidden et al. [23] for rooted and unrooted multifurcating trees in 2011.

Note that it is important to extend the study to multifurcating trees. For many biological data sets in practice [24], the reconstructed phylogenetic trees are in general not fully resolved. This may be due to either the tree reconstruction methods or the use of consensus trees. Evolutionary biologists often construct phylogenetic trees using methods that assign a measure of statistical support to each edge of the tree. Contracting edges with poor statistical support eliminates bipartitions that may be artifacts of the manner in which the tree was constructed, but the resulting trees will be multifurcating. Another component of tree comparisons is the ability to deal with multifurcating nodes in trees that are incompletely resolved. While most traditional phylogenetic methods produce completely resolved, strictly bifurcating trees, the statistical support at some of these bifurcating nodes may be extremely weak. Thus, a disagreement between two trees that is based on a weakly supported topological feature may be of no interest, and it is often preferable to collapse the corresponding feature into a multifurcating node. Finally, in broad phylogenomic studies most sets of putatively orthologous genes or proteins will not cover the entire reference tree, so it may be necessary to "project" the reference tree by removing the non-represented taxa before performing the comparison. *Polytomies*, alternatively called *multifurcations*, refer to vertices in a phylogenetic tree that have more than two direct descendants. A *hard polytomy* refers to an event during which an ancestral species gave rise to more than two offspring species at the same time [25], whereas a *soft polytomy* represents ambiguous evolutionary relationships as a result of insufficient information [26]. Recent research in biology and evolution [27] has shown that the majority of tree space in typical phylogenetic studies consists of multifurcating trees, and that multifurcations can introduce irresolvable problems for certain well-known tree-search algorithms (such as NNI algorithm [5]). An important problem that the biology literature [28] is attempting to solve is the SPR distance between unrooted multifurcating trees, whose complexity still remains open.

In this paper, we present an $O(4^k n^5)$-time fixed-parameter tractable algorithm for the Max-AF problem on unrooted multifurcating trees, resolving the open problem posed in [20,23]. It is interesting, and a bit surprising, to point out that our algorithm in fact follows very closely the ideas proposed by Hallett and McCartin [20], who presented a fixed-parameter tractable algorithm for the problem on unrooted binary trees and asked whether the problem on unrooted multifurcating trees is fixed-parameter tractable. Both algorithms in [20] and in our current paper are based on elimination of incompatible quartets in trees. However, to apply the techniques on multifurcating trees, a more complicated quartet topology, the star quartet, and many subtle topological structures in multifurcating trees that do not appear in binary trees must be carefully and correctly handled. From this point of view, the main contribution of our work is to provide new, thorough, and systematical methods in dealing with more complicated topological and combinatorial structures in non-binary trees and in the additional quartet structure.

We note that the optimization version of the Max-AF problem on multifurcating trees has also been studied. In particular, Rodrigues et al. [29] developed an approximation algorithm with rate $d+1$ for the problem, where d is the maximum number of children a vertex may have.

2 Max-MF, TBR distance, and quartets

For a phylogenetic forest F with a leaf-label set L, if we do not allow degree-1 non-leaf vertices, then each subset L' of L uniquely determines a subforest F' of F whose leaf-label set is L': the subforest F' is just the union of the paths in F that connect the pairs of leaves whose labels are in L'. The subforest F' will be called the *subforest induced by* L'.

We will assume that there are no degree-1 non-leaf vertices in a phylogenetic forest. During our process of phylogenetic forests, we may create such vertices—in this case, we simply remove them. Based on this observation, if we know that $F^* = \{T_1^*, T_2^*, \ldots, T_k^*\}$ is an agreement forest for two phylogenetic forests F and F' with a common leaf-label set L, then we can construct F^* from the forest F (or, similarly, from the forest F'), as follows. Let the leaf-label set for the tree T_i^* be L_i, for $i = 1, 2, \ldots, k$, where $L_1 \cup L_2 \cup \cdots \cup L_k = L$. We apply the following operation repeatedly, starting from the forest $F_1 = F$ and inductively assuming that there is no degree-1 non-leaf vertices in the forest F_i: (1) delete an edge in F_i to split a tree T_h in F_i into two subtrees T_h' and T_h'' such that no label set L_j contains elements in both T_h' and T_h''; (2) repeatedly remove degree-1 non-leaf vertices and let the resulting forest be F_{i+1}. Note that the above process is always possible because by definition, F^* is a subforest for the forest F. The above process stops at an integer i when F_i is homeomorphic to F^*. Therefore, the critical operation in constructing the agreement forest F^* for the two phylogenetic forests F and F' is to identify a set of edges in F whose removal splits the forest F into k subtrees with the label sets L_1, $L_2, \ldots, L_k$, respectively. Since removing a degree-1 non-leaf vertex can be implemented by first removing the incident edge then removing the degree-0 vertex, the agreement forest F^* can also be constructed from the forest F by first removing a proper set of edges to split F into a forest, then removing all non-leaf vertices whose degree has become 0.

Since there is one-to-one correspondence between the leaves of a phylogenetic forest F and its leaf-label set L, when there is no ambiguity, we sometimes may simply say that a label a in L is in a subtree of F if the leaf labeled a is in the subtree.

Allen and Steel [9] proved that the TBR distance between two unrooted binary phylogenetic trees is equal to the size of their MAF minus 1. The proof can be modified to give the same result for unrooted multifurcating trees. For completeness, we provide here details for this proof on multifurcating trees. We first extend the definition of TBR to multifurcating trees.

Definition 1. A *tree bisection and reconnection* (TBR) operation on a phylogenetic tree T is to remove an edge in T, resulting in two subtrees T_1 and T_2 in T, then reconnect T_1 and T_2 by a new edge e, where each end of the edge e can be on either a non-leaf vertex or the midpoint of an edge in T_1 and T_2.

Note that in the definition of the TBR operation on binary trees [9], it is required that the ends of the new edge e should be midpoints of edges in T_1 and T_2, to ensure that the resulting tree is a binary tree. For multifurcating trees, we relax this condition and allow the ends of the new edge e to join non-leaf vertices in T_1 and T_2.

The *TBR distance* $d_{\mathrm{tbr}}(T_1, T_2)$ between two phylogenetic trees T_1 and T_2 is the minimum number of TBR operations that transform T_1 into T_2. Clearly, $d_{\mathrm{tbr}}(T_1, T_2) = d_{\mathrm{tbr}}(T_2, T_1)$.

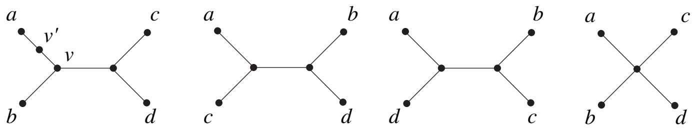
Theorem 1. For any two unrooted phylogenetic trees T_1 and T_2 with the same leaf-label set, $d_{\mathrm{tbr}}(T_1, T_2) = \textsc{maf}(T_1, T_2) - 1$, where $\textsc{maf}(T_1, T_2)$ is the size of an MAF for T_1 and T_2.

Proof. We prove the theorem by induction on $d_{\mathrm{tbr}}(T_1, T_2)$ and $\textsc{maf}(T_1, T_2)$.

For $d_{\mathrm{tbr}}(T_1, T_2) = 0$, we have $T_1 = T_2$, and T_1 (or T_2) itself clearly makes an MAF for T_1 and T_2. Thus, $\textsc{maf}(T_1, T_2) = |T_1| = 1$, and the theorem holds true. For the case $d_{\mathrm{tbr}}(T_1, T_2) = 1$, $T_1 \neq T_2$, and we can remove an edge e_1 in T_1, resulting in two subtrees T' and T'' of T_1, then reconnect T' and T'' with a new edge e_2 to obtain T_2. This implies that $\{T', T''\}$ is an agreement forest for T_1 and T_2. Since $T_1 \neq T_2$, $\{T', T''\}$ must be an MAF for T_1 and T_2, thus, $\textsc{maf}(T_1, T_2) = 2$. The theorem thus again holds true.

Now assume $d_{\mathrm{tbr}}(T_1, T_2) = d > 1$. Then there is a phylogenetic tree T_3 such that $d_{\mathrm{tbr}}(T_1, T_3) = d - 1$, and $d_{\mathrm{tbr}}(T_3, T_2) = 1$. By the inductive hypothesis, $\textsc{maf}(T_1, T_3) = d$, so T_1 and T_3 have an MAF $F = \{T_1', \ldots, T_d'\}$, where T_i' are disjointed subtrees in T_3 (and in T_1). From $d_{\mathrm{tbr}}(T_3, T_2) = 1$, the tree T_2 can be obtained from the tree T_3 by first removing an edge e_1 then reconnecting the two resulting subtrees by a new edge. This means that $T_3 \setminus \{e_1\}$ is a subforest of T_2. Since F is a subforest of T_3, $F \setminus \{e_1\}$ must be a subforest of T_2. Therefore, $F \setminus \{e_1\}$ is an agreement forest for T_1 and T_2 (note that F is also a subforest of T_1). Since $F \setminus \{e_1\}$ consists of at most $d + 1$ trees, we get $\textsc{maf}(T_1, T_2) \leqslant d + 1$, which shows that $d_{\mathrm{tbr}}(T_1, T_2) \geqslant \textsc{maf}(T_1, T_2) - 1$.

To see the other direction, let $\textsc{maf}(T_1, T_2) = d + 1$, where $d > 1$. Then T_1 and T_2 have an MAF



(a) (b) (c) (d)

Figure 1 Four topological structures for the quartet $Q_F(a,b,c,d)$.

$F=\{T_1',\ldots,T_{d+1}'\}$ with $d+1$ trees. Since $T_1',\ldots,T_{d+1}'$ are disjoint in T_1, there must be a simple path P_1 in T_1 that connects two trees in F such that no internal vertex of P_1 is in F. Without loss of generality, suppose that the path P_1 has its two end-vertices v_d and v_{d+1} in T_d' and T_{d+1}', respectively. We can construct a new tree T_3 as follows: first add a new edge e_1 between v_d and v_{d+1} in the tree T_2, which causes a unique cycle C in $T_2\cup\{e_1\}$. Since the edge e_1 crosses between two trees in F, there is an edge e_2 in the cycle C that is not in F. Now the tree T_3 is obtained from $T_2\cup\{e_1\}$ by removing the edge e_2. Note that $T_d''=T_d'\cup T_{d+1}'\cup\{e_1\}$ is a subtree in both T_1 and T_3 (for T_1, more precisely, T_d'' is homeomorphic to the subtree $T_d'\cup T_{d+1}'\cup\{P_1\}$). Therefore, $\{T_1',\ldots,T_{d-1}',T_d''\}$ is an agreement forest for T_1 and T_3, i.e., $\textsc{maf}(T_1,T_3)\leqslant d$. By the inductive hypothesis, $d_{\rm tbr}(T_1,T_3)\leqslant d-1$. Observing that the tree T_2 can be obtained from the tree T_3 by first removing the edge e_1 then adding the edge e_2, we conclude immediately that

$$d_{\rm tbr}(T_1,T_2)\leqslant d_{\rm tbr}(T_1,T_3)+1\leqslant (d-1)+1=d=\textsc{maf}(T_1,T_2)-1.$$

This completes the proof of the theorem.

The following definition will be critical in our discussion.

Definition 2. Let F be a phylogenetic forest with a leaf-label set L. For four elements a, b, c, and d in the set L that are in the same tree in F, define the *quartet* $Q_F(a,b,c,d)$ to be the unique 4-leaf subtree in F that is induced by the label subset $\{a,b,c,d\}$.

Up to homeomorphism, there are exactly four different structures for a quartet $Q_F(a,b,c,d)$, as given in Figure 1 (a)–(d), which will be named $ab|cd$, $ac|bd$, $ad|bc$, and $(abcd)$ structures, respectively. The $ab|cd$, $ac|bd$, and $ad|bc$ structures will be called *butterfly structures*, and the $(abcd)$ structure will be called a *star structure*. Note that a quartet by itself is also a phylogenetic tree. We say a quartet Q is *in a phylogenetic forest* F if Q is homeomorphic to a subtree in F.

Let F be a phylogenetic forest. Following the notations in [20], for a butterfly quartet $ab|cd$ in F, the unique vertex v in F that separates the leaves labeled by a, b, and c, respectively, is the *junction for* a (also the junction for b). The junction for c and d is defined similarly. For a star quartet $(abcd)$ in F, we similarly define the *junction for* a (also the junction for b, c, and d) to be the unique vertex in F that separates pairwise the four leaves labeled a, b, c, and d, respectively. For a leaf labeled a in a quartet Q in a phylogenetic forest F, if v is the junction for a, we will denote by $F^{(a,Q)}$ the connected component of $F\setminus\{v\}$ that contains a. All leaves in a quartet that share the same junction are called *siblings*. Therefore, each leaf in a butterfly quartet has a single sibling, while each leaf in a star quartet has three siblings.

Definition 3. Let F be a phylogenetic forest and let T be a phylogenetic tree with the same label set. A quartet $Q=Q_F(a,b,c,d)$ in F is a *minimum incompatible quartet for* T (briefly, an *F-MIQ for* T) if (1) Q is not a quartet in T; and (2) for each $x\in\{a,b,c,d\}$, if in the label set $\{a,b,c,d\}$ we replace the label of a sibling of x by any label in $F^{(x,Q)}$, then the resulting label set induces a quartet in the forest F that is also in the tree T.

Note that if a phylogenetic forest F contains a quartet $Q=Q_F(a,b,c,d)$ that is not in a phylogenetic tree T, then F must contain an F-MIQ for T, which can be constructed from the quartet Q as follows. If $Q=Q_F(a,b,c,d)$ is not an F-MIQ, then since Q is not in T, by definition, there must be an $x\in\{a,b,c,d\}$ and a label x' in $F^{(x,Q)}$ such that the quartet $Q'=Q_F(a',b',c',d')$ is also not in T, where $\{a',b',c',d'\}$ is the set $\{a,b,c,d\}$ with a sibling of x being replaced by x'. If Q' is still not an F-MIQ for T, then we repeat the above process on Q'. Since the total number of vertices in $F^{(a',Q')}\cup F^{(b',Q')}\cup F^{(c',Q')}\cup F^{(d',Q')}$ is strictly smaller than that in $F^{(a,Q)}\cup F^{(b,Q)}\cup F^{(c,Q)}\cup F^{(d,Q)}$, the above process must eventually terminate with a quartet in F that is an F-MIQ for T. As a consequence, if the phylogenetic forest F contains no

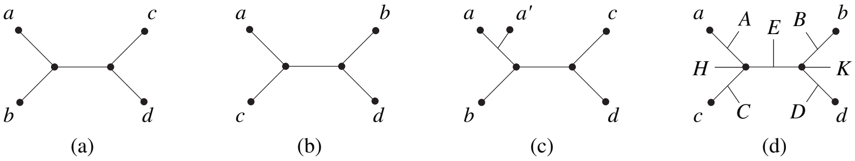


Figure 2 Q_F and Q_T are butterfly quartets. (a) Quartet Q in F; (b) quartet Q' in T; (c) label a' in F; (d) possible positions for a' in T.

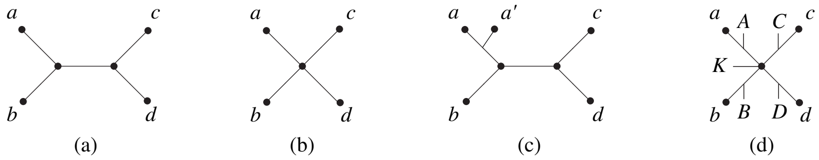


Figure 3 Q is a butterfly and Q' is a star. (a) Quartet Q in F; (b) quartet Q' in T; (c) label a' in F; (d) possible positions for a' in T.

F-MIQ for the phylogenetic tree T, then every quartet in F is in the tree T.

Definition 4. Let F be a phylogenetic forest and let $Q = Q_F(a,b,c,d)$ be a quartet in F. An edge e in F is a *prong* for the quartet Q if for some $x \in \{a,b,c,d\}$, the edge e is incident to the junction v for x and is on the path from v to x in F.

Note that removing a prong in a quartet Q will split the tree containing Q into two subtrees, one containing one label in Q and the other containing three labels in Q.

3 On minimum incompatible quartets

In this section, we study the structures of minimum incompatible quartets. We will fix a phylogenetic forest F and a phylogenetic tree T with the same leaf-label set L.

Assume that the forest F contains a quartet that is not in the tree T. By our discussion in the previous section, F has an F-MIQ Q for T. We discuss how we can delete edges in the forest F so that F contains no F-MIQ for T. We start with the following lemma.

Lemma 1. Let $Q = Q_F(a,b,c,d)$ be a quartet in the forest F that is an F-MIQ for the tree T. Let x be any label in $\{a,b,c,d\}$, and let x' be any label in $F^{(x,Q)}$. Then for the quartet $Q' = Q_T(a,b,c,d)$ induced by $\{a,b,c,d\}$ in the tree T, the label x' is in $T^{(x,Q')}$.

Proof. Since Q is an F-MIQ for T, the two quartets Q and Q' are different. Because of the symmetry, we can assume, without loss of generality, that $x = a$, and let a' be in $F^{(a,Q)}$.

Case 1. The quartet $Q = Q_F(a,b,c,d)$ in F is the butterfly quartet $ab|cd$, and the quartet $Q' = Q_T(a,b,c,d)$ in T is the butterfly quartet $ac|bd$. See Figure 2 (a) and (b).

Since a' is a label in $F^{(a,Q)}$, the tree in F induced by the labels $\{a,a',b,c,d\}$ must be the case as shown in Figure 2(c). Now consider the position of the label a' in the tree T. There are seven different possible positions, call them positions A, B, C, D, E, H, and K, respectively, in the tree T where the label a' can be located, as shown in Figure 2(d). If the label a' is in one of the positions B, D, E, and K, then the quartet $Q_T(a,a',c,d)$ in T is the butterfly quartet $ac|a'd$. If the label a' is in position C, then the quartet $Q_T(a,a',c,d)$ in T is $ad|a'c$. If the label a' is in position H, then the quartet $Q_T(a,a',c,d)$ in T is the star quartet $(aa'cd)$. Therefore, when a' is in any of the six positions B, C, D, E, H, and K, the butterfly quartet $aa'|cd$ in the forest F is not in the tree T. But this contradicts our assumption that the quartet Q is an F-MIQ for T, which, by definition, requires that if we replace label b by the label a' in $F^{(a,Q)}$, then the resulting quartet $Q_F(a,a',c,d)$ in F should be in the tree T. This proves that the label a' can only be in position A, i.e., a' must be in $T^{(a,Q')}$.

Case 2. The quartet $Q = Q_F(a,b,c,d)$ in F is the butterfly quartet $ab|cd$, and the quartet $Q' = Q_T(a,b,c,d)$ in T is the star quartet $(abcd)$. See Figure 3 (a) and (b).

The proof goes very similar to that for Case 1. Because a' is in $F^{(a,Q)}$, the quartet $Q_F(a,a',c,d)$ in F must be $aa'|cd$ (see Figure 3(c)). On the other hand, it is easy to verify that if a' is in any of

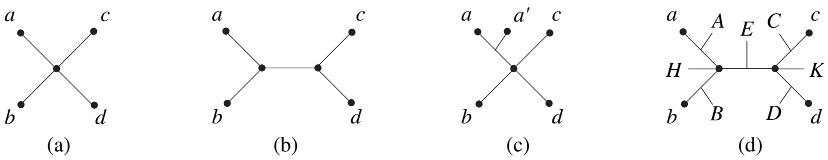


Figure 4 Q is a star and Q' is a butterfly. (a) Quartet Q in F; (b) quartet Q' in T; (c) label a' in F; (d) possible positions for a' in T.

the positions B, C, D, and K in the tree T in Figure 3(d), then the quartet $Q_T(a, a', c, d)$ induced by $\{a, a', c, d\}$ in the tree T is not $aa'|cd$. Therefore, in these cases, the quartet $Q_F(a, a', c, d)$ in F is not in T, contradicting the assumption that $Q = Q_F(a, b, c, d)$ is an F-MIQ for T. Thus, a' must be in position A in T in Figure 3(d), i.e., a' is in $T^{(a,Q')}$.

Case 3. The quartet $Q = Q_F(a, b, c, d)$ in F is the star quartet $(abcd)$, and the quartet $Q' = Q_T(a, b, c, d)$ in T is the butterfly quartet $ab|cd$. See Figure 4 (a) and (b).

The proof again goes similar to that for Case 1, but gets slightly more complicated. First consider the quartet $Q_1 = Q_F(a, a', c, d)$ in F (i.e., replace the sibling b of a by a' in $F^{(a,Q)}$). The quartet Q_1 is the butterfly quartet $aa'|cd$ (see Figure 4(c)). If a' is in one of the positions C, D, or K in the tree T as shown in Figure 4(d), then the quartet $Q_T(a, a', c, d)$ induced by $\{a, a', c, d\}$ in the tree T is not $aa'|cd$. Thus, in these cases, the quartet $Q_1 = Q_F(a, a', c, d)$ is not in the tree T, which contradicts the assumption that $Q = Q_F(a, b, c, d)$ is an F-MIQ for T. Now consider the quartet $Q_2 = Q_F(a, a', b, d)$ in F (i.e., replace the sibling c of a by a' in $F^{(a,Q)}$). The quartet Q_2 is the butterfly quartet $aa'|bd$ (see Figure 4(c)). If a' is in one of the positions B, E, or H in the tree T as shown in Figure 4(d), then the quartet $Q_T(a, a', b, d)$ induced by $\{a, a', b, d\}$ in the tree T is not $aa'|bd$. Thus, in these cases, the quartet $Q_2 = Q_F(a, a', b, d)$ is not in the tree T, which again contradicts the assumption that $Q = Q_F(a, b, c, d)$ is an F-MIQ for T. Thus, again a' must be in position A in the tree T in Figure 4(d), i.e., a' is in $T^{(a,Q')}$.

Because of symmetry, Cases 1–3 include all possible cases. Thus, the lemma is proved.

In the next lemma, we show that in an F-MIQ $Q = Q_F(a, b, c, d)$ for the tree T, if we replace a label $x \in \{a, b, c, d\}$ by any label x' in $F^{(x,Q)}$, we still get an F-MIQ for T.

Lemma 2. Let $Q_F = Q_F(a, b, c, d)$ be a quartet in the forest F that is an F-MIQ for the tree T, and let $x \in \{a, b, c, d\}$. Then for any label x' in $F^{(x,Q_F)}$, the quartet $Q'_F = Q_F(a', b, c, d)$ in F is also an F-MIQ for T.

Proof. Again by symmetry, we can assume that $x = a$, and let a' be in $F^{(a,Q_F)}$. Let $Q_T = Q_T(a, b, c, d)$ and $Q'_T = Q_T(a', b, c, d)$ be the quartets in the tree T induced by the label sets $\{a, b, c, d\}$ and $\{a', b, c, d\}$, respectively. By Lemma 1, the label a' must be in $T^{(a,Q_T)}$, i.e., in position A in Figures 2(d), 3(4), and 4(4). Therefore, the quartet $Q'_F = Q_F(a', b, c, d)$ in F can never be in the tree T: (1) in Figure 2, $Q'_F = a'b|cd$ in F while $Q'_T = a'c|bd$ in T; (2) in Figure 3, $Q'_F = a'b|cd$ in F while $Q'_T = (a'bcd)$ in T; and (3) in Figure 4, $Q'_F = (a', b, c, d)$ in F while $Q'_T = a'b|cd$ in T. Therefore, the quartet $Q'_F = Q_F(a', b, c, d)$ in F is not in the tree T.

In order to prove that Q'_F is an F-MIQ for T, we also need to prove that for any $x \in \{a', b, c, d\}$, if we replace in the set $\{a', b, c, d\}$ a sibling of x by a label in $F^{(x,Q'_F)}$, the resulting label set induces a quartet in F that is also in T. First note that in the forest F, the junction for a' in the quartet $Q'_F = Q_F(a', b, c, d)$ and the junction for a in the quartet $Q_F = Q_F(a, b, c, d)$ are the same vertex in F, and that the junctions for b, c, and d in the quartet Q_F are also the junctions for b, c, and d in the quartet Q'_F, respectively (see Figures 2(c), 3(3), and 4(3)). Therefore,

$$F^{(a,Q_F)} = F^{(a',Q'_F)}, \quad F^{(b,Q_F)} = F^{(b,Q'_F)}, \quad F^{(c,Q_F)} = F^{(c,Q'_F)}, \quad \text{and} \quad F^{(d,Q_F)} = F^{(d,Q'_F)}.$$

Similarly, and by Lemma 1 (see Figures 2(d), 3(4), and 4(4), and note that the label a' is in position A), we also have

$$T^{(a,Q_T)} = T^{(a',Q'_T)}, \quad T^{(b,Q_T)} = T^{(b,Q'_T)}, \quad T^{(c,Q_T)} = T^{(c,Q'_T)}, \quad \text{and} \quad T^{(d,Q_T)} = T^{(d,Q'_T)}.$$

Now suppose that we pick an a'' in $F^{(a',Q'_F)} = F^{(a,Q_F)}$. By Lemma 1, a'' is in $T^{(a,Q_T)} = T^{(a',Q'_T)}$. Thus, all three labels a, a', and a'' are in $F^{(a',Q'_F)} = F^{(a,Q_F)}$ in the forest F, and are in $T^{(a,Q_T)} = T^{(a',Q'_T)}$ in the tree T. Therefore, if we let y be a sibling of a' in Q'_F, then y is also a sibling of a in Q_F. Now let $\{z_1, z_2\} = \{a', b, c, d\} \setminus \{a', y\}$. It is easy to verify that if we replace the label a in quartet $Q_F(a, a'', z_1, z_2)$ in F by label a', we will get the quartet $Q_F(a', a'', z_1, z_2)$ in F; and if we replace the label a in quartet $Q_T(a, a'', z_1, z_2)$ in T by label a', we will get the quartet $Q_T(a', a'', z_1, z_2)$ in T. Because Q_F is an F-MIQ for T, the quartet $Q_F(a, a'', z_1, z_2)$ in F and the quartet $Q_T(a, a'', z_1, z_2)$ in T must be the same. In consequence, the quartet $Q_F(a', a'', z_1, z_2)$ in F and the quartet $Q_T(a', a'', z_1, z_2)$ in T must be the same. This shows that in the set $\{a', b, c, d\}$ if we replace a sibling y of a' by a label a'' in $F^{(a', Q'_F)}$, then the resulting label set will induce a quartet in F that is also a quartet in T.

In a similar (actually a bit simpler) way, we can verify that for each label z in $\{b, c, d\}$, if we replace in the label set $\{a', b, c, d\}$ a sibling of z by a label z' in $F^{(z, Q'_F)}$, then the resulting label set will induce a quartet in F that is also a quartet in T.

This completes the proof that the quartet $Q'_F = Q_F(a', b, c, d)$ in F is an F-MIQ for T.

Corollary 1. Let $Q_F = Q_F(a, b, c, d)$ be a quartet in the forest F that is an F-MIQ for the tree T, and for each $x \in \{a, b, c, d\}$, let x' be any label in $F^{(x, Q_F)}$. Then the quartet $Q'_F = Q_F(a', b', c', d')$ in F is an F-MIQ for T.

Proof. For a' in $F^{(a, Q_F)}$, by Lemma 2, $Q'_F = Q_F(a', b, c, d)$ is an F-MIQ for T. Note that the junction for a in Q_F and the junction for a' in Q'_F are the same vertex, and the labels b, c, and d also have the same junctions for the two quartets. Therefore, if we apply Lemma 2 on the quartet Q'_F and the label b', we will get that the quartet $Q''_F(a', b', c, d)$ is an F-MIQ for T. Repeat this analysis on labels c' and d', we will eventually prove that the quartet $Q'_F = Q_F(a', b', c', d')$ in F is an F-MIQ for T.

Recall that a *prong* for a quartet $Q_F(a, b, c, d)$ is an edge that is incident to the junction v of a label x in $\{a, b, c, d\}$ and is on the path from v to x. The following theorem is critical for our main algorithm for the Max-AF problem. Also recall that an agreement forest F^* of two forests F_1 and F_2 can be constructed by first removing a set E_F of edges in F_1 then removing non-leaf vertices whose degree has become 1. Therefore, if an edge e in the forest F_1 is in the set E_F, then we will say that the edge e is *not in the agreement forest* F^*.

Theorem 2. Let Q be a quartet in the forest F that is an F-MIQ for the tree T. Then for any agreement forest F^* for F and T, at least one of the prongs for Q is not in F^*.

Proof. First, suppose that F-MIQ $Q = Q_F(a, b, c, d)$ for T is a star quartet $(abcd)$. Let the unique junction for the labels a, b, c, and d in Q be the vertex v, and let $[v, u_a]$, $[v, u_b]$, $[v, u_c]$, and $[v, u_d]$ be the four prongs for Q. Assume the contrary that all four prongs are in the agreement forest F^*. Since the forest F^* contains no degree-1 non-leaf vertices, by extending the four edges $[v, u_a]$, $[v, u_b]$, $[v, u_c]$, and $[v, u_d]$, from the vertex v, in F^*, we will be able to find labels a' in $F^{(a,Q)}$, b' in $F^{(b,Q)}$, c' in $F^{(c,Q)}$, and d' in $F^{(d,Q)}$ such that the quartet $Q_F(a', b', c', d')$ in F is in the agreement forest F^*, which implies that the quartet $Q_F(a', b', c', d')$ in F is also in the tree T. But this is a contradiction because by our assumption the quartet $Q = Q_F(a, b, c, d)$ is an F-MIQ for T, and by Corollary 1, the quartet $Q_F(a', b', c', d')$ in F should also be an F-MIQ for T, thus, it should not be in the tree T.

Now suppose that the F-MIQ $Q = Q_F(a, b, c, d)$ is the butterfly quartet $ab|cd$. Let the junction for a and b be v_1, and let the junction for c and d be v_2. Let $[v_1, u_a]$, $[v_1, u_b]$, $[v_2, u_c]$, and $[v_2, u_d]$ be the four prongs for Q. Again assume the contrary that all four prongs are in the agreement forest F^*. There are two subcases.

Subcase 2.1. All edges on the path $P_F[v_1, v_2]$ from vertex v_1 to vertex v_2 in Q are also in F^*. Then the path $P_F[v_1, v_2]$ plus the four prongs are contained in a single tree in F^*. Similar to the above proof, by extending the prongs, we can get labels a' in $F^{(a,Q)}$, b' in $F^{(b,Q)}$, c' in $F^{(c,Q)}$, and d' in $F^{(d,Q)}$ such that the quartet $Q_F(a', b', c', d')$ in F is in the agreement forest F^* so is also in the tree T, contradicting the assumption that $Q_F(a', b', c', d')$ is an F-MIQ for T.

Subcase 2.2. An edge e in the path $P_F[v_1, v_2]$ is not in F^*. Then because the four prongs are all in F^*, by extending the two prongs incident to v_1 in F^*, we can get two labels a' in $F^{(a,Q)}$ and b' in $F^{(b,Q)}$ such that the path $P_F[a', b']$ from a' to b' in F^* is entirely in F^*; and by extending the two prongs incident to v_2 in F^*, we can get two labels c' in $F^{(c,Q)}$ and d' in $F^{(d,Q)}$ such that the path $P_F[c', d']$ from

c' to d' is entirely in F^*. Moreover, since the edge e separates the junctions v_1 and v_2, which are on the paths $P_F[a',b']$ and $P_F[c',d']$, respectively, the two paths $P_F[a',b']$ and $P_F[c',d']$ are vertex-disjoint. Because $P_F[a',b']$ and $P_F[c',d']$ are two vertex-disjoint paths in F^* and because F^* is a subforest of T, the path $P_T[a',b']$ that connects a' and b' in T and the path $P_T[c',d']$ that connects c' and d' in T should also be vertex-disjoint. However, this derives a contradiction, as follows. By Corollary 1, the quartet $Q_F(a',b',c',d')$ in F is an F-MIQ for T. Therefore, the quartet $Q_T(a',b',c',d')$ in the tree T is different from the quartet $Q_F(a',b',c',d')$ in the forest F. Since the quartet $Q = Q_F(a,b,c,d)$ is a butterfly quartet $ab|cd$, the quartet $Q_F(a',b',c',d')$ in F is the butterfly quartet $a'b'|c'd'$. Therefore, the quartet $Q_T(a',b',c',d')$ in the tree T should be either the butterfly quartet $a'c'|b'd'$, or the butterfly quartet $a'd'|b'c'$, or the star quartet $(a'b'c'd')$. However, in any of these cases, the paths $P_T[a',b']$ and $P_T[c',d']$ in the tree T cannot be vertex-disjoint, which is the contradiction.

This completes the proof of the theorem.

4 On conflicting edges

In this section, we consider the phylogenetic forest F and the phylogenetic tree T with the same leaf-label set L where F has no F-MIQ for T. Recall that this means that every quartet in the forest F is also in the tree T. For a vertex v, we will denote by $\deg(v)$ the degree of v.

Let F be the phylogenetic forest with a leaf-label set L. We say that three labels a, b, and c in L *meet* at a vertex v in F if a, b, and c are in the same tree in F, and if v is the degree-3 vertex in the subtree of F that is induced by $\{a,b,c\}$. Note that the vertex v where the three labels meet must be a non-leaf vertex.

Lemma 3. Let F be a phylogenetic forest and let T be a phylogenetic tree with the same leaf-label set such that F contains no F-MIQ for T. Then for any non-leaf vertex v with $\deg(v) \geqslant 3$ in the forest F, there is a unique vertex v' in T such that if three labels a, b, and c meet at v in the forest F, then they meet at v' in the tree T.

Proof. Let the non-leaf vertex v with $\deg(v) \geqslant 3$ be in a tree T_1 in F. Note that three labels meet at v in F if and only if they are from three different connected components of $T_1 \setminus \{v\}$.

Suppose that the three labels a, b, and c meet at v in F and meet at v' in T. Consider a fourth label $a' \notin \{a,b,c\}$ in the tree T_1 such that a', b, and c are from three different connected components of $T_1 \setminus \{v\}$. Then, a', b, and c also meet at v in F. We show that a', b, c must meet at v' in T. The four labels a, a', b, and c make a quartet $Q = Q_F(a,a',b,c)$ in F: if a and a' are in the same connected component of $T_1 \setminus \{v\}$, then Q is the butterfly quartet $aa'|bc$, and if a and a' are from different connected components of $T_1 \setminus \{v\}$, then Q is the star quartet $(aa'bc)$. Since F contains no F-MIQ for T, the quartet Q is also in the tree T. Now by examining the structures of the butterfly quartet $aa'|bc$ and of the star quartet $(aa'bc)$ in the tree T, we can easily verify that in the tree T the three labels a', b, and c must meet at the same vertex where the three labels a, b, and c meet, i.e., the three labels a', b, and c also meet at vertex v' in T.

Now consider arbitrary three labels a', b', and c' that meet at v in F. Since the three labels a, b, and c are from three different connected components of $T_1 \setminus \{v\}$, we can assume, without loss of generality, that the three label a', b, and c are from three different connected components of $T_1 \setminus \{v\}$, thus, they meet at v in F. If $a' = a$, then clearly a', b, and c also meet at v' in T. If $a' \neq a$, then the previous paragraph shows that a', b, and c also meet at v' in T. Repeating this process with the label triples (a',b,c) and (a',b',c), and assuming that a', b', and c are in three different connected components in $T_1 \setminus \{v\}$, we can derive that the three labels a', b', and c meet at v' in the tree T. Finally, we repeat the process again on the triple sets (a',b',c) and (a',b',c'), which derives that the labels a', b', and c' also meet at v' in the tree T.

This proves that any three labels meeting at v in F must meet at v' in T.

The following definition is an extension of a definition in [20] on binary phylogenetic forests. Our definition is applicable on general phylogenetic forests. Recall that we assumed that a phylogenetic tree does not have non-leaf vertices of degree less than 2.

Definition 5. Let F be a phylogenetic forest and let T be a phylogenetic tree with the same leaf-label set, such that F contains no F-MIQ for T. The *β-mapping* from vertices v with $\deg(v) \neq 2$ in F to vertices in T is defined as follows:

(1) If v is a leaf in F, then $\beta(v)$ is the leaf in T with the same label;

(2) If v is a non-leaf vertex with $\deg(v) \neq 2$, then $\beta(v)$ is the vertex v' in T such that if three labels meet at v in F, then the labels also meet at v' in T.

By Lemma 3, the β-mapping is well-defined.

Let F be a phylogenetic forest. A path with two end-vertices v_1 and v_2 in F will be denoted by $P_F[v_1, v_2]$. An *internal vertex* of a path is a vertex on the path that is not an end-vertex.

Lemma 4. Let F be a phylogenetic forest and let T be a phylogenetic tree with the same leaf-label set, such that F contains no F-MIQ for T. Let v and w be two vertices in the same tree in F such that $v \neq w$, $\deg(v) \neq 2$ and $\deg(w) \neq 2$. Then in the tree T, $\beta(v) \neq \beta(w)$.

Proof. If both v and w are leaves in F, or if one of them is a leaf and the other is a non-leaf, then by definition, $\beta(v) \neq \beta(w)$. Thus, we assume that both v and w are non-leaf vertices, with $\deg(v) \geqslant 3$ and $\deg(w) \geqslant 3$.

Since v and w are in the same tree in F, there is a unique path $P_F[v, w]$ between v to w in F. Since $\deg(v) \geqslant 3$ and $\deg(w) \geqslant 3$, we can find four labels a, b, c, and d such that the paths $P_F[a, v]$, $P_F[b, v]$, $P_F[c, w]$, $P_F[d, w]$, and $P_F[v, w]$ make a butterfly quartet $Q = ab|cd$ in F, in which v is the junction for a and b, and w is the junction for c and d. Since F contains no F-MIQ for T, the quartet $Q = ab|cd$ is also in T. Since a, b, and c meet at v, and a, c, and d meet at w in F, we must have a, b, and c meeting at $\beta(v)$, and a, c, and d meeting at $\beta(w)$ in T. Comparing this with the quartet $Q = ab|cd$ in T, we conclude that $\beta(v)$ is the junction for a and b, and $\beta(w)$ is the junction for c and d for the quartet Q in T. Therefore, $\beta(v)$ cannot be $\beta(w)$ – otherwise, the quartet Q in T would be a star quartet $(abcd)$.

Lemma 5. Let F be a phylogenetic forest and let T be a phylogenetic tree with the same leaf-label set, such that F contains no F-MIQ for T. Let v_1, v_2, and v_3 be three vertices in F whose degrees are not equal to 2 such that the vertex v_2 is on the path $P_F[v_1, v_3]$ in the forest F, then the vertex $\beta(v_2)$ is on the path $P_T[\beta(v_1), \beta(v_3)]$ in the tree T.

Proof. If $v_2 = v_1$ or $v_2 = v_3$, then the lemma obviously holds true. Thus, we can suppose that v_2 is an internal vertex of the path $P_F[v_1, v_3]$ with $\deg(v_2) \geqslant 3$.

If both v_1 and v_3 are leaves, labeled a_1 and a_3, respectively, then by extending from v_2, we can find a label a_2 such that the labels a_1, a_2, and a_3 meet at v_2 in F. By definition, the labels a_1, a_2, and a_3 meet at $\beta(v_2)$ in T. Thus, the vertex $\beta(v_2)$ is on the path $P_T[\beta(v_1), \beta(v_3)] = P_T[a_1, a_2]$ in the tree T.

If v_1 is a leaf labeled a_1 and v_3 is a non-leaf vertex with $\deg(v_3) \geqslant 3$, then by extending from v_3 we can find two labels a_3 and a_4, and by extending from v_2 we can find a fourth label a_2 such that a_1, a_2, a_3, and a_4 make a butterfly quartet $a_1a_2|a_3a_4$, where the junction for a_1 and a_2 is v_2, and the junction for a_3 and a_4 is v_3. Since F contains no F-MIQ, $a_1a_2|a_3a_4$ is also a quartet in T, where the junction for a_1 and a_2 is $\beta(v_2)$, and the junction for a_3 and a_4 is $\beta(v_3)$. This gives immediately that vertex $\beta(v_2)$ is on the path $P_T[\beta(v_1), \beta(v_3)] = P_T[a_1, \beta(v_3)]$ in T.

Finally, suppose that both v_1 and v_3 are non-leaf vertices. Extending from v_1 and v_3, respectively, we can find two labels a_1 and a_3 such that the path $P_F[a_1, a_3]$ contains the path $P_F[v_1, v_3]$. Since v_2 is on the path $P_F[v_1, v_3]$, we have the following relations in the forest F:

(1) v_1, v_2, and v_3 are on the path $P_F[a_1, a_3]$;

(2) v_1 is on the path $P_F[a_1, v_2]$, and v_3 is on the path $P_F[v_2, a_3]$.

Note for each of the paths $P_F[a_1, a_3]$, $P_F[a_1, v_2]$, and $P_F[v_2, a_3]$, at least one end-vertex is a leaf. Therefore, we can use what we have proved above, and get the following relations in T:

(1') $\beta(v_1)$, $\beta(v_2)$, and $\beta(v_3)$ are on the path $P_T[a_1, a_3]$;

(2') $\beta(v_1)$ is on the path $P_T[a_1, \beta(v_2)]$, and $\beta(v_3)$ is on the path $P_T[\beta(v_2), a_3]$.

This gives immediately that the vertex $\beta(v_2)$ is on the path $P_T[\beta(v_1), \beta(v_3)]$ in T.

A *chain* C in the forest F is a path with at least two vertices such that the end-vertices of C have degree $\neq 2$ and all internal vertices of C have degree 2. Note that if two distinct chains in F intersect, then they must intersect at a vertex that is the end-vertex of both chains. Moreover, two distinct chains can intersect at no more than one vertex.

Definition 6. Let F be a phylogenetic forest and let T be a phylogenetic tree with the same leaf-label set, such that F contains no F-MIQ for T. Two distinct chains $P_F[v_1, v_2]$ and $P_F[w_1, w_2]$ in F are *conflicting on* T if the number of vertices in F at which the chains $P_F[v_1, v_2]$ and $P_F[w_1, w_2]$ intersect is strictly smaller than the number of vertices in T at which the paths $P_T[\beta(v_1), \beta(v_2)]$ and $P_T[\beta(w_1), \beta(w_2)]$ intersect.

Since two distinct chains in F can intersect at no more than one vertex, two conflicting chains in F must be in one of the following two cases: (1) the chains $P_F[v_1, v_2]$ and $P_F[w_1, w_2]$ in F do not intersect but the paths $P_T[\beta(v_1), \beta(v_2)]$ and $P_T[\beta(w_1), \beta(w_2)]$ in T intersect; or (2) the chains $P_F[v_1, v_2]$ and $P_F[w_1, w_2]$ in F intersect at an end-vertex and the paths $P_T[\beta(v_1), \beta(v_2)]$ and $P_T[\beta(w_1), \beta(w_2)]$ in T intersect at more than one vertices.

Theorem 3. Let F be a phylogenetic forest and let T be a phylogenetic tree with the same leaf-label set, such that F contains no F-MIQ for T. Let C_1 and C_2 be two chains in F that are conflicting on T. Then no agreement forest for F and T can contain both chains C_1 and C_2.

Proof. Let $C_1 = P_F[v_1, v_2]$ and $C_2 = P_F[w_1, w_2]$. We split the proof for three different cases.

Case 1. the chains C_1 and C_2 share a common end-vertex $v_2 = w_2$.

Then v_2 ($= w_2$) is the only intersecting vertex for the chains C_1 and C_2, and v_2 is on the path $P_F[v_1, w_1]$ in the forest F. By Lemma 5, the vertex $\beta(v_2)$ ($= \beta(w_2)$) is on the path $P_T[\beta(v_1), \beta(w_1)]$ in the tree T. However, this would imply that $\beta(v_2)$ is the only vertex in T at which the paths $P_T[\beta(v_1), \beta(v_2)]$ and $P_T[\beta(w_1), \beta(w_2)]$ intersect, contradicting the assumption that the two chains C_1 and C_2 are conflicting on T. Therefore, this case is impossible.

Case 2. The chains C_1 and C_2 are in the same tree in F but do not intersect.

In this case, there is a unique path in F that connects an end-vertex in C_1 and an end-vertex in C_2 without passing through any other vertices in the chains. Without loss of generality, let this path be $P_F[v_2, w_2]$. Then in the forest F, we have, v_2 and w_2 on the path $P_F[v_1, w_1]$; v_2 on the path $P_F[v_1, w_2]$, and w_2 on the path $P_F[v_2, w_1]$. By Lemma 5, in the tree T, we must have $\beta(v_2)$ and $\beta(w_2)$ on the path $P_T[\beta(v_1), \beta(w_1)]$; $\beta(v_2)$ on the path $P_T[\beta(v_1), \beta(w_2)]$, and $\beta(w_2)$ on the path $P_T[\beta(v_2), \beta(w_1)]$. Moreover, by Lemma 4, $\beta(v_2) \neq \beta(w_2)$. Combining all these, we conclude immediately that the paths $P_T[\beta(v_1), \beta(v_2)]$ and $P_T[\beta(w_1), \beta(w_2)]$ do not intersect in T, again contradicting the assumption that the chains C_1 and C_2 are conflicting on T. Therefore, this case is also impossible.

Case 3. The chains C_1 and C_2 are in two different trees in the forest F.

Assuming the contrary, suppose that the two chains C_1 and C_2 in F are both contained in an agreement forest F^* for F and T. Note that the forest F^* can be obtained from F by deleting a proper subset of edges in F. Therefore, if C_1 and C_2 are in different trees in F, then they must be also in different trees T_1^* and T_2^*, respectively, in F^*. Here we should be more careful: the end-vertices of C_1 and C_2 may become degree 2 in F^*, so C_1 and C_2 may no longer be chains in F^*. On the other hand, by extending C_1 and C_2 in the trees T_1^* and T_2^*, respectively, we can always find two labels a_1 and b_1 in T_1^*, and two labels a_2 and b_2 in T_2^*, such that C_1 is contained in the path $P_{F^*}[a_1, b_1]$ and C_2 is contained in the path $P_{F^*}[a_2, b_2]$. Since the paths $P_{F^*}[a_1, b_1]$ and $P_{F^*}[a_2, b_2]$ are in different trees in the agreement forest F^* for F and T, the two corresponding paths $P_T[a_1, b_1]$ and $P_T[a_2, b_2]$ in the tree T must not intersect.

Note that the paths $P_{F^*}[a_1, b_1]$ and $P_{F^*}[a_2, b_2]$ in F^* are just the paths $P_F[a_1, b_1]$ and $P_F[a_2, b_2]$ in F because F^* can be obtained from F by deleting edges. Moreover, since in the forest F, the vertices v_1 and v_2 (in fact, the chain C_1) are on the path $P_F[a_1, b_1]$, and the vertices w_1 and w_2 (in fact, the chain C_2) are on the path $P_F[a_2, b_2]$, by Lemma 4, the vertices $\beta(v_1)$ and $\beta(v_2)$ (and the entire path $P_T[\beta(v_1), \beta(v_2)]$) are on the path $P_T[a_1, b_1]$, and the vertices $\beta(w_1)$ and $\beta(w_2)$ (and the entire path $P_T[\beta(w_1), \beta(w_2)]$) are on the path $P_T[a_2, b_2]$. Since the paths $P_T[a_1, b_1]$ and $P_T[a_2, b_2]$ in the tree T do not intersect, we derive that the two paths $P_T[\beta(v_1), \beta(v_2)]$ and $P_T[\beta(w_1), \beta(w_2)]$ do not intersect in the tree T, but this contradicts the assumption that the paths $P_F[v_1, v_2]$ and $P_F[w_1, w_2]$ in F are conflicting on the tree T.

Therefore, the agreement forest F^* for F and T cannot contain both chains C_1 and C_2 in F that are conflicting on T. This completes the proof of the theorem.

This leads to the following theorem, which is a very nice characterization for agreement forests for phylogenetic trees and forests.

Theorem 4. Let F be a phylogenetic forest and let T be a phylogenetic tree with the same leaf-label set, such that F has no F-MIQ for T, and contains no conflicting chains on T. Then F itself is an agreement forest for F and T.

Proof. Let T^* be a tree with a leaf-label set L^* in F, and let T^{**} be the subtree in T that is induced by the leaf-label set L^*. Since F has no F-MIQ for T, the trees T^* and T^{**} have the same collection of quartets. Since a phylogenetic tree is uniquely characterized by its quartet set [30], we conclude that the trees T^* and T^{**} are homeomorphic, i.e., the tree T^* is a subtree in T. Therefore, every tree in the forest F is a subtree in T.

If two trees T_1^* and T_2^* in F intersect at a vertex v in the tree T, then we must be able to find a chain $C_1 = P_F[v_1, v_2]$ in T_1^* and a chain $C_2 = P_F[w_1, w_2]$ in T_2^* such that the two paths $P_T[\beta(v_1), \beta(v_2)]$ and $P_T[\beta(w_1), \beta(w_2)]$ in the tree T intersect at v. The chains C_1 and C_2 are in different trees in F thus they obviously do not intersect. Thus, the chains C_1 and C_2 in F are conflicting on T, but this contradicts the assumption that the forest F contains no conflicting chains on T. Therefore, no two trees in the forest F can intersect in T.

It follows from above directly that the forest F is a subforest of the tree T, upto homeomorphism. Moreover, F is of course a subforest of itself. Therefore, the forest F is an agreement forest for F and T.

5 The algorithm

Now we are ready to present our algorithm for the Max-AF problem. For a phylogenetic forest F, denote by $|F|$ the number of trees in F. Instead of working on two phylogenetic trees, we work on a slightly more general case that constructs an agreement forest for a phylogenetic forest F and a phylogenetic tree T. We will also assume that during our process, whenever a non-leaf vertex of degree less than 2 is created, we will immediately remove the vertex.

```
Algorithm MaxAF($F, T; k$)
INPUT: $F$ is a phylogenetic forest, $T$ is a phylogenetic tree, and $k$ is the parameter
OUTPUT: an agreement forest $F^*$ of at most $|F| + k$ trees for $F$ and $T$ if such an $F^*$ exists, or 'No' if such an $F^*$ does not exist.
1.   if ($k < 0$) then return 'No';
2.   if ($F$ has an $F$-MIQ $Q$ for $T$) then
2.1      let $F_i$, $1 \leqslant i \leqslant 4$, be the four forests, each is obtained from $F$ by removing a prong in the $F$-MIQ $Q$;
2.2      recursively call MaxAF($F_i, T; k-1$), $1 \leqslant i \leqslant 4$;
2.3      if any of these calls is successful
         then output the solution returned by a(ny) successful call; else output('No');
3.   else if ($F$ contains two chains $C_1$ and $C_2$ conflicting on $T$) then
3.1      let $F_1'$ and $F_2'$ be the two forests obtained from $F$ by removing $C_1$ and $C_2$, respectively;
3.2      recursively call MaxAF($F_1', T; k-1$) and MaxAF($F_2', T; k-1$);
3.3      if any of these calls is successful
         then output the solution returned by a(ny) successful call; else output('No');
4.   else ouput $F$.
```

The correctness of the algorithm follows from the theoretical results presented in previous sections. For the phylogenetic forest F and the phylogenetic tree T, suppose that there is an agreement forest F^* of at most $|F| + k$ trees for F and T. If F has an F-MIQ Q for T, then by Theorem 2, at least one prong in Q is not in F^*. Thus, in step 2.1, at least one of the forests F_i contains the forest F^*. Let us assume without loss of generality that the forest F_1 contains F^*. Then F^* becomes an agreement forest for F_1 and T. Moreover, after deleting the prong, the number of trees in the forest is increased by 1, i.e., $|F_1| = |F| + 1$. Therefore, in this case, F_1 and T have an agreement forest F^* of at most $|F| + k = |F_1| + (k-1)$ trees. Therefore, inductively, the recursive call MaxAF$(F_1, T; k-1)$ will return a forest that is an agreement forest for F and T of at most $|F| + k$ trees. On the other hand, if none of the recursive calls returns an agreement forest, then the agreement forest F^* with at most $|F| + k$ trees for F and T cannot exist. This case is also correctly handled by step 2.3.

If F has no F-MIQ for T but contains two chains C_1 and C_2 conflicting on T, then by Theorem 3, at least one edge in the chains C_1 or C_2 is not in F^*. Note that once an edge in a chain is removed, all internal vertices of the chain will be removed by the process of repeatedly removing non-leaf vertices of degree less than 2. Therefore, we can simply remove the entire chain in a single step. Moreover, similar to the analysis for step 2, one of the recursive calls in step 3.2 will return an agreement forest of at most $|F| + k$ trees for F and T.

Finally, if the algorithm reaches step 4, then the forest F neither has F-MIQ for T nor contains chains conflicting on T. By Theorem 4, the forest F itself is an agreement forest with at most $|F| + k$ trees for F and T (note that because of step 1, we have $k \geqslant 0$ at step 4). This case is correctly handled by step 4.

Now we consider the complexity of the algorithm. Suppose that the leaf-label set L for F and T consists of n elements. Then, the number of vertices in F and T is bounded by $O(n)$. For each label quadruple (a, b, c, d), we can construct the corresponding quartets in F and T and compare if they are homeomorphic, in time $O(n)$. We do this for each of the $O(n^4)$ label quadruples in L. Note that we do not have to test if a quartet in F is an F-MIQ for T: by the comments following the definition of F-MIQ in Section 2, a quartet $Q = Q_F(a, b, c, d)$ in F that is not in T and minimizes the number of vertices in $F^{(a,Q)} \cup F^{(b,Q)} \cup F^{(c,Q)} \cup F^{(d,Q)}$ must be an F-MIQ for T. Therefore, step 2 of the algorithm takes time $O(n^5)$ before it makes the recursive calls. Step 3 takes time $O(n^3)$ because the number of pairs of chains in F is bounded by $O(n^2)$. Therefore, in time $O(n^5)$, the algorithm MaxAF reduces the given instance into at most four smaller instances with the parameter decreased by 1. This gives immediately that the running time of the algorithm MaxAF is bounded by $O(4^k n^5)$.

To solve the original Parameterized Maximum Agreement Forest problem (Max-AF) for two trees T_1 and T_2, with the parameter k, we simply call the algorithm MaxAF$(T_1, T_2; k - 1)$.

Theorem 5. The Parameterized Maximum Agreement Forest problem (Max-AF) can be solved in time $O(4^k n^5)$.

6 Conclusion

We presented an $O(4^k n^5)$-time algorithm for the Max-AF problem, which is the first fixed-parameter tractable algorithm for the problem on multifurcating phylogenetic trees. This resolves an open problem posed by several previous researches [20,23].

Recently, a new result for the Max-AF problem has been obtained. Chen et al. [31] proposed a parameterized algorithm of running time $O^*(3^k)$ by analyzing the bottommost sibling set. Much attention has been focused on the study of parameterized Maximum Agreement Forest problem on multiple trees [32]. We remark that our result is obtained independently. Compared with the above results, what is more interesting in our research is the systematic study on the graph theoretical structures of quartets and the quartet characterizations for agreement forests for multifurcating phylogenetic trees. These graph theoretical results should be useful for further study on similarities of multifurcating phylogenetic trees and for efficient exact and approximation algorithms for related problems.

Acknowledgements

This work was supported by National Natural Science Foundation of China (Grant Nos. 61232001, 61472449, 61370172, 61420106009), and Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20130162130001).

Conflict of interest The authors declare that they have no conflict of interest.

References

1 Hillis D M. Predictive evolution. Science, 1999, 286: 1866–1867

2 Ding Z, Filkov V, Gusfield D. A linear-time algorithm for the perfect phylogeny haplotyping (PPH) problem. In: Proceedings of 9th Annual International Conference of Research in Computational Molecular Biology (RECOMB 2005), Cambridge, 2005. 585–600

3 Warnow T, Ringe D, Taylor A. Reconstructing the evolutionary history of natural languages. In: Proceedings of 7th ACM-SIAM Symposium on Discrete Algorithms (SODA 1996), Atlanta, 1996. 314–322

4 Robinson D F, Foulds L R. Comparison of phylogenetic trees. Math Biosci, 1981, 53: 131–147

5 Li M, Tromp J, Zhang L. On the nearest neighbour interchange distance between evolutionary trees. J Theor Biol, 1996, 182: 463–467

6 DasGupta B, He X, Jiang T, et al. On distances between phylogenetic trees. In: Proceedings of 8th ACM-SIAM Symposium of Discrete Algorithms (SODA 1997), New Orleans, 1997. 427–436

7 Swofford D, Olsen G, Waddell P, et al Phylogenetic inference. In: Hillis D, Moritz C, Mable B, eds. Molecular Systematics. 2nd ed. Sunderland: Sinauer Associates, 1996. 407–513

8 Hein J, Jiang T, Wang L, et al. On the complexity of comparing evolutionary trees. Discrete Appl Math, 1996, 71: 153–169

9 Allen B L, Steel M. Subtree transfer operations and their induced metrics on evolutionary trees. Ann Comb, 2001, 5: 1–15

10 Bordewich M, Semple C. On the computational complexity of the rooted subtree prune and regraft distance. Ann Comb, 2005, 8: 409–423

11 Hickey G, Dehne F, Rau-Chaplin A, et al. SPR distance computation for unrooted trees. Evol Bioinform Online, 2008, 4: 17

12 Baroni M, Grnewald S, Moulton V, et al. Bounding the number of hybridisation events for a consistent evolutionary history. J Math Biol, 2005, 51: 171–182

13 Chen J E, Feng Q L. On unknown small subsets and implicit measures: new techniques for parameterized algorithms. J Comput Sci Technol, 2014, 29: 870–878

14 Feng Q L, Wang J X, Li S H, et al. Randomized parameterized algorithms for P2-Packing and Co-Path Packing problems. J Comb Optim, 2015, 29: 125–140

15 Feng Q L, Wang J X, Chen J E. Matching and weighted P2-Packing: algorithms and kernels. Theor Comput Sci, 2014, 522: 85–94

16 Feng Q L, Wang J X, Xu C, et al. Improved parameterized algorithms for minimum link-length rectilinear spanning path problem. Theor Comput Sci, 2014, 560: 158–171

17 Wang J X, Tan P Q, Yao J Y, et al. On the minimum link-length rectilinear spanning path problem: complexity and algorithms. IEEE Trans Comput, 2014, 63: 3092–3100

18 Wang J X, Li W J, Li S H, et al. On the parameterized vertex cover problem for graphs with perfect matching. Sci China Inf Sci, 2014, 57: 072107

19 Downy R, Fellows M. Parameterized Complexity. New York: Springer-Verlag, 1999

20 Hallett M, Mccartin C. A faster FPT algorithm for the maximum agreement forest problem. Theory Comput Syst, 2007, 41: 539–550

21 Whidden C, Zeh N. A Unifying View on Approximation and FPT of Agreement Forests. Berlin/Heidelberg: Springer, 2009

22 Linz S, Semple C. Hybridization in nonbinary trees. IEEE/ACM Trans Comput Biol Bioinform, 2009, 6: 30–45

23 Whidden C, Beiko R G, Zeh N. Fixed-parameter and approximation algorithms for maximum agreement forests. arXiv preprint, arXiv:1108.2664, 2011

24 Paun O, Lehnebach C, Johansson J T, et al. Phylogenetic relationships and biogeography of Ranunculus and allied genera (Ranunculaceae) in the Mediterranean region and in the European alpine system. Taxon, 2005, 54: 911–932

25 Willyard A, Wallace L E, Wagner W L, et al. Estimating the species tree for Hawaiian Schiedea (Caryophyllaceae) from multiple loci in the presence of reticulate evolution. Mol Phylogenet Evol, 2011, 60: 29–48

26 Maddison W. Reconstructing character evolution on polytomous cladograms. Cladistics, 1989, 5: 365–377

27 Whelan S, Money D. The prevalence of multifurcations in tree-space and their implications for tree-search. Mol Biol Evol, 2010, 27: 2674–2677

28 Beiko R G, Hamilton N. Phylogenetic identification of lateral genetic transfer events. BMC Evol Biol, 2006, 6: 15

29 Rodrigues E M, Sagot M F, Wakabayashi Y. The maximum agreement forest problem: approximation algorithms and computational experiments. Theor Comput Sci, 2007, 374: 91–110

30 Buneman P. The recovery of trees from measures of issimilarity. In: Hodson F, Kendall D, Tauta P, eds. Mathematics in the Archaeological and Historical Sciences. Edinburgh: Edinburgh University Press, 1971. 387–395

31 Chen J E, Fan J H, Sze S H. Parameterized and approximation algorithms for maximum agreement forest in multifurcating trees. Theor Comput Sci, 2015, 562: 496–512

32 Shi F, Wang J, Chen J E, et al. Algorithms for parameterized maximum agreement forest problem on multiple trees. Theor Comput Sci, 2014, 554: 207–216