



Libraries and Learning Services

# University of Auckland Research Repository, ResearchSpace

## Version

This is the Accepted Manuscript version. This version is defined in the NISO recommended practice RP-8-2008 <http://www.niso.org/publications/rp/>

## Suggested Reference

van Iersel, L., Kelk, S., Lekić, N., & Linz, S. (2016). Satisfying ternary permutation constraints by multiple linear orders or phylogenetic trees. *Theoretical Computer Science*, 609(1), 1-21. doi: [10.1016/j.tcs.2015.06.021](https://doi.org/10.1016/j.tcs.2015.06.021)

## Copyright

This is an open-access article distributed under the terms of the [Creative Commons Attribution-NonCommercial-NoDerivatives](https://creativecommons.org/licenses/by-nc-nd/4.0/) License.

Items in ResearchSpace are protected by copyright, with all rights reserved, unless otherwise indicated. Previously published items are made available in accordance with the copyright policy of the publisher.

<https://www.elsevier.com/about/company-information/policies/sharing>

<http://www.sherpa.ac.uk/romeo/issn/0304-3975/>

<https://researchspace.auckland.ac.nz/docs/uoa-docs/rights.htm>

# Satisfying ternary permutation constraints by multiple linear orders or phylogenetic trees

Leo van Iersel <sup>\*</sup>   Steven Kelk <sup>†</sup>   Nela Lekić <sup>†</sup>   Simone Linz <sup>‡</sup>

May 21, 2015

## Abstract

A ternary permutation constraint satisfaction problem (CSP) is specified by a subset  $\Pi$  of the symmetric group  $S_3$ . An instance of such a problem consists of a set of variables  $V$  and a set of constraints  $\mathcal{C}$ , where each constraint is an ordered triple of distinct elements from  $V$ . The goal is to construct a linear ordering  $\alpha$  on  $V$  such that, for each constraint  $(a, b, c) \in \mathcal{C}$ , the ordering of  $a, b, c$  induced by  $\alpha$  is in  $\Pi$ . Excluding symmetries and trivial cases there are eleven such problems, and their complexity is well known. Here we consider the variant of the problem, denoted 2- $\Pi$ , where we are allowed to construct two linear orders  $\alpha$  and  $\beta$  and each constraint needs to be satisfied by at least one of the two. We give a full complexity classification of all eleven 2- $\Pi$  problems, observing that in the switch from one to two linear orders the complexity landscape changes quite abruptly and that hardness proofs become rather intricate. To establish the proofs, we use several computer-aided searches. Each such search returns a small instance with a unique solution for a given problem. We then focus on one of the eleven problems in particular, which is closely related to the 2-CATERPILLAR COMPATIBILITY problem in the phylogenetics literature. We show that this particular CSP remains hard on three linear orders, and also in the biologically relevant case when we swap three linear orders for three phylogenetic trees, yielding the 3-TREE COMPATIBILITY problem. Lastly, we give extremal results concerning the minimum number of trees required, in the worst case, to satisfy a set of rooted triplet constraints on  $n$  leaf labels.

**Keywords:** Permutation constraint satisfaction problem, linear orders, phylogenetic trees

## 1 Introduction

A *ternary permutation* constraint satisfaction problem (CSP), sometimes also known as an *ordering* CSP, is specified by a subset  $\Pi$  of the symmetric group  $S_3$ . An instance of such a problem consists of a set of variables  $V$  and a set of constraints  $\mathcal{C}$ , where each constraint is an ordered triple of distinct elements from  $V$ . The goal is to construct a linear order  $\alpha$  on  $V$  such that, for each constraint  $(a, b, c) \in \mathcal{C}$ , the ordering of  $a, b, c$  induced by  $\alpha$  is in  $\Pi$ . For example, if  $\Pi = \{123, 132\}$  then for each constraint  $(a, b, c)$  we require that  $\alpha(a) < \alpha(b) < \alpha(c)$  or  $\alpha(a) < \alpha(c) < \alpha(b)$ , which can be summarized as  $\alpha(a) < \min(\alpha(b), \alpha(c))$ . Excluding symmetries and trivial cases there are

---

<sup>\*</sup>Delft Institute of Applied Mathematics, Technische Universiteit Delft, Mekelweg 4, 2628CD, Delft, The Netherlands, l.j.v.iersel@gmail.com

<sup>†</sup>Department of Knowledge Engineering (DKE), Maastricht University, P.O. Box 616, 6200 MD Maastricht, The Netherlands, steven.kelk@maastrichtuniversity.nl, nela.lekic@maastrichtuniversity.nl

<sup>‡</sup>Department of Computer Science, The University of Auckland, Auckland, New Zealand, s.linz@auckland.ac.nz

eleven such problems, some of which have acquired specific names in the literature, such as *betweenness* [7] and *cyclic ordering* [9]. A full complexity classification of the 11 problems is given in [13] and summarized in Table 1. Due to their fundamental character these problems have stimulated quite some interest from the approximation [11], parameterized complexity [12] and algebra [2] communities.

In this article we consider the variant of the problem, denoted  $k$ - $\Pi$ , where we are allowed to construct  $k$  linear orderings and each constraint needs to be satisfied by at least one of them. In Section 3, we give a full complexity classification of all eleven 2- $\Pi$  problems, observing that in the switch from one to two linear orders the complexity landscape does not behave monotonically (see Table 1). We note that for a single linear ordering the polynomial-time variants can be solved with variations of topological sorting, and the hard variants can be proven NP-complete using fairly straightforward reductions [13]. In the case of two linear orderings all the polynomial-time variants are *trivially* solvable while, for the other variants, establishing the NP-completeness is much more challenging, requiring a wide array of novel gadgets and constructions.

Our motivation for studying the  $k$ - $\Pi$  problems is that, in practice, constraints may come from different sources, and it might not be known which constraints come from a common source. In such situations, it makes sense to construct different solutions that commonly satisfy all given constraints. A good example of a practical problem where this phenomenon occurs can be found in *phylogenetics*, a branch of computational biology concerned with the inference of evolutionary histories [23]. We discuss this phylogenetics problem in Sections 5-8.

Here we are given a set of *rooted triplets* which are leaf labeled, rooted binary trees on three leaves. Rooted triplets have a central and recurring role within phylogenetics due to the fact that they can be viewed as the atomic building blocks of larger evolutionary histories [23]. Determining whether all elements in a set  $\mathcal{R}$  of rooted triplets can simultaneously be embedded into a tree-like hypothesis of evolution, known as a phylogenetic tree, can be computed in polynomial-time by using the classical algorithm of Aho et al. [1]. If such a phylogenetic tree exists, then  $\mathcal{R}$  is called *compatible*. Frequently however such a tree does not exist and one might be interested in finding a maximum-size compatible subset of  $\mathcal{R}$ . This problem was first shown to be NP-hard by Bryant [4] as well as later again by [15, 25], and several exact and approximation algorithms have been developed [6, 25].

Recently, the impact of reticulation events, like hybridization and horizontal gene transfer, on the evolutionary history of certain groups of species is widely acknowledged. In the presence of reticulation, different parts of a genome often contain different tree-like evolutionary signals that need to be untangled [14, 18, 19]. To do this, a possible approach is the following. Instead of constructing one phylogenetic tree that embeds all triplets of a maximum-size compatible subset  $S$  of  $\mathcal{R}$  and ignoring all evolutionary information of the triplets that are in  $\mathcal{R}$  but not in  $S$ , it seems sensible to ask for the smallest integer  $k$  such that  $\mathcal{R}$  can be partitioned into at most  $k$  compatible subsets of triplets. For each such subset, one can use the algorithm of Aho et al. to find a phylogenetic tree, which represents the evolutionary history of a part of the genome. The collection of all these phylogenetic trees can also be used as input to methods that construct a leaf-labeled acyclic digraph (called a “rooted phylogenetic network”) that contains all the computed phylogenetic trees and hence represents the evolution of the whole genome, see e.g. [14]. This approach was introduced in [16] and the associated decision problem is called  $k$ -TREE COMPATIBILITY. In their paper, the authors show that 2-TREE COMPATIBILITY is NP-complete. We observe that the problem 2- $\Pi_1$  (which corresponds to the  $\Pi = \{123, 132\}$  example given earlier) is equivalent to the problem studied in [16], with one extra restriction: the two phylogenetic trees we construct must be “caterpillars”, yielding the 2-CATERPILLAR COMPATIBILITY problem. Our hardness result for 2- $\Pi_1$  thus supplements the hardness result of Linz et al. (and, as a spin-off result presented in Section 4, establishes a link to the literature

		1LO	2LO
$\Pi_0$ (linear ordering)	123	P	NPC
$\Pi_1$	123, 132	P	NPC
$\Pi_2$	123, 213, 231	P	P
$\Pi_3$	123, 231, 312, 321	P	P
$\Pi_4$	123, 231	NPC	NPC
$\Pi_5$ (betweenness)	123, 321	NPC	NPC
$\Pi_6$	123, 132, 231	NPC	NPC
$\Pi_7$ (circular ordering)	123, 231, 312	NPC	P
$\Pi_8$	$S_3 \setminus 123, 231$	NPC	P
$\Pi_9$ (non-betweenness)	$S_3 \setminus 123, 321$	NPC	NPC
$\Pi_{10}$	$S_3 \setminus 123$	NPC	P

Table 1: The complexity of the 11 ternary permutations CSPs, in the case of 1 linear order (1LO) (see [13]) and 2 linear orders (2LO) (this article).

on the dichromatic number problem [3]). To further extend this result we show in Section 5 that  $3\text{-}\Pi_1$  is hard and, building on this machinery,  $3\text{-TREE COMPATIBILITY}$  is also hard. As with many of the hardness results in this article we make heavy use of special gadgets that are unique solutions to the set of constraints that they imply. Such uniqueness gadgets could be of independent interest. To find these gadgets, we use several computer-aided searches that are described in Section 6. Each such search returns a small instance of one of the eleven  $k\text{-}\Pi$  problems and provides a “backbone structure” for the construction used in the associated proof showing that the problem is NP-complete. Like other proofs that were established with the assistance of a computer (e.g. [8, 22]), our results emphasize the potential importance of computers in establishing hardness results.

In Section 7, we then explore  $k\text{-TREE COMPATIBILITY}$  from an extremal perspective. In particular, we ask how many subsets (i.e. phylogenetic trees) are required, in the worst case, to satisfy a set of triplets on  $n$  leaf labels? Empirical experiments show that this function  $\tau(n)$  grows so slowly that the question arises: does there exist a constant  $c$  such that  $c$  trees are always sufficient? We prove that the answer is *no*, showing that  $\tau(n) \rightarrow \infty$  as  $n \rightarrow \infty$ . This shows that the  $k\text{-TREE COMPATIBILITY}$  problem does not become trivial for any  $k > 1$ , which supports the conjecture of Linz et al. that the problem is NP-complete for every  $k > 1$ . We also show a logarithmic upper bound. In Section 8, we complement the theoretical results described in Section 7 with the description of an Integer Linear Program (ILP) to compute  $\tau(n)$ .

We conclude with some open problems and future directions for research in Section 9.

## 2 Preliminaries

This section provides notation and terminology that is used in the remainder of the paper. Preliminaries in the context of phylogenetics are given in the second part of this section.

### 2.1 Ternary permutation constraint satisfaction problems

In this paper, we investigate eleven ternary permutation constraint satisfaction problems that are all based on subsets of the symmetric group on three elements, i.e.

$$S_3 = \{123, 132, 213, 231, 312, 321\}.$$

An instance  $\mathcal{I} = (V, \mathcal{C})$  of such a problem consists of a set  $V$  of *variables* and a set  $\mathcal{C}$  of *constraints*, where each constraint is an ordered triple  $(v_1, v_2, v_3)$  of three distinct variables of  $V$ .

Let  $\alpha : V \rightarrow \{1, 2, \dots, m\}$  be a linear ordering of  $V$ , where  $|V| = m$ . We sometimes write

$$(\alpha^{-1}(1), \alpha^{-1}(2), \dots, \alpha^{-1}(m))$$

to denote  $\alpha$ .

Let  $V$  and  $V'$  be two sets of variables such that  $V \cap V' = \emptyset$ . Furthermore, let  $\alpha$  be a linear ordering of  $V$ , and let  $\beta$  be a linear ordering of  $V'$ . We use  $\bar{\alpha}$  to denote the reversal of  $\alpha$ . Furthermore, for  $S \subseteq V$ , the *restriction of  $\alpha$  to  $S$*  is the linear ordering, say  $\gamma$ , on  $S$  with the property that  $\gamma(v_j) < \gamma(v_{j'})$  if and only if  $\alpha(v_j) < \alpha(v_{j'})$  for each pair of elements  $v_j, v_{j'} \in S$ . We denote  $\gamma$  by  $\alpha[S]$ . Now, let  $\gamma$  be a linear ordering of  $V \cup V'$  such that  $\gamma(v_j) < \gamma(v_{j'})$  if and only if  $\alpha(v_j) < \alpha(v_{j'})$  for each pair of elements  $v_j, v_{j'} \in V$ . Then  $\gamma$  is said to *preserve*  $\alpha$ . Let  $\gamma$  be a linear ordering of  $V \cup V'$  that preserves  $\alpha$  and  $\beta$  and has the property that  $\gamma(v_j) < \gamma(v_{j'})$  for each  $v_j \in V$  and  $v_{j'} \in V'$ . We use  $\alpha||\beta$  to denote  $\gamma$ . Intuitively,  $\gamma$  is the concatenation of  $\alpha$  followed by  $\beta$ .

Referring back to Table 1, let  $\Pi_i$  be a subset of  $S_3$  for some  $i \in \{0, 1, \dots, 10\}$ , and let  $\mathcal{I} = (V, \mathcal{C})$  be an instance of a ternary permutation constraint satisfaction problem. Furthermore, let  $\alpha$  be a linear ordering of  $V$ . We say that a constraint  $(v_1, v_2, v_3)$  in  $\mathcal{C}$  is  $\Pi_i$ -*satisfied* by  $\alpha$ , if there is a permutation  $\pi \in \Pi_i$  such that  $\alpha(v_{\pi(1)}) < \alpha(v_{\pi(2)}) < \alpha(v_{\pi(3)})$  where  $\pi$  is assumed to map positions to symbols. For example, if  $(v_1, v_2, v_3)$  is  $\Pi_5$ -satisfied by  $\alpha$ , then either  $\alpha(v_1) < \alpha(v_2) < \alpha(v_3)$  or  $\alpha(v_3) < \alpha(v_2) < \alpha(v_1)$ .

For each  $i \in \{0, 1, 2, \dots, 10\}$ , we are now in a position to define the following decision problem.

$k$ - $\Pi_i$

**Instance.** A finite set  $V$  of variables, a set  $\mathcal{C}$  of ordered triples of distinct variables from  $V$  and a positive integer  $k$ .

**Question.** Do there exist at most  $k$  linear orderings of  $V$  such that each constraint in  $\mathcal{C}$  is  $\Pi_i$ -satisfied by one of these orderings.

If the answer to an instance  $\mathcal{I}$  of  $k$ - $\Pi_i$  is ‘yes’, we say that  $\mathcal{I}$  is  $k$ - $\Pi_i$ -*satisfiable* (or  $\Pi_i$ -*satisfiable* for short if  $k = 1$ ).

Lastly, let  $\alpha$  be a linear ordering of a set  $V$  of variables. We say that  $\alpha$  *implies a constraint*  $(v_1, v_2, v_3)$  *under*  $\Pi_i$  precisely if  $(v_1, v_2, v_3)$  is  $\Pi_i$ -satisfied by  $\alpha$ . Furthermore, we use  $\mathcal{C}_{\Pi_i}(\alpha)$  to denote the set of all constraints that are implied by  $\alpha$  under  $\Pi_i$ .

## 2.2 Phylogenetics background

This section contains preliminaries in the context of phylogenetics. For a more thorough overview, we refer the interested reader to [23].

A *binary unrooted phylogenetic tree* of order  $n$  is a tree in which all internal vertices have degree 3 and which has  $n$  leaves that are bijectively labeled with elements in  $X = \{1, 2, \dots, n\}$ . An element in  $X$  is referred to as *taxon*. For two binary unrooted phylogenetic trees  $T$  and  $T'$ , we say that  $T$  *displays*  $T'$  if  $T'$  can be obtained from a subtree of  $T$  by suppressing degree-2 vertices.

A *binary rooted phylogenetic tree* of order  $n$  is a rooted tree in which all edges are directed away from the root which has out-degree 2, all internal vertices have in-degree 1 and out-degree 2, and which has  $n$  leaves that are bijectively labeled with elements in  $\{1, 2, \dots, n\}$ . If two leaves  $a$  and  $b$  of a rooted phylogenetic tree  $T$  are adjacent to the same parent, then  $\{a, b\}$  is called a

*cherry* of  $T$ . Furthermore, a binary rooted phylogenetic tree that has exactly one cherry is called a *caterpillar*. For two vertices  $u$  and  $v$  of a binary rooted phylogenetic tree  $T$ , we write  $u <_T v$  to denote that there exists a directed path from  $u$  to  $v$  in  $T$ . Moreover,  $\text{lca}_T(u, v)$  denotes the lowest common ancestor of  $u$  and  $v$  in  $T$ , i.e.  $\text{lca}_T(u, v)$  is the unique vertex  $w$  such that  $w <_T u$ ,  $w <_T v$  and there is no vertex  $w' \neq w$  such that  $w' <_T u$ ,  $w' <_T v$  and  $w <_T w'$ . We sometimes write  $\text{lca}(u, v)$  instead of  $\text{lca}_T(u, v)$  if it is clear from the context which tree is meant.

Now, let  $T$  be a rooted phylogenetic tree whose leaves are bijectively labeled with elements in  $X$ , and let  $Y$  be a subset of  $X$ . We call  $X$  the *leaf set* of  $T$  and denote it by  $L(T)$ . Furthermore, the *minimal rooted subtree* of  $T$  that connects all the leaves in  $Y$  is denoted by  $T(Y)$ . Lastly, the *restriction of  $T$  to  $Y$* , denoted by  $T|Y$ , is the rooted phylogenetic tree obtained from  $T(Y)$  by suppressing all degree-2 vertices apart from the root.

Next, we introduce a special type of binary rooted phylogenetic tree. A (*rooted*) *triplet* is a binary rooted phylogenetic tree on three leaves (see Figure 1). We say that a binary rooted phylogenetic tree  $T$  *displays* a triplet  $ab|c$  (or, equivalently,  $ba|c$ ) if  $\text{lca}_T(a, c) = \text{lca}_T(b, c) <_T \text{lca}_T(a, b)$ . Moreover, for a triplet  $ab|c$ , we call  $c$  the *witness* of  $ab|c$ . Now, let  $\mathcal{R}$  be a set of triplets. If there exists a rooted phylogenetic tree  $T$  such that each triplet in  $\mathcal{R}$  is displayed by  $T$ , we say that  $\mathcal{R}$  is *compatible* and, otherwise, we say that  $\mathcal{R}$  is *incompatible*. Note that the three triplets  $ab|c$ ,  $ac|b$ , and  $bc|a$  are pairwise incompatible.

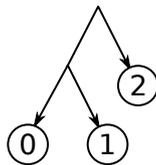


Figure 1: A rooted triplet  $01|2$ .

We are now in a position to state a decision problem that plays an important role in this paper and is strongly related to  $k\text{-}\Pi_1$ .

$k\text{-CATERPILLAR COMPATIBILITY}$

**Instance.** A set  $\mathcal{R}$  of rooted triplets.

**Question.** Do there exist at most  $k$  caterpillars such that each element in  $\mathcal{R}$  is displayed by at least one such caterpillar?

### 2.3 A note on $k\text{-}\Pi_1$ and $k\text{-CATERPILLAR COMPATIBILITY}$

A rooted triplet  $ab|c$  is a rooted binary phylogenetic tree on three leaves. It is important to note that  $a$  and  $b$  are indistinguishable from a phylogenetics point of view because a rooted binary phylogenetic tree  $T$  on three leaves  $a$ ,  $b$ , and  $c$  and with  $\{a, b\}$  being the cherry of  $T$  such that  $a$  is the right and  $b$  is the left child of the common parent of  $a$  and  $b$  is considered to be the same as the tree obtained from  $T$  by swapping  $a$  and  $b$ .

Let  $(1, 2, \dots, n-1, n)$  denote a caterpillar  $T$  on  $n$  leaves whose cherry is  $\{n-1, n\}$  and the path from each leaf labeled  $i \in \{1, 2, \dots, n-1\}$  to the root of  $T$  has length  $i$  while the path from the leaf labeled  $n$  to the root of  $T$  to has length  $n-1$ . Since  $n$  and  $n-1$  are indistinguishable,  $T$  naturally corresponds to the two linear ordering  $(n, n-1, n-2, \dots, 2, 1)$  and  $(n-1, n, n-2, \dots, 2, 1)$ . Moreover, each constraint  $(a, b, c)$  in an instance of  $k\text{-}\Pi_1$  can be satisfied by an ordering  $\alpha$  with  $\alpha(a) < \alpha(b) < \alpha(c)$  or  $\alpha(a) < \alpha(c) < \alpha(b)$  and corresponds to a rooted triplet  $bc|a$  that can be displayed by a caterpillar whose distance from  $a$  to the root is

shorter than the distance from  $b$  to the root and also shorter than the distance from  $c$  to the root. We summarize the strong relationship between the two problems in the following observation.

**Observation 1.** *The problem  $k$ - $\Pi_1$  is NP-complete if and only if  $k$ -CATERPILLAR COMPATIBILITY is NP-complete.*

### 3 Hardness results for all eleven CSP problems on two linear orderings

In this section, we settle the complexity of each ternary permutation CSP problem  $2$ - $\Pi_i$  with  $i \in \{0, 1, 2, \dots, 10\}$ . We start with the following observation.

**Observation 2.** *For each  $i \in \{2, 3, 7, 8, 10\}$ , the problem  $2$ - $\Pi_i$  is trivially polynomial-time solvable. In particular, every instance is a ‘yes’-instance.*

*Proof.* We give a proof for  $i = 2$  and note that a similar argument can be used to show that the result holds for each element in  $\{2, 3, 7, 8, 10\}$ . Let  $\mathcal{I} = (V, \mathcal{C})$  be an instance of  $2$ - $\Pi_2$ , and let  $\alpha$  be an ordering on  $V$ . Furthermore, let  $(v_1, v_2, v_3)$  be an element in  $\mathcal{C}$  that is not  $\Pi_2$ -satisfied by  $\alpha$ . Then  $\alpha(v_1) < \alpha(v_3) < \alpha(v_2)$ ,  $\alpha(v_3) < \alpha(v_1) < \alpha(v_2)$ , or  $\alpha(v_3) < \alpha(v_2) < \alpha(v_1)$ . In all three cases, a straightforward check shows that  $(v_1, v_2, v_3)$  is  $\Pi_2$ -satisfied by  $\bar{\alpha}$ . Hence, each constraint in  $\mathcal{C}$  is  $\Pi_2$ -satisfied by  $\alpha$  or  $\bar{\alpha}$ , thereby showing that  $\mathcal{I}$  is  $2$ - $\Pi_2$ -satisfiable. In particular, a solution can be found in polynomial time.  $\square$

The next observation is implicitly used throughout the remainder of this section.

**Observation 3.** *For each  $i \in \{0, 1, 2, \dots, 10\}$ , the problem  $2$ - $\Pi_i$  is in NP.*

*Proof.* Let  $\mathcal{I} = (V, \mathcal{C})$  be an instance of  $2$ - $\Pi_i$ , and let  $\alpha$  and  $\beta$  be two linear orderings of  $V$ . For each constraint  $(v_1, v_2, v_3)$  it can be verified in polynomial time whether or not it is  $\Pi_i$ -satisfied by  $\alpha$  or  $\beta$ .  $\square$

**Theorem 4.** *The problem  $2$ - $\Pi_0$  is NP-complete.*

*Proof.* To establish the result, we use a polynomial-time reduction from  $1$ - $\Pi_5$ . Let  $\mathcal{I} = (V, \mathcal{C})$  be an instance of  $1$ - $\Pi_5$ . Let  $\mathcal{C}'$  be the set of ternary constraints in which each constraint in  $\mathcal{C}$  is represented by two constraints. In particular, set

$$\mathcal{C}' = \bigcup_{(v_1, v_2, v_3) \in \mathcal{C}} \{(v_1, v_2, v_3), (v_3, v_2, v_1)\}.$$

Now, let  $\mathcal{I}' = (V, \mathcal{C}')$  be an instance of  $2$ - $\Pi_0$ . As  $|\mathcal{C}'| = 2|\mathcal{C}|$ , we have that  $\mathcal{I}'$  has polynomial size and that the reduction can be carried out in polynomial time. We now claim that  $\mathcal{I}$  is  $1$ - $\Pi_5$ -satisfiable if and only if  $\mathcal{I}'$  is  $2$ - $\Pi_0$ -satisfiable.

Suppose that  $\mathcal{I}$  is  $1$ - $\Pi_5$ -satisfiable. Let  $\alpha$  be a linear ordering of  $V$  that satisfies  $\mathcal{I}$ . Then, for each constraint  $(v_1, v_2, v_3)$  in  $\mathcal{C}$ , either  $\alpha(v_1) < \alpha(v_2) < \alpha(v_3)$  or  $\alpha(v_3) < \alpha(v_2) < \alpha(v_1)$ . In the first case,  $(v_1, v_2, v_3)$  is  $\Pi_0$ -satisfied by  $\alpha$  and  $(v_3, v_2, v_1)$  is  $\Pi_0$ -satisfied by  $\bar{\alpha}$ . In the second case,  $(v_1, v_2, v_3)$  is  $\Pi_0$ -satisfied by  $\bar{\alpha}$  and  $(v_3, v_2, v_1)$  is  $\Pi_0$ -satisfied by  $\alpha$ . Hence,  $\alpha$  and  $\bar{\alpha}$  are a solution to  $\mathcal{I}'$  and, so,  $\mathcal{I}'$  is  $2$ - $\Pi_0$ -satisfiable.

On the other hand, suppose that  $\mathcal{I}'$  is  $2$ - $\Pi_0$ -satisfiable. Let  $\alpha'$  and  $\beta'$  be two linear orderings of  $V$  that satisfy  $\mathcal{I}'$ , and let  $(v_1, v_2, v_3)$  be an element of  $\mathcal{C}$ . Then exactly one element in  $\{(v_1, v_2, v_3), (v_3, v_2, v_1)\}$  is  $\Pi_0$ -satisfied by  $\alpha'$ . Hence,  $\alpha'(v_1) < \alpha'(v_2) < \alpha'(v_3)$  or

$\alpha'(v_3) < \alpha'(v_2) < \alpha'(v_1)$ . Both cases imply that  $(v_1, v_2, v_3)$  is  $\Pi_5$ -satisfied by  $\alpha'$ . Hence,  $\alpha'$  is a solution to  $\mathcal{I}$  and, so  $\mathcal{I}$  is 1- $\Pi_5$ -satisfiable. The theorem now follows.  $\square$

**Theorem 5.** *The problem 2- $\Pi_1$  is NP-complete.*

*Proof.* Let  $\mathcal{I} = (V, \mathcal{C})$  be an instance of 2- $\Pi_0$ , where  $\mathcal{C} = \{C^1, C^2, \dots, C^n\}$ . Furthermore, for each  $i \in \{1, 2, \dots, n\}$ , let  $C^i = (v_1^i, v_2^i, v_3^i)$ , with  $\{v_1^i, v_2^i, v_3^i\} \subseteq V$ , and let  $v_d^i$  and  $v_e^i$  be two new variables, not contained in  $V$ , corresponding to each constraint. To show that the theorem holds, we reduce  $\mathcal{I}$  to an instance of 2- $\Pi_1$ . Set

$$V' = V \cup \{v_d^1, v_e^1, v_d^2, v_e^2, \dots, v_d^n, v_e^n\},$$

and set

$$\mathcal{C}' = \bigcup_{C^i \in \mathcal{C}} \{(v_1^i, v_2^i, v_d^i), (v_2^i, v_3^i, v_e^i), (v_e^i, v_1^i, v_2^i), (v_d^i, v_1^i, v_2^i), (v_1^i, v_e^i, v_d^i)\}.$$

Now, let  $\mathcal{I}' = (V', \mathcal{C}')$  be an instance of 2- $\Pi_1$ . Since  $|V'| = |V| + 2n$  and  $|\mathcal{C}'| = 5|\mathcal{C}|$ , the reduction can clearly be carried out in polynomial time, and  $\mathcal{I}'$  has polynomial size. The remainder of the proof consists of establishing that  $\mathcal{I}$  is 2- $\Pi_0$ -satisfiable if and only if  $\mathcal{I}'$  is 2- $\Pi_1$ -satisfiable.

First, suppose that  $\mathcal{I}$  is 2- $\Pi_0$ -satisfiable. Let  $\alpha$  and  $\beta$  be two linear orderings of  $V$  that satisfy  $\mathcal{I}$ . Let

$$W = \{v_d^i, v_e^i : C^i \text{ is } \Pi_0\text{-satisfied by } \alpha\}$$

and, similarly, let

$$W' = \{v_d^i, v_e^i : C^i \text{ is not } \Pi_0\text{-satisfied by } \alpha\}.$$

Furthermore, let  $\gamma$  be an arbitrary linear ordering of  $W$ , and let  $\gamma'$  be an arbitrary linear ordering of  $W'$ . Then  $\alpha' = \gamma \parallel \alpha \parallel \gamma$  and  $\beta' = \gamma \parallel \beta \parallel \gamma'$  are two linear orderings on  $V'$ . Now, for each  $C^i = (v_1^i, v_2^i, v_3^i)$  that is  $\Pi_0$ -satisfied by  $\alpha$  (resp.  $\beta$ ), the three constraints  $(v_1^i, v_2^i, v_d^i)$ ,  $(v_2^i, v_3^i, v_e^i)$ , and  $(v_1^i, v_e^i, v_d^i)$  are  $\Pi_1$ -satisfied by  $\alpha'$  (resp.  $\beta'$ ) while the two constraints  $(v_e^i, v_1^i, v_2^i)$  and  $(v_d^i, v_1^i, v_2^i)$  are  $\Pi_1$ -satisfied by  $\beta'$  (resp.  $\alpha'$ ). Hence  $\mathcal{I}'$  is 2- $\Pi_1$ -satisfiable.

Second, suppose that  $\mathcal{I}'$  is 2- $\Pi_1$ -satisfiable. Let  $\alpha'$  and  $\beta'$  be two linear orderings of  $V'$  that satisfy  $\mathcal{I}'$ . Assume that, for some  $i \in \{1, 2, \dots, n\}$ , the two constraints  $(v_1^i, v_2^i, v_d^i)$  and  $(v_2^i, v_3^i, v_e^i)$  are not both  $\Pi_1$ -satisfied by exactly one of  $\alpha'$  and  $\beta'$ . Then without loss of generality, we may assume that  $(v_1^i, v_2^i, v_d^i)$  is  $\Pi_1$ -satisfied by  $\alpha'$  and that  $(v_2^i, v_3^i, v_e^i)$  is  $\Pi_1$ -satisfied by  $\beta'$ . (The high-level idea is to show that, if this happens,  $v_1^i$  is necessarily sandwiched between  $v_d^i$  and  $v_e^i$  in both linear orderings, which in turn will contradict that  $(v_1^i, v_e^i, v_d^i)$  is  $\Pi_1$ -satisfied by  $\alpha'$  or  $\beta'$ . From this contradiction we will then conclude that  $(v_1^i, v_2^i, v_d^i)$  and  $(v_2^i, v_3^i, v_e^i)$  are both  $\Pi_1$ -satisfied by the same linear order, implying that  $(v_1^i, v_2^i, v_3^i)$  is  $\Pi_0$ -satisfied by that linear order.)

Since  $\beta'(v_2^i) < \beta'(v_e^i)$ , it follows that  $(v_e^i, v_1^i, v_2^i)$  is  $\Pi_1$ -satisfied by  $\alpha'$ . Similarly, since  $\alpha'(v_1^i) < \alpha'(v_d^i)$ , it follows that  $(v_d^i, v_1^i, v_2^i)$  is  $\Pi_1$ -satisfied by  $\beta'$ . Moreover, since  $\alpha'(v_e^i) < \alpha'(v_1^i)$  and  $\beta'(v_d^i) < \beta'(v_1^i)$ , this implies that  $(v_1^i, v_e^i, v_d^i)$  is neither  $\Pi_1$ -satisfied by  $\alpha'$  nor  $\beta'$ . Thus,  $(v_1^i, v_2^i, v_d^i)$  and  $(v_2^i, v_3^i, v_e^i)$  are both  $\Pi_1$ -satisfied by either  $\alpha'$  or  $\beta'$ . Hence, we have  $\alpha'(v_1^i) < \alpha'(v_2^i) < \alpha'(v_3^i)$  or  $\beta'(v_1^i) < \beta'(v_2^i) < \beta'(v_3^i)$ . It now follows that  $\mathcal{I}$  is 2- $\Pi_0$ -satisfied by the two linear orderings  $\alpha'$  restricted to  $V$  and  $\beta'$  restricted to  $V$ . The theorem now follows.  $\square$

**Theorem 6.** *The problem 2- $\Pi_4$  is NP-complete.*

*Proof.* To establish the result, we use a polynomial-time reduction from 1- $\Pi_9$ . Let  $\mathcal{I} = (V, \mathcal{C})$  be an instance of 1- $\Pi_9$ . Let  $\mathcal{C}'$  be the set of ternary constraints in which each constraint in  $\mathcal{C}$  is represented by two constraints. In particular, set

$$\mathcal{C}' = \bigcup_{(v_1, v_2, v_3) \in \mathcal{C}} \{(v_2, v_1, v_3), (v_2, v_3, v_1)\}.$$

Now, let  $\mathcal{I}' = (V, \mathcal{C}')$  be an instance of  $2\text{-}\Pi_4$ . As  $|\mathcal{C}'| = 2|\mathcal{C}|$ , we have that  $\mathcal{I}'$  has polynomial size and that the reduction can be carried out in polynomial time. We now claim that  $\mathcal{I}$  is  $1\text{-}\Pi_9$ -satisfiable if and only if  $\mathcal{I}'$  is  $2\text{-}\Pi_4$ -satisfiable.

Suppose that  $\mathcal{I}$  is  $1\text{-}\Pi_9$ -satisfiable. Let  $\alpha$  be a linear ordering of  $V$  that satisfies  $\mathcal{I}$ . It follows that, for each constraint  $(v_1, v_2, v_3)$  in  $\mathcal{C}$ , either  $\alpha(v_2) < \alpha(v_i) < \alpha(v_j)$ , or  $\alpha(v_{i'}) < \alpha(v_{j'}) < \alpha(v_2)$  with  $\{i, j\} = \{i', j'\} = \{1, 3\}$ . Moreover, in both cases, it is easily checked that exactly one of  $(v_2, v_1, v_3)$  and  $(v_2, v_3, v_1)$  is  $\Pi_4$ -satisfied by  $\alpha$  while the other constraint is  $\Pi_4$ -satisfied by  $\bar{\alpha}$ . Hence,  $\alpha$  and  $\bar{\alpha}$  are a solution to  $\mathcal{I}'$  and, so,  $\mathcal{I}'$  is  $2\text{-}\Pi_4$ -satisfiable.

Now, suppose that  $\mathcal{I}'$  is  $2\text{-}\Pi_4$ -satisfiable. Let  $\alpha'$  and  $\beta'$  be two linear orderings of  $V$  that satisfy  $\mathcal{I}'$ , and let  $(v_1, v_2, v_3)$  be an element of  $\mathcal{C}$ . Then exactly one element in  $\{(v_2, v_1, v_3), (v_2, v_3, v_1)\}$  is  $\Pi_4$ -satisfied by  $\alpha'$ . In particular, this implies that exactly one of the following holds:

- (i)  $\alpha'(v_2) < \alpha'(v_1) < \alpha'(v_3)$ .
- (ii)  $\alpha'(v_1) < \alpha'(v_3) < \alpha'(v_2)$ ,
- (iii)  $\alpha'(v_2) < \alpha'(v_3) < \alpha'(v_1)$ , or
- (iv)  $\alpha'(v_3) < \alpha'(v_1) < \alpha'(v_2)$ .

Regardless of which of (i)-(iv) holds,  $(v_1, v_2, v_3)$  is  $\Pi_9$ -satisfied by  $\alpha'$ . Hence,  $\alpha'$  is a solution to  $\mathcal{I}$ . The theorem now follows.  $\square$

**Lemma 7.** *Let  $V = \{1, 2, 3, 4, 5\}$ , and let  $\gamma = (1, 2, 3, 4, 5)$  and  $\gamma' = (5, 2, 3, 4, 1)$  be two linear orderings of  $V$ . Then the instance  $\mathcal{I} = (V, \mathcal{C}_{\Pi_5}(\gamma) \cup \mathcal{C}_{\Pi_5}(\gamma'))$  of  $2\text{-}\Pi_5$  has a unique solution (up to reversal). Moreover, each solution of  $\mathcal{I}$  consists of one element from  $\{\gamma, \bar{\gamma}\}$  and one element from  $\{\gamma', \bar{\gamma}'\}$ .*

*Proof.* Computational proof (see Section 6).  $\square$

**Theorem 8.** *The problem  $2\text{-}\Pi_5$  is NP-complete.*

*Proof.* Throughout the proof, let  $\gamma = (1, 2, 3, 4, 5)$  and  $\gamma' = (5, 2, 3, 4, 1)$  be two linear orderings of  $\{1, 2, 3, 4, 5\}$ . Let  $\mathcal{I} = (V, \mathcal{C})$  be an instance of  $1\text{-}\Pi_5$ , where  $\mathcal{C} = \{C^1, C^2, \dots, C^n\}$ . Furthermore, for each  $i \in \{1, 2, \dots, n\}$ , let  $C^i = (v_1^i, v_2^i, v_3^i)$ , with  $\{v_1^i, v_2^i, v_3^i\} \subseteq V$ , and let  $v_d^i$  and  $v_e^i$  be two new variables, not contained in  $V$ , for each constraint. To show that the theorem holds, we reduce  $\mathcal{I}$  to an instance of  $2\text{-}\Pi_5$ . Let  $D = \{v_d^1, v_e^1, v_d^2, v_e^2, \dots, v_d^n, v_e^n\}$ , and let

$$V' = V \cup D \cup \{1, 2, 3, 4, 5\},$$

where  $\{1, 2, 3, 4, 5\}$  neither intersects with  $D$  nor  $V$ . Furthermore, we define the following four new sets of constraints:

- (i)  $\mathcal{C}_1 = \mathcal{C}_{\Pi_5}(\gamma) \cup \mathcal{C}_{\Pi_5}(\gamma')$ ,
- (ii)  $\mathcal{C}_2 = \bigcup_{C^i \in \mathcal{C}} \{(v_1^i, v_d^i, v_3^i), (v_1^i, v_e^i, v_3^i), (v_d^i, v_2^i, v_e^i)\}$ ,
- (iii)  $\mathcal{C}_3 = \bigcup_{v_j \in V} \{(3, v_j, 4), (4, v_j, 5), (1, v_j, 2), (1, v_j, 3)\}$ , and
- (iv)  $\mathcal{C}_4 = \bigcup_{i \in \{1, 2, \dots, n\}} \{(2, v_d^i, 3), (2, v_e^i, 3), (1, v_d^i, 2), (1, v_e^i, 2), (4, v_d^i, 5), (4, v_e^i, 5), (3, v_d^i, 5), (3, v_e^i, 5)\}$ .

Now, let  $\mathcal{C}' = \mathcal{C}_1 \cup \mathcal{C}_2 \cup \mathcal{C}_3 \cup \mathcal{C}_4$ , and let  $\mathcal{I}' = (V', \mathcal{C}')$  be an instance of 2- $\Pi_5$ . Since  $\mathcal{C}_1$  contains a constant number of constraints it is easily checked that  $\mathcal{I}'$  has size polynomial in  $|V|$  and  $n$  and, so, the reduction can be carried out in polynomial time. We now claim that  $\mathcal{I}$  is 1- $\Pi_5$ -satisfiable if and only if  $\mathcal{I}'$  is 2- $\Pi_5$ -satisfiable.

First, suppose that  $\mathcal{I}$  is 1- $\Pi_5$ -satisfiable. Let  $\alpha$  be a linear ordering of  $V$  that satisfies  $\mathcal{I}$ . Let  $\delta$  be a linear ordering of  $V \cup D$  such that

$$\delta(v_1^i) < \delta(v_d^i) < \delta(v_2^i) < \delta(v_e^i) < \delta(v_3^i) \text{ or } \delta(v_3^i) < \delta(v_e^i) < \delta(v_2^i) < \delta(v_d^i) < \delta(v_1^i)$$

for each  $i \in \{1, 2, \dots, n\}$ . Since  $\alpha$  is a solution to  $\mathcal{I}$ , note that  $\delta$  exists.

Now, let

$$\alpha' = (1, 2, 3, 4, 5) \parallel \delta,$$

and let

$$\beta' = (5, 2) \parallel \delta[D] \parallel (3) \parallel \alpha \parallel (4, 1)$$

be two linear orderings of  $V'$ . We next argue that each constraint in  $\mathcal{C}'$  is  $\Pi_5$ -satisfied by  $\alpha'$  or  $\beta'$ . Since  $\alpha'$  preserves  $\gamma$  and since  $\beta'$  preserves  $\gamma'$ , it follows that each constraint in  $\mathcal{C}_1$  is  $\Pi_5$ -satisfied by  $\alpha'$  or  $\beta'$ . Furthermore, for each  $C^i \in \mathcal{C}$ , the three corresponding constraints in  $\mathcal{C}_2$  are, by construction,  $\Pi_5$ -satisfied by  $\alpha'$ . Turning to the constraints in  $\mathcal{C}_3$ , we observe that  $\max(\beta'(2), \beta'(3), \beta'(5)) < \beta'(v_j)$  and  $\beta'(v_j) < \min(\beta'(1), \beta'(4))$  and, hence, all four constraints that correspond to  $v_j$  in  $\mathcal{C}_3$  are  $\Pi_5$ -satisfied by  $\beta'$ . Similarly, for  $k \in \{d, e\}$ , a straightforward check shows that all eight constraints that correspond to  $v_k^i$  in  $\mathcal{C}_4$  are  $\Pi_5$ -satisfied by  $\beta'$ . Now, as each constraint in  $\mathcal{C}'$  is  $\Pi_5$ -satisfied by  $\alpha'$  or  $\beta'$ , we deduce that  $\mathcal{I}'$  is 2- $\Pi_5$ -satisfiable.

Second, suppose that  $\mathcal{I}'$  is 2- $\Pi_5$ -satisfiable. Let  $\alpha'$  and  $\beta'$  be two linear orderings of  $V'$  that satisfy  $\mathcal{I}'$ . Note that, by Lemma 7, each solution to the instance  $(\{1, 2, 3, 4, 5\}, \mathcal{C}_1)$  of 2- $\Pi_5$  consists of one element from  $\{\gamma, \bar{\gamma}\}$  and one element from  $\{\gamma', \bar{\gamma}'\}$ . Assume for the time being that  $\alpha'$  preserves  $\gamma$  and that  $\beta'$  preserves  $\gamma'$ . The high-level idea is to use the  $\mathcal{C}_3$  and  $\mathcal{C}_4$  constraints, acting on the “scaffolding” of  $\gamma$  and  $\gamma'$ , to make it impossible for  $\beta'$  to  $\Pi_5$ -satisfy the  $\mathcal{C}_2$  constraints. This forces the  $\mathcal{C}_2$  constraints to all be satisfied by  $\alpha'$ , which in turn ensures that each constraint  $(v_1^i, v_2^i, v_3^i)$  is  $\Pi_5$ -satisfied by  $\alpha'$ . We will exploit the fact that, on one hand, in  $\gamma$  the region between 3 and 4 has empty intersection with the regions between 4 and 5, between 1 and 2, and between 1 and 3 respectively. On the other hand in  $\gamma'$  the region between 3 and 4 intersects with *all* these three other regions. Consequently the  $\mathcal{C}_3$  constraints force, in  $\beta'$ , all the elements of  $V$  to be between 3 and 4. In a similar way the  $\mathcal{C}_4$  constraints force, in  $\beta'$ , all the elements of  $D$  to be between 2 and 3. Hence, in  $\beta'$  all the elements of  $D$  are on one side of 3, and all elements of  $V$  are on the other side of 3. Such an arrangement can never  $\Pi_5$ -satisfy constraints of type  $\mathcal{C}_2$ , where elements from  $V$  are required to be sandwiched between elements of  $D$ . This eliminates  $\beta'$  from further consideration, as required.

In more detail, assume that  $(3, v_j, 4)$  is  $\Pi_5$ -satisfied by  $\alpha'$  for some  $v_j \in V$ . Then each constraint in  $\{(4, v_j, 5), (1, v_j, 2), (1, v_j, 3)\}$  and, hence,  $(3, v_j, 4)$  is  $\Pi_5$ -satisfied by  $\beta'$ . On the other hand, assume that  $(3, v_j, 4)$  is not  $\Pi_5$ -satisfied by  $\alpha'$  for some  $v_j \in V$ . Then,  $(3, v_j, 4)$  is  $\Pi_5$ -satisfied by  $\beta'$ . Thus, regardless of whether  $(3, v_j, 4)$  is  $\Pi_5$ -satisfied by  $\alpha'$  or not, we have  $\beta'(3) < \beta'(v_j) < \beta'(4)$ . Similarly, assume that  $(2, v_k^i, 3)$  is  $\Pi_5$ -satisfied by  $\alpha'$  for some  $k \in \{d, e\}$  and  $i \in \{1, 2, \dots, n\}$ . Then each constraint in  $\{(1, v_k^i, 2), (4, v_k^i, 5), (3, v_k^i, 5)\}$  and, hence,  $(2, v_k^i, 3)$  is  $\Pi_5$ -satisfied by  $\beta'$ . On the other hand, assume that  $(2, v_k^i, 3)$  is not  $\Pi_5$ -satisfied by  $\alpha'$  for some  $v_k^i \in V$ . Then,  $(2, v_k^i, 3)$  is  $\Pi_5$ -satisfied by  $\beta'$ . Thus, regardless of whether  $(2, v_k^i, 3)$  is  $\Pi_5$ -satisfied by  $\alpha'$  or not, we have  $\beta'(2) < \beta'(v_k^i) < \beta'(3)$ . In summary, it follows that each constraint in  $\mathcal{C}_3 \cup \mathcal{C}_4$  is  $\Pi_5$ -satisfied by  $\beta'$ . Now, since  $\beta'(v_k^i) < \beta'(3) < \beta'(v_j)$  for each  $i \in \{1, 2, \dots, n\}$ ,  $k \in \{d, e\}$ , and  $v_j \in V$ , it follows that, for each  $C^i \in \mathcal{C}$ , the three constraints in  $\{(v_1^i, v_d^i, v_3^i), (v_1^i, v_e^i, v_3^i), (v_d^i, v_2^i, v_e^i)\}$  are  $\Pi_5$ -satisfied by  $\alpha'$ . It is now straightforward to check

that  $\alpha'(v_1^i) < \alpha'(v_2^i) < \alpha'(v_3^i)$  or  $\alpha'(v_3^i) < \alpha'(v_2^i) < \alpha'(v_1^i)$ . Hence,  $\alpha'[V]$  is a linear ordering of  $V$  that 1- $\Pi_5$ -satisfies each constraint in  $\mathcal{C}$  and, therefore, we have that  $\mathcal{I}$  is 1- $\Pi_5$ -satisfiable. We complete the proof of the converse by noting that a constraint is  $\Pi_5$ -satisfied by an ordering if and only if it is  $\Pi_5$ -satisfied by the reversal of that ordering. Hence, symmetrical arguments can be used to show that  $\mathcal{I}$  is 1- $\Pi_5$ -satisfiable if  $\alpha'$  preserves  $\bar{\gamma}$  (rather than  $\gamma$ ) and/or  $\beta'$  preserves  $\bar{\gamma}'$  (rather than  $\gamma'$ .) The theorem now follows by combining both cases.  $\square$

**Lemma 9.** *Let  $V = \{1, 2, 3, 4\}$ , and let  $\gamma = (1, 2, 3, 4)$  and  $\gamma' = (2, 4, 1, 3)$  be two linear orderings of  $V$ . Then  $\gamma$  and  $\gamma'$  form the unique solution to the instance  $(V, \mathcal{C}_{\Pi_6}(\gamma) \cup \mathcal{C}_{\Pi_6}(\gamma'))$  of 2- $\Pi_6$ .*

*Proof.* Computational proof (see Section 6).  $\square$

**Theorem 10.** *The problem 2- $\Pi_6$  is NP-complete.*

*Proof.* Throughout the proof, let  $\gamma = (1, 2, 3, 4)$  and  $\gamma' = (2, 4, 1, 3)$  be two linear orderings of  $\{1, 2, 3, 4\}$ . Let  $\mathcal{I} = (V, \mathcal{C})$  be an instance of 2- $\Pi_1$  with  $V = \{v_1, v_2, \dots, v_m\}$  and  $\mathcal{C} = \{C^1, C^2, \dots, C^n\}$ . Furthermore, for each  $i \in \{1, 2, \dots, n\}$ , let  $C^i = (v_1^i, v_2^i, v_3^i)$ , with  $\{v_1^i, v_2^i, v_3^i\} \subseteq V$ . Lastly, let  $D = \{1_{v_1}, 1_{v_2}, \dots, 1_{v_m}\}$  such that  $D \cap \{1, 2, 3, 4\} = \emptyset$ . To show that the result holds, we reduce  $\mathcal{I}$  to an instance  $\mathcal{I}'$  of 2- $\Pi_6$  in the following way. For each  $v_j \in V$ , we use  $\mathcal{C}(v_j)$  to denote the set obtained from  $\mathcal{C}_{\Pi_6}(\gamma) \cup \mathcal{C}_{\Pi_6}(\gamma')$  by replacing each occurrence of the variable 1 with  $1_{v_j}$ . Let  $V' = D \cup \{2, 3, 4\}$  be a set of variables. We next define two new sets of constraints. In particular, let

$$\mathcal{C}_1 = \bigcup_{v_j \in V} \mathcal{C}(v_j),$$

and let

$$\mathcal{C}_2 = \bigcup_{C^i \in \mathcal{C}} \{(1_{v_1^i}, 1_{v_2^i}, 1_{v_3^i}), (1_{v_1^i}, 1_{v_3^i}, 1_{v_2^i}), (1_{v_1^i}, 1_{v_2^i}, 3)\},$$

where each of  $1_{v_1^i}$ ,  $1_{v_2^i}$ , and  $1_{v_3^i}$  is an element in  $D$ .

Now, let  $\mathcal{I}' = (V', \mathcal{C}_1 \cup \mathcal{C}_2)$  and observe that each constraint in  $\mathcal{C}_1 \cup \mathcal{C}_2$  consists indeed of three elements in  $V'$ . Moreover, since  $\mathcal{C}_1$  contains a number of constraints that is polynomial in  $m$ , it is easily checked that  $\mathcal{I}'$  has size polynomial in  $m$  and  $n$  and, so, the reduction can be carried out in polynomial time. To complete the proof, we show that  $\mathcal{I}$  is 2- $\Pi_1$ -satisfiable if and only if  $\mathcal{I}'$  is 2- $\Pi_6$ -satisfiable.

First, suppose that  $\mathcal{I}$  is 2- $\Pi_1$ -satisfiable. Then there exist two linear orderings  $\alpha$  and  $\beta$  on  $V$  such that each  $C^j \in \mathcal{C}$  is  $\Pi_1$ -satisfied by  $\alpha$  or  $\beta$ . Let  $\alpha_1$  be the linear ordering of  $D$  obtained from  $\alpha$  by replacing each  $v_j$  with  $1_{v_j}$  and, similarly, let  $\beta_1$  be the linear ordering of  $D$  obtained from  $\beta$  by replacing each  $v_j$  with  $1_{v_j}$ . Now, let

$$\alpha' = \alpha_1 \parallel (2, 3, 4),$$

and let

$$\beta' = (2, 4) \parallel \beta_1 \parallel (3)$$

be two linear orderings on  $V'$ . We next show that each constraint in  $\mathcal{C}_1 \cup \mathcal{C}_2$  is  $\Pi_6$ -satisfied by  $\alpha'$  or  $\beta'$ . Since no constraint in  $\mathcal{C}_1$  contains two elements of  $D$ , it follows by Lemma 9 and regarding 1 as a placeholder for  $\alpha_1$  (resp.  $\beta_1$ ), that each constraint in  $\mathcal{C}_1$  is  $\Pi_6$ -satisfied by  $\alpha'$  or  $\beta'$ . Turning to the constraints in  $\mathcal{C}_2$ , we have that, if  $C^i \in \mathcal{C}$  is  $\Pi_1$ -satisfied by  $\alpha$ , then the first two constraints that correspond to  $C^i$  in  $\mathcal{C}_2$  are  $\Pi_6$ -satisfied by  $\alpha'$  while, if  $C^i$  is  $\Pi_1$ -satisfied by  $\beta$ , then the first two constraints that correspond to  $C^i$  in  $\mathcal{C}_2$  are  $\Pi_6$ -satisfied by  $\beta'$ . Moreover, since  $\alpha(v_1^i) < \alpha(v_2^i)$  or  $\beta(v_1^i) < \beta(v_2^i)$ , we have  $\alpha'(1_{v_1^i}) < \alpha'(1_{v_2^i}) < \alpha'(3)$  or  $\beta'(1_{v_1^i}) < \beta'(1_{v_2^i}), \beta'(3)$ . It follows that

$(1_{v_1^i}, 1_{v_2^i}, 3)$  is  $\Pi_6$ -satisfied by  $\alpha'$  or  $\beta'$ . Now, as each constraint in  $\mathcal{C}_1 \cup \mathcal{C}_2$  is  $\Pi_6$ -satisfied by  $\alpha'$  or  $\beta'$ , we deduce that  $\mathcal{I}'$  is 2- $\Pi_6$ -satisfiable.

Second, suppose that  $\mathcal{I}'$  is 2- $\Pi_6$ -satisfiable. Then, there exist two linear orderings  $\alpha'$  and  $\beta'$  on  $V'$  such that each constraint in  $\mathcal{C}_1 \cup \mathcal{C}_2$  is  $\Pi_6$ -satisfied by  $\alpha'$  or  $\beta'$ . The high-level idea is as follows. The  $\mathcal{C}_1$  constraints ensure that one of  $\alpha'$  and  $\beta'$ , say  $\alpha'$ , preserves  $\gamma = (1, 2, 3, 4)$  with all the  $1_{v_j}$  elements, which represent the elements in  $V$ , occupying the space where symbol 1 was originally found. Similarly,  $\beta'$  then preserves  $\gamma' = (2, 4, 1, 3)$  with the  $1_{v_j}$  elements occupying the place of symbol 1. In both orderings symbol 3 is to the right of all the  $1_{v_j}$  elements. This means that the  $(1_{v_1^i}, 1_{v_2^i}, 3)$  constraint in  $\mathcal{C}_2$  can only be  $\Pi_6$ -satisfied when  $1_{v_1^i}$  is to the left of both  $1_{v_2^i}$  and 3, in at least one of the orderings. Then the first two constraints in  $\mathcal{C}_2$  ensure that, in at least one of the two orderings,  $1_{v_1^i}$  is to the left of both  $1_{v_2^i}$  and  $1_{v_3^i}$  i.e. that ordering  $\Pi_1$ -satisfies each original  $(v_1^i, v_2^i, v_3^i)$  constraint, and we are done. (This is because the only other way for the first two constraints in  $\mathcal{C}_2$  to be satisfied is if one of the two orderings induces the explicit ordering  $1_{v_2^i} 1_{v_3^i} 1_{v_1^i}$  and the other  $1_{v_3^i} 1_{v_2^i} 1_{v_1^i}$ , but this is blocked by the fact that  $1_{v_1^i}$  is to the left of  $1_{v_2^i}$  in at least one of the orderings.)

The detailed argument is as follows. We show that, for each  $1_{v_j} \in D$  with  $j \in \{1, 2, \dots, m\}$ , we have  $\psi'(1_{v_j}) < \psi'(2)$  and  $\omega'(4) < \omega'(1_{v_j}) < \omega'(3)$  with  $\{\alpha', \beta'\} = \{\psi', \omega'\}$ . For a contradiction, assume that this is not the case. Then, there exists an element  $1_{v_{j'}} \in D$ , such that one of the followings holds:

- (i)  $\psi'(2) < \psi'(1_{v_{j'}})$ ,
- (ii)  $\omega'(1_{v_{j'}}) < \min(\omega'(3), \omega'(4))$ ,
- (iii)  $\max(\omega'(3), \omega'(4)) < \omega'(1_{v_{j'}})$ , or
- (iv)  $\omega'(3) < \omega'(1_{v_{j'}}) < \omega'(4)$ .

Let  $\delta$  be the restriction of  $\psi'$  to  $\{1_{v_{j'}}, 2, 3, 4\}$  and, similarly, let  $\delta'$  be the restriction of  $\omega'$  to the same four-element set. Regardless of which of (i), (ii), (iii), or (iv) holds, we obtain two linear orderings on  $\{1_{v_{j'}}, 2, 3, 4\}$  of which at least one is different from  $(1_{v_{j'}}, 2, 3, 4)$  and  $(2, 4, 1_{v_{j'}}, 3)$ . This contradicts the fact that, by Lemma 9, the instance  $(\{1_{v_{j'}}, 2, 3, 4\}, \mathcal{C}(v_{j'}))$  of 2- $\Pi_6$ , whose set of constraints is a subset of  $\mathcal{C}'$ , has the unique solution  $(1_{v_{j'}}, 2, 3, 4)$  and  $(2, 4, 1_{v_{j'}}, 3)$ . For the remainder of the proof, we may therefore assume that, for each  $1_{v_j} \in D$  with  $j \in \{1, 2, \dots, m\}$ , we have  $\psi'(1_{v_j}) < \psi'(2)$  and  $\omega'(4) < \omega'(1_{v_j}) < \omega'(3)$ .

Now, let  $\alpha$  be the linear ordering of  $V$  that is obtained from  $\psi'[D]$  by replacing each  $1_{v_j}$  with  $v_j$  for  $j \in \{1, 2, \dots, m\}$ . Similarly, let  $\beta$  be the linear ordering of  $V$  that is obtained from  $\omega'[D]$  by replacing each  $1_{v_j}$  with  $v_j$ . Consider an element  $C^i = (v_1^i, v_2^i, v_3^i)$  in  $\mathcal{C}$  with  $i \in \{1, 2, \dots, n\}$  and its three corresponding constraints in  $\mathcal{C}_2$ . If  $\psi'(1_{v_1^i}) < \min(\psi'(1_{v_2^i}), \psi'(1_{v_3^i}))$  or  $\omega'(1_{v_1^i}) < \min(\omega'(1_{v_2^i}), \omega'(1_{v_3^i}))$ , then it is easily checked that  $C^i$  is  $\Pi_1$ -satisfied by  $\alpha$  or  $\beta$ . Otherwise, as  $\psi'$  and  $\omega'$  is a solution to  $\mathcal{I}'$ , we have  $\psi'(1_{v_2^i}) < \psi'(1_{v_3^i}) < \psi'(1_{v_1^i})$  and  $\omega'(1_{v_3^i}) < \omega'(1_{v_2^i}) < \omega'(1_{v_1^i})$ . In particular, by Lemma 9 and the argument in the last paragraph, we have

$$\psi'(1_{v_2^i}) < \psi'(1_{v_3^i}) < \psi'(1_{v_1^i}) < \psi'(2) < \psi'(3) < \psi'(4)$$

and

$$\omega'(2) < \omega'(4) < \omega'(1_{v_3^i}) < \omega'(1_{v_2^i}) < \omega'(1_{v_1^i}) < \omega'(3).$$

It now follows that  $(1_{v_1^i}, 1_{v_2^i}, 3)$  is neither  $\Pi_6$ -satisfied by  $\alpha'$  nor  $\Pi_6$ -satisfied by  $\beta'$ ; a contradiction. Thus, we have  $\psi'(1_{v_1^i}) < \min(\psi'(1_{v_2^i}), \psi'(1_{v_3^i}))$  or  $\omega'(1_{v_1^i}) < \min(\omega'(1_{v_2^i}), \omega'(1_{v_3^i}))$ ; thereby implying that  $C^i$  is  $\Pi_1$ -satisfied by  $\alpha$  or  $\beta$ . Thus,  $\mathcal{I}$  is 2- $\Pi_1$ -satisfiable.

The theorem now follows by combining both cases.  $\square$

**Lemma 11.** *Let  $V = \{1, 2, 3, 4, 5, 6, 7\}$ , and let  $\gamma = (1, 2, 3, 4, 5, 6, 7)$  and  $\gamma' = (2, 5, 7, 3, 1, 6, 4)$  be two linear orderings of  $V$ . Then the instance  $\mathcal{I} = (V, \mathcal{C}_{\Pi_9}(\gamma) \cup \mathcal{C}_{\Pi_9}(\gamma'))$  of  $2\text{-}\Pi_9$  has a unique solution (up to reversal). In particular, each solution of  $\mathcal{I}$  consists of one element from  $\{\gamma, \bar{\gamma}\}$  and one element from  $\{\gamma', \bar{\gamma}'\}$ .*

*Proof.* Computational proof (see Section 6).  $\square$

**Theorem 12.** *The problem  $2\text{-}\Pi_9$  is NP-complete.*

*Proof.* Let  $\mathcal{I} = (V, \mathcal{C})$  be an instance of  $1\text{-}\Pi_5$  with  $\mathcal{C} = \{C^1, C^2, \dots, C^n\}$ . Furthermore, for each  $i \in \{1, 2, \dots, n\}$ , let  $C^i = (v_1^i, v_2^i, v_3^i)$ , with  $\{v_1^i, v_2^i, v_3^i\} \subseteq V$ . We next reduce  $\mathcal{I}$  to an instance of  $2\text{-}\Pi_9$ . For each  $C^i \in \mathcal{C}$ , let  $V^i$  be the set  $\{1^i, 2^i, 3^i, 5^i, v_1^i, v_2^i, v_3^i\}$  of variables, let  $\gamma^i = (1^i, 2^i, 3^i, v_1^i, 5^i, v_2^i, v_3^i)$  and  $\delta^i = (2^i, 5^i, v_3^i, 3^i, 1^i, v_2^i, v_1^i)$  be two linear orderings of  $V^i$ . By replacing the variables 4, 6, and 7 with  $v_1^i, v_2^i$ , and  $v_3^i$ , respectively, in the statement of Lemma 11, note that a solution to the instance  $(V^i, \mathcal{C}_{\Pi_9}(\gamma^i) \cup \mathcal{C}_{\Pi_9}(\delta^i))$  of  $2\text{-}\Pi_9$  consists of one element from  $\{\gamma^i, \bar{\gamma}^i\}$  and one element from  $\{\delta^i, \bar{\delta}^i\}$ . Indeed, the high-level idea is quite straightforward: each  $1\text{-}\Pi_5$  constraint  $(v_1^i, v_2^i, v_3^i)$  is encoded by an independent copy of the uniqueness gadget from Lemma 11, in which the elements  $v_1^i, v_2^i, v_3^i$  replace the symbols 4, 6, 7 respectively. In a solution to  $2\text{-}\Pi_9$ , and for each  $1\text{-}\Pi_5$  constraint, one ordering will carry an image of  $\gamma$  (possibly reversed) corresponding to that constraint, and the other an image of  $\gamma'$  (possibly reversed) corresponding to that constraint. But in both  $\gamma$  and  $\gamma'$  6 is between 4 and 7, so in fact both orderings  $\Pi_5$ -satisfy  $(v_1^i, v_2^i, v_3^i)$ .

In more detail, let  $\mathcal{I}' = (V', \mathcal{C}')$  be an instance of  $2\text{-}\Pi_9$  with

$$V' = \bigcup_{i \in \{1, 2, \dots, n\}} V^i$$

and

$$\mathcal{C}' = \bigcup_{i \in \{1, 2, \dots, n\}} (\mathcal{C}_{\Pi_9}(\gamma^i) \cup \mathcal{C}_{\Pi_9}(\delta^i)).$$

Since the number of elements in  $\mathcal{C}'$  and  $V'$  is polynomial in  $n$ , the reduction can be carried out in polynomial time. To complete the proof, we show that  $\mathcal{I}$  is  $1\text{-}\Pi_5$ -satisfiable if and only if  $\mathcal{I}'$  is  $2\text{-}\Pi_9$ -satisfiable.

First, suppose that  $\mathcal{I}$  is  $1\text{-}\Pi_5$ -satisfiable. Then there exists a linear ordering  $\alpha$  on  $V$  such that each constraint in  $\mathcal{C}$  is  $\Pi_5$ -satisfied by  $\alpha$ . Let  $\alpha'$  be a linear ordering of  $V'$  obtained from  $\alpha$  by preserving  $\alpha$  and adding all elements in  $V' - V$  such that, for each  $C^i \in \mathcal{C}$ , the constraints in  $\mathcal{C}_{\Pi_9}(\gamma^i)$  are  $\Pi_9$ -satisfied. More precisely, if  $\alpha(v_1^i) < \alpha(v_2^i) < \alpha(v_3^i)$ , then

$$\alpha'(1^i) < \alpha'(2^i) < \alpha'(3^i) < \alpha'(v_1^i) < \alpha'(5^i) < \alpha'(v_2^i) < \alpha'(v_3^i)$$

and, if  $\alpha(v_3^i) < \alpha(v_2^i) < \alpha(v_1^i)$ , then

$$\alpha'(v_3^i) < \alpha'(v_2^i) < \alpha'(5^i) < \alpha'(v_1^i) < \alpha'(3^i) < \alpha'(2^i) < \alpha'(1^i).$$

Similarly, let  $\beta'$  be a linear ordering of  $V'$  obtained from  $\alpha$  by preserving  $\alpha$  and adding all elements in  $V' - V$  such that, for each  $C^i \in \mathcal{C}$ , the constraints in  $\mathcal{C}_{\Pi_9}(\delta^i)$  are  $\Pi_9$ -satisfied. More precisely, if  $\alpha(v_1^i) < \alpha(v_2^i) < \alpha(v_3^i)$ , then

$$\beta'(v_1^i) < \beta'(v_2^i) < \beta'(1^i) < \beta'(3^i) < \beta'(v_3^i) < \beta'(5^i) < \beta'(2^i)$$

and, if  $\alpha(v_3^i) < \alpha(v_2^i) < \alpha(v_1^i)$ , then

$$\beta'(2^i) < \beta'(5^i) < \beta'(v_3^i) < \beta'(3^i) < \beta'(1^i) < \beta'(v_2^i) < \beta'(v_1^i).$$

By repeated applications of Lemma 11, it now follows that each constraint in  $\mathcal{C}'$  is  $\Pi_9$ -satisfied by  $\alpha'$  or  $\beta'$  and, thus,  $\mathcal{I}'$  is 2- $\Pi_9$ -satisfiable.

Second, suppose that  $\mathcal{I}'$  is 2- $\Pi_9$ -satisfiable. Then there exist two linear orderings  $\alpha'$  and  $\beta'$  on  $V'$  such that each constraint in  $\mathcal{C}'$  is  $\Pi_9$ -satisfied by  $\alpha'$  or  $\beta'$ . It follows from Lemma 11 that, each solution to the instance  $(V^i, \mathcal{C}_{\Pi_9}(\gamma^i) \cup \mathcal{C}_{\Pi_9}(\delta^i))$  consists of one element from  $\{\gamma^i, \bar{\gamma}^i\}$  and one element from  $\{\delta^i, \bar{\delta}^i\}$ . We will show that the result holds if  $\alpha'$  preserves  $\gamma^i$  and  $\beta'$  preserves  $\delta^i$ , and note that symmetrical arguments hold as long as one linear order preserves one element from  $\{\gamma^i, \bar{\gamma}^i\}$  and the other linear order preserves one element from  $\{\delta^i, \bar{\delta}^i\}$ . Let  $\alpha$  be the linear ordering  $\alpha'[V]$  on  $V$ . Since, for each  $C^i \in \mathcal{C}$ , we have  $\alpha'(v_1^i) < \alpha'(v_2^i) < \alpha'(v_3^i)$  or  $\alpha'(v_3^i) < \alpha'(v_2^i) < \alpha'(v_1^i)$ , it is easily checked that  $C^i$  is  $\Pi_5$  satisfied by  $\alpha$  and thus,  $\mathcal{I}$  is 1- $\Pi_5$ -satisfiable.

The theorem now follows by combining both cases.  $\square$

## 4 An alternative proof of Theorem 5

In this section, we establish an alternative proof that 2- $\Pi_1$  is NP-complete. While the proof is more involved than the one presented in Section 3, it highlights a link of 2- $\Pi_1$  to the graph-theoretic problem DICHROMATIC NUMBER. As an interesting by-product, we also strengthen the hardness result for DICHROMATIC NUMBER to a subclass of digraphs whose vertices have a bounded out-degree.

We start by introducing the decision problem DICHROMATIC NUMBER that generalizes the concept of the chromatic number to directed graphs [21]. More formally, DICHROMATIC NUMBER can be stated as follows.

DICHROMATIC NUMBER

**Instance.** A digraph  $D$  with vertex set  $V(D)$  and a positive integer  $k$ .

**Question.** Is  $D$   $k$ -dicolorable, i.e. does there exist a vertex coloring of  $V(D)$  with at most  $k$  colors such that the subgraph induced by each color does not contain any directed cycle?

It is shown in [3] that DICHROMATIC NUMBER is an NP-complete problem. In particular, the authors showed that the special case of DICHROMATIC NUMBER in which  $k$  is fixed to 2 is NP-complete. We refer to this variation of the problem as 2-DICHROMATIC NUMBER and it is exactly this problem that we use as a starting point to show that 2- $\Pi_1$  is NP-complete.

Let  $D$  be the input to an instance  $\mathcal{I}$  of 2-DICHROMATIC NUMBER. Following the notation introduced above, we say that  $\mathcal{I}$  is 2-dicolorable if  $\mathcal{I}$  is a ‘yes’-instance and refer to any valid coloring of  $D$  with exactly two colors as a 2-dicoloring. We will see later that a digraph that is 2-dicolorable corresponds to a carefully constructed ‘yes’-instance of 2- $\Pi_1$ .

Let  $\mathcal{R}$  be a set of rooted triplets, and let  $X$  be the label set of  $\mathcal{R}$ . Let  $D(\mathcal{R})$  be the digraph whose vertex set is  $X$  and for which  $(c, a)$  and  $(c, b)$  are arcs in  $D(\mathcal{R})$  precisely if  $ab|c$  is an element in  $\mathcal{R}$ . We call  $D(\mathcal{R})$  the *triplet digraph* of  $\mathcal{R}$ . Furthermore, we say that  $\mathcal{R}$  is *caterpillar-compatible* if there exists a caterpillar that displays each triplet in  $\mathcal{R}$ .

**Lemma 13.** *A set  $\mathcal{R}$  of triplets is caterpillar-compatible if and only if the triplet digraph  $D(\mathcal{R})$  is acyclic.*

*Proof.* Throughout the proof, let  $X$  be the label set of  $\mathcal{R}$  with  $|X| = n$ .

First, suppose that there exists a caterpillar  $T$  such that each triplet  $r \in \mathcal{R}$  is displayed by  $T$ . Towards a contradiction, assume that there is a cycle  $C = x_0, x_1, \dots, x_k, x_0$  in  $D(\mathcal{R})$ . Without loss of generality, we may assume that  $C$  is a simple cycle. First note that, by construction of  $D(\mathcal{R})$ , each arc of  $C$  corresponds to some triplet in  $\mathcal{R}$  and no two arcs of  $C$  correspond to

the same triplet. Now, for each arc  $(x_i, x_j)$  in  $C$  with  $i \in \{0, 1, \dots, k\}$  and  $j = (i + 1 \bmod k)$ , let  $r_{i,j}$  be a triplet in  $\mathcal{R}$  that corresponds to  $(x_i, x_j)$  in  $D(\mathcal{R})$ . We next show that the triplets  $r_{0,1}, r_{1,2}, \dots, r_{k,0}$  cannot all be displayed by  $T$ . To see that this is indeed not possible, observe that, for  $T$  to display each triplet  $r_{i,j}$  with  $i \in \{0, 1, \dots, k\}$  and  $j = (i + 1 \bmod k)$ , the path from  $x_i$  to the root of  $T$  is shorter than the path from  $x_j$  to the root of  $T$ . This gives a contradiction and, hence,  $D(\mathcal{R})$  is acyclic.

Second, suppose that  $D(\mathcal{R})$  is acyclic. Since  $D(\mathcal{R})$  is a directed acyclic graph, it has a topological ordering, say  $(x_n, x_{n-1}, \dots, x_2, x_1)$ , where  $x_i$  has no incoming arc in the graph obtained from  $D(\mathcal{R})$ , by deleting the vertices in  $\{x_n, x_{n-1}, \dots, x_{i+1}\}$ . Now, let  $T$  be a caterpillar on  $X$  such that  $x_1$  and  $x_2$  is the unique cherry of  $T$  and, for each  $i \in \{3, 4, \dots, n\}$ , the parent of  $x_{i-1}$  is a child of the parent of  $x_i$ . Let  $x_i x_j | x_k$  be a triplet of  $\mathcal{R}$ . By construction of  $T$  and because  $(x_k, x_i)$  and  $(x_k, x_j)$  are arcs in  $D(\mathcal{R})$ , it follows that  $\text{lca}_T(x_i, x_k) = \text{lca}_T(x_j, x_k) <_T \text{lca}_T(x_i, x_j)$ . Hence  $T$  displays each triplet in  $\mathcal{R}$ . Combining both cases establishes the lemma.  $\square$

We next introduce a new variant of the DICHROMATIC NUMBER problem. Let  $D$  be a digraph and  $v$  be a vertex of  $D$ . We use  $d^+(v)$  to denote the out-degree of  $v$  in  $D$  and  $\Delta^+(D)$  to denote the maximum out-degree of all vertices in  $D$ . Now consider the following decision problem.

### 2-DICHROMATIC NUMBER OUT-DEGREE 3

**Instance.** A digraph  $D$  with vertex set  $V(D)$  and  $\Delta^+(D) \leq 3$ .

**Question.** Is  $D$  2-dicolorable, i.e. does there exist a vertex coloring of  $V(D)$  with at most 2 colors such that the subgraph induced by each color does not contain any directed cycle?

The next theorem establishes NP-completeness of 2-DICHROMATIC NUMBER OUT-DEGREE 3.

**Theorem 14.** *The problem 2-DICHROMATIC NUMBER OUT-DEGREE 3 is NP-complete.*

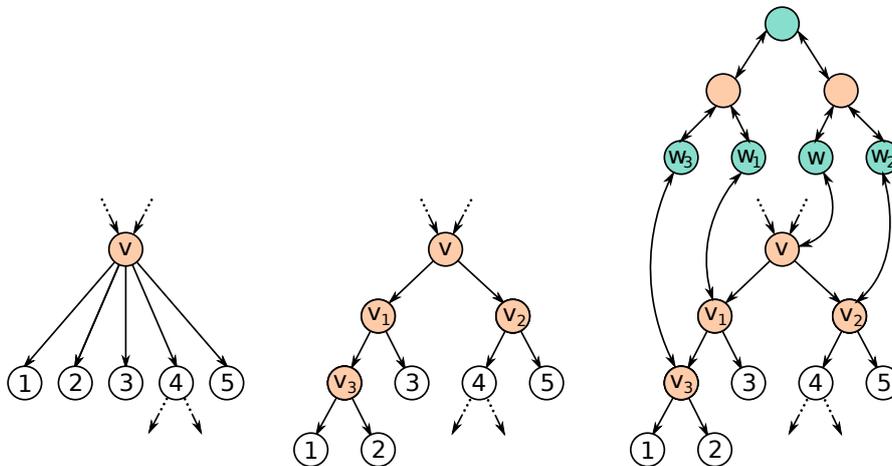


Figure 2: Left: A digraph  $D$ . Middle and right: A gadget for a vertex  $v$  of  $D$  whose out-degree is at least 3. For details, see the proof of Theorem 14.

*Proof.* The problem 2-DICHROMATIC NUMBER OUT-DEGREE 3 is clearly in NP since, given a coloring of a digraph it can be verified in polynomial time whether or not each color induces a subgraph that does not contain any directed cycle. We complete the proof, by using a reduction

from 2-DICHROMATIC NUMBER to 2-DICHROMATIC NUMBER OUT-DEGREE 3. Let a digraph  $D$  be the input to an instance of 2-DICHROMATIC NUMBER. We construct a digraph  $D'$  from  $D$  as follows. Start with  $D' = D$  and repeat the steps described in the next paragraph for each vertex  $v$  in  $D$  with  $d^+(v) \geq 3$ .

Let  $v$  be a vertex of  $D'$  that corresponds to a vertex in  $D$  with  $d = d^+(v) \geq 3$ . Replace  $v$  with a rooted balanced binary tree  $T_v$  on  $d$  leaves with  $v$  as its root (for an example, see Figure 4 (left and middle)). In other words, when we replace  $v$  by  $T_v$ , we still consider  $v$  to correspond to the vertex  $v$  in  $D$ . Furthermore, the leaves of  $T_v$ , which may or may not be leaves of the resulting graph, are labeled by the  $d$  children of  $v$  in  $D$ . Note that there are, excluding the root which is labeled with  $v$ , exactly  $d - 2$  internal vertices which are labeled with  $v_1, v_2, \dots, v_{d-2}$ . It is now easy to verify that  $d^+(v) = 2 \leq 3$  in  $D'$ . Next, we add a gadget to  $D'$  whose purpose is, as we will see shortly, to force  $v$  in  $D'$  to have the same color as all its descendants that are contained in  $\{v_1, v_2, \dots, v_{d-2}\}$  in any 2-dicoloring of  $D'$ . More precisely, we construct a rooted digraph  $D_v$  on  $d - 1$  leaves bijectively labeled with  $\{w, w_1, w_2, \dots, w_{d-2}\}$  that satisfies the following properties: **(i)**  $(u, u')$  is an arc in  $D_v$  if and only if  $(u', u)$  is an arc in  $D_v$ ; **(ii)** each leaf of  $D_v$  has the same distance from the root; and **(iii)**  $\Delta^+(D_v) \leq 3$ . Now connect  $D_v$  with  $D'$  by adding the following  $2(d - 1)$  arcs in

$$\{(w, v), (v, w), (w_1, v_1), (v_1, w_1), (w_2, v_2), (v_2, w_2), \dots, (w_{d-2}, v_{d-2}), (v_{d-2}, w_{d-2})\}.$$

It is now easily checked that the digraph  $D'$  that ultimately results from the construction described in the last paragraph has  $\Delta^+(D') \leq 3$ . Furthermore, the construction can be carried out in polynomial time since the size of  $D'$  is polynomial in the size of  $D$ . Lastly, note that each arc  $(v, u)$  in  $D$  either corresponds to a single arc  $(v, u)$  in  $D'$  if  $d^+(v) < 3$  in  $D$  or, corresponds to a path of vertices such that each vertex on that path except for the last is an element in  $\{v, v_1, v_2, \dots, v_{d-2}\}$  if  $d^+(v) \geq 3$  in  $D$ .

The remainder of the proof essentially consists of establishing the following claim.

**Claim.**  $D$  is 2-dicolorable if and only if  $D'$  is 2-dicolorable.

Throughout the proof of the claim, let  $V(D)$  (resp.  $V(D')$ ) be the vertex set of  $D$  (resp.  $D'$ ).

First, suppose that  $D$  is 2-dicolorable. Let  $c$  be a 2-dicoloring of  $D$ . We obtain a coloring of  $D'$  in the following way. Assign each vertex  $v$  of  $D'$  that corresponds to a vertex of  $D$  to  $c(v)$ . Now, for a vertex  $v$  of  $D$  with  $d^+(v) \geq 3$ , consider the rooted digraph  $D_v$  as defined in the construction of  $D'$ . Since  $D_v$  satisfies property **(i)**, the only way to color  $D_v$  with two colors  $\zeta_1$  and  $\zeta_2$  is to assign the root to  $\zeta_1$  and each other vertex to  $\zeta_1$  if it has an even distance to the root of  $D_v$  and to  $\zeta_2$  otherwise. Without loss of generality, we may assume that each leaf of  $D_v$  is assigned to  $\zeta_1$ . Now consider the rooted balanced binary tree  $T_v$  as defined in the construction of  $D'$  and, in particular the set of arcs that join a leaf of  $D_v$  to a vertex of  $T_v$  (and vice versa). Since  $w$  and  $v$  form a 2-cycle in  $D'$  and, for each  $i \in \{1, 2, \dots, d - 2\}$ ,  $w_i$  and  $v_i$  form a 2-cycle in  $D'$ , it follows that each vertex in  $\{v, v_1, v_2, \dots, v_{d-2}\}$  is assigned to  $\zeta_2$ . Up to interchanging  $\zeta_1$  and  $\zeta_2$ , it now follows that this is the unique 2-dicoloring of the subgraph of  $D'$  induced by the vertices of  $D_v$  and  $T_v$ . By repeating the described procedure for each vertex in  $D$  whose out-degree is at least 3, we obtain a coloring  $c'$  of  $D'$ . We next show that  $c'$  is a 2-dicoloring of  $D'$ . To the contrary, assume that there exists a directed cycle  $C'$  in  $D'$  whose vertices are all assigned to the same color. Clearly, if  $C'$  contains only vertices that are also vertices in  $D$  we are done. Therefore,  $C'$  contains a vertex that is not a vertex in  $D$ . Since all vertices of  $C'$  are assigned to the same color, no vertex of  $C'$  is a vertex of any subgraph  $D_v$ . Hence,  $C'$  only contains vertices of  $D$  and, possibly, for a vertex  $v$  in  $D$  with  $d = d^+(v) \geq 3$ , the vertex  $v$  itself and some vertices in  $\{v_1, v_2, \dots, v_{d-2}\}$ . In summary, this implies that  $C'$  contains a vertex  $v$  with

$d^+(v) \geq 3$  in  $D$ . Now, if we contract all vertices in  $\{v_1, v_2, \dots, v_{d-2}\}$  and repeat the procedure for each vertex of  $C'$  that corresponds to a vertex of  $D$  with out-degree at least 3, the procedure of vertex contractions in  $C'$  results in a directed cycle  $C$  that only contains vertices of  $D$  and, by construction, implies that  $c$  assigns all vertices in  $C$  to the same color; a contradiction.

Second, suppose that  $D'$  is 2-dicolorable. Since  $V(D) \subseteq V(D')$ , we can obtain a coloring  $c$  of  $D$  from a 2-dicoloring  $c'$  of  $D'$  by assigning each vertex in  $D$  to the same color as in  $D'$ . Now, towards a contradiction, assume that  $c$  is not a 2-dicoloring of  $D$ . Then, there exists a color  $\zeta$  that induces a subgraph of  $D$  that contains a directed cycle  $C$ . By construction of  $D'$ , it follows that  $D'$  contains a directed cycle  $C'$  such that the vertex set of  $C$  is a subset of the vertex set of  $C'$ . Moreover, each arc  $(v, u)$  of  $C$  either corresponds to an arc  $(v, u)$  in  $C'$  if  $d^+(v) < 3$  in  $D$  or corresponds to a directed path from  $v$  to  $u$  in  $C'$  if  $d^+(v) \geq 3$  in  $D$ . Furthermore, since all vertices on the directed path from  $u$  to  $v$  in  $D'$ , which consists of at least one arc, have the same color as  $v$  (as justified in the last paragraph), it now follows that, under  $c'$ ,  $\zeta$  induces a subgraph of  $D'$  that contains a directed cycle; a contradiction.

The theorem now follows by combining both cases.  $\square$

In the remainder of this section, we show that  $2\text{-}\Pi_1$  is NP-complete by using a reduction from  $2\text{-DICHROMATIC NUMBER OUT-DEGREE 3}$ . Formally, we reduce an instance of  $2\text{-DICHROMATIC NUMBER OUT-DEGREE 3}$  to an instance of  $2\text{-CATERPILLAR COMPATIBILITY}$ . Since  $2\text{-CATERPILLAR COMPATIBILITY}$  and  $2\text{-}\Pi_1$  are essentially equivalent (see Observation 1), the result then follows immediately.

**Theorem 15.** *The problem  $2\text{-CATERPILLAR COMPATIBILITY}$  is NP-complete.*

*Proof.* The problem  $2\text{-CATERPILLAR COMPATIBILITY}$  is clearly in NP because, given two caterpillars  $T$  and  $T'$ , it can be checked in polynomial time if each element of a set of rooted triplets is displayed by  $T$  or  $T'$ . We complete the proof, by using a reduction from  $2\text{-DICHROMATIC NUMBER OUT-DEGREE 3}$  to  $2\text{-CATERPILLAR COMPATIBILITY}$ . Let a digraph  $D$  be the input to an instance of  $2\text{-DICHROMATIC NUMBER OUT-DEGREE 3}$ . Let  $v$  be a vertex of  $D$ . Depending on  $d^+(v)$ , which is at most 3, let  $\mathcal{R}_v$  be one of the followings.

- (i) If  $v$  has exactly one child, say  $a$ , set  $\mathcal{R}_v = \{ad_v|v\}$ , where  $d_v$  does not label any vertex of  $D$ .
- (ii) If  $v$  has exactly two children, say  $a$  and  $b$ , set  $\mathcal{R}_v = \{ab|v\}$ .
- (iii) If  $v$  has exactly three children, say  $a$ ,  $b$ , and  $c$ , set  $\mathcal{R}_v = \{ab|v, ac|v, bc|v\}$ .

Now, let

$$\mathcal{R} = \bigcup_{v \in V(D)} \mathcal{R}_v,$$

where  $V(D)$  denotes the vertex set of  $D$ , be the input to an instance of  $2\text{-CATERPILLAR COMPATIBILITY}$ . Clearly, the reduction can be carried out in polynomial time and  $\mathcal{R}$  has size polynomial in the size of  $V(D)$ . Moreover, we make the following crucial observation that we will freely use throughout the rest of the proof. Note that  $D(\mathcal{R})$ , the triplet digraph of  $\mathcal{R}$ , is essentially the same graph as  $D$ . The only difference is that  $D(\mathcal{R})$  has an extra leaf  $d_v$  for each vertex  $v$  in  $V(D)$  with  $d^+(v) = 1$ . However, these extra leaves have no impact upon the acyclicity of  $D(\mathcal{R})$ .

We next establish the following claim.

**Claim.**  $D$  is 2-dicolorable if and only if there exists two caterpillars  $T$  and  $T'$  such that each triplet in  $\mathcal{R}$  is displayed by  $T$  or  $T'$ .

First, suppose that  $D$  is 2-dicolorable. Let  $\zeta_1$  or  $\zeta_2$  be the two colors that are used in a 2-dicoloring  $c$  of  $D$ . Furthermore, let  $\mathcal{R}_1$  and  $\mathcal{R}_2$  be the two sets of a bipartition of  $\mathcal{R}$  such that  $\mathcal{R}_1$  contains each triplet of  $\mathcal{R}$  whose witness corresponds to a vertex in  $D$  that is assigned to color  $\zeta_1$  and, similarly,  $\mathcal{R}_2$  contains each triplet of  $\mathcal{R}$  whose witness corresponds to a vertex in  $D$  that is assigned to color  $\zeta_2$ . Now assume that  $\mathcal{R}_1$  is not caterpillar-compatible. It then follows by Lemma 13 that  $D(\mathcal{R}_1)$  contains a directed cycle  $C$ . Moreover, the tail of each edge in  $C$  corresponds to a witness of some triplet in  $\mathcal{R}_1$ . Hence, all vertices of  $C$  are assigned to  $\zeta_1$ ; thereby contradicting that  $c$  is a 2-dicoloring of  $D$ .

Second, suppose that there exist two caterpillars  $T$  and  $T'$  such that each triplet in  $\mathcal{R}$  is displayed by either  $T$  or  $T'$ . Let  $\mathcal{R}_1$  be the subset of triplets of  $\mathcal{R}$  that are displayed by  $T$ , and let  $\mathcal{R}_2$  be the subset of triplets of  $\mathcal{R}$  that are not displayed by  $T$ . Observe that each triplet in  $\mathcal{R}$  is encoded by a unique vertex of  $D$  and that a vertex  $v$  of  $D$  is encoded by either a single triplet in  $\mathcal{R}$  (if  $d^+(v) < 3$ ) or three triplets in  $\mathcal{R}$  (if  $d^+(v) = 3$ ). Now, let  $v$  be a vertex of  $D$ . If  $v$  is encoded by a single triplet in  $\mathcal{R}$  that is displayed by  $T$  (resp. not displayed by  $T$ ), then assign  $v$  to color  $\zeta_1$  (resp.  $\zeta_2$ ). Furthermore, if  $v$  is encoded by three triplets in  $\mathcal{R}$ , then, by the pigeonhole principle, at least two of those triplets are displayed by  $T$  (resp. are not displayed by  $T$ ) in which case we assign  $v$  to  $\zeta_1$  (resp.  $\zeta_2$ ). It remains to show that the resulting coloring  $c$  of  $D$  is a 2-dicoloring. Assume the contrary. Then there exist a directed cycle  $C = v_0, v_1, \dots, v_k, v_0$  in  $D$  whose vertices are all assigned to the same color under  $c$ . Without loss of generality, we may assume that this color is  $\zeta_1$ . Let  $i \in \{0, 1, \dots, k\}$ , and let  $j = i + 1 \pmod k$ . We next apply an iterative procedure on each vertex in  $C$  and construct a set  $\mathcal{R}'_1$  of triplets. Initially, set  $\mathcal{R}'_1 = \emptyset$ . Now, repeat the following for each  $v_i$  in  $C$ . If  $d^+(v_i) = 1$  in  $D$ , reset  $\mathcal{R}'_1$  to  $\mathcal{R}'_1 \cup \{v_j d_{v_i} | v_i\}$ . If  $d^+(v_i) = 2$  in  $D$ , let  $a$  be the child of  $v_i$  that is not contained in  $C$  and reset  $\mathcal{R}'_1$  to  $\mathcal{R}'_1 \cup \{v_j a | v_i\}$ . Lastly, if  $d^+(v_i) = 3$  in  $D$ , let  $a$  and  $b$  be the two children of  $v_i$  that are not contained in  $C$ , let  $r_1$  and  $r_2$  be the two triplets in  $\{ab | v_i, av_j | v_i, bv_j | v_i\}$  that are displayed by  $T$  and reset  $\mathcal{R}'_1$  to  $\mathcal{R}'_1 \cup \{r_1, r_2\}$ . In all three cases, it follows by construction of  $c$  that, as  $v_i$  is assigned to  $\zeta_1$ , each triplet in  $\mathcal{R}'_1$  is displayed by  $T$ . It is now easily checked that  $D(\mathcal{R}'_1)$  contains  $C$ . Furthermore, by Lemma 13, we deduce that  $\mathcal{R}'_1$  is incompatible and, hence,  $\mathcal{R}_1$  is also incompatible; thereby contradicting that each triplet in  $\mathcal{R}$  is displayed by  $T$  or  $T'$ .

Combining both cases establishes the claim and, therefore the theorem.  $\square$

**Corollary 16.** *The problem 2- $\Pi_1$  is NP-complete.*

## 5 Three linear orderings and an extension to phylogenetic trees

In the last section, we have shown that 2- $\Pi_1$  is NP-complete. In this section we first show that the problem remains computationally hard if we allow for three instead of only two linear orderings. More formally, we show that 3-CATERPILLAR COMPATIBILITY is NP-complete and, hence, by Observation 1, 3- $\Pi_1$  is NP-complete as well.

In the second part of the section, we extend the hardness result of 3-CATERPILLAR COMPATIBILITY and show that the following decision problem is NP-complete.

### 3-TREE COMPATIBILITY

**Instance.** A set  $\mathcal{R}$  of rooted triplets.

**Question.** Do there exist at most three rooted phylogenetic trees such that each element in  $\mathcal{R}$  is displayed by at least one such tree?

In the flavor of Section 3, we start with a lemma whose proof is computational. It shows that the three caterpillars shown in Figure 3 are uniquely defined by their induced triplets, not just in the space of caterpillars but also in the wider space of phylogenetic trees.

**Lemma 17.** *Let  $C_1$ ,  $C_2$ , and  $C_3$  be the three caterpillars that are shown in Figure 3, and let  $\mathcal{C}$  be the set of all triplets that are displayed by  $C_1$ ,  $C_2$ , or  $C_3$ . Then, for each set of three trees, say  $T_1$ ,  $T_2$ , and  $T_3$ , that is a solution to 3-TREE COMPATIBILITY with input  $\mathcal{C}$ , we have  $\{T_1|_{\{0,1,\dots,5\}}, T_2|_{\{0,1,\dots,5\}}, T_3|_{\{0,1,\dots,5\}}\} = \{C_1, C_2, C_3\}$ .*

*Proof.* Computational proof (see Section 6). □

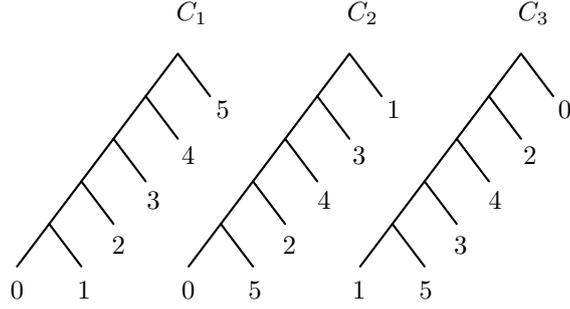


Figure 3: The three trees that are used in the proofs of Lemma 17, and Theorems 18 and 19.

To prove the first main result of this section (Theorem 18), we need a new definition. Let  $e$  and  $f$  be two leaves of a caterpillar  $C$ . If there exists a directed path from the parent of  $f$  to  $e$  in  $C$ , we say that  $e$  is *below*  $f$  or, equivalently,  $f$  is *above*  $e$  in  $D$  and write  $e \prec f$ .

**Theorem 18.** *The problems 3-CATERPILLAR COMPATIBILITY and, hence, 3- $\Pi_1$  are NP-complete.*

*Proof.* Trivially, 3-CATERPILLAR COMPATIBILITY is in NP. Let  $\mathcal{I}$  be an instance of 2-CATERPILLAR COMPATIBILITY, i.e.  $\mathcal{I}$  is a set of triplets. To show that the theorem holds, we reduce  $\mathcal{I}$  to an instance of 3-CATERPILLAR COMPATIBILITY. Let  $\mathcal{C}$  be the set of triplets as defined in the statement of Lemma 17. Furthermore, we define the following five sets of triplets, where, for each triplet  $ab|c \in \mathcal{I}$ ,  $\underline{ab}$  denotes a new taxon that is not a leaf label of a triplet in  $\mathcal{I}$ .

$$\begin{aligned} \mathcal{R}_1 &= \bigcup_{ab|c \in \mathcal{I}} \{3a|5, 3a|1, 4a|0, 3b|5, 3b|1, 4b|0, 3c|5, 3c|1, 4c|0, 3\underline{ab}|5, 3\underline{ab}|1, 4\underline{ab}|0\}, \\ \mathcal{R}_2 &= \bigcup_{ab|c \in \mathcal{I}} \{4\underline{ab}|2\}, \\ \mathcal{R}_3 &= \bigcup_{ab|c \in \mathcal{I}} \{0\underline{ab}|c\}, \\ \mathcal{R}_4 &= \bigcup_{ab|c \in \mathcal{I}} \{25|a, 12|a, 0a|2, 25|b, 12|b, 0b|2\}, \text{ and} \\ \mathcal{R}_5 &= \bigcup_{ab|c \in \mathcal{I}} \{5a|\underline{ab}, 5b|\underline{ab}, 1a|\underline{ab}, 1b|\underline{ab}\}. \end{aligned}$$

Intuitively, the idea of the new taxon  $\underline{ab}$  is to represent the parental vertex of  $a$  and  $b$  in the rooted triplet  $ab|c$ . Furthermore, the role of most triplets in  $\mathcal{R}_i$  with  $i \in \{1, 2, \dots, 5\}$  is to force

the parent of  $\underline{ab}$  to be an ancestor of the vertex  $\text{lca}(a, b)$  in two of the three resulting caterpillars that have the trees  $C_1$  and  $C_2$  as shown in Figure 3 as restricted subtrees while in the third caterpillar, that has  $C_3$  as a restricted subtree, the parent of  $\underline{ab}$  is forced to be a descendant of the vertex  $\text{lca}(a, b)$ . Then, finally, the triplet  $0\underline{ab}|c$  can only be displayed by one of the first two caterpillars and this is exactly the case if one of them displays the triplet  $ab|c$ .

Now, let  $\mathcal{R} = \mathcal{R}_1 \cup \mathcal{R}_2 \cup \dots \cup \mathcal{R}_5$ . Clearly, the number of elements in  $\mathcal{C}$  and  $\mathcal{R}$  is polynomial in 6 and  $|\mathcal{I}|$ , respectively. To complete the proof, we show that  $\mathcal{I}$  is a ‘yes’-instance of 2-CATERPILLAR COMPATIBILITY if and only if  $\mathcal{C} \cup \mathcal{R}$  is a ‘yes’-instance of 3-CATERPILLAR COMPATIBILITY.

First, suppose that  $\mathcal{I}$  is a ‘yes’-instance of 2-CATERPILLAR COMPATIBILITY. Then, there exist two caterpillars  $S_1$  and  $S_2$  such that each triplet in  $\mathcal{I}$  is displayed by  $S_1$  or  $S_2$ . Illustrated in Figure 4, we obtain three new caterpillars  $T_1$ ,  $T_2$ , and  $T_3$  such that each triplet in  $\mathcal{C} \cup \mathcal{R}$  is displayed by one of the three trees as follows. For  $i \in \{1, 2, 3\}$ , start by setting  $T_i = C_i$ , where  $C_i$  is the caterpillar shown in Figure 3 and insert  $S_1$  into  $T_1$  below 2 and above 1, and  $S_2$  into  $T_2$  below 2 and above 5. It is easily checked that the resulting trees display all triplets in  $\mathcal{C}$ . Furthermore, for each triplet  $ab|c \in \mathcal{I}$  that is displayed by  $S_1$ , add the corresponding  $\underline{ab}$  taxon to  $T_1$  such that  $\underline{ab}$  is above  $a$  and  $b$  and below  $c$  and add  $\underline{ab}$  to  $T_2$  such that  $\underline{ab}$  is above  $a$  and  $b$ , and below 2. Similarly, for each triplet  $ab|c \in \mathcal{I}$  that is not displayed by  $S_1$ , add the corresponding  $\underline{ab}$  taxon to  $T_2$  such that  $\underline{ab}$  is above  $a$  and  $b$  and below  $c$  and add  $\underline{ab}$  to  $T_1$  such that  $\underline{ab}$  is above  $a$  and  $b$ , and below 2. Lastly, for each triplet  $ab|c \in \mathcal{I}$ , add each taxon in  $\{a, b, \underline{ab}\}$  to  $T_3$  such that  $4 \prec \underline{ab} \prec 2 \prec a$ ,  $2 \prec a \prec 0$ , and  $2 \prec b \prec 0$  holds. We next show that each triplet in  $\mathcal{R}$  is displayed by at least one tree of  $T_1$ ,  $T_2$ , and  $T_3$ . For each  $x \in \{a, b, c, \underline{ab}\}$ , the triplet  $3x|5$  is displayed by  $T_1$ , the triplet  $3x|1$  is displayed by  $T_2$ , and the triplet  $4x|0$  is displayed by  $T_3$ . Hence, each triplet in  $\mathcal{R}_1$  is displayed by  $T_1$ ,  $T_2$ , or  $T_3$ . Furthermore, each triplet in  $\mathcal{R}_2$  is displayed by  $T_3$  and, depending on whether or not a triplet  $ab|c \in \mathcal{I}$  is displayed by  $S_1$ , the corresponding triplet in  $\mathcal{R}_3$  is displayed by  $T_1$  or  $T_2$ . Turning to the triplets in  $\mathcal{R}_4$ , a straightforward check shows that, for each  $ab|c \in \mathcal{I}$ , the two corresponding triplets  $0a|2$  and  $0b|2$  in  $\mathcal{R}_4$  are displayed by  $T_1$  (and  $T_2$ ), while the remaining four triplets in  $\mathcal{R}_4$  are displayed by  $T_3$ . Lastly, for each triplet  $ab|c \in \mathcal{I}$ , the first two corresponding triplets in  $\mathcal{R}_5$  are displayed by  $T_2$  while the last two corresponding triplets in  $\mathcal{R}_5$  are displayed by  $T_1$ . Hence, each triplet in  $\mathcal{C} \cup \mathcal{R}$  is displayed by one of  $T_1$ ,  $T_2$ , or  $T_3$  and, therefore,  $\mathcal{C} \cup \mathcal{R}$  is a ‘yes’-instance of 3-CATERPILLAR COMPATIBILITY.

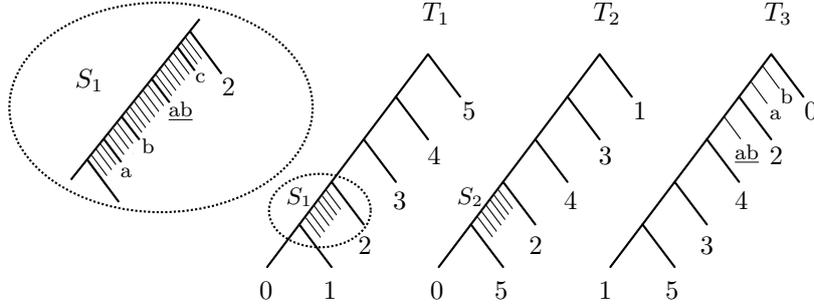


Figure 4: Right: A solution  $T_1$ ,  $T_2$ , and  $T_3$  to the transformed instance of 3-CATERPILLAR COMPATIBILITY, given a solution  $S_1$  and  $S_2$  to some 2-CATERPILLAR COMPATIBILITY instance. Left: A more detailed view of the part of the caterpillar  $T_1$  inside the dotted circle.

Second, suppose that  $\mathcal{C} \cup \mathcal{R}$  is a ‘yes’-instance of 3-CATERPILLAR COMPATIBILITY. Then, there exist three caterpillars  $T_1$ ,  $T_2$ , and  $T_3$  such that each triplet in  $\mathcal{C} \cup \mathcal{R}$  is displayed by  $T_1$ ,  $T_2$ , or  $T_3$ . By Lemma 17, we may assume without loss of generality that  $C_1 = T_1|_{\{0, 1, \dots, 5\}}$ ,  $C_2 = T_2|_{\{0, 1, \dots, 5\}}$ , and  $C_3 = T_3|_{\{0, 1, \dots, 5\}}$ , where  $C_1$ ,  $C_2$ , and  $C_3$  are the three caterpillars

that are shown in Figure 3. We make three observations that follow from the different triplet sets  $\mathcal{R}_1$ ,  $\mathcal{R}_2$ , and  $\mathcal{R}_3$ . Let  $ab|c \in \mathcal{I}$ .

- (1) For each  $x \in \{a, b, c, \underline{ab}\}$ , the triplet  $3x|5$  is only displayed by  $T_1$ , the triplet  $3x|1$  is only displayed by  $T_2$ , and the triplet  $4x|0$  is only displayed by  $T_3$ . In particular the root of  $T_1$  is the parent of 5 and, similarly, the root of  $T_2$  (resp.  $T_3$ ) is the parent of 1 (resp. 0).
- (2) The triplet  $4\underline{ab}|2$  is only displayed by  $T_3$  and, as a consequence,  $\underline{ab}$  is below 2 in  $T_3$ .
- (3) By Observation (1), the triplet  $0\underline{ab}|c$  is not displayed by  $T_3$  and, hence,  $\underline{ab}$  is below  $c$  in  $T_1$  or  $T_2$ .

Now consider the triplets in  $\mathcal{R}_4$ . We claim that, for each triplet  $ab|c \in \mathcal{I}$ , the two taxa  $a$  and  $b$  are above 2 in  $T_3$ . Assume that  $a$  is not above 2 in  $T_3$ . Then, by Observation (1),  $25|a$  is only displayed by  $T_2$  and  $12|a$  is only displayed by  $T_1$  and, therefore,  $a$  is above 2 in both of  $T_1$  and  $T_2$ . Hence,  $0a|2$  is not displayed by  $T_1$  or  $T_2$  and, by Observation (1), certainly not by  $T_3$ ; a contradiction. Similarly, assume that  $b$  is not above 2 in  $T_3$ . Then, by Observation (1),  $25|b$  is only displayed by  $T_2$  and  $12|b$  is only displayed by  $T_1$  and, therefore,  $b$  is above 2 in both  $T_1$  and  $T_2$ . Hence,  $0b|2$  is not displayed by  $T_1$  or  $T_2$  and, by Observation (1), certainly not by  $T_3$ ; a contradiction. Now, since  $a$  and  $b$  are both above 2 in  $T_3$ , it follows from Observation (2) that  $a$  and  $b$  are both above  $\underline{ab}$  in  $T_3$ . In turn, this implies that  $T_3$  does not display a triplet  $ya|\underline{ab}$  or  $yb|\underline{ab}$  with  $y \in \{1, 5\}$ . Finally consider the triplets in  $\mathcal{R}_5$ . For each triplet  $ab|c \in \mathcal{I}$ , the triplets  $5a|\underline{ab}$  and  $5b|\underline{ab}$  are only displayed by  $T_2$  due to Observation (1) and the previous claim. Similarly, the triplets  $1a|\underline{ab}$  and  $1b|\underline{ab}$  are only displayed by  $T_1$  due to Observation (1) and the previous claim. Hence,  $\underline{ab}$  is above  $a$  and  $b$  in both  $T_1$  and  $T_2$ . Now, recall that  $\underline{ab}$  is below  $c$  in at least one of  $T_1$  or  $T_2$  by Observation (3). It follows that each  $ab|c \in \mathcal{I}$  is displayed by  $T_1$  or  $T_2$  and, therefore,  $\mathcal{I}$  is a ‘yes’-instance of 2-CATERPILLAR COMPATIBILITY. By combining both cases, it follows that 3-CATERPILLAR COMPATIBILITY is NP-complete and, hence, by Observation 1, 3- $\Pi_1$  is also NP-complete.  $\square$

For a given set of triplets, we next show that it is not only NP-complete to decide if there exist three caterpillars such that each element in  $\mathcal{R}$  is displayed by at least one of these caterpillars, but also NP-complete to decide if there exist three (arbitrary) rooted phylogenetic trees such that each element in  $\mathcal{R}$  is displayed by at least one of these trees. We start with a few new definitions.

Let  $T$  be a caterpillar. Furthermore, let  $\{c, c'\}$  be the unique cherry in  $T$  and let  $x$  be the leaf of  $T$  such that the directed path from the root of  $T$  to  $x$  contains precisely one edge. We refer to the directed path from the root of  $T$  to the parent of  $c$  as the *spine* of  $T$  and to each other edge in  $T$  as a *leg* of  $T$ . Note that the definitions of spine and leg naturally carry over to each tree obtained from  $T$  by subdividing edges of  $T$  except for the edges directed into  $c$ ,  $c'$ , and  $x$  respectively. We call such a tree a *relaxed caterpillar*. Lastly, a rooted subtree of a rooted phylogenetic tree  $T$  is *pendant* if it can be detached from  $T$  by deleting a single edge. Note that each leaf of  $T$  is a pendant subtree of  $T$ .

**Theorem 19.** *The problem 3-TREE COMPATIBILITY is NP-complete.*

*Proof.* The proof of this theorem is similar to that of Theorem 18. Trivially, 3-TREE COMPATIBILITY is in NP. Let  $\mathcal{I}$  be an instance of 2-CATERPILLAR COMPATIBILITY, i.e.  $\mathcal{I}$  is a set of triplets. To show that the theorem holds, we reduce  $\mathcal{I}$  to an instance of 3-TREE COMPATIBILITY. Let  $\mathcal{C}$  be the set of triplets as defined in the statement of Lemma 17. Furthermore, we define the following six sets of triplets, where, for each triplet  $ab|c \in \mathcal{I}$ ,  $\underline{ab}$  denotes a new taxon that is not a leaf label of a triplet in  $\mathcal{I}$ .

$$\begin{aligned}
\mathcal{R}_1 &= \bigcup_{ab|c \in \mathcal{I}} \bigcup_{x \in \{a, b, c, \underline{ab}\}} \{2x|5, 3x|5, 4x|5, 2x|1, 3x|1, 4x|1, 2x|0, 3x|0, 4x|0\}, \\
\mathcal{R}_2 &= \bigcup_{ab|c \in \mathcal{I}} \{01|a, 05|a, 01|b, 05|b, 01|c, 05|c, 01|\underline{ab}, 05|\underline{ab}\}, \\
\mathcal{R}_3 &= \bigcup_{ab|c \in \mathcal{I}} \{3\underline{ab}|2, 4\underline{ab}|2, 5\underline{ab}|2\}, \\
\mathcal{R}_4 &= \bigcup_{ab|c \in \mathcal{I}} \{0\underline{ab}|c\}, \\
\mathcal{R}_5 &= \bigcup_{ab|c \in \mathcal{I}} \{25|a, 12|a, 0a|2, 25|b, 12|b, 0b|2\}, \text{ and} \\
\mathcal{R}_6 &= \bigcup_{ab|c \in \mathcal{I}} \{5a|\underline{ab}, 5b|\underline{ab}, 1a|\underline{ab}, 1b|\underline{ab}\}.
\end{aligned}$$

Note that only  $\mathcal{R}_1$ ,  $\mathcal{R}_2$ , and  $\mathcal{R}_3$  contain triplets that are not used in the reduction presented in the proof of Theorem 18. Now, let  $\mathcal{R} = \mathcal{R}_1 \cup \mathcal{R}_2 \cup \dots \cup \mathcal{R}_6$ . Clearly, the number of elements in  $\mathcal{C}$  and  $\mathcal{R}$  is polynomial in 6 and  $|\mathcal{I}|$ , respectively. To complete the proof, we show that  $\mathcal{I}$  is a ‘yes’-instance of 2-CATERPILLAR COMPATIBILITY if and only if  $\mathcal{C} \cup \mathcal{R}$  is a ‘yes’-instance of 3-TREE COMPATIBILITY.

First, suppose that  $\mathcal{I}$  is a ‘yes’-instance of 2-CATERPILLAR COMPATIBILITY. Then, there exist two caterpillars  $S_1$  and  $S_2$  such that each triplet in  $\mathcal{I}$  is displayed by  $S_1$  or  $S_2$ . Let  $T_1$ ,  $T_2$ , and  $T_3$  be the same phylogenetic trees (caterpillars) reconstructed from  $S_1$  and  $S_2$  as in the proof of Theorem 18. Building on this proof, it remains to show that all triplets in  $\mathcal{R}_1$ ,  $\mathcal{R}_2$ , and  $\mathcal{R}_3$  are displayed by at least one of  $T_1$ ,  $T_2$ , and  $T_3$ . For each  $x \in \{a, b, c, \underline{ab}\}$ , the triplets  $2x|5$ ,  $3x|5$ , and  $4x|5$  are displayed by  $T_1$ , the triplets  $2x|1$ ,  $3x|1$ , and  $4x|1$  are displayed by  $T_2$ , and the triplets  $2x|0$ ,  $3x|0$ ,  $4x|0$  are displayed by  $T_3$ . Hence, each triplet in  $\mathcal{R}_1$  is displayed by  $T_1$ ,  $T_2$ , or  $T_3$ . Furthermore, turning to the triplets in  $\mathcal{R}_2$ , each triplet  $01|x$  is displayed by  $T_1$  and each triplet  $05|x$  is displayed by  $T_2$ . Lastly, each triplet in  $\mathcal{R}_3$  is displayed by  $T_3$ . In conclusion, each triplet in  $\mathcal{C} \cup \mathcal{R}$  is displayed by one of  $T_1$ ,  $T_2$ , or  $T_3$  and, therefore,  $\mathcal{C} \cup \mathcal{R}$  is a ‘yes’-instance of 3-TREE COMPATIBILITY.

Second, suppose that  $\mathcal{C} \cup \mathcal{R}$  is a ‘yes’-instance of 3-TREE COMPATIBILITY. Then, there exist three trees  $T_1$ ,  $T_2$ , and  $T_3$  such that each triplet in  $\mathcal{C} \cup \mathcal{R}$  is displayed by  $T_1$ ,  $T_2$ , or  $T_3$ . Again, by Lemma 17, we may assume without loss of generality that  $C_1 = T_1|\{0, 1, \dots, 5\}$ ,  $C_2 = T_2|\{0, 1, \dots, 5\}$ , and  $C_3 = T_3|\{0, 1, \dots, 5\}$ , where  $C_1$ ,  $C_2$ , and  $C_3$  are the three caterpillars that are shown in Figure 3. For the remainder of the proof, we relax the definition of ‘above’ (resp. ‘below’) as defined prior to the proof of Theorem 18. In particular, for two leaves  $x$  and  $y$  in  $L(T_i)$  with  $i \in \{1, 2, 3\}$ , we say that  $x$  is *below*  $y$  or, equivalently,  $y$  is *above*  $x$  if there is a directed path from  $\text{lca}_{T_i}(c, y)$  to  $\text{lca}_{T_i}(c, x)$  that contains at least one edge and where  $c$  is a leaf of the cherry in  $C_i$ . We next consider the triplets in  $\mathcal{R}$ . Let  $ab|c \in \mathcal{I}$ .

- (1) We claim that the root of  $T_1$  is the parent of 5. To see that this is indeed true, consider the triplets  $2x|5, 3x|5, 4x|5$  with  $x \in \{a, b, c, \underline{ab}\}$ . For  $x$  being fixed, no two of these three triplets can simultaneously be displayed by  $T_2$  or  $T_3$ . Hence, at least one such triplet is only displayed by  $T_1$ ; thereby implying the correctness of the claim. A similar argument can be used to show that the parent of 1 is the root of  $T_2$  (by exploiting the triplets  $2x|1, 3x|1$ , and  $4x|1$ ) and that the parent of 0 is the root of  $T_3$  (by exploiting the triplets  $2x|0, 3x|0$ , and  $4x|0$ ).

- (2) By Observation (1), the triplets in  $\mathcal{R}_2$  guarantee that, for each  $x \in \{a, b, c, \underline{ab}\}$ , the triplet  $01|x$  is only displayed by  $T_1$  and the triplet  $05|x$  is only displayed by  $T_2$ . In other words  $\{0, 1\}$  is a cherry of  $T_1$  and  $\{0, 5\}$  is a cherry of  $T_2$ .
- (3) At least one of the three corresponding triplets in  $\mathcal{R}_3$  is only displayed by  $T_3$  and, hence,  $\underline{ab}$  is below 2 in  $T_3$ .
- (4) By Observation (1), the triplet  $0\underline{ab}|c$  in  $\mathcal{R}_4$  is not displayed by  $T_3$  and, hence,  $\underline{ab}$  is below  $c$  in  $T_1$  or  $T_2$ .
- (5) Considering  $\mathcal{R}_5$  and using the same argument as in the proof of Theorem 18, it follows that  $a$  and  $b$  are above 2 in  $T_3$ .
- (6) By Observations (3) and (5), no triplet in  $\mathcal{R}_6$  is displayed by  $T_3$ . Moreover, due to Observation (1), we have that  $5a|\underline{ab}$  and  $5b|\underline{ab}$  are only displayed by  $T_2$  and that  $1a|\underline{ab}$  and  $1b|\underline{ab}$  are only displayed by  $T_1$ ; thereby implying that  $\underline{ab}$  is above  $a$  and  $b$  in  $T_1$  and  $T_2$ .

Now, by combining Observations (2), (4) and (6), it now follows that each  $ab|c \in \mathcal{I}$  is displayed by  $T_1$  or  $T_2$ .

Guided by  $T_1$  and  $T_2$ , we next construct two caterpillars  $S_1$  and  $S_2$ . For  $i \in \{1, 2\}$ , let  $C'_i$  be the relaxed caterpillar  $T_i(\{0, 1, 2, 3, 4, 5\})$ , and let  $c_i$  be a leaf of the cherry in  $C'_i$ . By Observations (1) and (2), note that  $C'_i$  is indeed a relaxed caterpillar. Now, let  $S_i = T_i$ . For each maximum-size pendant subtree  $S$  of  $S_i$  whose leaf set  $\{x_1, x_2, \dots, x_n\}$  is a subset of  $L(S_i) - \{0, 1, 2, 3, 4, 5\}$ , repeat the following (illustrated in Figure 5) until the resulting trees are both caterpillars. If  $S$  can be detached from  $S_i$  by deleting an edge  $(u_1, v)$  such that  $u_1$  corresponds to a degree-2 vertex on the spine of  $C'_i$ , then delete  $(u_1, v)$ , subdivide the edge directed into  $u_1$  with  $n - 1$  new vertices  $u_2, u_3, \dots, u_n$ , and, for each  $j \in \{1, 2, \dots, n\}$ , add a new edge  $(u_j, x_j)$ . Otherwise,  $S$  can be detached from  $S_i$  by deleting an edge  $(u, v)$  such that  $u$  corresponds to a degree-2 vertex on a leg of  $C'_i$ . Let  $\ell$  be the unique element in  $\{0, 1, 2, 3, 4, 5\}$  such that there is a directed path from  $u$  to  $\ell$  in  $S_i$ . Then, delete  $(u, v)$  and contract the resulting degree-2 vertex  $u$ , subdivide the edge directed into  $\text{lca}_{S_i}(c_i, \ell)$  with  $n$  new vertices  $u_1, u_2, \dots, u_n$ , and, for each  $j \in \{1, 2, \dots, n\}$ , add a new edge  $(u_j, x_j)$ .

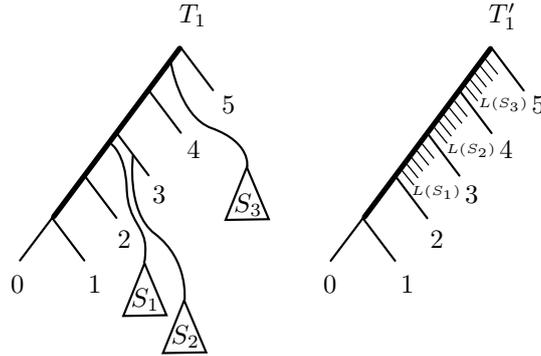


Figure 5: The construction of a caterpillar  $T'_1$  from  $T_1$  as described in the second part of the proof of Theorem 19.

We complete the proof by showing that each triplet  $ab|c \in \mathcal{I}$  is displayed by  $S_1$  or  $S_2$ . Intuitively, the transformation from trees into caterpillars is safe because, for any two leaves in

$L(T_i)$  with  $i \in \{1, 2\}$  for which the ‘above’-relationship holds in  $T_i$ , the ‘above’-relationship also holds in  $S_i$ .

To ease reading in this part of the proof, we pause and introduce new terminology. Let  $S$  be a subtree of  $T_i$  with  $L(S) \subseteq L(T_i) - \{0, 1, 2, 3, 4, 5\}$ . If  $S$  can be detached from  $T_i$  by deleting an edge  $(u, v)$  such that  $u$  corresponds to a degree-2 vertex on the spine of  $C'_i$ , we call  $S$  a *spine subtree* of  $T_i$ . Otherwise, we call  $S$ , an  $\ell$ -*leg subtree* of  $T_i$ , where  $\ell$  is the unique element in  $\{0, 1, 2, 3, 4, 5\}$  such that there exists a directed path from  $u$  to  $\ell$  in  $T_i$ . Now, let  $ab|c \in \mathcal{I}$ .

First, assume that  $0ab|c$  is displayed by  $T_1$ . (Then, because  $1a|ab$  and  $1b|ab$  are definitely displayed by  $T_1$ ,  $ab|c$  is displayed by  $T_1$ .) Since  $T_1$  displays  $0ab|c$ ,  $1a|ab$  and  $1b|ab$ , it follows that there does not exist a spine or leg subtree  $S$  in  $T_1$  with  $\{a, b, c\} \subseteq L(S)$ . Clearly,  $a$  and  $c$  cannot be together in a leg or spine subtree without  $b$ , and similarly  $b$  and  $c$  cannot be together without  $a$ , because this would contradict the fact that  $T_1$  displays  $ab|c$ . So suppose  $a$  and  $b$  are together without  $c$  in a leg or spine subtree. Due to the fact that  $0ab|c$ ,  $1a|ab$  and  $1b|ab$  are displayed by  $T_1$ , the transformation ensures that in  $S_1$  there is a directed path from the parent of  $c$  to  $\text{lca}_{T_1}(a, b)$  i.e. that  $ab|c$  is displayed. Let us then consider the remaining case when  $a$ ,  $b$  and  $c$  are in three distinct subtrees (spine or leg). Observe that it is not possible for all three distinct subtrees to be  $\ell$ -leg subtrees, for the same  $\ell$ . This, again, is because of the three triplets  $0ab|c$ ,  $1a|ab$  and  $1b|ab$ . Hence, at most  $a$  and  $b$  can be in distinct  $\ell$ -leg subtrees, for the same  $\ell$ . Under such circumstances the transformation again ensures that  $ab|c$  will be displayed by  $S_1$ .

Second, assume that  $0ab|c$  is displayed by  $T_2$  (and hence  $ab|c$  is displayed by  $T_2$ ). By replacing the triplets  $1a|ab$  and  $1b|ab$  with  $5a|ab$  and  $5b|ab$ , respectively, we can use an analogous argument to show that  $ab|c$  is displayed by  $S_2$ . In conclusion, each triplet  $ab|c \in \mathcal{I}$  is displayed by the caterpillar  $S_1$  or  $S_2$  and, thus,  $\mathcal{I}$  is a ‘yes’-instance of 2-CATERPILLAR COMPATIBILITY.

By combining both cases, and noting that the construction described in the previous paragraph of this proof can be done in polynomial time, it follows that 3-TREE COMPATIBILITY is NP-complete.  $\square$

## 6 Auxiliary computational proofs used in the hardness reductions

In several of the hardness proofs in Sections 3 and 5, we make use of small instances which, up to symmetry, are the unique solution to the set of constraints they induce. The symmetries differ per problem. For  $\Pi_5$  (betweenness) and  $\Pi_9$  (non-betweenness) the set of constraints induced by a linear order  $\alpha$  is equal to the set of constraints induced by  $\bar{\alpha}$ , so a linear order is indistinguishable from its reversal. For  $\Pi_1$  the first two elements of  $\alpha$  can be swapped without altering the set of constraints induced by the order. For  $\Pi_6$  no symmetries need to be eliminated.

In each case we located these uniquely defined instances by a brute-force search, written in Java. The Java code already verifies the uniqueness property itself, but to increase confidence in the computational proof we then use the constraint programming solver MiniZinc (<http://www.minizinc.org>, see also [20]) as a second check. In particular, the Java code outputs the discovered linear orders  $\{\alpha, \beta\}$  - or in the case of 3- $\Pi_1$ ,  $\{\alpha, \beta, \gamma\}$  - and the set of constraints induced by the linear orders. We then ask MiniZinc to look for linear orders that satisfy these constraints but which differ from  $\{\alpha, \beta\}$  in at least one position. MiniZinc concludes unsatisfiability. If we remove the “differs in at least one place” constraint, and ask MiniZinc to generate all possible solutions, it generates  $\{\alpha, \beta\}$  as the only valid solution. The combination of these results verifies uniqueness.

One of the results in the paper relies on the existence of linear orders  $\{\alpha, \beta, \gamma\}$  such that these are unique instances not just for 3- $\Pi_1$  but also for  $\Pi_1$  on the space of 3 phylogenetic trees. The

proof of uniqueness for  $3\text{-}\Pi_1$  is described above. For the variant on the space of 3 phylogenetic trees, we tackle the proof slightly differently. At a high level, we ask for 3 phylogenetic trees that satisfy all the induced constraints such that at least one of the trees is not a caterpillar. The goal, again, is to generate an unsatisfiability. To encode the “at least one of the trees is not a caterpillar”, we demand that one of the trees contains at least two *cherries*. A cherry is an internal vertex  $u$  such that  $u$  has exactly two children, both leaves. Every phylogenetic tree on at least two taxa has at least one cherry. To search through the space of phylogenetic trees, we observe that the  $\{\alpha, \beta, \gamma\}$  in question each have exactly 6 elements. A rooted binary phylogenetic tree on 6 taxa has exactly 4 internal edges. Each internal edge can be identified by a specific *cluster* i.e. the subset of taxa reachable in a downwards direction from it. The tree is uniquely defined by these 4 clusters. These 4 clusters form a laminar family (which corresponds to the concept of *compatibility*, in the phylogenetics literature). We use this characterisation to search through tree space, noting that a tree contains two or more cherries if and only if two or more of its 4 defining clusters have cardinality 2. MiniZinc, as expected, reports unsatisfiability. When the “at least two cherries” constraint is removed and MiniZinc is asked to generate all solutions, it generates  $\{\alpha, \beta, \gamma\}$  as the only solution (as usual, up to elimination of symmetries).

Full source-code for the computational proofs can be downloaded from <http://skelk.sdf-eu.org/ternary>.

## 7 A constant number of trees is not enough

We begin this section with several new definitions. Let  $T$  be an unrooted binary phylogenetic tree, and let  $T_r$  be a rooted binary phylogenetic tree. We say that  $T_r$  is a *rooting* of  $T$  if  $T_r$  can be obtained from  $T$  by subdividing an edge  $e$  of  $T$  by a new vertex  $\rho$ , which is regarded as the root, and directing all edges away from  $\rho$ . The edge  $e$  is then called the *root location* of  $T_r$  in  $T$ . The next definition introduces a special set of rooted triplets that will play an important role throughout this section. Let  $\mathcal{T}_n$  denote the full set of triplets over  $n$  leaves, i.e.

$$\mathcal{T}_n = \{abc \mid a, b, c \in \{1, 2, \dots, n\}, \ a \neq b \neq c \neq a\}.$$

Furthermore, we use  $\tau(n)$  to denote the minimum number of binary rooted phylogenetic trees of order  $n$ , such that each triplet in  $\mathcal{T}_n$  is displayed by at least one of these trees.

We will show that  $\tau(n) \rightarrow \infty$  when  $n \rightarrow \infty$ . In other words, we show that a constant number of trees does not suffice to display all possible triplet sets.

To prove this, we will use the following theorem shown by Martin and Thatte [17], which improves an earlier bound by Székely and Steel [24]. Let  $umast_k(n)$  denote the smallest number  $L$  such that, for any collection  $\mathcal{C}$  of  $k$  binary unrooted phylogenetic trees of order  $n$ , there exists a binary unrooted phylogenetic tree  $T^*$  of order  $L$  that is displayed by each tree in  $\mathcal{C}$ .

**Theorem 20** (Martin and Thatte). *For some constant  $c > 0$ ,*

$$umast_2(n) > c\sqrt{\log(n)}.$$

To use the above theorem, we first extend it to more than 2 trees.

**Lemma 21.** *For some constant  $c > 0$  and  $k > 2$ ,*

$$umast_k(n) > c\sqrt{\log(umast_{k-1}(n))}.$$

*Proof.* Let  $k \geq 3$ . Consider a collection  $\mathcal{C}$  of  $k$  unrooted phylogenetic trees of order  $n$ . Consider an arbitrary  $k - 1$ -sized subset  $\mathcal{C}'$  of  $\mathcal{C}$  and let  $T^*$  be the only tree in  $\mathcal{C} \setminus \mathcal{C}'$ . Let  $T'$  be an

unrooted phylogenetic tree that is displayed by each tree in  $\mathcal{C}'$  and has order  $\text{umast}_{k-1}(n)$  (i.e. it has a maximum number of leaves). Let  $T^{**}$  be the unique subtree of  $T^*$  that has the same leaf set as  $T'$ . By Theorem 20, there exists an unrooted phylogenetic tree  $T'^{**}$  that is displayed by both  $T'$  and  $T^{**}$  and has order  $\text{umast}_2(\text{umast}_{k-1}(n)) > c\sqrt{\log(\text{umast}_{k-1}(n))}$ . Since  $T'^{**}$  is displayed by each tree in  $\mathcal{C}$ , the lemma follows.  $\square$

The following lemma follows directly from Theorem 20 and Lemma 21 by induction on  $k$ .

**Lemma 22.** *For all  $k \in \mathbb{N}$ ,  $\text{umast}_k(n) \rightarrow \infty$  when  $n \rightarrow \infty$ .*

We now make the step from unrooted to rooted trees by proving the following lemma.

**Lemma 23.** *Let  $T$  be a binary unrooted phylogenetic tree of order  $n \geq 4$  and let  $T_1, \dots, T_k$  be  $k$  rootings of  $T$ . If  $n > k^2 - 6$ , then there exists a triplet in  $\mathcal{T}_n$  that is not displayed by any of  $T_1, \dots, T_k$ .*

*Proof.* Let  $c \geq 2$  be the number of cherries (i.e. pairs of leaves with a common neighbour) of  $T$ . Let  $\ell$  be the length of the longest chain (i.e. a path of which each vertex is adjacent to exactly one leaf) of  $T$ .

We claim that  $T$  has at most  $2c - 3$  chains. To see this, consider the tree  $T^*$  obtained from  $T$  by deleting all leaves and subsequently suppressing all degree-2 vertices. Then,  $T^*$  has  $c$  leaves and hence  $2c - 3$  edges. Each chain of  $T$  corresponds to an edge of  $T^*$  and hence  $T$  has at most  $2c - 3$  chains.

First assume that  $k < 2c$ . Then there exists some cherry  $\{a, b\}$  such that at most one of the edges incident to  $a$  and  $b$  is a root location of at least one of the trees  $T_1, \dots, T_k$  in  $T$ . Say that the edge incident to  $a$  is not a root location and let  $c$  be any leaf distinct from  $a$  and  $b$ . Then it can easily be checked that the triplet  $bc|a$  is not displayed by any of  $T_1, \dots, T_k$ .

Now assume that  $k \geq 2c$ . Tree  $T$  has at most 2 leaves per cherry plus at most  $\ell$  leaves per chain, hence

$$n \leq 2c + \ell(2c - 3).$$

It follows that

$$\ell \geq \frac{n - 2c}{2c - 3} \geq \frac{n - k}{k - 3}.$$

Now, since  $n > k^2 - 6$ , it follows that  $\ell > k + 2$ .

Consider some chain  $C$  of  $T$  with length at least  $k + 3$ . Since there are at most  $k$  edges that are root locations of the trees  $T_1, \dots, T_k$  in  $T$ , there exists some subpath  $(u, v, w)$  of  $C$ , with  $a, b, c$  being the leaves adjacent to  $u, v, w$  respectively, such that the edge from  $v$  to  $b$  is not a root location. Then it can easily be checked that the triplet  $ac|b$  is not displayed by any of  $T_1, \dots, T_k$ .  $\square$

We can now prove the main theorem of this section.

**Theorem 24.**  *$\tau(n) \rightarrow \infty$  when  $n \rightarrow \infty$*

*Proof.* Suppose to the contrary that there exists some natural number  $K$  such that  $\tau(n) \leq K$  for all  $n \in \mathbb{N}$ . By Lemma 22, there exists a natural number  $N \in \mathbb{N}$  such that  $\text{umast}_K(N) > K^2 - 6$ . Let  $\mathcal{C}_r$  be a collection of  $K$  binary rooted phylogenetic trees of order  $N$  such that each triplet in  $\mathcal{T}_N$  is displayed by at least one tree in  $\mathcal{C}_r$ . Such a collection exists by the assumption that  $\tau(N) \leq K$ .

Let  $\mathcal{C}_u$  be the collection of unrooted trees obtained from  $\mathcal{C}_r$  by omitting the orientations of the edges and suppressing the vertices with degree 2 (the former roots). Then there exists some binary unrooted phylogenetic tree  $T_u$  that is displayed by each tree in  $\mathcal{C}_u$  and has order  $\text{umast}_K(N)$ .

Let  $L$  be the leaf-set of  $T_u$  and consider the set of rooted trees  $\mathcal{C}_r|L$  obtained by restricting  $\mathcal{C}_r$  to  $L$ , i.e. for each tree  $T \in \mathcal{C}_r$ , the set  $\mathcal{C}_r|L$  contains the tree obtained from the smallest subtree of  $T$  containing all elements of  $L$  by suppressing all in-degree-1 out-degree-1 vertices. Observe that each tree in  $\mathcal{C}_r|L$  is a rooting of  $T_u$  and has order  $\text{umast}_K(N)$ . Hence, since  $\text{umast}_K(N) > K^2 - 6$ , it follows from Lemma 23 that there exists some triplet  $ab|c \in \mathcal{T}_N$  that is not displayed by any tree in  $\mathcal{C}_r|L$ . Hence,  $ab|c$  is not displayed by any tree in  $\mathcal{C}_r$ . This contradicts our assumption that each triplet in  $\mathcal{T}_N$  is displayed by at least one tree in  $\mathcal{C}_r$ .  $\square$

Let  $\tau_c(n)$  be the minimum number of binary rooted *caterpillar* trees of order  $n$ , such that each triplet in  $\mathcal{T}_n$  is displayed by at least one of these trees. Clearly,  $\tau(n) \leq \tau_c(n)$ . We have the following logarithmic upper bound:

**Lemma 25.**  $\tau(n) \leq \tau_c(n) \leq \left\lceil \frac{\log n(n-1)(n-2) - \log 2}{\log(3/2)} \right\rceil$

*Proof.* It is well known that, given a set of triplets, it is possible to find in polynomial time a tree (in fact, a caterpillar) that is consistent with at least  $1/3$  of the triplets, see [5] for a discussion. Observe that  $|\mathcal{T}_n| = 3\binom{n}{3}$  and that a triplet set containing only one triplet is trivially compatible. Combining these facts shows that  $k$  caterpillars are sufficient to display all triplets in  $\mathcal{T}_n$ , where  $k$  is the smallest integer that satisfies the following inequality:

$$3\binom{n}{3}(2/3)^k \leq 1$$

Rearranging for  $k$  gives the desired result.  $\square$

The lower bound implicit in Theorem 24 and the upper bound in Lemma 25 are very weak. To highlight this we computed  $\tau(n)$  and  $\tau_c(n)$  exactly for small values of  $n$  using Integer Linear Programming (ILP). We defer the details of the ILP to Section 8. The results are shown in the following table. For  $n \geq 13$  the ILP for computation of  $\tau(n)$  did not terminate in reasonable time, but the slightly more constrained ILP for computation of  $\tau_c(n)$  did.

$n$	3	4	5	6	7	8	9	10	11	12	$13 \leq n \leq 20$
$\tau(n)$	3	3	4	4	4	4	4	4	4	4	$\leq 5$
$\tau_c(n)$	3	3	4	4	4	4	4	4	4	4	5
Lemma 25 bound	3	7	9	11	12	13	14	15	16	17	17-21

The possibility thus remains that, for all  $n$ ,  $\tau(n) = \tau_c(n)$ . Resolving this is an interesting open problem. We do however already know that for some triplet sets with fewer than  $3\binom{n}{3}$  triplets the minimum number of caterpillars required is strictly larger than the minimum number of trees required. For example, the set of triplets  $\{13|4, 14|2, 14|3, 23|1, 24|1\}$  requires at least 3 caterpillars but only 2 trees. This can be verified by hand or by a simple adaptation of the ILP.

## 8 An Integer Linear Program for computation of $\tau(n)$ and $\tau_c(n)$

We begin with computation of  $\tau(n)$ . We describe an ILP that, for a fixed integer  $k \geq 1$ , determines whether  $\tau(n) \leq k$ . The ILP has  $3k\binom{n}{3}$  binary variables  $x_{ab|c,t}$  where  $t \in \{1, \dots, k\}$  and  $ab|c \in \mathcal{T}_n$ . Variable  $x_{ab|c,t}$  is 1 if and only if tree  $k$  displays  $ab|c$ . To ensure that every triplet

in  $\mathcal{T}_n$  is displayed by some tree we add the following covering constraint, where  $t$  is assumed to range over  $\{1, \dots, k\}$ ,

$$\sum_t x_{ab|c,t} \geq 1, \quad (1)$$

for every  $ab|c \in \mathcal{T}_n$ . We also need to add constraints to ensure that we correctly model trees. To do this, we observe firstly that for every subset  $\{a, b, c\}$  of 3 distinct leaves a rooted binary tree displays exactly one of the triplets  $ab|c, ac|b, bc|a$ . Hence, for every tree  $t$ , and for every subset  $\{a, b, c\}$  of 3 distinct leaves, we add the constraint

$$x_{ab|c,t} + x_{ac|b,t} + x_{bc|a,t} = 1. \quad (2)$$

Next we make use of the well-known fact that a compatible set of triplets can be characterised by avoidance of local conflicts. In particular, consider a set of triplets in which, for each subset  $\{a, b, c\}$  of 3 distinct leaves, exactly one of the three possible triplets with leaf set  $\{a, b, c\}$  is in the set. Then, as proven in [10], this set is compatible (and thus corresponds exactly to the set of triplets displayed by some rooted phylogenetic tree) if and only if, for every subset of four distinct leaves  $\{a, b, c, d\}$ , the subset of triplets whose leaves are in  $\{a, b, c, d\}$  are compatible. This, again referring to [10], is equivalent to constraining, for every subset of four distinct leaves  $\{a, b, c, d\}$ , that if  $ab|c$  and  $bc|d$  are in the set, then so are  $ab|d$  and  $ac|d$ . This naturally leads to an ILP with  $O(kn^4)$  constraints. Specifically, for every tree  $t$  and for every subset  $\{a, b, c, d\}$  of 4 distinct leaves, we add the constraints

$$x_{ab|c,t} + x_{bc|d,t} - x_{ab|d,t} \leq 1 \quad (3)$$

$$x_{ab|c,t} + x_{bc|d,t} - x_{ac|d,t} \leq 1. \quad (4)$$

This concludes the ILP for computation of  $\tau(n)$ . The ILP for  $\tau_c(n)$  is similar. The only difference is that we add extra constraints which forbid the trees from having two or more cherries. This can be achieved by adding, for every tree  $t$  and for every ordered tuple  $(a, b, c, d)$  of 4 distinct leaves, the constraint

$$x_{ab|c,t} + x_{cd|a,t} \leq 1. \quad (5)$$

## 9 Open problems

For each problem  $2-\Pi_i$  that is polynomial-time solvable, we know that all instances of the problem are “yes” instances, which means that  $k-\Pi_i$  is actually polynomial-time solvable for all  $k \geq 2$ . An obvious conjecture is that for each problem  $2-\Pi_i$  that has been shown to be NP-complete in this article,  $k-\Pi_i$  is actually NP-complete for all  $k \geq 2$ . However, generalizing the gadgetry used in this article will probably require a considerable effort and need to go beyond auxiliary computational proofs. In the same spirit, it seems plausible that  $k$ -TREE COMPATIBILITY is hard for every  $k \geq 3$ . Furthermore, it would be interesting to investigate special cases of  $k-\Pi_i$  that are polynomial-time solvable for  $k \geq 1$ .

The fact that  $\tau(n) \rightarrow \infty$  as  $n \rightarrow \infty$  means that the problem in any case does not become trivially polynomial-time solvable for a sufficiently large, constant number of trees. Other challenges include tightening the bounds on  $\tau(n)$  and  $\tau_c(n)$  and determining whether  $\tau(n) = \tau_c(n)$  for all  $n$ . In the applied domain, it will be interesting to explore whether the slow growth of  $\tau(n)$  has implications for analysis of incongruent biological datasets.

## 10 Acknowledgements

Leo van Iersel and Nela Lekić were supported by Veni and Vrije Competitie grants of The Netherlands Organization for Scientific Research (NWO). Simone Linz was partially supported by a Marie Curie International Outgoing Fellowship within the 7th European Community Framework Programme. We thank the anonymous reviewers for their helpful comments.

## References

- [1] A. V. Aho, Y. Sagiv, T. G. Szymanski, and J. D. Ullman. Inferring a tree from lowest common ancestors with an application to the optimization of relational expressions. *SIAM Journal on Computing*, 10(3):405–421, 1981.
- [2] M. Bodirsky and J. Kára. The complexity of temporal constraint satisfaction problems. *Journal of the ACM (JACM)*, 57(2):9, 2010.
- [3] D. Bokal, G. Fijavz, M. Juvan, P. M. Kayll, and B. Mohar. The circular chromatic number of a digraph. *Journal of Graph Theory*, 46(3):227–240, 2004.
- [4] D. Bryant. *Building Trees, Hunting for Trees, and Comparing Trees: Theory and Methods in Phylogenetic Analysis*. PhD thesis, University of Canterbury, Christchurch, New Zealand, 1997.
- [5] J. Byrka, P. Gawrychowski, K. T. Huber, and S. M. Kelk. Worst-case optimal approximation algorithms for maximizing triplet consistency within phylogenetic networks. *Journal of Discrete Algorithms*, 8(1):65–75, 2010.
- [6] S. Byrka, J. Guillemot and J Jansson. New results on optimizing rooted triplets consistency. *Discrete Applied Mathematics*, 158(11):1136–1147, June 2010.
- [7] B. Chor and M. Sudan. A geometric approach to betweenness. *SIAM Journal on Discrete Mathematics*, 11(4):511–523, 1998.
- [8] A. Erickson and F. Ruskey. Domino tatami covering is NP-complete. In *Combinatorial Algorithms*, pages 140–149. Springer, 2013.
- [9] Z. Galil and N. Megiddo. Cyclic ordering is NP-complete. *Theoretical Computer Science*, 5(2):179–182, 1977.
- [10] S. Guillemot and M. Mnich. Kernel and fast algorithm for dense triplet inconsistency. *Theoretical Computer Science*, 494:134–143, 2013.
- [11] V. Guruswami, J. Håstad, R. Manokaran, P. Raghavendra, and M. Charikar. Beating the random ordering is hard: Every ordering CSP is approximation resistant. *SIAM Journal on Computing*, 40(3):878–914, 2011.
- [12] G. Gutin, L. J. J. van Iersel, M. Mnich, and A. Yeo. Every ternary permutation constraint satisfaction problem parameterized above average has a kernel with a quadratic number of variables. *Journal of Computer and System Sciences*, 78(1):151–163, 2012.
- [13] W. Guttman and M. Maucher. Variations on an ordering theme with constraints. In *Fourth IFIP International Conference on Theoretical Computer Science-TCS 2006*, pages 77–90. Springer, 2006.

- [14] D.H. Huson, R. Rupp, and C. Scornavacca. *Phylogenetic Networks: Concepts, Algorithms and Applications*. Cambridge University Press, UK, 2011.
- [15] Jesper Jansson. On the complexity of inferring rooted evolutionary trees. In *Brazilian Symposium on Graphs, Algorithms and Combinatorics*, volume 7 of *Electronic Notes in Discrete Mathematics*, page 4. 2001.
- [16] S. Linz, K. St John, and C. Semple. Optimizing tree and character compatibility across several phylogenetic trees. *Theoretical Computer Science*, 513:129–136, 2013.
- [17] D. M. Martin and B. D. Thatte. The maximum agreement subtree problem. *Discrete Applied Mathematics*, 161(13):1805–1817, 2013.
- [18] D. A. Morrison. Introduction to phylogenetic networks. *RJR Productions*, 2011.
- [19] L. Nakhleh. Computational approaches to species phylogeny inference and gene tree reconciliation. *Trends in ecology & evolution*, 28(12):719–728, 2013.
- [20] N. Nethercote, P. J. Stuckey, R. Becket, S. Brand, G. J. Duck, and G. Tack. Minizinc: Towards a standard CP modelling language. In *Principles and Practice of Constraint Programming–CP 2007*, pages 529–543. Springer, 2007.
- [21] V. Neumann-Lara. The dichromatic number of a digraph. *Journal of Combinatorial Theory, Series B*, 33(3):265–270, 1982.
- [22] O. Ruepp and M. Holzer. The computational complexity of the KAKURO puzzle, revisited. In *Fun with Algorithms*, pages 319–330. Springer, 2010.
- [23] C. Semple and M. Steel. *Phylogenetics*. Oxford University Press, 2003.
- [24] M. Steel and L. A. Székely. An improved bound on the maximum agreement subtree problem. *Applied Mathematics Letters*, 22(11):1778–1780, 2009.
- [25] Bang Ye Wu. Constructing the maximum consensus tree from rooted triples. *Journal of Combinatorial Optimization*, 8(1):29–39, 2004.