Cophenetic Median Trees

Alexey Markin and Oliver Eulenstein

Abstract—Median tree inference under path-difference metrics has shown great promise for large-scale phylogeny estimation. Similar to these metrics is the family of cophenetic metrics that originates from a classic dendrogram comparison method introduced more than 50 years ago. Despite the appeal of this family of metrics, the problem of computing median trees under cophenetic metrics has not been analyzed. Like other standard median tree problems relevant in practice, as we show here, this problem is also NP-hard. NP-hard median tree problems have been successfully addressed by local search heuristics that are solving thousands of instances of a corresponding (local neighborhood) search problem. For the local neighborhood search problem under a cophenetic metric, the best known (naïve) algorithm has a time complexity that is typically prohibitive for effective heuristic searches. Building on the pioneering work on path-difference median trees we develop efficient algorithms for Manhattan and Euclidean cophenetic search problems that improve on the naïve solution by a linear and a quadratic factor, respectively. We demonstrate the performance and effectiveness of the resulting heuristic methods in a comparative study using benchmark empirical datasets.

Index Terms—Phylogenetic trees, median trees, supertrees, cophenetic distance, path-difference, gene tree parsimony

1 Introduction

Reconstruction of the evolutionary history, commonly modeled as a phylogenetic tree, is one of the central problems in biology. Solutions to this problem have direct applications in almost every discipline of natural sciences including medicine, epidemiology, biochemistry, agronomy, environmental sciences, and linguistics [1], [2], [3], [4], [5]. While the paramount goal of evolutionary biology is the representation of the true phylogenetic tree for millions of existing species, today the computation of phylogenies even for hundreds of species is a greatly challenging problem.

A well-established framework for large-scale phylogenetic tree inference comprises the construction of a species tree, called a *supertree*, from a collection of smaller trees each covering a subset of the species in question. Such collections of trees appear naturally either as so-called *gene trees* – the trees describing the evolution of a gene shared by species [6], [7]; or as trees collected from various sources in an attempt to unify the previously inferred knowledge and present a single phylogenetic tree with high confidence in it [8].

Supertree construction is a challenging problem on its own since in practice the input trees often appear to be in disagreement, meaning that they suggest discordant evolutionary histories [8]. A classic approach for addressing this problem entails the search for a supertree that *fits* all the input trees as best as possible. The goodness of fit is mathematically assessed by a *cost function* that computes dissimilarity between any two phylogenetic trees. The resulting supertree search problem is called the *median tree problem* under the respective cost function. The problem has been extensively studied both from the theoretical and applications perspectives [8]. While all studied median tree problems of interest are inherently hard, various heuristics estimating median trees, such as the widely-used matrix representation with parsimony (MRP), hold enormous value for the biological community and have become an essential tool in practice [9], [10], [11], [12], [13].

The recent surge of interest in median trees under one of the oldest and extensively studied metrics in comparative phylogenetics, the path-difference distance [14], [15], opened a promising path for studies of median trees under other vector-based objectives. Perhaps, the most well-established one of them is the *cophenetic distance* [16]. The cophenetic distance was established based on one of the most popular dendrogram (which can be seen to be equivalent to a bijectively labeled weighted phylogenetic tree) comparison methods introduced by Sokal and Rohlf more than 50 years ago [17].

Similarly to the path-difference distance that uses the encoding of trees as path-length vectors, the cophenetic distance uses a vector encoding called a *cophenetic vector*. A cophenetic vector contains information about a distance from the least common ancestor (LCA) of two taxa to the root of the tree for *each* pair of taxa in a given rooted tree. A cophenetic vector *equivalently* encodes a phylogenetic tree [16]; hence one can measure a distance between two trees in the cophenetic vector space. Such distance can be formulated in terms of common vector norms, such as the Manhattan norm (also known as the Taxicab norm) and the Euclidean norm; the corresponding tree metrics are called the *Manhattan cophenetic distance* and the *Euclidean cophenetic distance* respectively. In contrast to other popular comparative metrics, such as the Robinson-Foulds metric [18], the cophenetic metrics can be similarly defined for weighted phylogenetic trees as well, providing an additional prospective advantage.

While both the cophenetic metrics and the path-difference metrics use vector distances, apart from that the two distance families do not bare much similarity. Primarily, the cophenetic distance relies on the LCA-mappings which brings it semantically closer to the classic model-based *deep coalescence* cost function [19] and the related

• *A. Markin and O. Eulenstein are with the Department of Computer Science, Iowa State University, Ames, IA, 50011.*
E-mail: {amarkin, oeulenst}@iastate.edu

Manuscript received ?; revised ?

duplications with losses cost function [20]. Indeed, in this work we experimentally establish that the cophenetic metrics are more closely correlated with the classic model-based deep coalescence cost function than the path-difference metrics. Further, in contrast to the path-difference metrics, which are only insignificantly perturbed by the rootings of the trees under consideration, the cophenetic metrics rely heavily on the rootings, and therefore, are better suited for the inference of rooted phylogenies.

As most of the other median tree problems are inherently complex, unsurprisingly, here we show that the cophenetic median tree problem is NP-hard. In fact, even the commonly practiced local search estimation approach implemented naïvely turns out to be computationally heavy and already infeasible for an instance of the cophenetic median tree problem involving less than a hundred of species. In our work, we move the feasibility bound for the local search heuristic much farther by designing an efficient local neighborhood search algorithms. The algorithms result from (i) an extensive analysis of properties of the cophenetic vectors and (ii) the adapted preprocessing part of the recently developed algorithmic frameworks targeting the path-difference median tree heuristics. With the efficient heuristics at hand, we were able to compute the first large-scale cophenetic median tree estimates and evaluate them against supertrees constructed by other standard supertree methods on benchmark real-world datasets. The software implementing the developed cophenetic local search heuristics is freely available from the web-page http://genome.cs.iastate.edu/ComBio/software.htm.

Related work. The family of cophenetic metrics has been pioneered and extensively studied by Cardona et.al. [16]. Their work, on one hand, addresses the minimum and maximum values as well as distributions of the cophenetic metrics; on the other hand, compares the cophenetic metrics to other popular comparative metrics, such as path-difference distances and the Robinson-Foulds metric. The best known (naïve) algorithm for computing the cophenetic distance between any two phylogenetic trees of size n requires $\Theta(n^2)$ time.

There has been a large body of work focusing on the biological, mathematical, and algorithmic properties of median trees adopting various definitions of distance measures that have been effectively used in comparative phylogenetics [8]. As most of the studied median tree problems are NP-hard, the classic median tree estimation algorithms, including MRP, effectively employ the local search (hill-climbing) heuristics [21], [22], [23], [24], [25], which have provided credible estimates of large-scale species trees [21], [22]. Local search heuristics typically operate in a search space of all existing supertrees (candidate median trees) for the given collection of input trees. The search starts with a supertree called a *seed* and it maintains the candidate median tree updating it on each iteration. An iteration encompasses a search of a tree with the minimum distance to the input trees in the neighborhood of the current candidate tree. The neighborhood is typically defined in terms of a tree edit operation of choice. One of the most popular such operations is called *subtree prune and regraft (SPR)*, whose respective neighborhood contains $\Theta(n^2)$ trees, where n is the size of a median tree. Consequently, the SPR-neighborhood search problem under any cophenetic metric can be solved naïvely in $\Theta(kn^4)$ time, where k is the number of input trees.

The effectiveness of standard local search heuristics is typically highly dependent on the choice of the starting tree. Traditionally, greedy heuristics are employed to efficiently construct well-fitting starting trees; that is, trees that consistently have a significantly smaller distance to the input trees than a randomly chosen supertree. An extension of this approach, a *hybrid heuristic*, was introduced as a method to improve the effectiveness of the traditional approach by applying the local search heuristic on multiple stages of constructing the starting tree itself [15], [26].

The hybrid heuristic was introduced as a part of the recent work on path-difference median trees, where two efficient local search heuristic algorithms were developed (see [15] for the Manhattan distance study and [14] for the Euclidean distance study).

Our contribution. The focus of this work is the development and the applicability study of effective heuristics estimating cophenetic median trees. The design of a heuristic is a necessary component, as we show that the *cophenetic median tree problem* under any vector norm is NP-hard.

Design of efficient local search algorithms involves a structured analysis of how a cophenetic vector is altered by an SPR edit operation. After presenting such analysis, we demonstrate that the groundwork from the related Manhattan [15] and Euclidean [14] path-difference median tree studies can be adapted to develop the respective Manhattan and Euclidean *cophenetic* local search algorithms. As a result of applying involved dynamic programming solutions, we obtained a $\Theta(kn^2)$ Euclidean local neighborhood search algorithm and a $\Theta(kn^3)$ Manhattan local neighborhood search algorithms; where k is the number of input trees and n is the overall number of taxa. These two algorithms improve on the naïve solution by the factors of n^2 and n respectively, which enabled the local search median tree estimation approach to become significantly more scalable, as demonstrated in our first experimental study.

Further, in the applicability study, we employ the state-of-the-art hybrid heuristic powered by the developed local neighborhood search algorithms to evaluate cophenetic median trees on three published empirical datasets. Following the well-established approach, we compare the median trees constructed by the cophenetic hybrid heuristics to the supertrees and median trees constructed by other benchmark methods (including MRP, path-difference heuristics, and others). The comparison of supertrees is performed in terms of the distances to the respective input trees (goodness of fit) via various distance measures. Focusing on the cophenetic metric, we present a distribution analysis study, where we map the supertree cophenetic distances onto the estimated distributions; such mapping allows us to better analyze the significance of the developed heuristics and compare them to other methods.

Finally, motivated by the previous work and our results, we study the correlation of the Manhattan and Euclidean cophenetic metrics to the classic model-based cost functions including deep coalescence, duplications, and duplications with losses.

2 Basics and Preliminaries

Basic definitions. Throughout this paper we adhere to the definitions and notation introduced in [26]. A *(phylogenetic) tree* T is a rooted binary tree, where each leaf is uniquely labeled with a taxon, each internal node v has exactly two children nodes, denoted by $\mathsf{Ch}_T(v)$, and each node u except for the root has a single parent node denoted by $\mathsf{Pa}_T(u)$. In addition, we denote the node set, edge set and leaf set of T by $V(T)$, $E(T)$, and $\mathsf{L}(T)$ respectively. We denote the root by $\mathsf{Rt}(T)$ and a sibling of each non-root node u by $\mathsf{Sb}(u)$. We also set $T(v)$ to be a subtree of T rooted at $v \in V(T)$, and $\overline{T(v)}$ to be a tree obtained by pruning $T(v)$ from T. Occasionally we use the standard nested parenthesis notation to represent small trees.

We define a partial order $\preceq_T$ on the vertex set $V(T)$, such that $u \preceq v$, if v is a node on the path from u to $\mathsf{Rt}(T)$. Additionally, we say $u \prec v$, if $u \preceq v$ and $u \neq v$. The *least common ancestor (LCA)* of two nodes $u, v \in V(T)$, $\mathsf{LCA}_T(u, v)$, is the furthest from the root node, w, such that $v \preceq w$ and $u \preceq w$.

A set of leaves $\mathsf{L}(T(v))$ is called a *cluster* of the node v, and is denoted by C_v. Note that for convenience we identify the leaves in a phylogenetic tree with the respective labels (taxa).

Let $L \subseteq \mathsf{L}(T)$ and T' be the minimal subtree of T with leaf set L. We define the *leaf-induced subtree* $T[L]$ of T to be the tree obtained from T' by successively removing each node of degree two (except for the root) and adjoining its two neighbors (a parent and a child).

Let $\mathcal{P}$ be a set of phylogenetic trees $\{G_1, \ldots, G_k\}$. We extend the definition of a leaf set to a set of trees as follows: $\mathsf{L}(\mathcal{P}) := \cup_{i=1}^{k} \mathsf{L}(G_i)$. A tree S is called a *supertree* of $\mathcal{P}$, if $\mathsf{L}(S) = \mathsf{L}(\mathcal{P})$. A set of trees $\mathcal{P}$ is called *compatible* if there exist a supertree T consistent with every tree in $\mathcal{P}$, and a tree T is *consistent* with a tree G if $T[\mathsf{L}(G)] \equiv G$ up to isomorphism of rooted semi-labeled trees [27].

Cophenetic distance. In this part we follow the definitions presented in [16]. Given a phylogenetic tree T, let the *cophenetic value* of $u, v \in V(T)$, denoted by $\phi_{u,v}(T)$, be the length (measured in the number of edges) of the path from $\mathsf{LCA}_T(u, v)$ to $\mathsf{Rt}(T)$. Additionally, $\delta_u := \phi_{u,u}(T)$ is the *depth* of the node u. Given that, the *cophenetic vector* of T is

$$\phi(T) := (\phi_{i,j}(T))_{1 \leq i \leq j \leq |\mathsf{L}(T)|},$$

for some fixed ordering of leaves in T. The *cophenetic distance* between two trees G and S over the same leaf set is defined as

$$\mathsf{d}_{\phi,p}(G, S) := ||\phi(G) - \phi(S)||_p,$$

where $||\cdot||_p$ denotes an L_p norm of a vector for some fixed $p \geq 1$.

Next, let $\Phi(G, S)$ be the *cophenetic difference matrix* of size $|\mathsf{L}(G)| \times |\mathsf{L}(G)|$, such that for all $i, j \in \mathsf{L}(G)$, $\Phi_{i,j}(G, S) = \phi_{i,j}(G) - \phi_{i,j}(S)$. Note that $\mathsf{L}(G)$ should be equivalent to $\mathsf{L}(S)$.

We further extend the definition of the cophenetic distance to a set of trees. Given a set of trees $\mathcal{P}$ and a supertree of $\mathcal{P}$, S, the cophenetic distance between S and $\mathcal{P}$ is

$$\mathsf{d}_{\phi,p}(\mathcal{P}, S) := \sum_{G \in \mathcal{P}} \mathsf{d}_{\phi,p}(G, S[\mathsf{L}(G)]).$$

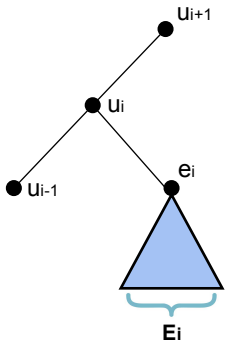

Fig. 1: An example of an exit-node and the corresponding subtree

In practice, the input trees are of different sizes and might be involved with different sets of taxa. Consequently, the supertrees are typically larger than the input trees. Note however that we defined the cophenetic distance only for trees over the same taxon set. Hence, in the above equation, we use the *minus method* [28] to account for the cophenetic distance between the supertree S and an input tree G (that is, we prune the extra information from S when comparing to G).

Path-induced subtrees. Let $P = (u_0, u_1, \ldots, u_k)$ be a simple path in a tree T, then *i-th exit node*, denoted by $e_i(P)$, for $0 < i < k$ is defined as follows:

(a) **If** $u_{i-1} \prec u_i \prec u_{i+1}$, then $e_i := \mathsf{Sb}_T(u_{i-1})$; Figure 1 depicts an example for this case;
(b) **If** $u_{i+1} \prec u_i \prec u_{i-1}$, then $e_i := \mathsf{Sb}_T(u_{i+1})$.

Additionally, if $u_{k-1} \prec u_k$, then $e_k := \mathsf{Sb}_T(u_{k-1})$; otherwise, $e_k := u_k$. For brevity, $E_i(P) := C_{e_i(P)}$ (*i-th exit cluster*), or simply E_i when the path P can be inferred from the context. Figure 2 illustrates the exit nodes induced by two different types of paths.

3 Cophenetic Median Tree Problems

In this section, we formulate and explore the basic properties of the median tree problems defined for cophenetic distances under different vector norms. Based on the definitions presented in the previous section we introduce a class of *cophenetic median tree problems* as follows:

Problem 2.1 (Cophenetic median tree (for an L_p norm) – decision version).
Instance: A set of input trees $\mathcal{P}$ and a real number q;
Question: Determine whether there exists a supertree S, such that $\mathsf{d}_{\phi,p}(\mathcal{P}, S) \leq q$.

3.1 Cophenetic median tree problems are NP-hard

We prove that Problem 2.1 is NP-hard under any L_p vector norm. The proof is based on the NP-hardness proof of the related path-difference median tree problem [15]. To show NP-hardness we provide a reduction from the NP-complete MaxRTC [29].

Problem 2.2 (Maximum Compatible Subset of Rooted Triplets – MaxRTC).
Instance: A set of rooted triplets $\mathcal{R}$ and an integer $0 \leq c \leq |\mathcal{R}|$;
Question: Is there a subset $\mathcal{R}' \subseteq \mathcal{R}$, such that $\mathcal{R}'$ is compatible and $|\mathcal{R}'| \geq c$.

Theorem 3.1. *The cophenetic median tree problem under an L_p norm is NP-hard for any $p \geq 1$.*

Proof. Consider a rooted triplet R and a tree S, such that $\mathsf{L}(R) \subseteq \mathsf{L}(S)$. Note that if S is *consistent* with R, then $\mathsf{d}_{\phi,p}(R, S[\mathsf{L}(R)]) = 0$; otherwise, $\mathsf{d}_{\phi,p}(R, S[\mathsf{L}(R)]) = 4^{\frac{1}{p}}$. The latter relationship can be easily verified by, for example, manually computing a cophenetic distance between two incompatible triplets $((a,b),c)$ and $((a,c),b)$.

Given that $\mathsf{d}_{\phi,p}(R, S[\mathsf{L}(R)])$ is a constant that depends only on whether S is consistent with R or not, we can map an instance $\langle \mathcal{R}, c\rangle$ of the MaxRTC problem to an instance $\langle \mathcal{R}, (|\mathcal{R}| - c)4^{\frac{1}{p}}\rangle$ of the cophenetic median tree problem. It is not difficult to observe that $\langle \mathcal{R}, (|\mathcal{R}| - c)4^{\frac{1}{p}}\rangle$ is a *yes*-instance if and only if $\langle \mathcal{R}, c\rangle$ is a *yes*-instance of MaxRTC. □

4 SPR-Local search for the cophenetic median tree problem

Here, we describe an efficient algorithm for estimating cophenetic median trees using a standard local search approach under the classic SPR tree edit operation.

4.1 *SPR*-Local search framework

We briefly introduce preliminary terminology and notions following [14], [15]. *Given a node $v \in V(S)\backslash\{\mathsf{Rt}(S)\}$, and a node $u \in V(\overline{S(v)})$, $SPR_S(v,u)$ is a tree obtained by the following modifications of the tree $S' = \overline{S(v)}$:*

1) *If u is a root of S', then a new root w' is introduced, so that u is a child of w'. Otherwise, an edge $(\mathsf{Pa}(u), u)$ is subdivided by a new node w'.*
2) *Connect the pruned subtree $S(v)$ to the node w'.*

Further, we define the following sets of trees that can be obtained from S by performing SPR:

$$SPR_S(v) := \bigcup_u SPR_S(v,u); \quad SPR_S := \bigcup_{v,u} SPR_S(v,u).$$

SPR_S is called an *SPR-neighborhood* of a tree S. It is easy to see from the definition that $|SPR_S| = O(n^2)$, where $n = |\mathsf{L}(S)|$.

Given a set of input trees $\mathcal{P} := \{G_1, ..., G_k\}$, the search space in an SPR local search problem could be viewed as a graph $\mathcal{T}$, where nodes represent all existing supertrees (candidate median trees) of $\mathcal{P}$. $\{S_1, S_2\}$ is an edge in $\mathcal{T}$, if S_1 could be transformed to S_2 with a single SPR operation.

At each iteration, the local search heuristic finds a candidate tree S' in the neighborhood of a current tree S, such that S' minimizes the cost function that we are interested in. In case $S \equiv S'$, the local search stops (reaches a local minimum). Otherwise, it proceeds to the next iteration with a tree S'. An instance (single iteration) of the SPR-based local neighborhood search problem could be formalized as follows:

Problem 3.1 (Cophenetic metric local neighborhood search).
Instance: An input set $\mathcal{P}$ and a candidate tree (supertree) S;
Question: Find a tree $S' = \underset{S' \in SPR_S}{\arg\min}\, \mathsf{d}_{\phi,p}(\mathcal{P}, S')$.

Naïve algorithm for the local neighborhood search. Given two trees, S and G, one can compute $\mathsf{d}_{\phi,p}(S,G)$ in $O(n^2)$ time for any p. Therefore, direct computation of the $\mathsf{d}_{\phi,p}(\mathcal{P}, S')$ scores for each $S' \in SPR_S$ would take $O(n^4 k)$ time, where $n = |\mathsf{L}(\mathcal{P})|$ and $k = |\mathcal{P}|$. Next, we show how to improve on this complexity under $p = 2$ (the *Euclidean* distance) and $p = 1$ (the Manhattan distance).

To fix the set up, let $\mathbf{G} \in \mathcal{P}$ be a fixed input tree, and let S_i be a supertree in the i-th iteration of the local search. Throughout the next section we refer to the restricted tree $S_i[\mathsf{L}(G)]$ as simply by $\mathbf{S}$.

4.2 Analysis of the SPR-environment

To design a faster algorithm for the cophenetic local neighborhood search problem we examine the structure of the SPR-environment of a candidate median tree. We are interested in the structure of the cophenetic difference matrix $\Phi(T,S)$ for some $T = SPR_S(v,w)$. Let $U_T := (v = u_0, \ldots, u_t = w)$ be the path between v and w in S, and let u_h be the node closest to the root of S on that path (i.e., $u_h \succeq u_i$ for all $0 \leq i \leq t$).

We distinguish three shapes of the path U_T that are relevant to our analysis.

(i) **Downward path**. Corresponds to $w \prec \mathsf{Sb}(v)$. That is, the node w is located in the subtree rooted at $\mathsf{Sb}(v)$. Figure 2 (left) depicts that case. Note that $u_h = u_1$.
(ii) **Upward path**. This is the case, when $v \prec w$. For a schematic example of such path see Figure 2 (right). Observe that $u_h = u_t$ in this case.
(iii) **Bended path**. This case can be described as $\mathsf{Sb}(v) \not\prec w$ and $\mathsf{Sb}(v) \not\succeq w$. That is, u_h is between u_1 and u_t exclusively. Figure 3 depicts this scenario.

Next we analyze the structure of matrix $\Phi(S,T)$ by considering a few major cases that provably cover the whole matrix.

(i) U_T is an **upward** path (see Figure 2 (right)).
  a) $\forall \boldsymbol{i \in C_v, j \in E_p}$: $\phi_{i,j}(T) = \phi_{i,j}(S) - (t-p)$ for $1 \leq p \leq t-1$. This case characterizes the change in depths of LCAs between leaves in C_v and leaves in exit clusters of the path U_T (see *path-induced subtrees* in Section 2). Note that while $\mathsf{LCA}_S(i,j) = u_p$, after regrafting we have $\mathsf{LCA}_T(i,j) = \mathsf{Pa}_T(w) = \mathsf{Pa}_T(v)$.
  b) $\forall \boldsymbol{i \in C_w \backslash C_v, j \in C_w \backslash (E_1 \cup C_v)}$: $\phi_{i,j}(T) = \phi_{i,j}(S) + 1$. This change is due to the fact that we add a new node, $\mathsf{Pa}_T(w)$, on the paths between the nodes u_p (for $2 \leq p \leq t$) and the root. Note, however, that the depth of the node e_1 remains unchanged.

(ii) U_T is a **downward** path (see Figure 2 (left)).

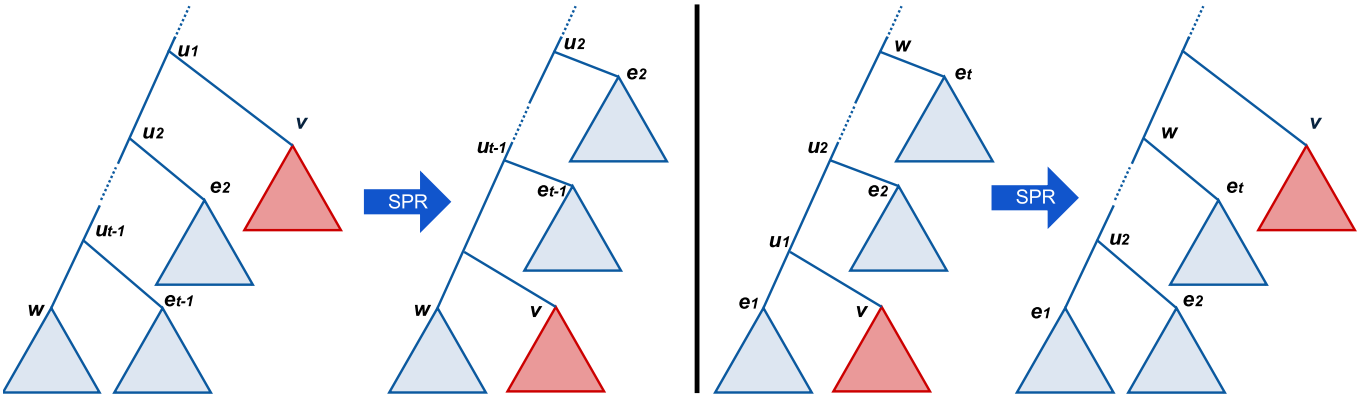

Fig. 2: (left) shows how the $SPR(v, w)$ operation changes the tree structure, when w is an descendant of $\mathsf{Sb}(v)$ – downward path; (right) shows how the $SPR(v, w)$ operation changes the tree, when w is an ancestor of v and $\mathsf{Sb}(v)$ – upward path.

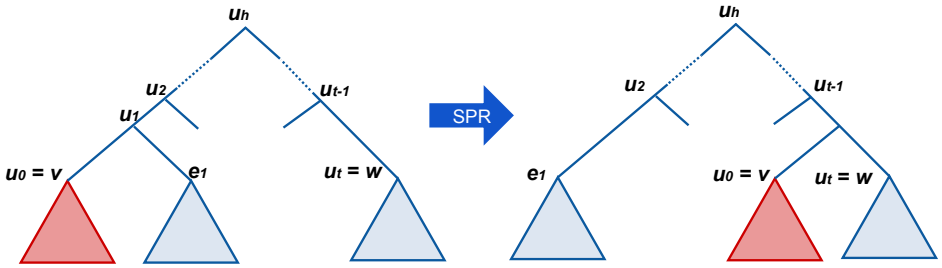

Fig. 3: Bended SPR path example.

a) $\forall \boldsymbol{i} \in \boldsymbol{C_v}, \boldsymbol{j} \in \boldsymbol{E_p} : \phi_{i,j}(T) = \phi_{i,j}(S) + (p-2)$ for $2 \leq p \leq t$. Due to the observation that $\mathsf{LCA}_S(i,j) = u_1$, while after regrafting $\mathsf{LCA}_T(i,j) = \mathsf{Pa}_T(e_p)$.

b) $\forall \boldsymbol{i} \in \boldsymbol{C_{u_2}}, \boldsymbol{j} \in \boldsymbol{C_{u_2}} \backslash \boldsymbol{E_t} : \phi_{i,j}(T) = \phi_{i,j}(S) - 1$. We removed the node, $\mathsf{Pa}_S(v)$, from the paths between the nodes u_p (for $2 \leq p \leq t-1$) and the root. However, the depth of the node $e_t = w$ remains unchanged.

(iii) U_T is a **bended** path (see Figure 3).

a) $\forall \boldsymbol{i} \in \boldsymbol{E_1}, \boldsymbol{j} \in \boldsymbol{E_1} : \phi_{i,j}(T) = \phi_{i,j}(S) - 1$. Node e_1 becomes one edge closer to the root.

b) $\forall \boldsymbol{i} \in \boldsymbol{C_w}, \boldsymbol{j} \in \boldsymbol{C_w} : \phi_{i,j}(T) = \phi_{i,j}(S) + 1$. Node w becomes one edge further from the root.

c) $\forall \boldsymbol{i} \in \boldsymbol{C_v}, \boldsymbol{j} \in \boldsymbol{E_p} : \phi_{i,j}(T) = \phi_{i,j}(S) + (p-h)$ for $1 \leq p \leq t, p \neq h$. For $p < h$, we have $\mathsf{LCA}_S(i,j) = u_p$ and $\mathsf{LCA}_T(i,j) = u_h$; as for $p > h$, we have $\mathsf{LCA}_S(i,j) = u_h$ and $\mathsf{LCA}_T(i,j) = \mathsf{Pa}_T(e_p)$.

(iv) For all three path shapes the following holds: $\forall \boldsymbol{i}, \boldsymbol{j} \in \boldsymbol{C_v} : \phi_{i,j}(T) = \phi_{i,j}(S) - \delta_v(S) + \delta_w(S) + 1$. Since we regraft the subtree $S(v)$ above w, depths of all nodes inside this subtree increase by $(\delta_v(T) - \delta_v(S))$, where $\delta_v(T) = \delta_w(S) + 1$; hence, the change in the cophenetic vector.

(v) Observe that for any other choice of i and j, the corresponding cophenetic value, $\phi_{i,j}$, is not affected.

Overall, the following clusters are involved in the changes in the cophenetic vector of S: i is one of the clusters in $\mathscr{C}_i = \{C_v, C_w, C_{u_2}, E_1\}$, and j appears in $\mathscr{C}_j = \{C_v, C_w, C_{u_2}, E_1, \ldots, E_t\}$. The key observation is that regardless of the form of U_T, there are only $O(t)$ pairs of clusters $(C_i, C_j) \in \mathscr{C}_i \times \mathscr{C}_j$ such that the respective cophenetic values are altered.

Next, based on this analysis we can calculate how the *cophenetic distance* changes, when an arbitrary SPR operation is performed on S.

4.2.1 Euclidean distance inference

Given $(C_i, C_j) \in \mathscr{C}_i \times \mathscr{C}_j$, let $d(C_i, C_j)$ be the value, such that $\forall (i,j) \in C_i \times C_j,\ \phi_{i,j}(T) = \phi_{i,j}(S) + d(C_i, C_j)$ according to the cases outlined above. For example, $d(C_v, C_v) = -\delta_v(S) + \delta_w(S) + 1$.

For technical reasons, related to the fact that the distance is calculated over vectors instead of matrices, we define

$$C_1 \otimes C_2 := \{(i,j) \in C_1 \times C_2 : i \leq j \text{ or } (j,i) \notin C_1 \times C_2\}$$

In most cases the fixed clusters C_1 and C_2 are clear from the context, and hence we use the shorthand notation, d and $\Phi_{i,j}$ for $d(C_1, C_2)$ and $\Phi_{i,j}(C_1, C_2)$ respectively. Below we provide the final equation that has been adopted from the work on Euclidean path-difference median trees [14].

$$\mathsf{d}^2_{\phi,2}(T,G) - \mathsf{d}^2_{\phi,2}(S,G) = \sum_{\substack{C_1 \in \mathscr{C}_i \\ C_2 \in \mathscr{C}_j}} \sum_{\substack{(i,j) \in \\ C_1 \otimes C_2}} \left((\Phi_{i,j} + d)^2 - \Phi^2_{i,j}\right)$$

$$= \sum_{\substack{C_1 \in \mathscr{C}_i \\ C_2 \in \mathscr{C}_j}} \left(|C_1 \otimes C_2| d^2 + 2d \sum_{\substack{(i,j) \in \\ C_1 \otimes C_2}} \Phi_{i,j} \right). \quad (1)$$

4.2.2 Manhattan distance inference

To establish the equation for the Manhattan cophenetic distance we use the same conventions as for the Euclidean distance above. The following equation has been adopted from the work on Manhattan path-difference median trees [15]

$$\mathsf{d}_{\phi,1}(T,G) - \mathsf{d}_{\phi,1}(S,G) = \sum_{\substack{C_1\in\mathscr{C}_i\\ C_2\in\mathscr{C}_j}} \sum_{\substack{(i,j)\in\\ C_1\otimes C_2}} \left(|\Phi_{i,j}+d| - |\Phi_{i,j}|\right)$$
$$= \sum_{\substack{C_1\in\mathscr{C}_i\\ C_2\in\mathscr{C}_j}} \left(\begin{array}{c} d\cdot\#\{(i,j)\in C_1\otimes C_2 | \Phi_{i,j}\geq -d)\} \\ -d\cdot\#\{(i,j)\in C_1\otimes C_2 | \Phi_{i,j}< -d)\} \\ +2\sum\limits_{\substack{(i,j)\in C_1\otimes C_2:\\ -d\leq\Phi_{i,j}<0}} \Phi_{i,j} \;-\; 2\sum\limits_{\substack{(i,j)\in C_1\otimes C_2:\\ 0\leq\Phi_{i,j}<-d}} \Phi_{i,j} \end{array} \right). \tag{2}$$

Observe that it immediately appears that the Manhattan case is computationally more complex than the Euclidean case, as it involves more parameters.

In the remaining parts of this section, we describe efficient dynamic programming algorithms for the computation of Equations 1 and 2 for all trees $T \in SPR_S$.

4.3 Efficient algorithms

We now apply the presented above analysis of the SPR-neighborhood to design more efficient local search algorithms for the L_2 and L_1 cophenetic metrics.

4.3.1 Local search iteration for the Euclidean distance

Preprocessing. According to Equation 1, in order to efficiently compute the distance $\mathsf{d}_{\phi,2}(T,G)$ for an arbitrary $T \in SPR_S$, we need to be able to compute the sum $\sum\limits_{\substack{(i,j)\in\\ C_1\otimes C_2}} \Phi_{i,j}$ efficiently for any two clusters C_1, C_2 of S.

A very similar pre-computation scheme was developed for the L_2 path-difference median tree problem [14], which we employ here. The pre-processing algorithm from [14] can be adapted to compute the matrix M with a column and a row for each cluster in S, and

$$M(C_1,C_2) = \sum_{(i,j)\in C_1\otimes C_2} \Phi_{i,j}.$$

Using the dynamic programming approach matrix M can be precomputed in only $\Theta(n^2)$ time given the difference matrix $\Phi(S,G)$. We now differentiate between all the instances of the difference matrix alteration outlined in Section 4.2 in order to compute Equation 1 given M. That is,

$$\mathsf{d}^2_{\phi,2}(T,G) - \mathsf{d}^2_{\phi,2}(S,G) = \tag{3}$$

(i) $\boldsymbol{C_1 = C_v,\ C_2 = C_v}$. In this case $d = d(C_v,C_v) = -\delta_v(S) + \delta_w(S) + 1$. Plugging that it as a part of Equation 1, we have

$$|C_v\otimes C_v|(d(C_v,C_v))^2 + 2d(C_v,C_v)M(C_v,C_v).$$

(ii) U_T is a **downward** path.

a) $\boldsymbol{C_1 = C_v,\ C_2 = E_p}$ for $2 \leq p < t$. $d = p-2$.

$$+\sum_{p=2}^{t-1} |C_v||E_p|(p-2)^2 + 2(p-2)M(C_v,E_p)$$

b) $\boldsymbol{C_1 = C_v,\ C_2 = E_t}$. $d = t-2$.

$$+|C_v||E_t|(t-2)^2 + 2(t-2)M(C_v,E_t)$$

c) $\boldsymbol{C_1 = C_{u_2},\ C_2 = C_{u_2}\backslash E_t}$. $d = -1$

$$+(|C_{u_2}\otimes C_{u_2}| - |E_t\otimes E_t|) - 2(M(C_{u_2},C_{u_2}) - M(E_t,E_t))$$

(iii) U_T is an **upward** path.

a) $\boldsymbol{C_1 = C_v,\ C_2 = E_p}$ for $1 \leq p \leq t-1$. We have $d = d(C_v,E_p) = -(t-p)$.

$$+\sum_{p=1}^{t-1} |C_v||E_p|(p-t)^2 + 2(p-t)M(C_v,E_p)$$

b) $\boldsymbol{C_1 = C_w\backslash C_v,\ C_2 = C_w\backslash(C_v\cup E_1)}$. $d = 1$

$$+ (|C_w\otimes C_w| - |E_1\otimes E_1| - |C_v\otimes C_w|)$$
$$+ 2(M(C_w,C_w) - M(E_1,E_1) - M(C_v,C_w)$$

(iv) U_T is a **bended** path. Recall that u_h is the node on the path with the smallest depth.

a) $\boldsymbol{C_1 = C_v,\ C_2 = E_p}$ for $1 \leq p < t, p \neq h$. We have $d = d(C_v,E_p) = p-h$.

$$+\sum_{p=1}^{t-1} |C_v||E_p|(p-h)^2 + 2(p-h)M(C_v,E_p)$$

b) $\boldsymbol{C_1 = C_v,\ C_2 = E_t}$. We have $d = t-h$.

$$+|C_v||E_t|(t-h)^2 + 2(t-h)M(C_v,E_t)$$

c) Combining the cases $\boldsymbol{C_1 = E_1,\ C_2 = E_1}$ and $\boldsymbol{C_1 = C_w,\ C_2 = C_w}$ we have

$$+|E_1\otimes E_1| - 2M(E_1,E_1) + |C_w\otimes C_w| + 2M(C_2,C_w)$$

Observe that most of the above parts of the equation can be computed in a constant time for an arbitrary path U_T, except for the items (ii)a, (iii)a, and (iv)a. Next, we develop a dynamic programming approach to compute the summations in these three items in constant time as well.

Efficient neighborhood traversal. For an arbitrary tree $T \in SPR_S$, let $\mathcal{Q}(T)$ denote the sum either in item (ii)a, (iii)a, or (iv)a, depending on the form of U_T. For example, if $U_T = (u_0,\ldots,u_t)$ and $u_0 \prec u_t$ (an upward path), then

$$\mathcal{Q}(T) = \sum_{p=1}^{t-1} |C_v||E_p|(p-t)^2 + 2(p-t)M(C_v,E_p)$$

Further, let $T, T' \in SPR_S$, such that the path $U_T = (u_0,\ldots,u_t)$ is a prefix of the path $U_{T'} = (u_0,\ldots,u_t,u_{t+1})$. Then we can consider two possibilities:

- $\boldsymbol{u_t \prec u_{t+1}}$. In this case both paths U_T and $U_{T'}$ must go *upward*. That is, both paths fall into the category (iii)a in the above differentiation. Then it can be verified that

$$\delta_U = \mathcal{Q}(T') - \mathcal{Q}(T) = \sum_{p=1}^{t} |C_v||E_p| - 2\sum_{p=1}^{t} M(C_v,E_p) - 2\sum_{p=1}^{t}(p-t)|C_v||E_p|$$

- $\boldsymbol{u_t \succ u_{t+1}}$. We distinguish two cases now based on the form of the path U_T.

– U_T is an upward path ($h = t$) or a bended path ($h < t$). In this case $U_{T'}$ must be a *bended* path.

$$\begin{aligned}\delta_B &:= \mathcal{Q}(T') - \mathcal{Q}(T)\\ &= \sum_{p=1}^{t} |C_v||E_p|(p-h)^2 + 2(p-h)M(C_v,E_p)\\ &- \sum_{p=1}^{t-1} |C_v||E_p|(p-h)^2 + 2(p-h)M(C_v,E_p)\\ &= |C_v||E_t|(t-h)^2 + 2(t-h)M(C_v,E_t).\end{aligned}$$

– U_T is a downward path, then $U_{T'}$ is a *downward* path as well.

$$\begin{aligned}\delta_D &:= \mathcal{Q}(T') - \mathcal{Q}(T)\\ &= |C_v||E_t|(t-2)^2 + 2(t-2)M(C_v,E_t).\end{aligned}$$

Observe now that in the second case ($u_t \succ u_{t+1}$) the value $\mathcal{Q}(T')$ can be computed in constant time given $\mathcal{Q}(T)$ without any additional information. That observation enables a simple dynamic programming structure, when we explore the constrained neighborhood $SPR_S(v)$ in the order of the increase of the lengths of paths U_T (for more detailed description, see [14]).

In case when $U_{T'}$ is an upward path, we need to have the following additional information available:

$$\begin{aligned}\mathrm{d}\mathcal{Q}_1(T) &= \sum_{p=1}^{t} |C_v||E_p|\\ \mathrm{d}\mathcal{Q}_2(T) &= \sum_{p=1}^{t} M(C_v,E_p)\\ \mathrm{d}\mathcal{Q}_2(T) &= \sum_{p=1}^{t} (p-t)M(C_v,E_p)\end{aligned}$$

In fact, these values can be also maintained using the dynamic programming structure by adapting an approach from [14].

Complexity analysis. As mentioned above, the required preprocessing step can be performed in $\Theta(n^2)$ time for a fixed pair of trees S and G. Further, the efficient neighborhood traversal approach enables the algorithm to compute the distance for each tree $T \in SPR_S$ in constant time. Given that $|SPR_S| = \Theta(n^2)$, all the computations for a fixed input tree G can be performed in $\Theta(n^2)$ time. Finally, for k input trees the overall time complexity of a single local search iteration will amount to $\Theta(kn^2)$.

4.3.2 Local search iteration for the Manhattan distance

Observe that for the computation of a Manhattan cophenetic distance using Equation 2, one need to be able to compute "count" and "sum" statistics over submatrices of $\Phi(S,G)$.

Preprocessing. In order to compute such statistics we adopt data structures developed in the work on Manhattan path-difference median trees [15]. Next, we briefly describe the resulting preprocessing idea.

Let $\mathscr{C}(S)$ be a set of all clusters in a tree S. For $\mathsf{L}_1, \mathsf{L}_2 \in \mathscr{C}(S)$, we define $\Sigma_{\geq}(\mathsf{L}_1,\mathsf{L}_2)$ to be a vector indexed from $-n$ to n, such that

$$\Sigma_{\geq}(\mathsf{L}_1,\mathsf{L}_2)[x] = \Sigma_{\geq}(\mathsf{L}_2,\mathsf{L}_1)[x] = \sum_{\substack{(i,j)\in L_1\otimes L_2:\\ \Delta_{i,j}\geq x}} \Phi_{i,j}$$

Similarly, we define a vector $\#_{\geq}(\mathsf{L}_1,\mathsf{L}_2)$ indexed from $-n$ to n, such that

$$\#_{\geq}(\mathsf{L}_1,\mathsf{L}_2)[x] = \#\{(i,j)\in L_1\otimes L_2 : \Phi_{i,j}\geq x\}$$

It is not difficult to check that given such vectors it is possible to compute $\mathsf{d}_{\phi,1}(T,G) - \mathsf{d}_{\phi,1}(S,G)$ for an arbitrary $T \in SPR_S$ in $O(n)$ time using Equation 2. For example, let's consider some $T = SPR_S(v,w)$, such that the corresponding path U_T is a part of a path from v to the root (i.e., $v \prec w$). That means that if we choose $C_1 = C_v$ and $C_2 = E_1$, then $d(C_1,C_2) = -t+1$ according to the analysis presented in Section 4.2, where $t \geq 2$ is the length of U_T in edges. We can now use the vectors defined above to compute the part of the sum in Equation 2 that corresponds to clusters C_1 and C_2. That is, we observe the following relations:

$$\#\{(i,j)\in C_1\otimes C_2 : \Delta_{i,j} \geq t-1\} = \#_{\geq}(C_1,C_2)[t-1]$$

$$\begin{aligned}\#\{(i,j)\in C_1\otimes C_2 : \Delta_{i,j} < t-1\} &= \#_{\geq}(C_1,C_2)[-n]\\ &- \#_{\geq}(C_1,C_2)[t-1])\end{aligned}$$

$$\begin{aligned}\sum_{\substack{(i,j)\in C_1\otimes C_2:\\ 0\leq\Delta_{i,j}<t-1}} \Delta_{i,j} &= \Sigma_{\geq}(C_1,C_2)[0]\\ &- \Sigma_{\geq}(C_1,C_2)[t-1]\end{aligned}$$

Complexity analysis. An algorithm for computing the vectors $\Sigma_{\geq}$ and $\#_{\geq}$ efficiently was presented in [15]. Although in [15] the vectors were defined in a slightly different way, the algorithm can be adopted for our needs (we omit the technical details for brevity).

The time complexity for computing these vectors is $\Theta(n^3)$ for a fixed input tree G. Having these vectors computed we can calculate the values $\mathsf{d}_{\phi,1}(T,G)$ in $\Theta(n)$ for all $T \in SPR_S$. Given that $|SPR_S| = \Theta(n^2)$ and the number of input trees is k, the overall time complexity is $\Theta(kn^3)$.

5 Experimental Evaluation

In this section, we explore the applicability of the cophenetic local search heuristics enabled by the developed algorithms. First, we present the runtime study of both Manhattan and Euclidean local search heuristics. Further, following the standard applicability study approach for supertree methods, we validate the two heuristics on three benchmark phylogenetic datasets. Finally, we experimentally demonstrate some properties of the cophenetic family of metrics.

5.1 Runtime analysis

We compare the runtime of pure local search strategies (when a starting tree is chosen randomly) between the improved and naïve algorithms.

Datasets. We estimate the runtime for randomly generated sets of input trees. We generated 12 random input sets with 10 trees in each over the number of taxa varying from 10 in the smallest dataset to 120 in the largest one, with a step of 10. The random input trees were generated from uniform tree distributions using PAUP* [9].

Experimental setting. We consider the following four local search heuristics: (i) Naïve Manhattan local search, (ii) Improved Manhattan local search (see Section 4.3.2), (iii) Naïve

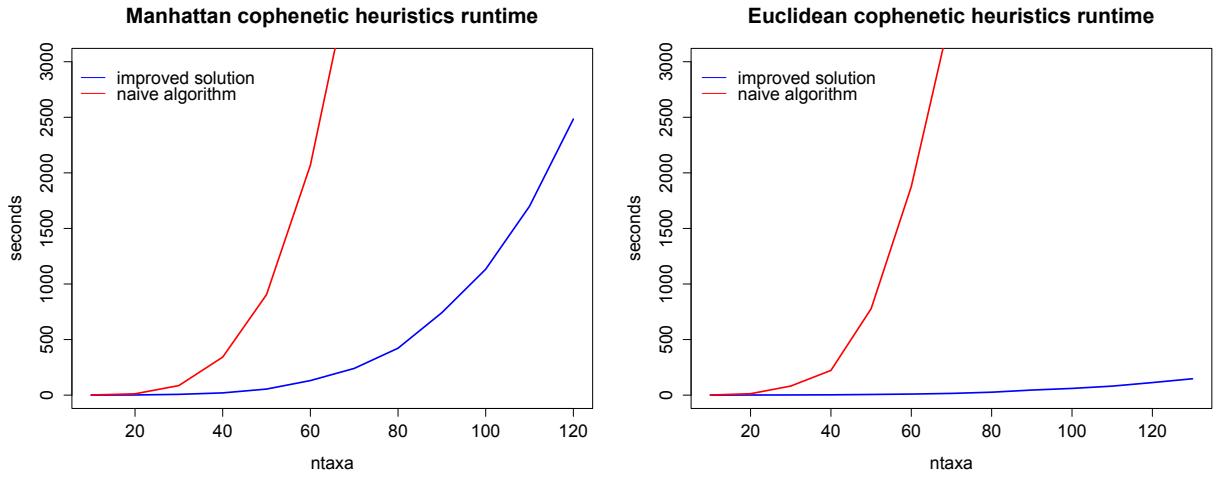

Fig. 4: The growth in runtime of randomized local search heuristics with a gradual increase of the number of taxa in input datasets. Mean runtimes among five trials are presented for the Manhattan local search heuristics (left) and the Euclidean local search heuristics (right).

Euclidean local search, and (iv) Improved Euclidean local search (Section 4.3.1) algorithms. For each of the generated datasets, we ran each of the listed heuristics five times. The repeated number of runs on the same dataset is necessary since the starting tree for local search is randomly generated and it might significantly affect the runtime. We present the average runtimes over the five trials in Figure 4.

Results. Figures 4 depicts that the runtime for the improved local search algorithms grows *substantially slower* than the runtime under the naïve algorithms. Both for the Manhattan and Euclidean cases the naïve algorithm becomes infeasible at approximately 70 taxa. However, when applying the here developed algorithms, the local search becomes feasible for much larger supertree (median tree) problem instances. In fact, in the following study, we demonstrate that the improved Manhattan and Euclidean local search heuristics can be applied to much larger phylogenetic datasets in practice.

Additionally, Figure 4 clearly showcases the advantage of using the Euclidean (cophenetic) median tree algorithm, as it provides a significantly better time complexity for the local search iteration. However, in certain cases, the Manhattan median tree heuristic could be more preferable, which is demonstrated in the next study.

5.2 Applicability study

In this section we estimate cophenetic median trees using the algorithms developed in this work and compare them to the related path-difference median trees and other supertrees constructed via several well-recognized methods. The study is conducted over two standard empirical datasets. Our objective is to (i) evaluate the applicability of the local search approach to the cophenetic median tree problem, and (ii) to identify relations among different supertree and median tree methods.

Datasets. Following the original work on path-difference median trees as well as other supertree studies, we evaluate our cophenetic median tree heuristics on published baseline datasets [15], [26].

(i) *Cetartiodactyla* dataset: contains 201 trees over 299 taxa overall [12];
(ii) *Marsupials* dataset: contains 158 trees over 272 taxa overall [10].
(iii) *Placental mammals* dataset: contains 725 trees over 126 taxa (including the artificial outgroup taxon) gathered in the influential study by Beck et al. [30].
These three datasets are considerably large and serve as benchmarks for many phylogenetic studies (e.g., see [31], [32], [33], [34]).

Methods. In order to obtain credible estimates for Manhattan cophenetic median trees (MCMT) and Euclidean cophenetic median trees (ECMT) we used the *hybrid heuristic* framework that was shown to outperform other standard local search paradigms when computing path-difference median trees [15].

The *hybrid heuristic*, similarly to other tree-building approaches (such as the classic greedy approach), aims to construct a starting tree for the local search procedure that will be by itself a good median tree estimate. The key feature of the hybrid heuristic is that it uses local search in the process of constructing such a starting tree. In spite of the expected overhead computational cost associated with using local search for the starting tree construction, it was demonstrated that the hybrid heuristic can consistently outperform the classic greedy approach in terms of both accuracy and runtime [15].

For comparison we use two path-difference median tree heuristics – one for Manhattan median trees (MMT) and another for Euclidean median trees (EMT). Both of them employ the hybrid heuristic approach as well.

Additionally, following the preceding studies [15], [26], [31], we include the following methods in our study. The *modified min-cut* (MMC) algorithm, which is an exact polynomial time algorithm for the computation of supertrees [35]. Two *triplet median tree heuristics* (TH) that are local search heuristics for the NP-hard triplet median tree problem [29] using SPR and TBR tree edit operations respectively [31]. Note that TBR (stands for tree bisection

TABLE 1: Empricial evaluation of supertree methods over two published phylogenetic datasets. The best scores under each objective function are shown in bold.

Dataset	Method	L_1 cophen.	L_2 cophen.	L_1 PDD	L_2 PDD	Triplet-sim	MAST	Pars. score
Marsup 158 trees 272 taxa	MMC	1,564,728	15,993.0	1,681,015	16,670.5	51.73 %	58.5 %	3901
	MRP	122,459	2,856.0	515,257	5,694.6	98.29 %	**74.2** %	**2274**
	TH(SPR)	143,398	2,996.1	515,906	5,866.3	**99.0** %	73.4 %	2312
	TH(TBR)	143,501	3,005.3	517,274	5,888.2	**99.0** %	73.4 %	2317
	EMT	260,787	3,611.2	327,379	**4,380.8**	85.2 %	70.5 %	2869
	MMT	286,357	4,482.5	**323,909**	5,063.3	54.7 %	62.3 %	3817
	ECMT	66,863	**2,225.2**	389,807	4,888.5	95.8 %	71.6 %	2649
	MCMT	**60,737**	2,460.9	372,719	4,974.58	90.9 %	67.7 %	3036
Cetartio 201 trees 299 taxa	MMC	1,004,359	17,465.7	918,639	16,206.2	70.0 %	57.1 %	4929
	MRP	186,582	4,756.7	365,870	6,991.4	96.5 %	**69.7** %	**2603**
	TH(SPR)	168,620	4,568.2	403,233	7,630.0	**97.3** %	67.0 %	2754
	TH(TBR)	168,497	4,564.0	401,327	7,591.1	**97.3** %	67.0 %	2754
	EMT	209,680	4,742.3	258,836	**5,639.2**	86.0 %	65.8 %	3394
	MMT	225,705	5,207.0	**258,424**	6,143.0	66.3 %	59.4 %	4218
	ECMT	78,742	**3,293.1**	286,016	6,158.2	84.0 %	66.0 %	3345
	MCMT	**74,135**	3,594.0	288,411	6,620.9	87.8 %	61.5 %	3895
Placental Mammals 725 trees 126 taxa	MRP	164,079	9,530.0	284,447	13,524.5	82.6 %	**70.5** %	**9486**
	TH(SPR)	160,467	9,514.9	285,697	13,702.4	**82.7** %	69.3 %	9671
	TH(TBR)	160,467	9,514.9	285,697	13,702.4	**82.7** %	69.3 %	9671
	EMT	177,936	9,590.5	255,288	**12,517.6**	81.4 %	69.7 %	9791
	MMT	183,640	9,790.7	**252,999**	12,647.9	80.7 %	68.3 %	10136
	ECMT	151,867	**8,903.3**	270,103	13,001.1	81.3 %	68.0 %	10162
	MCMT	**148,195**	9,436.1	284,866	14,058.9	79.6 %	66.4 %	10597

and reconnection) is an extension of the SPR operation, where the pruned subtree is allowed to be re-rooted before regrafting it. Finally, we include the classic *maximum parsimony with representation heuristic* (MRP). MRP was recognized as the most applied supertree method among practitioners [8]. Here we use the implementation of the MRP heuristic in the popular software package, PAUP* [9], under the TBR branch swapping [31].

Experimental setting. To compare the methods under consideration we used the results of their execution over both datasets (each method was executed 10 times, except for MMC, which is a deterministic method). Please note that we used the published MMC supertrees for the Marsupials and Cetartiodactyla datasets [31]; however, we are not aware of any published MMC supertree for the placental mammals dataset and we were unable to compute it.

We further evaluated each of the generated supertrees with the respective input dataset using seven relevant objectives: the Manhattan and Euclidean cophenetic metrics, the Manhattan and Euclidean path-difference distances, triplet similarity (the objective function for triplet heuristics), the average maximum agreement subtree (MAST) similarity, and the parsimony score. The best scores among the 10 trials under each objective are presented in Table 1.

Results. Analyzing the Manhattan and Euclidean cophenetic distance objectives, we observe that they significantly differ from others in terms of the distribution of scores across the methods under consideration. That is, as expected, the developed MCMT and ECMT methods perform best under the L_1 and L_2 cophenetic objectives respectively. We also observe that the MRP and TH heuristics produce trees that are better in terms of cophenetic distances than the trees produced by path-difference median tree methods. On the other hand, cophenetic median tree heuristics in most cases produce trees that score better in terms of path-difference objectives than trees generated by MRP and TH heuristics. In summary, we observe rather asymmetric relationships among the median tree estimates computed by cophenetic and path-difference heuristics.

Further, observe that ECMT and MCMT median tree estimates perform quite well in terms of other classic objectives such as triplet-similarity, MAST, and parsimony.

5.2.1 Distribution analysis

The computed distances from Table 1 do not bare much information by themselves. It is more informative to consider L_1 and L_2 cophenetic distances in the context of the respective distance distributions. While it is difficult to compute exact cophenetic distance distributions, we estimate such distributions under the standard phylogenetic tree model, the pure-birth process also known as the Yule model [36], [37].

For each of the phylogenetic datasets we generated a supertree collection containing 10,000 trees drawn from the Yule distribution. Then for each of the generated supertrees the cophenetic distances to the respective datasets were computed under L_1 and L_2 norms. The resulting frequency histograms are presented in Figure 5.

It is interesting to observe that while ECMT and MCMT markers are located significantly to the left of the respective mean values, the triplet heuristic, path-difference median tree heuristics, and MRP produced trees that are not better than many sampled Yule (pure-birth) trees in terms of the

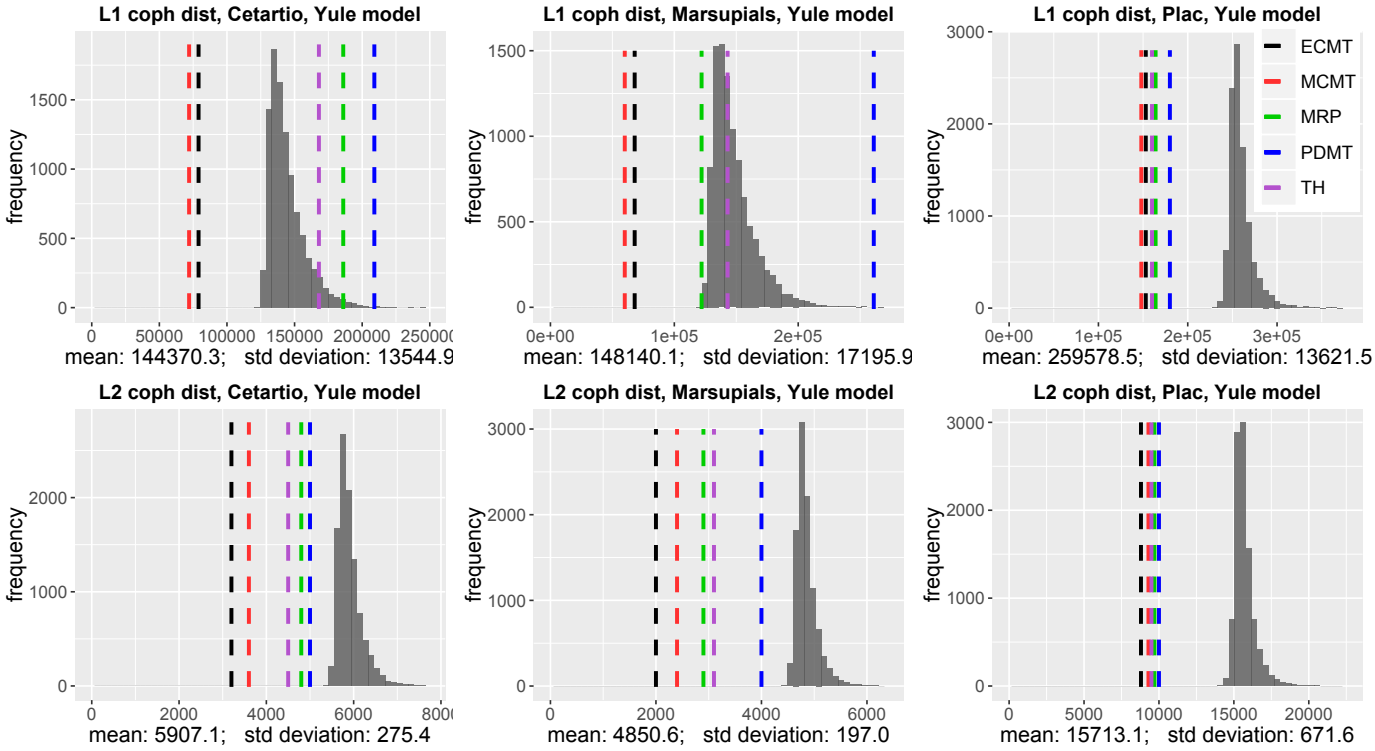

Fig. 5: Cophenetic distance distributions for the Cetartiodactyla, Marsupials, and Placental Mammals datasets under the Yule random tree model. The top three plots show the L_1 cophenetic distance distributions and the bottom three plots correspond to the L_2 distributions. Each plot contains markers (dashed lines) indicating the respective distances from Table 1. Note that TH(SPR) and TH(TBR) scores are combined and are represented by the *TH* marker. Similarly, EMT and MMT scores are combined and are represented by the *PDMT* marker.

L_1 cophenetic distance. The latter is observed for the first two datasets; however, this observation does not hold for the L_2 cophenetic distance distributions and the placental mammals dataset.

5.3 Correlations

Cardona et.al. studied correlations between the cophenetic metrics and other popular tree comparison metrics including the path-difference metrics and the classic Robinson-Foulds metric (RF) [16]. They have demonstrated that the Manhattan cophenetic distance (i) does not bare strong correlation with the Manhattan and Euclidean path-difference distances (Spearman correlation coefficient of ≈ 0.45), and (ii) has almost no significant correlation with the Robinson-Foulds distance (Spearman correlation coefficient of approximately ≈ -0.0008).

As was mentioned in the introduction, the cophenetic distance is dependent on the LCA mappings, and therefore, can be expected to be more closely related to the cost functions originating from gene tree parsimony (GTP) problems [38] than the path-difference distance or RF. We consider the following GTP cost functions: the gene duplication (GD) cost, the deep coalescences (DC) cost, and the duplications with losses (DL) cost. In fact, based on the formal definitions, the cophenetic distance is most closely related to the deep coalescence cost function, as both take into account the path-lengths between LCA mappings. In this section we test our hypothesis that the two cost functions are indeed correlated.

Experimental setting. In order to assess correlations between different cost functions we follow the Cardona et.al. setting. That is, we generated 5000 random pairs of bifurcating phylogenetic trees with 100 of labeled leaves each. The trees were drawn from the uniform distribution. Next, for each pair of trees, (T_1, T_2), we computed the Manhattan and Euclidean cophenetic distances as well as duplications, deep coalescence, and duplications with losses costs. Observe that the GD, DC, and DL cost functions are not symmetric. Thus, in order to compare it to the symmetric cophenetic distances, we computed the cost sums $C(T_1, T_2)+C(T_2, T_1)$, where $C \in \{GD, DC, DL\}$.

Based on the obtained scores for a thousand of tree pairs we computed Spearman correlation coefficients for each pair of cost functions (see Table 2).

TABLE 2: Spearman correlation coefficients for the five LCA-based cost functions.

	L_2 coph.	DC sum	GD sum	DL sum
L_1 coph.	0.987	0.771	0.090	0.762
L_2 coph	-	0.789	0.106	0.781
DC sum	-	-	0.322	0.999
GD sum	-	-	-	0.363

Results. It is important to note that the Manhattan and Euclidean cophenetic metrics are highly correlated (see Table 2). Further, close-to-one correlation is observed between the deep coalescence and the duplications with losses cost

functions. This was highly expected, since the DL cost function can be represented as a linear combination of DC and GD, where the DC cost is much more significant than the GD cost [39]. Further, in justification to our hypothesis, we observe a significant correlation between the Manhattan cophenetic distance and DC as well as the Euclidean cophenetic distance and DC. Observe that the Spearman correlation for these pairs of cost functions is higher than the respective correlation coefficients between the cophenetic distances and the path-difference distances. Plotting the Manhattan and Euclidean scores against DC also confirmed the existence of linear dependence.

6 Conclusion

The problem of discordance in phylogenetic trees has been addressed by the means of the median tree approach for over 20 years [8], [40], [41]. Median tree methods employ various objective cost functions, which can be classified into mathematically informed costs, and biologically informed costs. Biological costs are based on evolutionary processes causing discordance between two trees (e.g., deep coalescences, gene duplications, and gene duplications with losses [42]), which are typically used when gene trees are compared with species trees [42]. In contrast, independent of any evolutionary causes, mathematical costs between two trees are measuring the amount of elementary evolutionary information that is common (or different) in these trees, and are thought to be applicable as a tool of error-correction and formal maximization of common information among the input trees [8]. Another distinction between mathematical and biological costs is that the former typically satisfy the properties of a metric [43], while the latter once are not symmetric and do not satisfy the triangle inequality [42]. Despite the differences between mathematical and biological costs, in the median tree setting the cophenetic metrics, mathematical costs, and the deep coalescence cost function, a classic biological cost, showed to be strongly correlated in our experiments. This suggests that the cophenetic median tree methods may be used as a universal solution to the generalized supertree problem. Note also that as a metric, the cophenetic model provides valuable mathematical properties, which are not met by most of the biologically informed cost functions. The fact that the cophenetic metrics are not tied to a biological model also gives an advantage, as they can be naturally generalized for the comparison of weighted phylogenetic trees.

Here we presented an extensive study of two cophenetic median tree heuristics. The algorithmic contribution of this work made these heuristics feasible for large-scale phylogenetic analysis involving hundreds of taxa. The applicability study demonstrated that the two cophenetic heuristics produce well-correlated median trees; hence, we propose the *Euclidean* cophenetic median tree heuristic as a method of choice given that the observed runtime for this heuristic is significantly smaller compared to the Manhattan version. Further, while the trees generated by both heuristics performed well in terms of several classic distance/similarity measures, the Euclidean cophenetic heuristic performed often better than the Manhattan heuristic.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 1617626.

References

[1] S. Nik-Zainal, P. Van Loo, D. C. Wedge, L. B. Alexandrov, C. D. Greenman, K. W. Lau, K. Raine, D. Jones, J. Marshall, M. Ramakrishna, A. Shlien, S. L. Cooke, J. Hinton, A. Menzies, L. A. Stebbings, C. Leroy, M. Jia, R. Rance, L. J. Mudie, S. J. Gamble, P. J. Stephens, S. McLaren, P. S. Tarpey, E. Papaemmanuil, H. R. Davies, I. Varela, D. J. McBride, G. R. Bignell, K. Leung, A. P. Butler, J. W. Teague, S. Martin, G. Jönsson, O. Mariani, S. Boyault, P. Miron, A. Fatima, A. Langerød, S. A. J. R. Aparicio, A. Tutt, A. M. Sieuwerts, Å. Borg, G. Thomas, A. V. Salomon, A. L. Richardson, A.-L. Børresen-Dale, P. A. Futreal, M. R. Stratton, P. J. Campbell, and Breast Cancer Working Group of the International Cancer Genome Consortium, "The life history of 21 breast cancers," *Cell*, vol. 149, no. 5, pp. 994–1007, May 2012.

[2] R. A. Hufbauer, R. A. Marrs, A. K. Jackson, R. Sforza, H. P. Bais, J. M. Vivanco, and S. E. Carney, "Population structure, ploidy levels and allelopathy of Centaurea maculosa (spotted knapweed) and C. diffusa (diffuse knapweed) in North America and Eurasia," in *Proceedings of the XI International Symposium on Biological Control of Weeds, Canberra Australia*. Morgantown, WV.: USDA Forest Service. Forest Health Technology Enterprise Team, 2003, pp. 121–126.

[3] J. J. L. Roux, A. M. Wieczorek, M. M. Ramadan, and C. T. Tran, "Resolving the native provenance of invasive fireweed (Senecio madagascariensis Poir.) in the Hawaiian Islands as inferred Poir.) in the Hawaiian Islands as inferred from phylogenetic analysis," *Diversity and Distributions*, vol. 12, pp. 694–702, 2006.

[4] S. R. Harris, E. J. Cartwright, M. E. Török, M. T. Holden, N. M. Brown, A. L. Ogilvy-Stuart, M. J. Ellington, M. A. Quail, S. D. Bentley, J. Parkhill, and S. J. Peacock, "Whole-genome sequencing for analysis of an outbreak of meticillin-resistant staphylococcus aureus: a descriptive study," *Lancet Infect Dis*, vol. 13, no. 2, pp. 130–6, 2013.

[5] P. Forster and C. Renfrew, *Phylogenetic methods and the prehistory of languages*. McDonald Inst of Archeological, 2006.

[6] H. Gee, "Evolution: ending incongruence," *Nature*, vol. 425, no. 6960, p. 782, Oct 2003.

[7] J. O. McInerney, J. A. Cotton, and D. Pisani, "The prokaryotic tree of life: past, present... and future?" *Trends Ecol Evol*, vol. 23, no. 5, pp. 276–81, May 2008.

[8] O. R. Bininda-Emonds, Ed., *Phylogenetic Supertrees: Combining Information to Reveal the Tree of Life*, ser. Computational Biology. Springer Verlag, 2004, vol. 4.

[9] D. L. Swofford, "PAUP*. Phylogenetic analysis using parsimony (*and other methods). Version 4. Sinauer Associates, Sunderland, Massachusetts." 2002.

[10] M. Cardillo, O. R. P. Bininda-Emonds, E. Boakes, and A. Purvis, "A species-level phylogenetic supertree of marsupials," *Journal of Zoology*, vol. 264, pp. 11–31, 2004.

[11] M. Kennedy, R. D. Page, and R. Prum, "Seabird supertrees: combining partial estimates of procellariiform phylogeny," *The Auk*, vol. 119, no. 1, pp. 88–108, 2002.

[12] S. A. Price, O. R. P. Bininda-Emonds, and J. L. Gittleman, "A complete phylogeny of the whales, dolphins and even-toed hoofed mammals (cetartiodactyla)," *Biological Reviews*, vol. 80, no. 3, pp. 445–473, 2005.

[13] M. F. Wojciechowski, M. J. Sanderson, K. P. Steele, and A. Liston, "Molecular phylogeny of the "temperate herbaceous tribes" of papilionoid legumes: a supertree approach," *Advances in legume systematics*, vol. 9, pp. 277–298, 2000.

[14] A. Markin and O. Eulenstein, "Efficient local search for euclidean path-difference median trees," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2017.

[15] ——, "Computing manhattan path-difference median trees: a practical local search approach," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2017.

[16] G. Cardona, A. Mir, F. Rosselló, L. Rotger, and D. Sánchez, "Cophenetic metrics for phylogenetic trees, after sokal and rohlf," *BMC Bioinformatics*, vol. 14, no. 1, p. 3, 2013. [Online]. Available: http://dx.doi.org/10.1186/1471-2105-14-3

[17] R. R. Sokal and F. J. Rohlf, "The comparison of dendrograms by objective methods," *Taxon*, vol. 11, no. 2, pp. 33–40, 1962. [Online]. Available: http://www.jstor.org/stable/1217208
[18] D. Robinson, "Comparison of labeled trees with valency three," *Journal of Combinatorial Theory*, vol. 11, pp. 105–119, 1971.
[19] W. P. Maddison, "Gene trees in species trees," *Systematic biology*, vol. 46, no. 3, pp. 523–536, 1997.
[20] P. Bonizzoni, G. Della Vedova, and R. Dondi, "Reconciling a gene tree to a species tree under the duplication cost model," *Theoretical computer science*, vol. 347, no. 1-2, pp. 36–53, 2005.
[21] W. P. Maddison and L. L. Knowles, "Inferring phylogeny despite incomplete lineage sorting," *Syst Biol*, vol. 55, no. 1, pp. 21–30, 2006.
[22] C. Than and L. Nakhleh, "Species tree inference by minimizing deep coalescences," *PLoS Comput Biol*, vol. 5, no. 9, p. e1000501, 2009.
[23] M. S. Bansal, J. G. Burleigh, and O. Eulenstein, "Efficient genome-scale phylogenetic analysis under the duplication-loss and deep coalescence cost models," *BMC Bioinformatics*, vol. 11 Suppl 1, p. S42, 2010.
[24] R. Chaudhary, M. S. Bansal, A. Wehe, D. Fernández-Baca, and O. Eulenstein, "iGTP: a software package for large-scale gene tree parsimony analysis," *BMC Bioinformatics*, vol. 11, p. 574, 2010.
[25] H. T. Lin, J. G. Burleigh, and O. Eulenstein, "Consensus properties for the deep coalescence problem and their application for scalable tree search," *BMC Bioinformatics*, vol. 13 Suppl 10, p. S12, 2012.
[26] A. Markin and O. Eulenstein, "Path-difference median trees," in *Bioinformatics Research and Applications: 12th International Symposium, ISBRA 2016, Minsk, Belarus, June 5-8, 2016, Proceedings*, A. Bourgeois, P. Skums, X. Wan, and A. Zelikovsky, Eds. Cham: Springer International Publishing, 2016, pp. 211–223. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-38782-6_18
[27] C. Semple and M. A. Steel, *Phylogenetics*. Oxford: University Press, 2003.
[28] J. A. Cotton and M. Wilkinson, "Majority-rule supertrees," *Syst Biol*, vol. 56, no. 3, pp. 445–452, 2007.
[29] D. Bryant, "Hunting for trees, building trees andcomparing trees: theory and method in phylogeneticanalysis," Ph.D. dissertation, Department of Mathematics, University of Canterbury, New Zeeland, 1997.
[30] R. M. Beck, O. R. Bininda-Emonds, M. Cardillo, F.-G. R. Liu, and A. Purvis, "A higher-level mrp supertree of placental mammals," *BMC Evolutionary Biology*, vol. 6, no. 1, p. 93, Nov 2006. [Online]. Available: https://doi.org/10.1186/1471-2148-6-93
[31] H. T. Lin, J. G. Burleigh, and O. Eulenstein, "Triplet supertree heuristics for the tree of life," *BMC Bioinformatics*, vol. 10, no. Suppl 1, 2009.
[32] M. S. Bansal, J. G. Burleigh, O. Eulenstein, and D. Fernández-Baca, "Robinson-foulds supertrees," *Algorithms for Molecular Biology*, vol. 5, no. 1, pp. 1–12, 2010.
[33] S. Snir and S. Rao, "Quartets maxcut: A divide and conquer quartets algorithm," *IEEE/ACM TCBB*, vol. 7, no. 4, pp. 704–718, 2010.
[34] D. Chen, O. Eulenstein, D. Fernández-Baca, and J. Burleigh, "Improved heuristics for minimum-flip supertree construction," *Evolutionary Bioinformatics*, vol. 2, 2006.
[35] R. D. M. Page, "Modified mincut supertrees," in *Proceedings of the Second International Workshop on Algorithms in Bioinformatics*, ser. WABI '02. London, UK: Springer-Verlag, 2002, pp. 537–552.
[36] G. U. Yule, "A mathematical theory of evolution, based on the conclusions of dr. jc willis, frs," *Philosophical transactions of the Royal Society of London. Series B, containing papers of a biological character*, vol. 213, pp. 21–87, 1925.
[37] E. Harding, "The probabilities of rooted tree-shapes generated by random bifurcation," *Advances in Applied Probability*, vol. 3, no. 1, pp. 44–77, 1971.
[38] M. Goodman, J. Czelusniak, G. Moore, A. Romero-Herrera, and G. Matsuda, "Fitting the gene lineage into its species lineage. A parsimony strategy illustrated by cladograms constructed from globin sequences," *Systematic Zoology*, vol. 28, no. 2, pp. 132–163, 1979.
[39] L. Zhang, "From gene trees to species trees ii: species tree inference by minimizing deep coalescence events," *IEEE/ACM Trans Comput Biol Bioinform*, vol. 8, no. 6, pp. 1685–91, 2011.
[40] B. R. Baum, "Combining trees as a way of combining data sets for phylogenetic inference, and the desirability of combining gene trees," *Taxon*, vol. 41, no. 1, pp. 3–10, 1992.
[41] M. A. Ragan, "Phylogenetic inference based on matrix representation of trees," *Molecular phylogenetics and evolution*, vol. 1, no. 1, pp. 53–58, 1992.
[42] O. Eulenstein, S. Huzurbazar, and D. Liberles, *Evolution after Gene Duplication*. John Wiley, 2010, ch. Reconciling Phylogenetic Trees.
[43] M. A. Steel, "The complexity of reconstructing trees from qualitative characters and subtrees," *Journal of Classification*, vol. 9, pp. 91–116, 1992.

Alexey Markin received a B.S. degree in Computer Science from Higher School of Economics (Russia) in 2015. Since then he is a Ph.D. student of Computer Science at Iowa State University, where he works with Prof. Eulenstein on computational problems in biology with focus on evolutionary tree inference. His research interests include graph theory, discrete mathematics, statistics, and phylogenetics.

Oliver Eulenstein is a professor of computer science at Iowa State University. He earned his doctoral degree at the University of Bonn (Germany) in 1998, and held a postdoctoral position at the University of California Davis before joining the department of Computer Science at Iowa State University in 2000. His research interest is in Combinatorial Optimization, with special emphasis on Computational Biology and Bioinformatics.