Quartet-based Genome-scale Species Tree Inference using Multicopy Gene Family Trees

Abdur Rafi[1,†], Ahmed Mahir Sultan Rumi[1,†], Sheikh Azizul Hakim[1], and Md. Shamsuzzoha Bayzid[1,*]

[1]Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
Dhaka-1205, Bangladesh
[†]These authors contributed equally to this work
[*]Corresponding author: shams_bayzid@cse.buet.ac.bd

April 2025

Abstract

Species tree estimation from genome-wide data has transformed evolutionary studies, particularly in the presence of gene tree discordance. Gene trees often differ from species trees due to factors like incomplete lineage sorting (ILS) and gene duplication and loss (GDL). Quartet-based species tree estimation methods have gained substantial popularity for their accuracy and statistical guarantee. However, most of these methods (e.g., ASTRAL, wQFM, wQMC) rely on single-copy gene trees and models ILS and not GDL, limiting their applicability to large genomic datasets. ASTRAL-Pro, a recent advancement, has refined quartet similarity measures to incorporate both orthology and paralogy, improving species tree inference under GDL. Among other quartet-based methods, wQFM-DISCO converts multicopy gene family trees into single-copy gene trees using DISCO and applies the wQFM algorithm on the single-copy trees. However, ASTRAL-Pro has remained the only quartet-based summary method to explicitly model gene duplication and loss. In this study, we extend both wQFM (which requires quartet decomposition) and wQFM-TREE (which operates directly on gene trees) by modeling gene duplication and loss, leveraging the concept of speciation-driven quartets introduced in ASTRAL-Pro. Our algorithm consistently outperforms ASTRAL-Pro across most model conditions, offering a promising alternative for species tree estimation in the presence of GDL.

1 Introduction

Species tree inference is a fundamental problem in evolutionary biology, complicated by the discordance between gene trees and species trees. This discordance arises due to several biological processes, including incomplete lineage sorting (ILS), gene duplication and loss (DupLoss), horizontal gene transfer, and hybridization (1). Although many approaches have been proposed for species tree inference, the most scalable and widely used approach so far follows a two-step process. At first, gene trees are inferred independently from sequence data. Then they are aggregated to infer the species tree using summary methods based on a model of gene tree discordance.

Summary methods construct species trees by first inferring gene trees from different loci and then integrating these gene trees into a single species tree. Many of these methods are statistically consistent under the multispecies coalescent (MSC) model (2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13). A key principle in the design of some statistically consistent approaches is the use of fundamental phylogenetic units, such as triplets in rooted analyses and quartets in unrooted analyses (3; 11; 14). ASTRAL, the most popular summary method, tries to solve the Maximum Quartet Support Species Tree (MQSST) problem (11). It takes a set of gene trees as input and seeks to find a species tree so that the number of induced quartets in the gene trees that are consistent with the species tree is maximized. While ASTRAL uses an exact dynamic programming approach, alternative heuristics such as QMC (Quartet MaxCut), QFM (Quartet Fiduccia–Mattheyses), wQMC, and wQFM address the problem by first inferring individual quartets and then constructing a species tree through a divide-and-conquer strategy (14; 15; 16; 17). Although wQFM has consistently outperformed ASTRAL in multiple studies (17; 18), its limited scalability has hindered widespread adoption. In a previous study, we introduced wQFM-TREE (19) to overcome the computational bottleneck of explicit quartet enumeration in the original wQFM algorithm. By directly traversing gene trees, wQFM-TREE significantly enhanced computational efficiency while preserving the core optimization framework of wQFM and maintaining its superior performance.

Gene Duplication and Loss (GDL) events introduce paralogs — genes that originate from a common ancestor through duplication events—in addition to orthologs, which arise from speciation events (20; 21). As a result, gene families may contain multiple genes with the same species labels, leading to multi-copy gene trees. However, most summary methods, including ASTRAL, wQFM, and wQFM-TREE, are designed to work with gene trees containing at most one copy of each gene per species, making them unsuitable for use with GDL. Several methods have been developed to estimate species trees from multi-copy gene trees while addressing gene duplications and losses, without the need for orthology determination. These methods include PHYLDOG (22), iGTP (23), DupTree (24), DynaDup (25; 26; 27), MulRF (28), FastMulRFS (29), Guenomu (30), ASTRAL-Pro (31), and SpeciesRax (32). Most of these methods (23; 24; 25; 26) are parsimony-based, estimating a species tree by minimizing a specific distance

between the species tree and the given gene trees, such as the total number of duplication and loss events. ASTRAL-Pro is the only quartet-based species tree inference method that explicitly models gene duplication and loss. It takes a set of multi-copy gene family trees and computes a species tree that maximizes similarity to the input gene trees, using a new quartet similarity measure that accounts for both orthology and paralogy (31). Alternative quartet-based methods like TREE-QMC can handle multi-labeled trees but do not incorporate gene duplication and loss modelling. As both wQFM and wQFM-TREE has shown greater accuracy compared to ASTRAL in the past (albeit with higher running time), in this study we extend these algorithms modelling gene duplication and loss. We make the following contributions in this study.

- We extend wQFM-TREE for paralogy using the gene duplication and loss model proposed in (31). We extend the concept of satisfied and violated quartets of the wQFM framework for this task and design graph algorithmic techniques to calculate them efficiently directly from the gene trees.
- We observe that for algorithms that need explicit quartet generation such as wQFM and wQMC, it is sufficient to generate the quartet set with appropriate quartets following the gene duplication and loss model. We devise an algorithm to generate the quartet set for multicopy gene family trees which can be fed to wQFM for analysis of multicopy dataset. We observe that, for smaller datasets (taxa less than 100) where explicitly generating quartets is feasible, wQFM outperforms both wQFM-TREE and ASTRAL-Pro.
- We test the algorithm extensively on simulated datasets with varying duplication and loss rates, ILS rates, number of taxa and genes. wQFM and wQFM-TREE demonstrate superior performance in most of the model conditions. Interestingly, the difference becomes more prominent with increasing number of taxa.

2 Materials and Methods

We begin with a brief overview of the algorithm. Then, we describe the major steps of the algorithm in detail.

2.1 Overview of wQFM and wQFM-TREE

wQFM (17) and its latest version, wQFM-TREE (19), are both quartet-based summary methods designed to construct a species tree with maximum quartet support from a given set of gene trees over a specified taxa set. wQFM is a divide-and-conquer algorithm where each divide step of the algorithm splits the set of taxa into two disjoint subsets (a bipartition of the taxa set, each representing a subproblem). A unique "dummy taxon" is added to both of the subsets so that the solutions to the subproblems can be combined within a

divide-and-conquer framework. The technique to find a bipartition begins with an initial bipartition, followed by a heuristic iterative strategy to improve it. Within this iterative process, each iteration begins with the bipartition from the previous iteration and tries to improve it by moving taxa from one partition to another. The iterative improvement strategy is described in detail in (14).

wQFM needs to generate all the quartets induced by the input gene trees which limits its ability to analyze large datasets. wQFM-TREE solves this major bottleneck by avoiding the quartet generation phase and performing all the steps of the algorithm directly from the input gene trees which enable it to analyze datasets with thousands of taxa and gene trees within hours. In particular, wQFM-TREE employs the following two key strategies: i) using a gene tree consensus-based method to compute an initial bipartition for each divide step, and ii) utilizing combinatorial and graph theoretic techniques to compute the scores of candidate bipartitions directly from gene trees, thereby eliminating the need for generating and computing the weights of the induced quartets required in original wQFM.

2.2 Overview of the Extension

We extend both wQFM and wQFM-TREE for handling multicopy inputs. We utilize the concepts for modelling gene duplication and loss proposed in (31) for ASTRAL-Pro which remain appropriate for wQFM as both of the algorithms are quartet-based summary methods. The key observations of (31) are:

- Internal nodes of a gene tree correspond to either duplication or speciation events. Thus, we can tag nodes of the trees as speciation or duplication.
- Each quartet of leaves in the gene tree defines two anchor nodes, and the Least Common Ancestor (LCA) of the two anchors is defined as the anchor LCA. In a correctly tagged tree, a quartet has information about the speciation events only if it includes four distinct species and if the LCA of any three out of four leaves of the quartet is a speciation node. Thus, to define our measure of quartet similarity, we only include these speciation-driven quartets (SQs) and ignore the rest.
- All the SQs on the same four species that share the same anchor LCA share the same topology. Thus, once the topology of one of them is known, the others do not provide new information. These are SQs equivalent and counted as one equivalence class.

However, we had to make significant changes to the overall process and heuristics used in both wQFM and wQFM-TREE to incorporate these concepts. For wQFM, we propose an algorithm that constructs a quartet set containing only speciation-driven quartets, with a single representative from each equivalent class. This curated quartet set can then be used as input for algorithms like wQFM or wQMC that require quartet enumeration. No internal modifications to the wQFM algorithm are required.

In case of wQFM-TREE, for the initial bipartition needed for the underlying FM algorithm, we utilize an updated version of the consensus tree-based method used in wQFM-TREE for computing the initial bipartition at each divide step using DISCO and PAUP. Moreover, we design new combinatorial techniques to calculate only the speciation-driven quartets and count all equivalent quartets as one unit efficiently and directly from the gene trees without quartet enumeration.

2.3 Modifications to wQFM-TREE

Before we describe the key modifications, it is quite essential to understand the weighting scheme for dummy taxa used in wQFM-TREE. The process is described in detail in (19). However, we provide a short overview of the weighting scheme in the following subsection.

2.3.1 Tree structure of a dummy taxon and weighting scheme

In our divide step, we create a bipartition (A, B) of the taxa set and create two new subproblems, one with taxa set $A \cup \{X_1\}$ and the other with taxa set $B \cup \{X_2\}$, where X_1 and X_2 are dummy taxa. X_1 essentially represents the taxa in B, and X_2 represents the taxa in A. We model X_1 and X_2 as rooted tree structures, where the children of the roots are the taxa in B and A, respectively. If A, B contain any dummy taxon, they will appear as subtrees in the tree structure of X_2, X_1 respectively, resulting in a recursive tree structure (Figure 1). Therefore, the tree representation of a dummy taxon X is a rooted tree with height > 1. The leaves of this tree are real taxa, and any internal node is a dummy taxon which was introduced in an earlier subproblem. We say that the leaves or real taxa in the tree structure of X are *under dummy taxon* X, and denote this set of real taxa under X as X_R.

Now we describe how we assign weight $w(a)$ to a taxon a that is used to compute the score of a candidate bipartition as described in Section 2.3.3. We assign unit weight to the real taxa that are not under any dummy taxa of the current subproblem. Note that the contribution of a dummy taxon to the score of a candidate bipartition should be equal to that of a real taxon as both can be considered equivalent in a subproblem. Therefore, for a dummy taxon X, we assign fractional weights to the real taxa in X_R in such a way that $w(X) = \sum_{a \in X_R} w(a) = 1$. We first assign weights to the edges of the tree structure corresponding to X. We then use these edge weights to calculate $w(a)$ $(a \in X_R)$. Refer to Figure 1 for an example. The weight of an edge (u, v) with u as the parent node, is defined as the number of children of u. Let p_a be the set of edges in the path from root to a. We define $w(a)$ as $(\prod_{(u,v) \in p_a} w_{(u,v)})^{-1}$. Therefore, for any real taxon a, $w(a)$ is defined as follows.

$$w(a) = \begin{cases} (\prod_{(u,v) \in p_a} w_{(u,v)})^{-1} & \text{if } a \text{ is under a dummy taxon} \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

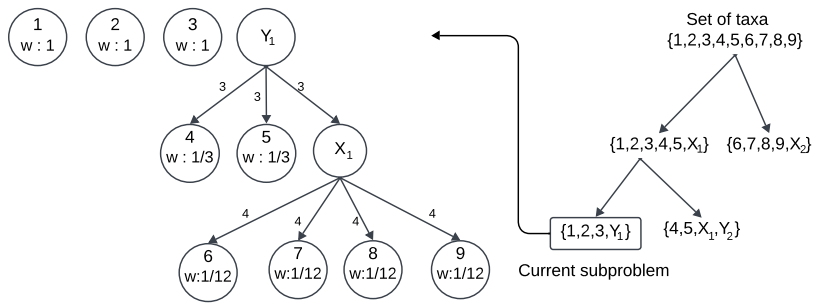

Figure 1: Tree structure of a dummy taxon, and the assignment of weights to real taxa under this dummy taxon. Here Y_1 is a dummy taxon that represents $\{4, 5, X_1\}$, where X_1 is another dummy taxon representing $\{6, 7, 8, 9\}$. The entire tree structure (root labeled by Y_1) is the representation of dummy taxon Y_1. The real taxa under Y_1 appear as leaves in the tree structure. Edge weights for each edge are shown. The weight of a leaf is the multiplicative inverse of the product of the edge weights from the root to the leaf. The figure is taken from (19)

2.3.2 Initial bipartition

In each divide step of our method, we initiate the process by generating an initial bipartition of the taxa set S, which is then iteratively refined. To guide this partitioning, we rely on a consensus tree. In wQFM-TREE, we previously used the PAUP package (33) to construct this consensus tree. However, since our current approach involves multicopy gene trees, PAUP cannot be applied directly. To address this, we first transform the multicopy gene trees into single-copy gene trees using DISCO (34; 35). With these single-copy gene trees, we then construct a greedy consensus tree using the Majority Rule Extended (MRE) method implemented in PAUP.

Each bipartition (A, B) corresponding to each edge of the consensus tree defines a bipartition (S_A, S_B) of S as follows. S may contain both real and dummy taxa. The real taxa in S are partitioned according to (A, B) meaning that if a taxon $s \in A$, then $s \in S_A$. We now describe how a dummy taxon X in S is assigned to a partition (S_A or S_B). We compute the weight of each taxon in X_R (the set of real taxa present in the tree structure of X) as described in Section 2.3.1. X_R can be partitioned to S_A and S_B according to (A, B) similar to how we partition the real taxa in S with respect to (A, B). Next, we assign X to the partition S_A if the sum of weights of the real taxa in X_R that belong to A is greater than the sum of weights of the real taxa in X_R that belong to B. Otherwise, we assign X to the partition S_B. Thus, we find a bipartition (S_A, S_B) of S with respect to (A, B) in the consensus tree. Next, we score

(S_A, S_B) using the scoring scheme described in Section 2.3.3. We find and score all such bipartitions of S corresponding to the bipartitons in the consensus tree. We choose the bipartition of S with the highest score as the initial bipartition.

2.3.3 Scoring a candidate bipartition

Let $\mathcal{G}$ be a set of rooted and tagged gene trees on taxa set $\mathcal{X}$. Let $\mathcal{X}^{(g)}$ be the set of real taxa in a gene tree $g \in \mathcal{G}$. Let (A, B) be the given bipartition. Let R_A and D_A be the sets of real and dummy taxa in A, respectively. Let F_A be the set of all real taxa that are either in R_A or under a dummy taxon $X \in D_A$, i.e., $F_A = (\bigcup_{X \in D_A} X_R) \cup R_A$. We similarly define R_B, D_B, and F_B .

Given a bipartition (A, B), each quartet $ab|cd$, where no two taxa in $\{a, b, c, d\}$ are under the same dummy taxon of the current subproblem, can be classified as one of the following three types.

1. **Satisfied:** where either $\{a, b\} \subseteq F_A$ and $\{c, d\} \subseteq F_B$ or $\{a, b\} \subseteq F_B$ and $\{c, d\} \subseteq F_A$
2. **Deferred:** where there exists some $P \subseteq \{a, b, c, d\}$ such that $P \subseteq F_A$ or $P \subseteq F_B$, and $|P| \geq 3$.
3. **Violated:** all other quartets.

Let $S^{(g)}$ and $V^{(g)}$ denote the sets of satisfied and violated speciation-driven quartets, respectively, each containing one representative from every equivalence class in a gene tree $g \in \mathcal{G}$. Note that all quartets within the same equivalence class have equal weights. The score $\text{Score}(A, B, \mathcal{G})$ of a candidate bipartition (A, B) with respect to $\mathcal{G}$ is then defined as $\text{Score}(A, B, \mathcal{G}) = \sum_{g \in \mathcal{G}} \big(w(S^{(g)}) - w(V^{(g)})\big)$.

Computing $w(S^{(g)})$, the sum of weights of satisfied speciation-driven quartets in a gene tree g. The main idea is that we can efficiently compute the sum of weights of all speciation-driven quartets with the same anchor LCA. It is essential to ensure that only speciation-driven quartets are included in the computation and that each equivalence class contributes only a single representative quartet.

The subtree of a fully resolved gene tree, when restricted to a quartet, contains two degree-three nodes. We refer to these nodes as *anchors*. As illustrated in Figure 1a, rooted quartets can have two distinct structural forms. The first, known as the **balanced structure**, consists of two pairs of taxa that are grouped together first, with a root connecting these two pairs. The second, called the **unbalanced structure**, forms a nested grouping where taxa are added sequentially at progressively deeper levels. To compute $w(S^{(g)})$, we divide the calculation into two parts. The weights of the balanced and unbalanced speciation-driven quartets are computed separately and then summed to obtain the final value.

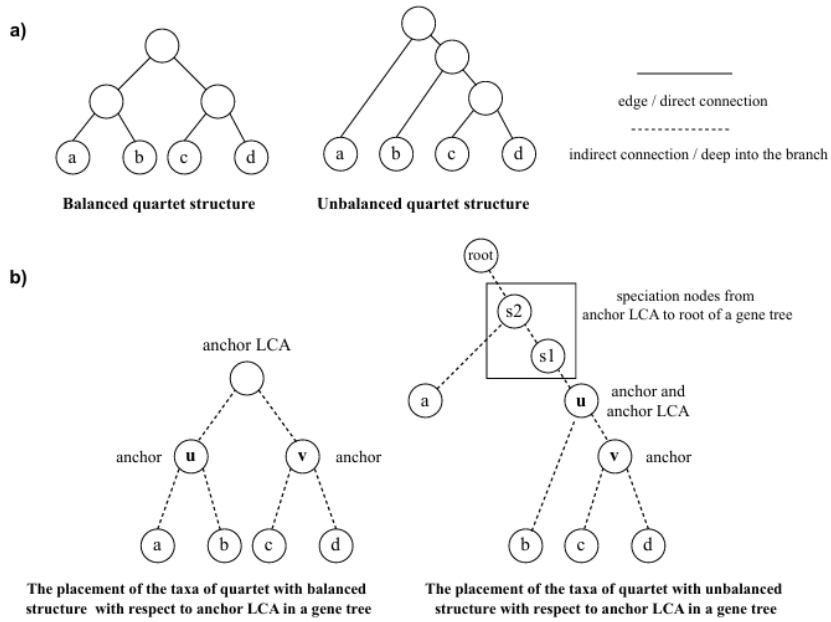

Figure 2: Calculating sum of weights of balanced and unbalanced quartets for scoring a candidate bipartiton.

Let u be an internal speciation node in g, with two child branches i and j. We define $PA_i^{(g,u)}$ as the set of unordered pairs $\{a, b\}$, where a and b belong to branch i, are not grouped under the same dummy taxon, and satisfy $a, b \in F_A$. Note that each taxon pair is considered only once, even if multiple copies of a taxon exist in the branch. Similarly, we define $PA_j^{(g,u)}$ for branch j. The sets $PB_i^{(g,u)}$ and $PB_j^{(g,u)}$ are defined analogously.

Now, for the quartets with the balanced structure with anchor LCA u, one of their anchors will be in branch i and the other in branch j, as shown in Figure 1b. This allows us to efficiently compute $w(S_1^{(g)})$, the total weight of quartets with a balanced structure.

$$w(S_1^{(g,u)}) = w(PA_i^{(g,u)})w(PB_j^{(g,u)}) + w(PA_j^{(g,u)})w(PB_i^{(g,u)})$$
$$w(S_1^{(g)}) = \sum_u w(S_1^{(g,u)})$$

For the unbalanced ones, one anchor will be present in either branch i or j. The other anchor is u itself. We calculate the weights of all possible pairs that can eventually create an anchor in one of the child branches. Both taxa must

belong to partition A or B, forming the set $PA_i^{(g,u)}$ or $PB_i^{(g,u)}$ for branch i, and $PA_j^{(g,u)}$ or $PB_j^{(g,u)}$ for branch j. This is because of the condition that for a speciation node, any three of the four leaves of a quartet must be a speciation node.

Now, let $PA_{i,k}^{(g,u)}$ be the set of unordered pairs $\{a, b\}$ where a belongs to branch i, b belongs to the parent branch k (and is a descendant of a speciation node from u to the root), both are not grouped under the same dummy taxon and satisfy $a, b \in F_A$. $PB_{i,k}^{(g,u)}$ is defined similarly.

From this, we can compute

$$\begin{aligned} w(S_2^{(g)}) &= w(PA_i^{(g,u)})w(PB_{j,k}^{(g,u)}) \\ &\quad + w(PA_j^{(g,u)})w(PB_{i,k}^{(g,u)}) \\ &\quad + w(PB_i^{(g,u)})w(PA_{j,k}^{(g,u)}) \\ &\quad + w(PB_j^{(g,u)})w(PA_{i,k}^{(g,u)}) \\ w(S_2^{(g)}) &= \sum_u w(S_2^{(g,u)}) \end{aligned}$$

The expressions such as $w(PA_i^{(g,u)})$, $w(PA_{i,j}^{(g,u)})$ are calculated efficiently using graph algorithms like BFS and DFS in a single pre-order or post-order traversal.

Finally,

$$w(S^{(g)}) = w(S_1^{(g)}) + w(S_2^{(g)})$$

Computing $w(V^{(g)})$, the sum of weights of violated speciation-driven quartets in a gene tree g The calculation of $w(V^{(g)})$ follows a similar approach. We compute the total weight of the violated quartets separately for balanced and unbalanced structures and then sum the results, following the same approach as before.

Again, we assume an internal speciation node u in g. It contains two child branches i and j. For the balanced violated quartets, each branch will contain a pair (one from partition A, the other from B) and an associated anchor node.

In other words, the pair of taxa from a partition are on separate branches, with one in branch i and the other in branch j. Let $PA_{i,j}^{(g,u)}$ be the set of unordered pairs $\{a, b\}$ where a belongs to branch i, b belongs to branch j, both are not grouped under the same dummy taxon and satisfy $a, b \in F_A$. $PB_{i,j}^{(g,u)}$ is similarly defined. Thus,

$$\begin{aligned} w(V_1^{(g,u)}) &= w(PA_{i,j}^{(g,u)})w(PB_{i,j}^{(g,u)}) \\ w(V_1^{(g)}) &= \sum_u w(V_1^{(g,u)}) \end{aligned}$$

For the unbalanced ones, using similar reasoning, we get

$$
\begin{aligned}
w(V_2^{(g)}) = {} & w(PA_{i,k}^{(g,u)})w(PB_{i,j}^{(g,u)}) \\
& + w(PA_{i,j}^{(g,u)})w(PB_{i,k}^{(g,u)}) \\
& + w(PA_{j,k}^{(g,u)})w(PB_{j,i}^{(g,u)}) \\
& + w(PA_{j,i}^{(g,u)})w(PB_{j,k}^{(g,u)}) \\
w(V_2^{(g)}) = {} & \sum_u w(V_2^{(g,u)})
\end{aligned}
$$

$$
w(V^{(g)}) = w(V_1^{(g)}) + w(V_2^{(g)})
$$

2.4 Modifications to wQFM

wQFM takes as input a set of weighted quartets. Here, the weight is the count of the quartet structure in the gene trees. Our main task is to create an appropriate quartet set that only contains speciation-driven quartets and only one representative of an equivalent class. The main wQFM algorithm remain unchanged. Again, we use a similar approach of considering balanced and unbalanced quartet structures separately. However, in this case, we are not calculating the sum of weights of quartets; we have to enumerate them with proper weights. Moreover, there is no notion of satisfied or violated quartets as there is no bipartition to consider in the beginning.

For each speciation node, we generate the speciation-driven quartets with anchor LCA at that node. For the balanced quartets, we enumerate the possible pairs coming form each child branch. Of course, we keep only one copy of a taxa pair as they are equivalent. Then, each of the pairs can create a valid quartet with any pair from the other branch. We generate all possible quartets by joining the pairs.

For the unbalanced quartets, One pair will come from a child branch. We generate all possible pairs for both child branches. For the other pair, one taxa will be present in the other child branch and one taxa in the parent branch. As discussed, not all taxa in the parent branch are considered here (Figure 1b). We list all taxa present in the child branches and the parent branch and merge them to create pairs. The pairs are then merged appropriately to enumerate quartets.

An important point to note is that we just generate the appropriate quartet file for wQFM to work on. The main wQFM algorithm remains the same. The algorithm for generating the quartet file is shown in Algorithm 1.

Algorithm 1: Algorithm for quartet decomposition for wQFM for multicopy dataset

```
Input  : 𝒢, set of tagged and rooted multicopy gene family trees
Output: The set of speciation-driven quartets in 𝒢 with only one
        representative for each equivalent class

Q ← ∅ ;                                   // quartet set initialization
foreach gene tree g ∈G do
    foreach internal speciation node s in g do
        l ← left child of s
        r ← right child of s

        qleft ← getRealTaxainSubtree(l) // list all taxa present in
           the subtree
        qright ← getRealTaxainSubtree(r)
        pairs_left ← getAllPairs(qleft) // generate all possible taxa
           pairs from the taxa list
        pairs_right ← getAllPairs(qright)
        Q1 ← generateQuartetsfromPairs(pairs_left,pairs_right)
        /* generate all valid quratets (pair of pairs) from the two list
           of pairs                                                  */

        par ← getRealTaxainParents(s)
        /* list all the taxa present in the other child subtree of all
           the speciation nodes on the path from s to root           */
        crosspairs_left ← getAllCrossPairs(par,qleft) // generate all
           possible pairs with one taxa from each of the taxa lists
        crosspairs_right ← getAllCrossPairs(par,qright)
        Q2 ← generateQuartetsfromPairs(pairs_left,crosspairs_right)
         ∪generateQuartetsfromPairs(crosspairs_left,pairs_right)

        Q ← Q1 ∪ Q2
    end
end
return Q
```

```
Function getRealTaxainParents(s)
    taxa_list ← ∅
    current ← s
    while current.parent ≠ null do
        if current.parent is speciation node then
            otherChild ← getSibling(current)
            taxa_list ← taxa_list ∪ getRealTaxainSubtree(otherChild)
        end
        current ← current.parent
    end
    return taxa_list
```

3 Experimental Studies

3.1 Dataset

We test the updated wQFM and wQFM-TREE against ASTRAL-Pro. In our tests, we use a simulated dataset, called S25 simulated and analyzed in (36). the data set was created by simulating true gene trees under the DLCoal model, simulating a sequence alignment from each true gene family trees, and finally estimating a gene tree from each gene alignment. For 25 taxa, a large number of model conditions were generated, varying parameters that control the rate of duplication, the rate of loss, the ILS level, and the number of genes. The number of replicates in each model condition is 50. Each replicate contains 1000 gene trees. Moreover, model conditions with 100, 250, and 500 taxa were generated in the toughest model condition (high rate of duplication and loss, high level of ILS) to test the scalability of the methods.

3.2 Results

3.2.1 Controlling Duplication and Loss Rates

We present results for model conditions varying duplication and loss rates for estimated gene trees with 100 bp and 500 bp alignments. As shown in Figure 3, wQFM consistently outperforms both ASTRAL and wQFM-TREE in most model conditions. Notably, as the duplication rate increases, all methods improve in performance which is consistent with the findings of (31). This is likely because additional gene copies provide more information, similar to increasing the number of genes. Both wQFM and ASTRAL exhibit similar trends. Although error rates increase with higher loss rates, the increase remains under 2% for wQFM. As expected, longer alignments (500 bp) lead to lower error rates.

Overall, across 32 model conditions, wQFM outperforms ASTRAL in 21 cases (with most differences being statistically significant) and ties in 4 cases. For the 25-taxon dataset, wQFM-TREE shows slightly lower accuracy compared to wQFM and ASTRAL. While wQFM and wQFM-TREE follow the same

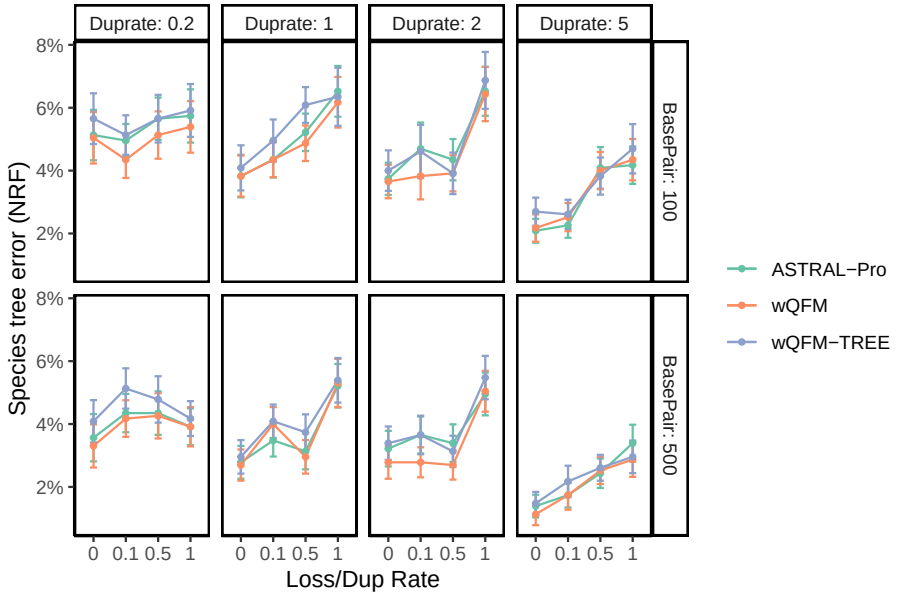

Figure 3: Species tree error on the S25 dataset for 25 taxa, 1000 gene trees, and estimated gene trees from 100 and 500 base pair alignments with varying duplication rate

core strategy, wQFM-TREE introduces heuristic approaches to make wQFM applicable for large datasets—such as constructing consensus tree from single-copy gene trees decomposed from multicopy gene families using DISCO. This may contribute to a slight decrease in accuracy.We recommend using wQFM for small datasets. However, as we will demonstrate, for large datasets, wQFM-TREE significantly outperforms ASTRAL-Pro.

3.2.2 Controlling ILS level

Naturally, as the ILS level increases, the performance of all the methods decrase significantly as observed in Figure 4. Again, wQFM outperforms other methods. However, the performance improvements remain relatively modest. Out of 24 model conditions, wQFM outperforms ASTRAL-Pro in 17 model conditions (with most differences being statistically significant) and ties in 1 condition. wQFM-TREE demonstrates slightly lower accuracy.

3.2.3 Controlling the number of species

Increasing taxa count from up to 500 in the most difficult model condition (high duplication and loss rate, high ILS) with estimated gene trees (for 100 bp) shows that both wQFM-TREE and ASTRAL are robust to a large number of species. An important point to note is that as wQFM requires quartet decomposition and

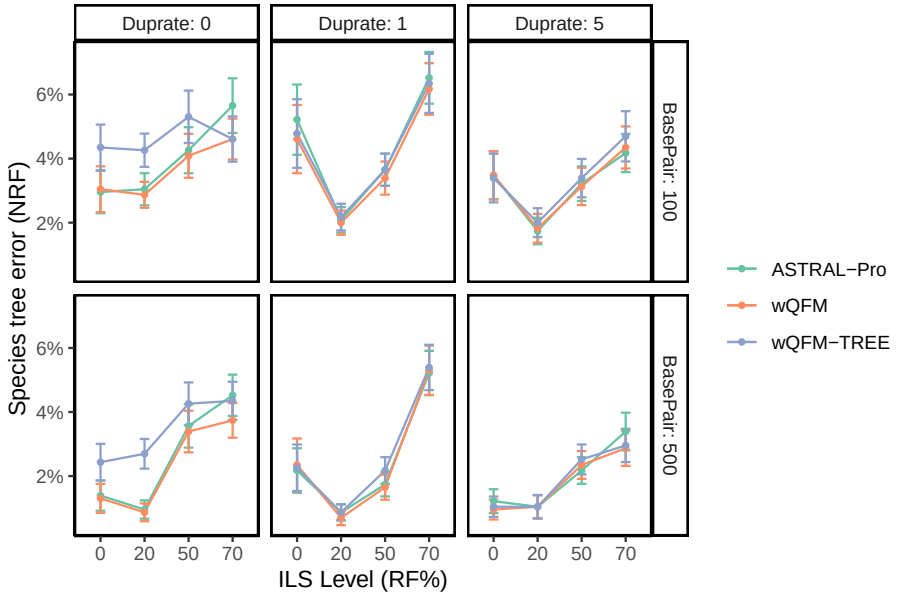

Figure 4: Species tree error on the S25 dataset for 25 taxa, 1000 gene trees, and estimated gene trees from 100 and 500 base pair alignments with varying ILS level

it cannot analyze model conditions with more than 100 species. Thus, we only use wQFM-TREE for this experiment. wQFM-TREE shows significantly better performance in case of larger taxa. Specially for 500 taxon dataset, wQFM-TREE constructs species trees with an average RF rate of 6.8% compared to 9.5% of ASTRAL-Pro.

4 Conclusion

In this study, we extend the algorithms wQFM and wQFM-TREE for paralogy. Both of the algorithms demonstrate excellent performance and outperform ASTRAL-Pro in most of the model conditions. Although wQFM requires quartet decomposition which is time-consuming, it performs slightly better than wQFM-TREE and we suggest using wQFM when it is feasible to generate quartet-list from gene trees. ASTRAL-Pro was the only quartet-based summary method to incorporate gene duplication and loss modeling to date. This study positions wQFM as a strong alternative to ASTRAL-Pro, demonstrating superior accuracy at the expense of increased computational cost.

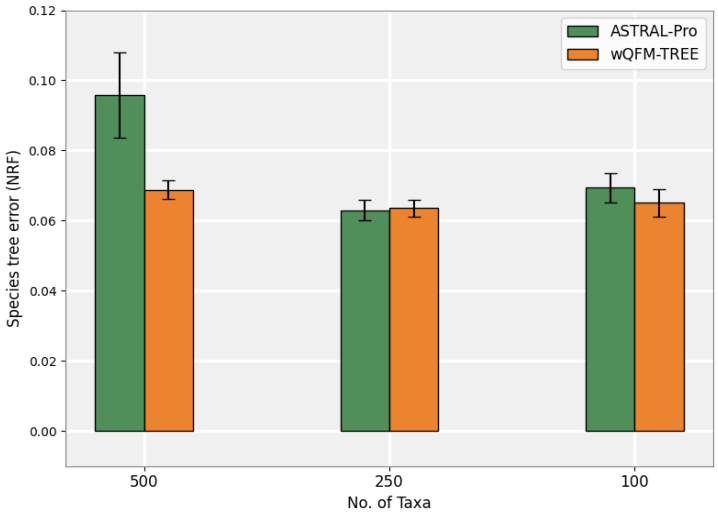

Figure 5: Comparison of methods on varying taxa with fixed ILS level and Loss/Dup rate.

References

[1] W. P. Maddison, "Gene trees in species trees," *Syst Biol*, vol. 46, pp. 523–536, 1997.

[2] D. Bryant, R. Bouckaert, J. Felsenstein, N. A. Rosenberg, and A. Roy-Choudhury, "Inferring species trees directly from biallelic genetic markers: bypassing gene trees in a full coalescent analysis," *Molecular biology and evolution*, vol. 29, no. 8, pp. 1917–1932, 2012.

[3] J. Chifman and L. Kubatko, "Quartet from snp data under the coalescent model," *Bioinformatics*, vol. 30, no. 23, pp. 3317–3324, 2014.

[4] J. Heled and A. J. Drummond, "Bayesian inference of species trees from multilocus data," *Mol Biol Evol*, vol. 27, pp. 570–580, 2010.

[5] M. Islam, K. Sarker, T. Das, R. Reaz, and M. S. Bayzid, "Stelar: A statistically consistent coalescent-based species tree estimation method by maximizing triplet consistency," *BMC genomics*, vol. 21, no. 1, pp. 1–13, 2020.

[6] L. S. Kubatko, B. C. Carstens, and L. L. Knowles, "Stem: Species tree estimation using maximum likelihood for gene trees under coalescence," *Bioinf*, vol. 25, pp. 971–973, 2009.

[7] B. Larget, S. K. Kotha, C. N. Dewey, and C. Ané, "BUCKy: Gene tree/species tree reconciliation with the Bayesian concordance analysis," *Bioinf*, vol. 26, no. 22, pp. 2910–2911, 2010.

[8] L. Liu, L. Yu, D. K. Pearl, and S. V. Edwards, "Estimating species phylogenies using coalescence times among sequences," *Systematic biology*, vol. 58, no. 5, pp. 468–477, 2009.

[9] L. Liu, L. Yu, and S. V. Edwards, "A maximum pseudo-likelihood approach for estimating species trees under the coalescent model," *BMC Evolutinary Biology*, vol. 10:302, 2010.

[10] L. Liu and L. Yu, "Estimating species trees from unrooted gene trees," *Systematic Biology*, vol. 60, no. 5, pp. 661–667, 2011.

[11] S. Mirarab, R. Reaz, M. S. Bayzid, T. Zimmermann, M. S. Swenson, and T. Warnow, "ASTRAL: genome-scale coalescent-based species tree estimation," *Bioinformatics*, vol. 30, no. 17, pp. i541–i548, 2014.

[12] E. Mossel and S. Roch, "Incomplete lineage sorting: consistent phylogeny estimation from multiple loci," *IEEE Comp Biol Bioinform*, vol. 7, no. 1, pp. 166–171, 2011.

[13] P. Vachaspati and T. Warnow, "Astrid: accurate species trees from internode distances," *BMC genomics*, vol. 16, no. 10, p. S3, 2015.

[14] R. Reaz, M. S. Bayzid, and M. S. Rahman, "Accurate phylogenetic tree reconstruction from quartets: A heuristic approach," *PLoS One*, vol. 9, no. 8, p. e104008, 2014.

[15] S. Snir and S. Rao, "Quartet maxcut: a fast algorithm for amalgamating quartet trees," *Molecular phylogenetics and evolution*, vol. 62, no. 1, pp. 1–8, 2012.

[16] E. Avni, R. Cohen, and S. Snir, "Weighted quartets phylogenetics," *Systematic biology*, vol. 64, no. 2, pp. 233–242, 2015.

[17] M. Mahbub, Z. Wahab, R. Reaz, M. S. Rahman, and M. S. Bayzid, "wQFM: highly accurate genome-scale species tree estimation from weighted quartets," *Bioinformatics*, vol. 37, pp. 3734–3743, Nov. 2021.

[18] S. Mahbub, S. Sawmya, A. Saha, R. Reaz, M. S. Rahman, and Md. S. Bayzid, "Quartet Based Gene Tree Imputation Using Deep Learning Improves Phylogenomic Analyses Despite Missing Data," *Journal of Computational Biology*, vol. 29, pp. 1156–1172, Nov. 2022.

[19] A. Rafi, A. M. S. Rumi, S. A. Hakim, Sohaib, M. T. Tahmid, R. J. I. Momin, T. A. Zaman, R. Reaz, and M. S. Bayzid, "wqfm-tree: highly accurate and scalable quartet-based species tree inference from gene trees," *Bioinformatics Advances*, vol. 5, no. 1, p. vbaf053, 2025.

[20] W. M. Fitch, "Homology: a personal view on some of the problems," *Trends in genetics*, vol. 16, no. 5, pp. 227–231, 2000.

[21] Y. Yang and S. A. Smith, "Orthology inference in nonmodel organisms using transcriptomes and low-coverage genomes: improving accuracy and matrix occupancy for phylogenomics," *Molecular biology and evolution*, vol. 31, no. 11, pp. 3081–3092, 2014.

[22] B. Boussau, G. J. Szöllősi, L. Duret, M. Gouy, E. Tannier, and V. Daubin, "Genome-scale coestimation of species and gene trees," *Genome research*, vol. 23, no. 2, pp. 323–330, 2013.

[23] 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*, pp. 574–574, 2010.

[24] A. Wehe, M. S. Bansal, J. G. Burleigh, and O. Eulenstein, "Duptree: A program for large-scale phylogenetic analyses using gene tree parsimony," *American Journal of Botany*, vol. 24, no. 13, pp. 1540–1541, 2008.

[25] M. S. Bayzid, S. Mirarab, and T. Warnow, "Inferring optimal species trees under gene duplication and loss," in *Proc. of Pacific Symposium on Biocomputing (PSB)*, vol. 18, pp. 250–261, 2013.

[26] M. S. Bayzid and T. Warnow, "Gene tree parsimony for incomplete gene trees: addressing true biological loss," *Algorithms for Molecular Biology*, vol. 13, p. 1, 2018.

[27] M. S. Bayzid, "Inferring optimal species trees in the presence of gene duplication and loss: Beyond rooted gene trees," *Journal of Computational Biology*, vol. 30, no. 2, pp. 161–175, 2023.

[28] R. Chaudhary, J. G. Burleigh, and D. Fernández-Baca, "Inferring species trees from incongruent multi-copy gene trees using the Robinson-Foulds distance," *Algorithms for Molecular Biology*, vol. 8, pp. 1–12, Dec. 2013.

[29] E. K. Molloy and T. Warnow, "FastMulRFS: fast and accurate species tree estimation under generic gene duplication and loss models," *Bioinformatics*, vol. 36, pp. i57–i65, July 2020.

[30] L. De Oliveira Martins, D. Mallo, and D. Posada, "A bayesian supertree model for genome-wide species tree reconstruction," *Systematic biology*, vol. 65, no. 3, pp. 397–416, 2016.

[31] C. Zhang, C. Scornavacca, E. K. Molloy, and S. Mirarab, "ASTRAL-Pro: Quartet-Based Species-Tree Inference despite Paralogy," *Molecular Biology and Evolution*, vol. 37, pp. 3292–3307, Nov. 2020.

[32] B. Morel, P. Schade, S. Lutteropp, T. A. Williams, G. J. Szöllősi, and A. Stamatakis, "Speciesrax: a tool for maximum likelihood species tree inference from gene family trees under duplication, transfer, and loss," *Molecular biology and evolution*, vol. 39, no. 2, p. msab365, 2022.

[33] D. Swofford, *PAUP*: Phylogenetic analysis using parsimony (* and other methods). Ver. 4.* Sinauer Associates, Sunderland, Massachusetts, 2002.

[34] J. Willson, M. S. Roddur, B. Liu, P. Zaharias, and T. Warnow, "DISCO: Species Tree Inference using Multicopy Gene Family Tree Decomposition," *Systematic Biology*, vol. 71, pp. 610–629, 08 2021.

[35] S. A. Hakim, M. R. Z. Ratul, and M. S. Bayzid, "wQFM-DISCO: DISCO-enabled wQFM improves phylogenomic analyses despite the presence of paralogs," *Bioinformatics Advances*, vol. 4, p. vbae189, 11 2024.

[36] C. Zhang, C. Scornavacca, E. K. Molloy, and S. Mirarab, "Astral-pro: quartet-based species-tree inference despite paralogy," *Molecular biology and evolution*, vol. 37, no. 11, pp. 3292–3307, 2020.