A near-linear kernel for bounded-state parsimony distance

Deen, Elise; van Iersel, Leo; Janssen, Remie; Jones, Mark; Murakami, Yukihiro; Zeh, Norbert

DOI
10.1016/j.jcss.2023.103477

Publication date
2024

Document Version
Final published version

Published in
Journal of Computer and System Sciences

Citation (APA)
Deen, E., van Iersel, L., Janssen, R., Jones, M., Murakami, Y., & Zeh, N. (2024). A near-linear kernel for bounded-state parsimony distance. *Journal of Computer and System Sciences*, *140*, Article 103477. https://doi.org/10.1016/j.jcss.2023.103477

Journal of Computer and System Sciences 140 (2024) 103477

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

journal homepage: www.elsevier.com/locate/jcss

A near-linear kernel for bounded-state parsimony distance ☆,☆☆

Elise Deen [a], Leo van Iersel [a], Remie Janssen [b], Mark Jones [a,*], Yukihiro Murakami [a], Norbert Zeh [c]

[a] Delft Institute of Applied Mathematics, Delft University of Technology, the Netherlands
[b] National Institute for Public Health and the Environment (RIVM), Bilthoven, the Netherlands
[c] Faculty of Computer Science, Dalhousie University, Halifax, Canada

ARTICLE INFO

Article history:
Received 1 December 2022
Received in revised form 17 July 2023
Accepted 5 September 2023
Available online 26 September 2023

Keywords:
Parameterized complexity
Kernelization
Phylogenetics
Phylogenetic tree
Distance measure
Parsimony
Maximum parsimony distance

ABSTRACT

The maximum parsimony distance $d_{\mathrm{MP}}(T_1, T_2)$ and the bounded-state maximum parsimony distance $d^t_{\mathrm{MP}}(T_1, T_2)$ measure the difference between two phylogenetic trees T_1, T_2 in terms of the maximum difference between their parsimony scores for any character (with t a bound on the number of states in the character, in the case of $d^t_{\mathrm{MP}}(T_1, T_2)$). While computing $d_{\mathrm{MP}}(T_1, T_2)$ was previously shown to be fixed-parameter tractable with a linear kernel, no such result was known for $d^t_{\mathrm{MP}}(T_1, T_2)$. In this paper, we prove that computing $d^t_{\mathrm{MP}}(T_1, T_2)$ is fixed-parameter tractable for all t. Specifically, we prove that this problem has a kernel of size $O(k \lg k)$, where $k = d^t_{\mathrm{MP}}(T_1, T_2)$. As the primary analysis tool, we introduce the concept of leg-disjoint incompatible quartets, which may be of independent interest.

1. Introduction

Parsimony [6] is a popular tool in bioinformatics used to measure how closely a phylogenetic tree T matches some data associated with its leaves (e.g., DNA sequences of the taxa represented by the leaves). Abstractly, given a labelling $f : \mathcal{L}(T) \to S$, called a *character*, where $\mathcal{L}(T)$ is the set of leaves of T, and S is a set of labels or *states*, the goal is to extend this labelling to the internal vertices of the tree so that the number of edges whose endpoints have different labels is minimized. Intuitively, these edges, called *mutation edges*, reflect the number of mutation events necessary to explain the observed data under the assumption that the tree reflects the evolution of the taxa represented by the leaves (with internal vertices representing speciation events).

Recently, the maximum parsimony distance d_{MP} has been introduced (independently by Fischer and Kelk [5], and Moulton and Wu [13]) as a new measure of (dis)similarity of two phylogenetic trees T_1 and T_2 with the same leaf set $X = \mathcal{L}(T_1) = \mathcal{L}(T_2)$. This distance is defined as the maximum difference between the parsimony scores of the two trees, where the maximum is taken over all possible characters.

☆ Research of Leo van Iersel and Mark Jones was partially funded by Netherlands Organisation for Scientific Research (NWO) grant OCENW.KLEIN.125.
☆☆ Research supported by the Natural Sciences and Engineering Research Council of Canada.

* Corresponding author.
E-mail address: m.e.l.jones@tudelft.nl (M. Jones).

https://doi.org/10.1016/j.jcss.2023.103477

Fischer and Kelk also introduced the bounded-state variant d^t_{MP}, where the maximum is taken over all possible characters with at most t states. They proved that the problems of computing d_{MP} and d^t_{MP} for $t \geq 2$ are both NP-hard [5], and that this holds even when the trees are binary [9].

Appealing properties of these similarity measures include that they are related to the popular optimization criterion maximum parsimony as well as to rearrangement operations as subtree prune and regraft (SPR) and tree bisection and reconnection (TBR) [4,5]. In addition, from a computational perspective it is useful that lower bounds can easily be computed by considering a particular character. This contrasts the situation for SPR and TBR distance where upper bounds can be found by providing a sequence of SPR/TBR moves turning T_1 into T_2.

For d_{MP}, which does not impose a bound on the number of states used by the optimal character, some algorithmic results are known. Kelk and Stamoulis [12] gave a single-exponential algorithm for calculating $d_{\mathrm{MP}}(T_1, T_2)$ with running time $O(\phi^n \cdot \mathrm{poly}(n))$, where n denotes the number of taxa in T_1 and T_2, and $\phi \approx 1.618$ is the golden ratio. Kelk et al. [10] showed that the well-known cherry and chain reduction rules (with minimum chain length 4) are safe for d_{MP}. These rules were previously used (with chain length 3) to give a linear kernel for the tree bisection and reconnection (TBR) distance $d_{\mathrm{TBR}}(T_1, T_2)$ [1]. As observed by Kelk et al. [10], the results of [1,10,12] together imply that d_{MP} is fixed-parameter tractable (FPT) with respect to $d_{\mathrm{TBR}}(T_1, T_2)$. Finally, Jones, Kelk, and Stougie [8] proved that the kernel produced by the reduction rules of Kelk et al. has a size that is linear also in $d_{\mathrm{MP}}(T_1, T_2)$. The results of [8,10] imply in particular that $d_{\mathrm{TBR}}(T_1, T_2)$ and $d_{\mathrm{MP}}(T_1, T_2)$ differ by at most a constant factor, for any two trees T_1 and T_2 over the same set of taxa X [8, Theorem 5].

In this paper we focus on computing d^t_{MP}, the variant of d_{MP} where the number of states is bounded. This is arguably the most biologically relevant version since biological data usually has a bounded number of states (e.g. 4 for DNA). However, to the best of our knowledge, for d^t_{MP} no results beyond NP-hardness were known prior to this paper. Our main result is that the maximum t-state parsimony distance $d^t_{\mathrm{MP}}(T_1, T_2)$ between two trees T_1 and T_2 has a near-linear kernel. Specifically, we show that there exists a polynomial-time algorithm reducing a pair of trees (T_1, T_2) to a pair (T'_1, T'_2) such that $d^t_{\mathrm{MP}}(T'_1, T'_2) = d^t_{\mathrm{MP}}(T_1, T_2)$, and T'_1 and T'_2 have $O(k \lg k)$ leaves,[1] where $k = d^t_{\mathrm{MP}}(T_1, T_2)$. We prove this in two steps:

First, we prove that the reduction rules used by Kelk et al. [10] are safe also for d^t_{MP}. This implies that d^t_{MP} has a kernel for which the number of leaves $|X|$ is linear in $d_{\mathrm{TBR}}(T_1, T_2)$. Proving this follows the same ideas used by Kelk el al. but requires some care to make the arguments work for as few as two states; the arguments used by Kelk et al. relied on constructing a character with a potentially large number of states. Moreover, our proof also implies that the reduction rules used by Kelk et al. are safe for d_{MP}, but it is significantly shorter than the original proof by Kelk et al.

Second, we prove that $d_{\mathrm{TBR}}(T_1, T_2) \in O(d^t_{\mathrm{MP}}(T_1, T_2) \cdot \lg |X|)$. These two results imply that the kernel has size $O(k \lg k)$, where $k = d^t_{\mathrm{MP}}(T_1, T_2)$. The constants in our construction are fairly large, and we do believe that they can be improved. This does, however, require additional insights into how to prove that a large kernel implies that the parsimony distance between the two trees is high.

Our proof that $d_{\mathrm{TBR}}(T_1, T_2) \in O(d^t_{\mathrm{MP}}(T_1, T_2) \cdot \lg |X|)$ is noteworthy for two reasons. First, while the previous result on kernelization for d_{MP} implies that $d_{\mathrm{TBR}}(T_1, T_2) \in O(d_{\mathrm{MP}}(T_1, T_2))$, it establishes this relationship indirectly, via the linear kernel. In contrast, our proof starts with an agreement forest (AF), which provides an upper bound on the TBR distance [1], and then uses this AF to construct a large set of incompatible quartets that lead to a high parsimony distance.

Second, the proof that the kernel produced by the reduction rules by Kelk et al. has size linear in $d_{\mathrm{MP}}(T_1, T_2)$ [8] relied on finding pairwise *disjoint* conflicting quartets between the two trees. Each such quartet contributes 1 to $d_{\mathrm{MP}}(T_1, T_2)$, so to show that $d_{\mathrm{MP}}(T_1, T_2) \geq k'$ it is enough to find k' pairwise disjoint conflicting quartets. Our key insight is that it suffices to construct a set of incompatible quartets that satisfy a much weaker disjointness condition in one of the two trees and can interact arbitrarily in the other tree. Such a set of quartets does not give a parsimony distance that is at least the number of quartets, but the parsimony distance is still linear in the number of quartets. We present a primal-dual algorithm based on an ILP formulation of the maximum agreement forest problem [14] that finds such a set Q of incompatible quartets and an AF of size $O(|Q| \cdot \lg |X|)$. This establishes the key claim that $d_{\mathrm{TBR}}(T_1, T_2) \in O(d^t_{\mathrm{MP}}(T_1, T_2) \cdot \lg |X|)$ mentioned earlier.

The remainder of this paper is organized as follows. Section 2 introduces the necessary terminology and notation, and discusses previous results we will build upon. Section 3 provides our proof that both cherry and chain reduction are safe for d^t_{MP}. Section 4 proves our bound of the size of the kernel as a function of d^t_{MP}. Section 5 offers conclusions and a discussion of future work.

2. Preliminaries

2.1. Definitions

Phylogenetic trees, induced subtrees, restrictions, pendant subtrees, and parents. Throughout this paper, a *tree on* X is an unrooted tree with leaf set X and whose internal vertices have degree at most 3. When X is clear from context, we refer to a tree on X simply as a *tree*. A *phylogenetic tree* on X is a tree on X with no vertices of degree 2.

Given a tree T on X and a subset $Y \subseteq X$, the *subtree of* T *induced by* Y, $T(Y)$, is the smallest subtree of T that contains all leaves in Y. The *restriction of* T *to* Y, $T|_Y$, is obtained from $T(Y)$ by suppressing all degree-2 vertices in $T(Y)$. To *suppress*

[1] Throughout this paper, we use the definition that $\lg x = \max(1, \log_2 x)$.

a degree-2 vertex v with neighbours u and w in a tree T is to remove v and its incident edges from T and add the edge (u, w) to T. The inverse operation is to *subdivide* an edge (u, w) in T by deleting the edge (u, w) from T and adding a new vertex v along with two edges (u, v) and (v, w) to T. Given a subset $Y \subseteq X$, a subtree T' of T is a *pendant subtree of* $T(Y)$ *in* T if T' and $T(Y)$ are vertex-disjoint, there exists an edge (u, v) in T with u a vertex of $T(Y)$ and v a vertex of T', and no other edge has exactly one vertex in T'. Though T and T' are unrooted, we call v the *root* of the pendant subtree T'.

Every leaf v of a tree T has a unique neighbour, which we call the *parent* of v even though T is unrooted.

Cherries and quartets. A *cherry* of a tree T on X is a pair of leaves (a, b) of T with the same parent.

A *quartet* of a tree T on X is a subset $\{a, b, c, d\} \subseteq X$ of size 4. If the path from a to b in T is disjoint from the path from c to d in T, then the restriction $T|_{\{a,b,c,d\}}$ of T to $\{a, b, c, d\}$ has the two cherries (a, b) and (c, d). We write $T|_{\{a,b,c,d\}} = ab|cd$ in this case.

A quartet q is *compatible* with a pair of trees (T_1, T_2) on X if $T_1|_q = T_2|_q$. Otherwise, q is *incompatible* with (T_1, T_2).

Tree bisection and reconnect distance and agreement forests. A tree bisection and reconnect (TBR) operation [1] on a phylogenetic tree T deletes an arbitrary edge (u, v) from T, thereby splitting T into two subtrees T_u and T_v that contain u and v, respectively. It then subdivides some edge in T_u and some edge in T_v, thereby creating two new vertices $u' \in T_u$ and $v' \in T_v$, and reconnects T_u and T_v by adding the edge (u', v'). Finally, it suppresses u and v (which have degree 2 after deleting the edge (u, v)). If u is a leaf, then there is no edge to subdivide in T_u. In this case, we set $u' = u$ and do not suppress u after adding the edge (u', v'). The case when v is a leaf is handled similarly. 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 necessary to transform T_1 into T_2.

An *agreement forest* (AF) of two trees T_1 and T_2 on X is a partition $F = \{X_1, \ldots, X_k\}$ of X such that

- $T_1|_{X_i} = T_2|_{X_i}$, for all $1 \le i \le k$,
- $T_1(X_i)$ and $T_1(X_j)$ are edge-disjoint for all $1 \le i < j \le k$, and
- $T_2(X_i)$ and $T_2(X_j)$ are edge-disjoint for all $1 \le i < j \le k$.

A *maximum agreement forest* (MAF) of T_1 and T_2 is an agreement forest with the minimum number of components $X_1, \ldots, X_k$. We refer to this number of components as the *size* $|F|$ of the forest. It was shown by Allen and Steel [1] that $d_{\mathrm{TBR}}(T_1, T_2) = |F| - 1$, for any MAF F of T_1 and T_2.

Characters, extensions, states, parsimony, and maximum parsimony distance. Given a tree T on X, a *character on* X is a mapping $f : X \to S$, for some non-empty set S. We call the elements of S *states*. We say that f is a *t-state character* if $|S| = t$. Note that there is no requirement that $f(X) = S$.

An *extension* of a character f on X to T is a labelling $\bar{f} : V(T) \to S$, where $V(T)$ denotes the set of vertices of T, such that $f(v) = \bar{f}(v)$ for every leaf $v \in X$.

A *mutation edge* of T with respect to some extension $\bar{f}$ is an edge (u, v) such that $\bar{f}(u) \neq \bar{f}(v)$. We use $\Delta_{\bar{f}}(T)$ to denote the number of mutation edges of T with respect to $\bar{f}$.

The *parsimony score* $l_f(T)$ of T with respect to some character f is defined as $l_f(T) = \min_{\bar{f}} \Delta_{\bar{f}}(T)$, where the minimum is taken over all extensions $\bar{f}$ of f. We call an extension $\bar{f}$ of f to T *optimal* if $\Delta_{\bar{f}}(T) = l_f(T)$.

The *(unbounded-state) maximum parsimony distance* $d_{\mathrm{MP}}(T_1, T_2)$ between two trees on X is defined as $d_{\mathrm{MP}}(T_1, T_2) = \max_f |l_f(T_1) - l_f(T_2)|$, where the maximum is taken over all characters on X. The *t-state maximum parsimony distance* $d^t_{\mathrm{MP}}(T_1, T_2)$ between T_1 and T_2 is defined analogously, but the maximum is taken only over all t-state characters on X. Throughout this paper, we refer to the unbounded-state maximum parsimony distance $d_{\mathrm{MP}}(T_1, T_2)$ as $d^{\infty}_{\mathrm{MP}}(T_1, T_2)$ to make it explicit that it imposes no upper bound on the number of states used by the optimal character.

All results in this paper apply to d^t_{MP} for any $t \in \mathbb{N} \cup \{\infty\}$ that satisfies $t \ge 2$. We refer to this set of valid values of t as $\mathbb{N}^{\infty}_{\ge 2}$.

Parameterized problems, kernelization, and reduction rules. A *parameterized problem* is a language $\mathcal{L} \subseteq \Sigma^* \times \mathbb{N}$, where Σ is a fixed, finite alphabet. For an instance $(\sigma, k) \in \Sigma^* \times \mathbb{N}$, k is called the *parameter* of (σ, k). We call (σ, k) a *yes-instance* if $(\sigma, k) \in \mathcal{L}$. Otherwise, (σ, k) is a *no-instance*. In the case of parsimony distance, the string σ encodes the pair of trees (T_1, T_2) and the bound t on the number of states, so we refer to an instance (σ, k) as the instance (T_1, T_2, t, k). This instance is a yes-instance if $d^t_{\mathrm{MP}}(T_1, T_2) \le k$.

A *kernelization algorithm*, or simply *kernel*, for some parameterized problem $\mathcal{L}$ is a polynomial-time algorithm which given an instance $(\sigma, k) \in \Sigma^* \times \mathbb{N}$, computes another instance $(\sigma', k') \in \Sigma^* \times \mathbb{N}$ such that

- $(\sigma, k) \in \mathcal{L}$ if and only if $(\sigma', k') \in \mathcal{L}$,
- $k' \le k$, and
- The size $|\sigma'| + k'$ of (σ', k') is bounded by $f(k)$, where f is some computable function $f : \mathbb{N} \to \mathbb{N}$.

The function f is called the *size* of the kernel.

Kernels are often obtained using repeated application of reduction rules. A *safe reduction rule* is a polynomial-time algorithm which given an instance $(\sigma, k) \in \Sigma^* \times \mathbb{N}$, computes a strictly smaller instance $(\sigma', k') \in \Sigma^* \times \mathbb{N}$ such that $(\sigma, k) \in \mathcal{L}$

if and only if $(\sigma', k') \in \mathcal{L}$. A reduction rule comes with a condition or conditions that need to be satisfied for this rule to be applicable. An instance (σ, k) is *fully reduced* with respect to a set of reduction rules if none of these rules is applicable to (σ, k), that is, if (σ, k) does not satisfy the conditions associated with any of the reduction rules. A kernelization algorithm based on a set of reduction rules repeatedly applies these rules until it obtains a fully reduced instance with respect to these rules. This is the kernel the algorithm returns.

2.2. Fitch's algorithm

An optimal extension of a character on a tree T can be computed in polynomial time using the Fitch-Hartigan algorithm [6,7]. The algorithm subdivides an arbitrary edge of T and uses the vertex this introduces as the root of the tree, thereby defining a parent-child relationship on the vertices of the tree. The algorithm now proceeds in two phases:

The *bottom-up phase* assigns a candidate set of states to every vertex of T. For a leaf v with state $f(v)$, its candidate set of states is $F(v) = \{f(v)\}$. For an internal vertex u with children v and w, its candidate set $F(u)$ is defined as

$$F(u) = \begin{cases} F(v) \cup F(w) & \text{if } F(v) \cap F(w) = \emptyset \\ F(v) \cap F(w) & \text{if } F(v) \cap F(w) \neq \emptyset \end{cases}.$$

In the first case, we call u a *union vertex*. In the second case, we call it an *intersection vertex*.

The function $F : V(T) \to 2^S$ is called the *Fitch map* of f, and we will refer to the set $F(v)$ associated with a vertex v as v's *Fitch set*.

The second, *top-down phase* uses the Fitch map to compute an optimal extension $\bar{f}$ of f to T: For the root r of T, we choose an arbitrary state $\bar{f}(r) \in F(r)$. For any other vertex v with parent u, we choose $\bar{f}(v) = \bar{f}(u)$ if $\bar{f}(u) \in F(v)$. Otherwise, we choose $\bar{f}(v)$ to be an arbitrary state in $F(v)$. Finally, we suppress the root of T that was introduced at the start of the algorithm, keeping the same assignment of states to all other vertices. Note that this does not change the number of mutation edges in T, as the root is always assigned a state that is assigned to at least one of its children.

We call an extension $\bar{f}$ computed using the Fitch-Hartigan algorithm a *Fitch extension* of f. We will also refer to an extension $\bar{f}$ of f to the rooted version of T, before suppressing r, as a Fitch extension of f. The meaning will be clear from context. Note that there are optimal extensions of f that are not Fitch extensions.

Lemma 1 *(Hartigan [7]). A Fitch extension $\bar{f}$ of f is an optimal extension of f, that is, $\Delta_{\bar{f}}(T) = l_f(T)$. Moreover, $\Delta_{\bar{f}}(T)$ equals the number of union vertices in the rooted version of T with respect to f's Fitch map F.*

Lemma 1 justifies a slight overload of notation: We use $l_F(T)$ to denote the number of union vertices of T with respect to the Fitch map F. By Lemma 1, $l_F(T) = l_f(T)$.

2.3. Characters on induced subtrees and restricted subtrees

In this section, we prove some simple results on the parsimony scores and maximum parsimony distance of restrictions of trees on X to subsets of their leaves. These results will be used to prove that the reduction rules in Section 3 are safe.

Lemma 2. *Let T be a tree on X, let $Y \subseteq X$, and let f be a character on Y. Then $l_f(T(Y)) = l_f(T|_Y)$.*

Proof. We show first that $l_f(T(Y)) \leq l_f(T|_Y)$. Consider an optimal extension $\bar{f}$ of f to $T|_Y$. We define an extension $\tilde{f}$ of f to $T(Y)$ such that $\Delta_{\tilde{f}}(T(Y)) = \Delta_{\bar{f}}(T|_Y) = l_f(T|_Y)$. Since $l_f(T(Y)) \leq \Delta_{\tilde{f}}(T(Y))$, it follows that $l_f(T(Y)) \leq l_f(T|_Y)$.

By definition, $T(Y)$ can be obtained from $T|_Y$ by subdividing edges, that is, by replacing edges with paths whose internal vertices have degree 2. Every vertex $v \in T|_Y$ is also a vertex of $T(Y)$. For any such vertex v, we define $\tilde{f}(v) = \bar{f}(v)$. For every edge (u, v) of $T|_Y$ that is replaced by a path $(u, p_1, \ldots, p_k, v)$ in $T(Y)$, we let $\tilde{f}(p_i) = \bar{f}(u)$ for all $1 \leq i \leq k$. The only mutation edge on the path $(u, p_1, \ldots, p_k, v)$, if there is any, is the edge (p_k, v) because $\tilde{f}(u) = \tilde{f}(p_1) = \cdots = \tilde{f}(p_k)$. If the edge (p_k, v) is a mutation edge, then $\bar{f}(u) = \tilde{f}(u) = \tilde{f}(p_k) \neq \tilde{f}(v) = \bar{f}(v)$, that is, the edge (u, v) also is a mutation edge with respect to $\bar{f}$. This shows that $\Delta_{\tilde{f}}(T(Y)) = \Delta_{\bar{f}}(T|_Y)$.

To show that $l_f(T|_Y) \leq l_f(T(Y))$, let $\tilde{f}$ be an optimal extension of f to $T(Y)$. We obtain an extension $\bar{f}$ of f to $T|_Y$ as the restriction of $\tilde{f}$ to $T|_Y$. Now consider any edge (u, v) of $T|_Y$. If $\bar{f}(u) \neq \bar{f}(v)$, then $\tilde{f}(u) \neq \tilde{f}(v)$. Therefore, the path $(u, p_1, \ldots, p_k, v)$ in $T(Y)$ corresponding to (u, v) must contain at least one mutation edge. Thus, $l_f(T|_Y) \leq \Delta_{\bar{f}}(T|_Y) \leq \Delta_{\tilde{f}}(T(Y)) = l_f(T(Y))$. □

The next corollary follows immediately:

Corollary 3. *Let T_1 and T_2 be trees on X, and let $Y \subseteq X$. Then $d^t_{MP}(T_1|_Y, T_2|_Y) = d^t_{MP}(T_1(Y), T_2(Y))$ for any $t \in \mathbb{N}^{\infty}_{\geq 2}$.*

Lemma 4. *Let T be a tree on X, let $Y \subseteq X$, let f be a character on X, and let f' be the restriction of f to Y. Then $l_{f'}(T(Y)) \leq l_f(T)$.*

Proof. Let $\bar{f}$ be an optimal extension of f to T. The restriction of $\bar{f}$ to $T(Y)$ is an extension $\bar{f}'$ of f' to $T(Y)$. Every mutation edge in $T(Y)$ with respect to $\bar{f}'$ is also a mutation edge in T with respect to $\bar{f}$. Thus, $l_{f'}(T(Y)) \leq \Delta_{\bar{f}'}(T(Y)) \leq \Delta_{\bar{f}}(T) = l_f(T)$. □

Given an induced subtree $T(Y)$ and a labelling $\bar{f}$ of the vertices of $T(Y)$, we define the *parsimonious extension* of $\bar{f}$ to T as the unique labelling $\tilde{f}$ of the vertices in T such that $\tilde{f}(v) = \bar{f}(v)$ for all $v \in T(Y)$ and $\tilde{f}(v) = \bar{f}(w_v)$ for all $v \notin T(Y)$, where w_v is the vertex in $T(Y)$ closest to v. It follows immediately that there are no mutation edges of T with respect to $\tilde{f}$ that do not belong to $T(Y)$, and that an edge of $T(Y)$ is a mutation edge with respect to $\tilde{f}$ if and only if it is a mutation edge with respect to $\bar{f}$. Thus, we have the following observation:

Observation 5. *If T is a tree on X, $Y \subseteq X$, $\bar{f}$ is a labelling of the vertices of $T(Y)$, and $\tilde{f}$ is the parsimonious extension of $\bar{f}$ to T, then $\Delta_{\bar{f}}(T(Y)) = \Delta_{\tilde{f}}(T)$.*

The following result is a bounded-state analogue of Corollary 3.5 in [13].

Lemma 6. *Let T_1 and T_2 be trees on X, and let $Y \subseteq X$. Then $d^t_{MP}(T_1(Y), T_2(Y)) = d^t_{MP}(T_1|_Y, T_2|_Y) \leq d^t_{MP}(T_1, T_2)$, for any $t \in \mathbb{N}^{\infty}_{\geq 2}$.*

Proof. Corollary 3 states that $d^t_{\mathrm{MP}}(T_1(Y), T_2(Y)) = d^t_{\mathrm{MP}}(T_1|_Y, T_2|_Y)$. Thus, it suffices to prove that $d^t_{\mathrm{MP}}(T_1(Y), T_2(Y)) \leq d^t_{\mathrm{MP}}(T_1, T_2)$.

Let f be a t-state character on Y such that $d^t_{\mathrm{MP}}(T_1(Y), T_2(Y)) = |l_f(T_1(Y)) - l_f(T_2(Y))|$. Moreover, assume that $l_f(T_1(Y)) \leq l_f(T_2(Y))$, so $d^t_{\mathrm{MP}}(T_1(Y), T_2(Y)) = l_f(T_2(Y)) - l_f(T_1(Y))$. Let $\bar{f}$ be an optimal extension of f to $T_1(Y)$, let $\tilde{f}$ be the parsimonious extension of $\bar{f}$ to T_1, and let f' be the restriction of $\tilde{f}$ to the leaves of T_1 (i.e., to X). Thus, f is the restriction of f' to Y. Then $l_{f'}(T_1) \leq \Delta_{\tilde{f}}(T_1) = \Delta_{\bar{f}}(T_1(Y)) = l_f(T_1(Y))$, by Observation 5. By Lemma 4, we have $l_f(T_2(Y)) \leq l_{f'}(T_2)$. Thus, $d^t_{\mathrm{MP}}(T_1, T_2) \geq l_{f'}(T_2) - l_{f'}(T_1) \geq l_f(T_2(Y)) - l_f(T_1(Y)) = d^t_{\mathrm{MP}}(T_1(Y), T_2(Y))$. □

3. Reduction rules

Previous kernelization results for SPR distance [2], TBR distance [1], and hybridization number [3] employ two simple reduction rules: cherry reduction and chain reduction (see below). It was shown that these rules produce kernels of size linear in the SPR distance [2], TBR distance [1] or hybridization number [3] of the two input trees.[2] Kelk et al. [10] proved that these rules are safe also for the unbounded-state maximum parsimony distance, as long as chain reduction is applied only to chains of length greater than 4. Jones, Kelk, and Stougie [8] proved that these reduction rules once again produce a kernel of size linear in $d^{\infty}_{\mathrm{MP}}(T_1, T_2)$.

In this section, we prove that cherry reduction and chain reduction are safe also for the t-state parsimony distance, for any $t \geq 2$, again as long as we apply chain reduction only to chains of length greater than 4. This shows that there exists a linear-size kernel for $d^t_{\mathrm{MP}}(T_1, T_2)$ parameterized by the TBR distance $d_{\mathrm{TBR}}(T_1, T_2)$, as summarized in the following theorem:

Theorem 7. *There exists a set of safe reduction rules for d^t_{MP} such that the two trees T_1 and T_2 on X in a fully reduced instance (T_1, T_2, t, k) satisfy $|X| \leq 20 \cdot d_{TBR}(T_1, T_2)$, for any $t \in \mathbb{N}^{\infty}_{\geq 2}$.*

Kelk and Linz proved that a fully reduced instance with respect to cherry reduction and chain reduction applied to chains of length greater than 3 has size at most $15 \cdot d_{\mathrm{TBR}}(T_1, T_2) - 9$ [11]. Since we apply chain reduction only to chains of length greater than 4, we obtain a kernel that is up to a factor of $\frac{4}{3}$ bigger, which gives the bound of $\frac{4}{3} \cdot 15 \cdot d_{\mathrm{TBR}}(T_1, T_2) = 20 \cdot d_{\mathrm{TBR}}(T_1, T_2)$ on the size of a fully reduced instance in Theorem 7.

3.1. Cherry reduction

Cherry reduction eliminates common cherries of the two input trees:

Reduction Rule 8 *(Cherry Reduction). If T_1 and T_2 have a common cherry (x, y), that is, if x and y are two leaves that have the same parent in both T_1 and T_2, then remove y from both T_1 and T_2 and suppress the parent of x and y.*

[2] Technically, for hybridization number, the reduction rules by Bordewich and Semple [3] do not yield a linear kernel but a linear *compression*: the reduction takes an instance of the maximum acyclic agreement forest (MAAF) problem, which is equivalent to hybridization number, and produces an instance of linear size of a *weighted* version of the MAAF problem.

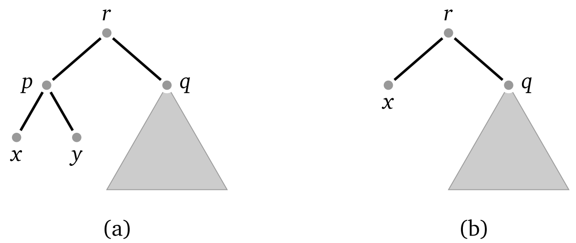


Fig. 1. The rooted version of the tree T_1 or T_2 in the input to cherry reduction (a) and the corresponding reduced tree T_1' or T_2' (b).

Another way to state cherry reduction is that we replace T_1 and T_2 with their restrictions to $X \setminus \{y\}$. The following lemma shows that applying cherry reduction to a pair of trees (T_1, T_2) does not change their maximum parsimony distance. We adopt the approach of [10]; a similar result is also implicit in [9].

Lemma 9. *If $T_1' = T_1|_{X\setminus\{y\}}$ and $T_2' = T_2|_{X\setminus\{y\}}$ are the two trees obtained from T_1 and T_2 by applying cherry reduction to a common cherry (x, y) of T_1 and T_2, then $d^t_{MP}(T_1', T_2') = d^t_{MP}(T_1, T_2)$, for any $t \in \mathbb{N}^{\infty}_{\geq 2}$.*

Proof. By Lemma 6, we have that $d^t_{\mathrm{MP}}(T_1', T_2') \leq d^t_{\mathrm{MP}}(T_1, T_2)$. Therefore, it is sufficient to show that $d^t_{\mathrm{MP}}(T_1', T_2') \geq d^t_{\mathrm{MP}}(T_1, T_2)$.

Since x and y have the same parent in both T_1 and T_2, we use p to refer to this common parent in both T_1 and T_2, considering it the same vertex whether it belongs to T_1 or T_2. Similarly, we consider the third neighbour of p in T_1 and T_2 to be the same vertex q. In other words, the neighbourhood of p in both T_1 and T_2 is $\{x, y, q\}$. In T_1' and T_2', y and p are removed, and q becomes x's parent. We argue about rooted versions of T_1, T_2, T_1', and T_2'. In T_1 and T_2, we subdivide the edge (p, q) using a new vertex r, and we make r the root of T_1 and T_2. After pruning y and suppressing p, this results in r being x's parent in T_1' and T_2'. See Fig. 1.

Let f be a t-state character on X with Fitch maps F_1 and F_2 on T_1 and T_2, respectively, such that

$$|l_{F_1}(T_1) - l_{F_2}(T_2)| = d^t_{\mathrm{MP}}(T_1, T_2).$$

Without loss of generality, assume that $l_{F_1}(T_1) \leq l_{F_2}(T_2)$, so $d^t_{\mathrm{MP}}(T_1, T_2) = l_{F_2}(T_2) - l_{F_1}(T_1)$. To prove that $d^t_{\mathrm{MP}}(T_1', T_2') \geq d^t_{\mathrm{MP}}(T_1, T_2)$, we construct a t-state character f' on $X' = X \setminus \{y\}$ whose Fitch maps F_1' and F_2' on T_1' and T_2' satisfy

$$l_{F_2'}(T_2') - l_{F_1'}(T_1') \geq l_{F_2}(T_2) - l_{F_1}(T_1).$$

This implies that

$$d^t_{\mathrm{MP}}(T_1', T_2') \geq l_{F_2'}(T_2') - l_{F_1'}(T_1') \geq l_{F_2}(T_2) - l_{F_1}(T_1) = d^t_{\mathrm{MP}}(T_1, T_2).$$

We define f' by choosing $f'(z) = f(z)$ for all $z \neq x$. We choose $f'(x)$ arbitrarily from $F_1(p) \cap F_1(q)$ if r is an intersection vertex in T_1. Otherwise, we choose $f'(x) = f(x)$. For every vertex $z \in T_i'$ such that $z \notin \{x, r\}$, we have $F_i'(z) = F_i(z)$ because any such vertex has the same set of descendant leaves in T_i and T_i' and any such descendant leaf z' satisfies $f'(z') = f(z')$ (see Fig. 1). Therefore, every vertex $z \neq r$ is a union vertex in T_i' if and only if it is a union vertex in T_i. Moreover, p is a union vertex in T_1 if and only if it is a union vertex in T_2. Thus,

$$l_{F_2'}(T_2') - l_{F_1'}(T_1') = l_{F_2}(T_2) - l_{F_1}(T_1) + (u_2' - u_2) - (u_1' - u_1),$$

where

$$u_i = \begin{cases} 1 & \text{if } r \text{ is a union vertex in } T_i \\ 0 & \text{otherwise} \end{cases}$$

and

$$u_i' = \begin{cases} 1 & \text{if } r \text{ is a union vertex in } T_i' \\ 0 & \text{otherwise,} \end{cases}$$

for $i \in \{1, 2\}$. Therefore,

$$l_{F_2'}(T_2') - l_{F_1'}(T_1') \geq l_{F_2}(T_2) - l_{F_1}(T_1)$$

if and only if

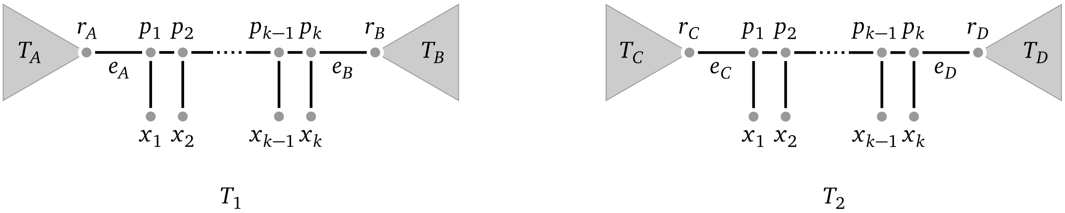


Fig. 2. The various definitions of vertices, edges, and subtrees in the discussion of chain reduction.

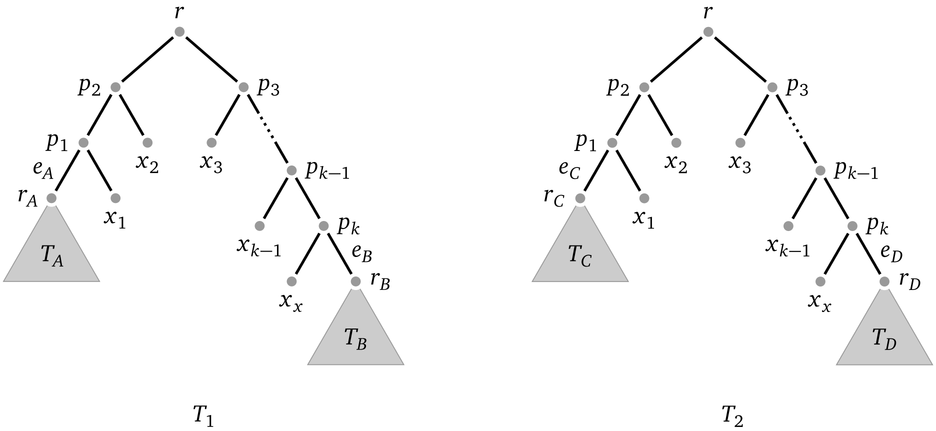


Fig. 3. The rooted versions of T_1 and T_2 considered for the discussion of chain reduction.

$$u_2' - u_2 \geq u_1' - u_1.$$

To prove that this inequality holds, note first that the choice of $f'(x)$ when r is an intersection vertex in T_1 ensures that r is an intersection vertex also in T_1'. Thus, $u_1' - u_1 \leq 0$.

Next observe that no matter whether $f'(x)$ is chosen from $F_1(p) \cap F_1(q)$ or $f'(x) = f(x)$, we have $f'(x) \in F_1(p) = F_2(p)$. Thus, if r is a union vertex in T_2, it is also a union vertex in T_2'. Therefore, $u_2' - u_2 \geq 0$.

Since $u_2' - u_2 \geq 0$ and $u_1' - u_1 \leq 0$, we have $u_2' - u_2 \geq u_1' - u_1$, as desired. □

3.2. Chain reduction

A *chain of length* k in a tree T is an ordered sequence of leaves $\langle x_1, \ldots, x_k \rangle$ such that $\langle p_1, \ldots, p_k \rangle$ is a path in T, where p_i is the parent of x_i in T, for all $1 \leq i \leq k$. It is possible to have $p_1 = p_2$ and/or $p_{k-1} = p_k$. If this is the case, the chain is called *pendant* in T. A *common chain* of T_1 and T_2 is an ordered sequence of leaves $\langle x_1, \ldots, x_k \rangle$ that is a chain in both T_1 and T_2. Chain reduction ensures that the trees in a fully reduced instance do not have long common chains:

Reduction Rule 10 *(Chain Reduction). If T_1 and T_2 have a common chain $\langle x_1, x_2, \ldots, x_k \rangle$ of length $k \geq 5$, then remove the leaves $x_3, \ldots, x_{k-2}$ from both T_1 and T_2, and suppress their parents in both trees.*

It was shown by Kelk et al. [10] that chain reduction preserves $d_{\mathrm{MP}}^{\infty}(T_1, T_2)$. The argument by Kelk et al. uses what was called a *less constrained roots argument* in that paper. Here we extend this argument to bounded-state characters to prove that chain reduction also preserves $d_{\mathrm{MP}}^{t}(T_1, T_2)$, for any finite t.

The tree T_1 consists of the chain $\langle x_1, \ldots, x_k \rangle$ plus two pendant subtrees T_A and T_B whose roots are adjacent to p_1 and p_k, respectively. See Fig. 2. If $\langle x_1, \ldots, x_k \rangle$ is a pendant chain of T_1, then T_A or T_B is empty, possibly both. Similarly, T_2 consists of the chain $(x_1, \ldots, x_k)$ plus two pendant subtrees T_C and T_D whose roots are adjacent to p_1 and p_k, respectively. Again, T_C or T_D may be empty, possibly both. For $P \in \{A, B, C, D\}$, let X_P be the set of leaves in T_P, let r_P be the root of T_P, and let e_P be the edge connecting r_P to p_1 or p_k. Obviously, $X_P = \emptyset$, and r_P and e_P do not exist, if T_P is empty. Note that $X_A \cup X_B = X_C \cup X_D = X \setminus \{x_1, \ldots, x_k\}$.

Throughout the remainder of this subsection, we consider rooted versions of T_1 and T_2, where the root r is placed on the edge (p_2, p_3) in both trees. See Fig. 3. This makes r the common parent of p_2 and p_{k-1} in the two trees obtained from T_1 and T_2 by applying chain reduction. Now fix an optimal t-state character $f : X \to S$, that is, a t-state character such that $d_{\mathrm{MP}}^{t}(T_1, T_2) = |l_f(T_1) - l_f(T_2)|$, and assume w.l.o.g. that $l_f(T_1) \leq l_f(T_2)$, so $d_{\mathrm{MP}}^{t}(T_1, T_2) = l_f(T_2) - l_f(T_1)$. For $i \in \{1, 2\}$, let F_i be the Fitch map of T_i defined by f, and let u_i be the number of union vertices among $r, p_1, \ldots, p_k$ in T_i. Finally, let

f_P be the restriction of f to T_P and let S_P be the Fitch set of r_P defined by the character f_P, for $P \in \{A, B, C, D\}$. If T_P is empty, then let $S_P = S$. Then

$$l_f(T_1) = l_{f_A}(T_A) + l_{f_B}(T_B) + u_1,$$
$$l_f(T_2) = l_{f_C}(T_C) + l_{f_D}(T_D) + u_2.$$

In particular,

$$d^t_{\mathrm{MP}}(T_1, T_2) = l_{f_C}(T_C) + l_{f_D}(T_D) - l_{f_A}(T_A) - l_{f_B}(T_B) + u_2 - u_1. \tag{1}$$

The less constrained roots argument now bounds the difference $u_2 - u_1$ depending on whether $S_A \subseteq S_C$ and $S_B \subseteq S_D$. The name refers to the fact that, for example, $S_A \subseteq S_C$ implies that the choice of the state $\bar{f}_2(r_C)$ in a Fitch extension $\bar{f}_2$ of f to T_2 is less constrained than the choice of $\bar{f}_1(r_A)$ in a Fitch extension $\bar{f}_1$ of f to T_1.

Lemma 11. *Let $\delta_{AC} = 0$ if $S_A \subseteq S_C$, and $\delta_{AC} = 1$ otherwise. Similarly, let $\delta_{BD} = 0$ if $S_B \subseteq S_D$, and $\delta_{BD} = 1$ otherwise. Then $u_2 - u_1 \le \delta_{AC} + \delta_{BD}$.*

Proof. Suppose first that all four subtrees T_A, T_B, T_C, T_D are non-empty, and consider a Fitch extension $\bar{f}_1$ of f to T_1. Then $\bar{f}_1(r_A) = a \in S_A$, and $\bar{f}_1(r_B) = b \in S_B$. We construct an extension $\bar{f}_2$ of f to T_2 as follows:

We start by setting $\bar{f}_2(v) = f(v)$ for every leaf $v \in X$. To define the labels of all internal vertices of T_2, we pick states $c \in S_C$ and $d \in S_D$, and set $\bar{f}_2(r_C) = c$ and $\bar{f}_2(r_D) = d$. If $S_A \subseteq S_C$, then we choose $c = a$. If $S_B \subseteq S_D$, then we choose $d = b$. We label the remaining vertices in T_C and T_D so that the restriction of $\bar{f}_2$ to T_C is a Fitch extension of f_C, and the restriction of $\bar{f}_2$ to T_D is a Fitch extension of f_D. Finally, we complete $\bar{f}_2$ by setting $\bar{f}_2(p_i) = \bar{f}_1(p_i)$ for all $1 \le i \le k$.[3]

This ensures that $T_1(\{x_1, \ldots, x_k\})$ and $T_2(\{x_1, \ldots, x_k\})$ contain the same mutation edges with respect to $\bar{f}_1$ and $\bar{f}_2$, respectively. The edge e_C is a mutation edge only if e_A is or $S_A \nsubseteq S_C$. The edge e_D is a mutation edge only if e_B is or $S_B \nsubseteq S_D$. Since the restrictions of $\bar{f}_1$ and $\bar{f}_2$ to T_A, T_B, T_C, and T_D are Fitch extensions of f_A, f_B, f_C, and f_D, this shows that

$$\Delta_{\bar{f}_2}(T_2) - \Delta_{\bar{f}_1}(T_1) \le l_{f_C}(T_C) + l_{f_D}(T_D) - l_{f_A}(T_A) - l_{f_B}(T_B) + \delta_{AC} + \delta_{BD}.$$

Since $\bar{f}_2$ is an extension of f to T_2 and $\bar{f}_1$ is a Fitch extension of f to T_1, we also have $l_f(T_2) \le \Delta_{\bar{f}_2}(T_2)$ and $l_f(T_1) = \Delta_{\bar{f}_1}(T_1)$. Thus,

$$d^t_{\mathrm{MP}}(T_1, T_2) = l_f(T_2) - l_f(T_1) \le l_{f_C}(T_C) + l_{f_D}(T_D) - l_{f_A}(T_A) - l_{f_B}(T_B) + \delta_{AC} + \delta_{BD}.$$

Together with (1), this shows that

$$u_2 - u_1 \le \delta_{AC} + \delta_{BD}.$$

To complete the proof, we consider the case when at least one of the subtrees T_A, T_B, T_C, T_D is empty. If $T_A = \emptyset$ or $T_C = \emptyset$, then we construct a *rooted* tree T' with t leaves. (Recall that $t = |S|$ is the number of available states.) If $T_A = \emptyset$, then we have $p_1 = p_2$. In this case, we subdivide the edge (p_2, x_1) in T_1 with a new vertex q, and add an edge between q and the root of T'. This effectively sets $T_A = T'$ and makes q the parent of x_1, that is, $p_1 = q$ after adding T' to T_1. If $T_A \neq \emptyset$, then we subdivide the edge (p_1, r_A) with a new vertex q and again add an edge between q and the root of T'. Similarly, in T_2 we add an edge between the root of T' and a new vertex q, where q subdivides the edge (p_2, x_1) or (p_1, r_C) depending on whether $S_C = \emptyset$. This is illustrated in Fig. 4. We add a tree T'' in a similar fashion if $T_B = \emptyset$ or $T_D = \emptyset$. Let T'_1 and T'_2 be the two trees obtained from T_1 and T_2 by the addition of T', and possibly T''; let f' be the character on the leaf set of T'_1 and T'_2 obtained by setting $f'(v) = f(v)$ for all $v \in X$, giving each leaf in T' a different label in S, and giving each leaf in T'' a different label in S; and let F'_1 and F'_2 be the Fitch maps of f' on T'_1 and T'_2, respectively. Similar to $p_1, \ldots, p_k$, we use q to denote the vertex adjacent to the root of T' in both T'_1 and T'_2.

Observe that every non-leaf vertex in T' is a union vertex in both T'_1 and T'_2, while q is an intersection vertex in both T'_1 and T'_2. Moreover, if $T_A = \emptyset$, then $F'_1(q) = \{f(x_1)\} = F_1(x_1)$; if $T_A \neq \emptyset$, then $F'_1(q) = F_1(r_A)$. Similarly, $F'_2(q) = F_2(x_1)$ if $T_C = \emptyset$, and $F'_2(q) = F_2(r_C)$ if $T_C \neq \emptyset$. This implies that $F_1(v) = F'_1(v)$ for every vertex $v \in T_1$, and $F_2(v) = F'_2(v)$ for every vertex $v \in T_2$. In particular, the addition of T' introduces the non-leaf vertices of T' as union vertices into T'_1 and T'_2 and apart from this, T_1 and T'_1 have the same sets of union vertices, as do T_2 and T'_2. By a similar argument, the addition of T'' if $T_B = \emptyset$ or $T_D = \emptyset$ introduces the same number of union vertices into both T'_1 and T'_2.

This implies that

$$l_f(T_2) - l_f(T_1) = l_{f'}(T'_2) - l_{f'}(T'_1).$$

[3] Note that this is well defined because $T_A, T_B, T_C, T_D \neq \emptyset$ implies that no two leaves x_i and x_j have the same parent in either T_1 or T_2.

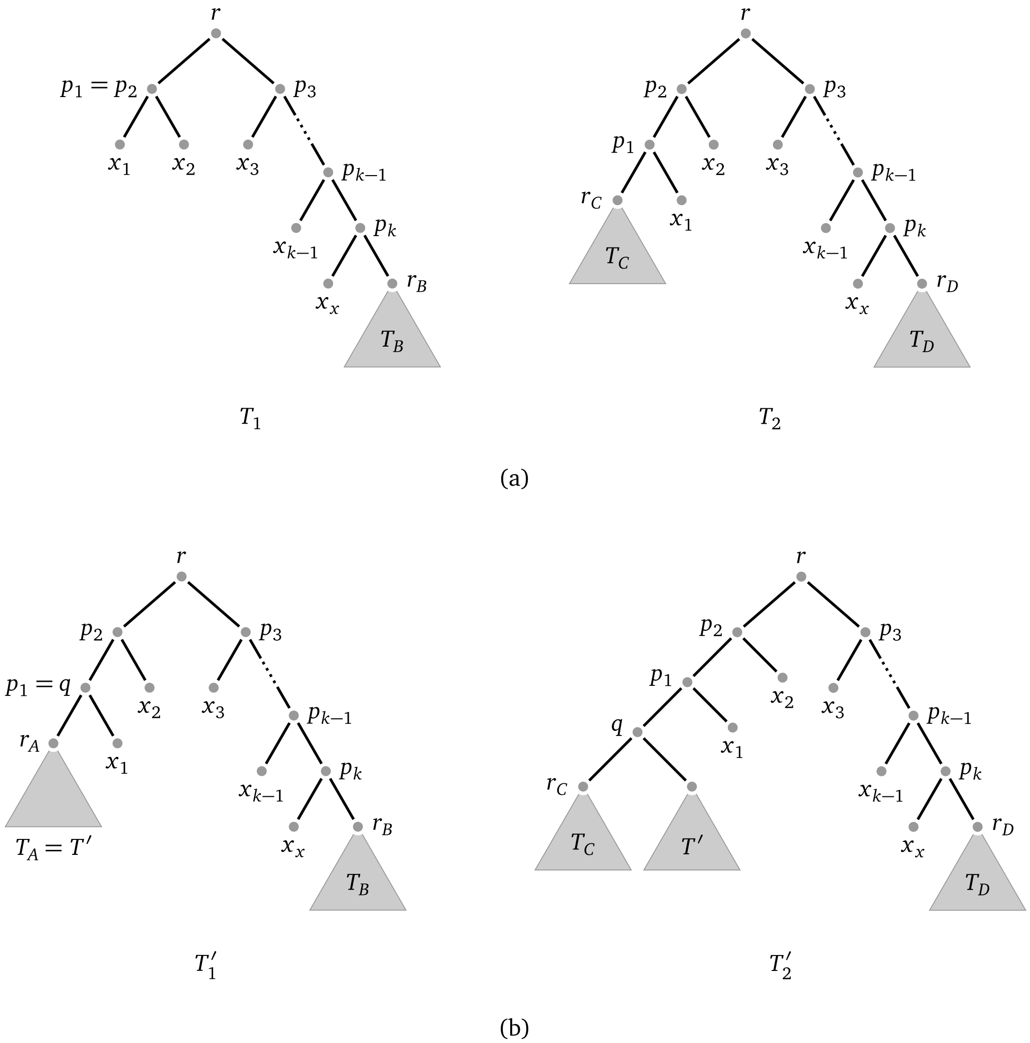


Fig. 4. (a) A common chain $\langle x_1, \dots, x_k \rangle$ of T_1 and T_2 that is pendant in T_1: T_A is empty. (b) The two trees T'_1 and T'_2 obtained by attaching a tree $T_A = T'$ to T_1 and adding T' to T_C.

Finally observe that the sets S_A, S_B, S_C, and S_D, and thus δ_{AC}, δ_{BD}, are the same for T_1 and T_2 as for T'_1 and T'_2. Indeed, if $T_A \neq \emptyset$, then $S_A = F_1(r_A) = F'_1(q)$. If $T_A = \emptyset$, then we define $S_A = S$ for T_1. In T'_1, $S_A = S$ because $T_A = T'$ in T'_1 and all internal vertices in T' are union vertices. In both cases, q plays the role of r_A in T'_1. Analogous arguments show that S_B, S_C, and S_D are the same for T_1 and T_2 as for T'_1 and T'_2.

By applying the case when T_A, T_B, T_C, T_D are all non-empty to T'_1 and T'_2, we conclude that

$$l_f(T_2) - l_f(T_1) = l_f(T'_2) - l_f(T'_1) \leq l_{f_C}(T_C) + l_{f_D}(T_D) - l_{f_A}(T_A) - l_{f_B}(T_B) + \delta_{AC} + \delta_{BD},$$

that is, once again,

$$u_2 - u_1 \leq \delta_{AC} + \delta_{BD}. \quad \square$$

We are ready to prove that chain reduction is safe now:

Lemma 12. *Let T_1 and T_2 be two trees on X, let $\langle x_1, \dots, x_k \rangle$ be a common chain of T_1 and T_2 of length $k \geq 5$, and let T'_1 and T'_2 be the two trees obtained by removing the leaves $x_3, \dots, x_{k-2}$ from both T_1 and T_2 and suppressing their parents. Then $d^t_{MP}(T_1, T_2) = d^t_{MP}(T'_1, T'_2)$ for all $t \in \mathbb{N}^{\infty}_{\geq 2}$.*

Proof. The proof is based on the proof by Kelk et al. [10] but presents the argument much more succinctly, and obviously makes adjustments to ensure that the proof is correct for t-state characters.

By Lemma 6, we have that $d^t_{\mathrm{MP}}(T_1', T_2') \le d^t_{\mathrm{MP}}(T_1, T_2)$. Therefore, it suffices to show that $d^t_{\mathrm{MP}}(T_1', T_2') \ge d^t_{\mathrm{MP}}(T_1, T_2)$. Let f be an optimal character for (T_1, T_2), and assume that $d^t_{\mathrm{MP}}(T_1, T_2) = l_f(T_2) - l_f(T_1)$. We construct a t-state character f' on the leaf set $X \setminus \{x_3, \dots, x_{k-2}\}$ of T_1' and T_2' such that $l_{f'}(T_2') - l_{f'}(T_1') \ge l_f(T_2) - l_f(T_1) = d^t_{\mathrm{MP}}(T_1, T_2)$. Since $d^t_{\mathrm{MP}}(T_1', T_2') \ge l_{f'}(T_2') - l_{f'}(T_1')$, this proves the claim. We define f' as

$$f'(v) = \begin{cases} a & \text{if } v \in \{x_1, x_2\} \\ b & \text{if } v \in \{x_{k-1}, x_k\} \\ f(v) & \text{otherwise,} \end{cases}$$

where $a, b \in S$ are appropriate states chosen as discussed below.

For this character, we use F_1' and F_2' to denote the Fitch maps it defines on T_1' and T_2'. Note that the choice of f' ensures that p_2 and p_{k-1} are intersection vertices in both T_1' and T_2' and that $F_1'(p_2) = F_2'(p_2) = \{a\}$ and $F_1'(p_{k-1}) = F_2'(p_{k-1}) = \{b\}$. In turn, the latter implies that r is a union vertex in T_1' if and only if it is a union vertex in T_2'. Thus,

$$l_{f'}(T_2') - f_{f'}(T_1') = l_{f_C}(T_C) + l_{f_D}(T_D) - l_{f_A}(T_A) - l_{f_B}(T_B) + \chi_1 + \chi_k,$$

where

$$\chi_i = \begin{cases} -1 & \text{if } p_i \text{ is a union vertex in } T_1' \text{ but not in } T_2' \\ 1 & \text{if } p_i \text{ is a union vertex in } T_2' \text{ but not in } T_1' \\ 0 & \text{otherwise,} \end{cases}$$

for $i \in \{1, k\}$.

By (1) and Lemma 11, we have that

$$l_f(T_2) - l_f(T_1) = d^t_{\mathrm{MP}}(T_1, T_2) \le l_{f_C}(T_C) + l_{f_D}(T_D) - l_{f_A}(T_A) - l_{f_B}(T_B) + \delta_{AC} + \delta_{BD}.$$

Thus, to prove that $l_{f'}(T_2') - l_{f'}(T_1') \ge l_f(T_2) - l_f(T_1)$, it suffices to prove that we can choose f' so that $\chi_1 \ge \delta_{AC}$ and $\chi_k \ge \delta_{BD}$.

We prove that we can choose a so that $\chi_1 \ge \delta_{AC}$. An analogous argument shows that we can choose b so that $\chi_k \ge \delta_{BD}$.

We choose $a \in S_A$. If $T_A \ne \emptyset$, this ensures that p_1 is an intersection vertex. If $T_A = \emptyset$, then $p_1 = p_2$ and $f'(x_1) = f'(x_2) = a$, so *any* choice of a ensures that p_1 is an intersection vertex. Thus, $\chi_1 \ge 0$. In particular, $\chi_1 \ge \delta_{AC}$ if $\delta_{AC} = 0$.

If $\delta_{AC} = 1$, then $S_A \nsubseteq S_C$. Thus, we can choose $a \in S_A \setminus S_C$. This ensures not only that p_1 is an intersection vertex in T_1' but also that it is a union vertex in T_2'. (In particular, $S_A \nsubseteq S_C$ implies that $S_C \ne S$, so $T_C \ne \emptyset$.) Thus, $\chi_1 = 1 = \delta_{AC}$ in this case. □

We note that this reduction rule is the best possible, in the sense that reducing chains to length 3 instead of length 4 does not preserve $d^t_{\mathrm{MP}}(T_1, T_2)$, for all finite t. Indeed, Kelk et al. give an example of two trees T_1, T_2 where $d^\infty_{\mathrm{MP}}(T_1, T_2) = 2$, but removing one vertex from a common chain of length 4 produces trees T_1', T_2' such that $d^\infty_{\mathrm{MP}}(T_1, T_2) = 1$ [10, Fig. 5]. For the same example, $d^2_{\mathrm{MP}}(T_1, T_2) = 2$ and $d^2_{\mathrm{MP}}(T_1, T_2) = 1$. Thus reducing chains to length 3 does not preserve $d^2_{\mathrm{MP}}(T_1, T_2)$.

4. A lower bound on d^t_{MP}

By Theorem 7, parsimony distance has a kernel of size linear in the TBR distance between the two trees. In this section, we bound the size of this kernel as a function of the parsimony distance itself. Specifically, we prove the following result:

Theorem 13. *Any two trees T_1 and T_2 on X satisfy $d_{TBR}(T_1, T_2) \le 54k(\lg|X| + 1)$, where $k = d^t_{MP}(T_1, T_2)$, for any $t \in \mathbb{N}^\infty_{\ge 2}$.*

Together with Theorem 7, this implies the following corollary:

Corollary 14. *There exists a set of reduction rules for d^t_{MP} such that a fully reduced yes-instance (T_1, T_2, t, k) with $t \in \mathbb{N}^\infty_{\ge 2}$ and $k \ge 1$ consists of a pair of trees on T_1 and T_2 on X with $|X| \le 1{,}484k(\lg k + 11) \in O(k \lg k)$.*

Proof. Let (T_1, T_2, t, k) be a fully reduced yes-instance, let $n = |X|$, and let $k' = d_{\mathrm{TBR}}(T_1, T_2)$. By Theorem 7, we have

$$n \le 20k'$$

and, by Theorem 13,

$$k' \le 54k(\lg n + 1).$$

This gives

$$n \le 1{,}080k(\lg n + 1). \tag{2}$$

For notational convenience, let $c = 11 \cdot 1{,}484 = 16{,}324$. If $n \le c$, then the bound on n claimed in the corollary holds, as $11 \cdot 1{,}484 \le 1{,}484k(\lg k + 11)$. So assume that $n > c$. Then $\lg n > \lg c$, so $\lg n + 1 < \left(1 + \frac{1}{\lg c}\right) \lg n$ and

$$n < 1{,}080 \left(1 + \frac{1}{\lg c}\right) k \lg n \le 1{,}158k \lg n.$$

Since $n > c \ge 8$, we also have $\lg n \le n^{\frac{\lg \lg c}{\lg c}}$ (indeed, for $n \ge 8$, we have $\lg n = n^{\frac{\lg \lg n}{\lg n}}$, and $\frac{\lg \lg n}{\lg n}$ is a decreasing function).[4] Therefore,

$$\begin{aligned} n &\le 1{,}158kn^{\frac{\lg \lg c}{\lg c}}, \\ n^{\frac{\lg c - \lg \lg c}{\lg c}} &\le 1{,}158k, \\ n &\le (1{,}158k)^{\frac{\lg c}{\lg c - \lg \lg c}}. \end{aligned}$$

This implies that

$$\lg n \le \frac{\lg c}{\lg c - \lg \lg c} \cdot (\lg k + \lg 1{,}158),$$

so by (2),

$$n \le 1{,}080k \left(\frac{\lg c}{\lg c - \lg \lg c} \cdot (\lg k + \lg 1{,}158) + 1 \right) \le 1{,}484k(\lg k + 11). \quad \square$$

It remains to prove Theorem 13. As stated in the introduction, the key is to show that $d_{\mathrm{MP}}^t(T_1, T_2)$ is large if T_1 and T_2 have a large number of "leg-disjoint" incompatible quartets. This is similar to the lower bound on $d_{\mathrm{MP}}^\infty(T_1, T_2)$ by Jones, Kelk, and Stougie [8], where it was shown that $d_{\mathrm{MP}}^\infty(T_1, T_2)$ is large if T_1 and T_2 have a large number of *disjoint* incompatible quartets. Leg-disjointness is a much weaker condition. We define the concept of leg-disjoint incompatible quartets in Section 4.1, and show that they provide a lower bound on $d_{\mathrm{MP}}^t(T_1, T_2)$, for any $t \in \mathbb{N}_{\ge 2}^\infty$. To prove Theorem 13, we then show, in Section 4.2, how to find a set of at least $\frac{d_{\mathrm{TBR}}(T_1, T_2)}{2(\lg |X| + 1)}$ leg-disjoint incompatible quartets.

4.1. Leg-disjoint quartets

Given a tree T on X, we call two quartets $q_1, q_2 \subseteq X$ *fully T-disjoint* if $T(q_1)$ and $T(q_2)$ are disjoint. Given a quartet $q = \{a, b, c, d\}$ such that $T|_q = ab|cd$, we call the paths from a to b and from c to d in T the *legs* of q. The path composed of all edges in $T(q)$ not included in the legs of q is the *backbone* of q. The endpoints of the backbone are the *joints* of q. We call two quartets *T-leg-disjoint* if their legs in T are disjoint. Note that this implies that the quartets are themselves disjoint subsets of X.

The main result in this section proves that $d_{\mathrm{MP}}^t(T_1, T_2)$ is large if there exists a large set Q of pairwise T_1-leg-disjoint incompatible quartets of T_1 and T_2. In the remainder of this section, we refer to the quartets in Q simply as leg-disjoint, omitting the explicit reference to the tree T_1 in which their legs are disjoint.

Proposition 15. *Let Q be a set of pairwise leg-disjoint incompatible quartets of two trees T_1 and T_2 on X. Then $d_{MP}^t(T_1, T_2) \ge \frac{|Q|}{27}$, for all $t \in \mathbb{N}_{\ge 2}^\infty$.*

Note that Proposition 15 does not impose *any* constraints on the manner in which the quartets in Q interact in T_2, nor does it require their backbones in T_1 to be disjoint from each other or from the legs of other quartets in Q. Contrast this with the definition of disjoint quartets used by Jones, Kelk, and Stougie [8], which considers two quartets q_1 and q_2 to be disjoint if they are both fully T_1-disjoint and fully T_2-disjoint.

Proof. To simplify the proof, we may assume w.l.o.g. that every leaf in X is part of a quartet in Q. Indeed, if this is not the case, then let $Y \subset X$ be the set of leaves that belong to quartets in Q. By Lemma 6, we have $d_{\mathrm{MP}}^t(T_1, T_2) \ge d_{\mathrm{MP}}^t(T_1|_Y, T_2|_Y)$, and Q is also a set of pairwise leg-disjoint incompatible quartets of $T_1|_Y$ and $T_2|_Y$. Therefore, we may replace T_1 and T_2 with $T_1|_Y$ and $T_2|_Y$ in what follows.

To prove the proposition, we construct a subset $Q' \subseteq Q$ such that $|Q'| \ge \frac{|Q|}{9}$ and $d_{\mathrm{MP}}^t(T_1, T_2) \ge \frac{|Q'|}{3}$. Thus, $d_{\mathrm{MP}}^t(T_1, T_2) \ge \frac{|Q|}{27}$, as claimed.

[4] Note that as $\lg n = \log_2 n$ and $\lg \lg n = \log_2 \log_2 n$ for $n \ge 8$, the identity $\lg n = n^{\frac{\lg \lg n}{\lg n}}$ follows from the fact that this identity also holds for $\log_2$.

To describe the construction of this subset $Q' \subseteq Q$, we need some notation. We use X' to refer to the set of leaves of the quartets in Q': $X' = \bigcup_{q \in Q'} q$. Let $q = \{a, b, c, d\}$ be an incompatible quartet and assume that $T_1|_q = ab|cd$. For a labelling $\bar{f} : V(T_2(X'')) \to S$ of some subtree $T_2(X'')$ of T_2 with $q \subseteq X''$, we define

$$\beta_{\bar{f}}(q) = |\{(x, y) \in \{(a, b), (c, d)\} \mid \bar{f}(x) \neq \bar{f}(y)\}|.$$

In words, $\beta_{\bar{f}}(q)$ is the number of legs of q in T_1 whose endpoints are assigned different states by $\bar{f}$. Furthermore, for any subset $Q'' \subseteq Q$ such that every quartet $q \in Q''$ satisfies $q \subseteq X'$, we define

$$\beta_{\bar{f}}(Q'') = \sum_{q \in Q''} \beta_{\bar{f}}(q).$$

Since the quartets in $Q' \subseteq Q$ are pairwise leg-disjoint in T_1, any character f on X' satisfies $l_f(T_1(X')) \geq \beta_{\bar{f}}(Q')$, where $\bar{f}$ is an arbitrary extension of f to $T_2(X')$. Indeed, every leg of a quartet $q \in Q'$ that contributes to $\beta_{\bar{f}}(q)$ must include a mutation edge and thus increases $l_f(T_1(X'))$ by 1 because the quartets in Q' are pairwise leg-disjoint. We also have $\Delta_{\bar{f}}(T_2(X')) \geq l_f(T_2(X'))$. Thus, it suffices to construct a subset $Q' \subseteq Q$ and an extension $\bar{f}$ of a 2-state character $f : X' \to S$ to the vertices of $T_2(X')$ such that $|Q'| \geq \frac{|Q|}{9}$ and $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X')) \geq \frac{|Q'|}{3}$. Indeed, this implies that $d^2_{\text{MP}}(T_1(X'), T_2(X')) \geq l_f(T_1(X')) - l_f(T_2(X')) \geq \beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X')) \geq \frac{|Q'|}{3} \geq \frac{|Q|}{27}$. By Lemma 6, we have $d^t_{\text{MP}}(T_1, T_2) \geq d^2_{\text{MP}}(T_1, T_2) \geq d^2_{\text{MP}}(T_1(X'), T_2(X'))$, so $d^t_{\text{MP}}(T_1, T_2) \geq \frac{|Q|}{27}$.

We assume that $S = \{\text{red}, \text{blue}\}$ from here on. Accordingly, we call the states in S *colours* and refer to $\bar{f} : V(T_2(X')) \to S$ as a *colouring* of $T_2(X')$.

We construct the desired subset $Q' \subseteq Q$ and colouring $\bar{f}$ of $T_2(X')$ in two phases, maintaining the invariant that $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X')) \geq \frac{|Q'|}{3}$. We prove that once we are unable to find more quartets to add to Q' in the second phase, we have $|Q'| \geq \frac{|Q|}{9}$. Thus, the set Q' and the colouring $\bar{f}$ obtained after the second phase have the desired properties.

Phase 1: Select a maximal subset of pairwise fully T_2-disjoint quartets. We select a maximal subset $Q' \subseteq Q$ of quartets that are pairwise fully T_2-disjoint. The vertices of $T_2(X')$ can easily be coloured so that $\beta_{\bar{f}}(Q') = 2|Q'|$ and $\Delta_{\bar{f}}(T_2(X')) = |Q'|$:

Consider the forest F obtained from $T_2(X')$ by deleting one edge e_q from the backbone in T_2 of each quartet $q \in Q'$. Let T' be the tree obtained from $T_2(X')$ by contracting every connected component of F into a single vertex. Since T' is a tree, it is bipartite and thus can be 2-coloured. Let $f' : V(T') \to \{\text{red}, \text{blue}\}$ be such a 2-colouring of T'. Then we choose $\bar{f}$ so that it colours every vertex in the connected component of F represented by v with the colour $f'(v)$, for every vertex $v \in V(T')$.

Since F has $|Q'| + 1$ connected components, we have $\Delta_{\bar{f}}(T_2(X')) = |Q'|$.

Next consider any quartet $q \in Q'$ and assume w.l.o.g. that $T_1|_q = ab|cd$ and $T_2|_q = ac|bd$. Then a and c belong to the same connected component C_1 of F, b and d belong to the same connected component C_2 of F, $C_1 \neq C_2$, and the two vertices v_1 and v_2 in T' representing C_1 and C_2 are adjacent. Indeed, if a and c belonged to different connected components of F, b and d belonged to different connected components of F or v_1 and v_2 were not adjacent in T', then $T_2(q)$ would contain an edge $e_{q'}$ in the backbone of another quartet $q' \in Q'$, a contradiction because the quartets in Q' are fully T_2-disjoint. The fact that C_1 and C_2 are different connected components follows because deleting the edge e_q separates a and c from b and d in T_2.

Since $a, c \in C_1$, $b, d \in C_2$, and v_1 and v_2 are adjacent, we have $\bar{f}(a) \neq \bar{f}(b)$ and $\bar{f}(c) \neq \bar{f}(d)$. Thus, $\beta_{\bar{f}}(q) = 2$. Since this is true for every quartet $q \in Q'$, we have $\beta_{\bar{f}}(Q') = 2|Q'|$. This shows that $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X')) = |Q'| \geq \frac{|Q'|}{3}$.

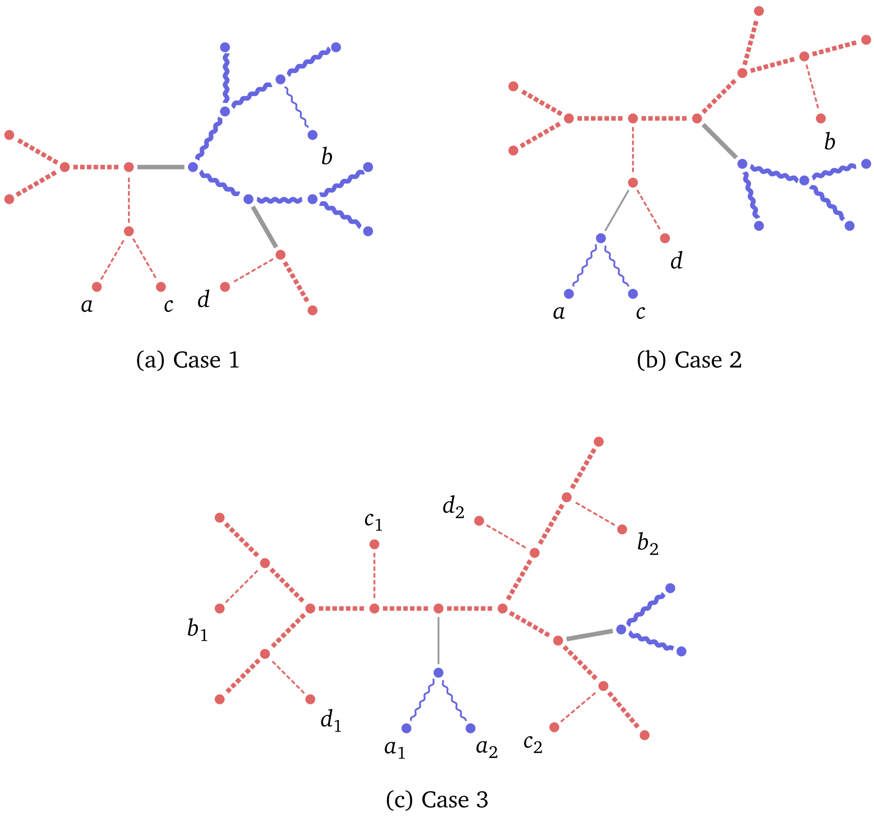
Phase 2: Greedily add quartets to Q'. Let $U = Q \setminus Q'$ be the set of uncoloured quartets. We add quartets from U to Q' one, two or three quartets at a time. For each added group of quartets, we extend the colouring $\bar{f}$ to $T_2(X')$ and modify it in a manner that ensures that $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X'))$ increases by at least 1. Thus, the inequality $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X')) \geq \frac{|Q'|}{3}$ is maintained by each addition.

To move quartets from U to Q', we consider several cases, choosing the first case that applies:

Case 1: Quartets with a "good" parsimonious extension. Recall that for any Y with $X' \subseteq Y \subseteq X$, the parsimonious extension of $\bar{f}$ to $T_2(Y)$ is the unique labelling $\tilde{f}$ of the vertices in $T_2(Y)$ such that $\tilde{f}(v) = \bar{f}(v)$ for all $v \in T_2(X')$ and $\tilde{f}(v) = \bar{f}(w_v)$ for all $v \notin T_2(X')$, where w_v is the vertex in $T_2(X')$ closest to v.

If there exists a quartet $q \in U$ such that the parsimonious extension $\bar{f}'$ of $\bar{f}$ to $T_2(X' \cup q)$ satisfies $\beta_{\bar{f}'}(q) > 0$, then we add q to Q' and set $\bar{f} = \bar{f}'$. See Fig. 5a. This increases $\beta_{\bar{f}}(Q')$ by at least 1 and leaves $\Delta_{\bar{f}}(T_2(X'))$ unchanged (note that since we add q to Q', $T_2(X')$ now includes the leaves of q). Thus, $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X'))$ increases by at least 1.

The remaining cases assume that Case 1 is not applicable. Thus, the parsimonious extension $\bar{f}'$ of $\bar{f}$ to $T_2(X' \cup q)$ satisfies $\beta_{\bar{f}'}(q) = 0$ for every quartet $q \in U$. Consider the pendant subtrees of $T_2(X')$ in T_2. Since we initialized Q' to be a maximal subset of quartets that are pairwise fully T_2-disjoint, there is no quartet in U that has all its leaves in one of these pendant subtrees.



(a) Case 1

(b) Case 2

(c) Case 3

Fig. 5. The updated colouring in Cases 1–3 of the proof of Proposition 15. Only $T_2(X')$ and the leaves of q are shown. Bold edges are in $T_2(X')$. Thin edges are the new edges added to $T_2(X' \cup q)$ or, in Case 3, to $T_2(X' \cup q_1 \cup q_2)$. The edges between red vertices are shown in red/dotted. The edges between blue vertices are shown in blue/wavy. Straight solid grey edges are ones whose endpoints have different colours. (See online version for colour figures.)

Case 2: Quartets with at least two leaves in the same pendant subtree. Suppose that there exists a quartet $q \in U$ that has at least two of its leaves in the same pendant subtree T' of $T_2(X')$. The parsimonious extension $\bar{f}'$ of $\bar{f}$ to $T_2(X' \cup q)$ colours all leaves of q in T' the same colour. Assume that $T_1|_q = ab|cd$ and $T_2|_q = ac|bd$. Since T' contains at least two leaves of q, we can assume w.l.o.g. that $a, c \in T'$, and that $\bar{f}'$ colours a and c red. Since $\beta_{\bar{f}'}(q) = 0$, this implies that $\bar{f}'$ also colours b and d red. We add q to Q', set $\bar{f} = \bar{f}'$, and then change the colour of a and c to blue and colour all vertices on the path from a to c in T_2 blue. See Fig. 5b. This ensures that $\beta_{\bar{f}}(q) = 2$, so $\beta_{\bar{f}}(Q')$ increases by 2. $\Delta_{\bar{f}}(T_2(X'))$ is easily verified to increase by 1 (since (a, c) must be a cherry in $T_2|_{X'}$). Thus, $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X'))$ increases by 1.

If neither Case 1 nor Case 2 applies, then every pendant subtree of $T_2(X')$ contains at most one leaf from each quartet in U.

Case 3: A pendant subtree with leaves from more than one quartet. If there exists a pendant subtree T' of $T_2(X')$ that contains leaves from at least two quartets in U, then pick two such quartets q_1 and q_2 and let $\bar{f}'$ be the parsimonious extension of $\bar{f}$ to $T_2(X' \cup q_1 \cup q_2)$. Assume that $\bar{f}'$ colours the leaves of $q_1 \cup q_2$ in T' red. We add q_1 and q_2 to Q' and set $\bar{f} = \bar{f}'$. Then we change the colour of every vertex in $T_2(X')$ that belongs to T' to blue. (Since we added q_1 and q_2 to Q', $T_2(X')$ now includes vertices in T'.) See Fig. 5c. Since both q_1 and q_2 have a single leaf in T', this ensures that $\beta_{\bar{f}}(Q')$ increases by 2, whereas $\Delta_{\bar{f}}(T_2(X'))$ increases by 1. Thus, $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X'))$ increases by 1.

Case 4: All pendant subtrees are singletons. If we reach this case, then Cases 1–3 do not apply. Thus, every pendant subtree of $T_2(X')$ contains at most one leaf that belongs to a quartet in U. Since we assumed that every leaf in X belongs to some quartet in Q, this implies that every pendant subtree of $T_2(X')$ consists of a single leaf (and this leaf belongs to some quartet in U).

Let a *side* of $T_2(X')$ be a maximal path in $T_2(X')$ whose internal vertices have degree 2 in $T_2(X')$. Then every leaf l of a quartet in U is adjacent to some side of $T_2(X')$ (that is, l is adjacent to an internal vertex of that side). Moreover, for any quartet $q \in U$ with $T_1|_q = ab|cd$, either all leaves of q are adjacent to the same side, or at least one of the pairs $\{a, b\}, \{c, d\}$ has its elements adjacent to two different sides. If this were not the case, then a, b would be adjacent to one side and c, d would be adjacent to another. This would imply that $T_2|q = ab|cd$, contradicting that q is an incompatible quartet. See Fig. 6. We can assume from here on that all internal vertices of a side of $T_2(X')$ have the same colour. If not, we can change $\bar{f}$ so that this is true, without increasing $\Delta_{\bar{f}}(T_2(X'))$.

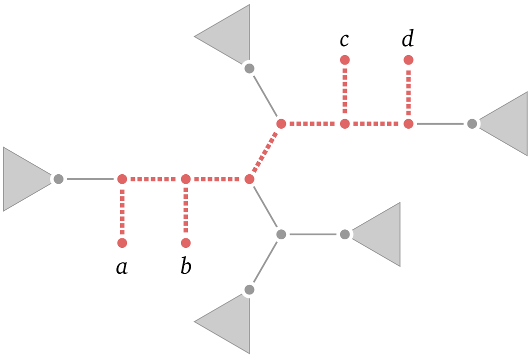


Fig. 6. A quartet q with $T_1(q) = ab|cd$ and with a and b adjacent to one side of T_2 and c and d adjacent to another side of T_2 cannot be incompatible.

Case 4.1: A side with adjacent leaves from at least three quartets. Suppose that there exists a side P that has adjacent leaves[5] from at least three quartets $q_1, q_2, q_3 \in U$ and for each $i \in \{1, 2, 3\}$, P has a_i but not b_i as an adjacent vertex, where $T_1|q_i = a_i b_i|c_i d_i$. Let $\bar{f}'$ be the parsimonious extension of $\bar{f}$ to $T_2(X' \cup q_1 \cup q_2 \cup q_3)$. We add q_1, q_2, and q_3 to Q' and set $\bar{f} = \bar{f}'$. Assume that the colour of all internal vertices of P is red. Then a_1, a_2, a_3 are also red. This in turn implies that b_1, b_2, b_3 are also red, as otherwise $\beta_{\bar{f}'}(q_i) > 0$ for some i, and Case 1 would apply. We change the colour of the internal vertices of P to blue and change the colour of all adjacent leaves of P that are leaves of q_1, q_2 or q_3 to blue. See Fig. 7a. Since a_i is now coloured blue, and b_i red, for each $i \in \{1, 2, 3\}$, this increases $\beta_{\bar{f}}(Q')$ by at least 3. The only mutation edges introduced into $T_2(X')$ are the first and last edge of P. Thus, $\Delta_{\bar{f}}(T_2(X'))$ increases by at most 2. Overall, $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X'))$ increases by at least 1.

Case 4.2: A side with an adjacent quartet and an adjacent leaf. Next suppose that there exists a side P of $T_2(X')$ and two quartets $q_1, q_2 \in U$ with $T_1|_{q_1} = a_1 b_1|c_1 d_1$, $T_1|_{q_2} = a_2 b_2|c_2 d_2$, and such that all leaves of q_1 are adjacent to P and a_2 but not b_2 is adjacent to P. Assume w.l.o.g. that $T_2|_{q_1} = a_1 c_1|b_1 d_1$, and suppose we walk along P such that a_1 and c_1 appear before b_1 and d_1. Assume further that a_2 occurs after a_1 and c_1 along P. (The other case is symmetric using b_1 and d_1 in place of a_1 and c_1.) Then let $\bar{f}'$ be the parsimonious extension of $\bar{f}$ to $T_2(X' \cup q_1 \cup q_2)$. We add q_1 and q_2 to Q' and set $\bar{f} = \bar{f}'$. Assume that the colour of all internal vertices of P is red. Then we change the colour of all vertices of q_1 and q_2 that occur after a_1 and c_1 along P to blue, and we colour all internal vertices of P that belong to paths between these leaves blue. See Fig. 7b. This increases $\Delta_{\bar{f}}(T_2(X'))$ by at most 2. At the same time, we obtain $\beta_{\bar{f}}(q_1) = 2$ and $\beta_{\bar{f}}(q_2) \geq 1$ (as a_2 changes colour but b_2 does not). Thus, $\beta_{\bar{f}}(Q')$ increases by at least 3. Overall, $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X'))$ increases by at least 1.

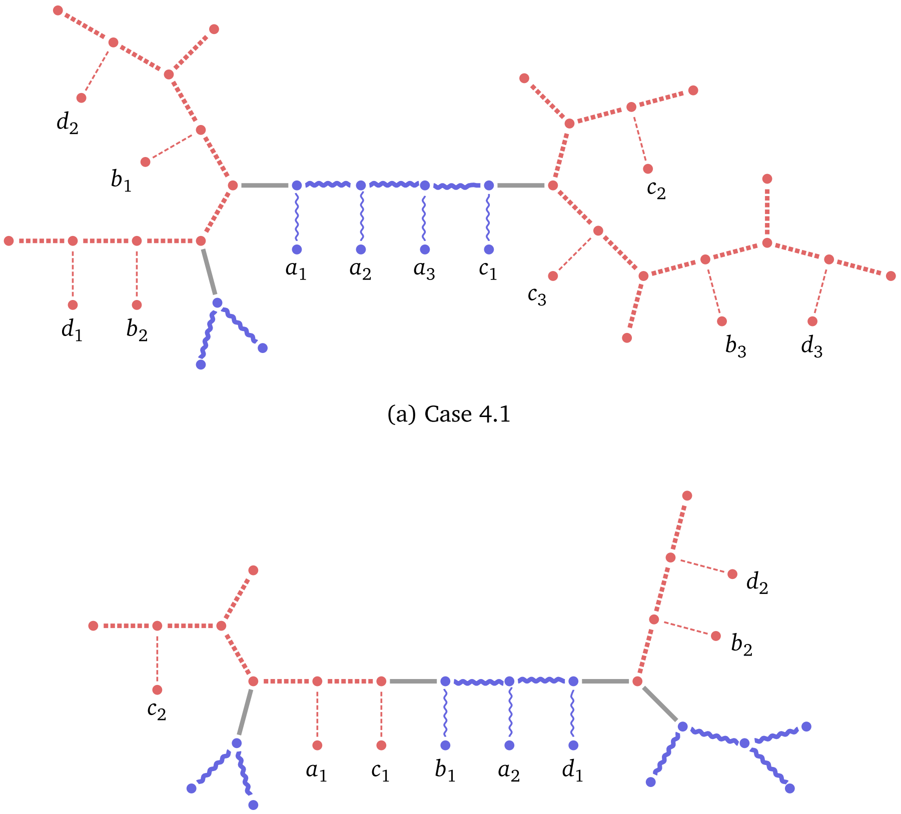
Case 4.3: A side with two adjacent quartets. The final case we consider is when there are two quartets $q_1, q_2 \in U$ such that all leaves of q_1 and q_2 are adjacent to the same side P of $T_2(X')$. Assume w.l.o.g., that $T_1|_{q_1} = a_1 b_1|c_1 d_1$, $T_1|_{q_2} = a_2 b_2|c_2 d_2$, $T_2|_{q_1} = a_1 c_1|b_1 d_1$, and $T_2|_{q_2} = a_2 c_2|b_2 d_2$. Assume further that the leaves of q_1 occur in the order a_1, c_1, b_1, d_1 along P and, following P in the same direction, the leaves of q_2 occur in the order a_2, c_2, b_2, d_2 along P. (If c_1 occurs before a_1, then we may swap the roles of a_1 and c_1 in the argument that follows; similarly for the pairs (b_1, d_1), (a_2, c_2), (b_2, d_2).)

If both c_1 and c_2 occur before both b_1 and b_2 along P, then let $\bar{f}'$ be the parsimonious extension of $\bar{f}$ to $T_2(X' \cup q_1 \cup q_2)$. We add q_1 and q_2 to Q' and set $\bar{f} = \bar{f}'$. Assume that the colour of all internal vertices of P is red. We change the colour of a_1, c_1, a_2, c_2 to blue and also change the colour of all vertices on the paths between these four leaves to blue. See Fig. 8a. This increases $\Delta_{\bar{f}}(T_2(X'))$ by at most 2 and ensures that $\beta_{\bar{f}}(q_1) = \beta_{\bar{f}}(q_2) = 2$. Thus, $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X'))$ increases by at least 2.

If c_1 and c_2 do not both occur before b_1 and b_2, then the four leaves b_1, c_1, b_2, c_2 must occur in the order c_1, b_1, c_2, b_2 or c_2, b_2, c_1, b_1 along P. Assume that the order is c_1, b_1, c_2, b_2 (the other case is symmetric). Then observe that a_1 occurs before c_1 and d_2 occurs after b_2. Thus, these six leaves occur in the order $a_1, c_1, b_1, c_2, b_2, d_2$. We distinguish the possible positions of the two leaves a_2 and d_1 and in each case update $\bar{f}$ so that $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X'))$ increases by at least 1. In each case, the starting point is the parsimonious extension $\bar{f}'$ of $\bar{f}$ to $T_2(X' \cup q_1 \cup q_2)$. We assume that $\bar{f}'$ colours all vertices on P and all leaves of q_1 and q_2 red.

If a_2 occurs before b_1, then we change the colours of b_1, d_1, c_2, b_2, and d_2 and the colours of all vertices on the paths between them in T_2 to blue. See Fig. 8b. This ensures that $\beta_{\bar{f}}(q_1) = 2$ and $\beta_{\bar{f}}(q_2) = 1$. Thus, $\beta_{\bar{f}}(Q')$ increases by 3. At the same time, we introduce at most two mutation edges into P, so $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X'))$ increases by at least 1.

[5] We note that in this context by “adjacent leaves” we mean leaves that are adjacent to the side in question, not to each other (and similarly for “adjacent quartets”).



(a) Case 4.1

(b) Case 4.2

Fig. 7. The updated colouring in Cases 4.1 and 4.2 of the proof of Proposition 15. Only $T_2(X')$ and the leaves of q are shown. Bold edges are in $T_2(X')$. Thin edges are the new edges added to $T_2(X' \cup q_1 \cup q_2 \cup q_3)$ (in Case 4.1) or to $T_2(X' \cup q_1 \cup q_2)$ (in Case 4.2). The edges between red vertices are shown in red/dotted. The edges between blue vertices are shown in blue/wavy. Straight solid grey edges are ones whose endpoints have different colours.

The case when d_1 occurs after c_2 is analogous to the case when a_2 occurs before b_1. We colour a_1, c_1, b_1, a_2, c_2, and all vertices on the paths between them blue. This ensures that $\beta_{\bar{f}}(q_1) = 1$ and $\beta_{\bar{f}}(q_2) = 2$ and introduces at most two mutation edges into P. Thus, $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X'))$ increases by at least 1.

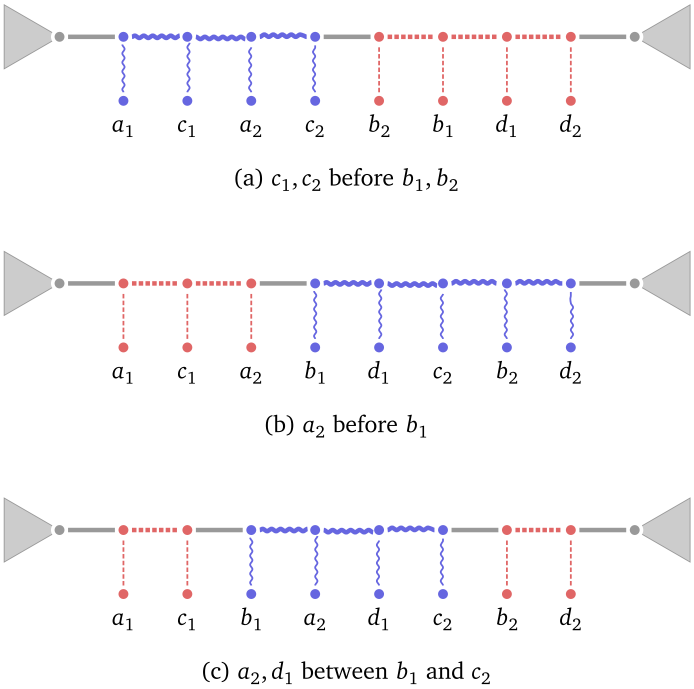
This leaves the case when both a_2 and d_1 occur between b_1 and c_2. In this case, we change the colour of b_1, d_1, a_2, c_2, and of all vertices on the paths between them in T_2 to blue. See Fig. 8c. This ensures that $\beta_{\bar{f}}(q_1) = \beta_{\bar{f}}(q_2) = 2$ and $\Delta_{\bar{f}}(T_2(X'))$ increases by 2. Thus, $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X'))$ increases by 2.

Once none of these cases is applicable, we obtain a subset $Q' \subseteq Q$ and a colouring $\bar{f}$ of $T_2(X')$ such that $\beta_{\bar{f}}(Q') - \Delta_{\bar{f}}(T_2(X')) \geq \frac{|Q'|}{3}$. It remains to prove that $|Q'| \geq \frac{|Q|}{9}$. Since $Q = Q' \cup U$, this follows if we can prove that $|U| \leq 8|Q'|$.

Consider any quartet $q \in U$ with $T_1|_q = ab|cd$. As argued before, once none of Cases 1–3 applies, either all leaves of q are adjacent to one side of $T_2(X')$, or w.l.o.g., a and b are adjacent to different sides of $T_2(X')$, because q is incompatible. We partition U into two subsets U_1 and U_2, containing the quartets in U whose leaves are all adjacent to the same side and those whose leaves are adjacent to at least two sides, respectively.

Now we charge the quartets in U to the sides of $T_2(X')$. We charge each quartet $q \in U_1$ to the side of $T_2(X')$ to which the leaves of q are adjacent. We charge each quartet $q \in U_2$ to the *two* sides of $T_2(X')$ to which a and b are adjacent.

Since Case 4.3 does not apply to the quartets in U, there is no side that is charged for more than one quartet in U_1. Since Case 4.1 does not apply, there is no side that is charged for more than two quartets in U_2. Since Case 4.2 does not apply, there is no side that is charged for a quartet in U_1 and for at least one quartet in U_2. Thus, every side of $T_2(X')$ that is charged for any quartet is charged for one quartet in U_1 or for at most two quartets in U_2. Since every quartet in U_1 is charged to one side of $T_2(X')$, and every quartet in U_2 is charged to two sides of $T_2(X')$, the number of sides of $T_2(X')$ is thus at least $|U_1| + |U_2| = |U|$. On the other hand, since $T_2(X')$ has $|X'| = 4|Q'|$ leaves, it has at most $2|X'| = 8|Q'|$ sides. Thus, $|U| \leq 8|Q'|$. This finishes the proof that $|Q'| \geq \frac{|Q|}{9}$ and thus the proof of the proposition. $\square$


a_1 c_1 a_2 c_2 b_2 b_1 d_1 d_2
a_1 c_1 a_2 b_1 d_1 c_2 b_2 d_2
a_1 c_1 b_1 a_2 d_1 c_2 b_2 d_2

(a) c_1, c_2 before b_1, b_2

(b) a_2 before b_1

(c) a_2, d_1 between b_1 and c_2

Fig. 8. The updated colouring in Case 4.3 of the proof of Proposition 15. Only $T_2(X')$ and the leaves of q are shown. Bold edges are in $T_2(X')$. Thin edges are the new edges added to $T_2(X' \cup q_1 \cup q_2)$. The edges between red vertices are shown in red/dotted. The edges between blue vertices are shown in blue/wavy. Straight solid grey edges are ones whose endpoints have different colours.

4.2. Finding leg-disjoint incompatible quartets

It remains to find a set of leg-disjoint incompatible quartets of size at least $\frac{d_{\mathrm{TBR}}(T_1,T_2)}{2(\lg n+1)}$, where $n = |X|$. In combination with Proposition 15, this implies that $d^t_{\mathrm{MP}}(T_1, T_2) \geq \frac{d_{\mathrm{TBR}}(T_1,T_2)}{54(\lg n+1)}$, as claimed in Theorem 13. To do this, we use an ILP formulation of the unrooted MAF problem by Van Wersch et al. [14]. For a pair of trees (T_1, T_2) on X, let Q be the set of incompatible quartets of T_1 and T_2. For a quartet $q \in Q$, let $\mathcal{L}(q)$ be the set of edges of T_1 that belong to the legs of q. Van Wersch et al. proved that the following ILP expresses the unrooted MAF problem, where E_1 is the set of edges of T_1 and $x_e \in \{0, 1\}$ indicates whether we include e in a set of edges we cut to obtain an AF of (T_1, T_2):

$$\begin{aligned} &\text{Minimize} \sum_{e \in E_1} x_e \\ &\text{s.t.} \sum_{e \in \mathcal{L}(q)} x_e \geq 1 \qquad \forall q \in Q \\ &\qquad x_e \in \{0,1\} \quad \forall e \in E_1. \end{aligned} \tag{3}$$

The constraints express that we obtain an AF of (T_1, T_2) by cutting a subset of edges in T_1 that contains at least one edge in $\mathcal{L}(q)$ for every incompatible quartet $q \in Q$. For the remainder of this section, we say that an edge set E' *hits* a quartet $q \in Q$ if E' contains at least one edge in $\mathcal{L}(q)$. The objective function expresses the goal to cut as few edges as possible, to obtain an MAF. Recall that the number of edges cut to produce a MAF of T_1 and T_2 is exactly $d_{\mathrm{TBR}}(T_1, T_2)$, so the objective function value of any feasible solution of (3) is an upper bound on $d_{\mathrm{TBR}}(T_1, T_2)$.

Interestingly, the integral version of the dual of this LP corresponds to choosing a subset of quartets from Q that are pairwise leg-disjoint:

$$\begin{aligned} &\text{Maximize} \sum_{q \in Q} y_q \\ &\text{s.t.} \sum_{q \in Q : e \in \mathcal{L}(q)} y_q \leq 1 \qquad \forall e \in E_1 \\ &\qquad y_q \in \{0,1\} \quad \forall q \in Q. \end{aligned} \tag{4}$$

Indeed, the variable y_q for each quartet $q \in Q$ indicates whether it is chosen. The constraints in (4) ensure that no edge of T_1 is included in the legs of more than one chosen quartet, that is, all quartets are leg-disjoint.

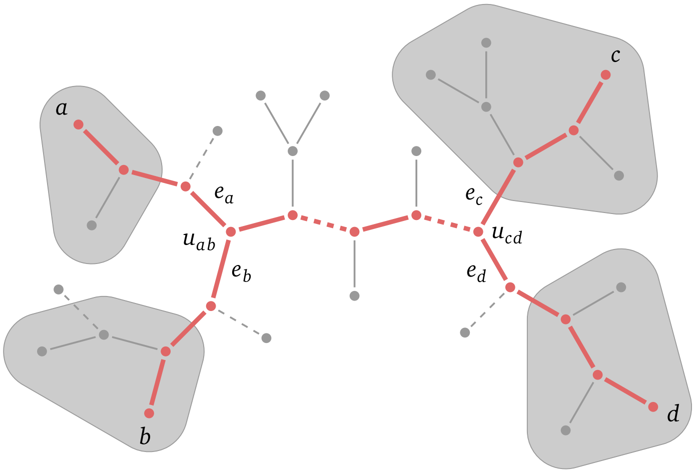


Fig. 9. Example showing the vertices u_{ab}, u_{cd}, edges e_a, e_b, e_c, e_d, and sets X_a, X_b, X_c, X_d, for some quartet $q = ab|cd$, and a subset of edges E' of T_1 that does not hit the legs of q. Dashed edges are edges in E'. The bold red edges are those in $T_1(q)$.

Note that by strong duality, the linear relaxations of these ILPs (where the variables x_e or y_q are not required to be integer) have the same optimum. This makes it reasonable to hope that the integral version also have optimums that are close — and thus, that the maximum number of pairwise leg-disjoint conflicting quartets is close to $d_{\text{TBR}}(T_1, T_2)$. This observation motivates our approach in the proof of the following proposition.

Proposition 16. *For any pair of trees T_1 and T_2 on X, there exists a set Q' of pairwise leg-disjoint incompatible quartets such that $|Q'| \geq \frac{d_{TBR}(T_1,T_2)}{2(\lg n+1)}$, where $n = |X|$.*

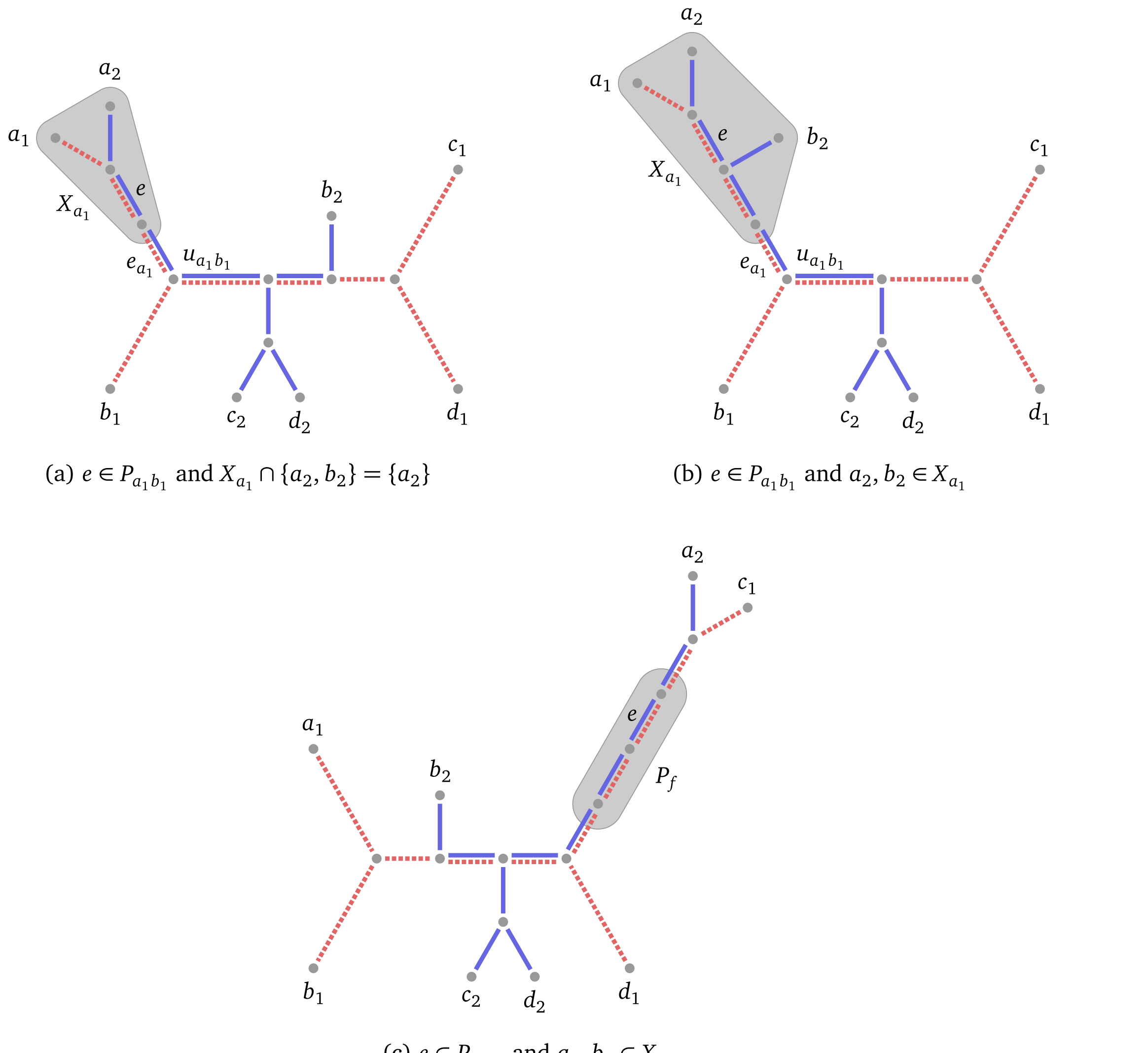
Proof. Our goal is to find feasible solutions $\hat{x}$ and $\hat{y}$ of the ILPs (3) and (4) such that $\sum_{e\in E_1} \hat{x}_e \leq 2(\lg n + 1) \cdot \sum_{q\in Q} \hat{y}_q$. In other words, we want to find a subset $E' = \{e \in E_1 \mid \hat{x}_e = 1\}$ that hits all quartets in Q and a subset $Q' = \{q \in Q \mid \hat{y}_q = 1\}$ of leg-disjoint quartets in Q such that $|E'| \leq |Q'| \cdot 2(\lg n + 1)$. As $|E'|$ gives an upper bound on $d_{\text{TBR}}(T_1, T_2)$, this implies that $|Q'| \geq \frac{d_{\text{TBR}}(T_1,T_2)}{2(\lg n+1)}$, as required.

To describe the choice of quartets in Q' and edges in E', we need a bit of notation. Let E' be some set of edges in T_1 that we have selected at some point in the algorithm, and consider a quartet $q \in Q$ that is not hit by E'. Throughout this section, we refer to such a quartet q with $T_1|_q = ab|cd$ as the quartet $ab|cd$, we use P_{ab} to denote the leg with endpoints a and b, and we use P_{cd} to denote the leg with endpoints c and d. u_{ab} and u_{cd} are the joints of q included in P_{ab} and P_{cd}, respectively. Let X_a be the set of all leaves reachable from a via paths in T_1 that do not include u_{ab} nor any edges in E' (note that X_a always contains a itself). We define sets X_b, X_c, and X_d analogously. Let e_a be the first edge on the path from u_{ab} to a in T_1, and let e_b be the first edge on the path from u_{ab} to b in T_1. These definitions are illustrated in Fig. 9. Note that the sets X_a, X_b, X_c, X_d depend on the choice of E' as well as q; when we need to specify E' or q (for instance, when a leaf is part of two different quartets under consideration or we refer to the states of E' before and after an update), we will denote these sets by $X_a^{E'}, X_b^{E'}, X_c^{E'}, X_d^{E'}$ or $X_a^q, X_b^q, X_c^q, X_d^q$, or $X_a^{E',q}, X_b^{E',q}, X_c^{E',q}, X_d^{E',q}$ when we need to specify both. We use F_1 to denote the forest obtained from T_1 by cutting the edges in E', suppressing degree-2 vertices, and deleting unlabelled vertices of degree less than 2. When it is necessary to specify the set of edges E' cut to obtain F_1, we refer to F_1 as $F_1^{E'}$. Every edge $e \in F_1$ corresponds to a path between its endpoints in T_1. We refer to this path as P_e. For two vertices a and b in the same connected component of F_1, we use $\tilde{P}_{ab}$ to refer to the path from a to b in F_1. Note that this implies that $P_{ab} = \bigcup_{e\in\tilde{P}_{ab}} P_e$.

To construct E' and Q', we use a simple greedy algorithm: We start by setting $E' = \emptyset$ and $Q' = \emptyset$. We maintain the invariant that $|E'| \leq |Q'| \cdot 2(\lg n + 1)$ and that $Q' \subseteq Q$ is a subset of leg-disjoint quartets. Thus, once E' hits all quartets in Q, we obtain the desired sets Q' and E'.

As long as there exists a quartet $q \in Q$ that is not being hit by E' yet, we choose such a quartet q, add a subset of the edges in $\mathcal{L}(q)$ to E', and add q to Q'. We choose the quartet $q = ab|cd$ that minimizes $\left|X_a^{E',q} \cup X_b^{E',q}\right|$, where we assume that $\left|X_a^{E',q} \cup X_b^{E',q}\right| \leq \left|X_c^{E',q} \cup X_d^{E',q}\right|$. Among all such quartets, we prefer one that minimizes $\left|\tilde{P}_{cd}\right|$. If ties remain, we choose an arbitrary quartet from the remaining quartets. When adding q to Q', we also add the edges e_a and e_b to E', and we add an arbitrary edge in P_e to E', for every edge $e \in \tilde{P}_{cd}$. Since this ensures that E' now hits at least one more quartet than before, namely q, E' will eventually hit all quartets and the algorithm terminates. At that point, we obviously have that

Observation 17. *The set of edges E' computed by the algorithm hits all quartets in Q.*



(a) $e \in P_{a_1b_1}$ and $X_{a_1} \cap \{a_2, b_2\} = \{a_2\}$

(b) $e \in P_{a_1b_1}$ and $a_2, b_2 \in X_{a_1}$

(c) $e \in P_{a_1b_1}$ and $a_2, b_2 \in X_{a_1}$

Fig. 10. Illustration of the different cases in the proof of Lemma 18. The subtrees $T_1(q_1)$ and $T_1(q_2)$ are shown in red/dotted and blue/solid, respectively. The subtree with leaf set X_{a_1} is shaded in Figures (a) and (b). In Figure (c), the subpath P_f is shaded.

The next lemma shows the correctness of the construction of Q'.

Lemma 18. *The set of quartets Q' computed by the algorithm is leg-disjoint.*

Proof. Assume that there exist two quartets $q_1 = a_1b_1|c_1d_1$ and $q_2 = a_2b_2|c_2d_2$ in Q' whose legs share an edge e. See Fig. 10. Since we add quartets to Q' one at a time, we can assume that we add q_1 to Q' before we add q_2. Let E_1' be the set of edges in E' at the beginning of the iteration that adds q_1 to Q', and let $F_1 = F_1^{E_1'}$. Let E_2' be the set of edges in E' at the beginning of the iteration that adds q_2 to Q'. Then $E_1' \subseteq E_2'$ and neither E_1' nor E_2' hits q_2. Assume w.l.o.g. that e belongs to the path $P_{a_2b_2}$. (We do not use the fact that $\left|X_{a_2}^{E_1',q_2} \cup X_{b_2}^{E_1',q_2}\right| \le \left|X_{c_2}^{E_1',q_2} \cup X_{d_2}^{E_1',q_2}\right|$, nor will we consider any of the edges that are added to E' as a result of adding q_2 to Q', so the case when $e \in P_{c_2d_2}$ is symmetric.)

First suppose that $e \in P_{a_1b_1}$ (Figs. 10a and 10b). Then a_1, b_1, a_2, b_2 are all in the same connected component of $T_1 - E_1'$, and at least one of a_2, b_2 is in $X_{a_1}^{E_1',q_1}$ or $X_{b_1}^{E_1',q_1}$ (whether a_2, b_2 are in the same set depends on whether $u_{a_1b_1}$ is on the path $P_{a_2b_2}$). Suppose w.l.o.g. that $a_2 \in X_{a_1}^{E_1',q_1}$. If $b_2 \notin X_{a_1}^{E_1',q_1}$ (Fig. 10a), then $P_{a_2b_2}$ includes the edge e_{a_1}, which belongs to E_2', so E_2' hits q_2, a contradiction. If $b_2 \in X_{a_1}^{E_1',q_1}$ (Fig. 10b), then $X_{a_2}^{E_1',q_2} \cup X_{b_2}^{E_1',q_2} \subseteq X_{a_1}^{E_1',q_1} \subset X_{a_1}^{E_1',q_1} \cup X_{b_1}^{E_1',q_1}$ (recall that $X_{b_1}^{E_1',q_1}$

is non-empty and disjoint from $X_{a_1}^{E_1',q_1}$ by definition). Thus, q_1 does not minimize $\left|X_{a_1}^{E_1'} \cup X_{b_1}^{E_1'}\right|$ among the quartets not hit by E_1' whether $\left|X_{a_2}^{E_1',q_2} \cup X_{b_2}^{E_1',q_2}\right| \le \left|X_{c_2}^{E_1',q_2} \cup X_{d_2}^{E_1',q_2}\right|$ or $\left|X_{a_2}^{E_1',q_2} \cup X_{b_2}^{E_1',q_2}\right| > \left|X_{c_2}^{E_1',q_2} \cup X_{d_2}^{E_1',q_2}\right|$. This contradicts the choice of q_1.

Now suppose that $e \in P_{c_1d_1}$ (Fig. 10c). This in turn implies that $e \in P_f$, for some edge $f \in \tilde{P}_{c_1d_1}$. Thus, $P_{a_2b_2}$ and P_f overlap in e. Let x and y be the endpoints of P_f. Then the path from any internal vertex of P_f to any leaf of T_1 must include x, y or an edge in E_1' because otherwise, P_f would not correspond to a single edge f in F_1. Since $P_{a_2b_2}$ is not hit by E_1' but does contain e, this implies that in fact all edges between x and y are in $P_{a_2b_2}$, that is, P_f is a subpath of $P_{a_2b_2}$. Since E_2' includes an edge in P_f, it therefore hits $P_{a_2b_2}$, and thus q_2, again a contradiction. □

Since each iteration that adds a quartet $q = ab|cd$ to Q' adds $\left|\tilde{P}_{cd}\right| + 2$ edges to E', it suffices to prove the following lemma to prove the invariant that $|E'| \le |Q'| \cdot 2(\lg n + 1)$:

Lemma 19. *The quartet $q = ab|cd$ chosen in each iteration satisfies $\left|\tilde{P}_{cd}\right| \le 2\lg n$.*

Proof. Recall that $q = ab|cd$ is chosen from among the quartets in Q not hit by E' such that $\left|X_a^q \cup X_d^q\right|$ is minimized, and among these, such that $\left|\tilde{P}_{cd}\right|$ is minimized. Thus, it suffices to show that there exist c', d' such that $q' = ab|c'd' \in Q$, q' is not hit by E', and $\left|\tilde{P}_{c'd'}\right| \le 2\lg n$. We will do this by choosing an appropriate $q'' \in Q$ not hit by E', and 'moving' the leaves c' and d' as close together as possible. This gives us the desired bound on $\left|\tilde{P}_{c'd'}\right|$, and by choice of q'' we will see that the resulting quartet is still a conflicting quartet not hit by E'.

So now choose a quartet $q'' = ab|c''d'' \in Q$ not hit by E' and such that $\left|X_{c''}^{q''} \cup X_{d''}^{q''}\right|$ is minimized. Such a quartet exists because $q \in Q$ is not hit by E'. Since $q'' \in Q$, we can assume that $T_2|_{q''} = ac''|bd''$ (the case when $T_2|_{q''} = ad''|bc''$ is symmetric). This is shown in Fig. 11. Let $w_{c''}$ be the vertex in F_1 closest to c'' that separates $X_{c''}^{q''}$ from $X_{d''}^{q''}$, and let $w_{d''}$ be the vertex in F_1 closest to d'' that separates $X_{c''}^{q''}$ from $X_{d''}^{q''}$ (see Fig. 11). Note that $w_{c''}$ and $w_{d''}$ are either adjacent or have $u_{c''d''}$ as a common neighbour in F_1. We choose $c' \in X_{c''}^{q''}$ such that the distance from c' to $w_{c''}$ in F_1 is minimized, and we choose $d' \in X_{d''}^{q''}$ such that the distance from d' to $w_{d''}$ in F_1 is minimized. We claim that $q' = ab|c'd' \in Q$, that E' does not hit q', and that $\left|\tilde{P}_{c'd'}\right| \le 2\lg n$.

The fact that E' does not hit q' can be seen as follows: Observe that the edges that form the legs of q' belong to the legs of q'', to the path from c'' to c' or to the path from d'' to d'. By the choice of q'', E' does not hit the legs of q''. By the definition of $X_{c''}$ and $X_{d''}$, E' does not hit the paths from c'' to c' and from d'' to d' either. Thus, E' does not hit q'.

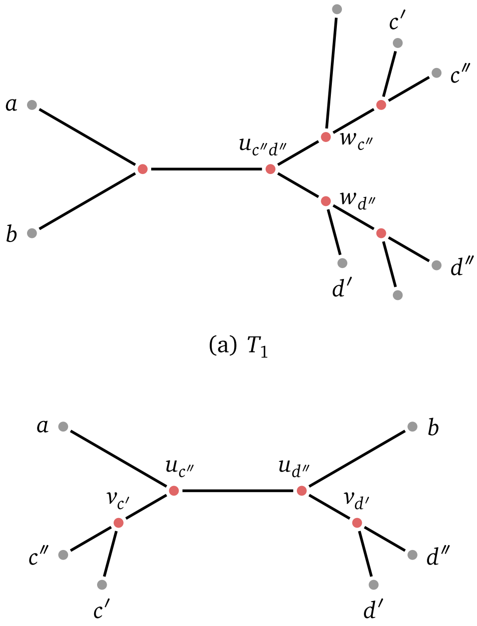
To see the bound on the length of the path $\tilde{P}_{c'd'}$: Since c' is the leaf in $X_{c''}^{q''}$ closest to $w_{c''}$ in F_1, and all internal vertices of F_1 have degree 3, we have that $\left|\tilde{P}_{c'w_{c''}}\right| \le \lg\left|X_{c''}^{q''}\right|$. By a similar argument $\left|\tilde{P}_{w_{d''}d'}\right| \le \lg\left|X_{d''}^{q''}\right|$. Then $\left|\tilde{P}_{c'd'}\right| \le \left|\tilde{P}_{c'w_{c''}}\right| + \left|\tilde{P}_{w_{c''}w_{d''}}\right| + \left|\tilde{P}_{w_{d''}d'}\right| \le \lg\left|X_{c''}^{q''}\right| + 2 + \lg\left|X_{d''}^{q''}\right| \le 2\lg n$.

It remains to prove that $q' \in Q$. To this end, let $q_{c'} = ab|c'c''$ and $q_{d'} = ab|d'd''$. Observe that $q_{c'} \notin Q$ and $q_{d'} \notin Q$. Indeed, E' does not hit the path P_{ab} nor the path $P_{c'c''}$. Thus, if $q_{c'} \in Q$, then we have $\left|X_{c''}^{q''} \cup X_{d''}^{q''}\right| \le \left|X_{c''}^{q_{c'}} \cup X_{c'}^{q_{c'}}\right|$, by the choice of q''. However, $X_{c''}^{q_{c'}} \cup X_{c'}^{q_{c'}} \subseteq X_{c''}^{q''} \subset X_{c''}^{q''} \cup X_{d''}^{q''}$, so $\left|X_{c''}^{q_{c'}} \cup X_{c'}^{q_{c'}}\right| < \left|X_{c''}^{q''} \cup X_{d''}^{q''}\right|$, a contradiction. The argument that $q_{d'} \notin Q$ is similar.

Now, let $u_{c''}$ be the degree-3 vertex in $T_2(\{a, b, c''\})$, let $u_{d''}$ be the degree-3 vertex in $T_2(\{a, b, d''\})$, let $v_{c'}$ be the vertex in $T_2(\{a, b, c''\})$ closest to c', and let $v_{d'}$ be the vertex in $T_2(\{a, b, d''\})$ closest to d'. If $v_{c'}$ is not an internal vertex of the path from $u_{c''}$ to c'', then the paths from a to b and from c' to c'' in T_2 overlap, so $q_{c'} = ab|c'c'' \in Q$, a contradiction. By an analogous argument, $v_{d'}$ must be an internal vertex of the path from $u_{d'}$ to d''. As illustrated in Fig. 11, this implies that T_2 contains the quartet $ac'|bd'$, that is, $q' = ab|c'd' \in Q$. This finishes the proof. □

To summarize: Our algorithm produces a set of conflicting quartets Q' and a set of edges E' in T_1. By Lemma 18, the quartets of Q' are pairwise leg-disjoint, as required by Proposition 16. Lemma 19 implies that we add at most $2 + 2\lg n$ edges to E' for each quartet added to Q', and thus that $|E'| \le |Q'| \cdot 2(\lg n + 1)$. Moreover, by Observation 17, the set E' hits all quartets in Q, and thus $|E'|$ is an upper bound on $d_{\mathrm{TBR}}(T_1, T_2)$. This implies that $|Q'| \ge \frac{|E'|}{2(\lg n+1)} \ge \frac{d_{\mathrm{TBR}}(T_1,T_2)}{2(\lg n+1)}$, which completes the proof of Proposition 16. □

By combining Propositions 15 and 16, we obtain that any two trees T_1 and T_2 on X satisfy $k \ge \frac{1}{27} \cdot \frac{d_{\mathrm{TBR}}(T_1,T_2)}{2(\lg n+1)}$, where $n = |X|$ and $k = d_{\mathrm{MP}}^t(T_1, T_2)$ for any $t \in \mathbb{N}_{\ge 2}^{\infty}$, from which it follows that $d_{\mathrm{TBR}}(T_1, T_2) \le 54k(\lg n + 1)$. This completes the proof of Theorem 13.



(a) T_1

(b) T_2

Fig. 11. Illustration of the proof of Lemma 19.

5. Conclusion

The central tool developed in this paper is leg-disjoint conflicting quartets. The relative flexibility of these conflicting quartets, compared to conflicting quartets that must be pairwise disjoint in both trees, was the key to establishing a lower bound on d^t_{MP} in terms of the TBR distance, which resulted in the near-linear kernel for d^t_{MP} obtained in this paper. It appears promising to approach other problems, such as improved approximation algorithms for TBR distance, from the angle of leg-disjoint incompatible quartets.

The main open question is whether d^t_{MP} admits a linear kernel, or whether the current logarithmic gap between the linear kernel for d_{MP} and our $O(k \lg k)$ kernel for d^t_{MP} reflects a real difference in difficulty between the two variants of parsimony distance. This also raises a number of related smaller questions: Is the kernel obtained using cherry reduction and chain reduction in this paper in fact a linear kernel, that is, is the logarithmic gap merely a caveat of our analysis? Can we find a larger set of leg-disjoint incompatible quartets, linear in the TBR distance, to prove that our kernel is indeed a linear kernel, or is a different technique needed to establish the linear size of our kernel? If cherry reduction and chain reduction are too weak to produce a linear kernel for d^t_{MP}, what other techniques exist to produce a smaller kernel?

CRediT authorship contribution statement

Elise Deen: Conceptualization, Methodology, Writing – original draft, Writing – review & editing. **Leo van Iersel:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing. **Remie Janssen:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing. **Mark Jones:** Conceptualization, Methodology, Writing – original draft. **Yukihiro Murakami:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing. **Norbert Zeh:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] B. Allen, M.A. Steel, Subtree transfer operations and their induced metrics on evolutionary trees, Ann. Comb. 5 (2001) 1–15.

[2] M. Bordewich, C. Semple, On the computational complexity of the rooted subtree prune and regraft distance, Ann. Comb. 8 (4) (Jan. 2005) 409–423.

[3] M. Bordewich, C. Semple, Computing the hybridization number of two phylogenetic trees is fixed-parameter tractable, IEEE/ACM Trans. Comput. Biol. Bioinform. 4 (3) (July 2007) 458–466.

[4] T.C. Bruen, D. Bryant, Parsimony via consensus, Syst. Biol. 57 (2) (2008) 251–256.

[5] M. Fischer, S. Kelk, On the maximum parsimony distance between phylogenetic trees, Ann. Comb. 20 (1) (Mar. 2016) 87–113.

[6] W.M. Fitch, Toward defining the course of evolution: minimum change for a specific tree topology, Syst. Biol. 20 (1971) 406–416.

[7] J.A. Hartigan, Minimum mutation fits to a given tree, Biometrics 29 (1973) 53.

[8] M. Jones, S. Kelk, L. Stougie, Maximum parsimony distance on phylogenetic trees: a linear kernel and constant factor approximation algorithm, J. Comput. Syst. Sci. 117 (May 2021) 165–181.

[9] S. Kelk, M. Fischer, On the complexity of computing MP distance between binary phylogenetic trees, Ann. Comb. 21 (4) (Dec. 2017) 573–604.

[10] S. Kelk, M. Fischer, V. Moulton, T. Wu, Reduction rules for the maximum parsimony distance on phylogenetic trees, Theor. Comput. Sci. 646 (2016) 1–15.

[11] S.M. Kelk, S. Linz, A tight kernel for computing the tree bisection and reconnection distance between two phylogenetic trees, SIAM J. Discrete Math. 33 (2019) 1556–1574.

[12] S.M. Kelk, G. Stamoulis, A note on convex characters, Fibonacci numbers and exponential-time algorithms, Adv. Appl. Math. 84 (2017) 34–46.

[13] V. Moulton, T. Wu, A parsimony-based metric for phylogenetic trees, Adv. Appl. Math. 66 (2015) 22–45.

[14] R. van Wersch, S. Kelk, S. Linz, G. Stamoulis, Reflections on kernelizing and computing unrooted agreement forests, Ann. Oper. Res. 309 (1) (2022) 425–451.