ERRATUM

This manuscript contains a minor error that should be corrected: All occurrences of the reconciliation cost “$10k + 39m - 12n$” should be replaced with “$k + 48m - 12n$”.

On the Complexity of Duplication-Transfer-Loss Reconciliation with Non-Binary Gene Trees

Misagh Kordi[1] and Mukul S. Bansal[1,2]

[1] Department of Computer Science and Engineering, University of Connecticut, Storrs, USA
[2] Institute for Systems Genomics, University of Connecticut, Storrs, USA
misagh.kordi@uconn.edu, mukul@engr.uconn.edu

Abstract. Duplication-Transfer-Loss (DTL) reconciliation has emerged as a powerful technique for studying gene family evolution in the presence of horizontal gene transfer. DTL reconciliation takes as input a gene family phylogeny and the corresponding species phylogeny, and reconciles the two by postulating speciation, gene duplication, horizontal gene transfer, and gene loss events. Efficient algorithms exist for finding optimal DTL reconciliations when the gene tree is binary. However, gene trees are frequently non-binary. With such non-binary gene trees, the reconciliation problem seeks to find a binary resolution of the gene tree that minimizes the reconciliation cost. Given the prevalence of non-binary gene trees, many efficient algorithms have been developed for this problem in the context of the simpler Duplication-Loss (DL) reconciliation model. Yet, no efficient algorithms exist for DTL reconciliation with non-binary gene trees and the complexity of the problem remains unknown. In this work, we resolve this open question by showing that the problem is, in fact, NP-hard. Our reduction applies to both the dated and undated formulations of DTL reconciliation. By resolving this long-standing open problem, this work will spur the development of both exact and heuristic algorithms for this important problem.

1 Introduction

Duplication-Transfer-Loss (DTL) reconciliation is one of the most powerful techniques for studying gene and genome evolution in microbes and other non-microbial species engaged in horizontal gene transfer. DTL reconciliation accounts for the role of gene duplication, gene loss, and horizontal gene transfer in shaping gene families and can infer these evolutionary events through the systematic comparison and reconciliation of gene trees and species trees. Specifically, given a gene tree and a species tree, DTL reconciliation shows the evolution of the gene tree inside the species tree, and explicitly infers duplication, transfer, and loss events. Accurate knowledge of gene family evolution has many uses in biology, including inference of orthologs, paralogs and xenologs for functional genomic studies, e.g., [1, 2], reconstruction of ancestral gene content, e.g., [3, 4], and accurate gene tree and species tree construction, e.g., [2, 5–7], and the DTL reconciliation problem has therefore been widely studied, e.g., [4, 8–15].

DTL reconciliation is typically formulated using a parsimony framework where each evolutionary event is assigned a cost and the goal is to find a reconciliation with minimum total cost. The resulting optimization problem is called the *DTL-reconciliation* *problem*. DTL-reconciliations can sometimes be *time-inconsistent*; i.e, the inferred transfers may induce contradictory constraints on the dates for the internal nodes of the species tree. The problem of finding an optimal *time-consistent* reconciliation is known to be NP-hard [10, 16]. Thus, in practice, the goal is to find an optimal (not necessarily time-consistent) DTL-reconciliation [4, 10, 11, 13, 15] and this problem can be solved in $O(mn)$ time [11], where m and n denote the number of nodes in the gene tree and species tree, respectively. Interestingly, the problem of finding an optimal time-consistent reconciliation actually becomes efficiently solvable [9, 17] in $O(mn^2)$ time if the species tree is fully dated. Thus, these two efficiently solvable formulations, regular and dated, are the two standard formulations of the DTL-reconciliation problem.

Both these formulations of the DTL-reconciliation problem assume that the input gene tree and species tree are binary. However, gene trees are frequently non-binary in practice. This is due to the fact that there is often insufficient information in the underlying gene sequences to fully resolve gene tree topologies. When the input consists of a non-binary gene tree, the reconciliation problem seeks to find a binary resolution of the gene tree that minimizes the reconciliation cost. Given the prevalence of non-binary gene trees, many efficient algorithms have been developed for this problem in the context of the simpler Duplication-Loss (DL) reconciliation model [5, 18–20], with the most efficient of these algorithms having an optimal $O(m + n)$ time complexity [20]. However, the DTL reconciliation model is more general and significantly more complex than the DL reconciliation model. Consequently, no efficient algorithms exist for DTL reconciliation with non-binary gene trees and the complexity of the problem remains unknown. As a result, DTL reconciliation is currently inapplicable to non-binary gene trees, significantly reducing its utility in practice.

In this work, we settle this open problem by proving that the DTL-reconciliation problem on non-binary gene trees is, in fact, NP-hard. Our proof is based on a reduction from the minimum 3-set cover problem and applies to both formulations of the DTL-reconciliation problem. An especially desirable feature of our reduction is that it implies NP-hardness for biologically relevant settings of the event cost parameters, showing that the problem is difficult even for biologically meaningful scenarios. The uncertainty about the complexity of DTL-reconciliation for non-binary gene trees has prevented the development of any algorithms, exact or heuristic, for the problem. By settling this question, our work will spur the development of both exact (better than brute-force) and efficient approximation and heuristic algorithms for this important problem.

We develop our NP-hardness proof in the context of the regular (undated) DTL-reconciliation formulation, and revisit dated DTL-reconciliation later in Section 4. The next section introduces basic definitions and preliminaries, and we present the NP-hardness proof for the optimal gene tree resolution problem in Section 3. Concluding remarks appear in Section 5. In the interest of brevity, proofs for all Lemmas are deferred to the full version of this paper.

2 Definitions and preliminaries

We follow the basic definitions and notation from [11]. Given a tree T, we denote its node, edge, and leaf sets by $V(T)$, $E(T)$, and $Le(T)$ respectively. If T is rooted, the root node of T is denoted by $\mathit{rt}(T)$, the parent of a node $v \in V(T)$ by $\mathit{pa}_T(v)$, its set of children by $\mathit{Ch}_T(v)$, and the (maximal) subtree of T rooted at v by $T(v)$. The set of *internal nodes* of T, denoted $I(T)$, is defined to be $V(T) \setminus \mathit{Le}(T)$. We define $\leq_T$ to be the partial order on $V(T)$ where $x \leq_T y$ if y is a node on the path between $\mathit{rt}(T)$ and x. The partial order $\geq_T$ is defined analogously, i.e., $x \geq_T y$ if x is a node on the path between $\mathit{rt}(T)$ and y. We say that y is an *ancestor* of x, or that x is a *descendant* of y, if $x \leq_T y$ (note that, under this definition, every node is a descendant as well as ancestor of itself). We say that x and y are *incomparable* if neither $x \leq_T y$ nor $y \leq_T x$. Given a non-empty subset $L \subseteq \mathit{Le}(T)$, we denote by $\mathit{lca}_T(L)$ the last common ancestor (LCA) of all the leaves in L in tree T. Throughout this work, the *term* tree refers to rooted trees. A tree is *binary* if all of its internal nodes have exactly two children, and *non-binary* otherwise. We say that a tree T' is a *binary resolution* of T if T' is binary and T can be obtained from T' by contracting one or more edges. We denote by $\mathcal{BR}(T)$ the set of all binary resolutions of a non-binary tree T.

Gene trees may be either binary or non-binary while the species tree is always assumed to be binary. Throughout this work, we denote the gene tree and species tree under consideration by G and S, respectively. If G is restricted to be binary we refer to it as G^B and as G^N if it is restricted to be non-binary. We assume that each leaf of the gene tree is labeled with the species from which that gene was sampled. This labeling defines a *leaf-mapping* $\mathcal{L}_{G,S}\colon \mathit{Le}(G) \to \mathit{Le}(S)$ that maps a leaf node $g \in \mathit{Le}(G)$ to that unique leaf node $s \in \mathit{Le}(S)$ which has the same label as g. Note that gene trees may have more than one gene sampled from the same species. We will implicitly assume that the species tree contains all the species represented in the gene tree.

2.1 Reconciliation and DTL-scenarios

A binary gene tree can be reconciled with a species tree by mapping the gene tree into the species tree. Next, we define what constitutes a valid reconciliation; specifically, we define a Duplication-Transfer-Loss scenario (DTL-scenario) [10, 11] for G^B and S that characterizes the mappings of G^B into S that constitute a biologically valid reconciliation. Essentially, DTL-scenarios map each gene tree node to a unique species tree node in a consistent way that respects the immediate temporal constraints implied by the species tree, and designate each gene tree node as representing either a speciation, duplication, or transfer event.

Definition 1 (DTL-scenario). *A DTL-scenario for G^B and S is a seven-tuple $\langle \mathcal{L}, \mathcal{M}, \Sigma, \Delta, \Theta, \Xi, \tau \rangle$, where $\mathcal{L}\colon \mathit{Le}(G^B) \to \mathit{Le}(S)$ represents the leaf-mapping from G^B to S, $\mathcal{M}\colon V(G^B) \to V(S)$ maps each node of G^B to a node of S, the sets Σ, Δ, and Θ partition $I(G^B)$ into speciation, duplication, and transfer nodes respectively, Ξ is a subset of gene tree edges that represent transfer edges, and $\tau\colon \Theta \to V(S)$ specifies the recipient species for each transfer event, subject to the following constraints:*

1. *If $g \in \mathit{Le}(G^B)$, then $\mathcal{M}(g) = \mathcal{L}(g)$.*
2. *If $g \in I(G^B)$ and g' and g'' denote the children of g, then,*
   (a) $\mathcal{M}(g) \not<_S \mathcal{M}(g')$ and $\mathcal{M}(g) \not<_S \mathcal{M}(g'')$,
   (b) At least one of $\mathcal{M}(g')$ and $\mathcal{M}(g'')$ is a descendant of $\mathcal{M}(g)$.

3. *Given any edge $(g, g') \in E(G^B)$, $(g, g') \in \Xi$ if and only if $\mathcal{M}(g)$ and $\mathcal{M}(g')$ are incomparable.*
4. *If $g \in I(G^B)$ and g' and g'' denote the children of g, then,*
   (a) $g \in \Sigma$ only if $\mathcal{M}(g) = lca(\mathcal{M}(g'), \mathcal{M}(g''))$ and $\mathcal{M}(g')$ and $\mathcal{M}(g'')$ are incomparable,
   (b) $g \in \Delta$ only if $\mathcal{M}(g) \geq_S lca(\mathcal{M}(g'), \mathcal{M}(g''))$,
   (c) $g \in \Theta$ if and only if either $(g, g') \in \Xi$ or $(g, g'') \in \Xi$.
   (d) If $g \in \Theta$ and $(g, g') \in \Xi$, then $\mathcal{M}(g)$ and $\tau(g)$ must be incomparable, and $\mathcal{M}(g')$ must be a descendant of $\tau(g)$, i.e., $\mathcal{M}(g') \leq_S \tau(g)$.

DTL-scenarios correspond naturally to reconciliations and it is straightforward to infer the reconciliation of G^B and S implied by any DTL-scenario. Figure 1 shows an example of a DTL-scenario. Given a DTL-scenario α, one can directly count the minimum number of gene losses, $Loss_\alpha$, in the corresponding reconciliation. For brevity, we refer the reader to [11] for further details on how to count losses in DTL-scenarios.

Let P_Δ, P_Θ, and P_{loss} denote the non-negative costs associated with duplication, transfer, and loss events, respectively. The reconciliation cost of a DTL-scenario is defined as follows.

Definition 2 (Reconciliation cost of a DTL-scenario). *Given a DTL-scenario $\alpha = \langle \mathcal{L}, \mathcal{M}, \Sigma, \Delta, \Theta, \Xi, \tau \rangle$ for G^B and S, the* reconciliation cost *associated with α is given by $\mathcal{R}_\alpha = P_\Delta \cdot |\Delta| + P_\Theta \cdot |\Theta| + P_{loss} \cdot Loss_\alpha$.*

A most parsimonious reconciliation is one that has minimum reconciliation cost.

Definition 3 (Most Parsimonious Reconciliation (MPR)). *Given G^B and S, along with P_Δ, P_Θ, and P_{loss}, a* most parsimonious reconciliation (MPR) *for G^B and S is a DTL-scenario with minimum reconciliation cost.*

2.2 Optimal gene tree resolution

Non-binary gene trees cannot be directly reconciled against a species tree. Thus, given a non-binary gene tree G^N, the problem is to find a binary resolution of G^N whose MPR with S has the smallest reconciliation cost. An example of a non-binary gene tree and a binary resolution is shown in Figure 1.

Problem 1 (Optimal Gene Tree Resolution (OGTR)) *Given G^N and S, along with P_Δ, P_Θ, and P_{loss}, the* Optimal Gene Tree Resolution (OGTR) *problem is to find a binary resolution G^B of G^N such that the MPR of G^B and S has the smallest reconciliation cost among all $G^B \in \mathcal{BR}(G^N)$.*

3 NP-hardness of the OGTR problem

We claim that the OGTR problem is NP-hard; specifically, that the corresponding decision problem is NP-Complete. The decision version of the OTGR problem is as follows:

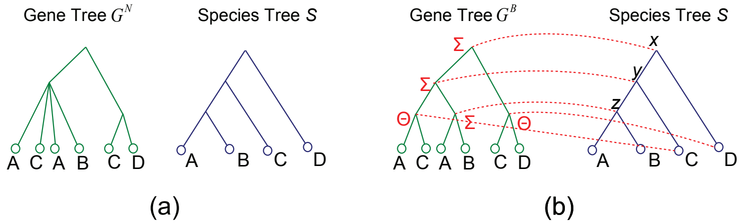

Fig. 1. DTL reconciliation and OGTR problem. Part (a) shows a non-binary gene tree G^N and binary species tree S. Part (b) shows a DTL reconciliation between a possible binary resolution G^B of G^N and species tree S. The dotted arcs show the mapping $\mathcal{M}$ (with the leaf mapping being specified by the leaf labels on the gene tree), and the label at each internal node of G^B specifies the type of event represented by that node. This reconciliation invokes two transfer events.

Problem 2 (Decision-OGTR (D-OGTR))

Instance: *G^N and S, event costs P_Δ, P_Θ, and P_{loss}, and a non-negative integer l.*
Question: *Does there exist a $G^B \in \mathcal{BR}(G^N)$ such that the MPR of G^B and S has reconciliation cost at most l?*

Theorem 1. *The D-OGTR problem is NP-Complete.*

The D-OGTR problem is clearly in NP. In the remainder of this section we will show that the D-OGTR problem is NP-hard using a poly-time reduction from the decision version of the NP-hard *minimum 3-set cover* problem [21].

3.1 Reduction from minimum 3-set cover

The decision version of minimum 3-set cover can be stated as follows.

Problem 3 (Minimum 3-Set Cover (M3SC))

Instance: *Given a set of n elements $U = \{u_1, u_2, \ldots, u_n\}$, a set $A = \{A_1, A_2, ..., A_m\}$ of m subsets of U such that $|A_i| = 3$ for each $1 \leq i \leq m$, and a nonnegative integer $k \leq m$.*
Question: *Is there a subset of A of size at most k whose union is U?*

We point out that the M3SC problem as defined above is a slight variation of the traditional minimum 3-set cover problem: In our formulation the subsets of U in A are restricted to have *exactly* three elements each while the traditional formulation allows for the subsets to have *less than or equal to* three elements [21]. However, it is easy to establish that the NP-Completeness of the traditional version directly implies the NP-Completeness of our formulation of the M3SC problem. We will also assume, without any loss of generality, that each element u_i appears in at least two subsets from A.

Consider an instance ϕ of the M3SC problem with $U = \{u_1, u_2, \ldots, u_n\}$, $A = \{A_1, A_2, ..., A_m\}$, and k given. We now show how to transform ϕ into an instance λ of the D-OGTR problem by constructing G^N and S and setting the three event costs in such a way that there exists a YES answer to the M3SC instance ϕ if and only if there exists a YES answer to the D-OGTR instance λ with $l = 10k + 39m - 12n$.

3.2 Gadget

Gene tree. We first show how to construct the gene tree G^N. Note that each element of U occurs in at least two of the subsets from A. We will treat each of the occurrences of an element separately and will order them according to the indices p of the A_p's which contain that element. More precisely, for an element $u_i \in U$, we denote by $x_{i,j}$ the j^{th} occurrence of u_i in A. For instance, if element u_5 occurs in the subsets A_2, A_4, A_{10}, and A_{25}, then $x_{5,2}$ refers to the occurrence of u_5 in A_4, while $x_{5,4}$ refers to the occurrence of u_5 in A_{25}.

Let c_i denote the cardinality of the set $\{A_p \colon u_i \in A_p, \text{ for } 1 \leq p \leq m\}$. Then, $x_{i,j}$ is well defined as long as $1 \leq i \leq n$ and $1 \leq j \leq c_i$. Each $x_{i,j}$ will correspond to exactly four leaves, $x_{i,j,1}$, $x_{i,j,2}$, $x_{i,j,3}$, and $x_{i,j,4}$ in the gene tree G^N. In addition, the leaf set of G^N also contains a special node labeled *start*, provided for orientation.

Thus, $\mathit{Le}(G^N) = \{x_{i,j,1}, x_{i,j,2}, x_{i,j,3}, x_{i,j,4} \colon 1 \leq i \leq n \text{ and } 1 \leq j \leq c_i\} \cup \{\mathit{start}\}$. The overall structure of G^N is shown in Figure 2(a). As shown, the root node of the gene tree is unresolved and has $3m + 3n + 1$ children consisting of (i) the *start* node, (ii) the $\sum_{i=1}^{n} c_i = 3m$ leaf nodes, collectively called *blue* nodes, and (iii) the $3n$ internal nodes labeled g_i, g_i', and g_i'', for each $1 \leq i \leq n$. These internal nodes represent the n elements in U and the subtrees rooted at those nodes have the structure shown in Figure 2(a). Note that the number of children for each of the internal nodes labeled g_i, g_i', and g_i'', for $1 \leq i \leq n$, is c_i. These nodes may thus be either binary or non-binary. The leaves labeled $x_{i,j,3}$ appear in the node g_i', those labeled $x_{i,j,4}$ appear in g_i'', and those labeled $x_{i,j,1}$ or $x_{i,j,2}$ appear in g_i. The $x_{i,j,1}$'s also appear in the collection of blue nodes and thus appear twice in the gene tree. Note, also, that all the children of a node g_i, for $1 \leq i \leq n$, are themselves internal nodes and are labeled $y_{i,j}$, where $1 \leq j \leq c_i$.

Species tree. Next, we show how to construct the species tree S. The tree S is binary and consists of m subtrees whose root nodes are labeled $s_1, \ldots s_m$, each corresponding to a subset from A, connected together through a backbone tree as shown in Figure 2(b). The exact structure of this backbone tree is unimportant, as long as each s_i is sufficiently separated from the roots of the rest of the subtrees. For concreteness, we will assume that this backbone consists of a "caterpillar" tree as shown Figure 2(b), and that $9m$ extraneous leaves (not present in the gene tree) have been added to this backbone as shown in the figure to ensure that each pair of subtrees is sufficiently separated.

Recall that we use $x_{i,j}$ to denote the j^{th} occurrence of u_i in A. Assuming that $u_i \in A_p$ and that $x_{i,j}$ refers to the occurrence of u_i in A_p, we define $f(i,p)$ to be j. In other words, if the j^{th} occurrence of an element u_i is in the subset A_p, then we assign $f(i,p)$ to be j. Each S_i corresponds to the subset A_i and has the structure depicted in Figure 2(b). In particular, if A_i contains the three elements u_a, u_b, and u_c, then S_i contains the 12 leaves labeled $x_{a,f(a,i),j}$, $x_{b,f(b,i),j}$, and $x_{c,f(c,i),j}$, for $1 \leq j \leq 4$.

Event costs. We assign the following event costs for problem instance λ: $P_\Delta = 2$, $P_\Theta = 4$, and $P_{loss} = 1$.

Note that the D-OGTR instance λ can be constructed in time polynomial in m and n.

Claim 1 *There exists a YES answer to the M3SC instance ϕ if and only if there exists a YES answer to the D-OGTR instance λ with $l = 10k + 39m - 12n$.*

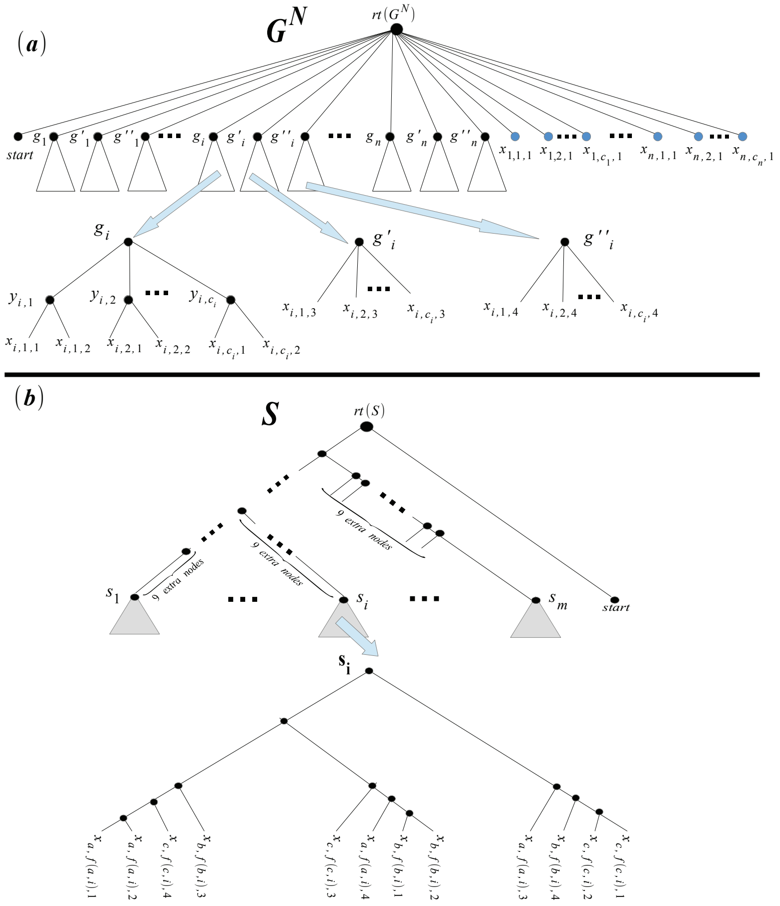

Fig. 2. Construction of non-binary gene tree and species tree. (a) Structure of the non-binary gene tree G^N. (b) Structure of the species tree S.

The remainder of this section is devoted to proving this claim which, in turn, would complete our proof for Theorem 1. We begin by explaining the main idea of the reduction and describing the association between the instances ϕ and λ, and then prove the forward and reverse directions of the claim.

3.3 Key insight

The main idea behind our reduction can be explained as follows: In the gene tree G^N, subtrees $G^N(g_i)$, $G^N(g'_i)$ and $G^N(g''_i)$ correspond to the element u_i, for each $1 \leq i \leq n$, while in the species tree the subtree $S(s_j)$ corresponds to the subset A_j, for each $1 \leq j \leq m$. Let G^B be any binary resolution of G^N. It can be shown that in any MPR of any optimal binary resolution G^B of G^N the following must hold: For each $i \in \{1, \ldots, n\}$, g_i (along with g'_i and g''_i) must map to an $S(s_j)$ for which $u_i \in A_j$.

Under these restrictions on the mappings, observe that if we were to solve the OGTR problem on G^N and S and then choose all those A_j's for which the subtree $S(s_j)$ has at least one of the g_i's mapping into it, then the set of chosen A_j's would cover all the elements of U.

The source of the optimization is that, due to the specific construction of the gene tree and species tree, it is more expensive (in terms of reconciliation cost) to use more $S(s_j)$'s for the mapping. Thus, all the g_i's (along with g_i''s and g_i'''s) must map to as few of the subtrees, $S(s_j)$'s, as possible. Recall that the OGTR problem optimizes the topology of the binary resolution G^B in such a way that its MPR with S has minimum reconciliation cost. Thus, the OGTR problem effectively optimizes the topology of G^B in a way that minimizes the total number of $S(s_j)$'s receiving mappings from the g_i's, g_i''s, or g_i'''s, yielding a set cover of smallest possible size. This is the key idea behind our reduction and we develop this idea further in the next subsection.

3.4 Proof of Claim 1

Forward direction. Let us assume that we have a YES answer for the M3SC instance ϕ. We will show how to create a binary resolution G^B of G^N whose MPR with S has reconciliation cost at most $10k + 39m - 12n$.

We first show how to resolve the subtrees $G^N(g_i)$, $G^N(g_i')$, and $G^N(g_i'')$, for $1 \leq i \leq n$. Recall that, for any fixed i, these three subtrees correspond to element u_i of U. The $y_{i,j}$'s in $G^N(g_i)$ correspond to the different occurrences of element u_i in the subsets from A. The same holds for the $x_{i,j,3}$'s in $G^N(g_i')$ and the $x_{i,j,4}$'s in $G^N(g_i'')$.

Suppose a solution to instance ϕ consists of the k subsets $A_{r(1)}, A_{r(2)}, \ldots, A_{r(k)}$. Since every element in U must be covered by at least one of these k subsets, we can designate a *covering subset* for each element $u_i \in U$, $1 \leq i \leq n$, chosen arbitrarily from among those subsets in the solution that contain u. Suppose that element u_i is assigned the covering subset A_j (so we must have $u_i \in A_j$ and $A_j \in \{A_{r(1)}, A_{r(2)}, \ldots, A_{r(k)}\}$). The subtree $G^N(g_i)$ will then be resolved as follows: The $y_{i,j}$ corresponding to the occurrence of u_i in A_j, i.e., $y_{i,f(i,j)}$, will be separated out as one of the two children of g_i. The other child of g_i will be the root of an arbitrary caterpillar tree on all the remaining $y_{i,j}$'s in $G^N(g_i)$. This is depicted in Figure 3(d). The subtrees $G^N(g_i')$ and $G^N(g_i'')$ are resolved similarly, except that in $G^N(g_i')$ the leaf node $x_{i,f(i,j),3}$ is separated out and in $G^N(g_i'')$ the leaf node $x_{i,f(i,j),4}$ is separated out. Thus, the resolution of $G^N(g_i)$, $G^N(g_i')$, and $G^N(g_i'')$ is done based on the assigned covering subset of element u_i. This is repeated for all i, where $1 \leq i \leq n$.

Next, we show how to resolve the root node of G^N to obtain G^B. The *start* node will become an outgroup to the rest of G^B. The backbone of the rest of G^B consists of an arbitrary caterpillar tree on k "leaf" nodes as shown in Figure 3(a). These k nodes are labeled $h_{r(1)}, \ldots h_{r(k)}$ and are the root nodes of k subtrees. Each of the k subtrees corresponds to one of the subsets $A_{r(1)}, A_{r(2)}, \ldots, A_{r(k)}$. In particular, subtree $G^B(h_{r(i)})$, for $1 \leq i \leq k$ corresponds to the subset $A_{r(i)}$. Each of the blue nodes and the subtrees rooted at the g_i's, g_i''s, and g_i'''s, for $1 \leq i \leq n$ will be included in one of these k subtrees. Specifically, the subtree $G^B(h_{r(j)})$ will include all those g_i's, g_i''s, and g_i'''s for which the covering subset of the corresponding u_i is $A_{r(j)}$. Since there may be 0, 1, 2, or 3 i's for which the covering subset of u_i is $A_{r(j)}$, the sizes of different $G^B(h_{r(j)})$ subtrees may vary. The structure of $G^B(h_{r(j)})$ when there are 3 $i's$ is depicted in Figure 3(b). The structure of $G^B(h_{r(j)})$ when there are only 1 or 2 such $i's$ is similar and is the induced subtree, on the relevant i's, of the full subtree for all 3 i's. As shown in the figure, note that each subtree $G^B(h_{r(j)})$ also includes exactly three blue nodes, corresponding to the three elements in $A_{r(j)}$. These three blue nodes are included even for cases where there are fewer than 3 i's. Thus, when there are 0 such i's, which can happen when the size of the minimum set cover for instance ϕ is less than k, the subtree $G^B(h_{r(j)})$ consists of the three blue nodes.

This results in the assignment of all g_i's, g'_i's, and g''_i's, for $1 \leq i \leq n$ to one of the subtrees $G^B(h_{r(j)})$, for $1 \leq j \leq k$. As discussed above, $3k$ out of the $3m$ blue nodes also get assigned in this process. The remaining $3m - 3k$ of the blue nodes are organized into an arbitrary caterpillar tree and added to the subtree $G^B(h_{r(k)})$ as shown in Figure 3(c).

This finishes our description of G^B. The following two lemmas imply the forward direction of Claim 1. The next lemma follows from the construction of G^B above.

Lemma 1. *Gene tree G^B is a binary resolution of G^N.*

It is not difficult to construct a DTL-scenario for G^B and S with cost exactly $10k + 39m - 12n$, yielding the following lemma.

Lemma 2. *Any MPR of G^B with S has reconciliation cost at most $10k + 39m - 12n$.*

Reverse direction. Conversely, let us assume that we have a YES answer for the OGTR instance λ with $l = 10k + 39m - 12n$. We will show that there exists a solution of size at most k for the set cover instance ϕ. We first characterize the structure of optimal resolutions and their most parsimonious reconciliations.

Lemma 3. *For any optimal binary resolution G^B of G^N there exists an MPR of G^B with S such that:*

1. *For any $i \in \{1, \ldots, n\}$, g_i, g'_i and g''_i map to the same subtree $S(s_j)$, where j is such that $u_i \in A_j$.*
2. *If there is a subtree $S(s_j)$ for which at least one of the nodes of G^B labeled g_i, g'_i, or g''_i, for any $i \in \{1, \ldots, n\}$, maps to a node in $S(s_j)$, then there exists an $i \in \{1, \ldots, n\}$ such that g_i, g'_i and g''_i all map to $S(s_j)$.*
3. *If g_i maps to a node in subtree $S(s_j)$, then g_i, g'_i, g''_i, and the three blue nodes corresponding to the elements in A_j are arranged in such a way that the subtree of G^B connecting these six nodes does not contain any transfer nodes.*
4. *If two nodes, say a and b map to different subtrees $S(s_j)$, for $1 \leq j \leq m$, then the path connecting them in G^B must contain at least one transfer event.*

Lemma 4. *For any optimal binary resolution G^B of G^N, all MPRs of G^B with S must be such that:*

1. *Each $G^B(g_i)$, $G^B(g'_i)$ and $G^B(g''_i)$, for $1 \leq i \leq n$, has exactly $(c_i - 1)$ transfer nodes, no duplications, and invokes no losses.*

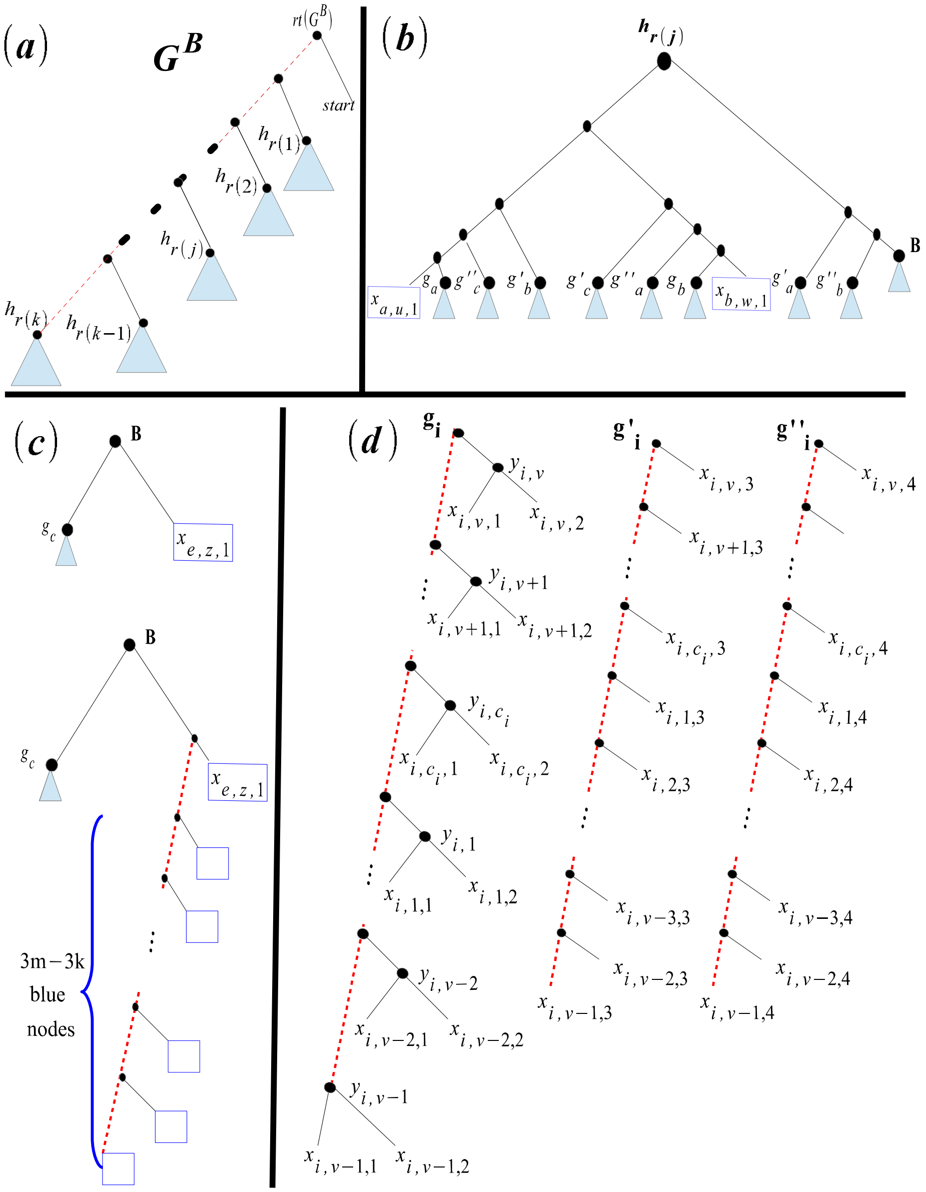

Fig. 3. Resolution of G^N into G^B. (a) The structure of the backbone of the gene tree G^B. (b) Structure of the subtree $h_{r(j)}$ for any $j \in \{1, \ldots, k\}$. (c) The two possible structures of the subtree with root B in $h_{r(j)}$. For any $j \in \{1, \ldots, k\}$, this subtree is as shown at the top of part (c) while, for $j = k$, it is as shown at the bottom and includes all the "remaining" $3m - 3k$ blue nodes. (d) The resolution of the g_i's, g_i''s, g_i'''s. In the figure, u_a, u_b, and u_c represent the three elements in $A_{r(j)}$, with $u = f(a, r(j))$, $w = f(b, r(j))$, and $z = f(c, r(j))$. In part (d), if the covering subset of element u_i is A_p, then v represents $f(i, p)$. The labels inside the blue boxes represent blue nodes.

2. *Each blue node that maps to an $S(s_j)$, $1 \leq j \leq m$, to which none of the g_i's map must be the recipient of a transfer edge.*

The next lemma implies the reverse direction and is based on the two lemmas above.

Lemma 5. *If there exists a binary resolution of G^N such that its MPR with S has reconciliation cost at most $10k + 39m - 12n$, then there exists a solution of size at most k for the M3SC instance ϕ.*

4 Extension to dated DTL reconciliation

An alternative model of DTL reconciliation has been proposed when the internal nodes of the species tree can be fully ordered in time [9]. We refer to this model as the *Dated-DTL* reconciliation model. Dated-DTL reconciliation makes use of the total order on the species nodes to ensure that the reconstructed optimal reconciliation is time-consistent. A key feature of this model is that it subdivides the species tree into different *time slices* [9] and then restricts transfer events to only occur within the same time slice.

We show how to assign divergence times to each node of the species tree. Observe that all subtrees $S(s_i)$, for each $i \in \{1 \ldots m\}$, have identical structure. All nodes at the same level in each $S(s_i)$ are assigned the same divergence time across all the subtrees. The rest of the nodes in S may be assigned arbitrary divergence times respecting the topology of S. It can be shown that there exists an optimal resolution of the gene tree for which an MPR exists that only invokes transfer events that respect the timing constraints of this dated species tree as required by the dated-DTL reconciliation model. This implies that, for our gadget, any optimal resolution of the gene tree under the undated DTL reconciliation model has the same minimum reconciliation cost as the dated-DTL reconciliation model.

Theorem 2. *The OGTR problem under the dated-DTL reconciliation model is NP-hard.*

5 Conclusion

In this work, we have shown that the OGTR problem, i.e., the problem of reconciling non-binary gene trees with binary species trees under the DTL reconciliation model, is NP-hard. Our reduction applies to both the undated and dated formulations of DTL-reconciliation and, furthermore, shows that the problem is NP-hard even for a biologically meaningful event cost assignment of 1, 2, and 4 for losses, duplications, and transfers, respectively. The uncertainty about its complexity has prevented the development of algorithms for the OGTR problem. This work will lead to the development of effective exact, approximate, and heuristic algorithms for this problem, making it possible to apply the powerful DTL reconciliation framework to non-binary gene trees. Interesting open problems include determining if efficient algorithms exist for the special case when the degree of each gene tree node is bounded above by a constant, and investigating the approximability of the dated and undated OGTR problems.

Funding: This work was supported in part by startup funds from the University of Connecticut to MSB.

References

1. Koonin, E.V.: Orthologs, paralogs, and evolutionary genomics. Annual Review of Genetics **39**(1) (2005) 309–338
2. Vilella, A.J., Severin, J., Ureta-Vidal, A., Heng, L., Durbin, R., Birney, E.: Ensemblcompara genetrees: Complete, duplication-aware phylogenetic trees in vertebrates. Genome Research **19**(2) (2009) 327–335
3. Chen, K., Durand, D., Farach-Colton, M.: Notung: dating gene duplications using gene family trees. In: RECOMB. (2000) 96–106
4. David, L.A., Alm, E.J.: Rapid evolutionary innovation during an archaean genetic expansion. Nature **469** (2011) 93–96
5. Durand, D., Halldórsson, B.V., Vernot, B.: A hybrid micro-macroevolutionary approach to gene tree reconstruction. J. Comput. Biol. **13**(2) (2006) 320–335
6. Burleigh, J.G., Bansal, M.S., Eulenstein, O., Hartmann, S., Wehe, A., Vision, T.J.: Genome-scale phylogenetics: Inferring the plant tree of life from 18,896 gene trees. Syst. Biol. **60**(2) (2011) 117–125
7. Scornavacca, C., Jacox, E., Szllosi, G.J.: Joint amalgamation of most parsimonious reconciled gene trees. Bioinformatics (in press)
8. Gorbunov, K.Y., Liubetskii, V.A.: Reconstructing genes evolution along a species tree. Molekuliarnaia Biologiia **43**(5) (2009) 946–958
9. Doyon, J.P., Scornavacca, C., Gorbunov, K.Y., Szöllosi, G.J., Ranwez, V., Berry, V.: An efficient algorithm for gene/species trees parsimonious reconciliation with losses, duplications and transfers. In: RECOMB-CG. (2010) 93–108
10. Tofigh, A., Hallett, M.T., Lagergren, J.: Simultaneous identification of duplications and lateral gene transfers. IEEE/ACM Trans. Comput. Biology Bioinform. **8**(2) (2011) 517–535
11. Bansal, M.S., Alm, E.J., Kellis, M.: Efficient algorithms for the reconciliation problem with gene duplication, horizontal transfer and loss. Bioinformatics **28**(12) (2012) 283–291
12. Stolzer, M., Lai, H., Xu, M., Sathaye, D., Vernot, B., Durand, D.: Inferring duplications, losses, transfers and incomplete lineage sorting with nonbinary species trees. Bioinformatics **28**(18) (2012) 409–415
13. Bansal, M.S., Alm, E.J., Kellis, M.: Reconciliation revisited: Handling multiple optima when reconciling with duplication, transfer, and loss. J. Comput. Biol. **20**(10) (2013) 738–754
14. Scornavacca, C., Paprotny, W., Berry, V., Ranwez, V.: Representing a set of reconciliations in a compact way. J. Bioinform. Comput. Biol. **11**(02) (2013) 1250025
15. Libeskind-Hadas, R., Wu, Y.C., Bansal, M.S., Kellis, M.: Pareto-optimal phylogenetic tree reconciliation. Bioinformatics **30**(12) (2014) i87–i95
16. Ovadia, Y., Fielder, D., Conow, C., Libeskind-Hadas, R.: The cophylogeny reconstruction problem is NP-complete. J. Comput. Biol. **18**(1) (2011) 59–65
17. Libeskind-Hadas, R., Charleston, M.: On the computational complexity of the reticulate cophylogeny reconstruction problem. J. Comput. Biol. **16** (2009) 105–117
18. Chang, W., Eulenstein, O.: Reconciling gene trees with apparent polytomies. In: Computing and Combinatorics, 12th Annual International Conference, COCOON 2006, Taipei, Taiwan, August 15-18, 2006, Proceedings. (2006) 235–244
19. Lafond, M., Swenson, K., El-Mabrouk, N.: An optimal reconciliation algorithm for gene trees with polytomies. In Raphael, B., Tang, J., eds.: Algorithms in Bioinformatics. Volume 7534 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2012) 106–122
20. Zheng, Y., Zhang, L.: Reconciliation with non-binary gene trees revisited. In Sharan, R., ed.: Research in Computational Molecular Biology. Volume 8394 of LNCS. (2014) 418–432
21. Karp, R.M.: Reducibility among combinatorial problems. In: Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York. (1972) 85–103