Simultaneous Reconstruction of Duplication Episodes and Gene-Species Mappings

Paweł Górecki
Faculty of Mathematics, Informatics and Mechanics, University of Warsaw, Poland

Natalia Rutecka
Faculty of Mathematics, Informatics and Mechanics, University of Warsaw, Poland

Agnieszka Mykowiecka
Faculty of Mathematics, Informatics and Mechanics, University of Warsaw, Poland

Jarosław Paszek
Faculty of Mathematics, Informatics and Mechanics, University of Warsaw, Poland

Abstract

We present a novel problem, called MetaEC, which aims to infer gene-species assignments in a collection of gene trees with missing labels by minimizing the size of duplication episode clustering (EC). This problem is particularly relevant in metagenomics, where incomplete data often poses a challenge in the accurate reconstruction of gene histories. To solve MetaEC, we propose a polynomial time dynamic programming (DP) formulation that verifies the existence of a set of duplication episodes from a predefined set of episode candidates. We then demonstrate how to use DP to design an algorithm that solves MetaEC. Although the algorithm is exponential in the worst case, we introduce a heuristic modification of the algorithm that provides a solution with the knowledge that it is exact. To evaluate our method, we perform two computational experiments on simulated and empirical data containing whole genome duplication events, showing that our algorithm is able to accurately infer the corresponding events.

2012 ACM Subject Classification Mathematics of computing → Combinatorial optimization; Applied computing → Computational genomics

Keywords and phrases Genomic Duplication, Gene-Species Mapping, Duplication Episode, Gene Tree, Species Tree

Digital Object Identifier 10.4230/LIPIcs.WABI.2023.6

Supplementary Material *Software (Source Code and Datasets)*: `https://bitbucket.org/pgor17/metaec/`; archived at `swh:1:dir:c2cc43b86e0954bda7806f5d7007188ca33938b5`

Funding *Paweł Górecki*: National Science Centre grant nr 2017/27/B/ST6/02720.
Natalia Rutecka: National Science Centre grant nr 2017/27/B/ST6/02720.
Agnieszka Mykowiecka: National Science Centre grant nr 2017/27/B/ST6/02720.
Jarosław Paszek: National Science Centre grant nr 2020/39/D/ST6/03321.

1 Introduction

In the field of computational biology, the use of gene families and the reconciliation model has become increasingly popular for studying the evolution of diverse organisms. These tools have facilitated the development of new algorithms and computational methods capable of handling large and complex datasets and exploring various types of evolutionary events. These events range from simple macroevolutionary processes such as gene duplications, gene losses, and horizontal gene transfers to more complex ones such as genomic duplications, speciations, and hybridizations. The reconciliation model has enabled researchers to reconstruct evolutionary histories by reconciling gene trees with species trees, and identifying the evolutionary events that have led to the observed patterns of gene evolution. In the context of metagenomics,

23rd International Workshop on Algorithms in Bioinformatics (WABI 2023).
Editors: Djamal Belazzougui and Aïda Ouangraoua; Article No. 6; pp. 6:1–6:18
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

the reconciliation model has also been used to detect missing gene-species assignments using polynomial time algorithms. These developments have led to a better understanding of the evolutionary processes.

A classical reconciliation model [10, 26] defines a mapping from every node from a gene tree into a node in the species tree and determines if such a node is related to a speciation or can be classified as a single gene duplication [19]. In result, an embedding of the set of gene trees into a species tree can be interpreted as a joint evolutionary scenario [11]. The classical least common ancestor (LCA) mapping minimizes the number of single gene duplications and losses for one gene tree and the species tree [11].

The whole genome duplication (WGD) phenomenon incorporates additional copies of a complete genome into the original genetic material, thus creating an opportunity to introduce novel evolutionary traits [16, 25]. From a macro perspective, this phenomenon played a crucial role in the divergence and formation of species and shaped the evolution of almost all major lineages of life. In particular, many WGDs were uncovered in the evolutionary histories of plants, especially crops. WDGs potentially enabled the successful domestication of plants [36] and are important in the fight against famine [40]. Many traces and evidence of whole genome duplications can be found in the genomes of yeast and other fungal species [16, 39]. From the perspective of single cell evolution studies, WGDs are prevalent in cancer progression [18] and can lead us to the prognosis of advanced cancer stages [3] or the creation of strategies for targeted therapy [33].

Guigó et al. [13] proposed the first approach for detecting multiple gene duplication episodes from a collection of rooted gene trees. They designed a heuristic that aggregates single gene duplication events into a large gene duplication, given a collection of rooted gene trees and a rooted species tree. This approach was formalized and improved by Page and Cotton [27], who defined the problem of *episode clustering* (EC) as the task of identifying the minimal number of locations in the species tree where all duplications from the input gene trees can be placed. Fellows [8] applied this model in the context of the supertree problem. Polynomial-time solutions for two types of multiple gene duplication problems episode clustering and a more general variant of clustering called *minimum episodes* (ME) were proposed in [1, 4]. Luo et al. [17] proposed linear time and space algorithms, partially based on [22], for these problems. [29] introduced a unified approach by proposing a concept of interval models with a linear time and space solution to a broad class of clustering problems including EC and ME. Alternative approaches include generalization to unrooted gene trees; however, such approaches are often computationally complex [28, 30]. Other approaches include variants of clustering rules that depend on the maximal number of duplication episodes placed in one path [15, 32]. A comprehensive analysis of various models is available in [29]. Furthermore, [31] proposes an integer linear programming formulation that simplifies the process of testing these models. Relevant computational complexity results on the ME problem are presented in [6].

Metagenomic studies provide valuable information for analyzing entire communities of organisms and revealing a complete picture of their functional and adaptive capacities crucial for ecology [35] or human health [38]. Genetic material isolated in such studies can be used to detect whole genome duplication events.

One of the steps in metagenomic analysis is called binning. The aim of this procedure is to assign sequenced DNA fragments to the appropriate taxonomic groups. The assignment of certain genes to species can be ambiguous due to the limitations of annotation methods. A precise and comprehensive gene tree topology is essential for the accurate identification of potential duplication sites. The absence or misplacement of duplications in gene trees can, in turn, result in incorrect outcomes of methods aimed at determining whole genome duplication events.

To tackle the challenge of missing gene-species assignments in evolutionary studies, previous research has introduced methods based on the reconciliation score using gene duplication and loss events [2, 42]. In a related work, Mykowiecka et al. [24] extended this model by including horizontal gene transfer to better analyze bacterial evolution and proposed polynomial time algorithms for these models. These approaches utilize tree reconciliation according to the classical scheme, in which the gene tree includes symbols representing sequences with unclear species assignment in addition to the known gene labels. The objective is to assign the unknown gene labels to their corresponding species by resolving the missing labels in a gene tree while minimizing the total reconciliation score, which is typically a weighted sum of evolutionary events such as gene duplication, gene loss, and horizontal gene transfer.

Here, we present a novel problem, called MetaEC, which aims to infer gene-species assignments in a collection of gene trees with missing labels by minimizing the size of episode clustering. This problem is particularly relevant in metagenomics, where incomplete data often poses a challenge in the accurate reconstruction of gene histories. To solve MetaEC, we propose a dynamic programming (DP) algorithm that verifies the existence of a set of duplication episodes from a predefined set of episode candidates. We then demonstrate how to use DP to design an algorithm that solves MetaEC. Although the algorithm is exponential in the worst case, we introduce a heuristic modification of the algorithm that provides a solution with the knowledge that it is exact. To evaluate our method, we perform two computational experiments on simulated and empirical data containing WGD events, showing that our algorithm is able to accurately infer the corresponding events.

2 Basic Definitions

2.1 Gene trees, species trees, and the duplication cost

We begin by recalling some basic definitions from graph theory. All trees in this article are rooted and binary, therefore we refer to them as *trees*. For a tree $T = (V(T), E(T))$, by $\mathsf{root}(T)$ we denote the root, and by $L(T)$ we denote the set of all leaves. Every non-leaf node will be called *internal*. A *species tree* is a tree whose leaves are called *species*. A *gene tree* over a species tree S is a tree with leaves labeled by the species from S. The set of all species present in a species tree or a gene tree T is denoted by $\mathcal{L}(T)$. Note that for a species tree S, $L(S) = \mathcal{L}(S)$. Also, for a gene tree G over S, $\mathcal{L}(G) \subseteq L(S)$.

For nodes a and b, $a \preceq b$ means that a and b are on the same path from the root, with b being closer to the root than a. We write $a \prec b$ if $a \preceq b$ and $a \neq b$.

For a gene tree G over a species tree S, *the least common ancestor (lca) mapping* between G and S is a function $\mathsf{M}_G \colon V(G) \to V(S)$ defined as follows. If v is a leaf in G then $\mathsf{M}_G(v)$ is the label of v. When v is an internal node in T having two children a and b, then $\mathsf{M}_G(v)$ is the least common ancestor of $\mathsf{M}_G(a)$ and $\mathsf{M}_G(b)$ in S. An internal node $g \in V(G)$ is called a *duplication* if $\mathsf{M}_G(g) = \mathsf{M}_G(a)$ for a child a of g. *The duplication cost*, denoted by $\mathsf{D}(G, S)$, is the total number of duplications in G. Each non-duplication internal node of G we call a *speciation*. In the latter part of the article, a duplication g is called an s-duplication if $\mathsf{M}_G(g) = s$. Similarly, we use the notation for an s-speciation and an s-leaf. An example of tree reconciliation and the lca-mapping is depicted in the leftmost part of Figure 1.

2.2 Episode Clustering Problems

Below we present a model of duplication episodes proposed in [29]. In short, this model admits all evolutionary scenarios using duplication and loss events with a minimal number of gene duplications.

Formally, the model of gene duplication episodes allows for relocating a gene duplication from its lca-mapping node to one of its ancestors under some additional constraints required to preserve the biological soundness of the scenario. For a gene tree G over S, a mapping $\mathsf{F}_G \colon V(G) \to V(S)$ is called *valid* if the following conditions are satisfied:

- $\mathsf{F}_G(a) \preceq \mathsf{F}_G(b)$ if $a \preceq b$ (time consistency),
- $\mathsf{F}_G(a) = \mathsf{M}_G(a)$ for any speciation node a (fixed speciations),
- $\mathsf{F}_G(a) \succeq \mathsf{M}_G(a)$ for any duplication node a (duplication can be raised),
- $\mathsf{F}_G(a) \prec \mathsf{M}_G(b)$ for any speciation node b such that $a \prec b$ (fixed number of gene duplications).

Note that the model of valid mappings described above is more comprehensive than the model presented in [1].

Figure 1 provides an example of valid mappings that uniquely define an evolutionary scenario that can be represented as a tree with an additional decoration of nodes. For more information on the formal modeling of evolutionary scenarios, refer to [11].

We denote by $\mathsf{Dup}_T \subset V(T)$, the set of all duplication nodes in T. Let $G_1, G_2, \ldots, G_n$ be a collection of rooted gene trees over a species tree S. Assume that, for every $i \in \{1, 2, \ldots, n\}$, F_i is a valid mapping between G_i and S. Every element $s \in \bigcup_i \mathsf{F}_i[\mathsf{Dup}_{G_i}]$ denotes the location of multiple gene duplication events in S. We will refer to such locations as *duplication episodes* or simply *episodes*. Later on, we may also use the term episode to refer to the set of duplications that are mapped into it.

▶ **Problem 1** (Episode Clustering, EC). *Given a collection of rooted gene trees $G_1, G_2, \ldots, G_n$ over a species tree S. Compute the minimal number of duplication episodes, denoted by* $\mathsf{EC}(G_1, G_2, \ldots, G_n, S)$*, in the set of all valid mappings* $\mathsf{F}_1, \mathsf{F}_2, \ldots, \mathsf{F}_n$ *such that* $\mathsf{F}_i \colon V(G_i) \to V(S)$.

This problem can be solved in linear time and space [17].

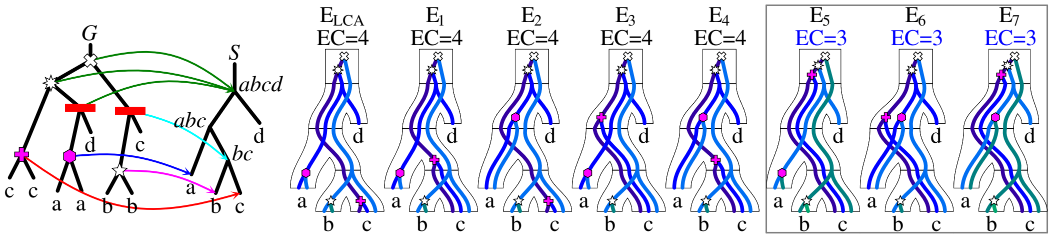

Figure 1 From the left side: a gene tree G and a species tree S with the lca-mapping M shown using arrows from the internal nodes of G to the nodes of S. There are 5 gene duplications, 2 speciation nodes in G (red bars), and 8 valid mappings depicted as embeddings of G into S [11], where the blue lines in these embeddings correspond to the edges of G. E_{LCA} is induced by the lca-mapping. Here, $\mathsf{EC}(G, S) = 3$ as indicated in the three rightmost scenarios with episode sets $\{a, b, abcd\}$ for E_5 and $\{a, abc, abcd\}$ for E_6 and E_7. The example trees are partially adopted from [32].

2.3 Gene-species mappings

We present the main problem for joint reconstruction of gene-species mappings and minimizing the set of episodes.

A *partial gene tree* is a rooted binary tree where each leaf is labeled by a species or has no label. Let G be a partial gene tree over S. By $\Lambda_G \colon L(G) \to L(S)$ we denote the partial *leaf labelling function* such that $\Lambda_G(g)$ is the label (species) of the leaf g in G if defined. Note that any gene tree is a partial gene tree with the leaf labeling being a total function. If a leaf in G has no label we write $\Lambda_G(g) = \perp$. We say that a gene tree G^* over S *extends* a partial gene tree G over S if G and G^* are isomorphic as graphs (i.e., $V(G) = V(G^*)$ and $E(G) = E(G^*)$), and, Λ_{G^*} is a total function that extends Λ_G.

2.4 Inferring labelings by minimizing episodes

Now, we present the problem of the simultaneous reconstruction of leaf labelings and duplication episodes from collections of partial gene trees.

▶ **Problem 2** (MetaEC). *Given a collection of partial gene trees $G_1, G_2, \ldots, G_k$ over a species tree S. Compute the minimum $\mathsf{EC}(G_1^*, G_2^*, \ldots, G_k^*, S)$, denoted $\mathsf{EC}(G_1, G_2, \ldots, G_k, S)$, in the set of all collections of gene trees $G_1^*, G_2^*, \ldots, G_k^*$ such that G_i^* extends G_i, for each i.*

For example, if $(a, (\perp, (\perp, \perp)))$ is a single gene tree with three undefined labels and $(a, (b, c))$ is a species tree, then the problem is to replace all occurrences of $\perp$ by a, b or c such that the total number of duplication episodes is minimized. In this case, the optimal cost is 1, since at least one duplication is needed when the gene tree has four leaves and there are only three species.

3 Methods

We first solve a simpler problem in which we assume that the set of duplication episodes is constrained to a given set of species tree nodes. Then, we show how to solve MetaEC for a single gene tree. Section 3.3 presents the general solution.

3.1 Episode Feasibility Problem

We start with a related constrained problem. Given a partial gene tree, we are interested in the question, of whether there is an extension of the partial gene tree such that the set of corresponding duplication episodes is contained in a given fixed set of episode candidates.

▶ **Problem 3** (Episode Feasibility). *Given a partial gene tree G over a species tree S and $X \subseteq V(S)$. Does there exist a gene tree G^* and a valid mapping F_{G^*} such that G^* extends G and $\mathsf{F}_{G^*}(\mathsf{Dup}_{G^*}) \subseteq X$?*

If a partial gene tree G satisfies the above property, we call G *X-feasible* with respect to a species tree S. If the context is clear, we omit the reference to S.

The solution to Episode Feasibility is a dynamic programming (DP) formulation expressed using Łukasiewicz's Three-Valued Logic Ł$_3$ [43] with three constants True, False, and Unknown (representing uncertainty) and ordered linearly: False < Unknown < True. The logic has binary operators $\vee$ (disjunction, max), $\wedge$ (conjunction, min), and two unary operators L (certainty) and M (possibility). See the interpretation in Figure 2.

$\vee$	$\mathbb{F}$	$\mathbb{U}$	$\mathbb{T}$
$\mathbb{F}$	$\mathbb{F}$	$\mathbb{U}$	$\mathbb{T}$
$\mathbb{U}$	$\mathbb{U}$	$\mathbb{U}$	$\mathbb{T}$
$\mathbb{T}$	$\mathbb{T}$	$\mathbb{T}$	$\mathbb{T}$

$\wedge$	$\mathbb{F}$	$\mathbb{U}$	$\mathbb{T}$
$\mathbb{F}$	$\mathbb{F}$	$\mathbb{F}$	$\mathbb{F}$
$\mathbb{U}$	$\mathbb{F}$	$\mathbb{U}$	$\mathbb{U}$
$\mathbb{T}$	$\mathbb{F}$	$\mathbb{U}$	$\mathbb{T}$

L	
$\mathbb{F}$	$\mathbb{F}$
$\mathbb{U}$	$\mathbb{F}$
$\mathbb{T}$	$\mathbb{T}$

M	
$\mathbb{F}$	$\mathbb{F}$
$\mathbb{U}$	$\mathbb{T}$
$\mathbb{T}$	$\mathbb{T}$

Figure 2 Boolean operations in Three-Valued-Logic used in DP. Here $\mathbb{F}$= False, $\mathbb{U}$= Unknown, and $\mathbb{T}$= True.

For a node g of a gene tree G, by $G|g$ we denote the subtree of G rooted at g. For any non-leaf node t in a tree, by t' and t'', we denote the children of t. To simplify the notation, we assume that the set $X \subseteq V(S)$ is fixed. Then, we have the following dynamic programming formulas that solve Episode Feasibility. Let $g \in V(G)$ and $s \in V(S)$.

$$\delta(g,s) = \begin{cases} \delta^*(g,s) & g \text{ is internal and } s \in X, & (1)\\ \delta^*(g,s) \wedge \mathsf{Unknown} & g \text{ is internal and } s \notin X, & (2)\\ \mathsf{False} & \text{otherwise}, & (3) \end{cases}$$

$$\delta^{\downarrow}(g,s) = \begin{cases} \epsilon(g,s) & s \text{ is a leaf}, & (4)\\ \epsilon(g,s) \vee \mathsf{M}\,\delta^{\downarrow}(g,s') \vee \mathsf{M}\,\delta^{\downarrow}(g,s'') & s \text{ internal, and } s \in X. & (5)\\ \epsilon(g,s) \vee \delta^{\downarrow}(g,s') \vee \delta^{\downarrow}(g,s'') & \text{otherwise}, & (6) \end{cases}$$

$$\sigma(g,s) = \begin{cases} \mathsf{L}\left(\delta^{\downarrow}(g',s') \wedge \delta^{\downarrow}(g'',s'') \vee \delta^{\downarrow}(g',s'') \wedge \delta^{\downarrow}(g'',s')\right) & g \text{ and } s \text{ are internal}, & (7)\\ \mathsf{True} & g \in L(G),\ \Lambda_G(g) \in \{s, \perp\}, & (8)\\ \mathsf{False} & \text{otherwise}, & (9) \end{cases}$$

where

$$\epsilon(g,s) = \sigma(g,s) \vee \delta(g,s), \qquad (10)$$

$$\delta^*(g,s) = \epsilon(g',s) \wedge \delta^{\downarrow}(g'',s) \vee \epsilon(g'',s) \wedge \delta^{\downarrow}(g',s). \qquad (11)$$

In the next Lemma, we express properties satisfied by the above formulas.

For a partial gene tree G over S, and nodes $g \in V(G)$ and $s \in V(S)$, we say that a valid mapping $\mathsf{F}_T\colon V(G|g) \to V(S|s)$ is *feasible* for (g,s,X), if and only if, T extends $G|g$ and $\mathsf{F}_T(\mathsf{Dup}_T) \subseteq X$. Feasible mappings represent episode scenarios that correspond to partial solutions to the instance of Episode Feasibility that forces duplications from $G|g$ to be present in episodes from $X \cap V(S|s)$.

We say a duplication g in a gene tree T is *upper* if the path from g to the root of T contains only duplications. The set of all upper duplications in a tree T we denote by UDup_T. We say that a valid mapping $\mathsf{F}_T\colon V(G|g) \to V(S|s)$ is *weakly feasible* for (g,s,X), if and only if, there is no feasible mapping for (g,s,X), but there is T that extends $G|g$, T has at least one upper duplication d such that $F_T(d) \notin X$ and $\mathsf{F}_T(\mathsf{Dup}_T \setminus \mathsf{UDup}_T) \subseteq X$. In contrast to feasible mappings, in weakly feasible mappings we constrain only non-upper duplications present in $G|g$. Here, the upper duplications are elements of episodes $s \notin X$. This situation is modeled by Unknown value returned from $\delta^{\downarrow}(g,s)$ and $\delta(g,s)$ calls, meaning that there is at least one duplication that needs to be assigned later (if possible) to an episode from $X \setminus V(S|s)$, which eventually occurs at levels of recursion shallower than the level of (g,s).

Informally, the meaning of DP formulas can be understood as follows. Below, let T be an extension of the subtree of G rooted at a node g. The value of $\delta(g,s)$ is True if there exists T where g is an s-duplication and all duplications are assigned to the episodes from X (where $s \in X$ as well). Similarly, the value of $\sigma(g,s)$ is True if there exists T, where g represents an s-speciation or an s-leaf node. Next, $\delta(g,s)$ is Unknown if the condition for $\delta(g,s) =$ True is not met. However, there still exists T, where g represents an s-duplication, and all non-upper duplications from T are assigned to the episodes from X, while the upper duplications are assigned to episodes outside of X. It is important to note that in this case, s is not an element of X. Note that $\sigma(g, s)$ cannot be $\mathsf{Unknown}$ since speciation nodes are fixed. Moving on, $\delta^{\downarrow}(g, s)$ is True if there exists T where all duplications are assigned to the episodes from X. Lastly, $\delta^{\downarrow}(g, s)$ is $\mathsf{Unknown}$ if the condition for $\delta^{\downarrow}(g, s) = \mathsf{True}$ is not met. However, there still exists T, where all non-upper duplications from T are assigned to the episodes from X, while the upper duplications are assigned to episodes outside of X.

While ϵ and δ^* should be treated as "local" in the main formulas (i.e., they should not form separate arrays in implementation), their properties can be formulated as follows. Generally, if $\epsilon(g, s)$ is True, then there is a tree T, where g is mapped into s and all duplications are assigned to the episodes from X. Since $\sigma(g, s)$ cannot be $\mathsf{Unknown}$, $\epsilon(g, s)$ is $\mathsf{Unknown}$ only if $\sigma(g, s)$ is False and $\delta(g, s) = \mathsf{Unknown}$. Again, here g is mapped to s. Now, $\delta^*(g, s)$ is True only if there is T where g is an s-duplication, and all duplication nodes below g are assigned to episodes from X. Importantly, at least one of the children of g must be mapped to s, which is captured by ϵ. Furthermore, $\delta^*(g, s)$ resembles $\delta(g, s)$, but the condition that duplications must be assigned to episodes from X only applies to the duplications (or upper-duplications) below g if $\delta^*(g, s)$ is True (or $\mathsf{Unknown}$, respectively).

The following Lemma formalizes the conditions described above.

▶ **Lemma 1.** *Given a partial gene tree G over a species tree S and $X \subseteq V(S)$. Let $g \in V(G)$, $s \in V(S)$. Then,*

P1 $\delta(g, s) = \mathsf{True}$ *if and only if there is a gene tree T and a feasible mapping F_T for (g, s, X) such that g is an s-duplication in T.*

P2 $\delta(g, s) = \mathsf{Unknown}$ *if and only if there is a gene tree T and a weakly feasible mapping F_T for (g, s, X) such that g is an s-duplication in T.*

P3 $\sigma(g, s) = \mathsf{True}$ *if and only if there a gene tree T and a feasible mapping F_T for (g, s, X) such that and g is an s-speciation or an s-leaf in T.*

P4 *For any g and s, $\sigma(g, s) \neq \mathsf{Unknown}$.*

P5 $\delta^{\downarrow}(g, s)$ *is* True *if and only if there is a feasible mapping for (g, s, X).*

P6 $\delta^{\downarrow}(g, s)$ *is* $\mathsf{Unknown}$ *if and only if there is a weakly feasible mapping for (g, s, X).*

To solve Episode Feasibility, we have to apply $\delta^{\downarrow}$ on the roots of the input trees.

▶ **Theorem 2** (Correctness). *Given a partial gene tree G over a species tree S and $X \subseteq V(S)$. G is X-feasible if and only if $\delta^{\downarrow}_X(\mathsf{root}(G), \mathsf{root}(S))$ is* True.

Proof. The proof follows immediately from P5 of Lemma 1: $\delta^{\downarrow}_X(\mathsf{root}(G), \mathsf{root}(S))$ is True if and only if there is a feasible F_T for $(\mathsf{root}(G), \mathsf{root}(S), X))$ such that T extends G and $\mathsf{F}_T(\mathsf{Dup}_T) \subseteq X$. ◀

▶ **Theorem 3** (Complexity). *Given a partial gene tree G over a species tree S and $X \subseteq V(S)$. The time and space complexity of solving Episode Feasibility by the dynamic programming algorithm is $O(|V(G)||V(S)|)$.*

Proof. We have three arrays δ_X, $\delta^{\downarrow}_X$ and σ_X (note that ϵ and δ^* can be directly inserted in their calls), each of size $O(|V(G)||V(S)|)$ and every cell of an array can be computed in $O(1)$ time. ◀

An example of DP execution with a feasible solution is depicted in Figure 3.

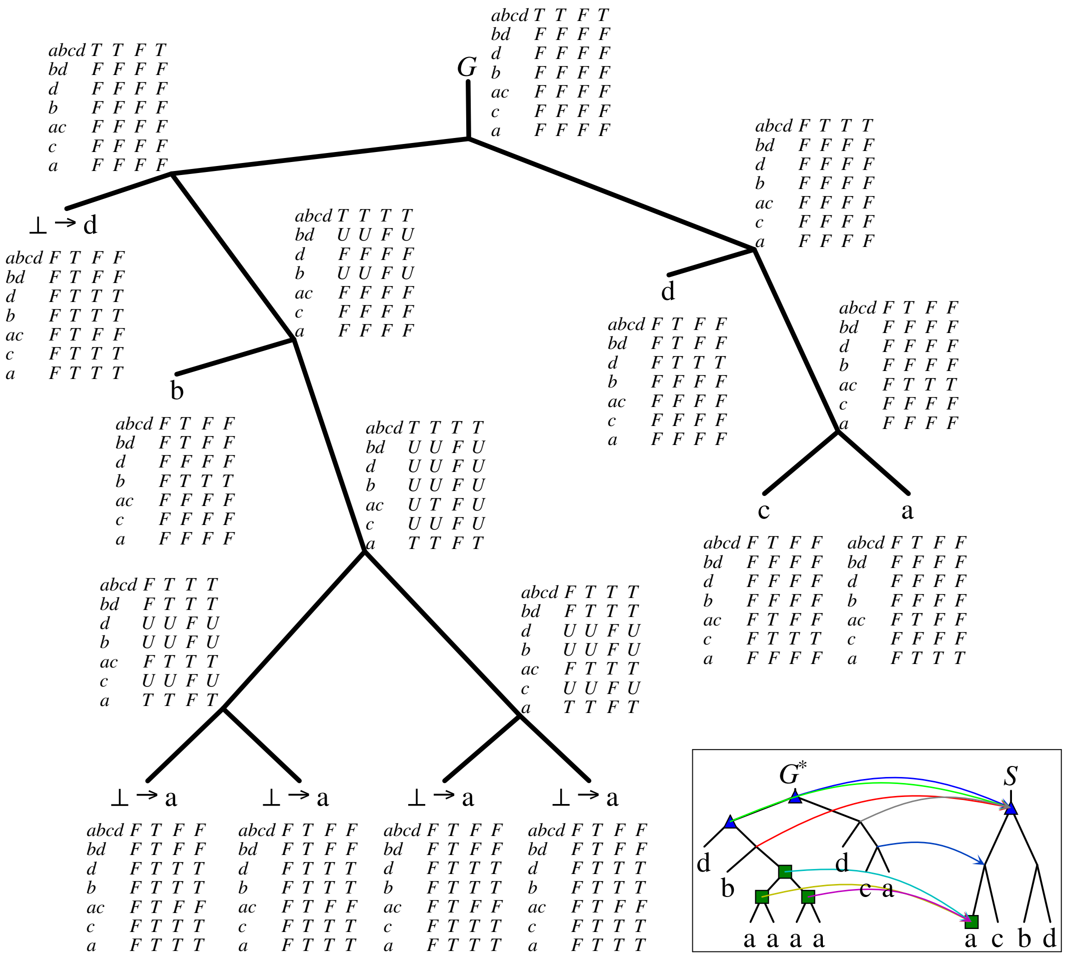

Figure 3 An example of the dynamic programming algorithm (from Section 3.1) execution. The partial gene tree is $G = ((\bot, (b, ((\bot, \bot), (\bot, \bot)))), (d, (c, a)))$, which contains five missing labels. The species tree, denoted as S, is represented as $((a, c), (b, d))$. The marked nodes in S indicate episode candidates from X: the root of S ($abcd$) and the leaf node a. By applying dynamic programming, we obtain a feasible solution, depicted in the bottom-right corner. The resulting extension of the partial gene tree G is G^*, where the valid mapping between G^* and S is the lowest common ancestor (LCA) mapping. In G^*, each duplication node is marked with a triangle or a square denoting their corresponding episode in S. Each node in G is decorated with an array that represents the values of DP formulas, where each row corresponds to a node in S, starting from $abcd$, bd, and so on as indicated in the first column. The next columns have the values of δ, $\delta^{\downarrow}$, σ, and ϵ, respectively, for the gene tree node and the corresponding species tree node. For example, considering the root of G and the root of S, the top row of the array contains the following values: $\delta(\mathsf{root}(G), \mathsf{root}(S)) = \delta^{\downarrow}(\mathsf{root}(G), \mathsf{root}(S)) = \epsilon(\mathsf{root}(G), \mathsf{root}(S)) = \mathsf{True}$, while $\sigma(\mathsf{root}(G), \mathsf{root}(S)) = \mathsf{False}$.

3.2 Solving MetaEC for a single partial gene tree

Here we describe the main algorithm to solve MetaEC for instances with a single gene tree. First, we characterize an important property of episodes.

▶ **Lemma 4** (Fixed Episodes). *Given a partial gene tree G over a species tree S. Assume that there are nodes g in G and s in S such that*

- *if s is the root of S, then at least one proper subtree of G contains species (leaf-labels) from both children of s.*
- *otherwise, let p be the parent of s, then $G|g$ is a gene tree, g is a p-speciation and a child of g is an s-duplication.*

Then, for any G^* that extends G, s is an episode in every valid mapping between G^* and S.

The nodes satisfying the above conditions we call *fixed episodes* (for G and S). For example, for trees from Figure 1, there are two fixed episodes: the root of S and the leaf b, where the duplications with fixed mappings are depicted using white marks in the exemplary gene tree G. The set of all fixed episodes can be computed in linear time and space by bottom-up traversal of the partial gene tree G and by using LCA-queries in the species tree S as follows. For each node g from $V(G)$, the algorithm computes a tuple (u, s, d), where

- $u \in \{\mathsf{True}, \mathsf{False}\}$ is True if and only if $\perp$ is reachable from g,
- $s \in V(S) \cup \{\mathsf{None}\}$ is the least common ancestor of all non-$\perp$ labels reachable from g in S and None if only $\perp$'s are visible from g,
- and $d \in \{\mathsf{True}, \mathsf{False}\}$ is True if and only if $u = \mathsf{False}$ and g is a duplication node in a gene tree $G|g$.

Then, for each g and its tuple (u, s, d), and for each child of g with a tuple (u', s', d'):

- if $s = s' = \mathsf{root}(S)$, then s (the root of S) is a fixed episode,
- if $u = d = \mathsf{False}$, $d' = \mathsf{True}$ and the parent of s' is s, then s' is a fixed episode.

We omit correctness and complexity proofs for brevity. Note that the number of fixed episodes is the lower bound of $\mathsf{EC}(G, S)$.

Algorithm 1 takes as input a partial gene tree G over a species tree S and outputs $\mathsf{EC}(G, S)$. It first computes the set of fixed episodes F (see Lemma 4). The algorithm then starts with an initial maximal episode number b equal to the number of nodes in S. In each iteration of a while loop, the algorithm checks if there is a set C of size $b - |F| - 1$ from the vertices of S that are not in F, such that the partial gene tree G is $C \cup F$-feasible using the dynamic programming algorithm. This step requires $\binom{|V(S)|-|F|}{b-|F|-1}$ calls of DP in the worst case. If such a set C exists, the algorithm computes $\mathsf{EC}(G^*, S)$ by the linear time algorithm from [29], where G^* is the gene tree obtained by backtracking from the corresponding call of DP, and updates b with the result. Note that b is not assigned the value of $|C \cup F|$, since the minimal set of episodes for G^* and S is a subset of $C \cup F$, and it is often significantly smaller than $C \cup F$ in early steps of iteration. Updating b with $\mathsf{EC}(G^*, S)$ guarantees the minimal number of episodes, where some elements of C may be unused. This is an important optimization step. If such a set C does not exist, the algorithm terminates and returns the current value of b.

The correctness of the algorithm follows from the fact that if there is no set X of size $b - 1$ such that G is X-feasible, then there is no set of any size smaller than b that satisfies the property. Since, b represents the number of episodes from some valid mapping, it is also minimal in such a case. Therefore, when the algorithm terminates, $b = \mathsf{EC}(G, S)$, and the algorithm returns the correct value.

Algorithm 1 Solution to MetaEC with a single gene tree.

```
Data: A partial gene tree $G$ over a species tree $S$
Result: $\mathsf{EC}(G, S)$
$F \leftarrow$ the set of all fixed episodes for $G$ and $S$ ;                 // See Lemma 4
$b \leftarrow |V(S)|$ ;                           // the initial maximal episode number
while there is $C \subseteq V(S) \setminus F$ of the size $b - |F| - 1$ and $G$ is $(C \cup F)$-feasible do
    $G^* \leftarrow$ the gene tree obtained from the DP from Section 3.1 by backtracking;
    $b \leftarrow \mathsf{EC}(G^*, S)$;    // $O(n)$-time algorithm from [29]; $\mathsf{EC}(G^*, S) \leq |C \cup F|$.
return $b$;
```

The algorithm's worst-case time complexity is $\sum_{k=f}^{n-f} \binom{n-f}{k} nm = O(nm2^n)$, where f is the size of the set of fixed episodes ($f = |F|$), n denotes the number of vertices in S, and m denotes the number of vertices in G. Despite the exponential time complexity, in our experiments on both simulated and empirical data, we were able to compute exact solutions after only a few executions of the main loop.

3.2.1 Extensions

To identify the optimal solution within the main loop, enumerating all possible combinations of size $b-f-1$ from the set of episode candidates $V(S) \setminus F$ may be time-consuming for larger instances. To address this issue, we propose a heuristic approach that randomly samples combinations of size $b-f-1$ if $\binom{n-f}{b-f-1}$ is large (e.g., > 1000), and adds a stopping condition based on the number of dynamic programming (DP) calls without improvement (e.g., after 100 calls). This approach not only speeds up the algorithm but also provides additional information on whether the returned value is exact or an upper bound obtained by switching to a heuristic mode. See Figure 6 in Section 4 for more details.

Furthermore, based on our experiments, we have observed that the solution is often close to the set of fixed episodes. To leverage this observation, we propose a bottom-up algorithm that explores candidate sets starting from sizes 0, 1, 2, and so on until a feasible solution is found. In this case, the internal search has a time complexity of $O(\binom{n-f}{i})$, starting from $i = 0$. This algorithm can be combined with the heuristic variant described earlier to improve its effectiveness. However, the experimental evaluation did not show significant improvement compared to the top-down method in Algorithm 1.

3.3 MetaEC in the general case

Here we show that MetaEC in a general case, can be solved using a single partial gene tree under an additional assumption. Given a collection of partial gene trees $G_1, G_2, \ldots, G_k$ over a species tree S, let ω be a new species, called *outgroup*, not present in S. We first add the outgroup to every input tree. Let S^ω be species tree (S, ω), $G_1^\omega = (G_1, \omega)$ and $G_i^\omega = ((G_i, \omega), G_{i-1}^\omega)$, for $i > 1$. Then, by ω-MetaEC we define the problem MetaEC with a single partial gene tree, where the extension of a partial labeling cannot introduce ω, i.e., if $\Lambda_{G_1}(v) = \perp$ then $\Lambda_{G_1^*}(v) \neq \omega$. See Figure 4 for illustration.

We have the following property.

▶ **Lemma 5.** *Given a collection of at least two partial gene trees $G_1, G_2, \ldots, G_k$ over a species tree S such that $\omega \notin L(S)$. $X \subseteq V(S)$ is the set of episodes that yields the solution of MetaEC for $G_1, G_2, \ldots, G_k$ and S if and only if $X \cup \{\mathsf{root}(S^\omega)\}$ is the set of episodes that yields the solution to the instance G_k^ω and S^ω of ω-MetaEC.*

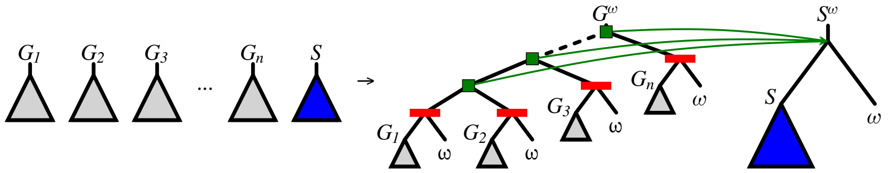

Figure 4 Converting a multiple gene tree instance to a single gene tree instance using an outgroup ω. Red bars in G^ω denote speciation nodes mapped to the root of S^ω. Green squares represent new duplications clustered at a new duplication episode in the root of S^ω.

Note that the algorithms provided in the previous sections can be easily modified to solve ω-MetaEC, by replacing case (8) with:

$$\sigma(g, s) = \mathsf{True} \text{ if } g \in L(G) \text{ and } (\Lambda_G(g) = s \text{ or } (\Lambda_G(g) = \perp \text{ and } s \neq \omega)).$$

Then, DP will exclude extensions of $\perp$ by ω.

4 Experiments

In this Section we present two computational studies based on simulated and empirical data.

4.1 Simulated Trees

Our algorithm was evaluated on five datasets consisting of simulated gene trees and a species tree, each with one or two whole duplication events (as reported in [31]). We generated these datasets and subsequently modified the gene trees to account for the uncertainty commonly associated with metagenomic data.

To begin, we describe how the datasets were generated. Then, we explain the modifications made to the gene trees to account for the uncertainty of metagenomic data. Finally, we present the results provided by our algorithms, specifically in regard to their ability to infer whole genome duplication events.

4.1.1 Dataset preparation

First, we briefly summarize the simulation procedure from [31]. The simulated dataset was generated by SimPhy [20] with parameter settings used in a simulated study [23] that was based on an empirical dataset of 16 Fungi species [34]. The species tree S of 20 taxa was generated by SimPhy with the speciation rate parameter equal to 1.8×10^{-9} and the tree height parameter set to 1.8×10^{9}. To simulate a whole genome duplication (WGD), a node v in the species tree S was chosen as the location of the event. Subsequently, a modified species tree, denoted as S', was constructed by substituting a subtree $S|v$ with a duplicated version of itself. This duplication involved creating a new root connected to the original root of $S|v$ and the root of its copy.

For every S_i one hundred gene trees were then generated from locus trees produced by SimPhy using the modified species tree S', with a duplication and loss rate parameter set to 2^{-10} events per generation per lineage. To minimize the effect of incomplete lineage sorting, the population size parameter was set to 10. In summary, the simulated data from [31] was partitioned into five datasets of gene trees denoted $\mathcal{G}_i = \{G^i_1, G^i_2, \ldots, G^i_{100}\}$, where each dataset comprises 100 gene trees generated using the same species tree S but with a different WGD scenario. The WGD variants used in the simulations are illustrated in Figure 5, where

S_1 represents a single recent WGD Γ, S_2 represents a single ancient WGD Θ, S_3 represents two WGDs Λ and Δ, with Δ occurring after Λ, S_4 represents two close WGDs Π and Π' at the same branch, and S_5 represents two recent independent WGDs Ψ and Φ.

In our evaluation study, for each dataset $\mathcal{G}_i$ from [31], we generated a set of partial gene trees $\tilde{\mathcal{G}}_i^{k,p}$ by randomly removing each leaf label from every gene tree G_j^i in $\mathcal{G}_i$ with probability p. We considered values of p from 0.1 to 0.6 in increments of 0.1, and generated 10 instances of $\tilde{\mathcal{G}}_i^{k,p}$ for each p value and $k = 1, 2, \dots, 10$. This resulted in a total of 300 instances $(\tilde{\mathcal{G}}_i^{k,p}, S)$ of MetaEC, where S is the species tree used to generate $\mathcal{G}_i$. In addition to these instances, we also included the result for $p = 0$, which corresponds to no removal of leaf labels.

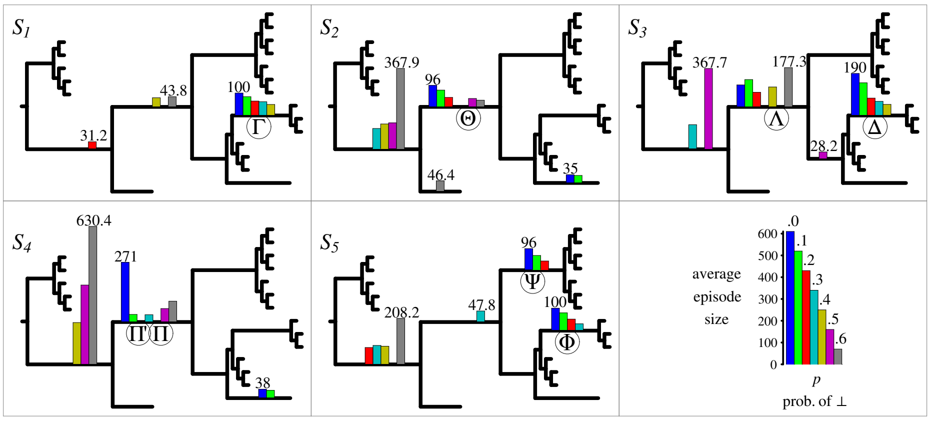

Figure 5 This figure presents a summary of the inferred gene-species mappings and duplication episodes based on the simulated dataset from [31]. Locations of simulated whole-genome duplication (WGD) events are denoted by Greek letters. For clarity, all leaf labels have been removed from the visualization of species trees (see [31] for details). Each bar in the histograms shows the average episode size obtained from the set of partial gene trees $\tilde{\mathcal{G}}_i^{k,p}$, where the average is computed over 10 instances of partial gene trees for fixed p and i (if $p > 0$). For $p = 0$, the bars represent the results for the set of simulated gene trees $\mathcal{G}_i$ (with no $\perp$'s). The key to the diagrams is present at the bottom-right corner. The number above a single bar represents the maximum height of a bar in its histogram. Bars with an average below 25 duplications were omitted as they were deemed insignificant. Additionally, histograms for the root have been moved to Figure 6 (bottom left diagram).

4.1.2 Results

The results of our algorithm on the simulated partial gene trees are depicted in Figure 5, where we summarize the episode sizes in the form of histograms. Also, additional data is provided in Figure 6. The evaluation took about 5 hours of a computing server with 64 cores. In general, we observed that the runtime and the number of DP calls grow linearly with the value of parameter p on average as indicated in the bottom-right diagram of Figure 6. We used the heuristic variant of the algorithm, where random sampling was applied if the number of combinations exceeded 100 trees, and with the stopping criterion equal to 50. Out of 300 instances, 240 were completed with the exact solution (see the top-left diagram of Figure 6). The resulting costs without exact guarantee, were more often obtained for larger values of p. Additionally, we observed that the lower bound given by the number of fixed episodes was a tight approximation of the inferred cost (see the top-right diagram of Figure 6). As p increased, the number of fixed episodes decreased. Note that for $p = 1$

(not included in our analysis), the solution to MetaEC is 1. In such a case the leaf labelings are constant functions and all internal nodes of each gene tree are duplications. Then there is just one episode, which can be placed at the root of the species tree.

Now, we briefly summarize the outcome of WGD detection. First, we present the analysis for the root episodes in a separate diagram in the bottom-left diagram of Figure 6 since the model overestimates the size of the root episode. This is partly due to the property that the last node to which a duplication can be assigned via a valid mapping is the root. In more detail, since the number of fixed episodes is a tight lower bound of the EC cost, the algorithm more likely assigns a duplication at the nearest fixed episode and at the root in the case where such an episode is not present.

The results for S_1 in Figure 5 show that Γ WGD is supported when $p \leq .4$. When $p > .4$, the duplications are relocated to more ancient locations. In S_2, Θ is supported only for $p \leq .2$, while for $p > .2$ large episodes are present at the parent of the WGD location. In S_3, Δ is well supported when $p \leq .4$. The case of Λ is more complicated, since for $p \leq .2$, $p = .4$ and $p = .6$ we see good support, while for the remaining values, the parent location is more supported. The ancient double WGDs in S_4 have perhaps the worst support in our study. Here, we observe generally low supports at the WGDs node. Most of the duplications were shifted to more ancient nodes of S_4, i.e., to the root or to the parent location. In S_5, the support is for smaller values of p.

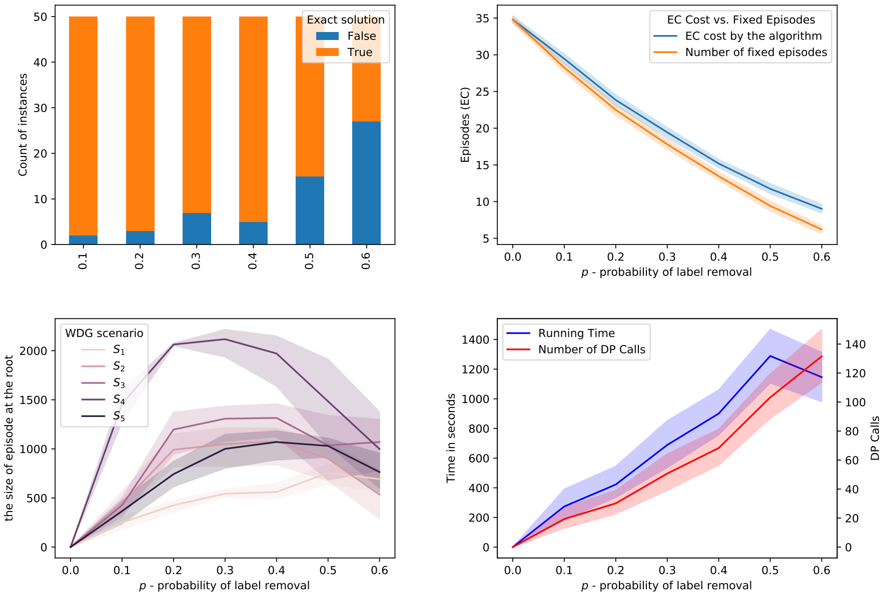

Figure 6 Summary of simulated dataset experiments. All diagrams, except for the top-left, display mean values with 95% confidence intervals. The top-left diagram shows histograms of exact and heuristic solutions returned by Algorithm 1. The top-right diagram presents the EC cost and the number of fixed episodes. The bottom-left diagram displays the sizes of root episodes in all five WGD scenarios. The bottom-right diagram shows the runtime in seconds and the number of executed DP calls.

4.2 Empirical evaluation

To ensure that our algorithm was properly tested, we required a dataset that would capture the characteristics of the metagenomic data as closely as possible, while allowing us to assess the quality and accuracy of the results obtained. For this reason, we decided to prepare a dataset consisting of gene trees for species identified during metagenomic analysis. To simulate potentially missing gene-species assignments, we artificially removed some of the gene labels from the gene trees and retained information about their taxonomic origin for further analysis of the results. Another important issue was the presence of a previously described whole-genome duplication event that occurred in the evolutionary tree of selected species. Given the above requirements, we decided to use proteomes belonging to yeast species identified during metagenomic analysis of kefir [41].

4.2.1 Data preparation

The eight selected species are: *Kazachstania Africana*, *Kazachstania naganishii*, *Naumovozyma dairenensis*, *Tetrapisispora blatte*, *Tetrapisispora phaffi*, *Torulaspora delbrueckii*, *Zygosaccharomyces rouxii* and *Saccharomyces cerevisiae*. A species tree containing the listed species, consistent with the NCBI taxonomy and many papers on yeast evolution, is shown in Figure 7. It also shows the location of the whole-genome duplication event confirmed by previous studies [9, 21].

The proteomes used to infer gene trees were sourced from the UniProt database [5]. Protein families were created by dividing the proteins into groups using the *mcl* program [37] with parameters $I = 2$ and $I = 5$. However, since the differences between the obtained sets were minimal, we used the set obtained for $I = 2$ in subsequent steps. The protein sequences in each group were aligned with the MUSCLE algorithm [7], and unrooted gene trees were inferred using the *phyml* program [14] with the default parameter setting. Rooting of the gene trees was performed by *URec* program [12] using the minimal duplication-loss cost as the rooting criterion.

We removed trees containing fewer than 3 leaves or 3 species, as well as trees with edges of length 0 from the final set of trees. This resulted in 3430 rooted gene trees. Similar to the first experiment, we created 10 datasets for each $p \in \{0.1, 0.2, \ldots, 0.6\}$ by randomly removing each leaf label with the probability p. This resulted in 60 datasets plus the original dataset representing $p = 0$.

4.2.2 Results

Figure 7 depicts histograms showing the results obtained for the described dataset. The evaluation was performed on the same computing server as before and took approximately one hour. For this evaluation, we set the sampling threshold and stopping criterion to 50, which yielded exact solutions for all cases. The number of fixed episodes was consistent across datasets with $p < .6$, at 12 (note that the number of leaves in a tree was 15). For $p = 0.6$, the number of fixed episodes fell within the range of 9 to 12. The number of DP calls ranged from 2 to 3 for $p < .6$ and between 2 and 11 for $p = .6$.

The results obtained by the algorithm for the yeast dataset are consistent with our knowledge of the whole-genome duplication localization. For the dataset with all leaf labels present and for $p = .1$, we have the highest support for the WGD event. The number of supporting single duplications decreases gradually for successive $p's$. For values of $p \geq .5$, the correct WGD localization is still supported by a significantly large number of single duplications. It is worth noting that even for the $p = .6$, the right location is supported by three times as many duplications as the second most supported location, which is in the root. Additionally, we observed an increase in duplications at the leaves of the species tree as p increased. Since most of the leaves are fixed episodes, the algorithm often assigned labels to create duplications at the leaves, resulting in larger sizes of episodes at leaves.

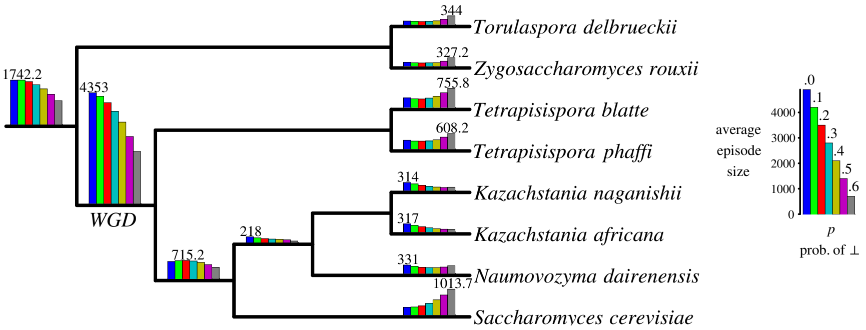

Figure 7 Summary of gene-species mappings and duplication episodes inference for the yeast dataset consisting of 3430 gene trees. WGD denotes the whole genome duplication event postulated in [9, 21]. For the description of symbols refer to Figure 5.

4.3 Software

The software package, which is partially based on the embretnet repository, has been written in Python and is available with all datasets at `https://bitbucket.org/pgor17/metaEC`.

5 Conclusions and Future Outlook

In this article, we presented a novel problem that integrates gene-species mapping inference and genomic duplication detection. We proposed efficient algorithms to solve the problem exactly in the majority of instances, along with a heuristic modification for cases where exact solutions are not feasible. To demonstrate the effectiveness and accuracy of our proposed algorithm, we conducted computational experiments on both simulated and empirical data. The results showed that our algorithm was able to accurately infer the corresponding events when the number of missing labels was relatively small for simulated data. Moreover, the algorithm performed even better on empirical data, demonstrating its robustness and applicability to real-world scenarios.

However, to maximize topological similarities between a gene tree and its species tree, speciation nodes should more frequently appear in the resulting extensions of input partial gene trees. We observe that the optimization model tends to reconstruct leaf labels in a way that prioritizes duplication events assigned to the nearest fixed episodes or the root, in the absence of such episodes. This is confirmed by the property that fixed episodes are tight approximations of the EC cost, leading to a reduction in the number of speciation events in the final gene tree extensions. As a consequence, the model's effectiveness may be limited in some cases when the number of missing labels in partial gene trees is significant.

In the future, we plan to extend the analyzed model to address the importance of speciation nodes in the EC cost formulation. Alternatively, one may limit the distance between the lca-mapping of a gene duplication and its destination mapping in the final scenario similarly to [31]. Additionally, there are models of genomic duplications providing a higher level of detail than EC, such as minimum episodes (ME) [29] and RMP [15], which can be adapted in a similar way to infer gene-species mappings and minimize the number of duplication episodes simultaneously. These models can be further combined with more general models of valid mappings, which allow the introduction of more duplication events than the minimum obtained by the lca-mapping [11]. The combination of these models can provide a more comprehensive approach to inferring gene-species mappings and identifying the minimum number of duplication episodes.

References

1 Mukul S Bansal and Oliver Eulenstein. The multiple gene duplication problem revisited. *Bioinformatics*, 24(13):i132–i138, 2008.

2 Arkadiusz Betkier, Paweł Szczęsny, and Paweł Górecki. Fast algorithms for inferring gene-species associations. In *Bioinformatics Research and Applications: 11th International Symposium, ISBRA 2015 Norfolk, USA, June 7-10, 2015 Proceedings 11*, pages 36–47. Springer, 2015.

3 Craig M. Bielski, Ahmet Zehir, Alexander V. Penson, Mark T. A. Donoghue, Walid Chatila, Joshua Armenia, Matthew T. Chang, Alison M. Schram, Philip Jonsson, Chaitanya Bandlamudi, Pedram Razavi, Gopa Iyer, Mark E. Robson, Zsofia K. Stadler, Nikolaus Schultz, Jose Baselga, David B. Solit, David M. Hyman, Michael F. Berger, and Barry S. Taylor. Genome doubling shapes the evolution and prognosis of advanced cancers. *Nature Genetics*, 50(8):1189–1195, 2018.

4 J Gordon Burleigh, Mukul S Bansal, Andre Wehe, and Oliver Eulenstein. Locating multiple gene duplications through reconciled trees. In *Research in Computational Molecular Biology: 12th Annual International Conference, RECOMB 2008, Singapore, March 30-April 2, 2008. Proceedings 12*, pages 273–284. Springer, 2008.

5 The UniProt Consortium. Uniprot: the universal protein knowledgebase in 2023. *Nucleic Acids Research*, 51(D1):D523–D531, 2023.

6 Riccardo Dondi, Manuel Lafond, and Celine Scornavacca. Reconciling multiple genes trees via segmental duplications and losses. *Algorithms for Molecular Biology*, 14:7, 2019.

7 Robert C Edgar. Muscle: a multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics*, 5(1):1–19, 2004.

8 Michael Fellows, Michael Hallet, and Ulrike Stege. On the multiple gene duplication problem. In *9th International Symposium on Algorithms and Computation (ISAAC'98), Lecture Notes in Computer Science 1533*, pages 347–356, Taejon, Korea, 1998.

9 Bing Feng, Yu Lin, Lingxi Zhou, Yan Guo, Robert Friedman, Ruofan Xia, Fei Hu, Chao Liu, and Jijun Tang. Reconstructing yeasts phylogenies and ancestors from whole genome data. *Scientific Reports*, 7(1):1–12, 2017.

10 Morris Goodman, John Czelusniak, G. William Moore, A. E. Romero-Herrera, and Genji Matsuda. Fitting the gene lineage into its species lineage, a parsimony strategy illustrated by cladograms constructed from globin sequences. *Systematic Zoology*, 28(2):132–163, 1979.

11 Paweł Górecki and Jerzy Tiuryn. DLS-trees: A model of evolutionary scenarios. *Theoretical Computer Science*, 359(1-3):378–399, 2006.

12 Paweł Górecki and Jerzy Tiuryn. Urec: a system for unrooted reconciliation. *Bioinformatics*, 23(4):511–512, 2007.

13 Roderic Guigó, Ilya B. Muchnik, and Temple F. Smith. Reconstruction of ancient molecular phylogeny. *Molecular Phylogenetics and Evolution*, 6(2):189–213, 1996.

14 Stéphane Guindon, Jean-François Dufayard, Lefort Vincent, Maria Anisimova, Wim Hordijk, and Olivier Gascuel. New algorithms and methods to estimate maximum-likelihood phylogenies: Assessing the performance of phyml 3.0. *Systematic Biology*, 59(3):307–321, 2010.

15 Leo Van Iersel, Remie Janssen, Mark Jones, Yukihiro Murakami, and Norbert Zeh. Polynomial-Time Algorithms for Phylogenetic Inference Problems involving duplication and reticulation. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2019.

16 Elena Kuzmin, Benjamin VanderSluis, Alex N. Nguyen Ba, Wen Wang, Elizabeth N. Koch, Matej Usaj, Anton Khmelinskii, Mojca Mattiazzi Usaj, Jolanda van Leeuwen, Oren Kraus, Amy Tresenrider, Michael Pryszlak, Ming-Che Hu, Brenda Varriano, Michael Costanzo, Michael Knop, Alan Moses, Chad L. Myers, Brenda J. Andrews, and Charles Boone. Exploring whole-genome duplicate gene retention with complex genetic interaction analysis. *Science*, 368(6498):eaaz5667, 2020.

17 Cheng-Wei Luo, Ming-Chiang Chen, Yi-Ching Chen, Roger W. L. Yang, Hsiao-Fei Liu, and Kun-Mao Chao. Linear-time algorithms for the multiple gene duplication problems. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(1):260–265, 2011.

18 Saioa López, Emilia L Lim, Stuart Horswell, Kerstin Haase, Ariana Huebner, Michelle Dietzen, Thanos P Mourikis, Thomas B K Watkins, Andrew Rowan, Sally M Dewhurst, Nicolai J Birkbak, Gareth A Wilson, Peter Van Loo, Mariam Jamal-Hanjani, TRACERx Consortium, Charles Swanton, and Nicholas McGranahan. Interplay between whole-genome doubling and the accumulation of deleterious alterations in cancer evolution. *Nature Genetics*, 52(3):283–293, 2020.

19 Bin Ma, Ming Li, and Louxin Zhang. From gene trees to species trees. *SIAM Journal on Computing*, 30(3):729–752, 2000.

20 Diego Mallo, Leonardo De Oliveira Martins, and David Posada. Simphy: Phylogenomic simulation of gene, locus, and species trees. *Systematic Biology*, 65(2):334–344, 2016.

21 Marina Marcet-Houben and Toni Gabaldón. Beyond the whole-genome duplication: phylogenetic evidence for an ancient interspecies hybridization in the baker's yeast lineage. *PLoS biology*, 13(8):e1002220, 2015.

22 Vacharapat Mettanant and Jittat Fakcharoenphol. A linear-time algorithm for the multiple gene duplication problem. In *The 12th National Computer Science and Engineering Conference (NCSEC)*, pages 198–203, 2008.

23 Erin K Molloy and Tandy Warnow. FastMulRFS: fast and accurate species tree estimation under generic gene duplication and loss models. *Bioinformatics*, 36(Supplement_1):i57–i65, 2020.

24 Agnieszka Mykowiecka, Paweł Szczęsny, and Paweł Górecki. Inferring gene-species assignments in the presence of horizontal gene transfer. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 15(5):1571–1578, 2017.

25 Susumu Ohno. *Evolution by gene duplication.* Springer-Verlag, Berlin, 1970.

26 Roderic D. M. Page. Maps Between Trees and Cladistic Analysis of Historical Associations among Genes, Organisms, and Areas. *Systematic Biology*, 43(1):58–77, 1994.

27 Roderic D.M. Page and James A. Cotton. Vertebrate phylogenomics: Reconciled trees and gene duplications. *Pacific Symposium on Biocomputing*, pages 536–547, 2002.

28 Jarosław Paszek and Paweł Górecki. Genomic duplication problems for unrooted gene trees. *BMC Genomics*, 17(1):165–175, 2016.

29 Jarosław Paszek and Paweł Górecki. Efficient algorithms for genomic duplication models. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 15(5):1515–1524, 2018.

30 Jarosław Paszek and Paweł Górecki. Inferring duplication episodes from unrooted gene trees. *BMC Genomics*, 19(S5), 2018.

31 Jarosław Paszek, Alexey Markin, Paweł Górecki, and Oliver Eulenstein. Taming the duplication-loss-coalescence model with integer linear programming. *Journal of Computational Biology*, 28(8):758–773, 2021.

32 Jarosław Paszek, Jerzy Tiuryn, and Paweł Górecki. Minimizing genomic duplication episodes. *Computational Biology and Chemistry*, 89:107260, 2020.

33 Ryan J Quinton, Amanda DiDomizio, Marc A Vittoria, Kristýna Kotýnková, Carlos J Ticas, Sheena Patel, Yusuke Koga, Jasmine Vakhshoorzadeh, Nicole Hermance, Taruho S Kuroda, Neha Parulekar, Alison M Taylor, Amity L Manning, Joshua D Campbell, and Neil J Ganem. Whole-genome doubling confers unique genetic vulnerabilities on tumour cells. *Nature*, 590(7846):492–497, 2021.

34 Matthew D. Rasmussen and Manolis Kellis. Unified modeling of gene duplication, loss, and coalescence using a locus tree. *Genome Research*, 22(4):755–765, 2012.

35 Marta Royo-Llonch, Pablo Sánchez, Clara Ruiz-González, Guillem Salazar, Carlos Pedrós-Alió, Marta Sebastián, Karine Labadie, Lucas Paoli, Federico M. Ibarbalz, Lucie Zinger, Benjamin Churcheward, Tara Oceans Coordinators, Samuel Chaffron, Damien Eveillard, Eric Karsenti, Shinichi Sunagawa, Patrick Wincker, Lee Karp-Boss, Chris Bowler, and Silvia G. Acinas. Compendium of 530 metagenome-assembled bacterial and archaeal genomes from the polar Arctic Ocean. *Nature Microbiology*, 6(12):1561–1574, 2021.

36 Ayelet Salman-Minkov, Niv Sabath, and Itay Mayrose. Whole-genome duplication as a key factor in crop domestication. *Nature Plants*, 2:16115, 2016.

37 Stijn Van Dongen. Graph clustering via a discrete uncoupling process. *SIAM Journal on Matrix Analysis and Applications*, 30(1):121–141, 2008.

38 Jakob Wirbel, Paul Theodor Pyl, Ece Kartal, Konrad Zych, Alireza Kashani, Alessio Milanese, Jonas S Fleck, Anita Y Voigt, Albert Palleja, Ruby Ponnudurai, Shinichi Sunagawa, Luis Pedro Coelho, Petra Schrotz-King, Emily Vogtmann, Nina Habermann, Emma Niméus, Andrew M Thomas, Paolo Manghi, Sara Gandini, Davide Serrano, Sayaka Mizutani, Hirotsugu Shiroma, Satoshi Shiba, Tatsuhiro Shibata, Shinichi Yachida, Takuji Yamada, Levi Waldron, Alessio Naccarati, Nicola Segata, Rashmi Sinha, Cornelia M Ulrich, Hermann Brenner, Manimozhiyan Arumugam, Peer Bork, and Georg Zeller. Meta-analysis of fecal metagenomes reveals global microbial signatures that are specific for colorectal cancer. *Nature Medicine*, 25(4):679–689, 2019.

39 Kenneth H Wolfe and Denis C Shields. Molecular evidence for an ancient duplication of the entire yeast genome. *Nature*, 387(6634):708–713, 1997.

40 Shan Wu, Kin H Lau, Qinghe Cao, John P Hamilton, Honghe Sun, Chenxi Zhou, Lauren Eserman, Dorcus C Gemenet, Bode A Olukolu, Haiyan Wang, Emily Crisovan, Grant T Godden, Chen Jiao, Xin Wang, Mercy Kitavi, Norma Manrique-Carpintero, Brieanne Vaillancourt, Krystle Wiegert-Rininger, Xinsun Yang, Kan Bao, Jennifer Schaff, Jan Kreuze, Wolfgang Gruneberg, Awais Khan, Marc Ghislain, Daifu Ma, Jiming Jiang, Robert O M Mwanga, Jim Leebens-Mack, Lachlan J M Coin, G Craig Yencho, C Robin Buell, and Zhangjun Fei. Genome sequences of two diploid wild relatives of cultivated sweetpotato reveal targets for genetic improvement. *Nature Communications*, 9(1):4580, 2018.

41 Birsen Yilmaz, Emine Elibol, H Nakibapher Jones Shangpliang, Fatih Ozogul, and Jyoti Prakash Tamang. Microbial communities in home-made and commercial kefir and their hypoglycemic properties. *Fermentation*, 8(11):590, 2022.

42 Louxin Zhang and Yun Cui. An efficient method for dna-based species assignment via gene tree and species tree reconciliation. In *Algorithms in Bioinformatics: 10th International Workshop, WABI 2010, Liverpool, UK, September 6-8, 2010. Proceedings 10*, pages 300–311. Springer, 2010.

43 Jan Łukasiewicz. *Selected Works*, volume 1. North-Holland Publishing Company, Amsterdam, 1970.