Branch Length Transforms using Optimal Tree Metric Matching

Shayesteh Arasti,[1†] Puoya Tabaghi,[2†] Yasamin Tabatabaee[3] and Siavash Mirarab[4*]

[1] *Department of Computer Science and Engineering, UC San Diego, CA 92093, USA*
[2] *Halıcıoğlu Data Science Institute, UC San Diego, CA 92093, USA*
[3] *Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA*
[4] *Department of Electrical and Computer Engineering, UC San Diego, CA 92093, USA*
[*] Corresponding author: smirarab@ucsd.edu

Abstract

The abundant discordance between evolutionary relationships across the genome has rekindled interest in methods for comparing and averaging trees on a shared leaf set. However, most attempts at comparing and matching trees have focused on tree topology. Comparing branch lengths has been more elusive due to several challenges. Species tree branch lengths can be measured in various units, often different from gene trees. Moreover, rates of evolution change across the genome, the species tree, and specific branches of gene trees. These factors compound the stochasticity of coalescence times and estimation noise, making branch lengths highly heterogeneous across the genome. For many downstream applications in phylogenomic analyses, branch lengths are as important as the topology, and yet, existing tools to compare and combine weighted trees are limited. In this paper, we address the question of matching one tree to another, accounting for their branch lengths. We define a series of computational problems called Topology-Constrained Metric Matching (TCMM) that seek to transform the branch lengths of a query tree based on a reference tree. We show that TCMM problems can be solved in quadratic time and memory using a linear algebraic formulation coupled with dynamic programming preprocessing. While many applications can be imagined for this framework, we explore two applications in this paper: embedding leaves of gene trees in Euclidean space to find outliers potentially indicative of errors and summarizing gene tree branch lengths onto the species tree. In these applications, our method, when paired with existing methods, increases their accuracy at limited computational expense.

Keywords

Phylogenomics, Convex Optimization, Species Tree Branch Length, Phylogenetic Branch Length Estimation

Code and Data availability

The software is available at https://github.com/shayesteh99/TCMM.git. Data is available on Github https://github.com/shayesteh99/TCMM-Data.git.

[†] These authors contributed equally to this work.

Introduction

Discordance among evolutionary histories across the genome (Maddison, 1997) is now understood to be pervasive and has motivated the study of region-specific histories (gene trees) and the speciation history (the species tree). Having alternative trees leads to a need for comparing and combining trees. We need metrics to quantify tree dissimilarity, algorithms for mapping gene trees to the species tree to reconcile them (Doyon et al., 2011), and median tree methods to combine gene trees into a single species tree (Barthélemy and McMorris, 1986). Solutions to these problems have been developed and are widely adopted. Measures of tree difference such as Robinson and Foulds (1981) (RF), quartet distance (Estabrook et al., 1985), and matching split (MS) (Bogdanowicz and Giaro, 2012) are commonly used to quantify gene tree differences. Biologists often use reconciliation methods in comparative genomics to understand the evolution of gene function. Inferring the species tree by finding the tree with the minimum total quartet distance to gene trees (quartet median tree) has been widely adopted due to its statistical consistency under several models of gene tree discordance (Mirarab et al., 2021).

The rich literature on comparing gene trees and species trees has focused on tree topology far more than branch length. Discordance, however, is not just limited to topology. One of the main causes of discordance is incomplete lineage sorting (ILS) as modeled by the multi-species coalescent (MSC) model (Degnan and Rosenberg, 2009). Under MSC, internal nodes of a gene tree correspond to coalescence events that predate species divergences. As a result, gene tree branch lengths are guaranteed to differ from the species tree even if there is no topological discordance (a phenomenon Edwards and Beerli (2000) referred to as species versus genic divergences.) Beyond true biological differences, challenges in inferring gene trees compound the problem. From sequences alone, we can only measure branch lengths in the unit of the expected number of substitutions per site (SU), which are a function of not just divergence times but also mutation rates, that vary across the tree, across the genome, or both (Kumar, 2005; Lopez et al., 2002; Rasmussen and Kellis, 2007). Furthermore, gene trees are often inferred from short sequences (to avoid recombination within a gene), which makes branch length estimates noisy, especially for short branches.

Branch lengths are as important as topology to many downstream applications such as dating, diversification studies, comparative genomics, trait evolution, and even microbiome analyses. Nevertheless, the myriad sources of heterogeneity (ILS, rate change, noise) have made it challenging to reconcile gene tree branch lengths with each other and with the species tree. Bayesian co-estimation methods (e.g., Boussau et al., 2013; Heled and Drummond, 2010; Liu, 2008) account for branch lengths by jointly modeling gene trees and species trees and introducing relevant parameters. These methods tend to be accurate (Yang and

Warnow, 2011) and readily produce a reconciliation of species trees and gene trees. However, they do not scale to large datasets (Ogilvie et al., 2017; Zimmermann et al., 2014). Moreover, since complex rate models further reduce scalability, these methods often model only limited rate heterogeneity.

The alternative two-step approach of inferring gene trees and then summarizing them to find the species tree often ignores branch lengths. Theoretically justifiable methods that consider branch lengths in inferring species trees (e.g., GLASS and STEM; Kubatko et al., 2009; Mossel and Roch, 2010) have been less accurate than topology-focused methods given estimated gene trees (DeGiorgio and Degnan, 2014; Yang and Warnow, 2011), partially due to challenges mentioned earlier (Huang et al., 2010). Others have focused on gene tree branch lengths (SU) for rooting them (Mai et al., 2017; Tria et al., 2017), for finding outliers indicative of errors (de Vienne et al., 2012; Mai and Mirarab, 2018), for clustering gene trees (Gori et al., 2016), and for summarizing them to infer the species tree branch lengths (Binet et al., 2016; Tabatabaee et al., 2025, 2023). These attempts have been more successful but lack an underlying mechanism for matching trees that can be used across applications. Each deals in some way with the underlying difficulty that two trees that differ in both topology and branch length are hard to compare since the same branches are not present in both (see Kuhner and Yamato (2015) for tree comparison metrics that include branch length).

To arrive at a general method for comparing trees while accounting for branch lengths, we propose a framework that decouples topological changes and branch length changes to an adjustable extent. We use a linear algebraic formulation based on patristic distances (pairwise path lengths) to define an optimization problem called Topology-Constrained Metric Matching (TCMM). The goal of this problem is to assign branch lengths to a query tree such that its patristic distances match a set of one or more reference trees as much as possible without changing its topology. Using an additional regularization term, we can balance the competing goals of matching the reference tree and keeping the original branch length patterns of the query tree. All TCMM problems are convex and can be solved efficiently (in quadratic time and memory) using standard techniques, paired with a dynamic programming preprocessing step to prepare the inputs. While this ability to align the metric structure of one tree with another offers a powerful tool, in this paper, we focus on two particular applications: 1) embedding gene tree leaves in a shared space to enable outlier detection, and 2) transferring gene tree branch lengths to a species tree. We show that in both applications, using TCMM to match gene tree branch lengths paired with existing methods can improve their accuracy.

Material and Methods

Problem formulation

For an unrooted tree $T = (V_T, E_T)$, we let L_T denote the leaf set and $P_T(v, v')$ denote the path between any pairs of vertices $v, v' \in V_T$. For a weighted tree, each edge $e \in E_T$ has length $w_T(e) \in \mathbb{R}^+$. We use $\mathcal{T}(T)$ to denote the set of all weighted trees with the same topology as tree T. A weighted tree T with N vertices indexed by $[N] = \{1, \ldots, N\}$ defines a metric $d_T : V_T \times V_T \to \mathbb{R}^+$ where $d_T(v, v') = \sum_{e \in P_T(v,v')} w(e)$. We define the *weight* vector $w_T = (w_T(e))_{e \in E_T} \in \mathbb{R}^{|E_T|}$ as a vector of all the weights indexed by edges and, with a slight abuse of notation, the *pairwise distance* vector $d_T = (d_T(v, v'))_{v,v' \in L_T, v<v'} \in \mathbb{R}^{\binom{|L_T|}{2}}$ as a vector of all patristic distances indexed by pairs of nodes. We define a binary *path matrix* A_T of size $\binom{|L_T|}{2} \times |E_T|$ where rows correspond to pairs of vertices in L_T, columns represent the edge set. For two distinct vertices $v, v' \in L_T$ and edge $e \in E_T$, an element $((v, v'), e)$ of A_T is one if and only if the edge e is on the path between v and v'. Otherwise, it is equal to zero. The path matrix A_T completely encodes the topology of T.

Consider a query tree T and a reference weighted tree R, both on a shared leafset L. For any tree T' with the same topology as T, let $\text{cost}(T'; R)$ measure the branch length discrepancy of T' to the reference tree R. For any such cost function, we can seek $T^* \in \arg\min_{T' \in \mathcal{T}(T)} \text{cost}(T'; R)$. Since the topology is fixed in this problem, we can reparameterize the cost with respect to the weight vector w_T to get $w^* \in \arg\min_{w'} \text{cost}(w'; R)$, where $w' \in \mathbb{R}^{+|E_T|}$ are constrained to be nonnegative. In this paper, we focus on a particular cost function to quantify the discrepancy of metric information between the two trees and successively refine it. Akin to least squares error distance-based tree estimation, we set

$$\text{cost}(w_T; R) = \sum_{v,v' \in L} |d_R(v, v') - d_T(v, v')|^2 = \|d_R - d_T\|_2^2$$

as the square of the ℓ_2 distance between pairwise distance vectors. As it has been long appreciated (see Desper and Gascuel, 2003), we can find a separable form for pairwise distances in terms of topology and edge weights. For $v, v' \in L$, we can write $d_T(v, v') = \sum_{e \in P_T(v,v')} w_T(e) = \langle a_{v,v'}, w_T \rangle$ where $a_{v,v'} = \left(1_{e \in P_T(v,v')}\right)_{e \in E_T} \in \{0, 1\}^{|E_T|}$ and 1_x is the indicator function. Note that $a_{v,v'}$ is a row of the path matrix A_T. Thus, for any tree $T = (V_T, E_T)$, we have $d_T = A_T w_T$, and

$$\text{cost}(w_T; R) = \sum_{v,v' \in L} |d_R(v, v') - d_T(v, v')|^2 = \|d_R - A_T w_T\|_2^2 \,. \tag{1}$$

We can now define a specific problem, similar to the ordinary least squares problem:

Problem 1 (Topology-Constrained Metric Matching— under ℓ_2^2)**.** *Let T, R be weighted trees with edge weights w_T and w_R on the common leafset L with at least two elements. The weights of the closest tree to R matching T in topology with edge weights lower bounded by $\varepsilon \geq 0$ is:*

$$w^* \in \underset{w \in \mathbb{R}^{|E_T|}, w \geq \varepsilon}{\arg\min} \|d_R - A_T w\|_2^2.$$

The cost function in Problem 1 is agnostic to the original branch lengths of T, w_T, and thus, it can dramatically alter them. For some applications, it may be desirable to retain the original branch lengths to some user-adjustable extent. We can accommodate that goal by adding a regularizing term. For an optimal edge weight vector w_{T^*}, we define the *adjustment rate vector* by element-wise dividing w_{T^*} and w_T, to get w_{T^*}/w_T. To incorporate w_T in the problem statement, we penalize the deviation of estimated w_{T^*} from the initial weight vector w_T, save for a total scaling factor shared by all branches, as in the following problem.

Problem 2 (Regularized Topology-Constrained Metric Matching — under ℓ_2^2)**.** *Let T, R be weighted trees with edge weight functions w_T and w_R on the common leafset L. Let λ be a fixed regularization coefficient and $\sigma^2(w/w_T)$ denote the variance of the adjustment rates vector w/w_T. We seek the tree $T^* \in \mathcal{T}(T)$ with edge weights*

$$w_{T^*} \in \underset{w \in \mathbb{R}^{|E_T|}, w \geq \varepsilon}{\arg\min} \|d_R - A_T w\|_2^2 + \lambda \sigma^2(w/w_T) \ . \tag{2}$$

The regularizing term, which is a quadratic function of the weight vector w, leads to lowered variance in the adjustment rate vector, w/w_T, and thus, lowered change in *relative* branch lengths compared to T. Clearly, if the variance of adjustment rate values is 0, the tree T has been simply scaled up or down, without any real change to its relative branch lengths. Higher λ values encourage less change to the relative branch lengths of T. As $\lambda \to \infty$, all rate multipliers converge to a solution that simply scales branch lengths of T with a fixed global rate $r*$:

$$r^* \in \underset{r \cdot \min_e w_T(e) \geq \varepsilon}{\arg\min} \|d_R - r A_T w_T\|_2^2 \ .$$

To make λ values more interpretable across all datasets with varying sizes and heights, we first compute the optimal branch lengths w^* according to Problem 1. We then reformulate Problem 2 as:

$$w_{T^*} \in \underset{w \in \mathbb{R}^{|E_T|}, w \geq \varepsilon}{\arg\min} \frac{\|d_R - A_T w\|_2^2}{\text{cost}(w^*; R)} + \lambda \sigma^2(w/w_T).$$

where $\text{cost}(w^*; R)$ is defined in Eq. (1). This adjustment ensures that the first term is scale-free and provides an initial solution that can be used as input for the optimizer. We will use this revised formulation of the Regularized TCMM Problem in all experiments. We can also extend to multiple inputs.

Problem 3 (Topology-Constrained Consensus Metric Matching)**.** *Let $\{R_k : k \in [K]\}$ be a set of weighted reference trees, each with weights w_k. Given a tree T, we seek to find the optimal (adjusted) tree T^* with the same topology as T and weights w_{T^*} such that:*

$$w_{T^*} \in \underset{\substack{w \in \mathbb{R}^{|E_R|} \\ w \geq \varepsilon}}{\arg\min} \sum_{k \in [K]} \alpha_k^2 \|d_{R_k} - A_T w\|_2^2 + \lambda \sigma^2(w/w_T),$$

where α_k is a nonnegative weight controlling the contribution of reference tree R_k in the cost function, $\lambda > 0$ is a regularizing coefficient, $\sigma^2(w/w_T)$ penalizes the weights that result in a large variance for the adjustment rate vector w/w_R, and $\varepsilon \geq 0$ is a lower bound for the estimated edge weights.

Remark 1 (supplementary material) shows that problem 3 can be solved by solving Problem 2 after averaging distances across k genes.

A Scalable Convex Optimization Solution

TCMM problems are all convex

We can rewrite the cost functions in Problems 1–3 in the quadratic form. For simplicity, we use $A = A_T$ and $d = d_R$. Note that Problem 1 minimizes $l_1(w) \overset{\text{def.}}{=} \|d - Aw\|_2^2$. We convert l_1 to the quadratic form

$$l_1(w) = \frac{1}{2} w^\top B w + c^\top w + \text{const}, \tag{3}$$

where $B = 2A^\top A$ is a positive semidefinite matrix, $c = -2A^\top d$, and const $= \|d\|_2^2$. Since $B \succeq 0$ by construction, l_1 is a convex function of the weight parameter w. Adding the regularization term $\sigma^2(r)$ — where $r = w/w_T$ — does not change the convexity as we can convert it to the following quadratic form:

$$l_2(w) \overset{\text{def.}}{=} \sigma^2(r) = \frac{1}{|E|}(r - 1\bar{r})^\top (r - 1\bar{r}) = \frac{1}{|E|} r^\top (P_1^\perp)^\top P_1^\perp r = \frac{1}{|E|} r^\top P_1^\perp r$$

where $1 \in \mathbb{R}^{|E|}$ is the vector of all ones, $\bar{r}$ is a scalar that is the average value of elements in r, $P_1^\perp = I - \frac{1}{|E|} 1 1^\top$. It is a known fact that $P_1^\perp$ is an orthogonal projection operator, that is, $(P_1^\perp)^\top P_1^\perp = P_1^\perp$. Furthermore, it is a positive semidefinite matrix with $0 - 1$ eigenvalues. Therefore, the regularizing term $\sigma^2(w/w_T)$ is indeed a convex function of the weight parameter w. Thus, Problems 1,2, and 3 all admit globally optimal solutions.

Proposition 1. *The cost functions in Problems 1, 2, and 3 admit unique minimums but not necessarily unique minimizers. We can arrive at an optimal solution using a projected gradient descent method.*

Proofs are provided in the supplementary material, Appendix A.6. By Proposition 1, we can use projected

gradient descent to solve Problems 1 – 3 using the gradient terms shown in the proof. We implemented and optimized the quadratic functions l_1 and l_2 using built-in functions in the Python package `CVXPY`.

Dynamic programming enables calculating B and c in quadratic time

Luckily, our formulation does not need to ever compute A_T and d_R — whose dimensions grow cubically with N. Rather, it needs the matrix B, which grows quadratically, and vector c, which grows linearly with N. Naively computing $A_T^\top A_T$ from its building blocks requires $O(N^4)$ time and $O(N^3)$ memory. Similarly, computing $A_T^\top d_R$ naively requires $O(N^3)$ time and $O(N^3)$ memory. However, we can compute B and c directly in quadratic time and memory using dynamic programming without ever computing A_T and d_R.

Computing $c = A_T^\top d_R \in \mathbb{R}^{|E_T|}$. Each edge $e \in E_T$ corresponds to a bipartition of leaves of T. For an arbitrary rooting of T, we define $C_T(e)$ as the set of leaves of T below e and $C'_T(e)$ as $L_T \setminus C_T(e)$. As Remark 2 (supplementary material) shows, the element of c that corresponds to e is

$$c_e = \sum_{v \in C_T(e)} \sum_{v' \in C'_T(e)} d_R(v, v') \tag{4}$$

To compute this sum, we color each leaf $v \in L_R$ in the reference tree R as blue if $v \in C_T(e)$ and red if $v \in C'_T(e)$ (See Figure 1). Using this coloring scheme and a single bottom-up traversal of the reference tree (Algorithm 1), c_e can be computed in linear time for each edge $e \in E_T$ and thus quadratic time and memory for all edges. In the consensus Problem 3, if no distances are missing, we can simply average the terms $\{A_T^\top d_{R_k}\}_{k\in[K]}$ instead of averaging distance vectors $\{d_{R_k}\}_{k\in[K]}$. The consensus cost function allows this:

$$\sum_{k\in[K]} \alpha_k^2 \|d_{R_k} - A_T w\|_2^2 + \lambda\sigma^2(w/w_T) = w^\top A_T^\top A_T w \Big(\sum_{k\in[K]} \alpha_k^2\Big) - 2\Big(\sum_{k\in[K]} \alpha_k^2 \big(A_T^\top d_{R_k}\big)^\top\Big) w + \lambda\sigma^2(w/w_T) + \text{const}$$

Computing $B = A_T^\top A_T$. Given the definition of the path matrix A_T, it can be easily deduced that for two edges e and e', $B_{e,e'}$ is the number of pairs of leaves $v, v' \in L$ where $e, e' \in P_T(v, v')$. We easily pre-compute $|C_T(e)|$ and $|C'_T(e)|$ for each edge $e \in E_T$ in linear time and the least common ancestor (LCA) of all pairs of nodes of T in quadratic time. With these, we can easily compute $B_{e,e'}$ for all $e, e' \in E_T$ in quadratic time using simple calculations given in Algorithm 1 (supplementary material).

We have assumed so far that query and reference trees have the same species, but this will not always be the case. Gene trees almost always miss some taxa due to gene loss or data incompleteness. Leaves of the reference tree R that are absent from the query tree T do not create a problem as they can be easily pruned from R without affecting the optimization problem. Leaves in T missing from R need care. Appendix A.4 details how missing data are handled in our approach, and Figure S1 provides evidence of its effectiveness.

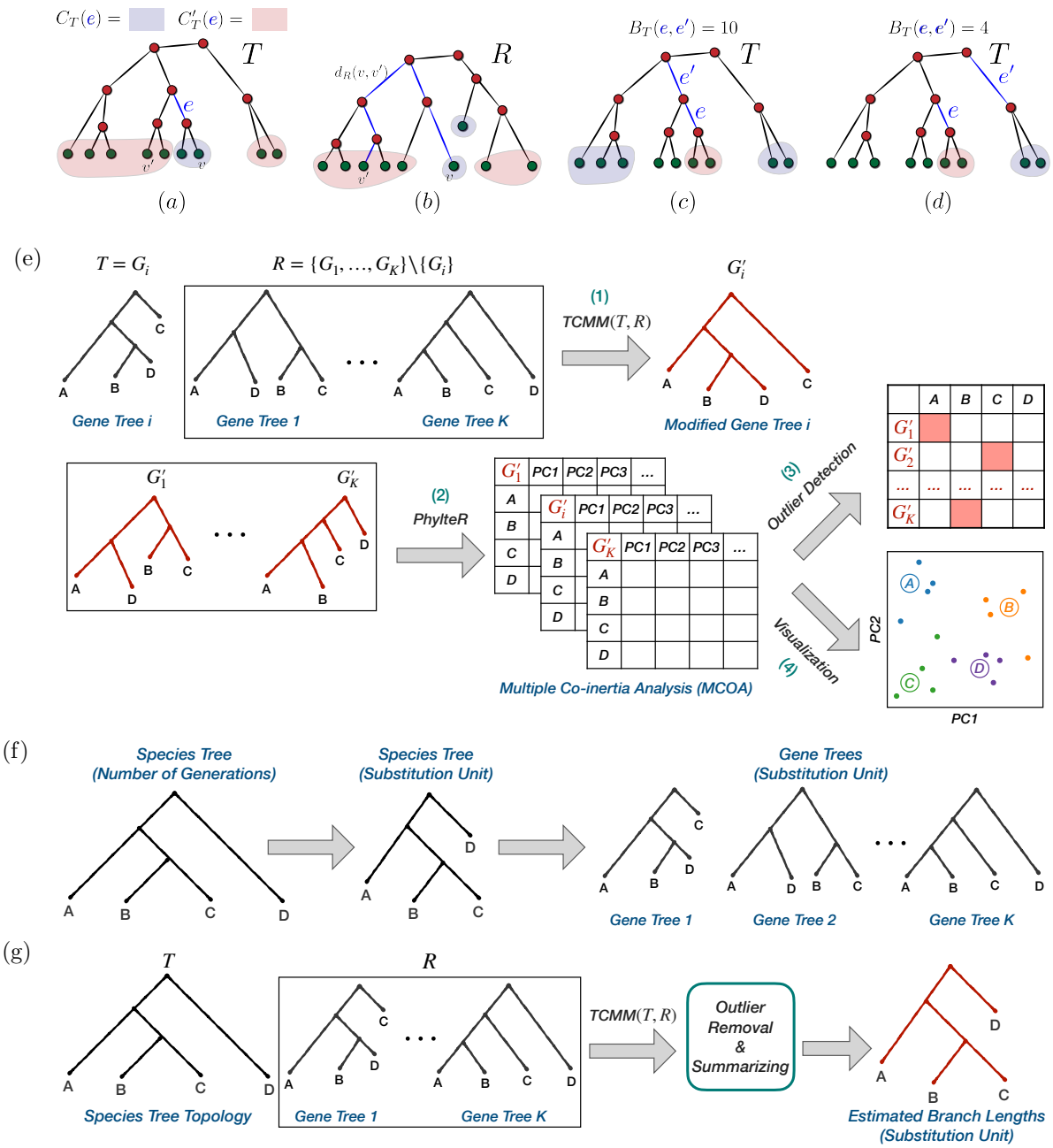

Figure 1: (a) Tree T with $C_T(e)$ and $C'_T(e)$ highlighted in blue and red for an edge e. (b) Tree R; Recall $c_e = \sum_{v \in C_T(e)} \sum_{v' \in C'_T(e)} d_R(v, v')$. An Example path $P_T(v, v')$ corresponding to $d_R(v, v')$ is shown in blue. (c, d) For edges e, e', $B_T(e, e')$ is the number of pairs of leaves v, v' where $e, e' \in P_T(v, v')$. c) If e is a descendant of e', $B_T(e, e') = C_T(e)C'_T(e')$. d) Otherwise, $B_T(e, e') = C_T(e)C_T(e')$. (e) The TCMM pipeline for outlier detection. Each modified gene tree G'_i is the output of running TCMM with the original gene tree G_i as the query and the rest of gene trees as the reference trees. The modified gene trees are then fed into the PhylteR pipeline where the multiple co-inertia analysis (MCOA) is performed. Lastly, the principal coordinates (PCs) for each modified gene tree are used for the outlier detection and visualization tasks. (f) Assumed data generation model and (g) The TCMM pipeline for branch length estimation with the species tree topology as the query and the gene trees with branch lengths in substitution unit (SU) as the reference trees. The final output is the species tree with SU branches.

Application 1: Outlier in Gene trees using MCOA

The phylogenomic analysis pipeline introduces various forms of error into gene trees, compounding biological processes such as ILS that create real gene tree discordance. Some errors leave subtle traces, while others distort the inferred trees more dramatically by introducing abnormally long branches or drastically altering the tree topology. These errors can impact downstream analyses (Laurin-Lemay et al., 2012; Philippe et al., 2017; Salichos and Rokas, 2013; Springer and Gatesy, 2018). As a result, several efforts have been made to detect anomalies in gene trees automatically. Mai and Mirarab (2017) proposed the TreeShrink method to identify species with unexpectedly long branches in a tree. Gori et al. (2016) proposed clustering gene trees using a metric of distance. Going further, de Vienne et al. (2012) introduced Phylo-MCOA to use the multiple co-inertia analysis (MCOA) to embed all gene trees into a single Euclidean space such that pairwise distances of leaves match those of the input trees as much as possible. Using the principal coordinates of this space enables the detection of outlier genes and species in a two-step process employing Tukey's method (Hubert and Vandervieren, 2008) followed by more complex steps. A recent implementation of Phylo-MCOA is available in an R package called PhylteR (Comte et al., 2023).

Here, we propose to use TCMM with PhylteR to improve outlier detection with the following rationale. Differing rates of evolution and heterotachy can make pairwise distances across different gene trees dissimilar and render embeddings noisy, even absent any errors. Those changes can dominate principal coordinates, masking real outliers. TCMM, by matching gene trees to a shared metric space, has the potential to better reveal outliers. With $\lambda = 0$, we match gene trees maximally to each other and thus expect to find topological outliers only. As λ increases, the ratios between original branch lengths are increasingly maintained, giving us a way to control whether topological or branch length outliers are sought. We apply the unweighted consensus version of TCMM to each gene tree, using it as the query tree and treating the remaining gene trees as reference trees. The modified gene trees are then fed into the PhylteR pipeline to generate Euclidean embeddings, which are used for visualization and outlier detection. For simplicity, we skip the second step of outlier detection used in Phylo-MCOA and report only the outliers identified through Tukey's method. We apply this procedure with and without regularization and examine how varying the regularization coefficient ($\lambda \in \{0, 1\text{e}{-5}, 1\text{e}{-4}, \ldots, 10, 100\}$) influences the type and number of outliers detected. The same process is repeated with the original gene trees without running TCMM.

We evaluate TCMM's performance on both simulated and biological datasets, where the true or estimated gene trees have branch lengths measured in units of expected substitutions per site (SU). To measure the effectiveness of outlier detection, we calculate the improvement in the Robinson and Foulds (1981) (RF) distance to the species tree S above random removal, using a metric called ΔRF, and the normalized quartet

score (nqs) of the species tree, both defined in Appendix A.2. We quantify whether outliers are the same as those found by the branch length-based method TreeShrink by reporting the containment Jaccard similarity between the outliers detected by TCMM +PhylteR and those detected by TreeShrink.

Application 2: Species tree branch lengths

Two-step methods of species tree inference produce a species tree topology from a set of gene trees. To be useful for downstream analyses (e.g., dating), the species tree requires branch lengths. Scalable and accurate methods such as ASTRAL (Mirarab et al., 2014b) and NJst/ASTRID (Liu and Yu, 2011; Vachaspati and Warnow, 2015) produce only topology. We can assign coalescent unit (CU) branch lengths for internal branches of the species tree using gene tree topologies alone, but this approach can lead to systematic underestimation (Forthman et al., 2022; Sayyari and Mirarab, 2016) and the lack of terminal branch lengths makes it less useful. Since gene trees have SU branch lengths and many downstream analyses require SU lengths, we can seek to transfer the SU branch lengths from gene trees onto the species tree. Binet et al. (2016) have proposed a method called ERaBLE that uses a least squares optimization applied to patristic distances of the gene trees to estimate species tree lengths, effectively taking a weighted average. ERaBLE allows for rate heterogeneity only across loci. It estimates a single scalar rate per gene tree by matching them simultaneously to the species tree. Tabatabaee et al. (2025, 2023) have shown that ERaBLE overestimates terminal branches because it ignores deep coalescence. They propose a coalescent-based method called CASTLES (-Pro) to explicitly model the differences between the species and genic divergence times. While CASTLES-Pro is more accurate than ERaBLE in MSC simulations (Tabatabaee et al., 2025), its accuracy can decrease under high levels of HGT. This observation raises the question: Can a flexible model-free method assign SU lengths to the species tree while attaining high accuracy when gene tree heterogeneity is not solely due to ILS?

Here, we use TCMM to summarize gene tree SU branch lengths onto the species tree. Unlike ERaBLE, which estimates only one scalar parameter per gene tree, we assign one parameter for each branch of each gene tree. Counterintuitively, we use gene trees as the reference and the species tree as the query in one of two ways. One approach, called consensus, uses Problem 3 to jointly match the species tree (as query T) to multiple gene trees (as the reference R). Here, we simply set all the α_k weights to 1. Our second (and default) approach is to run TCMM per gene (i.e., Problem 1), and summarize results at the end. Results provide us with a set of estimated branch lengths $\tilde{w}_{e,k}$ where $e \in E_S$ and $G_k \in \mathcal{G}$. These lengths can then be further summarized into a single value $\hat{w}_e$ for each branch using common averaging methods — arithmetic mean, harmonic mean, and geometric mean. When using arithmetic means, this approach is similar to using the consensus method; Remark 3 (supplementary material) shows that the two are identical if all output lengths

are strictly positive. Despite this (near) equivalence, the per-gene approach offers the benefit of enabling outlier removal. We remove both extremely small outliers introduced by TCMM and large outliers, which can arise due to gene tree errors using the procedures detailed in Appendix A.3. We address missing data using the approaches noted in Appendix A.4.

We can pair TCMM with other branch length estimation tools, such as CASTLES-Pro (Tabatabaee et al., 2025), by applying the regularized version of TCMM (Problem 2). This approach will preserve existing branch lengths to an extent controlled by λ. One challenge is selecting the parameter λ. Here, we use an automatic approach to select λ from a predefined set of options. We do this by choosing the λ that minimizes the sum of the estimated species tree branch lengths. This choice is rooted in the minimum-evolution principle (Rzhetsky and Nei, 1993) used by a family of distance-based approaches.

We evaluate TCMM for this application using both simulated and real datasets. On simulated data, the model species tree branches have true SU lengths (w) obtained by multiplying their lengths (number of generations) with varying mutation rates, creating non-ultrametric trees. We compare the true w values to estimated values $\hat{w}$ to quantify the mean absolute error ($|w - \hat{w}|$), absolute logarithmic error ($|\log(\hat{w}/w)|$), and bias ($\hat{w} - w$), averaged across all branches of each species tree after excluding the outgroup. We compare TCMM to four leading methods studied by Tabatabaee et al. (2025), all run on the true species tree topology. We include CASTLES-Pro, which accounts for coalescence times predating species divergence times. We examine the method most commonly used by biologists: using RAxML (Stamatakis, 2006) on concatenated gene alignments to estimate branch lengths of a fixed species tree. We also include two methods that like TCMM use patristic distances. A simple alternative is averaging patristic distance matrices across gene trees and using a distance-based method (FastME, Lefort et al., 2015) to assign branch lengths to S. A more advanced version of this idea is ERaBLE (default), mentioned earlier.

Datasets

We studied one simulated dataset and five biological datasets for application 1 and two simulated datasets and three biological datasets for application 2. The biological datasets are detailed in the results section.

S100 simulated dataset. We reused a published 101-taxon 1000-gene simulated dataset by Zhang et al. (2018), which Tabatabaee et al. (2023) have updated to include species trees with SU branch lengths. Ultrametric species trees with generation times are simulated using a birth-death model; each branch length is multiplied by a rate multiplier drawn from a Gamma distribution (mean: 1; variance: from 0.0153 to 2.6027) and a mutation rate to get SU length. The 50 replicates of this dataset are highly heterogeneous in terms of ILS, with the mean RF distance between the model species tree and the true gene trees (AD) at

46%. Gene trees were estimated using FastTree-2 (Price et al., 2010) from 200bp, 400bp, 800bp, and 1600bp alignments, leading to average RF distance between true and estimated gene trees of 23%, 31%, 42% and 55%, respectively. We use this dataset in both applications.

HGT simulated dataset. We generated a 51-taxon (50 ingroups + 1 outgroup) simulated dataset with gene tree discordance due to ILS and Horizontal Gene Transfer (HGT). We used the parameters of Davidson et al. (2015) but generated species trees with branch lengths in SU and CU units, using a modified version of SimPhy (Mallo et al., 2016) created by Tabatabaee et al. (2023). This dataset has six model conditions that are characterized by different rates of HGT; the base ILS level is moderate (30% AD) and the same across all model conditions. The HGT rates for the six model conditions (0, 2e-9, 5e-9, 2e-8, 2e-7, 5e-7) correspond to the expected number of HGT events per gene: 0, 0.08, 0.2, 0.8, 8, and 20, respectively. Total discordance is a product of both ILS and HGT. The AD levels of the six conditions are 30%, 30%, 31%, 34%, 53%, and 68%, respectively, with the increase in AD being entirely due to HGT. This dataset has 50 replicates, each containing 1000 gene trees. We use the estimated gene trees for application 2.

Results

Application 1: topology focused MCOA

Simulated Dataset (S100)

The MCOA embeddings of species across gene trees become far more concentrated when they are first matched to each other using TCMM, and they become progressively more dispersed as λ increases (Fig. 2ab and Fig. S2a). The mean distance from per-gene embeddings to the central embedding (computed by MCOA) for each species decreases by a factor of 4.8 as we move from the original gene trees to $\lambda = 0$. Thus, as expected, TCMM matching is making gene trees more similar to each other.

The more concentrated embeddings have a major impact on outlier detection, as evident from how the choice of λ influences which outliers are detected by TCMM+ PhylteR. When λ is high, similar to not using TCMM, PhylteR flags gene trees with unusually long branches as outliers (e.g., gene 16 in Fig. 2c). As λ decreases, TCMM adjusts gene tree branch lengths to better match median pairwise distances, causing those gene trees to look normal, while gene trees with highly discordant topologies are flagged as outliers (e.g., gene 17 in Fig. 2b). Thus, at lower λ, topology plays a larger role in outlier detection, while at higher λ and the original trees, branch lengths become more significant. We can quantify this pattern by measuring the similarity of outlier sets detected by PhylteR and TreeShrink (which focuses on aberrant branch lengths).

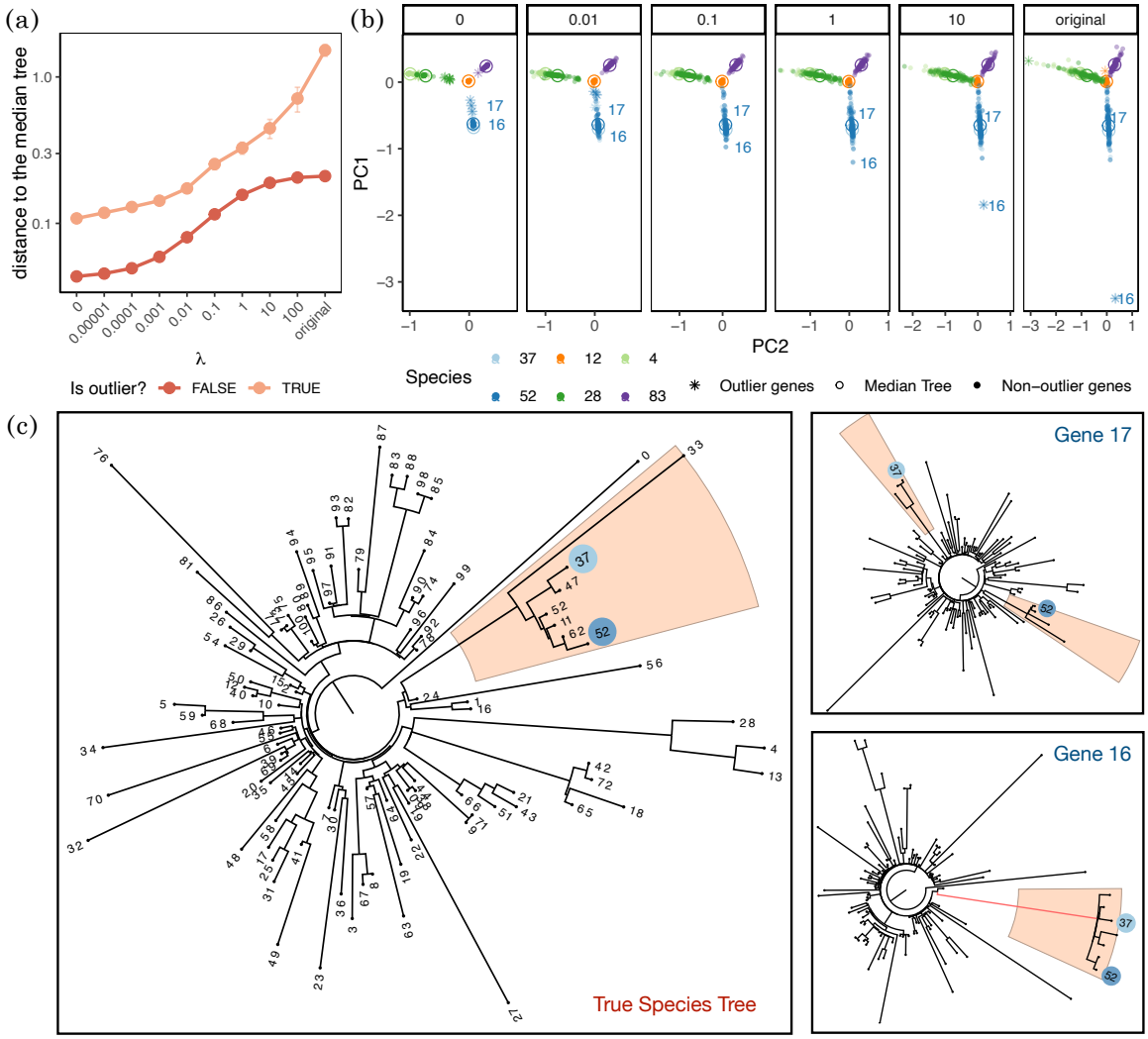

Figure 2: (S100 Dataset) (a) Comparing the euclidean distance to the median tree for true gene trees. (See Fig. S2a for all levels of gene tree error). (b) The PCA plot shows the projections of six species onto the first and second principal coordinates (PCs) with different regularization coefficient (λ) values. The position and the status (outlier or non-outlier) of genes 16 and 17 are marked for species 37 and 52 on each plot. (c) An example of the outliers detected by TCMM+PhylteR for one replicate. Left: the true species tree with the two species 37 and 52 and their corresponding subtree highlighted. Right: two outlier genes (16 and 17) for these species. Gene 17 is an outlier due to discordance from the true species tree topology, whereas gene 16 is an outlier due to a long branch (marked in red) separating species 37 and 52 from other species.

We observe that decreasing λ successively makes the outliers detected by TCMM+ PhylteR less similar to those detected by TreeShrink (Fig. 3a).

The reduced emphasis on branch lengths makes the detected outliers more aberrant topologically (Fig. 3a). Compared to random removal of the same number of leaves, outliers detected by TCMM+ PhylteR lead to a greater reduction in RF distance for lower λ values. For example, with 1600bp gene trees, ΔRF is around 3 with small λ and only 0.2 with PhylteR without TCMM. Although this pattern holds across all levels of gene tree error, it is more pronounced in higher-quality gene trees.

Incorporating TCMM with PhylteR also increases the similarity of the gene trees to the true species tree

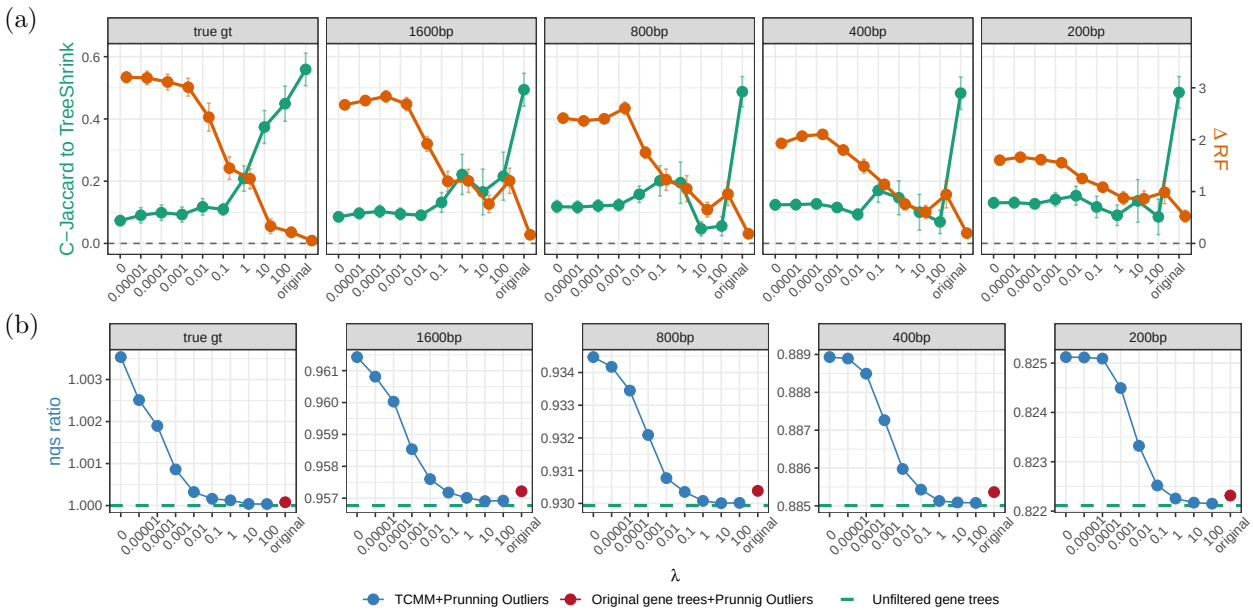

Figure 3: S100 Dataset. (a) Similarity to TreeShrink outliers in Containment Jaccard (green) and ΔRF distance (orange), defined as the decrease in the RF distance of gene trees to the species tree after removing outliers, normalized by the same quantity with random removal of the same number of leaves (Appendix A.1). Above 0 indicates better than random. (b) Comparing the ratio of the normalized quartet score to the true species tree over the true gene trees for different levels of gene tree error. The unfiltered estimated and true gene trees are shown in green.

(which was not used in the filtering) after pruning outliers (Fig. 3b). The increase in the quartet similarity between the gene trees and the species tree is especially pronounced at lower λ values, with $\lambda = 0$ showing the greatest improvement across all levels of gene tree error. Two factors contribute to this pattern: smaller λ values result in more outliers (Fig. S2b); moreover, lower λ values emphasize topological discordance, which leads to the removal of topological outliers (Fig. 3a). Note, however, that for true gene trees, outlier removal leads to an underestimation of discordance. When gene trees are estimated, discordance is overestimated (i.e., the quartet score of estimated gene trees versus true species trees is below the score of true gene trees versus species trees), and filtering reduces this overestimation.

Lower λ values tend to lead to a higher number of outliers across all levels of gene tree error (Fig. S2b). At lower λ, embeddings of species across gene trees become more concentrated as the effects of branch length variation diminish (Fig. 2b), allowing topological changes to dominate outlier detection. Additionally, there is a noticeable drop in the number of outliers when applying PhylteR to the original gene trees compared to trees with high λ values. To understand the cause, note that even at $\lambda = \infty$, TCMM scales the gene trees to a unified total branch length while preserving relative branch lengths; this can mask the effects of per-gene rate variation on embeddings, thereby reducing the number of outliers.

To summarize, using TCMM prior to PhylteR results in a fundamental shift in the type of outliers detected – those that create topological discordance as opposed to branch length discordance. As a result,

for estimated gene trees, where input data often exhibits extra discordance compared to true gene trees, there is a noticeable reduction in discordance with the species tree after outlier detection. These patterns are consistent with the theory that methods like PhylteR, if applied to original gene trees, may incorrectly label genuine cases of rate variation as outliers. Using TCMM before embedding will help PhylteR identify outliers that are topologically difficult to reconcile with other gene trees.

Biological Dataset

We next studied six biological datasets (mammals, frogs, plants, and two datasets on Xenacoelomorpha, Insects) adopted from Mai and Mirarab (2018). These datasets cover a range of number of species, number of genes, and evolutionary depth (Appendix A.1). Some patterns observed in the simulated datasets persist. As λ decreases, MCOA embeddings become more concentrated and closer to the median embedding for each species (Fig. 4a). Also, the similarity of outliers detected by TCMM+PhylteR to those identified by TreeShrink tends to decline with lower λ, albeit less distinctly than in the simulated datasets (Fig. 4c). Other patterns differ from the simulated datasets. The impact of λ on the number of outliers depends on the dataset, with some like mammals behaving like simulated data and others like insects and plants showing a reduction in the number of outliers after applying TCMM (Fig. 4b). The proportion of outlier species varies across datasets, with the mammals dataset showing the highest rate – 1.9% across all gene trees at $\lambda = 0$.

The ΔRF distance to the species tree also has different patterns across datasets (Fig. 4c). In some datasets (e..g, plants and mammals) $\Delta\text{RF} \gg 0$ indicates effective outlier detection. In others, such as frogs and the Xenacoelomorpha dataset of Rouse et al. (2016), for certain λ values, ΔRF is close to zero suggesting ineffective outlier removal. Irrespective of this pattern, we did not observe the uniform decrease in ΔRF with λ seen in the simulated data. One possible explanation is the lack of access totrue species trees for biological datasets; instead, the ASTRAL estimated species tree is used for all analyses, and the presence of outliers may influence this estimate. Another possibility is that outliers in these real data show themselves through a combination of unusual branch lengths and topological discordance.

Removing outliers identified by TCMM+PhylteR generally increases the similarity to the estimated species tree for most, but not all, datasets (Fig. 4d). In particular, for mammals and frogs, TCMM+PhylteR improves quartet scores more than using PhylteR alone with any λ, particularly with lower λ values. For plants and insects, some λ values are better than PhylteR alone while others are not, but all cases lead to improved quartet scores. In the case of the Xenacoelomorpha datasets Cannon et al. (2016), pairing TCMM with PhylteR can reduce improvements compared to PhylteR alone though it still results in improvements. The Xenacoelomorpha dataset of Rouse et al. (2016), however, was different; removing outliers makes the gene trees more dissimilar to the species tree with TCMM+PhylteR but not with PhylteR alone or high λ.

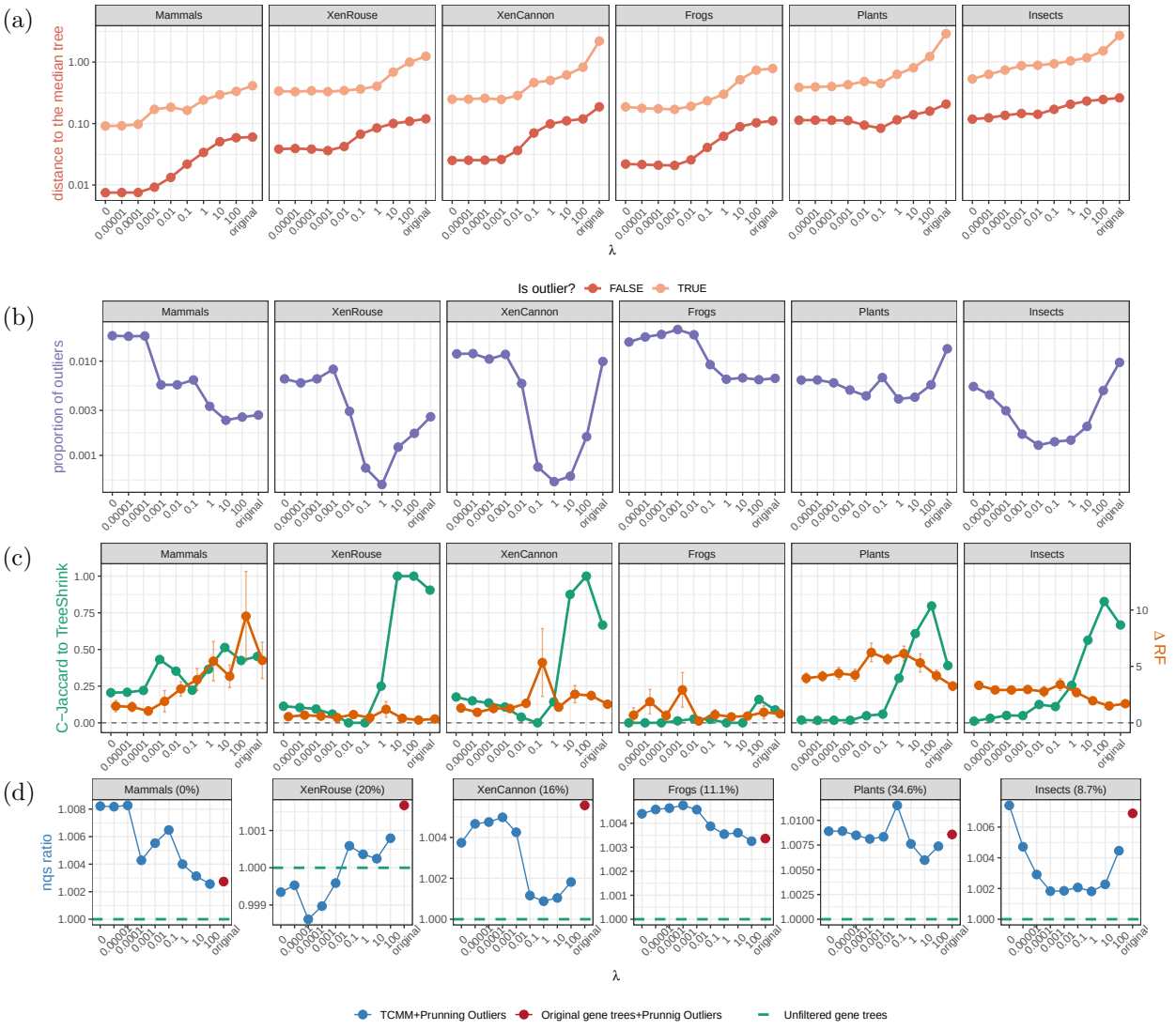

Figure 4: (Biological Dataset) (a) Comparing the euclidean distance to the median tree for outlier and non-outlier species. (b) Total number of outliers over total number of species for each λ. (c) Comparing similarity to TreeShrink outliers in Containment Jaccard (green) and ΔRF distance (orange) (Appendix A.1). Above 0 indicates better RF improvement than random. (d) Comparing the ratio of the normalized quartet score to the ASTRAL estimated species tree over the unfiltered original gene trees. The unfiltered original gene trees are shown in green. The amount of missing taxa from each dataset is written in percentage.

Application 2: Species tree branch lengths

Simulated Data

Choosing a version of TCMM

The per-gene approach with arithmetic mean exhibits lower bias compared to the consensus version of TCMM without compromising on error (Fig. S3a), making it the preferred choice. Another advantage of the per-gene approach is its ability to perform outlier detection and removal, effectively eliminating

extremely small estimated branch lengths that are often present in the consensus approach. In fact, the consensus method can assign branch lengths that are essentially zero, particularly for smaller λ values. However, these underestimations are mitigated in the per-gene method due to outlier removal (Fig. S3bc). Incorporating outlier removal also reduces bias without affecting overall error (Fig. S3a). While all three methods (consensus, per-gene, and per-gene with outlier removal) perform similarly when initial branch lengths are available, the per-gene method with outlier removal is adopted as the default version of TCMM due to its performance in the absence of initial branch lengths and the advantages of outlier removal in biological datasets. Furthermore, examining varying λ reveals that as λ values increase, branch lengths estimated by the default TCMM method become progressively more similar to the initial branch lengths (CASTLES-Pro) in terms of their relative values (Fig. S3bc).

Comparison to other methods

On the ILS-only dataset (S100), TCMM outperforms both the commonly used concatenation method (Concat+RAxML) and distance-based methods (ERaBLE, and AVG+FastME) in terms of overall mean absolute error and bias (Fig. S4). However, it does not match the accuracy of the coalescent-based CASTLES-Pro. Combining CASTLES-Pro with TCMM (using automatic λ) is substantially better than TCMM alone; yet, it does not outperform CASTLES-Pro. This result confirms that CASTLES-Pro, which is specifically designed and optimized for ILS, outperforms methods when gene tree discordance is dominated by ILS.

The patterns shift in simulations with HGT where the best method is combining TCMM and CASTLES-Pro with automatic λ selection (Fig. 5). The estimation error of all methods increases as the rate of HGT increases, especially for the two highest HGT rates. CASTLES-Pro remains better than TCMM alone and other methods. TCMM is better than methods other than CASTLES-pro for terminal branches and slightly worse for internal ones. All methods are biased, and these biases increase as the rate of HGT increases. CASTLES-Pro has an underestimation bias, while the other methods have an overestimation bias, which is concentrated on terminal branches; TCMM has the lowest overestimation bias among methods other than TCMM+CASTLES-Pro (e.g., on average 50.4% lower bias than concatenation).

On this dataset, however, TCMM+CASTLES-Pro has the best accuracy for all measures of error and bias. On average, the mean absolute error of CASTLES-Pro+TCMM is 8.2% lower than CASTLES-Pro and 27.3% lower than TCMM alone. The clearest advantage is the reduction in bias, particularly at higher HGT rates: While TCMM reduces bias by 26.7% compared to CASTLES-Pro across all HGT rates, the TCMM+CASTLES-Pro combination achieves reductions of 27.3% and 42.1% over TCMM alone at HGT rates of 2×10^{-7} and 5×10^{-7}, respectively. These results establish TCMM+CASTLES-Pro as a reliable choice for running TCMM across all tested datasets and model conditions (Fig. S5).

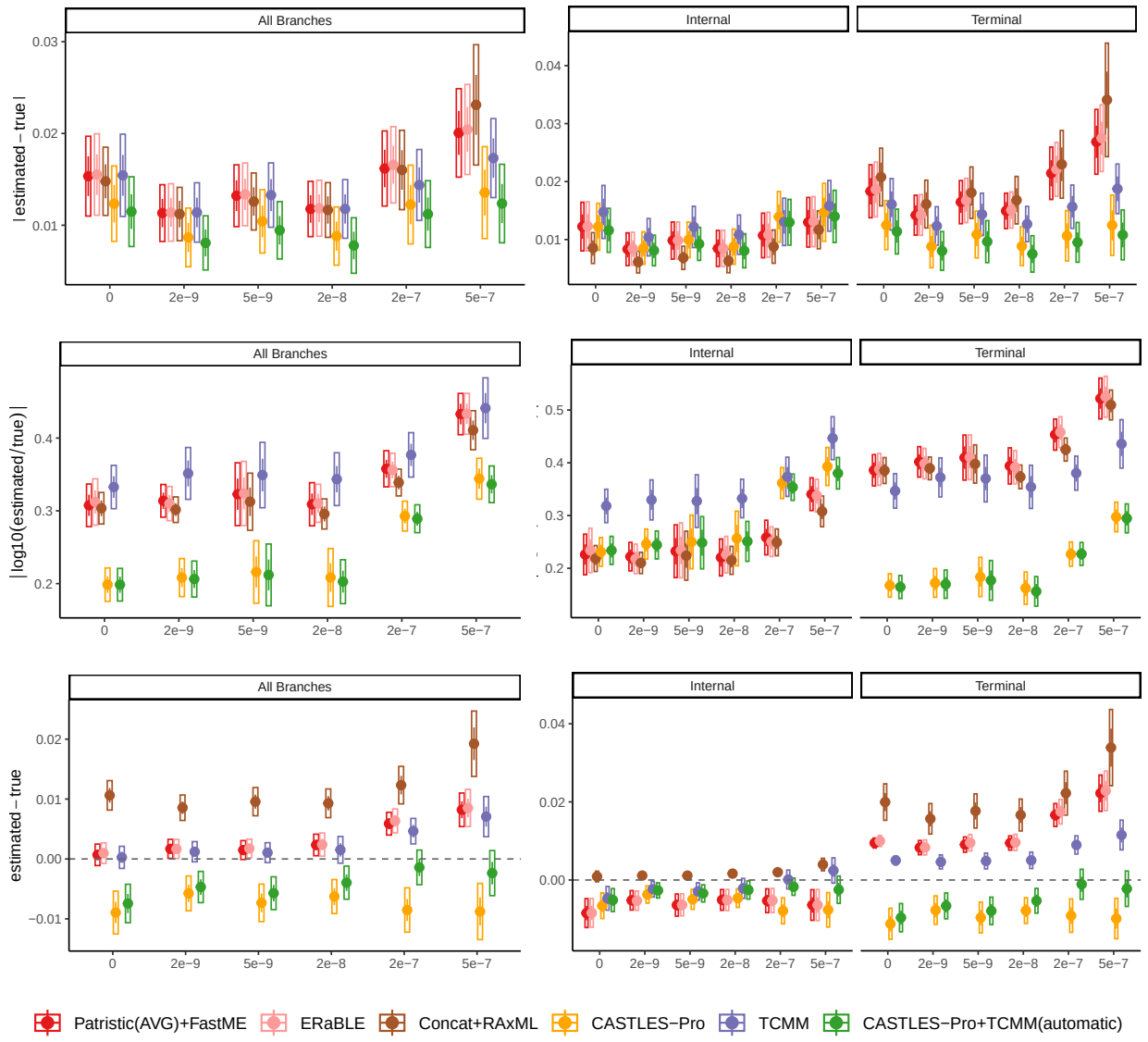

Figure 5: Comparing TCMM with the alternative methods on the HGT dataset with different rates of HGT (x-axis). TCMM is run with $\lambda = 0$ (TCMM) and automatic λ selection (CASTLES-Pro+TCMM). The methods are compared using two error metrics (mean absolute error and mean absolute log error) as well as bias.

No single λ consistently outperforms others across all settings (Fig. S5). This variability makes it useful to select a λ automatically, eliminating the need for manual tuning. Automatic λ selection tends to choose larger λ values (Fig. S6), likely due to TCMM's overestimation bias, which causes smaller λ values to result in overall higher estimated branch lengths. Note that even with high λ, CASTLES-Pro+TCMM benefits from outlier removal, a feature absent in CASTLES-Pro alone.

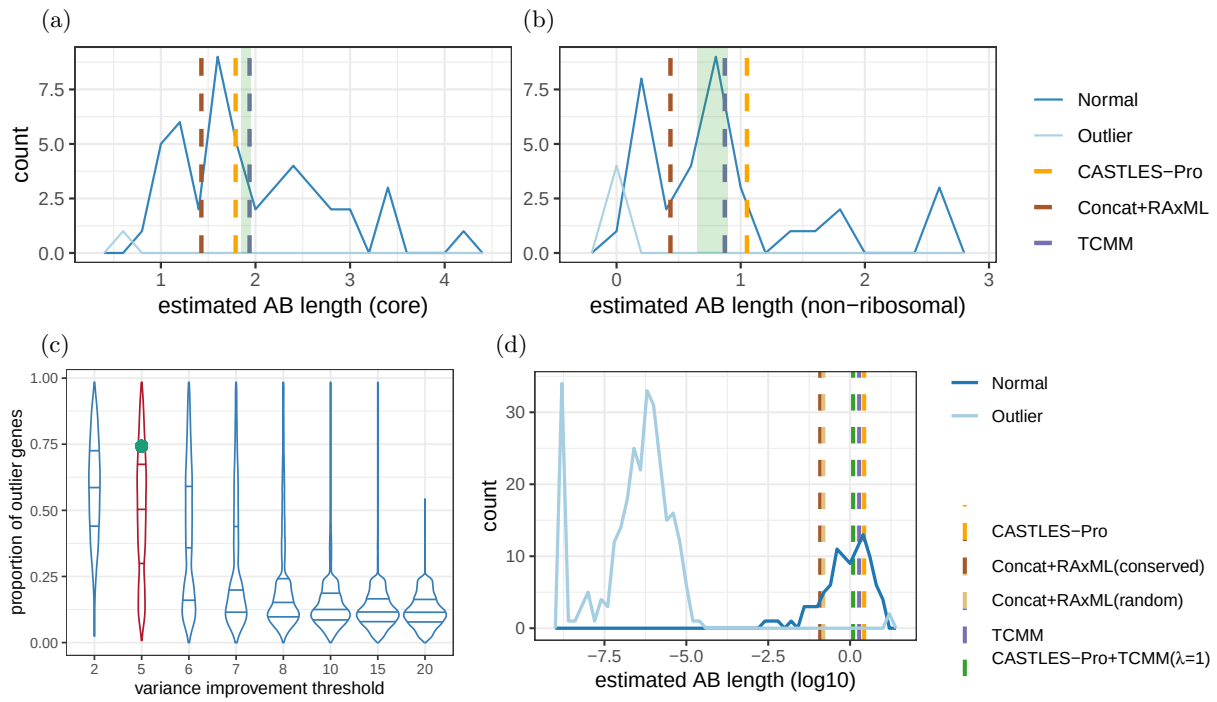

Figure 6: (a), (b) The distribution of the TCMM ($\lambda = 0$) estimated length for the branch separating bacteria and archaea per gene tree using the 49 core gene (a), the 38 non-ribosomal genes (b). Vertical dashed lines show the geometric mean estimate by TCMM excluding the outliers, estimates of CASTLES-Pro, and concatenation. The range of the estimated branch lengths for the AB branch by CASTLES-Pro+TCMM for different values of λ is shown in green. (c), (d) (Zhu et al. (2019a) Dataset) The distribution of the outlier genes removed by TCMM ($\lambda = 0$) per branch for different variance improvement thresholds (The default threshold (5) and the AB branch for this threshold are shown in red and green, respectively.) (c), and the distribution of the TCMM estimated length for the AB branch per gene tree (d).

Biological data: AB branch length

We analyzed several microbial datasets focusing on the separation between bacteria and archaea (AB branch). Moody et al. (2022) claimed that lower divergences obtained by some recent studies (Petitjean et al., 2015; Zhu et al., 2019b) were the result of highly abundant HGT in microbial datasets (Arnold et al., 2022).

Here, we reanalyzed a 72-taxon tree (Williams et al., 2020) with 49 core genes including ribosomal proteins and other conserved genes, and a 108-taxon dataset (Petitjean et al., 2015) with 38 genes that only includes non-ribosomal proteins. We also studied the microbial dataset by Zhu et al. (2019a) with 10,575 species sampled evenly from bacteria and archaea, and 381 genes. The species tree used in this analysis was inferred using ASTRAL and is rooted on the branch separating bacteria from archaea. Gene trees in this dataset lack 28% of taxa on average. On all datasets, we compare the length of the AB branch inferred using TCMM, CASTLES-Pro, and concatenation, all drawn on species topologies built using ASTRAL-III (Zhang et al., 2018) from the two gene sets. Here, ERaBLE fails to return an output, and AVG+FastME produces an estimated tree with several long negative branches around the root; thus, we exclude these methods.

Using 49 core genes (including ribosomal genes), CASTLES-Pro, TCMM, and CASTLES-Pro+TCMM (automatic λ selected as 1) estimate a length of 1.79, 1.94, and 1.91 SU for the AB branch, respectively, compared to 1.43 SU for concatenation. In contrast, on the 38 non-ribosomal genes (presumably with more HGT), the AB branch is much shorter, as noted by Moody et al. (2022). TCMM and CASTLES-Pro+TCMM (with λ selected as 100) estimate AB branch length to be 0.87 and 0.76, respectively, which are higher than concatenation (0.43) but lower than CASTLES-Pro (1.05). The observation that both gene tree-based methods yield higher branch lengths than concatenation for non-ribosomal genes is consistent with Moody et al. (2022)'s findings that concatenation can underestimate branch length in the presence of HGT. The substantial difference between ribosomal and non-ribosomal genes observed using methods that account for gene tree discordance confirms the fundamentally different patterns of evolution between the two. In per-gene mode, TCMM estimates a separate branch length for each gene tree (Fig. 6 a,b). An examination of the distribution reveals substantial diversity among genes, with some indicating an AB branch length of more than 4 SU, while others suggest a length below 0.5 SU.

On the microbial dataset by Zhu et al. (2019a), CASTLES-Pro, TCMM, and CASTLES-Pro+TCMM (λ fixed to 1 due to high running time) estimate branch lengths of 2.68, 1.89, and 1.24 SU, respectively, for the AB branch. As a result, both CASTLES-Pro and TCMM estimate drastically longer branch lengths than concatenation, which estimates 0.15 SU when selecting 100 random sites per gene and 0.12 SU when selecting the 100 most conserved sites. Once again, the distribution of estimated AB branch lengths across individual gene trees shows significant variation, ranging from as low as 10^{-9} to over 18 SU for some gene trees (Fig. 6d). In computing the final estimate, the outlier removal step of the per-gene method removes 283 out of 381 estimates before summarization (281 low outliers and 2 high outliers). This high level of outlier detection may seem excessive, a topic we will return to in the discussions. Across the WoL tree, many branches have large numbers of outliers removed, presumably due to rampant HGT. Compared to all branches in the tree, the AB branch is among those with the highest proportions of outliers (Fig. 6c).

Discussion

We introduced TCMM, a versatile framework for reconciling the branch lengths of a query tree to match one or more reference trees. At its core, TCMM provides a way to match tree metric spaces, thereby decreasing branch length discordance between trees without changing topological discordance. The regularization provides a way to adjust how far this adjustment goes. However, with or without regularization, the output branch lengths do not automatically have a biological meaning; instead, they are intended to serve as input in downstream applications. Our paper focused on two applications: outlier detection and species tree branch

length estimation. In both applications, combining TCMM with existing methods improved accuracy in many but not all conditions.

In the outlier detection application, we used TCMM to harmonize the metric space of gene trees so that the remaining outliers reflect topological differences. The main idea was that without such harmonization, embeddings of leaves in a tree might not align due to abnormally divergent rates of evolution in entire genes or some branches of certain genes, rather than errors in data. Using TCMM and λ, one can control whether such cases are marked as outliers. Beyond making topological outliers more visible, this type of harmonization can also be useful in other applications, such as detecting changes in rates of evolution.

The heterogeneity of rates of evolution across genes and branches is a key feature of evolution, impacting both applications we explored. Using TCMM to match a species tree to gene trees for SU branch length calculations (per-gene mode) provides a length *distribution* for each species tree branch. Allowing one rate per branch per gene effectively models across-genes heterotachy (Lopez et al., 2002), a feature missing from other methods. ERaBLE uses only one mutation rate per gene, making it far less versatile. CASTLES-Pro models rate heterogeneity across the species tree and overall rate changes across genes; however, it assumes that all genes have the same relative rates for species tree branches. Our biological results showed evidence (Fig. 6) of heterotachy. Even when outliers are excluded, the remaining loci still provide a range of lengths for the AB branch and other branches across the microbial tree of life, pointing to heterotachy.

The real power of TCMM for species tree branch length estimation lies in outlier detection. When input gene trees were generated using models assumed by CASTLES-Pro, we observed no advantage in pairing it with TCMM. In contrast, in the presence of HGT, which is not modeled by CASTLES-Pro, we observed substantially lower bias when CASTLES-Pro is followed by TCMM. The outlier detection step can eliminate gene trees with widely incongruent topologies, which, when matched to the species tree, would otherwise lead to extremely short branch lengths. We observed many such cases for the AB branch length in the microbial dataset.

Labeling the majority of genes (283 out of 381) as outliers, as our method does for the AB branch, may seem counterintuitive: How can more data be classified as outliers than not? This point gets at the heart of the debate about the AB branch. By removing enough horizontally transferred genes, we can estimate high values for the AB length, as Moody et al. (2022) argued. Concatenation (or even TCMM without outlier removal) estimates much shorter lengths given a large number of marker genes. Is removing many (or most) genes biologically justifiable? The answer depends on how we interpret the branch lengths and what we use them for. In one view, despite HGT, there is an underlying tree (vertically inherited), providing the backbone of the network representing the full evolutionary history. Then, to find the number of substitutions on a branch of the tree, we should remove horizontally transferred genes, which add not just

noise but an underestimation bias. Thus, we do not consider "outliers" merely as data points that deviate from the majority for unknown reasons. Rather, we know *a priori* that high topological discordance due to HGT results in short estimated lengths on the backbone species tree (near zero for TCMM). Thus, the primary role of outlier detection (responsible for removing 281 out of 283 outliers for the AB branch) is to eliminate these near-zero estimates, which are prevalent due to the high levels of HGT around the branch. In a different view, if horizontal evolution is pervasive and concentrated enough to render the backbone tree an inadequate model, one might instead aim to estimate an overall genome-wide level of divergence between two groups of taxa (e.g., archaea and bacteria). In such cases, removing large parts of the genome is less defensible. Regardless, by providing a distribution per branch, TCMM offers a way to better represent the full evolutionary history. A promising direction of future research is to investigate whether these distributions can be incorporated into downstream applications such as UniFrac calculation (Lozupone and Knight, 2005).

Our simulation results raise an important question: Should CASTLES-Pro be followed by TCMM for branch length estimation? The answer, based on our findings, is nuanced. If we believe the cause of discordance is mostly incomplete lineage sorting or gene duplication and loss, the answer seems to be no. Additionally, TCMM in its current form is not equipped to handle multi-copy gene trees—a potential avenue for future development. However, in scenarios involving single-copy gene trees with substantial levels of HGT or other aberrant sources of discordance (e.g., caused by hidden paralogy, alignment errors, or missing data), TCMM can enhance accuracy by detecting and removing outliers.

While we focused on two applications, the TCMM framework can have wide-ranging applications beyond those explored in this study. Once the species tree branch lengths are estimated, TCMM can also be used to match gene trees to the species tree (opposite of our approach here) for downstream applications such as visualization (e.g., using Bouckaert (2010)), estimation of average rates, and perhaps even dating individual gene trees, topics that can be explored in the future. The linear algebraic approach employed here effectively compares two trees by matching one to the other. Unlike many commonly used phylogenetic metrics, this matching incorporates both tree topology and branch lengths. Thus, the results of the matching could potentially be translated to new metrics of tree distance and new median tree problems. Future research should investigate these metrics and whether they provide additional insights beyond those offered by existing methods.

Acknowledgment

This work was supported by the National Institutes of Health (1R35GM142725) grant to S.M. This work used Expanse at San Diego Supercomputing Center through allocation ASC150046 from the Advanced

Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by U.S. National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296.

References

Arnold, B. J., Huang, I.-T., and Hanage, W. P. (2022). Horizontal gene transfer and adaptive evolution in bacteria. *Nature Reviews Microbiology*, 20(4):206–218.

Barthélemy, J.-P. and McMorris, F. R. (1986). The median procedure for n-trees. *Journal of Classification*, 3(2):329–334. Publisher: Springer.

Binet, M., Gascuel, O., Scornavacca, C., P. Douzery, E. J., and Pardi, F. (2016). Fast and accurate branch lengths estimation for phylogenomic trees. *BMC Bioinformatics*, 17(1):23.

Bogdanowicz, D. and Giaro, K. (2012). Matching split distance for unrooted binary phylogenetic trees. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.

Bouckaert, R. R. (2010). DensiTree: Making sense of sets of phylogenetic trees. *Bioinformatics*, 26(10):1372–1373. ISBN: 1367-4803.

Boussau, B., Szöllősi, G. J., and Duret, L. (2013). Genome-scale coestimation of species and gene trees. *Genome Research*, 23(2):323–330.

Cannon, J. T., Vellutini, B. C., Smith, J. r., Ronquist, F., Jondelius, U., and Hejnol, A. (2016). Xenacoelomorpha is the sister group to nephrozoa. *Nature*, 530(7588):89–93.

Comte, A., Tricou, T., Tannier, E., Joseph, J., Siberchicot, A., Penel, S., Allio, R., Delsuc, F., Dray, S., and de Vienne, D. M. (2023). PhylteR: Efficient Identification of Outlier Sequences in Phylogenomic Datasets. *Molecular Biology and Evolution*, 40(11):msad234.

Davidson, R., Vachaspati, P., Mirarab, S., and Warnow, T. (2015). Phylogenomic species tree estimation in the presence of incomplete lineage sorting and horizontal gene transfer. *BMC Genomics*, 16(Suppl 10):S1.

de Vienne, D. M., Ollier, S., and Aguileta, G. (2012). Phylo-mcoa: a fast and efficient method to detect outlier genes and species in phylogenomics using multiple co-inertia analysis. *Mol Biol Evol*, 29(6):1587–1598.

DeGiorgio, M. and Degnan, J. H. (2014). Robustness to Divergence Time Underestimation When Inferring Species Trees from Estimated Gene Trees. *Systematic Biology*, 63(1):66–82.

Degnan, J. H. and Rosenberg, N. A. (2009). Gene tree discordance, phylogenetic inference and the multispecies coalescent. *Trends in Ecology and Evolution*, 24(6):332–340. ISBN: 0169-5347 Publisher: Department of Human Genetics, University of Michigan, Ann Arbor, MI 48109, USA.

j.degnan@math.canterbury.ac.nz.

Desper, R. and Gascuel, O. (2003). Theoretical Foundation of the Balanced Minimum Evolution Method of Phylogenetic Inference and Its Relationship to Weighted Least-Squares Tree Fitting. *Molecular Biology and Evolution*, 21(3):587–598. ISBN: 0737-4038 (Print).

Doyon, J.-P. J., Ranwez, V., Daubin, V., and Berry, V. (2011). Models, algorithms and programs for phylogeny reconciliation. *Briefings in Bioinformatics*, 12(5):392–400.

Edwards, S. and Beerli, P. (2000). Perspective: gene divergence, population divergence, and the variance in coalescence time in phylogeographic studies. *Evolution*, 54(6):1839–1854.

Estabrook, G. F., McMorris, F. R., and Meacham, C. A. (1985). Comparison of Undirected Phylogenetic Trees Based on Subtrees of Four Evolutionary Units. *Systematic Biology*, 34(2):193–200.

Feng, Y.-J., Blackburn, D. C., Liang, D., Hillis, D. M., Wake, D. B., Cannatella, D. C., and Zhang, P. (2017). Phylogenomics reveals rapid, simultaneous diversification of three major clades of gondwanan frogs at the cretaceous–paleogene boundary. *Proceedings of the National Academy of Sciences*, 114(29):E5864–E5870.

Forthman, M., Braun, E. L., and Kimball, R. T. (2022). Gene tree quality affects empirical coalescent branch length estimation. *Zoologica Scripta*, 51(1):1–13.

Gatesy, J. and Springer, M. S. (2014). Phylogenetic Analysis at Deep Timescales: Unreliable Gene Trees, Bypassed Hidden Support, and the Coalescence/Concatalescence Conundrum. *Molecular phylogenetics and evolution*, 80:231–266. Publisher: Elsevier Inc.

Gori, K., Suchan, T., Alvarez, N., Goldman, N., and Dessimoz, C. (2016). Clustering Genes of Common Evolutionary History. *Molecular Biology and Evolution*. arXiv: 1510.02356 ISBN: 1537-1719 (Electronic)\r0737-4038 (Linking).

Heled, J. and Drummond, A. J. (2010). Bayesian inference of species trees from multilocus data. *Molecular Biology and Evolution*, 27(3):570–580.

Huang, H., He, Q., Kubatko, L. S., and Knowles, L. L. (2010). Sources of Error Inherent in Species-Tree Estimation: Impact of Mutational and Coalescent Effects on Accuracy and Implications for Choosing among Different Methods. *Systematic Biology*, 59(5):573–583. ISBN: 1076-836X (Electronic)\r1063-5157 (Linking).

Hubert, M. and Vandervieren, E. (2008). An adjusted boxplot for skewed distributions. *Computational Statistics and Data Analysis*, 52(12):5186–5201.

Kubatko, L. S., Carstens, B. C., and Knowles, L. L. (2009). STEM: species tree estimation using maximum likelihood for gene trees under coalescence. *Bioinformatics*, 25(7):971–973.

Kuhner, M. K. and Yamato, J. (2015). Practical performance of tree comparison metrics. *Systematic Biology*.

ISBN: 1063-5157.

Kumar, S. (2005). Molecular clocks: four decades of evolution. *Nature Reviews Genetics*, 6(8):654–662.

Laurin-Lemay, S., Brinkmann, H., and Philippe, H. (2012). Origin of land plants revisited in the light of sequence contamination and missing data. *Current Biology*. ISBN: 1879-0445 (Electronic)\r0960-9822 (Linking).

Lefort, V., Desper, R., and Gascuel, O. (2015). FastME 2.0: A comprehensive, accurate, and fast distance-based phylogeny inference program. *Molecular Biology and Evolution*, 32(10):2798–2800. ISBN: 1537-1719 (Electronic)\r0737-4038 (Linking).

Liu, L. (2008). BEST: Bayesian estimation of species trees under the coalescent model. *Bioinformatics*, 24(21):2542–2543. ISBN: 1367-4803 Publisher: Musuem of Comparative Zoology, Harvard University, Cambridge, MA, USA. lliu@oeb.harvard.edu.

Liu, L. and Yu, L. (2011). Estimating Species Trees from Unrooted Gene Trees. *Systematic Biology*, 60(5):661–667. ISBN: 1063-5157\r1076-836X.

Lopez, P., Casane, D., and Philippe, H. (2002). Heterotachy, an Important Process of Protein Evolution. *Molecular Biology and Evolution*, 19(1):1–7.

Lozupone, C. and Knight, R. (2005). UniFrac : a New Phylogenetic Method for Comparing Microbial Communities UniFrac : a New Phylogenetic Method for Comparing Microbial Communities. *Applied and environmental microbiology*, 71(12):8228–8235.

Maddison, W. P. (1997). Gene Trees in Species Trees. *Systematic Biology*, 46(3):523–536.

Mai, U. and Mirarab, S. (2017). TreeShrink: Efficient Detection of Outlier Tree Leaves. In Meidanis, J. and Nakhleh, L., editors, *Comparative Genomics. RECOMB-CG 2017. Lecture Notes in Computer Science, vol 10562*, pages 116–140. Springer International Publishing, Cham.

Mai, U. and Mirarab, S. (2018). TreeShrink: fast and accurate detection of outlier long branches in collections of phylogenetic trees. *BMC Genomics*, 19(S5):272. ISBN: 9783319679785.

Mai, U., Sayyari, E., and Mirarab, S. (2017). Minimum variance rooting of phylogenetic trees and implications for species tree reconstruction. *PLOS ONE*, 12(8):e0182238. Publisher: Public Library of Science ISBN: 1111111111.

Mallo, D., De Oliveira Martins, L., and Posada, D. (2016). SimPhy : Phylogenomic Simulation of Gene, Locus, and Species Trees. *Systematic Biology*, 65(2):334–344.

Mirarab, S., Bayzid, M. S., Boussau, B., and Warnow, T. (2014a). Statistical binning enables an accurate coalescent-based estimation of the avian tree. *Science*, 346(6215):1250463.

Mirarab, S., Bayzid, M. S., and Warnow, T. (2016). Evaluating Summary Methods for Multilocus Species

Tree Estimation in the Presence of Incomplete Lineage Sorting. *Systematic Biology*, 65(3):366–380. Publisher: Oxford University Press ISBN: 1063-5157.

Mirarab, S., Nakhleh, L., and Warnow, T. (2021). Multispecies Coalescent: Theory and Applications in Phylogenetics. *Annual Review of Ecology, Evolution, and Systematics*, 52(1):247–268.

Mirarab, S., Reaz, R., Bayzid, M. S., Zimmermann, T., Swenson, M. S., and Warnow, T. (2014b). ASTRAL: genome-scale coalescent-based species tree estimation. *Bioinformatics*, 30(17):i541–i548.

Misof, B., Liu, S., Meusemann, K., Peters, R. S., Donath, A., Mayer, C., Frandsen, P. B., Ware, J., Flouri, T., Beutel, R. G., Niehuis, O., Petersen, M., Izquierdo-Carrasco, F., Wappler, T., Rust, J., Aberer, A. J., Aspöck, U., Aspöck, H., Bartel, D., Blanke, A., Berger, S., Böhm, A., Buckley, T. R., Calcott, B., Chen, J., Friedrich, F., Fukui, M., Fujita, M., Greve, C., Grobe, P., Gu, S., Huang, Y., Jermiin, L. S., Kawahara, A. Y., Krogmann, L., Kubiak, M., Lanfear, R., Letsch, H., Li, Y., Li, Z., Li, J., Lu, H., Machida, R., Mashimo, Y., Kapli, P., McKenna, D. D., Meng, G., Nakagaki, Y., Navarrete-Heredia, J., Ott, M., Ou, Y., Pass, G., Podsiadlowski, L., Pohl, H., von Reumont, B. M., Schütte, K., Sekiya, K., Shimizu, S., Slipinski, A., Stamatakis, A., Song, W., Su, X., Szucsich, N. U., Tan, M., Tan, X., Tang, M., Tang, J., Timelthaler, G., Tomizuka, S., Trautwein, M., Tong, X., Uchifune, T., Walzl, M. G., Wiegmann, B. M., Wilbrandt, J., Wipfler, B., Wong, T. K. F., Wu, Q., Wu, G., Xie, Y., Yang, S., Yang, Q., Yeates, D. K., Yoshizawa, K., Zhang, Q., Zhang, R., Zhang, W., Zhang, Y., Zhao, J., Zhou, C., Zhou, L., Ziesmann, T., Zou, S., Li, Y., Xu, X., Zhang, Y., Yang, H., Wang, J., Wang, J., Kjer, K. M., and Zhou, X. (2014). Phylogenomics resolves the timing and pattern of insect evolution. *Science*, 346(6210):763–767.

Moody, E. R., Mahendrarajah, T. A., Dombrowski, N., Clark, J. W., Petitjean, C., Offre, P., Szöllősi, G. J., Spang, A., and Williams, T. A. (2022). An estimate of the deepest branches of the tree of life from ancient vertically evolving genes. *eLife*, 11:e66695.

Mossel, E. and Roch, S. (2010). Incomplete lineage sorting: consistent phylogeny estimation from multiple loci. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 7(1):166–171.

Ogilvie, H. A., Bouckaert, R. R., and Drummond, A. J. (2017). StarBEAST2 Brings Faster Species Tree Inference and Accurate Estimates of Substitution Rates. *Molecular Biology and Evolution*, 34(8):2101–2114. ISBN: 0737-4038.

Petitjean, C., Deschamps, P., López-García, P., and Moreira, D. (2015). Rooting the domain archaea by phylogenomic analysis supports the foundation of the new kingdom proteoarchaeota. *Genome biology and evolution*, 7(1):191–204.

Philippe, H., Brinkmann, H., Copley, R. R., Moroz, L. L., Nakano, H., Poustka, A. J., Wallberg, A., Peterson, K. J., and Telford, M. J. (2011). Acoelomorph flatworms are deuterostomes related to xenoturbella. *Nature*,

470(7333):255–258.

Philippe, H., Vienne, D. M. d., Ranwez, V., Roure, B., Baurain, D., and Delsuc, F. (2017). Pitfalls in supermatrix phylogenomics. *European Journal of Taxonomy*.

Price, M. N., Dehal, P. S., and Arkin, A. P. (2010). FastTree-2 – Approximately Maximum-Likelihood Trees for Large Alignments. *PLoS ONE*, 5(3):e9490. Publisher: Public Library of Science.

Rasmussen, M. D. and Kellis, M. (2007). Accurate gene-tree reconstruction by learning gene- and species-specific substitution rates across multiple complete genomes. *Genome Research*, 17(12):1932–1942. ISBN: 1088-9051.

Robinson, D. and Foulds, L. (1981). Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1-2):131–147.

Rouse, G. W., Wilson, N. G., Carvajal, J. I., and Vrijenhoek, R. C. (2016). New deep-sea species of xenoturbella and the position of xenacoelomorpha. *Nature*, 530(7588):94–97.

Rzhetsky, A. and Nei, M. (1993). Theoretical foundation of the minimum-evolution method of phylogenetic inference. *Molecular Biology and Evolution*, 10(5):1073–1095.

Salichos, L. and Rokas, A. (2013). Inferring ancient divergences requires genes with strong phylogenetic signals. *Nature*, 497(7449):327–31. Publisher: Nature Publishing Group ISBN: doi:10.1038/nature12130.

Sayyari, E. and Mirarab, S. (2016). Fast Coalescent-Based Computation of Local Branch Support from Quartet Frequencies. *Molecular Biology and Evolution*, 33(7):1654–1668.

Sayyari, E., Whitfield, J. B., and Mirarab, S. (2017). Fragmentary gene sequences negatively impact gene tree and species tree reconstruction. *Molecular Biology and Evolution*, 34(12):3279–3291.

Song, S., Liu, L., Edwards, S. V., and Wu, S. (2012). Resolving conflict in eutherian mammal phylogeny using phylogenomics and the multispecies coalescent model. *Proceedings of the National Academy of Sciences*, 109(37):14942–14947.

Springer, M. S. and Gatesy, J. (2018). On the importance of homology in the age of phylogenomics. *Systematics and Biodiversity*, 16(3):210–228.

Stamatakis, A. (2006). RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics*, 22(21):2688–2690.

Tabatabaee, Y., Zhang, C., Arasti, S., and Mirarab, S. (2025). Species tree branch length estimation despite incomplete lineage sorting, duplication, and loss. *bioRxiv*. 10.1101/2025.02.20.639320.

Tabatabaee, Y., Zhang, C., Warnow, T., and Mirarab, S. (2023). Phylogenomic branch length estimation using quartets. *Bioinformatics*, 39(Supplement_1):i185–i193.

Tria, F. D. K., Landan, G., and Dagan, T. (2017). Phylogenetic rooting using minimal ancestor deviation.

Nature Ecology & Evolution, 1(7):0193.

Vachaspati, P. and Warnow, T. (2015). ASTRID: Accurate Species TRees from Internode Distances. *BMC Genomics*, 16(Suppl 10):S3. Publisher: BioMed Central Ltd.

Wang, C. and Xiu, N. (2000). Convergence of the gradient projection method for generalized convex minimization. *Computational optimization and applications*, 16(2):111–120.

Wickett, N. J., Mirarab, S., Nguyen, N., Warnow, T., Carpenter, E., Matasci, N., Ayyampalayam, S., Barker, M. S., Burleigh, J. G., Gitzendanner, M. A., Ruhfel, B. R., Wafula, E., Der, J. P., Graham, S. W., Mathews, S., Melkonian, M., Soltis, D. E., Soltis, P. S., Miles, N. W., Rothfels, C. J., Pokorny, L., Shaw, A. J., DeGironimo, L., Stevenson, D. W., Surek, B., Villarreal, J. C., Roure, B., Philippe, H., dePamphilis, C. W., Chen, T., Deyholos, M. K., Baucom, R. S., Kutchan, T. M., Augustin, M. M., Wang, J., Zhang, Y., Tian, Z., Yan, Z., Wu, X., Sun, X., Wong, G. K.-S., and Leebens-Mack, J. (2014). Phylotranscriptomic analysis of the origin and early diversification of land plants. *Proceedings of the National Academy of Sciences*, 111(45):E4859–E4868.

Williams, T. A., Cox, C. J., Foster, P. G., Szöllősi, G. J., and Embley, T. M. (2020). Phylogenomics provides robust support for a two-domains tree of life. *Nature ecology & evolution*, 4(1):138–147.

Yang, J. and Warnow, T. (2011). Fast and accurate methods for phylogenomic analyses. *BMC Bioinformatics*, 12(Suppl 9):S4. Publisher: BioMed Central Ltd.

Zhang, C., Rabiee, M., Sayyari, E., and Mirarab, S. (2018). ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. *BMC Bioinformatics*, 19(S6):153.

Zhu, Q., Mai, U., Pfeiffer, W., Janssen, S., Asnicar, F., Sanders, J. G., Belda-Ferre, P., Al-Ghalith, G. A., Kopylova, E., McDonald, D., Kosciolek, T., Yin, J. B., Huang, S., Salam, N., Jiao, J.-Y., Wu, Z., Xu, Z. Z., Cantrell, K., Yang, Y., Sayyari, E., Rabiee, M., Morton, J. T., Podell, S., Knights, D., Li, W.-J., Huttenhower, C., Segata, N., Smarr, L., Mirarab, S., and Knight, R. (2019a). Phylogenomics of 10,575 genomes reveals evolutionary proximity between domains bacteria and archaea. *Nature Communications*, 10(1):5477.

Zhu, Q., Mai, U., Pfeiffer, W., Janssen, S., Asnicar, F., Sanders, J. G., Belda-Ferre, P., Al-Ghalith, G. A., Kopylova, E., McDonald, D., Kosciolek, T., Yin, J. B., Huang, S., Salam, N., Jiao, J.-Y., Wu, Z., Xu, Z. Z., Cantrell, K., Yang, Y., Sayyari, E., Rabiee, M., Morton, J. T., Podell1, S., Knights, D., Li1, W.-J., Huttenhower, C., Segata, N., Smarr, L., Mirarab, S., and Knight, R. (2019b). WoL: reference phylogeny for microbes (data pre-release).

Zimmermann, T., Mirarab, S., and Warnow, T. (2014). BBCA: Improving the scalability of *BEAST using random binning. *BMC genomics*, 15(Suppl 6):S11.

Supplementary material for

Branch Length Transforms using Optimal Tree Metric Matching

Shayesteh Arasti, Puoya Tabaghi, Yasamin Tabatabaee, Siavash Mirarab

Contents

A Supplementary text **1**

A.1 Datasets . . . 1

A.1.1 Biological datasets for application 1 . . . 1

A.2 Measurement metrics . . . 1

A.2.1 Definition of ΔRF . . . 1

A.2.2 nqs ratio . . . 2

A.2.3 Containment Jaccard . . . 2

A.3 Outlier removal for branch length estimation . . . 3

A.4 Missing data . . . 3

A.5 Supplementary remarks . . . 4

A.6 Proofs . . . 6

B Supplementary figures and tables **8**

A Supplementary text

A.1 Datasets

A.1.1 Biological datasets for application 1

Mammals: We studied a mammalian dataset by Song et al. (2012). This dataset contains 37 species (36 mammals and Chicken as outgroup) and 424 genes, where complete gene trees were inferred using RAxML in the re-analysis of the dataset by Mirarab et al. (2014a). Previous analyses (Gatesy and Springer, 2014; Mirarab et al., 2016) have identified several sources of error in this data, making it a good candidate for error detection.

Xen-Cannon and Xen-Rouse: We examined two datasets aimed at determining the phylogenetic position of Xenacoelomorpha, a phylum of bilaterally symmetrical marine worms, in the tree of life. The first dataset, referred to as XenCannon in our results, was studied by Cannon et al. (2016) and consists of 78 species and 213 genes. The second dataset, XenRouse, was studied by Rouse et al. (2016) and includes 26 species and 393 genes. The question of whether Xenacoelomorpha is the sister clade to the remaining Bilateria has been a long-standing debate (e.g., Philippe et al. (2011)), making these two datasets particularly significant for analysis.

Frogs: We analyzed a dataset published by Feng et al. (2017) that explores the evolutionary history and rate of evolution in frogs. This dataset includes 164 species, with 156 frog species and 8 outgroups, and is composed of 95 genes. For our analysis, we use the RAxML gene trees from this study.

Plants: We analyzed a plants and algae dataset published by Wickett et al. (2014), consisting of 104 species (including 4 outgroups) and 852 genes. Among the datasets used in our analysis, this one has one of the highest levels of missing taxa, with an average of 34.6% missing taxa.

Insects: We studied a phylotranscriptomic dataset of insects first published by Misof et al. (2014) to resolve the conflict around the evolutionary relationships between major insect orders. This datasets includes 144 species and 1478 genes. In our analysis, we use the RAxML estimated gene trees published by Sayyari et al. (2017).

A.2 Measurement metrics

A.2.1 Definition of ΔRF

Let S be a reference species tree. On simulated datasets, we use the true species tree as S, while on biological datasets, we use the species tree inferred by ASTRAL-III. For a gene tree G, let $\mathcal{O}$ denote the

outliers identified by a method. Let the $G \setminus \mathcal{O}$ denote a pruned version of tree G removing leaves of $\mathcal{O}$ and suppressing degree 1 nodes. We randomly remove the same number of species removed by a method from the gene tree ($|\mathcal{O}|$) in 10 replicates ($\mathcal{R}_1 \ldots \mathcal{R}_{10}$, $|\mathcal{R}_j| = |\mathcal{O}|$). The improvement in RF distance is defined as

$$\Delta\text{RF} = \frac{RF(G \setminus \mathcal{O}, S \setminus \mathcal{O}) - RF(G, S)}{1/10 \sum_j RF(G \setminus \mathcal{R}_j, S \setminus \mathcal{R}_j) - RF(G, S)} - 1 = \frac{RF(G \setminus \mathcal{O}, S \setminus \mathcal{O}) - 1/10 \sum_j RF(G \setminus \mathcal{R}_j, S \setminus \mathcal{R}_j)}{1/10 \sum_j RF(G \setminus \mathcal{R}_j, S \setminus \mathcal{R}_j) - RF(G, S)} \tag{5}$$

Defining before $= RF(G, S)$, after $= RF(G \setminus \mathcal{R}_j, S \setminus \mathcal{R}_j)$, random $= 1/10 \sum_j RF(G \setminus \mathcal{R}_j, S \setminus \mathcal{R}_j)$, we can write:

$$\Delta\text{RF} = \frac{\text{after} - \text{random}}{\text{random} - \text{before}} \tag{6}$$

Note that since we use RF distance, the denominator of ΔRF score is always positive. Thus, we measure how much more *topologically* similar gene trees become to the species tree due to outlier removal compared to random removal. A positive ΔRF score indicates better than random and negative ΔRF score indicates worse than random removal. The logic behind this is that true outliers introduced by error should render gene trees more dissimilar to the species tree than other species.

A.2.2 nqs ratio

We use the normalized quartet score (nqs) ratio to measure the improvement in the similarity between the gene trees and the species tree after pruning detected outliers. The nqs ratio is defined as the normalized quartet similarity of the gene trees to the species tree after pruning outliers, divided by the normalized similarity before outlier removal. An nqs ratio above 1 indicates an improvement in the topological similarity of the pruned gene trees to the species tree. For this calculation, we use the true species tree when available (simulated datasets) and the ASTRAL-III estimated species tree for biological datasets. We use ASTRAL-III to compute the normalized quartet score.

A.2.3 Containment Jaccard

To quantify whether outliers are the same as those found by the branch length-based method TreeShrink, we report the containment Jaccard similarity between the outliers detected by TCMM +PhylteR ($\mathcal{O}_{\text{TCMM+P}}$) and those detected by TreeShrink($\mathcal{O}_{\text{T}}$). Containment Jaccard index is defined as:

$$\text{ContainmentJaccard}(\mathcal{O}_{\text{TCMM+P}}, \mathcal{O}_{\text{T}}) = \frac{(\mathcal{O}_{\text{TCMM+P}} \cap \mathcal{O}_{\text{T}})}{|\mathcal{O}_{\text{TCMM+P}}|} \ .$$

A.3 Outlier removal for branch length estimation

There are two types of outliers that need to be considered and we deal with them separately since they have very different properties.

Small outliers. TCMM tends to assign exceedingly small values to *some* branches that are in the query but not the reference tree (as we will see), creating outliers. To detect and remove these outliers, we apply Jenks natural breaks to the log-transformed branch length estimates. The log transformation helps by converting the skewed branch length distribution into a more bimodal one, where the outliers form a distinct group of extremely small values easily separable by a clustering method such as Jenks natural breaks. We identify a single breakpoint and remove the data below the breakpoint only if splitting the data at this point reduces the variance by more than fivefold. The variance reduction is calculated for the two equal-sized sets of data points, $x1$ and $x2$, using the formula:

$$2\frac{\sigma^2(x_1\|x_2)}{(\sigma^2(x_1)+\sigma^2(x_2))}$$

where $x_1\|x_2$ denotes the concatenation of the two sets. Since the two resulting groups may differ in size, we balance them by randomly sampling from the larger group to match the size of the smaller one.

Large outliers. The per-gene method can also introduce extremely high values, which can arise from aberrantly long branches in the the gene trees or specific forms of discordance. These kinds of outliers are much less common than small values. As a result, they cannot be effectively separated using the same method as for lower outliers. To address these extreme high values, we use the simple boxplot method: we eliminate any data point exceeding $Q_3 + 3 \times IQR$, where Q_3 is the third quartile and $IQR = Q_3 - Q_1$ is the interquartile range. Note that this step follows the previous step, ensuring the lower values are removed first. This approach helps minimize the impact of outliers on the final branch length estimates, improving accuracy (as shown in the results).

A.4 Missing data

For a branch $e \in E_T$, let $p_T(e)$ represent the number of leaf pairs (u, v) where $u, v \in L_T$ such that $e \in P_T(u, v)$, and let $p^*(e)$ be the number of such leaf pairs that also exist in R; note $p^*(e) \leq p_T(e)$, with equality indicating no missing data. When $p^*(e) < p_T(e)$, the value of $(A_T^\top d_R)_e$ tends to be underestimated. To see this, consider a case where R and T are identical and note that $(A_T^\top d_R)_e$ gives the sum of patristic distances that include e on their path. Now if some leaves are removed in R, that sum necessarily decreases. We propose two

solutions to reduce this bias. The first approach simply scales $A_T^\top d_R$ by multiplying each $(A_T^\top d_R)_e$ with $p_T(e)/p^*(e)$ if $p^*(e) > 0$. This approach assumes that the pairs of taxa missing from R have similar mean patristic distances to those that are present. While we can construct cases where this assumption is violated (e.g., when outgroups are missing), this assumption seems reasonable empirically (See Fig. S1), and the method largely eliminates the underestimation bias.

The scaling method, however, fails when $p^*(e) = 0$, meaning that no pair around an edge is present in R. An alternative approach that works even when $p^*(e) = 0$ is imputation using neighbors. In this method, for each leaf $l \in L_T \setminus L_R$, the closest leaf $l' \in L_R$ is selected and l is added as a sibling of l' to R to form a cherry with both terminal branches set to zero.

We deal with the missing data slightly differently depending on whether consensus or per-gene modes are used. In the consensus version (including all cases of application 1), for each branch e, we apply the scaling heuristic mentioned earlier for gene trees that include at least one pair around e and ignore gene trees that do not have any; we assume that around every edge e, there is at least one pair that appears together in at least one gene tree.

To deal with missing data in the per-gene mode, we use a combination of scaling and placement heuristics. For any branch e in the species tree, if none of the pairs around it appear in a particular gene tree, we use the placement method to add it; otherwise, we apply the scaling heuristic to fix $A_T d$ values for that gene tree.

A.5 Supplementary remarks

Remark 1. *Problem 3 reduces to Problem 2.*

Proof. We can rewrite the objective function of Problem 3 as

$$\begin{aligned}\sum_{k\in[K]} \alpha_k^2 \|d_{R_k} - A_T w\|_2^2 + \lambda\sigma^2(w/w_T) &= w^\top A_T^\top A_T w\big(\sum_{k\in[K]} \alpha_k^2\big) - 2\big(\sum_{k\in[K]} \alpha_k^2 d_{R_k}\big)^\top A_T w + \lambda\sigma^2(w/w_T) + \text{const} \\ &= \text{const}''\|\sum_{k\in[K]} \frac{\alpha_k^2}{\sum_{n\in[K]}\alpha_n^2} d_{R_k} - A_T w\|_2^2 + \lambda\sigma^2(w/w_T) + \text{const}' \,,\end{aligned}$$

which equals Eq. (2) if we set

$$d = \sum_{k\in[K]} \frac{\alpha_k^2}{\sum_{n\in[K]}\alpha_n^2} d_{R_k} \,.$$

Thus, we simply need to compute the weighted average of reference distance vectors and solve the problem with this matrix as input. □

Remark 2 (Equation (4))**.** *Recall* $c = A_T^\top d_R \in \mathbb{R}^{|E_T|}$*. Then,*

$$c_e = \sum_{v \in C_T(e)} \sum_{v' \in C'_T(e)} d_R(v, v') \ . \tag{7}$$

Proof. By definition, the column of A_T corresponding to $e \in E_T$ has a 1 only for rows (v, v') where e is on the path $P_T(v, v')$. Therefore, the element of $c = A_T^\top d_R$ that corresponds to $e \in E_T$ is

$$c_e = \sum_{v, v' \in L_T, v < v'} d_R(v, v') 1_{e \in P_T(v, v')} \tag{8}$$

Next, note the fact that each path in the first equation is counted only if it includes e. Such paths will, by definition, have one leaf from each side of e. Thus,

$$\sum_{v, v' \in L_T, v < v'} d_R(v, v') 1_{e \in P_T(v, v')} = \sum_{v \in C_T(e)} \sum_{v' \in C'_T(e)} d_R(v, v')$$

□

Remark 3 (Equivalency of Consensus and Per-gene average)**.** *Consider the consensus Problem 3 with uniform weights ($\alpha_k = 1$ for all $k \in [K]$) and Problem 2 applied to each reference tree separately. If the optimal solutions (branch vectors) to all these problems are all strictly greater than ε, then the arithmetic average of solutions to Problem 2 is also the solution to the consensus Problem 3.*

Proof. Let us assume the optimal branch vector w^* are strictly greater than ε. Then, w^* satisfies the first-order optimality condition:

$$\nabla_w \frac{1}{2K} \sum_{k \in [K]} \|d_{R_k} - A_T w\|_2^2 + \lambda \sigma^2(w/w_T)\Big|_{w=w^*} = A_T^\top A_T w^* - A_T^\top \left(\frac{1}{K} \sum_{k \in [K]} d_{R_k} \right) + \lambda \frac{2}{|E_T|} P_1^\perp(w^*/w_T^2) = 0.$$

Let us now compute the optimal solution to the regularized version of Problem 2 between T and each R_k:

$$\forall k \in [K] : \nabla_w \frac{1}{2} \|d_{R_k} - A_T w\|_2^2 + \lambda \sigma^2(w/w_T)\Big|_{w=w_k^*} = A_T^\top A_T w_k^* - A_T^\top d_{R_k} + \lambda \frac{2}{|E_T|} P_1^\perp(w_k^*/w_T^2) = 0.$$

This holds true because we assumed that optimal branch vectors w_k^* for all references are strictly greater than ε. Averaging both sides of this equation across $k \in [K]$, we obtain

$$A_T^\top A_T \left(\frac{1}{K} \sum_{k \in [K]} w_k^* \right) - A_T^\top \left(\frac{1}{K} \sum_{k \in [K]} d_{R_k} \right) + \lambda \frac{2}{|E_T|} P_1^\perp \big((\frac{1}{K} \sum_{k \in [K]} w_k^*)/w_T^2 \big) = 0 \ ,$$

which matches the condition for the consensus problem, that is, $w^* = \frac{1}{K}\sum_{k\in[K]} w_k^*$. □

A.6 Proofs

Proof of Proposition 1. The gradient of the cost function in Eq. (3) with respect to the weights $w \in \mathbb{R}^{|E|}$ is:

$$\nabla_w l_1(w) = Bw + c,$$

where $B \in \mathbb{R}^{|E|\times|E|}$ and $c \in \mathbb{R}^{|E|}$. The Hessian $\nabla_w^2 l_1(w) = B$ is a positive semidefinite matrix; thus, $l_1(w)$ is a convex function everywhere in its domain. We can also compute the gradient of $l_2(w)$ as follows:

$$\nabla_w l_2(w) = \frac{2}{|E|}(P_1^{\perp} r)(1/w_T) = \frac{2}{|E|}P_1^{\perp}(w/w_T^2).$$

Furthermore, its Hessian $\nabla_w^2 l_2(w) = \frac{2}{|E|}\mathrm{diag}(1/w_T)P_1^{\perp}\mathrm{diag}(1/w_T)$ is a positive semidefinite matrix, making $l_2(w)$ a convex function of w as well. Since both $l_1(w)$ and $l_2(w)$ are convex functions of weight parameter w and cost functions in Problems 1, 2, and 3 are linear functions of l_1 and l_2 with *positive coefficients* (λ and α_k^2 terms), each cost function is also a convex function of w. In Problem 1, the constraint set $w \geq 0$ is nonempty, closed, and convex. In Problems 2 and 3, the constraint set $w \geq \varepsilon$ is also nonempty, closed, and convex. Therefore, all three problems involve convex function minimizations over nonempty, closed, and convex sets. Since convex functions over nonempty, bounded, convex sets have a minimum, the projected gradient descent method converges to the global minimum of the cost functions (Wang and Xiu, 2000). □

Algorithm 1 Computing $A_T^\top A_T$ and $A_T^\top d_R$

procedure COMPUTE_$A^\top A$(Tree T)
  `pre-compute` $|C_T|$ `and` $|C'_T|$ `for each edge of` T `in a bottom-up traversal`
  for e `in` E_T **do**
    for e' `in` E_T **do**
      if $e == e'$ **then**
        $A^\top A[e, e'] \leftarrow |C_T(e)| \cdot |C'_T(e)|$
      else
        if e `is a descendant of` e' **then**
          $A^\top A[e, e'] \leftarrow |C_T(e)| \cdot |C'_T(e')|$
        else if e' `is a descendant of` e **then**
          $A^\top A[e, e'] \leftarrow |C_T(e')| \cdot |C'_T(e)|$
        else
          $A^\top A[e, e'] \leftarrow |C_T(e)| \cdot |C_T(e')|$
  return $A^\top A$
procedure COMPUTE_$A^\top d$(Tree T, Tree R)
  for $e \in E_T$ **do**
    $A^\top d[e] \leftarrow 0$
    $pairs \leftarrow 0$
    for $n \in$ bottom-up traversal of V_R **do**
      $e' \leftarrow$ the edge above n
      if n `is leaf` **then**
        $C_b(n), C_r(n) = 1_{n \in C_T(e)}, 1_{n \notin C_T(e)}$ ▷ Number of blue and red leaves below n
        $L_b(n), L_r(n) = w_{e'} \cdot C_b(n), w_{e'} \cdot C_r(n)$ ▷ Sum of the distances of blue and red leaves up to the parent of n
      else
        $c_l, c_r \leftarrow$ children of n
        $pairs \mathrel{+}= C_b(c_l) \cdot C_r(c_r) + C_r(c_l) \cdot C_b(c_r)$
        $A^\top d[e] \mathrel{+}= C_b(c_l) \cdot L_r(c_r) + C_r(c_l) \cdot L_b(c_r) + C_b(c_r) \cdot L_r(c_l) + C_r(c_r) \cdot L_b(c_l)$
        $C_b(n), C_r(n) = C_b(c_l) + C_b(c_r), C_r(c_l) + C_r(c_r)$
        $L_b(n), L_r(n) = L_b(c_l) + L_b(c_r) + w_{e'} \cdot C_b(n), L_r(c_l) + L_r(c_r) + w_{e'} \cdot C_r(n)$
    $A^\top d[e] = \frac{|C_T(e)| \cdot |C'_T(e)|}{pairs} \cdot A^\top d[e]$
  return $A^\top d$

B Supplementary figures and tables

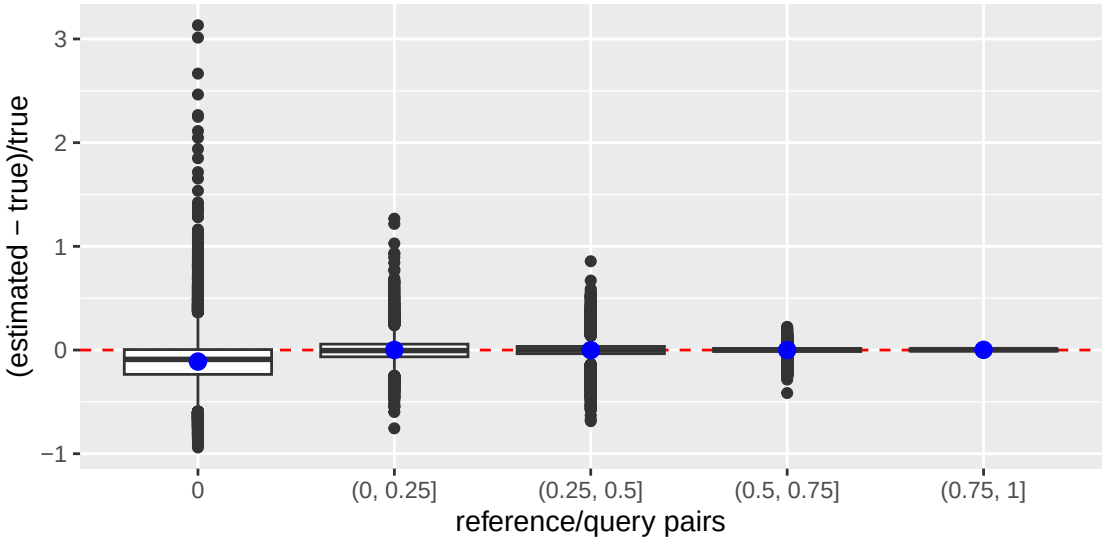

Figure S1: (S100 Dataset) Comparing estimated values for $A_T^\top d_R$ to the true values. Missing taxa are introduced into the true gene trees by removing m_g species from gene tree g, where $m_g \sim \text{Poisson}(\lambda = 0.34n)$ and n is the total number of leaves in the complete gene trees. The scaling method is used to estimate $(A_T^\top d_R)_e$ where $p_*(e) > 0$ for a branch e and the imputation method is used otherwise. The true values are computed using complete true gene trees. The figure shows results for the first gene tree as the query and the first 10 gene trees as the reference trees (pruned trees for estimated values and complete trees for true values).

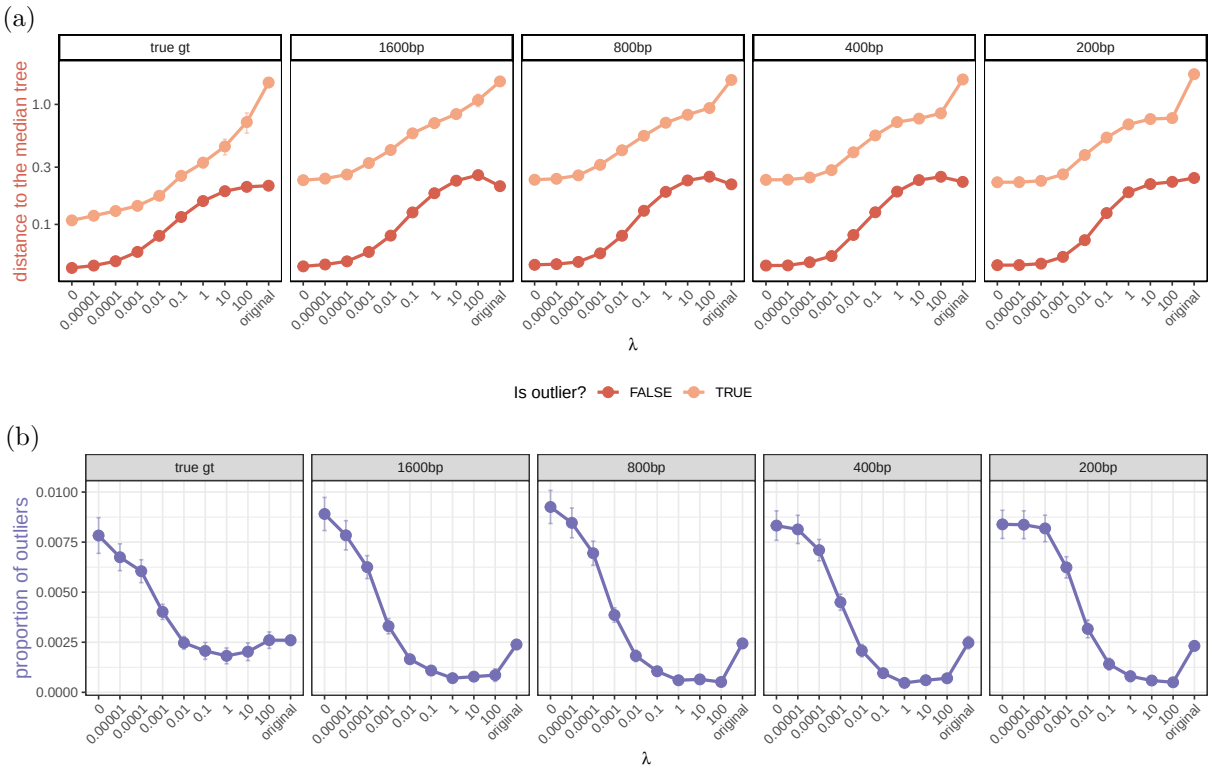

Figure S2: (S100 Dataset). (a) Comparing the euclidean distance to the median tree for all levels of gene tree error. (b) Total number of outliers over total number of species per replicate, averaged over all replicates. Mean and standard error are shown for each λ.

Figure S3: (HGT Dataset, HGT rate = 2e-7) (a) Comparison of different versions of TCMM in terms of mean absolute error (top) and bias (bottom), showing mean (dot) and 0.95% normal CI (box). Summarization methods are arithmetic mean (*mean*) or geometric mean (*geo mean*). The regularization coefficient (λ) was set to zero, except for the CASTLES-Pro+per-gene and CASTLES-Pro+consensus method, where λ is selected automatically. (b) The distribution of the estimated branch lengths of two versions of TCMM divided by the CASTLES-Pro branch lengths. (c) The effect of the regularization coefficient (λ) on the estimated branch lengths. TCMM output length per gene (before summarization) over the input length (CASTLES-Pro) is shown; the variance of this ratio (spread of distribution) indicates the change in the relative length of branches between input and output of TCMM. Outliers are identified by our per-gene method.

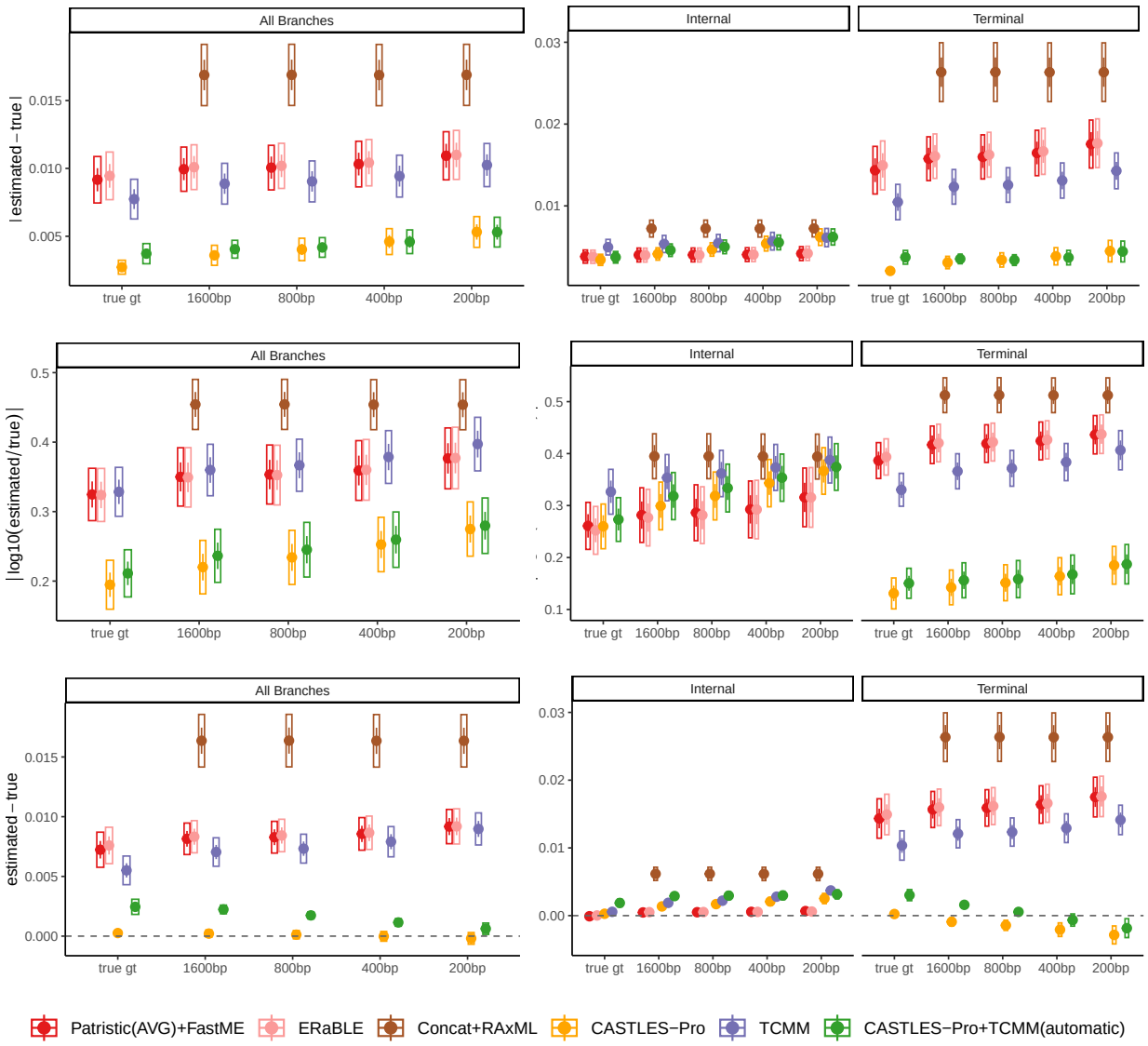

Figure S4: Comparing TCMM with the alternative methods on the S100 dataset with different levels of gene tree error (x-axis). TCMM is run with $\lambda = 0$ (TCMM) and automatic λ selection (CASTLES-Pro+TCMM). The methods are compared using two error metrics (mean absolute error and mean absolute log error) as well as bias.

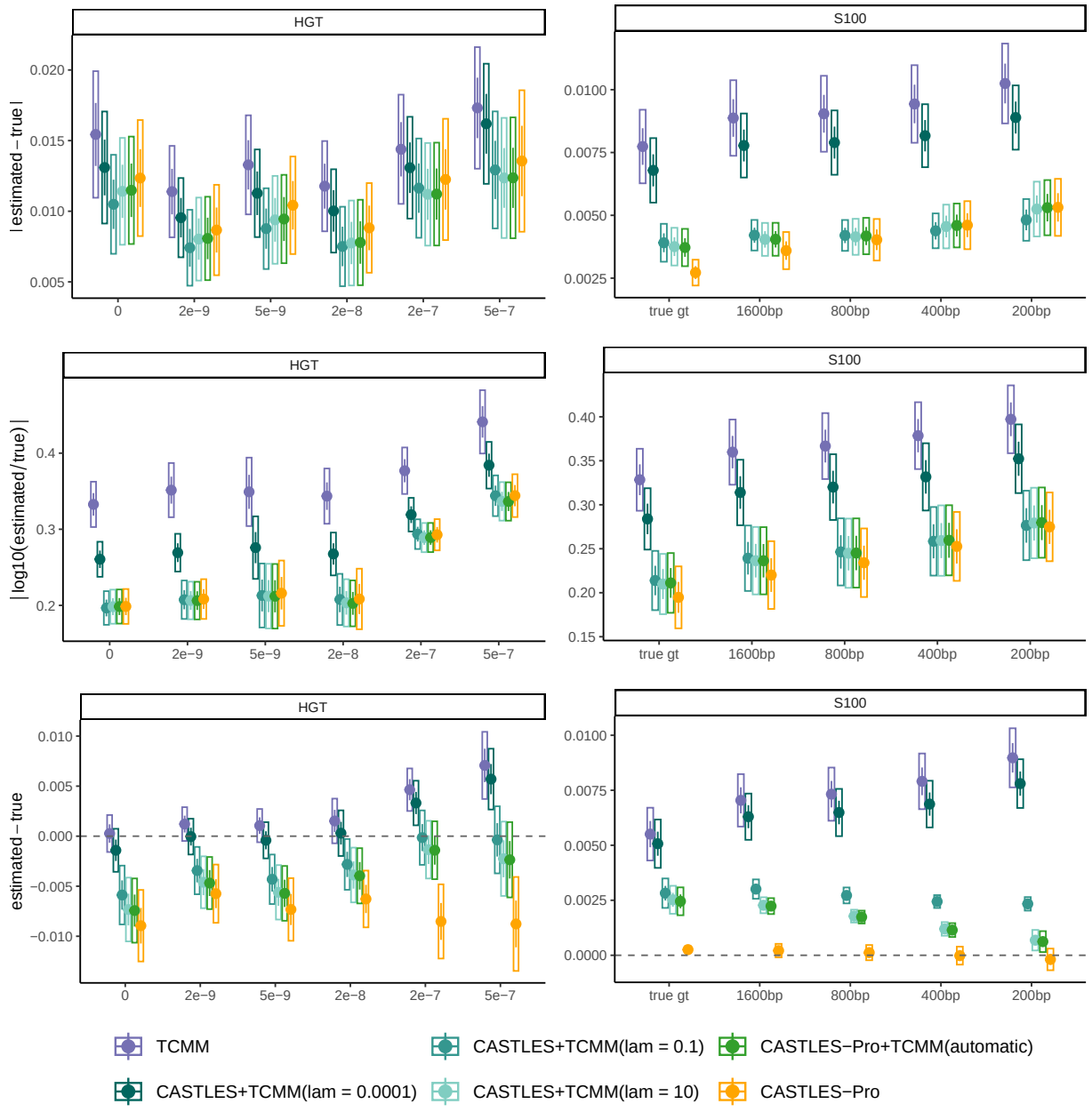

Figure S5: Comparing different λ values for CASLTES-Pro+TCMM (including automatic λ selection) with CASTLES-Pro and TCMM alone on the HGT dataset (left) and S100 dataset (right). The methods are compared using two error metrics (mean absolute error and mean absolute log error) as well as bias.

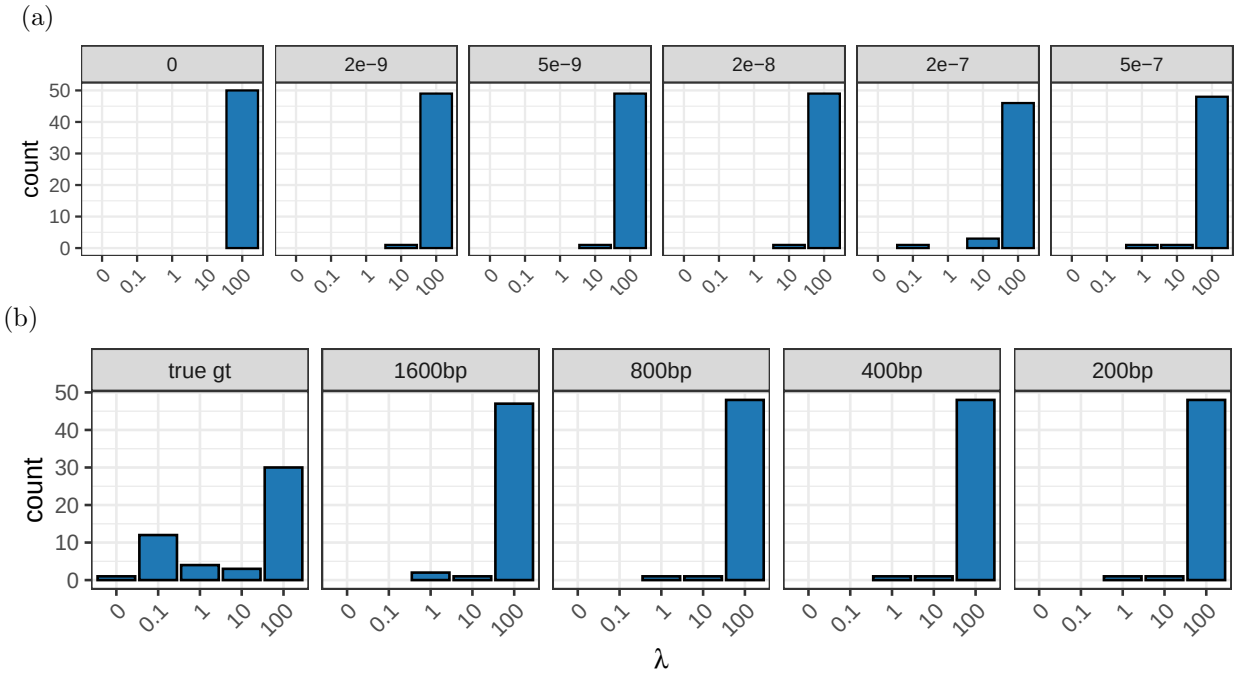

Figure S6: Histogram of the λ chosen by automatic λ selection process of TCMM for different model conditions for the HGT dataset (a) and S100 dataset (b).