Algorithmica (2016) 76:867–889
DOI 10.1007/s00453-015-0087-6

Approximating Maximum Agreement Forest on Multiple Binary Trees

Jianer Chen[1,2] · **Feng Shi**[1] · **Jianxin Wang**[1]

Received: 20 September 2014 / Accepted: 27 October 2015 / Published online: 2 November 2015

Abstract Given a collection of phylogenetic trees on the same leaf label-set, the MAXIMUM AGREEMENT FOREST problem (MAF) asks for a largest common subforest of these trees. The MAF problem on two binary phylogenetic trees has been studied extensively. In this paper, we are focused on the MAF problem on multiple (i.e., two or more) binary phylogenetic trees and present two polynomial-time approximation algorithms, one for the MAF problem on multiple rooted trees, and the other for the MAF problem on multiple unrooted trees. The ratio of our algorithm for the MAF problem on multiple rooted trees is 3, which is an improvement over the previous best ratio 8 for the problem. Our approximation algorithm of ratio 4 for the MAF problem on multiple unrooted trees is the first constant ratio approximation algorithm for the problem.

A preliminary version of this work was reported in the *Proceedings of the 20th International Computing and Combinatorics Conference*, Lecture Notes in Computer Science, vol. 8591, pp. 381–392, 2014.
This work is supported by the National Natural Science Foundation of China under Grants (61232001, 61472449, 61370172, 61420106009), the Major Science and Technology Research Program for Strategic Emerging Industry of Hunan (Grant No. 2012GK4054), and the Research Fund for the Doctoral Program of Higher Education of China (NO. 20130162130001).

✉ Jianxin Wang
jxwang@mail.csu.edu.cn

Jianer Chen
chen@cse.tamu.edu

1 School of Information Science and Engineering, Central South University, Changsha 410083, Hunan, People's Republic of China

2 Department of Computer Science, Texas A&M University, College Stataion, TX 77843-3112, USA

Keywords Maximum agreement forest · Approximation algorithm · Phylogenetic tree

1 Introduction

Phylogenetic (evolutionary) trees have been widely used in the study of evolutionary biology to represent the tree-like evolution of a collection of species. Given the same set of species, different data sets and different building methods may result in the construction of different trees. In order to facilitate the comparison of these different phylogenetic trees, several distance metrics have been proposed, such as *Robinson and Foulds* [17] distance, the *Nearest Neighbor Interchange* (NNI) distance [15], the *Tree-Bisection and Reconnection* (TBR) distance and the *Subtree-Prune and Regraft* (SPR) distance [7,23]. In particular, SPR and TBR distances have been commonly used in phylogenetic inference [12], and SPR operations have been applied to investigate lateral genetic transfer [3,26] and MCMC search [27].

A graph theoretical model, the *maximum agreement forest* (maf) of two phylogenetic trees, has been formulated for the TBR distance and for the SPR distance [14] for phylogenetic trees. Define the *order* of a forest to be the number of connected components in the forest.[1] Allen and Steel [2] proved that the TBR distance between two unrooted binary phylogenetic trees is equal to the order of their maf minus 1, and Bordewich and Semple [6] proved that the rSPR distance between two rooted binary phylogenetic trees is equal to the order of their rooted version of maf minus 1. In terms of computational complexity, it is known that the MAXIMUM AGREEMENT FOREST problem (MAF), i.e., constructing an maf, is NP-hard and MAX SNP-hard for two unrooted binary phylogenetic trees [14], as well as for two rooted binary phylogenetic trees [6].

Approximation algorithms have been studied for the MAF problem, mainly on two trees. For the MAF problem on two rooted binary phylogenetic trees, Hein et al. [14] proposed an approximation algorithm and claimed that the ratio of the algorithm was 3. Later Rodrigues et al. [18] found a subtle error in [14], showed that the algorithm in [14] has ratio at least 4, and presented a new approximation algorithm which they claimed had ratio 3. Bonet et al. [4] provided a counterexample and showed that both the algorithms in [14] and [18] compute a 5-approximation of the rSPR distance between two rooted binary trees. The approximation ratio was improved to 3 by Bordewich et al. [5], but at the expense of an increased running time of $O(n^5)$. A second 3-approximation algorithm presented in [19] achieves a running time of $O(n^2)$. Whidden and Zeh [24] presented the third 3-approximation algorithm, whose running time is linear. Recently, Shi et al. [21] presented a approximation algorithm of ratio 2.5, which is the best known approximation algorithm for the MAF problem on two rooted

[1] Some definitions in the study of maximum agreement forests have been somewhat confusing and misleading. If *size* denotes the number of edges in a forest, then the size of a forest is equal to the number of vertices minus its order. Thus, when the number of vertices is fixed, a forest of large size implies a small order. The terminology of "maximum agreement forest" means an agreement forest of the maximum size. However, as it has been studied in the literature, the maximum agreement forest problem is indeed a minimization problem, with the objective of minimizing the order of an agreement forest.

binary trees. For the MAF problem on two unrooted binary phylogenetic trees, the best approximation algorithm is due to Whidden and Zeh [24], which runs in linear-time and has a ratio 3.

There are also a couple of approximation algorithms for the MAF problem on two general (i.e., binary and non-binary) phylogenetic trees. Rodrigues et al. [19] developed an approximation algorithm of ratio $d + 1$ for the MAF problem on two rooted general trees, where d is the maximum number of children a vertex in the input trees may have. Chen et al. [9] developed an approximation algorithm of ratio 3 recently, which is the first constant-ratio approximation algorithm for the MAF problem on two unrooted general trees.

The MAF problem on multiple phylogenetic trees has not been studied as extensively as that on two trees. To our best knowledge, there is currently no known approximation algorithm for the MAF problem on multiple unrooted binary phylogenetic trees. The only approximation algorithm for the problem on multiple phylogenetic trees is an 8-approximation algorithm developed by Chataigner [8], which is for the problem on two or more rooted binary trees.

We remark that it makes perfect sense to investigate the MAF problem on more than two phylogenetic trees: we may construct two or more different phylogenetic trees for the same collection of species according to different data sets and different building methods. An maf of order k for a set of phylogenetic trees means that for *any* two phylogenetic trees T_i and T_j in the given set, the TBR distance (if the trees are unrooted) or the rSPR distance (if the trees are rooted) between T_i and T_j is not larger than $k - 1$. Moreover, the order of an maf of a collection of rooted trees is a lower bound on their hybridization number as it is a lower bound on the order of a maximum acyclic agreement forest of the trees, which is equal to the minimum number of hybridization nodes (nodes with multiple incoming edges) over all hybridization networks displaying the given collection of trees [28]. On the other hand, it seems much more difficult to construct an maf for more than two trees than that for two trees. For example, while there have been several polynomial-time approximation algorithms of ratio 3 for the MAF problem on two rooted binary phylogenetic trees [5,19,24], the best polynomial-time approximation algorithm [8] for the MAF problem on more than two rooted binary phylogenetic trees has a ratio 8. Also, to our best knowledge, there are currently no known polynomial-time approximation algorithms for the MAF problem on multiple unrooted binary phylogenetic trees.

In the current paper, we are focused on polynomial-time approximation algorithms for the MAF problem on multiple (i.e., two or more) binary phylogenetic trees, for both the version of rooted trees and the version of unrooted trees. We propose a very general framework for approximation algorithms for the MAF problem, which is valid for both rooted trees and unrooted trees. Our major contribution is the introduction of the concept of "edge-removal meta-steps" (or simply "meta-steps") and of the metric that evaluates the quality of the meta-steps. Roughly speaking, each meta-step is a sequence of consecutive edge removal operations, and the metric measures the ratio of the number of "essential edges" over the number of "correct edges" removed by the meta-step. A subtle issue is how to define and identify essential edges and correct edges in the entire set of edges removed by a meta-step. Our framework consists of meta-steps. We formally prove that as long as the meta-steps meet certain given con-

ditions in terms of their metric, the corresponding algorithm based on our framework is an approximation algorithm with a specific approximation ratio. We then work on the careful development of the meta-steps, for rooted trees and then for unrooted trees, focusing on achieving meta-steps that are good in terms of the proposed metric. This development results in a polynomial-time 3-approximation algorithm for the MAF problem on multiple rooted binary phylogenetic trees, which is an improvement over the previous best 8-approximation algorithm for the problem, and whose ratio matches the best known approximation ratio for the problem on two rooted binary trees. We also present a polynomial-time 4-approximation algorithm for the MAF problem on multiple unrooted binary phylogenetic trees, giving the first constant-ratio approximation algorithm for the problem.

2 Problem Formulations

We assume that readers are familiar with the general terminology of graph theory [11]. Our definitions for the study in maximum agreement forests are consistent with those used in the literature [6,13,14,24]. A *single-vertex tree* is a tree that consists of a single vertex, and a *single-edge tree* is a tree that consists of a single edge. A tree is *binary* if either it is a single-vertex tree or each of its vertices has degree either 1 or 3. The degree-1 vertices are *leaves* and the degree-3 vertices are *non-leaves* of the tree. For a subset E' of edges in a graph G, we will denote by $G \setminus E'$ the graph G with the edges in E' removed (so $G \setminus E'$ and G have the same vertex set).

The problem in our discussion has two versions, one is on unrooted trees and the other is on rooted trees. We first give the terminologies on the unrooted version, then remark on the differences for the rooted version. Let X be a fixed *label-set*.

2.1 X-Trees and X-Forests: The Unrooted Version

A binary tree is *unrooted* if no root is specified in the tree—in this case no ancestor–descendant relation is defined in the tree. For a label-set X, an unrooted *binary phylogenetic X-tree*, or simply an unrooted *X-tree*, is an unrooted binary tree whose leaves are labeled bijectively by the label-set X (and all non-leaves are unlabeled). An unrooted X-tree will also be called an (unrooted) *leaf-labeled tree* when there is no need to specify the label-set X. An unrooted *X-forest* F is a collection of disjoint leaf-labeled trees whose label-sets are disjoint such that the union of the label-sets is equal to X.

A subtree T' of an unrooted X-tree may contain unlabeled vertices of degree <3. In this case we apply the *forced contraction* operation on T', which, repeatedly, replaces each degree-2 vertex v and its incident edges with an edge connecting the two neighbors of v, and removes all unlabeled vertices of degree smaller than 2. An X-forest F is *irreducible* if forced contraction is not applicable to F. When we want to emphasize that forced contraction has been applied on a graph G, we add a subscript "fc" and write it as $(G)_{\text{fc}}$, which is irreducible. After forced contraction, an unlabeled vertex in an unrooted X-forest is always of degree 3.

Two leaf-labeled forests F_1 and F_2 are *isomorphic* if there is a graph isomorphism between F_1 and F_2 in which each leaf of F_1 is mapped to a leaf of F_2 with the same label. We will simply say that a leaf-labeled forest F' is a *subgraph* of another leaf-labeled forest F if $(F')_{\text{fc}}$ is isomorphic to $(F'')_{\text{fc}}$ for some subgraph F'' of F.

2.2 X-Trees and X-Forests: The Rooted Version

A binary tree is *rooted* if a particular *leaf* is designated as the root (so it is *both* a root and a leaf), which specifies a well-defined ancestor–descendant relation in the tree. A rooted X-*tree* is a rooted binary tree whose leaves are labeled bijectively by the label-set X. The root of an X-tree will always be labeled by a special label ρ in X. A subtree T' of a rooted X-tree T is a connected subgraph of T that contains at least one leaf in T. In order to preserve the ancestor–descendant relation in the rooted tree T, we should define the root of the subtree T'. If T' contains the leaf labeled ρ, then, certainly, ρ is the root of the subtree T'; otherwise, the vertex in T' that is in T the least common ancestor of all the labeled leaves in T' is defined to be the root of T'. A rooted X-*forest* F is a subgraph of a rooted X-tree T that contains all leaves of T. Thus, the X-forest F is a collection of disjoint (rooted) subtrees of the rooted X-tree T with disjoint leaf label-sets whose union is equal to X. In particular, one of the subtrees in a rooted X-forest F must have the leaf labeled ρ as its root.

We again have the forced contraction operation applied on a subtree T' of a rooted X-tree. However, if the root r of the subtree T' is of degree 2, then the forced contraction operation will *not* be applied on r, in order to preserve the ancestor–descendant relation in T'. Therefore, after forced contraction, the root of a subtree T' of a rooted X-forest is either an unlabeled vertex of degree 2, or the vertex labeled ρ of degree 1, or a labeled vertex of degree 0. Every unlabeled vertex in the subtree T' that is not the root of T' has degree 3.

2.3 Agreement Forests

The following terminologies are used for both rooted and unrooted versions. The *order* of an X-forest F, denoted $\text{Ord}(F)$, is the number of connected components of F that contain at least one leaf of F, or equivalently, $\text{Ord}(F)$ is equal to the number of connected components of $(F)_{\text{fc}}$.

An *agreement forest* for a collection $\{F_1, F_2, \ldots, F_m\}$ of X-forests is an X-forest that is a subgraph of F_i, for all $1 \leq i \leq m$. Note that since the concept of "subgraph" in X-forests is defined based on the forced contracted versions of the X-forests, forced contraction on any related X-forest will not affect the construction of an agreement forest for a given collection of X-forests. This fact has been well observed and used in the research on the MAF problems, see, for example, [2,4–6,13,14].

A *maximum agreement forest* (abbr. maf) for the collection $\{F_1, F_2, \ldots, F_m\}$ of X-forests is an agreement forest for $\{F_1, F_2, \ldots, F_m\}$ of the minimum order over all agreement forests for $\{F_1, F_2, \ldots, F_m\}$.

The problems we are focused on in this paper are formally described as follows.

The ROOTED MAXIMUM AGREEMENT FOREST problem (rooted MAF)
Input: A set $\{F_1, \ldots, F_m\}$ of rooted X-forests
Output: an maf, i.e., an agreement forest of the minimum order for $\{F_1, \ldots, F_m\}$

The UNROOTED MAXIMUM AGREEMENT FOREST problem (unrooted MAF)
Input: A set $\{F_1, \ldots, F_m\}$ of unrooted X-forests
Output: an maf, i.e., an agreement forest of the minimum order for $\{F_1, \ldots, F_m\}$

When each of the X-forests $F_1, \ldots, F_m$ is an X-tree, the above problems become the standard MAXIMUM AGREEMENT FOREST problems on multiple binary phylogenetic trees, for the rooted version and for the unrooted version, respectively.

3 Approximating MAF: A General Framework

We now present a general framework for approximation algorithms for the MAF problems. The discussion is valid for both rooted and unrooted versions of the problem.

In this section, we will assume that the forced contraction operation is *not* applied unless we explicitly require it. Therefore, a subgraph F' of an X-forest F may contain vertices of degree <3 that are non-leaves in the original X-forest F. We will call the vertices in F' "labeled vertices" and "unlabeled vertices" to refer to the leaves and non-leaves in the X-forest F, respectively. We will relax our definition and call such a forest F' an X-forest if there is a one-to-one mapping between the label-set X and the labeled vertices of F'. A connected component of F' is an *l-component* if it contains at least one labeled vertex. The order $\mathrm{Ord}(F')$ of F' is the number of l-components of F'.

For any edge set E' in an X-forest F, we have $\mathrm{Ord}(F \setminus E') \leq \mathrm{Ord}(F) + |E'|$. An edge subset E' of an X-forest F is an *essential edge-subset* (abbr. ee-set) if $\mathrm{Ord}(F \setminus E') = \mathrm{Ord}(F) + |E'|$. Note that every subset of an ee-set for F is an ee-set: if a subset E'' of an ee-set E' for F is not an ee-set for F, then the forest $F \setminus E''$ has its order smaller than $\mathrm{Ord}(F) + |E''|$ so the order of the forest $F \setminus E' = (F \setminus E'') \setminus (E' \setminus E'')$ is smaller than $\mathrm{Ord}(F) + |E''| + |E' \setminus E''| = \mathrm{Ord}(F) + |E'|$, contradicting the fact that E' is an ee-set for F. On the other hand, the union of ee-sets for F may not be an ee-set: for example, in an unrooted tree with a single non-leaf and three labeled leaves, every edge makes an ee-set but the union of the three edges is not an ee-set. Nevertheless, we have the following result.

Lemma 1 *Let F be an X-forest and let E_1 be an edge subset in F. Then for every ee-set $E'_1 \subseteq E_1$ for F and for every ee-set E_2 for $F \setminus E_1$, $E'_1 \cup E_2$ is an ee-set for F.*

Proof Let E''_1 be a largest ee-set for F that is a subset of E_1 and contains E'_1. Thus, $\mathrm{Ord}(F \setminus E_1) = \mathrm{Ord}(F) + |E''_1|$. We first show that $E''_1 \cup E_2$ is an ee-set for F. Let $F_1 = F \setminus (E''_1 \cup E_2)$.

Claim F_1 and $F_1 \setminus (E_1 \setminus E''_1)$ have the same order.

To prove the claim, assume the contrary that the order of $F_1 \setminus (E_1 \setminus E''_1)$ is larger than that of F_1. Then removing the edges of $E_1 \setminus E''_1$ from F_1 would split some l-component of F_1 into at least two l-components. Since each l-component of $F_1 = F \setminus (E''_1 \cup E_2)$ is a subgraph of an l-component of $F \setminus E''_1$, removing the edges of $E_1 \setminus E''_1$ from $F \setminus E''_1$ would also split some l-component of $F \setminus E''_1$ into at least

two l-components (note that all these l-components are trees). This implies that the order of $(F \setminus E_1'') \setminus (E_1 \setminus E_1'') = F \setminus E_1$ is larger than the order of $F \setminus E_1''$. But this contradicts the assumption that E_1'' is an ee-set for F and that $\mathrm{Ord}(F \setminus E_1) = \mathrm{Ord}(F) + |E_1''|$. This contradiction proves the claim that F_1 and $F_1 \setminus (E_1 \setminus E_1'')$ have the same order.

Since E_2 is an ee-set for $F \setminus E_1$, the order of $(F \setminus E_1) \setminus E_2 = (F \setminus (E_1'' \cup E_2)) \setminus (E_1 \setminus E_1'') = F_1 \setminus (E_1 \setminus E_1'')$ is equal to $\mathrm{Ord}(F \setminus E_1) + |E_2| = \mathrm{Ord}(F) + |E_1''| + |E_2|$. By the above claim, the order of $F_1 = F \setminus (E_1'' \cup E_2)$ is also $\mathrm{Ord}(F) + |E_1''| + |E_2| = \mathrm{Ord}(F) + |E_1'' \cup E_2|$, which derives that $E_1'' \cup E_2$ is an ee-set for F.

Since every subset of an ee-set for F is also an ee-set, and since $E_1' \cup E_2$ is a subset of the ee-set $E_1'' \cup E_2$ for F, we conclude that $E_1' \cup E_2$ is an ee-set for F. □

It is easy to see that for any X-subforest F' of an X-forest F, there is an ee-set E' of $\mathrm{Ord}(F') - \mathrm{Ord}(F)$ edges in F such that $(F')_{\mathrm{fc}} = (F \setminus E')_{\mathrm{fc}}$.

Up to forced contraction, every irreducible agreement forest F' for an instance $\{F_1, \ldots, F_m\}$ of MAF corresponds to a *unique* subgraph F_i' of F_i, for each i. Thus, without any confusion, we can simply say that an edge e in F_i is in or is not in the agreement forest F', as long as the edge e is in or is not in the corresponding unique subforest F_i' of F_i, respectively.

Our approximation algorithms for MAF consist of a sequence of "meta-steps". An *edge-removal meta-step* (or simply a *meta-step*) in an algorithm for MAF is a collection of consecutive computational steps in the algorithm that on an instance $\{F_1, \ldots, F_m\}$ of MAF removes certain edges in the forests in $\{F_1, \ldots, F_m\}$ (and then applies forced contraction). Our approximation algorithms have the following general framework (for both rooted and unrooted versions).

The performance of the algorithm Apx-MAF heavily depends on the quality of the meta-steps we employ in step 2 of the algorithm. For this, we introduce the following concept that measures the quality of a meta-step, where $r \geq 1$ is an arbitrary real number.

Definition-R A meta-step σ, which removes a set E^σ of edges in the forests in $\mathcal{I} = \{F_1, \ldots, F_m\}$, *keeps a ratio* r, where $r \geq 1$, if E^σ contains a subset E_1^σ of edges in F_1 such that no edge in $E^\sigma \setminus E_1^\sigma$ is in *any* agreement forest for $\{F_1 \setminus E_1^\sigma, F_2, \ldots, F_m\}$, and for each agreement forest F' for $\mathcal{I}$, there is an ee-set $E_{1,F'}^\sigma$ for F_1, $E_{1,F'}^\sigma \subseteq E_1^\sigma$, $|E_{1,F'}^\sigma| \geq |E_1^\sigma|/r$, and no edge in $E_{1,F'}^\sigma$ is in F'.

Remark 1 The meta-step σ above may also remove other edges in the forest F_1 that are not in the subset E_1^σ, as long as these edges are not in any agreement forest for $\{F_1 \setminus E_1^\sigma, F_2, \ldots, F_m\}$.

Remark 2 By definition, adding to the meta-step σ more edge removals that remove edges not in any agreement forest for $\{F_1 \setminus E_1^\sigma, F_2, \ldots, F_m\}$ does not change the ratio of the meta-step σ. In particular, if the meta-step σ removes only edges not in any agreement forest for $\{F_1, F_2, \ldots, F_m\}$, then we can let $E_1^\sigma = \emptyset$, and the meta-step σ keeps a ratio 1.

Remark 3 Definition-R looks rather complicated, for which we give some intuitive explanations. Our algorithm operates on an instance $\{F_1, F_2, \ldots, F_m\}$ by deleting

edges in $F_1, F_2, \ldots, F_m$ to eventually make $F_1, F_2, \ldots, F_m$ identical, which thus gives an agreement forest for the original $\{F_1, F_2, \ldots, F_m\}$. Therefore, allowing the edge set E^σ removed by the meta-step σ to contain edges not only in F_1 but also in $F_2, \ldots, F_m$ seems necessary. However, we require that the edge set E^σ contain an "important" subset E_1^σ in F_1 such that removing E^σ is not worse than removing E_1^σ (this is the condition that no edge in $E^\sigma \setminus E_1^\sigma$ is in any agreement forest for $\{F_1 \setminus E_1^\sigma, F_2, \ldots, F_m\}$). Moreover, for each agreement forest F' for $\{F_1, F_2, \ldots, F_m\}$, we require that there be a "correct" subset $E_{1,F'}^\sigma$ of E_1^σ (this is given by the conditions that $E_{1,F'}^\sigma$ is an ee-set for F_1 and that no edge in $E_{1,F'}^\sigma$ is in F') such that removing E_1^σ is not worse than r times removing $E_{1,F'}^\sigma$ (this is given by the condition $E_{1,F'}^\sigma \subseteq E_1^\sigma, |E_{1,F'}^\sigma| \geq |E_1^\sigma|/r$). Combining these observations, we get the condition that for any agreement forest F' for $\{F_1, F_2, \ldots, F_m\}$, removing E^σ is not worse than r times removing a correct edge subset $E_{1,F'}^\sigma$. From this, the reason why we call σ a meta-step keeping a ratio r becomes obvious.

The *optimal value* for an instance $\mathcal{I}$ of the MAF problem is the order of an maf for $\mathcal{I}$.

Theorem 1 *Let $r \geq 1$ be a real number. Suppose that each meta-step in step 2 of the algorithm Apx-MAF keeps a ratio not larger than r and that the algorithm Apx-MAF halts on an instance $\mathcal{I}_0$ of* MAF*, then the output of the algorithm Apx-MAF is an agreement forest for $\mathcal{I}_0$ whose order is at most r times the optimal value for $\mathcal{I}_0$.*

Proof Let $\mathcal{I}_0 = \{F_1^{(0)}, \ldots, F_m^{(0)}\}$. First note that each execution of step 3 of the algorithm Apx-MAF can also be regarded as a meta-step. To find the ratio of this meta-step, suppose that step 3 is applied on an instance $\{F_1'', \ldots, F_m''\}$, which is obtained by the i-th execution of step 2 on an instance $\{F_1', \ldots, F_m'\}$, where i is any integer between 1 and $m-1$. Because of the i-th execution of step 2, F_1'' and F_{i+1}'' are X-subforests of F_1' and F_{i+1}', respectively, and $F_j'' = F_j'$ for $j \neq 1, i+1$. By induction, it is easy to see that $F_1' = F_2' = \cdots = F_i'$. Therefore, F_1'' is also an X-subforest of F_j'' for $j = 2, \ldots, i$. In particular, for each $j, 2 \leq j \leq i$, any edge in F_j'' but not in F_1'' cannot be in any agreement forest for $\{F_1'', \ldots, F_m''\}$. Therefore, the meta-step made by step 3 on $\{F_1'', \ldots, F_m''\}$ removes no edge in any agreement forest for $\{F_1'', \ldots, F_m''\}$. By Remark 2, this meta-step keeps a ratio 1, which is bounded by r. Moreover, by the condition given in the theorem, each meta-step in step 2 of the algorithm keeps a ratio bounded by r.

Therefore, the algorithm Apx-MAF applies a sequence of meta-steps $\sigma_1, \sigma_2, \ldots, \sigma_t$, where t is a finite number because we assume that the algorithm halts on $\mathcal{I}_0$. By the above discussion, each meta-step σ_i keeps a ratio bounded by r. By Definition-R, for each $i, 1 \leq i \leq t$, the meta-step σ_i removes a set E^{σ_i} of edges in the forests in $\mathcal{I}_{i-1} = \{F_1^{(i-1)}, \ldots, F_m^{(i-1)}\}$ and produces an instance $\mathcal{I}_i = \{F_1^{(i)}, \ldots, F_m^{(i)}\}$, where the set E^{σ_i} contains a subset $E_1^{\sigma_i}$ of edges in $F_1^{(i-1)}$ such that no edge in $E^{\sigma_i} \setminus E_1^{\sigma_i}$ is in any agreement forest for $\{F_1^{(i-1)} \setminus E_1^{\sigma_i}, F_2^{(i-1)}, \ldots, F_m^{(i-1)}\}$, and that for each agreement forest F' for $\mathcal{I}_{i-1}$, there is an ee-set $E_{1,F'}^{\sigma_i}$ for $F_1^{(i-1)}$, $E_{1,F'}^{\sigma_i} \subseteq E_1^{\sigma_i}, |E_{1,F'}^{\sigma_i}| \geq |E_1^{\sigma_i}|/r$, and no edge in $E_{1,F'}^{\sigma_i}$ is in F'. Since the meta-step σ_i only removes edges in the forests in

$\mathcal{I}_{i-1}$, for each j, $F_j^{(i)}$ is an X-subforest of $F_j^{(i-1)}$. In particular, $F_j^{(t)}$ is an X-subforest of $F_j^{(0)}$. Since at the end of the algorithm, we have $F_1^{(t)} = F_2^{(t)} = \cdots = F_m^{(t)}$, the output $F_1^{(t)}$ of the algorithm Apx-MAF is an X-subforest of $F_j^{(0)}$ for all j, $1 \le j \le m$. This proves that the output $F_1^{(t)}$ of the algorithm Apx-MAF is an agreement forest for the input $\mathcal{I}_0$ of the algorithm.

Now consider the order of the X-forest $F_1^{(t)}$. Fix an maf F_0 for $\mathcal{I}_0$. Inductively, for a given $i \ge 0$, suppose that we have an agreement forest F_i for $\mathcal{I}_i = \{F_1^{(i)}, F_2^{(i)}, \ldots, F_m^{(i)}\}$ with $\mathrm{Ord}(F_i) \le \mathrm{Ord}(F_0) + \frac{r-1}{r} \sum_{h=1}^{i} |E_1^{\sigma_h}|$ (this certainly holds true for the case $i = 0$). Because the meta-step σ_{i+1} keeps a ratio bounded by r, for the agreement forest F_i for $\mathcal{I}_i$, there is an ee-set $E_{1,F_i}^{\sigma_{i+1}}$ for $F_1^{(i)}$, $E_{1,F_i}^{\sigma_{i+1}} \subseteq E_1^{\sigma_{i+1}}$, $|E_{1,F_i}^{\sigma_{i+1}}| \ge |E_1^{\sigma_{i+1}}|/r$, and no edge in $E_{1,F_i}^{\sigma_{i+1}}$ is in F_i. Thus, $E_1^{\sigma_{i+1}}$ contains at least $|E_1^{\sigma_{i+1}}|/r$ edges not in F_i (recall that F_i can be treated as a subgraph of $F_1^{(i)}$), so $E_1^{\sigma_{i+1}}$ contains at most $\frac{r-1}{r}|E_1^{\sigma_{i+1}}|$ edges in F_i. Therefore, the order of $F_i \setminus E_1^{\sigma_{i+1}}$ is bounded by $\mathrm{Ord}(F_i) + \frac{r-1}{r}|E_1^{\sigma_{i+1}}|$. Let $F_{i+1} = F_i \setminus E_1^{\sigma_{i+1}}$. Then F_{i+1} is an agreement forest for $\{F_1^{(i)} \setminus E_1^{\sigma_{i+1}}, F_2^{(i)}, \ldots, F_m^{(i)}\}$. By the properties of $E_1^{\sigma_{i+1}}$, no edge in $E^{\sigma_{i+1}} \setminus E_1^{\sigma_{i+1}}$ is in F_{i+1}. Thus, F_{i+1} is also an agreement forest for $\mathcal{I}_{i+1} = \{F_1^{(i+1)}, F_2^{(i+1)}, \ldots, F_m^{(i+1)}\}$, which is obtained from $\mathcal{I}_i = \{F_1^{(i)}, F_2^{(i)}, \ldots, F_m^{(i)}\}$ with the edges in $E^{\sigma_{i+1}}$ removed by the meta-step σ_{i+1}.

Thus, $F_{i+1} = F_i \setminus E_1^{\sigma_{i+1}}$ makes the induction go through: F_{i+1} is an agreement forest for $\mathcal{I}_{i+1} = \{F_1^{(i+1)}, F_2^{(i+1)}, \ldots, F_m^{(i+1)}\}$, and the order of F_{i+1}, by the inductive hypothesis, satisfies

$$\mathrm{Ord}\,(F_{i+1}) \le \mathrm{Ord}\,(F_i) + \frac{r-1}{r}\left|E_1^{\sigma_{i+1}}\right| \le \mathrm{Ord}(F_0) + \frac{r-1}{r}\sum_{h=1}^{i+1}\left|E_1^{\sigma_h}\right|.$$

This gives an agreement forest F_t for $\mathcal{I}_t = \{F_1^{(t)}, F_2^{(t)}, \ldots, F_m^{(t)}\}$ whose order satisfies $\mathrm{Ord}(F_t) \le \mathrm{Ord}(F_0) + \frac{r-1}{r}\sum_{h=1}^{t}|E_1^{\sigma_h}|$. Since F_t is an X-subforest of the X-forest $F_1^{(t)}$, we also have

$$\mathrm{Ord}\left(F_1^{(t)}\right) \le \mathrm{Ord}\,(F_t) \le \mathrm{Ord}\,(F_0) + \frac{r-1}{r}\sum_{h=1}^{t}\left|E_1^{\sigma_h}\right|. \tag{1}$$

To complete the proof, we need to compare $\mathrm{Ord}(F_1^{(t)})$ with the optimal value $\mathrm{Ord}(F_0)$. For this, we introduce one more notation. For each $i \ge 1$, let $E_{1+}^{\sigma_i}$ be the set of edges in $F_1^{(i-1)}$ that are removed by the meta-step σ_i. Thus, $E_{1,F_{i-1}}^{\sigma_i} \subseteq E_1^{\sigma_i} \subseteq E_{1+}^{\sigma_i} \subseteq E^{\sigma_i}$, and $F_1^{(i)} = F_1^{(i-1)} \setminus E_{1+}^{\sigma_i}$, where $E_{1,F_{i-1}}^{\sigma_i}$ is an ee-set for $F_1^{(i-1)}$.

It is easy to see that for $i \ne j$, the sets $E_1^{\sigma_i}$ and $E_1^{\sigma_j}$ are disjoint: suppose $i < j$, then $E_1^{\sigma_i} \subseteq E_{1+}^{\sigma_i}$ while the edges in $E_{1+}^{\sigma_i}$ are removed from $F_1^{(i-1)}$ by σ_i, so they cannot be in $F_1^{(h)}$ for any $h \ge i$. On the other hand, the edges in $E_1^{\sigma_j}$ are in $F_1^{(j-1)}$.

Inductively, suppose that for an integer $i \geq 0$ we have proved that the set $E_i = \bigcup_{h=1}^{i} E_{1,F_{h-1}}^{\sigma_h}$ is an ee-set for $F_1^{(0)}$, and that no edge in E_i is in F_0 (this is true for $i = 1$ by the definition of the set $E_{1,F_0}^{\sigma_1}$). Now consider the set $E_{1,F_i}^{\sigma_{i+1}}$ in $F_1^{(i)}$. By its properties, no edge in $E_{1,F_i}^{\sigma_{i+1}}$ is in F_i. Since $F_i = F_0 \setminus (\bigcup_{h=1}^{i} E_1^{\sigma_h})$, and $E_{1,F_i}^{\sigma_{i+1}}$ is disjoint with $E_1^{\sigma_h}$ for all $1 \leq h \leq i$ (note $E_{1,F_i}^{\sigma_{i+1}} \subseteq E_1^{\sigma_{i+1}}$), we derive that no edge in $E_{1,F_i}^{\sigma_{i+1}}$ is in F_0. Thus, no edge in the edge set $E_{i+1} = \bigcup_{h=1}^{i+1} E_{1,F_{h-1}}^{\sigma_h}$ is in F_0. Moreover, since E_i is an ee-set for $F_1^{(0)}$, $F_1^{(i)} = F_1^{(0)} \setminus (\bigcup_{h=1}^{i} E_{1+}^{\sigma_h})$, $E_i \subseteq \bigcup_{h=1}^{i} E_{1+}^{\sigma_h}$, and $E_{1,F_i}^{\sigma_{i+1}}$ is an ee-set for $F_1^{(i)}$, by Lemma 1, $E_i \cup E_{1,F_i}^{\sigma_{i+1}} = E_{i+1}$ is an ee-set for $F_1^{(0)}$. So the induction goes through. In particular, we derive that $E_t = \bigcup_{h=1}^{t} E_{1,F_{h-1}}^{\sigma_h}$ is an ee-set for $F_1^{(0)}$, and that no edge in E_t is in F_0. Since E_t is an ee-set for $F_1^{(0)}$, we have

$$\begin{aligned}\mathrm{Ord}\left(F_1^{(0)} \setminus E_t\right) = \mathrm{Ord}\left(F_1^{(0)}\right) + |E_t| &= \mathrm{Ord}\left(F_1^{(0)}\right) + \sum_{h=1}^{t} \left|E_{1,F_{h-1}}^{\sigma_h}\right| \\ &\geq \mathrm{Ord}\left(F_1^{(0)}\right) + \sum_{h=1}^{t} \left|E_1^{\sigma_h}\right| / r.\end{aligned}$$

The last equality is from the disjointness of the sets $E_{1,F_{h-1}}^{\sigma_h}$, which follows directly from the disjointness of the sets $E_1^{\sigma_h}$. Since no edge in E_t is in F_0, F_0 is an X-subforest of $F_1^{(0)} \setminus E_t$, so,

$$\mathrm{Ord}(F_0) \geq \mathrm{Ord}\left(F_1^{(0)} \setminus E_t\right) \geq \sum_{h=1}^{t} \left|E_1^{\sigma_h}\right| / r. \tag{2}$$

Combining (1) and (2), we get $\mathrm{Ord}(F_1^{(t)}) \leq r \cdot \mathrm{Ord}(F_0)$. Since $F_1^{(t)}$ is the output of the algorithm Apx-MAF and F_0 is an maf for $\mathcal{I}_0$, this inequality proves the theorem. □

Let $r \geq 1$ be a real number. An algorithm for the MAF problem is an *r-approximation algorithm* if on any instance $\mathcal{I}$ of MAF, the algorithm produces an agreement forest $F_{\mathcal{I}}$ for $\mathcal{I}$ such that the order of $F_{\mathcal{I}}$ is at most r times the optimal value for $\mathcal{I}$. By Theorem 1, if the meta-steps in step 2 can be constructed and keep ratios bounded by r, and if they guarantee that the algorithm halts on every instance of the MAF problem, then the algorithm Apx-MAF will be an r-approximation algorithm for the MAF problem. In the next two sections, we present such meta-steps for the rooted version and for the unrooted version of the MAF problem, respectively, which thus lead to the desired approximation algorithms for these problems.

4 Meta-steps for Rooted X-Forests

We develop meta-steps for rooted MAF in this section. Thus, all leaf-labeled forests considered in this section are rooted. Because of the bijection between the leaves in

Algorithm Apx-MAF($F_1, \ldots, F_m$)
Input: a collection $\{F_1, \ldots, F_m\}$ of X-forests
Output: an agreement forest F for $\{F_1, \ldots, F_m\}$
\\ We assume that the forced contraction operation is applied whenever it is possible.
1. **for** $i = 1$ **to** $m - 1$ **do**
2. **while** $F_1 \neq F_{i+1}$ **do**
apply a meta-step on (F_1, F_{i+1});
3. **for** $j = 2$ **to** i **do**
delete edges in F_j to make $F_j = F_1$;
4. **return**(F_1).

Fig. 1 Approximation algorithm for the MAF problem

an X-forest F and the elements in the label-set X, sometimes we will use, without confusion, an element in X to refer to the corresponding leaf in F, or vice versa.

As described in the algorithm Apx-MAF (see Fig. 1), for each execution of step 2 of the algorithm, we are given a fixed integer $i > 1$ and an instance $\mathcal{I} = \{F_1, F_2, \ldots, F_m\}$ of the rooted MAF problem, which is a collection of rooted X-forests, with $F_1 = F_2 = \cdots = F_{i-1}$, and, as long as $F_1 \neq F_i$, meta-steps are applied on F_1 and F_i.[2] In the following, we show how these meta-steps are constructed based on different structures of F_1 and F_i so that they can keep a ratio bounded by 3. Suppose, without loss of generality, that both F_1 and F_i are irreducible.

Two leaves of a rooted leaf-labeled forest are *siblings* if they have a common parent. Note that by definition, the root ρ, which is also a leaf, has no sibling.

Suppose that there are two elements a and b in the label-set X that are sibling leaves in both F_1 and F_i. Because our objective is to make $F_1 = F_i$, and the local structure consisting of a, b and their parent will not distinguish F_1 and F_i, we can treat the local structure as an un-decomposable unit. To implement this, we can simply replace, in both F_1 and F_i, the subtree rooted at the parent of a and b by a single leaf with a new label $\underline{ab}$. We will call such an operation as "shrinking a and b into a single leaf," and denote it by σ_1. In the further processing of F_1 and F_i, we will simply treat $\underline{ab}$ as a leaf in the forests F_1 and F_i.

The operation σ_1 changes the label-set for F_1 and F_i from X to $X' = X \setminus \{a, b\} \cup \{\underline{ab}\}$, which introduces certain subtle issues when we consider agreement forests for $\{F_1, F_2, \ldots, F_m\}$. In particular, the leaves a and b might *not* be siblings in some forests F_j with $j \neq 1, i$, so it might be impossible to shrink a and b in these X-forests. Moreover, because the operation σ_1 may be applied recursively, the labels a and b may already be composed labels. Therefore, in the general form for our discussion of the meta-steps in step 2 of the algorithm Apx-MAF, the leaf-labeled forests F_1 and F_i are X'-forests for some label-set X', while F_j, with $j \neq 1, i$, are X-forests. Each leaf in F_1 (resp. F_i) corresponds to a subtree of the original X-forest F_1 (resp. F_i) and its label in X' is given by a collection of the elements in X structured in the form to uniquely describe the subtree. To indicate these differences, we will use F_1' and F_i',

[2] The indices used here are slightly different from that used in the algorithm Apx-MAF: in the algorithm Apx-MAF, step 2 operates on F_1 and F_{i+1} for $1 \leq i \leq m - 1$, which simplifies the notations in the proof of Theorem 1; while in this section, we let step 2 of the algorithm operate on F_1 and F_i for $2 \leq i \leq m$ to simplify the descriptions of our meta-steps.

instead of F_1 and F_i, in our description of the meta-steps in step 2 of the algorithm Apx-MAF. In particular, with the new label-set X', from the X'-forests F'_1 and F'_i we can easily reconstruct the corresponding X-forests F_1 and F_i: F'_1 and F'_i are just F_1 and F_i with certain subtrees shrunk into single leaves. Thus, if F'_1, F'_i, F_1, and F_i are all irreducible, then

(1) an edge in F'_1 (resp. F'_i) is an edge in F_1 (resp. F_i);
(2) a non-leaf vertex in F'_1 (resp. F'_i) is a non-leaf vertex in F_1 (resp. F_i);
(3) an ee-set in F'_1 (resp. F'_i) is an ee-set in F_1 (resp. F_i);
(4) $F'_1 = F'_i$ as X'-forests if and only if $F_1 = F_i$ as X-forests; and
(5) edge-removal meta-steps on F'_1 and F'_i are also edge-removal meta-steps on F_1 and F_i.

In the following discussions on cases 1–3, we assume that the irreducible X'-forest F'_i has two leaves a and b that are siblings. Let τ_a and τ_b be the subtrees in both F_1 and F_i that correspond to the leaves a and b in F'_1 and F'_i, respectively. Let e'_a and e'_b be the two edges in F'_i that are incident to a and b, respectively. Thus, e'_a and e'_b are also edges in F_i that are not in $\tau_a \cup \tau_b$ but are incident to the roots of τ_a and τ_b, respectively.

Our first meta-step σ_1 now can be described as follows.

Case 1 The elements a and b are also siblings in F'_1.
Meta-step σ_1: In both F'_1 and F'_i, shrink a and b into a single leaf labeled $\underline{ab}$.

Meta-step σ_1 is a special meta-step that removes no edges in a given instance $\{F_1, F_2, \ldots, F_m\}$. Instead, it groups certain structures in F'_1 and F'_i (thus in F_1 and F_i) into un-decomposable units. Using the notation in Definition-R, we have $E^{\sigma_1} = \emptyset$. Thus, we can let $E_1^{\sigma_1} = \emptyset$, and for all agreement forests F' for $\{F_1, F_2, \ldots, F_m\}$, let $E_{1,F'}^{\sigma_1} = \emptyset$. By Definition-R, we have

Lemma 2 *Meta-step σ_1 keeps a ratio 1 on a given instance $\{F_1, F_2, \ldots, F_m\}$.*

Case 2 The elements a and b are in different connected components in F'_1.
Meta-step σ_2: In case 2, if at least one of a and b is a single-vertex tree in F'_1, then remove the edge(s) in F'_i that are incident to the corresponding leaves (a or b or both) that are single-vertex trees in F'_1; otherwise, remove in both F'_1 and F'_i the edges incident to a and b.

Lemma 3 *Meta-step σ_2 keeps a ratio 2 on a given instance $\{F_1, F_2, \ldots, F_m\}$.*

Proof We consider the first subcase. Suppose that a is a single-vertex tree in F'_1, then τ_a is a connected component of F_1. Therefore, no agreement forest for $\{F_1, F_2, \ldots, F_m\}$ can have a connected component that contains both leaves in τ_a and leaves not in τ_a. This means that the edge e'_a in F_i cannot be in any agreement forest for $\{F_1, F_2, \ldots, F_m\}$. The same argument also holds true for the leaf b. Therefore, if at least one of a and b is a single-vertex tree in F'_1, then the edge set E^{σ_2} removed by σ_2 in F'_1 and F'_i (thus in F_1 and F_i) is entirely in F_i, and contains no edge in any agreement forest for $\{F_1, F_2, \ldots, F_m\}$. Thus, for this subcase, we can let $E_1^{\sigma_2} = \emptyset$ and for every agreement forest F' for $\{F_1, F_2, \ldots, F_m\}$ let $E_{1,F'}^{\sigma_2} = \emptyset$. It is easy to verify that these sets E^{σ_2}, $E_1^{\sigma_2}$ and $E_{1,F'}^{\sigma_2}$ satisfy Definition-R with a ratio $r = 1$. Thus, in this subcase, the meta-step σ_2 keeps a ratio 1, which is <2.

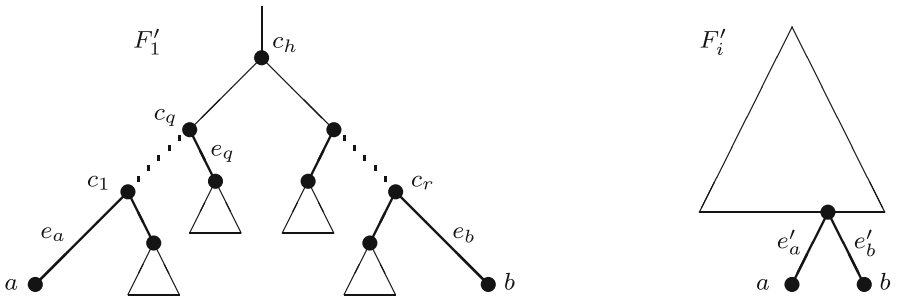

Fig. 2 The path connecting the labels a and b in F_1'

Now consider the subcase where neither of a and b is a single-vertex tree in F_1'. Let e_a and e_b be the edges incident to a and b in F_1' (thus in F_1), respectively. We have $E^{\sigma_2} = \{e_a, e_b, e_a', e_b'\}$. Let $E_1^{\sigma_2} = \{e_a, e_b\}$ and we show that $E_1^{\sigma_2}$ satisfies all conditions in Definition-R to make the meta-step σ_2 to keep a ratio 2. Obviously, $E_1^{\sigma_2} \subseteq F_1$. In the forest $F_1 \setminus E_1^{\sigma_2}$, τ_a and τ_b are by themselves two connected components. Therefore, no agreement forest for $\{F_1 \setminus E_1^{\sigma_2}, F_2, \ldots, F_m\}$ can contain an edge in $E^{\sigma_2} \setminus E_1^{\sigma_2} = \{e_a', e_b'\}$. Now let F' be an arbitrary agreement forest for $\{F_1, F_2, \ldots, F_m\}$. If both e_a' and e_b' in F_i were in F', then some leaf in τ_a and some leaf in τ_b would be in the same connected component in F'. However, this is impossible because τ_a and τ_b belong to different connected components in F_1. Therefore, for the agreement forest F' for $\{F_1, F_2, \ldots, F_m\}$, at least one of e_a' and e_b' is not in F'. As a consequence, at least one of the edges e_a and e_b in F_1 is not in F'. Let $E_{1,F'}^{\sigma_2}$ be the set of edges in $E_1^{\sigma_2} = \{e_a, e_b\}$ that are not in F', then $|E_{1,F'}^{\sigma_2}| \geq 1 = |E_1^{\sigma_2}|/2$. Finally, since a and b belong to different connected components and are not single-vertex trees in F_1', it is easy to verify that $E_{1,F'}^{\sigma_2}$ is an ee-set for F_1', thus is also an ee-set for F_1. This shows that in this subcase, meta-step σ_2 keeps a ratio 2. □

Case 3 The elements a and b are in the same connected component but are not siblings in F_1'.

Let $P = \{a, c_1, c_2, \ldots, c_r, b\}$ be the unique path in F_1' that connects a and b, in which c_h is the least common ancestor of a and b in F_1', $1 \leq h \leq r$. Since a and b are not siblings in F_1', $r \geq 2$. Let c_q be any non-leaf vertex on the path P with $c_q \neq c_h$, and let e_q be the edge incident to c_q but not on the path P (note that F_1' is binary and irreducible). Let e_a and e_b be the edges incident to a and b in F_1', respectively. See Fig. 2 for an illustration.

Meta-step σ_3: In case 3, remove the edges e_a, e_b, e_q in F_1' (thus in F_1), and remove the edges e_a' and e_b' in F_i' (thus in F_i) (see Fig. 2).

Lemma 4 *Meta-step σ_3 keeps a ratio 3 on a given instance $\{F_1, F_2, \ldots, F_m\}$.*

Proof First note that the path P is also a path in the X-forest F_1, with the two ends a and b replaced by the roots of the two subtrees τ_a and τ_b. Thus, all edges removed by σ_3 are also edges in F_1 and F_i. Again we use the notations in Definition-R. Thus, $E^{\sigma_3} = \{e_a, e_b, e_q, e_a', e_b'\}$. We let $E_1^{\sigma_3} = \{e_a, e_b, e_q\}$ and show that $E_1^{\sigma_3}$ satisfies all conditions in Definition-R and makes the meta-step σ_3 to keep a ratio 3.

Both τ_a and τ_b by themselves become connected components in the X-forest $F_1 \setminus E_1^{\sigma_3}$. Thus, a connected component of an agreement forest for $\{F_1 \setminus E_1^{\sigma_3}, F_2, \ldots, F_m\}$ either contains leaves only in τ_a, or contains leaves only in τ_b, or contains no leaves in $\tau_a \cup \tau_b$. Therefore, no edge in $E^{\sigma_3} \setminus E_1^{\sigma_3} = \{e'_a, e'_b\}$ can be in any agreement forest for $\{F_1 \setminus E_1^{\sigma_3}, F_2, \ldots, F_m\}$.

Let F' be any agreement forest for $\{F_1, F_2, \ldots, F_m\}$. We have three possible cases:

(1) The edge e'_a of F_i is not in F'. Then, a connected component of F' either contains only leaves in τ_a or contains no leaves in τ_a. In this case, we can pick $\{e_a\}$ as the set $E_{1,F'}^{\sigma_3}$, which satisfies: $E_{1,F'}^{\sigma_3} \subseteq E_1^{\sigma_3}$, $|E_{1,F'}^{\sigma_3}| = 1 \geq |E_1^{\sigma_3}|/3$, and the edge e_a in $E_{1,F'}^{\sigma_3}$ is not in F'. Moreover, since F'_1 is irreducible and a is not a single-vertex tree in F'_1, the set $E_{1,F'}^{\sigma_3}$ is an ee-set for F'_1, thus also an ee-set for F_1. Thus, for the agreement forest F' not containing e'_a, the set $E_{1,F'}^{\sigma_3} = \{e_a\}$ satisfies all conditions in Definition-R to make the meta-step σ_3 to keep a ratio 3.

(2) The edge e'_b is not in F'. Then similarly we let $E_{1,F'}^{\sigma_3} = \{e_b\}$, and can verify that for the agreement forest F' not containing e'_b, the set $E_{1,F'}^{\sigma_3} = \{e_b\}$ satisfies all conditions in Definition-R to make the meta-step σ_3 to keep a ratio 3.

(3) Both edges e'_a and e'_b are in F'. Since a and b are siblings in F'_i, the roots of the subtrees τ_a and τ_b in F_i must have a common parent p in F'. Since F' is a subgraph of F_1 that must preserve the ancestor-descendent relations in F_1, the vertex c_h in F_1 must correspond to the vertex p in F'. As a consequence, no edge in F_1 that is incident to a vertex c_j on the path P with $c_j \neq c_h$ but not on the path P can be in F' (see Fig. 2 for references). In particular, the edge e_q is not in F'. So in this case, we let $E_{1,F'}^{\sigma_3} = \{e_q\}$, and can verify easily that for the agreement forest F' containing both e'_a and e'_b, the set $E_{1,F'}^{\sigma_3} = \{e_q\}$ satisfies all conditions in Definition-R to make the meta-step σ_3 to keep a ratio 3. Note that the fact $E_{1,F'}^{\sigma_3}$ is an ee-set for F_1 follows from the irreducibilities of the X'-forest F'_1 and the X-forest F_1.

This verifies that the set $E_1^{\sigma_3}$ satisfies all conditions in Definition-R to make the meta-step σ_3 to keep a ratio 3. Thus, the meta-step σ_3 keeps a ratio 3. □

Cases 1–3 cover all cases in which a and b are sibling leaves in F'_i. If the X'-forest F'_i has no sibling leaves, then it must be in one of the following two cases: (1) F'_i contains no edges; and (2) all connected components of F'_i are single-vertex trees, except one that is a single-edge tree, and the single-edge tree has a root labeled ρ and a leaf labeled $a \in X'$, $a \neq \rho$.

Case 4 F'_i contains no edges.
Meta-step σ_4: In case 4, remove all edges in F'_1.

Lemma 5 *Meta-step σ_4 keeps a ratio 1 on a given instance $\{F_1, F_2, \ldots, F_m\}$.*

Proof In case 4, it is obvious that no edges in F'_1, which are also edges in F_1, can be in any agreement forest for $\{F_1, F_2, \ldots, F_m\}$. By Remark 2, the meta-step σ_4 keeps a ratio 1. □

Now we come to our last case.

Case 5 F_i' has a single edge, which makes a single-edge tree rooted at ρ with a leaf a, $a \neq \rho$.
Meta-step σ_5: In case 5, remove all edges in F_1' except those that are on the path between ρ and a. If F_1' becomes a collection of single-vertex trees, then also remove the edge $[\rho, a]$ in F_i'.

Lemma 6 *Meta-step σ_5 keeps a ratio 1 on a given instance $\{F_1, F_2, \ldots, F_m\}$.*

Proof Let $\rho, a, b_1, \ldots, b_h$ be the leaves in F_i', where each b_i is a single-vertex tree in F_i'. Let τ_a and τ_{b_i}, $1 \leq i \leq h$, be the subtrees in F_i that correspond to the leaves a and b_i in F_i', respectively. Then, for each subtree τ_{b_i}, a connected component of an agreement forest F' for $\{F_1, F_2, \ldots, F_m\}$ either contains only leaves in τ_{b_i} or contains no leaves in τ_{b_i}. Therefore, if there is an edge e_{b_i} incident to b_i in F_1', which is also the edge between the root of τ_{b_i} and its parent in F_1, then the edge e_{b_i} cannot be in F'. This observation plus the forced contraction operation shows that the edges that are not on the path between ρ and a in F_1' cannot be in any agreement forest for $\{F_1, F_2, \ldots, F_m\}$. Thus, all edges in F_1' (thus also in F_1) that are removed by the meta-step σ_5 are not in any agreement forest for $\{F_1, F_2, \ldots, F_m\}$. Finally, if F_1' becomes a collection of single-vertex trees after σ_5 removes edges in F_1' (this is the case when ρ and a are not in the same connected component in F_1'), then no agreement forest for $\{F_1, F_2, \ldots, F_m\}$ can have a connected component containing both ρ and a leaf in τ_a. Therefore, in this case, the edge $[\rho, a]$ in F_i' (thus in F_i) cannot be in any agreement forest for $\{F_1, F_2, \ldots, F_m\}$. In summary, no edge in the edge set E^{σ_5} removed by the meta-step σ_5 can be in an agreement forest for $\{F_1, F_2, \ldots, F_m\}$. By Remark 2, the meta-step σ_5 keeps a ratio 1. □

Now we are ready for our main theorem in this section. Suppose $|X| = n$. Each X-forest has a size (i.e., the number of vertices plus the number of edges) $O(n)$. Therefore, the size of an instance $\{F_1, F_2, \ldots, F_m\}$ of the rooted MAF problem is $n_0 = O(nm)$.

Theorem 2 *If step 2 of the algorithm Apx-MAF uses the meta-steps σ_1–σ_5, then the algorithm Apx-MAF is a 3-approximation algorithm for the rooted* MAF *problem that runs in time $O(n_0 \log n_0)$ on an instance of size n_0.*

Proof By Lemmas 2–6, each of the meta-steps σ_1–σ_5 keeps a ratio bounded by 3. By Theorem 1, if the algorithm Apx-MAF uses these meta-steps in step 2, and halts on an instance $\mathcal{I}$ of rooted MAF, then the algorithm produces an agreement forest for the instance $\mathcal{I}$ whose order is bounded by three times the optimal value for $\mathcal{I}$. Therefore, to show that the algorithm Apx-MAF is a 3-approximation algorithm for the rooted MAF problem, it suffices to show that on any instance $\mathcal{I}$ of size n_0, the algorithm Apx-MAF runs in time $O(n_0 \log n_0)$.

Let the instance $\mathcal{I}$ be $\{F_1, F_2, \ldots, F_m\}$. Thus, $n_0 = O(nm)$. Now fix an i, and consider the processing of F_1 and F_i in step 2 of the algorithm Apx-MAF. By the algorithm, a meta-step in step 2 is applied on F_1 and F_i only when $F_1 \neq F_i$. Under the condition $F_1 \neq F_i$, it is easy to verify that each of the meta-steps σ_2–σ_5 removes at least one edge in $F_1 \cup F_i$. Therefore, the total number of times these meta-steps

can be applied is bounded by $O(n)$. Moreover, each application of the meta-step σ_1 shrinks three vertices into a single vertex, in each of F_1 and F_i (recall that we are operating on F_1' and F_i'). Therefore, the meta-step σ_1 can be applied at most $O(n)$ times. Summarizing all these, we conclude that if the algorithm Apx-MAF uses the meta-steps σ_1–σ_5 in step 2, then the total number of times the meta-steps are applied for processing F_1 and F_i for each i in the execution of step 2 is $O(n)$.

It is easy to see that each meta-step can be implemented to run in time $O(n)$, which then directly gives a simple $O(n_0^2)$-time implementation of the algorithm Apx-MAF. In the following, we explain how the running time of the algorithm can be further improved to $O(n_0 \log n_0)$.

Each of the meta-steps σ_4 and σ_5 is applied at most once in step 2 for processing F_1 and F_i. Each of the meta-steps σ_1–σ_3 is called on two sibling leaves in the forest F_i. Therefore, step 2 of the algorithm can be implemented by a depth-first search on the forest F_i, which continuously presents siblings in F_i for possible applications of the meta-steps σ_1–σ_3, until the meta-steps σ_4 and σ_5 become applicable. This depth-first search process, without counting the complexity of the calls to the meta-steps, runs in time $O(n)$.

The meta-steps σ_1–σ_3 also require efficient determination on whether two leaves are in the same connected component in F_1. Note that the connected component structure of F_1 is dynamically changing, in particular when the meta-step σ_3 removes the edge e_q that can be connected to a non-trivial subtree (see Fig. 2). For this, we can organize the leaves in F_1 in depth-first search order so that all leaves in a subtree appear in a consecutive segment. Such a sequence then can be stored in a 2-3 tree that supports logarithmic-time insertion, deletion, splice, and split [1]. Based on this data structure, the connected component structure of F_1 can be dynamically updated in time $O(\log n)$ for each application of the meta-steps σ_1–σ_3.

With the above implementations, we conclude that each of the meta-steps σ_1–σ_3 takes time $O(\log n)$, thus, for a given i, the running time of step 2 of the algorithm is $O(n \log n)$. Also note that step 3 of the algorithm Apx-MAF is actually "virtual", for which we can, without doing any real computation, simply record that $F_1 = F_j$ for all $1 \leq j \leq i$. As a consequence, the total running time of the algorithm Apx-MAF is bounded by $O(n \log n \cdot m) = O(n_0 \log n_0)$, where $n_0 = O(nm)$ is the size of the input instance $\mathcal{I}$. □

If the original input of our algorithm is a collection of X-trees, then the algorithm Apx-MAF will return an agreement forest for the trees. Thus, the algorithm Apx-MAF is a 3-approximation algorithm for the standard Maximum Agreement Forest problem on multiple rooted binary phylogenetic trees.

5 Meta-steps for Unrooted X-Forests

For the unrooted Maf problem, the meta-steps used in step 2 of the algorithm Apx-MAF and their analysis proceed in a manner similar to those for rooted Maf. However, since an unrooted X-tree enforces no ancestor–descendant relation in the tree, subforests in the X-tree have no requirement of preserving such a relation. This fact

induces certain subtle differences. As a starting point, note that in an irreducible unrooted X-forest, every non-leaf has degree 3.

Again, for each execution of step 2 of the algorithm Apx-MAF, we are given a fixed integer $i > 1$ and an instance $\mathcal{I} = \{F_1, F_2, \ldots, F_m\}$ of the unrooted MAF problem, which is a collection of unrooted X-forests, with $F_1 = F_2 = \cdots = F_{i-1}$, and, as long as $F_1 \neq F_i$, meta-steps are applied on F_1 and F_i. We present the meta-steps and show that these meta-steps keep a ratio bounded by 4. Suppose, without loss of generality, that both F_1 and F_i are irreducible.

Two leaves a and b of an unrooted X-forest F are *edge-siblings* if they are the two leaves of a single-edge tree in F, and are *vertex-siblings* if they are adjacent to the same non-leaf vertex p in F (in this case, the vertex p is called the "parent" of a and b). The leaves a and b are *siblings* if they are either vertex-siblings or edge-siblings.

Again for two leaves a and b that are vertex-siblings in both F_1 and F_i, we can replace the subtree consisting of a, b, and their parent with a single leaf labeled $\underline{ab}$. Similarly, for two leaves a and b that are edge-siblings in both F_1 and F_i, we can replace the single-edge tree $[a, b]$ with a single-vertex tree labeled $\underline{ab}$. We will call the above operations on vertex-siblings and edge-siblings as "shrinking the siblings a and b into a single leaf $\underline{ab}$." Again because of this, we will use two X'-forests F_1' and F_i' for some label-set X', instead of the X-forests F_1 and F_i, in the description of our meta-steps in step 2 of the algorithm Apx-MAF, where each element in the label-set X' is a collection of elements in the label-set X structured to represent a subtree of F_1 and F_i. In other words, the X'-forests F_1' and F_i' are just the X-forests F_1 and F_i with certain subtrees shrunk into single leaves. In particular, edges and non-leaf vertices of F_1' and F_i', respectively, are also edges and non-leaf vertices of F_1 and F_i.

In the discussions on cases 1–3 below, we assume that the irreducible X'-forest F_i' has two leaves a and b that are siblings (either edge-siblings or vertex-siblings). Let τ_a and τ_b be the subtrees in both F_1 and F_i that correspond to the leaves a and b in F_1' and F_i', respectively. Let e_a' and e_b' be the edges in F_i' that are incident to a and b, respectively (if a and b are edge-siblings, then $e_a' = e_b'$). Note that e_a' and e_b' are the edges in F_i that are not in $\tau_a \cup \tau_b$ but are incident to the roots of τ_a and τ_b, respectively.

Case 1 The elements a and b are also siblings in F_1'.

Meta-step ω_1: In case 1, shrink a and b into a single leaf $\underline{ab}$ in both F_1' and F_i'. If $\underline{ab}$ is a single-vertex tree in exactly one of F_1' and F_i', then remove the edge incident to $\underline{ab}$ in the other.

Lemma 7 *Meta-step ω_1 keeps a ratio 1 on a given instance $\{F_1, F_2, \ldots, F_m\}$.*

Proof If a and b are either edge-siblings in both F_1' and F_i', or vertex-siblings in both F_1' and F_i', then after shrinking a and b into $\underline{ab}$, the vertex $\underline{ab}$ either is a single-vertex tree in both F_1' and F_i', or is a single-vertex tree in neither of F_1' and F_i'. Therefore, in this case, the meta-step ω_1 removes no edges in F_1' and F_i', thus also removes no edges in F_1 and F_i. As we discussed for the meta-step σ_1 in Lemma 2, in this case, the meta-step ω_1 keeps a ratio 1.

Now suppose that a and b are edge-siblings in exactly one of F_1' and F_i'. Without loss of generality, suppose that a and b are edge-siblings in F_1' but are vertex-siblings in F_i'. Let e_0' be the edge incident to the parent of a and b in F_i' such that $e_0' \neq e_a'$ and $e_0' \neq e_b'$. Because of the single-edge tree $[a, b]$ in F_1', no connected component

of an agreement forest F' for $\{F_1, F_2, \ldots, F_m\}$ can contain both leaves in $\tau_a \cup \tau_b$ and leaves not in $\tau_a \cup \tau_b$. Therefore, the edge e'_0 in F'_i cannot be in F' and can be removed. After removing e'_0 from F'_i and by applying a forced contraction, a and b become edge-siblings in F'_i, thus we can shrink a and b in both F'_1 and F'_i. Note that this is equivalent to the meta-step ω_1 that first shrinks the edge-siblings a and b in F'_1 and the vertex-siblings a and b in F'_i, then removes the edge incident to $\underline{ab}$ in F'_i (which is just e'_0). As a fact, in this case, the meta-step ω_1 only removes an edge (i.e., e'_0) that is not in any agreement forest for $\{F_1, F_2, \ldots, F_m\}$. By Remark 2, in this case, the meta-step ω_1 also keeps a ratio 1. □

Case 2 The elements a and b are in different connected components in F'_1.
Meta-step ω_2: In case 2, if at least one of a and b is a single-vertex tree in F'_1, then remove the edge(s) in F'_i that are incident to the corresponding leaves (a or b or both) that are single-vertex trees in F'_1; otherwise, remove in both F'_1 and F'_i the edges incident to a and b.

Lemma 8 *Meta-step ω_2 keeps a ratio 2 on a given instance $\{F_1, F_2, \ldots, F_m\}$.*

Proof The proof for this lemma is similar to that for Lemma 3. If an element in $\{a, b\}$ is a single-vertex tree in F'_1, then the edge incident to that element in F'_i cannot be in any agreement forest for $\{F_1, \ldots, F_m\}$. Thus, by Remark 2, in this subcase, the meta-step ω_2 keeps a ratio 1.

Now assume that neither of a and b is a single-vertex tree in F'_1. Let e_a and e_b be the two edges in F'_1 that are incident to a and b, respectively. Note that even though it is possible that $e'_a = e'_b$, we must have $e_a \neq e_b$ because a and b are in different connected components in F'_1.

Using the notations in Definition-R, $E^{\omega_2} = \{e_a, e_b, e'_a, e'_b\}$. Let $E_1^{\omega_2} = \{e_a, e_b\}$. The proof that the set $E_1^{\omega_2}$ satisfies all conditions in Definition-R to make the meta-step ω_2 to keep a ratio 2 goes exactly the same as that for the corresponding subcase in the proof for Lemma 3, no matter whether $e'_a = e'_b$,. Therefore, we conclude that the meta-step ω_2 keeps a ratio 2. □

Case 3 The elements a and b are in the same connected component in F'_1, but are not siblings.

This case is the one that is most different from its corresponding case for the rooted version. Let $P = \{a, c_1, c_2, \ldots, c_r, b\}$ be the unique path in the unrooted X'-forest F'_1 that connects a and b, where $r \geq 2$ because we assume that a and b are neither edge-siblings nor vertex-siblings. Let e_a and e_b be the edges in F'_1 that are incident to a and b, respectively. Moreover, let e_1 be the edge in F'_1 that is incident to c_1 but not on the path P, and let e_r be the edge in F'_1 that is incident to c_r but not on the path P. See Fig. 3 for an illustration. Note that since F'_1 is irreducible, e_a, e_b, e_1, and e_r are four well-defined distinct edges in F'_1 (thus also in F_1). Moreover, the path P is also a path in F_1 in which the two ends a and b are replaced by the roots of the two subtrees τ_a and τ_b of F_1, respectively.
Meta-step ω_3: In case 3, remove the edges e_a, e_b, e_1, e_r in F'_1, and the edges e'_a, e'_b in F'_i.

Lemma 9 *Meta-step ω_3 keeps a ratio 4 on a given instance $\{F_1, F_2, \ldots, F_m\}$.*

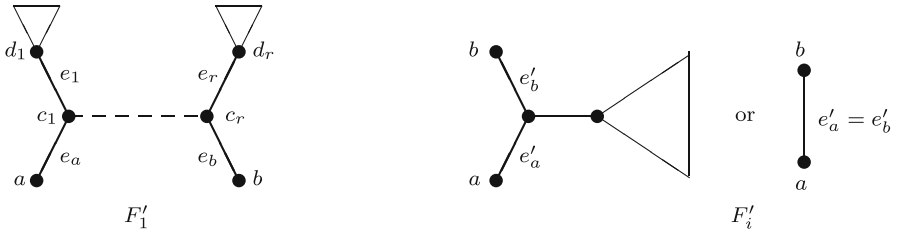

Fig. 3 The path connecting the elements a and b in F_1'

Proof Using the notations in Definition-R, we have (note that it is possible that $e'_a = e'_b$) $E^{\omega_3} = \{e_a, e_b, e_1, e_r, e'_a, e'_b\}$. Let $E_1^{\omega_3} = \{e_a, e_b, e_1, e_r\}$. We show that the set $E_1^{\omega_3}$ satisfies all conditions in Definition-R and makes the meta-step ω_3 to keep a ratio 4. First note that $|E_1^{\omega_3}| = 4$ no matter whether $e'_a = e'_b$.

In the X'-forest $F_1' \setminus E_1^{\omega_3}$, both a and b become single-vertex trees. Thus, both subtrees τ_a and τ_b by themselves become connected components in the X-forest $F_1 \setminus E_1^{\omega_3}$. As a consequence, a connected component of an agreement forest for $\{F_1 \setminus E_1^{\omega_3}, F_2, \ldots, F_m\}$ either contains leaves only in τ_a, or contains leaves only in τ_b, or contains no leaves in $\tau_a \cup \tau_b$. Therefore, no edge in $E^{\omega_3} \setminus E_1^{\omega_3} = \{e'_a, e'_b\}$ can be in any agreement forest for $\{F_1 \setminus E_1^{\omega_3}, F_2, \ldots, F_m\}$. Note that this holds true no matter whether $e'_a = e'_b$.

Let F' be any agreement forest for $\{F_1, F_2, \ldots, F_m\}$. We have three possible cases:

(1) The edge e'_a is not in F'. In this case, no connected component in F' can contain both leaves in τ_a and leaves not in τ_a. So we can pick $\{e_a\}$ as the set $E_{1,F'}^{\omega_3}$, which satisfies: $E_{1,F'}^{\omega_3} \subseteq E_1^{\omega_3}$, $|E_{1,F'}^{\omega_3}| = 1 \geq |E_1^{\omega_3}|/4$, and the edge e_a in $E_{1,F'}^{\omega_3}$ is not in F'. Moreover, since F_1' is irreducible, the set $E_{1,F'}^{\omega_3}$ is an ee-set for F_1', thus is also an ee-set for F_1. Thus, for the agreement forest F' for $\{F_1, F_2, \ldots, F_m\}$ that does not contain e'_a, the set $E_{1,F'}^{\omega_3} = \{e_a\}$ satisfies all conditions in Definition-R to make the meta-step ω_3 to keep a ratio 4.

(2) The edge e'_b is not in F'. Then similarly we can let $E_{1,F'}^{\omega_3} = \{e_b\}$, and verify that for the agreement forest F' for $\{F_1, F_2, \ldots, F_m\}$ that does not contain e'_b, the set $E_{1,F'}^{\omega_3} = \{e_b\}$ satisfies all conditions in Definition-R to make the meta-step ω_3 to keep a ratio 4.
If $e'_a = e'_b$ and the edge is not in F', then we can apply either (1) or (2) above to have a set $E_{1,F'}^{\omega_3}$ that satisfies all conditions in Definition-R to make the meta-step ω_3 to keep a ratio 4.

(3) The agreement forest F' contains both edges e'_a and e'_b, which includes the case where F' contains $e'_a = e'_b$. In this case, because a and b are siblings in F_i', in the X-forest F_i there must be a leaf l_a in the subtree τ_a and a leaf l_b in the subtree τ_b, where $l_a, l_b \in X$, such that l_a and l_b are in the same connected component in F_i. Observe that because a and b are siblings in F_i', the path between the roots of the two subtrees τ_a and τ_b in F_i can contain at most one non-leaf vertex in F_i. Since F' is a subgraph of F_i, the path between l_a and l_b in F' contains at most one non-leaf vertex that is not in $\tau_a \cup \tau_b$. Since both vertices c_1 and c_r in F_1 are on the path between l_a and l_b and are not in $\tau_a \cup \tau_b$, at most one of c_1 and c_r can become

a non-leaf vertex in F'. As a consequence, at most one of the two edges e_1 and e_r in F'_1 can be in F' (see Fig. 3 for references). Note that e_1 and e_r are also edges in F_1. Thus, if we let $E^{\omega_3}_{1,F'}$ be the set of edges in $\{e_1, e_r\}$ that are not in F', then the set $E^{\omega_3}_{1,F'}$ satisfies: $E^{\omega_3}_{1,F'} \subseteq E^{\omega_3}_1$, $|E^{\omega_3}_{1,F'}| \geq 1 = |E^{\omega_3}_1|/4$, and the edges in $E^{\omega_3}_{1,F'}$ are not in F'. Moreover, it is not difficult to verify that $\{e_1, e_r\}$ is an ee-set for F'_1 (thus an ee-set for F_1). Since a subset of an ee-set is also an ee-set, the set $E^{\omega_3}_{1,F'}$ is also an ee-set for F_1. Thus, in this case, the set $E^{\omega_3}_{1,F'}$ defined as this satisfies all conditions in Definition-R to make the meta-step ω_3 to keep a ratio 4.

This verifies that the set $E^{\omega_3}_1$ satisfies all conditions in Definition-R to make the meta-step ω_3 to keep a ratio 4. Thus, the meta-step ω_3 keeps a ratio 4. □

Cases 1–3 cover all cases in which the X'-forest F'_i contains siblings. If F'_i contains no siblings, then F'_i contains no edges. This case is handled by the following meta-step.

Case 4 F'_i contains no edges.

Meta-step ω_4: In case 4, remove all edges in F'_1.

It is rather easy to see that the meta-step ω_4 removes no edge in any agreement forest for $\{F_1, F_2, \ldots, F_m\}$. By Remark 2, we have

Lemma 10 *Meta-step ω_4 keeps a ratio 1 on a given instance $\{F_1, F_2, \ldots, F_m\}$.*

Now we are ready for our main theorem in this section.

Theorem 3 *If step 2 of the algorithm Apx-MAF uses the meta-steps ω_1–ω_4, then the algorithm Apx-MAF is a 4-approximation algorithm for the unrooted* MAF *problem with running time $O(n_0 \log n_0)$, where n_0 is the size of the input instance.*

Proof By Lemmas 7–10, each of the meta-steps ω_1–ω_4 keeps a ratio bounded by 4. Thus, by Theorem 1, in order to prove that Apx-MAF is a 4-approximation algorithm for the unrooted MAF problem, it suffices to prove that the algorithm, when using the meta-steps ω_1–ω_4 in its step 2, runs in time $O(n_0 \log n_0)$ on an instance of size n_0 of the unrooted MAF problem.

Suppose that the given instance of the unrooted MAF problem is $\{F_1, F_2, \ldots, F_m\}$, where each F_h is an unrooted X-forest, and $|X| = n$. Thus, $n_0 = O(nm)$. According to the algorithm Apx-MAF, meta-steps in the i-th execution of step 2 are applied on F'_1 and F'_i (thus on F_1 and F_i) only when $F'_1 \neq F'_i$. When $F'_1 \neq F'_i$, each of the meta-steps ω_2–ω_4 removes at least one edge in $F'_1 \cup F'_i$ (thus in $F_1 \cup F_i$). Therefore, the total number of times these meta-steps are applied is bounded by $O(n)$. Moreover, similar to our discussion in Theorem 2, each application of the meta-step ω_1 reduces the number of vertices in F'_1 and F'_i by at least 2. Therefore, the meta-step ω_1 can be applied at most $O(n)$ times.

In summary, each execution of step 2 of the algorithm Apx-MAF, when using the meta-steps ω_1–ω_4, applies at most $O(n)$ meta-steps. Using data structures similar to those we used in Theorem 2, each of the meta-steps ω_1–ω_4 can be implemented to run in time $O(\log n)$. This then leads to an $O(n_0 \log n_0)$-time implementation of the algorithm Apx-MAF for the unrooted MAF problem, where n_0 is the size of the input instance. □

If the original input of the algorithm is a collection of unrooted X-trees, then the algorithm Apx-MAF will return an agreement forest for the trees. In this case, the algorithm Apx-MAF is a 4-approximation algorithm for the standard MAXIMUM AGREEMENT FOREST problem on multiple unrooted binary phylogenetic trees.

6 Conclusion and Future Research

In this paper, we presented two polynomial-time approximation algorithms for the MAF problem on multiple binary phylogenetic trees: one for rooted trees with a ratio 3 and the other for unrooted trees with a ratio 4. The 3-approximation algorithm for rooted trees is an improvement over the previous best approximation algorithm for the problem due to Chataigner [8], which has a ratio 8,[3] and the 4-approximation algorithm for unrooted trees is, to our best knowledge, the first constant ratio approximation algorithm for the problem.

As suggested by Whidden et al. [25] in their recent publication in *SIAM J. Comput.*, "*The most important open problem is extending our approach to computing MAFs and MAAFs for multifurcating trees and for more than two trees.*" Our result is a response to this call and makes an important step towards this direction. We believe that our general framework, the algorithm Apx-MAF, will have further applications in the study of approximation algorithms for the MAF problem. Indeed, by Theorem 1, any further improvement on the ratio of the meta-steps will directly lead to improvements in the corresponding approximation algorithms. Moreover, by combining our general framework and related techniques presented in the current paper with the techniques developed recently by Chen et al. [9] for multifurcating trees (i.e., general trees, instead of only binary trees), we believe that we should also be able to develop approximation algorithms for the MAF problem on multiple multifurcating trees.

Further improvements on the approximation ratio of polynomial-time approximation algorithms for the MAF problem, either for two binary or multifurcating phylogenetic trees, or more general for multiple binary or multifurcating phylogenetic trees, are certainly desired. In particular, the best approximation algorithm for the MAF problem on two unrooted binary X-trees has a ratio 3 [24], while our approximation algorithm for the problem on multiple unrooted binary X-trees has a ratio 4 (Theorem 3). The disparity appears because our meta-step ω_3 has a ratio 4 (Lemma 9), while in handling the same situation for two unrooted binary X-trees, the algorithm proposed in [24] is able to limit the number of removed edges by 3, instead of 4 (see Theorem 6 in [24]). Unfortunately, the operation described in [24] cannot be easily translated into an efficient meta-step. In fact, a direct translation of the operation given in [24] will result in a meta-step that does not guarantee *any* positive ratio. This is also the main reason why our algorithm has an extra $\log n$ factor in its time complexity. It will be interesting to see how this gap can be closed, either by strengthening the definition of the meta-step metric or by developing new algorithmic techniques.

[3] During the preparation of the final version of this manuscript, the authors were informed by an anonymous referee that Mukhopadhyay and Bhabak had announced an $O(kn^5)$-time approximation algorithm of ratio 3 for the MAF problem on k rooted binary phylogenetic trees [16].

Approximation algorithms for the MAF problem, in particular approximation algorithms for the SPR distance have been used in a branch-and-bound fashion to quickly compute exact SPR distance for two phylogenetic trees [26,27]. Our methods and formulations presented in the current paper may be used in the same fashion for the multiple tree problem.

Accompanied by the research in approximation algorithms for the MAF problem, there is also an active line of research on parameterized algorithms for the problem [2,9,10,13,20,22,24,25]. In particular, work has been done on parameterized algorithms for the MAF problem on multiple binary trees [10,20]. The parameterized algorithms in [10,20] and the approximation algorithms presented in the current paper share a common idea of using sibling pairs in one tree to identify edges in the other trees that may potentially cause inconsistence. The parameterized algorithms, which are based on a branch-and-search process, then branch on removing each of these potentially inconsistent edges, while the approximation algorithms simply remove all these potentially inconsistent edges. However, it seems that the analysis for the parameterized algorithms based on this idea is much easier: the algorithms only need to ensure that at least one branch in the branch-and-search process traces an *optimal* solution. On the other hand, after removing all potentially inconsistent edges, it becomes much more difficult to characterize the optimal solutions in the resulting instance. Thus, among the removed edges, we have to identify "irrelevant edges", and find a more accurate way to compute the ratio of the number of "essential edges" over the number of "correct edges". In particular, simply counting the number of removed edges and the number of "correctly" removed edges might give a very loose estimation on the resulting approximation ratio of the algorithms. This difference has forced us to build a very different model to enable more precise analysis on approximation algorithms for the MAF problem on multiple trees.

Acknowledgments We would like to thank the anonymous referees, whose comments and suggestions have greatly improved the presentation of this paper. In particular, a referee provided further pointers to applications of algorithms for maximum agreement forests on multiple trees, and another referee updated us of the status of approximation algorithms for maximum agreement forests on multiple rooted trees.

References

1. Aho, A., Hopcroft, J., Ullman, J.: The Design and Analysis of Computer Algorithms. Addison-Wesley, Reading (1974)
2. Allen, B., Steel, M.: Subtree transfer operations and their induced metrics on evolutionary trees. Ann. Comb. **5**(1), 1–15 (2001)
3. Beiko, R.G., Hamilton, N.: Phylogenetic identification of lateral genetic transfer events. BMC Evolut. Biol. **6**(1), 15 (2006)
4. Bonet, M., John, K.S., Mahindru, R., Amenta, N.: Approximating subtree distances between phylogenies. J. Comput. Biol. **13**(8), 1419–1434 (2006)
5. Bordewich, M., McCartin, C., Semple, C.: A 3-approximation algorithm for the subtree distance between phylogenies. J. Discrete Algorithms **6**(3), 458–471 (2008)
6. Bordewich, M., Semple, C.: On the computational complexity of the rooted subtree prune and regraft distance. Ann. Comb. **8**(4), 409–423 (2005)
7. Buneman, P.: The recovery of trees from measures of dissimilarity. In: Kendall, D., Tauta, P. (eds.) Mathematics in the Archaeological and Historical Sciences, pp. 387–395. Edinburgh University Press, Edinburgh (1971)

8. Chataigner, F.: Approximating the maximum agreement forest on k trees. Inf. Process. Lett. **93**, 239–244 (2005)
9. Chen, J., Fan, J.-H., Sze, S.-H.: Parameterized and approximation algorithms for maximum agreement forest in multifurcating trees. Theor. Comput. Sci. **562**, 496–512 (2015)
10. Chen, Z., Wang, L.: Algorithms for reticulate networks of multiple phylogenetic trees. IEEE/ACM Trans. Comput. Biol. Bioinform. **9**, 372–384 (2012)
11. Diestel, R.: Graph Theory, 4th edition. Graduate Texts in Mathematics, vol. 173. Springer, Heidelberg (2010)
12. Dudas, G., Bedford, T., Lycett, S., Rambaut, A.: Reassortment between influenza B lineages and the emergence of a coadapted PB1-PB2-HA gene complex. Mol. Biol. Evol. **32**(1), 162–172 (2014). (supplemental information)
13. Hallett, M., McCartin, C.: A faster FPT algorithm for the maximum agreement forest problem. Theory Comput. Syst. **41**(3), 539–550 (2007)
14. Hein, J., Jiang, T., Wang, L., Zhang, K.: On the complexity of comparing evolutionary trees. Discrete Appl. Math. **71**, 153–169 (1996)
15. Li, M., Tromp, J., Zhang, L.: On the nearest neighbour interchange distance between evolutionary trees. J. Theor. Biol. **182**(4), 463–467 (1996)
16. Mukhopadhyay, A., Bhabak, P.: A 3-factor approximation algorithm for a minimum acyclic agreement forest on k rooted, binary phylogenetic trees. CoRR abs/1407.7125 (2014)
17. Robinson, D., Foulds, L.: Comparison of phylogenetic trees. Math. Biosci. **53**(1–2), 131–147 (1981)
18. Rodrigues, M., Sagot, M., Wakabayashi, Y.: Some approximation results for the maximum agreement forest problem. In: Proceedigs of the RANDOM-APPROX 2001, Lecture Notes in Computer Science, vol. 2129, pp. 159–169 (2001)
19. Rodrigues, E., Sagot, M., Wakabayashi, Y.: The maximum agreement forest problem: approximation algorithms and computational experiments. Theor. Comput. Sci. **374**, 91–110 (2007)
20. Shi, F., Wang, J., Chen, J., Feng, Q., Guo, J.: Algorithms for parameterized maximum agreement forest problem on multiple trees. Theor. Comput. Sci. **554**, 207–216 (2014)
21. Shi, F., Feng, Q., You, J., Wang, J.: Improved approximation algorithm for maximum agreement forest of two rooted binary phylogenetic trees. J. Comb. Optim. (2015a). doi:10.1007/s10878-015-9921-7
22. Shi, F., Wang, J., Yang, Y., Feng, Q., Li, W., Chen, J.: A fixed-parameter algorithm for the maximum agreement forset problem on multifurcating trees. Sci. China Inf. Sci. (2015b). doi:10.1007/s11432-015-5355-1
23. Swofford, D., Olsen, G., Waddell, P., Hillis, D.: Phylogenetic inference. In: Hillis, D., Moritz, D., Mable, B. (eds.) Molecular Systematics, 2nd edn, pp. 407–514. Sinauer Associates, Sunderiand (1996)
24. Whidden, C., Zeh, N.: A unifying view on approximation and FPT of agreement forests. In: Proceedings of the WABI 2009, Lecture Notes in Computer Science, vol. 5724, pp. 390–401 (2009)
25. Whidden, C., Beiko, R.G., Zeh, N.: Fixed-parameter algorithms for maximum agreement forests. SIAM J. Comput. **42**(4), 1431–1466 (2013)
26. Whidden, C., Zeh, N., Beiko, R.G.: Supertrees based on the subtree prune-and-regraft distance. Syst. Biol. **63**(4), 566–581 (2014)
27. Whidden, C., Matsen IV, F.A.: Quantifying MCMC exploration of phylogenetic tree space. Syst. Biol. **64**(3), 472 (2015)
28. Wu, Y.: Close lower and upper bounds for the minimum reticulate network of multiple phylogenetic trees. Bioinformatics **26**(12), i140–i148 (2010)