

Reducibility and solvability of some classes of Kryuchkov binary tree pairs

Maria Madonia and Giuseppe Scollo

Department of Mathematics and Computer Science
University of Catania

V.le A. Doria 6, I-95125 Catania, Italy

{madonia,scollo}@dmi.unict.it

Submitted: Apr 30, 2011; Accepted: Dec 22, 2011; Published: Jan 2, 2012

Mathematics Subject Classification: 05C05, 05C15, 05C22, 05C30

Dedicated to Doron Zeilberger on the occasion of his sixtieth birthday

Abstract

This paper addresses the problem of characterizing classes of pairs of binary trees of equal size for which a signed reassociation sequence, in the Eliahou-Kryuchkov sense, can be shown to exist, either with a size induction hypothesis (reducible pairs), or without it (solvable pairs). A few concepts proposed by Cooper, Rowland and Zeilberger, in the context of a language-theoretic approach to the problem, are here reformulated in terms of signed reassociation sequences, and some of their results are recasted and proven in this framework. A few strategies, tactics and combinations thereof for signed reassociation are introduced, which prove useful to extend the results obtained by the aforementioned authors to new classes of binary tree pairs. In particular, with reference to path trees, *i.e.* binary trees that have a leaf at every level, we show the reducibility of pairs where (at least) one of the two path trees has a triplication at the first turn below the top level, and we characterize a class of weakly mutually crooked path tree pairs that are neither reducible nor solvable by any previously known result, but prove solvable by appropriate reassociation strategies. This class also includes a subclass of mutually crooked path tree pairs. A summary evaluation of the achieved results, followed by an outline of open questions and future research directions conclude the paper.

1 Introduction

The Four Colour Theorem (4CT), is a paradigmatic case of potential applicability of methods and results that are offspring of research on translations between different frameworks, because of the several equivalent formulations of the planar map colouring problem. One of the earliest cases of this phenomenon is a well known result by Tait [10, 11], that shows the equivalence

between the 4CT with *vertex colouring* and the 3CT with *edge colouring*. The latter means proper colouring of edges rather than vertices of a planar triangulation, where “proper” means the condition that adjacent edges, i.e. border of a same triangular face, must be assigned different colours, whereas adjacent vertices must be assigned different colours by a proper vertex colouring.

The 4CT was first proven by Appel and Haken[1, 2] using substantial computing resources. The currently most parsimonious proof of the same kind is [9], where reducibility still requires a fairly impressive 633-case analysis. An equivalent statement, of great relevance to the present work, is the Eliahou-Kryuchkov (EK) Conjecture (which actually holds, thanks to the aforementioned proofs of the 4CT, but one should like to get an independent proof of its validity, hoping to find a combinatorially simpler proof of the 4CT that way), that owes its name to the independent findings in [7], where the problem is formulated in terms of signed reassociation of binary trees, and in [4], dealing with signed diagonal flips of planar triangulations. Kauffman [5] has a similar reformulation in terms of the vector cross-product algebra. In his more recent work [6], the connection between that reformulation and the EK Conjecture is elucidated, and a nice feature of the reassociative approach is pointed out; in his words: “Here is a remarkable game! [...] These reassociation moves on the colorings are particularly nice in that they do not involve changing the colors only reconfiguring the graph.” This game is indeed going to be the main business in the present paper.

A language-theoretic approach toward an independent proof of Kauffman’s Theorem is presented by Cooper, Rowland and Zeilberger in [3], which is the main source of inspiration for the present work. An especially interesting feature of their approach is that, besides proving the solvability of specific families of problem instances, in several cases it also provides an explicit count of the number of solutions to given instances, thereby uncovering fruitful links between formal language theory and enumerative combinatorics[12]. The binary parse trees considered in [3] are well suited to represent edge 3-colourings of the cubic graph that is obtained by joining the roots and pairwise, in left-right reverse order, the leaves of two binary trees of equal size that have a common parse word. Tree vertices thus correspond to edges of the cubic graph so obtained, and thereby as well to edges of the triangulation of the sphere where each triangular face bijectively corresponds to a vertex of the cubic graph, and each edge in the triangulation corresponds to that edge in the cubic graph which joins the vertices which correspond to the faces which share the given edge as common border in the triangulation.

Pairs of binary trees of equal size are also well suited to represent the signed reassociation version of the edge 3-colouring problem [4, 7], but under a somewhat different, albeit related labelling and interpretation. The relationship is easily determined by considering that solutions of the edge 3-colouring problem are unique up to colour permutation, whereas in the signed reassociation version of the problem each sign represents one of the two classes of circular permutations of the three colours, viz. clockwise and anticlockwise [8, 5], and solutions are unique up to sign permutation.

A solution of the signed reassociation problem is a sign assignment to the internal vertices of either tree in the pair that allows a sequence of signed reassociation moves that turns its tree structure to that of the other tree in the pair. Signed reassociation moves look like applications of the associativity rule for a binary operation whose terms are represented by the subject trees,

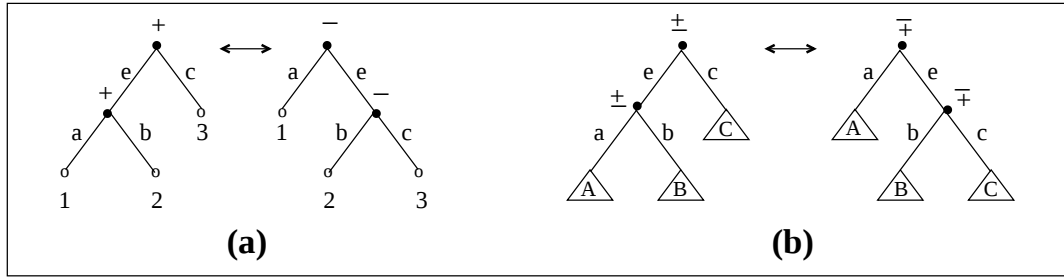


Figure 1: Signed reassociation

but such applications are only allowed if the vertices which are joined by the upper-staying edge involved in the move, which we shall refer to as the *reassociated edge*, are equally signed; their signs are changed (for both vertices) as an outcome of the move.

Figure 1(a) displays a signed reassociation of term $(12)3$ to $1(23)$, while Figure 1(b) gives a pictorial representation of the signed reassociation rule (which is obviously symmetric, viz. signed reassociation moves are reversible), in the general case where the reassociated edge e has sibling and children edges that are not necessarily terminal edges, viz. incident with a leaf. Terminal edge cases are just similar, obtained from this picture by replacing subtrees with leaves.

From any solution of the signed reassociation problem a solution of the 3-colouring problem is determined by an arbitrary colour choice for the root of the binary parse tree, and then by recursively determining the colours of each pair of sibling vertices according to the colour of their parent vertex in the parse tree and to its sign in the signed tree, that tells whether the 3-colour sequence for parent and (ordered) siblings is to be clockwise or anticlockwise. In this way, since the set of solutions of each instance of the signed reassociation problem is closed under sign complementation, from each pair of equivalent (*i.e.* sign-complementary) solutions in this set one gets six colour permutation equivalent solutions of the corresponding instance of the 3-colouring problem.

Sign complementation equivalence of solutions for the signed reassociation problem may be factored out if one replaces vertex sign labels with (in)equality relation labels for internal edges, viz. those edges which connect internal vertices of the binary tree. The so denoted (in)equality relation is indeed that between the signs of the internal vertices joined by the labelled edge. This obviously requires a coherent reformulation of the signed reassociation rule, whereby the reassociated edge of an edge-labelled reassociation move is required to bear the equality label before the move, and keeps that label after the move, whereas any other of its four possible neighbour internal edges, viz. parent, sibling and up to two children edges, gets its label complemented after the move.

Signed (or edge-labelled) reassociation moves preserve leaf order, so it is convenient to endow leaves with identifiers, such as their ordinal position in left-to-right sequential order (or anticlockwise around the tree root). Since solutions are delivered by successful reassociation sequences, as we are going to see, it is as well convenient to endow internal edges with identifiers, in order to represent those sequences. Search of solutions may be carried out by starting with either tree in the given pair as initially unlabelled *source* tree of the reassociation sequence, and

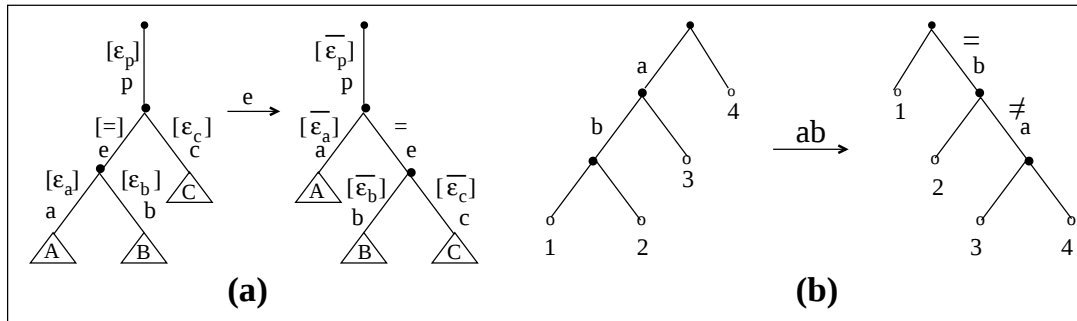


Figure 2: Edge-labelled reassociation

then exploring subsequent reassociation moves aimed at obtaining the other, *target* tree structure; edge labels are introduced “by need” along the search, that is to say, the reassociated edge of a reassociation move is required to bear the equality label or no label before the move, and in both cases it takes the equality label after the move, whereas any other of its four possible neighbour internal edges that is labelled before the move gets its label complemented after the move, whereas it stays unlabelled if so is before the move. In this way, for each given source tree, there is a map which sends each allowed sequence of reassociation moves to a tree of the same size and with a partial labelling of its edges. This induces an equivalence relation on allowed reassociation sequences for the given source tree, that is the map kernel equivalence.

Figure 2(a) depicts the edge-labelled reassociation rule specified above, for the general case where all four neighbour edges may be internal ones, with square brackets denoting optionality of labels, and label complementation denoted by overlining of labels. Again, terminal edge cases are just similar, obtained from this picture by replacing subtrees with leaves (terminal edges are always unlabelled, of course). Furthermore, these rules are oriented, since the optional equality label on edge e in the left-hand-side tree is not optional in the right-hand-side tree—it is “produced” by the rule application. However, these reassociation rules are closed under *chiral symmetry*, that is to say, each rule has a chiral mirror image one, where left-right edge orientations are switched. Figure 2(b) shows the edge-labelled tree produced by an allowed reassociation sequence on a given, unlabelled tree.

(In)equality labelling of internal edges enables parsimonious pictures of problem instances. Binary trees of interest here are *complete* ones, *i.e.* every internal vertex is the parent of two children vertices. One may equivalently say that every internal edge is the parent of two children edges. Now, since only internal edges are labelled, terminal edges may be safely disposed of in the picture of the binary tree (thus roughly halving the number of edges explicitly represented), or replaced by a leaf number when needed. For instance, Figure 3(a) gives the same information as Figure 2(b), but by only displaying the internal edge *backbones* of the subject binary trees, which in this case happen to be path trees (hence backbones are paths).

Figure 3(b) illustrates the solution, produced by the given reassociation sequence for the subject problem instance, expressed as a (total) assignment of (in)equality relation labels to internal edges. In this case, the labeling produced by the reassociation sequence on the target tree happens to be total, therefore it represents a single solution, to the reverse reassociation

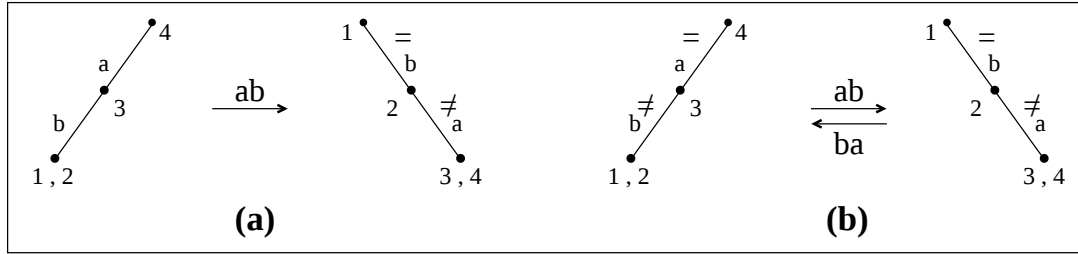


Figure 3: Backbone representation of reassociations and of solutions

problem though. In order to find the corresponding solution to the original problem, viz. a labelling of the source tree, one may apply the reverse reassociation sequence to the labelled target tree just obtained. In general, the labelling of the target tree produced by a successful reassociation sequence need not be total, and it provides a *set of solutions* of cardinality 2^k , where k is the number of unlabelled internal edges in the produced target tree.

2 Basic concepts, terminology and notation

As mentioned at the end of Section 1, allowed reassociation sequences generally produce a partial assignment of (in)equality relation labels to edges of the reassociated tree. A binary tree endowed with such a partial assignment will be referred to as a *constrained tree*. A pair of (possibly constrained) binary trees of the same size is *solvable* if there is a reassociation sequence from either tree, taken as source tree, that is allowed by its constraints and turns its tree structure into that of the other, target tree, while producing a constraint assignment that extends the given one on the target tree, under the appropriate renaming of edges that is determined by the tree structure matching. While the EK Conjecture amounts to solvability of every pair of unconstrained binary trees of equal size, not all pairs of constrained ones are solvable.

A pair of (unconstrained) binary trees of equal size is *reducible* if it is solvable under the size induction hypothesis that all pairs of smaller (yet equally) sized, unconstrained binary trees are solvable. Most often, reducibility of a given tree pair is shown by exhibiting a specific, smaller tree pair, together with an effective way of getting a solution for the given pair out of any solution for the smaller one. Clearly, if a tree pair is reducible to a solvable tree pair, then it is solvable, too.

Owing to closure of reassociation rules under chiral symmetry, both solvability and reducibility are also closed under chiral symmetry. More precisely, let (s', t') be the pair of chiral images of a given tree pair (s, t) ; if the latter is solvable and α is a successful reassociation sequence $s \xrightarrow{\alpha} t$, then also (s', t') is solvable, with the same successful reassociation sequence $s' \xrightarrow{\alpha} t'$, whereas if the solvability of the given pair (s, t) is reducible to that of a smaller-sized pair (s'', t'') , then the solvability of (s', t') is reducible to that of the pair of chiral images of (s'', t'') .

Two relevant cases of reducibility are put forward in [3], viz. *decomposability* of a tree pair, and existence of a *triplication* of the same triple of consecutive leaves in both trees, or, as we shall say, a *shared triplication* of leaves in the pair; that is to say, there exists a triple of leaves

$(i, i+1, i+2)$ whose parent vertices are connected by a 2-edge path in both trees. Those binary tree pairs which have no shared triplication are called *weakly mutually crooked* in [3], whereas *mutually crooked* tree pairs are those which have no *shared duplication*, meaning that no pair of consecutive leaves $(i, i+1)$ have their parent vertices connected by a 1-edge path in both trees. While tree pairs which fail to be weakly mutually crooked are shown to be reducible in [3], the reducibility of the larger class of non mutually crooked tree pairs is only conjectured.

When considered in terms of reassociation sequences, decomposability delivers the possibility of constructing a successful sequence as a parallel shuffle of subsequences thereof, which are defined on pairwise disjoint sets of internal edges; as a matter of fact, one may decompose the subject tree pair into a set of smaller tree pairs which feature mutually independent solutions. It so happens that this possibility also exists in some cases where the subject tree pair is not decomposable in the sense of [3], such as the following simple one.

In Section 5.3 of [3] the decomposability of path tree pairs with common top leaf (*i.e.* taking the same ordinal position in the leaf ordering in the two trees) is pointed out, so these pairs are reducible. One may equivalently say that path tree pairs with *covariant* (that is, equally oriented) top (internal) edges are reducible. This statement is amenable to a small generalization, where reducibility is warranted unless the 2-edge top paths consist of edge pairs that are covariant in each tree and *contravariant* (that is, not covariant) with respect to the other tree. Edges hereafter referred to are internal ones, unless otherwise stated.

Proposition 2.1. *If a path tree pair (s, t) has a shared duplication at the top, or if at least one of their pairs of top leaves is not duplicated, then (s, t) is reducible.*

Proof.

The shared duplication case is an instance of the decomposability case mentioned in Section 5.3 of [3], thus only the case where top edges are contravariant deserves further consideration, with at least one of the trees having its top leaf pair not duplicated by hypothesis. Without loss of generality, one may assume source tree s to be duplication-free at top level (otherwise the following argument applies to the converse pair, with reversibility of successful reassociation sequences completing the proof). Then reassociation of the top edge of s makes this edge covariant with the top edge of t , with both trees having the same leaf at the top level, and the (s, t) solvability problem then reduces to that of the smaller tree pair (s', t') , where s' is the subtree just below the top edge in the tree resulting from the reassociation, while t' is the subtree of t just below its top edge. If e is the top edge of s and α is a successful reassociation sequence for (s', t') , then $e\alpha$ is a successful reassociation sequence for (s, t) , where α exists by size induction hypothesis, since s' is unconstrained. $\square$

It is instructive to see why the path tree pair explicitly considered in the proof of Prop. 2.1 fails to be decomposable in the general case. Figure 4(a) displays the first, top edge reassociation move, in the backbone-only representation of path trees, for one of the two possible relative orientations of contravariant top edges of the two trees (the other case being the chirally symmetric one). It is apparent that the target path tree, *e.g.* the left-hand-side tree in Figure 4(b), does form a decomposable pair with the *e-derivative* of the source tree, *i.e.* the tree resulting from the *e*-reassociation move, rather than with the source tree as given; while the derivative is a constrained tree, the produced constraint is confined to the upper, 1-edge subpath of the tree

backbone. This already matches the corresponding subpath of the target backbone, so no further change is needed there, while the rest of the source tree backbone is wholly unconstrained. Figure 4(b) illustrates top edge reassociation as first move when the first two top edges in the source tree are covariant, viz. the tree has a top level duplication; the derivative is not a path tree in this case, which fact motivates the choice of the path tree with nonduplicated top leaves as source tree in the proof.

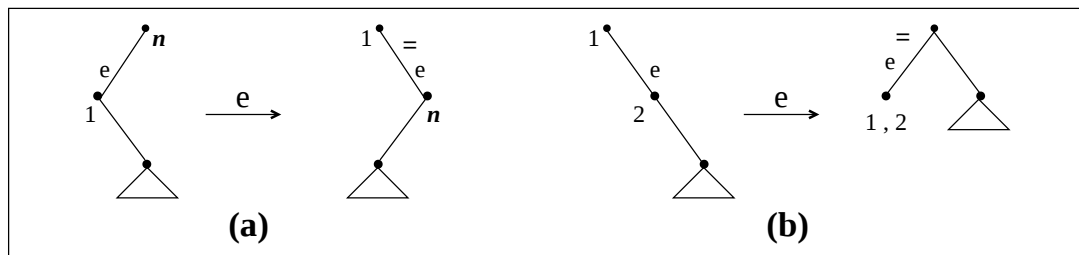


Figure 4: Top edge reassociation as first move

A special case of decomposability of general binary trees, proven in Prop. 9 of [3], gives the opportunity to introduce a basic tool in the development of the present work. The statement under consideration asserts the reducibility of any pair of binary trees (of the same size, this will be henceforth understood) that share a pair of *bottom leaves*, viz. consecutive leaves that are each other's sibling. For example, in Figure 2(b), leaves 1,2 are bottom leaves in the left-hand-side tree, whereas this property is recognized in leaves 3,4 of the right-hand-side tree. A bottom leaf pair shows up in the backbone-only representation of binary trees as the pair of leaves attached to a *bottom edge* in the backbone, that is a backbone edge with no backbone edge underneath. For example, edge e is a bottom edge of the right-hand-side tree in Figure 4(b).

While the cited statement points out the possibility to *expand* any solvable pair of trees by replacing any shared leaf (*i.e.* leaves that take the same ordinal position in the leaf ordering in the two trees) with a shared pair of bottom leaves, the converse operation, which may be termed *bottom edge contraction* in the present context, turns out to be a useful solution search tool. This operation may be applied to both trees in a pair when they share a bottom leaf pair; it consists in replacing the bottom leaf pair, together with the bottom edge incident with it, by a single leaf (by a terminal edge actually, not displayed in the backbone representation of the binary tree), hereby obtaining a pair of trees of smaller size. This operation is applicable regardless of any constraint on the contracted bottom edge in either tree. The single leaf (attached to the terminal edge) which replaces the contracted bottom edge, is referred to as an *abstraction leaf*.

Bottom edge contraction plays also a rôle in a simple proof of yet another reducibility case from [3] (Prop. 10 there), viz. that of any pair of trees where a pair of bottom leaves in either tree is duplicated in the other. This amounts to a reassociation move on the edge joining the duplicated leaves, that turns them into a pair of bottom leaves, thus enabling the subject edge contraction. Such a move is always allowed if it is the first reassociation of the edge in question, otherwise it is only allowed if the edge is labelled by the “=” constraint. In the present context, where the choice of the source tree for the reassociation sequence is made beforehand, it is useful to distinguish between *target bottom edge contraction*, when the duplicated pair of leaves is in the

source tree (while it is a bottom leaf pair in the target tree), and *source bottom edge contraction* in the other case. If e is the contracted edge and α is a successful reassociation sequence for the contracted tree pair, then $e\alpha$ is a successful reassociation sequence for the original tree pair in the case of target bottom edge contraction, whereas αe is such a sequence in the case of source bottom edge contraction. In the latter case, since e has no occurrence in α , the reassociation move, that is needed to turn the bottom leaf pair into a duplicated one after the α reassociation sequence, is always allowed if edge e is unconstrained in the source tree before its contraction.

Reduction to a pair of smaller trees by the aforementioned three cases of bottom edge contraction may be expressed by using the following notation. Let $c \in \{B, SB, TB\}$ denote the bottom edge contraction type, resp. shared Bottom leaves, Source Bottom edge, Target Bottom edge. We write $(s, t) \xrightarrow[c]{[e]} (s', t')$ if one of the following conditions holds:

- c is B and s', t' resp. result from contraction of bottom edge e in s , and of the corresponding bottom edge in t , i.e. that which has the leaf pair in the same ordinal position underneath in t ;
- c is TB, e is the connecting edge, not labelled by “ $\neq$ ”, of two duplicated leaves in source tree s that are bottom leaves in target tree t , and s', t' resp. result from contraction of edge e in s and of the corresponding bottom edge in t ;
- c is SB, e is an unconstrained bottom edge in source tree s , while its corresponding edge in target tree t is the connecting edge of two duplicated leaves, and s', t' resp. result from contraction of edge e in s and of the corresponding edge in t .

Tree pair reduction by bottom edge contraction may occur iteratively, in which case the abstraction leaf resulting from a previous reduction step may itself be involved in further contractions. It is then useful to introduce a convenient notation for abstraction leaves which result as a cumulative outcome of a sequence of edge contractions. Since leaves which are joined in a contraction step are always consecutive ones in the leaf order, it proves handy to let $(m-n)$ denote the abstraction leaf resulting from a sequence of contractions which has joined all leaves in the closed interval $[m, n]$, with $m < n$. Of course, this notation does not tell how the abstraction leaf is to be expanded to the subtree it stands for, neither in derivatives of the original source tree nor in the original target tree. This information may be recovered from the context, viz. the sequence of individual contraction steps which produced the subject abstraction leaf; the respectively corresponding expansion steps then apply in reverse order.

With path tree pairs, reduction to a pair of smaller trees is also viable whenever the top leaf is on the same side as in the target tree, such as it happens, e.g., after the reassociation move in Figure 4(a), with a target tree structure as on the left-hand-side of Figure 4(b). The reduced path tree pair then consists of the subtrees under the respective top edges, which are removed in the reduced tree pair together with the pair of top leaves. Actually, this reduction technique also applies to general binary tree pairs, provided their backbones have only one edge at the top, these top edges in the two trees being covariant. Most often this condition is met by an e -derivative of the source tree, where e is one of its top edges. In reduction steps it is thus convenient to adopt a similar notation as for bottom edge contraction, by convening that T is the top edge contraction type designator. There is a difference, though; the designated edge in a reduction by bottom edge contraction is *not* reassociated, whilst it is so, by our notation, in a reduction by top edge contraction.

3 Reducibility of path tree pairs with triplications

Theorem 1. *If a binary tree pair (s, t) , with s a constrained binary tree, has a shared triplification, where neither of the edges which connect the triplicated leaves in s is labelled by “=”, then (s, t) is reducible.*

Proof.

The solvability of (s, t) is reduced to that of (s', t') , which is obtained from (s, t) by contraction of the triplicated leaves, together with their pair of connecting edges to a single leaf, hereafter referred to as a *triplication leaf*. Since the corresponding expansion is to a 2-edge straight path, it proves necessary to distinguish between its two possible orientations. If $i, i+1, i+2$ are the triplicated leaves, then $i/i+2$ and $i \setminus i+2$ denote the triplication leaf after contraction, together with the contracted edge pair orientation.

Now, let $\alpha = e_1 e_2 \dots e_k$ be a successful reassociation sequence for (s', t') , the following procedure is claimed to map α to a successful reassociation sequence for (s, t) .

Let $s'_0 = s'$, and s'_i be the e_i -derivative of s'_{i-1} , for $1 \leq i \leq k$; let us say that e_i is a *neighbour edge occurrence*, henceforth abbreviated *neo*, of the triplication leaf in α , if edge e_i is incident with (the terminal edge leading to) it, as either parent or sibling, in s'_{i-1} . The triplication expansion procedure replaces each such *neo* with a 3-edge sequence, and progressively applies an appropriate edge renaming permutation to the α suffix following that *neo*, possibly already renamed by previous applications of the same procedure. Both the 3-edge expansion sequence and the permutation depend on what kind of neighbour, viz. parent or sibling, the edge denoted by the *neo* is, in the relevant derivative tree; furthermore, in the case of parent edge, the permutation also depends on the relative orientation of parent edge and triplication leaf (of its contracted edge pair, actually). Let $s_0 = s$, and s_i be the expanded tree which corresponds to s'_i , for $1 \leq i \leq k$. Let h_i and l_i respectively denote the higher and lower edge of the 2-edge path connecting the triplicated leaves in s_{i-1} . Let g_i be the image of e_i under sequential composition of the renaming maps associated to the preceding *neo*'s of the triplication leaf in α . The 3-edge expansion of *neo* e_i is $g_i h_i l_i$ if edge e_i is the parent of the triplication leaf in s'_{i-1} , whereas it is $g_i l_i h_i$ if e_i is its sibling edge. The edge renaming permutation is defined by the following maps in the three subject cases, and is the identity elsewhere:

- (i) covariant parent edge: $g_i \leftrightarrow l_i$;
- (ii) contravariant parent edge: $g_i \mapsto l_i, l_i \mapsto h_i, h_i \mapsto g_i$
- (iii) sibling edge: $g_i \mapsto h_i, h_i \mapsto l_i, l_i \mapsto g_i$

Figure 6 illustrates the application of the expansion rules in the three subject cases, justifies the previous definition of the edge renaming map, and is helpful to quickly see the validity of the stated claim. First, it is apparent that both edges connecting the triplicated leaves after application of an expansion rule are labelled “ $\neq$ ”, and that this is to be their label, if any, before application of the rule; the hypothesis about triplication edge constraints is so justified. Furthermore, expansion rules are context-preserving, under the edge renaming maps, with respect to not only context tree structure, but also neighbour edge constraints, as it may be inspected in the picture. Expansion rules obviously preserve triplication of the expanded leaves, which are thus triplicated in the final derivative produced by the expansion of α . This coincides with the triplication expansion of s'_k , under the edge renaming map resulting from the sequential composition of the renaming maps associated to the triplication leaf *neo*'s in α , by induction on the number of those *neo*'s. $\square$

Any tree pair where one of the trees has a turnfree path backbone is easily solvable; in the case of path trees, with a turnfree backbone target, the solution set is found by the top-down sequence of reassociation moves, starting from the highest source edge that is contravariant

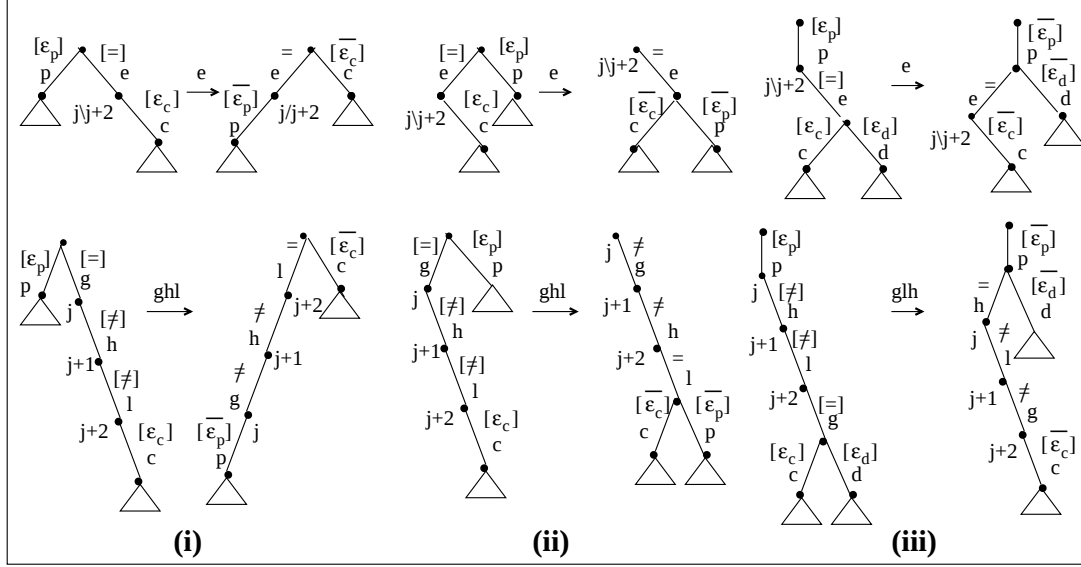


Figure 6: Triplication expansion rules

with the target backbone edge between the same levels. The following result applies to path tree pairs where both tree backbones have turns.

Theorem 2. *If either tree in a path tree pair (s, t) has a triplication at the first turn below the top, then (s, t) is reducible.*

Proof.

In view of the fact established by Prop. 2.1, path tree pairs are assumed to have a nonshared duplication at the top of both trees. Let n be the number of backbone edges in either path tree, thus having $n+2$ leaves. Without loss of generality (by closure of reducibility of binary tree pairs under chiral symmetry), assume the target tree has the sequence of consecutive, pairwise duplicated leaves $1, 2, \dots, h_t$, with leaf 1 at the top level and leaves h_t, h_t+1 not duplicated, whereas the source tree has the sequence of consecutive, pairwise duplicated leaves $n+3-h_s, n+3-(h_s-1), \dots, n+2$, with leaf $n+2$ at the top level and leaves $n+2-h_s, n+3-h_s$ not duplicated, and both $h_t \geq 2$ and $h_s \geq 2$ by the stated assumption.

Let $k_s=1$ if leaves 1, 2 are not duplicated in the source, otherwise let $1, 2, \dots, k_s$ be the sequence of consecutive, pairwise duplicated leaves in the source such that leaves k_s, k_s+1 are not duplicated. Similarly, let $k_t=1$ if leaves $n+1, n+2$ are not duplicated in the target, otherwise let $n+3-k_t, n+3-(k_t-1), \dots, n+2$ be the sequence of consecutive, pairwise duplicated leaves in the target such that leaves $n+2-k_t, n+3-k_t$ are not duplicated.

By Theorem 1, the tree pair may be assumed to have no shared triplication; this entails the following implications hold: $k_s > 2 \Rightarrow h_t = 2$, and $k_t > 2 \Rightarrow h_s = 2$. Now, assume $k_t > 2$ and consider first the case $k_s = 1$, viz. tree backbones as in Figure 7(a). Then the 2-move reassociation sequence $e_2 e_1$ yields a source tree derivative as in Figure 7(b), that shares triplication $n/n+2$ with the target tree, featuring a constraint on the connecting 2-edge path that agrees with the hypothesis of Theorem 1, and with top edge that is covariant with the target tree top edge, whence further reduction is enabled. The solvability of the given pair thus reduces to that of a

pair of smaller, unconstrained path trees in the case $k_s = 1$.

Consider now the case $k_s > 1$, where tree backbones are as in Figure 8(a). Then the 4-move reassociation sequence $e_2 e_1 e_3 e_2$ shows reducibility in this case as well, where the argument depends on whether $k_s = 2$ or $k_s > 2$. In both cases the source tree derivative is as in Figure 8(b). In the first case, this shares triplication $n/n+2$ with the target tree, again featuring constraints that allow triplication reduction, and furthermore it shares the top level duplication of leaves 1, 2 with the target tree, whereby reduction to a pair of smaller, unconstrained path trees is achieved. When $k_s > 2$, one has $h_t = 2$ by the previously stated implications, whereby the source tree derivative shares the two top level duplications of leaves 1, 2, and $n+2, n+1$, with the target tree, and the consequent reduction is again to a pair of smaller, unconstrained path trees.

Finally, the statement also holds in the case where $k_t \leq 2$ and $k_s > 2$, by considering the reverse pair of chiral images of the given trees, and by reversibility of successful reassociation sequences. $\square$

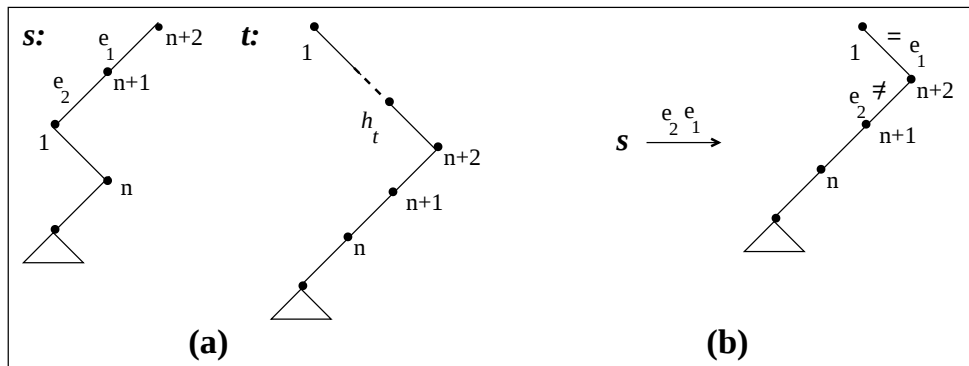


Figure 7: Reducibility in the $k_s = 1$ case

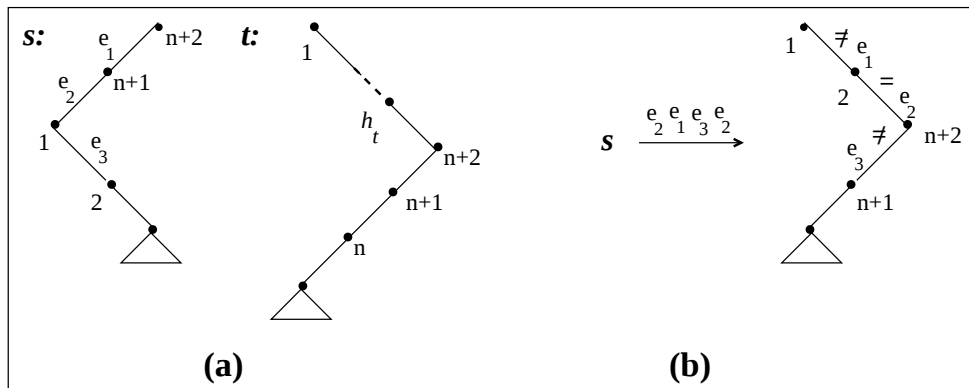


Figure 8: Reducibility in the $k_s > 1$ case

4 Strategies and tactics for constrained reassociation of binary trees

Strictly speaking, in the context of formal derivation systems, a strategy is a function that selects a 1-step derivative out of the set of 1-step derivatives of any well-formed formal expression, whenever that set is nonempty. This concept may be rendered, in the context of the general problem of constrained reassociation of binary trees of the same size, as a deterministic criterion to select a reassociation move out of those allowed for tree s , for any given pair (s, t) of binary trees, where s generally bears (in)equality constraints on its edges. It is useful, though, to consider a 2-level articulation of the strategy concept, where a strategy, in this refined sense, operates at the global level by selecting a *tactical goal* out of a set of viable ones, whereas *tactics*, which may be seen as local strategies in the former sense, select reassociation moves, and are aimed at achieving those goals. The conceptual framework may be further broadened to include *nondeterministic* strategies, where nondeterminism may be introduced at either or both levels: at the local one, by tactics that select a *set* of allowed reassociation moves, *e.g.* to explore parallel search of alternatives to achieve the goal, and/or at the global one, by a strategy that selects a set of tactical goals, *e.g.* to explore parallel search of alternative blends thereof. Finally, *multimove tactics* select allowed *sequences* of reassociation moves, rather than single moves, and may be deterministic (a single sequence) as well as nondeterministic (selecting a set of allowed sequences of reassociation moves).

The *bottom-up joining of target bottom leaves*, henceforth referred to as the *BU strategy*, is a first, fairly simple case of locally nondeterministic strategy. The tactical goal here is to get a source tree derivative where a deterministically chosen pair of consecutive leaves that are bottom leaves in the target tree become siblings (hence bottom leaves) in the source tree derivative as well. The choice is necessarily deterministic for path trees, where there is only one pair of bottom leaves, and may be made deterministic in the general case of binary trees, if so desired, by selecting the target leaf pair on the basis of leaf order.

Let $E_s(i, i+1)$ be the set of edges in the cycle-free path linking target bottom leaves $(i, i+1)$ in (possibly constrained) source tree s . A convenient, nondeterministic tactics for the BU strategy selects the set of reassociation moves with edges in $E_s(i, i+1)$ that are not labelled by “ $\neq$ ” in s , and iterates this selection until emptiness of the selected set. Each such reassociation move shortens by 1 the length of that path, for it removes the reassociated edge from $E_s(i, i+1)$. Thus, the produced reassociation sequences for the given tactical goal are of length at most k , and the goal is achieved if at least one sequence of length k is produced. Such a locally successful sequence may happen to fail globally, however, precisely in the case that the source tree derivative s' it delivers, forms an unsolvable pair (s', t) with the given target tree.

Locally unsuccessful sequences may be swept out of sight by defining the nondeterministic, multimove tactics for the BU strategy as that which selects the set of reassociation sequences which are permutations of the edges in $E_s(i, i+1)$ and are allowed by the constraints on them in s . Of course, such a set may well happen to be empty. When this does not happen, the achievement of the goal enables a bottom edge contraction of type B, by the classification introduced in Section 2 (this does not entail reducibility of the given pair, since the reduced

source tree derivative may bear constraints on its edges). In general binary trees, the abstraction leaf obtained after contraction is not necessarily a bottom leaf in the target tree, but it does happen to be such in path trees; here the strategy progressively moves the target abstraction leaf upwards, along the backbone path.

Finally, one may note that the set of reassociation sequences selected by the multimove tactics, will often contain equivalent alternatives, under the kernel equivalence introduced in Section 1. For example, this surely happens when $E_s(i, i+1)$ has at least 3 edges and these are all unconstrained in s , since $3! = 6$ while the Catalan number $C_3 = 5$ is an upper bound for the number of distinct tree derivatives out of reassociation sequences that are permutations of the edges in $E_s(i, i+1)$, when $|E_s(i, i+1)| = 3$. Implementations of nondeterministic, multi-move tactics ought to take the aforementioned equivalence into account, to prevent unnecessary computational effort.

As an example of the functioning of the BU strategy, consider the path tree pair (s, t) displayed in Figure 9(a). Two solutions are found by the BU strategy for this pair, respectively displayed in Figure 9(c.1) and 9(c.2). Figure 9(b.1) and 9(b.2) illustrate the respective derivation and reduction steps leading to these solutions. The notation adopted is similar to that employed in Figure 5, except that target tree reducts are omitted (as mentioned above, they are easily obtained by stepwise moving the target abstraction leaf upwards, along the backbone path), and so is the indication of the default contraction type, which is TB, inherently to tactical goals in the BU strategy, except that when the abstraction leaf happens to have the same leaf sibling in the source reduct as in the target one, then reduction takes place by an edge contraction of type B, in which case the contraction type is explicitly indicated (no reassociation move goes along with the reduction in this case).

Each of the reassociation and reduction steps displayed in Figure 9(b.1) and 9(b.2) relates to the choice of a multimove reassociation sequence out of the set of the allowed ones selected by the BU tactics. The initial set consists of the two sequences de and ed , respectively developed in Figure 9(b.1) and 9(b.2). The former features a set of four allowed multimoves after the second step, but two of them are equivalent, viz. bdc and dbc ; this fact is represented by specifying the two sequences using the shuffle operator “|”, with left-associative parsing. Furthermore, two out of the three reduced derivatives turn out to have the empty set of allowed multimoves for the fourth step, as represented by the “ ϕ ” occurrences in the picture. The third reduced derivative out of the third step, on the contrary, proves globally successful eventually. Similar notes apply to the two developments in Figure 9(b.2), only one of which yields a solution for the given tree pair. The reader may verify that the two successful final derivative path trees are those depicted on the right of Figure 9(c.1) and 9(c.2), respectively produced by reassociation sequences $defcbdac$ and $edfecbfcaf$, and that application of the respective reverse sequences to these path trees yields the solutions depicted on the left of Figure 9(c.1) and 9(c.2), respectively.

The *top-down resizing of sibling subtrees*, henceforth referred to as the *TD strategy*, is a second, somewhat more complex case of locally nondeterministic strategy. The tactical goal here is to get a source tree derivative where the left subtree of the root has the same size as that in the target tree, and therefore similarly does the right subtree with respect to its target counterpart. Top-down iteration of the tactical goal along the tree structure, when successful, eventually results in identical tree structures.

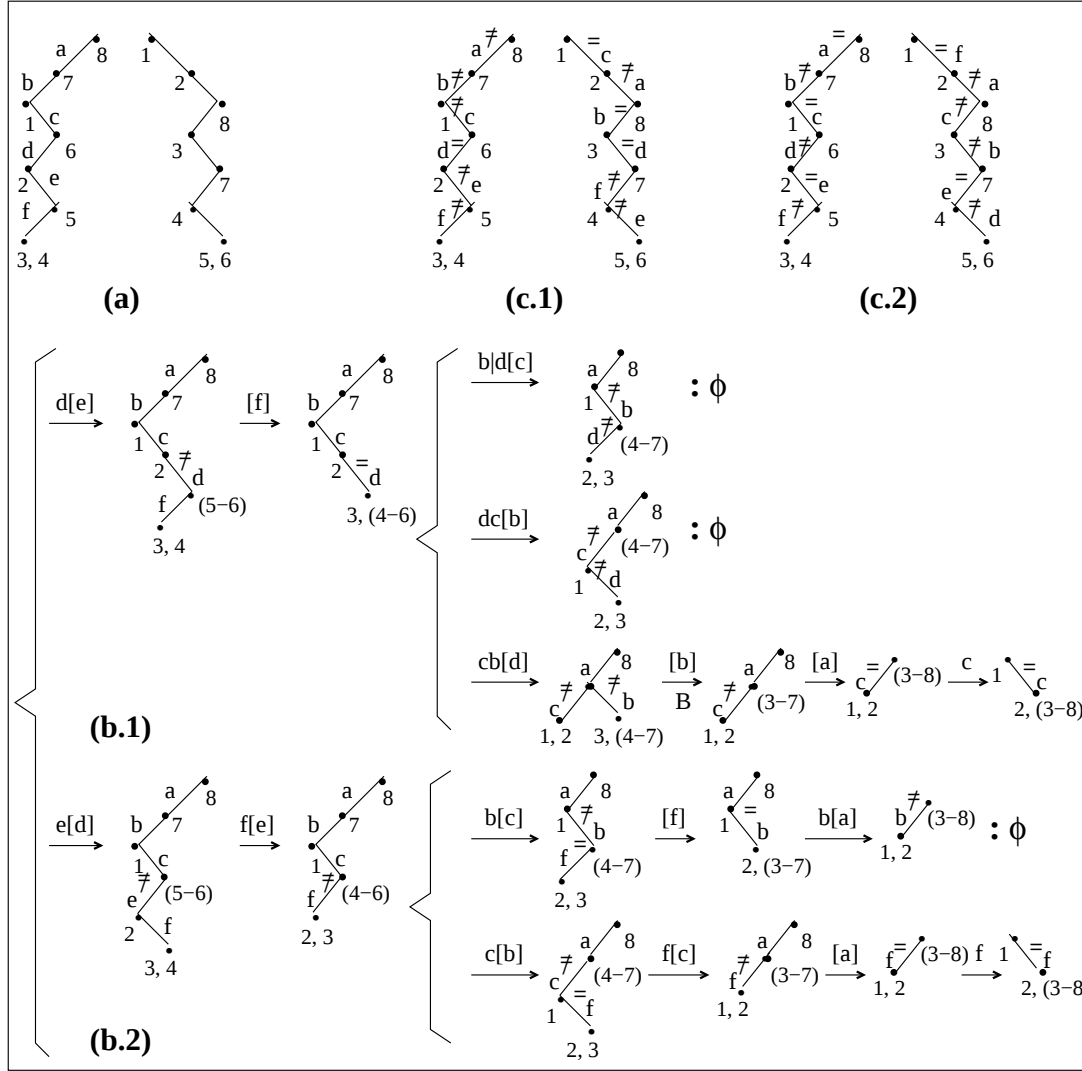


Figure 9: Solutions by the BU strategy

The identification of effective tactics for the TD strategy requires some extra care, in comparison with the BU strategy case, for the achievement of the tactical goal generally requires the transfer of a certain number of leaves from one of the two top subtrees to the other. A couple of definitions and related facts will enable a quick identification of appropriate multimove tactics for the TD strategy.

Definition 1. Let (s, t) be a pair of binary trees of the same size, where s has l_s leaves in its top left subtree and t has l_t leaves in its top left subtree. The **resizing edge set of s to t** , $E_{s,t}$, is defined as the empty set if $l_s = l_t$, otherwise as the set of edges of s in its **resizing path to t** , that is the path linking the root of s to its **resizing bottom** vertex, which is the closest common ancestor of leaves l_t, l_t+1 in s .

Note that the resizing bottom is the parent of leaf l_t if both l_t and l_t+1 are left leaves in s ,

while it is the parent of leaf l_{t+1} if both l_t and l_{t+1} are right leaves in s . If $p_{s,t}$ is the resizing path of s to t , we let $|p_{s,t}| = |E_{s,t}|$ denote its length.

As we are going to see, certain permutations of the edges in $E_{s,t}$ define reassociation sequences that, if allowed by constraints in s , yield derivatives where the respective sizes of the two top sibling subtrees are the same as in the target tree. Since not all permutations of $E_{s,t}$ edges always enjoy this property, it is useful to characterize those which do. To this purpose, the language of *postfix concatenation/shuffle expressions* is a useful tool. Terminal symbols of this language are a set of edge names; let E be a nonterminal symbol with productions $E ::= e$ for all edge names e . The subject language is then generated from the start symbol S by the following three further productions: $S ::= \epsilon$, $S ::= SE$, $S ::= S|E$.

The intended meaning $\llbracket S \rrbracket$ of a language expression S is a set of sequences of edge names, that is defined below. Two binary infix operators form language expressions, viz. juxtaposition, which denotes sequential concatenation, and “|”, which is a parallel shuffle operator. The syntactic restriction on both operators, to take a single edge name rather than a language expression as their right argument, forces left-associative parsing of language expressions, where no parentheses are thus needed.

The meaning $\llbracket S \rrbracket$ of language expressions, ranged over by S , is defined by structural induction, as follows. The set denoted by a shuffle-free expression is the singleton containing that expression as the only sequence in the set. Then $\llbracket Se \rrbracket$ consists of all sequences that are formed by postfixing e to a sequence in the set $\llbracket S \rrbracket$, while $\llbracket S|e \rrbracket$ consists of all sequences that are formed by inserting an e occurrence at an arbitrary position of a sequence in the set $\llbracket S \rrbracket$, while keeping unchanged the relative order of the elements which form that sequence.

Definition 2. Let (s, t) be a pair of binary trees of the same size, with $E_{s,t}$ its resizing edge set of s to t , and s possibly constrained.

- (i) The **resizing set specification** of s to t , $S_{s,t}$, is the postfix concatenation/shuffle expression that is obtained from the sequence of edges in $E_{s,t}$ in bottom-up order, from the resizing bottom vertex up to the root, by inserting a “|” in between each pair of adjacent names of edges that are covariant in s .
- (ii) The **resizing set** of s to t , $R_{s,t}$, is the subset of those sequences in $\llbracket S_{s,t} \rrbracket$ which specify allowed reassociation sequences, under the constraints that edges in $E_{s,t}$ have in s .

Once equipped with the previous definitions, the identification of a nondeterministic, multi-move tactics for the TD strategy proves easy, as it boils down to state that the selected set is the resizing set of the source to the target tree. Whenever this set is nonempty, the goal is achieved by each reassociation sequence in the resizing set, as it is going to be shown, and this enables a twofold top edge contraction of the given pair, to the respective tree pairs (with constrained source trees) formed by the top left subtrees and by the top right subtrees of the source derivative tree and given target tree.

Please note that, according to definitions 1 and 2, if the resizing edge set $E_{s,t}$ is empty, then the resizing set specification is the empty sequence, and the resizing set $R_{s,t}$ is nonempty, as it is the singleton containing the empty sequence.

One may further note that if no edge but (either or both of) the two top ones is constrained in the source tree derivative, then the given given tree pair is reducible. This observation enables the following straightforward generalization of Prop. 2.1.

Proposition 4.1. *If a binary tree pair (s, t) has $E_{s,t}$ or $E_{t,s}$ of cardinality ≤ 2 , and in the case of cardinality 2, with shuffle-free $S_{s,t}$ or $S_{t,s}$, i.e. the 2-edge resizing path has a turn, then the pair (s, t) is reducible.*

Proof.

The resizing set is a singleton, and the constrained tree derivative which results from the specified reassociation sequence has no constraints outside the top edges, which are removed by the subsequent top edge contraction, so the solvability of (s, t) is reduced to that of two pairs of smaller, unconstrained trees. $\square$

It will often be useful to view a backbone path as composed of a sequence of *path segments*, where a segment is a maximal contiguous subsequence of covariant edges in the backbone path. Segments are thus delimited by *turns* in the path, but for extremal segments, which are (also) delimited by the root and by the bottom vertex in the path.

A formal motivation of the proposed tactics for the TD strategy is provided by the next theorem, which is introduced by a couple of useful lemmas.

Lemma 4.1. *Let (s, t) be a pair of binary trees of the same size, where s has l_s leaves in its top left subtree and t has l_t leaves in its top left subtree, with $l_s \neq l_t$; let ρ be a permutation of a subset E_ρ of the edges in s that specifies an allowed reassociation sequence for s . Then the ρ -derivative of s has l_t leaves in its top left subtree only if $E_{s,t} \subseteq E_\rho$.*

Proof.

By hypothesis, the sequence specified by ρ separates leaf l_t , living in the top left subtree of the ρ -derivative of s , from leaf l_t+1 , living in the top right subtree of the ρ -derivative of s . By contradiction, suppose there exists $e \in E_{s,t}, e \notin E_\rho$. Since $e \in E_{s,t}$, e belongs to the resizing path of s to t and, therefore, both leaves l_t and l_t+1 are in the subtree of s rooted at the lower vertex incident with e , for this vertex is in the resizing path, which ends at the closest common ancestor of leaves l_t and l_t+1 . Since e does not occur in ρ , edge e is not reassociated in the sequence specified by ρ , hence it still holds in the ρ -derivative of s that all leaves of that subtree are the leaves of the subtree rooted at the lower vertex incident with e , therefore leaves l_t and l_t+1 are in the same top subtree (left or right) of the ρ -derivative of s , against the hypothesis. $\square$

Let s be a binary tree and let $p = e_1 \dots e_n$ be a path in s starting from the root. For any i , $1 \leq i \leq n$, edge e_i in p meets one and only one of the following *variance conditions*:

- (i) e_i is the last edge in p , i.e. $i = n$,
- (ii) e_i is not the last edge in p , i.e. $i \neq n$, and e_i is covariant with e_{i+1} ,
- (iii) e_i is not the last edge in p , i.e. $i \neq n$, and e_i is contravariant with e_{i+1} .

Remark 4.1. *Let (s, t) be a pair of binary trees of the same size and let $p = e_1 \dots e_n$ be the resizing path of s to t . Let s' be the e_i -derivative of s for some i , $1 \leq i \leq n$, and let p' be the resizing path of s' to t . By the reassociation rule, we have that $|p'| = |p|$ iff e_i meets variance condition (iii) in p (in which case e_i belongs to p'), and that $|p'| = |p| - 1$ iff e_i meets either variance condition (i) or (ii) in p (in both cases e_i does not belong to p').*

Lemma 4.2. *Let (s, t) be a pair of binary trees of the same size, where s has l_s leaves in its top left subtree and t has l_t leaves in its top left subtree, with $l_s \neq l_t$. Let $\rho = e_{i_1} \dots e_{i_n}$ be a*

permutation of $E_{s,t}$ that specifies an allowed reassociation sequence for s . Moreover, let $s_0 = s$, let s_j be the e_{i_j} -derivative of s_{j-1} , for $1 \leq j \leq n$, and let p_j be the resizing path of s_j to t , for $0 \leq j \leq n$. Then the ρ -derivative of s has l_t leaves in its top left subtree iff, for every $1 \leq j \leq n$, e_{i_j} does not meet variance condition (iii) in p_{j-1} .

Proof.

It suffices to observe that the ρ -derivative of s , viz. s_n , has l_t leaves in its top left subtree iff p_n , the resizing path of s_n to t , has length 0. But this is true iff, for every $1 \leq j \leq n$, $|p_j| = |p_{j-1}| - 1$ (since $|p_0| = |E_{s,t}| = n$) iff, by Remark 4.1, for every $1 \leq j \leq n$, e_{i_j} does not meet variance condition (iii) in p_{j-1} . $\square$

Theorem 3. Let (s, t) be a pair of binary trees of the same size, where s has l_s leaves in its top left subtree and t has l_t leaves in its top left subtree, with $l_s \neq l_t$; let ρ be a permutation of $E_{s,t}$ that specifies an allowed reassociation sequence for s . Then the ρ -derivative of s has l_t leaves in its top left subtree iff $\rho \in R_{s,t}$, viz. iff $\rho \in \llbracket S_{s,t} \rrbracket$.

Proof.

Let $p = e_1 \dots e_n$ be the resizing path of s to t , and suppose $\rho = e_{i_1} \dots e_{i_n}$. Let $s_0 = s$, and let s_j be the e_{i_j} -derivative of s_{j-1} and p_j be the resizing path of s_j to t , for $0 \leq j \leq n$. By Lemma 4.2, it suffices to show that $\rho \in \llbracket S_{s,t} \rrbracket$ iff for every $1 \leq j \leq n$, e_{i_j} does not meet variance condition (iii) in p_{j-1} .

Suppose first that p is turnfree; then, by the reassociation rule, no e_{i_j} -reassociation in the sequence specified by ρ may change the orientation of any edge in p_j with respect to that which it has in p_{j-1} (the orientation of e_{i_j} itself does change, but e_{i_j} does not belong to p_j), therefore every p_j is turnfree, for $0 \leq j \leq n$, which entails that no e_{i_j} may ever meet variance condition (iii) in p_{j-1} , for $1 \leq j \leq n$, while, on the other hand, every permutation of $E_{s,t}$ is in $\llbracket S_{s,t} \rrbracket$, since p is turnfree.

Suppose now that p is not turnfree, hence there is an edge e_h , with $1 \leq h < n$, that meets variance condition (iii) in p . For any edge below it in p , say e_k with $h < k \leq n$, let $f_0 = e_h$ and $f_1, \dots, f_m$ be the possibly empty, not necessarily contiguous subsequence of p that consists of those edges which follow e_h and precede e_k in p , and meet variance condition (iii) in p .

Assume no e_{i_j} meets variance condition (iii) in p_{j-1} , for $1 \leq j \leq n$. Then, for each $0 \leq q \leq m$, the reassociation of every edge in the segment just following f_q in p must precede that of f_q in the sequence specified by ρ , since the orientation of f_q may only change by its own reassociation (the alternative being a reassociation of its contravariant predecessor edge e_{i_j} in some p_{j-1} , against the current assumption), while if some of the edges in the segment following f_q in p is not reassociated before f_q , then the reassociation of f_q would violate the current assumption. We may thus infer that e_k precedes f_m in ρ , and, if $m > 0$, that f_q precedes f_{q-1} in ρ , for $1 \leq q \leq m$, whence e_k precedes $f_0 = e_h$ in ρ by precedence transitivity. Since this holds for every e_h that meets variance condition (iii) in p and every e_k that follows e_h in p , we may conclude that $\rho \in \llbracket S_{s,t} \rrbracket$, by the semantics of postfix concatenation/shuffle expressions, recalling their left-associative reading and the way $S_{s,t}$ is obtained from the resizing path p .

Conversely, assume $\rho \in \llbracket S_{s,t} \rrbracket$, then the aforementioned semantics entails that e_k precedes e_h in ρ whenever e_h meets variance condition (iii) in p , for all $h < k \leq n$. By contradiction, assume e_{i_j} meets variance condition (iii) in p_{j-1} for some $1 \leq j \leq n$, and consider the smallest such j .

Then, one of the following cases should be inferred about the successor of e_{i_j} in p_{j-1} : 1) It is reassigned before e_{i_j} , but still is in the resizing path p_{j-1} (cf. Remark 4.1); then this successor edge should be $e_{i_{j'}}$, for some $j' < j$, and it should meet variance condition (iii) in $p_{j'-1}$, against the stated minimality of j ; clearly, this case is to be rejected. 2) The successor of e_{i_j} in p_{j-1} is reassigned after e_{i_j} ; by the stated minimality of j , for every $1 \leq j' < j$, $e_{i_{j'}}$ meets variance condition (i) or (ii) in $p_{j'-1}$, which fact entails that every edge in p_{j-1} has the same orientation it has in p , and that their relative ordering in p_{j-1} is the same as in p ; then, the occurrence of e_{i_j} in ρ precedes in ρ that of an edge, its successor in p_{j-1} , that is contravariant with it in p , by the aforementioned preservation of edge orientation, and where it follows e_{i_j} , by the aforementioned preservation of edge relative ordering in the resizing path. Two final subcases arise from this inference: 2.1) e_{i_j} meets variance condition (iii) in p , in which case $\rho \notin \llbracket S_{s,t} \rrbracket$, against the assumption; 2.2) e_{i_j} meets variance condition (ii) in p , in which case all edges below it in its segment in p are reassigned before it (since it meets variance condition (iii) in p_{j-1}), including the lowest one of them, say e' ; this one is thus reassigned before e_{i_j} , which is reassigned before its successor in p_{j-1} ; then, by precedence transitivity, e' is reassigned before the successor of e_{i_j} in p_{j-1} , but the latter edge follows e' in p , where e' meets variance condition (iii); whence the same absurd conclusion is drawn as in the previous subcase, viz. $\rho \notin \llbracket S_{s,t} \rrbracket$. $\square$

The specialization of the TD strategy for path trees enjoys the property that each tactical goal boils down to get that all leaves but one be placed in one of the two top-level subtrees, the other consisting of the top leaf. The achievement of the goal then enables top edge contraction, whereby the root of the reduced target tree progressively moves downwards, along the backbone path. In the case of path trees, thus, BU and TD strategies feature a sort of symmetric dynamics. Figure 10 illustrates the application of the TD strategy to the same path tree pair considered in Figure 9, with similar notational conventions, except that here the default contraction type is T. When no edge reassociation is required for top edge contraction, viz. $R_{s,t} = \{\epsilon\}$, then this fact is made explicit by subscripting the reduction step with “ $\equiv$ ”. Three solutions are found by the TD strategy in the subject example, resp. depicted in Figure 10(c.1), 10(c.2), 10(c.3); none of them coincides with any of those found by the BU strategy. The reader may verify that the path trees on the right of Figure 10(c.1), 10(c.2), 10(c.3), are respectively produced by reassociation sequences $abdcaefd$, $abcdacfeaf$, $abcdacefae$, and that application of the respective reverse sequences to them yields the solutions on the left of Figure 10(c.1), 10(c.2), 10(c.3), respectively.

The nondeterminism of tactics employed in both strategies introduced above, entails that either of them may deliver a set of solutions, not only because the final target constraint assignment may be partial, but also because a set of not necessarily equivalent reassociation sequences may prove globally successful. This possibility raises the principle question whether any of these strategies is *complete*, that is, able to deliver the whole set of solutions for any given pair of binary trees. The examined case settles the answer to this question in the negative, both for the BU and for the TD strategy, where either strategy proves unable to find solutions found by the other. This motivates the consideration of a third strategy, which subsumes the previous ones, while featuring a higher degree of nondeterminism at the global level. This strategy essentially is the *nondeterministic combination of BU and TD strategies*, hereafter shortly referred



The definition of the BU|TD strategy entails that the set of solutions it delivers always includes those delivered by the standalone BU and TD strategies; however, it may also contain solutions found by neither of them. This is hard to testify in the previous case, where *e.g.* the BU|TD strategy gets the solution displayed in Figure 11, by the reassociation sequence `edfeabcfac`. It appears, though, that this solution is the same as the third one delivered by the TD strategy, resulting from the reassociation sequence `abcdacefae`, as displayed in Fig-

- (i) the BU strategy finds only one solution, that is displayed in Figure 12(b), by the reassociation sequence fedcbgfcafg ;
- (ii) the TD strategy finds two solutions, as displayed in Figure 12(c.1) and 12(c.2), respectively by reassociation sequences $\text{ba}(\text{cb}|\text{ed})\text{fegbec}$ and badcbefgbfcd ;
- (iii) the BU|TD strategy also finds the solution displayed in Figure 12(e), by the reassociation sequence febadcbfgb , whose development is presented in Figure 12(d). It is apparent that this reassociation sequence is not equivalent to any of those found by the BU and TD strategies.

The question about completeness of the BU|TD strategy is open. A perhaps more interesting principle question relates to the *effectiveness* of the subject strategies, where a strategy is effective if it always finds some solution for solvable tree pairs. It is not yet known whether any of the strategies defined above is effective. Note that these questions are independent of the validity of the EK Conjecture, since neither completeness nor effectiveness of any strategy ensures that the set of solutions is never empty.

5 Strategies that solve path tree pairs with a shared bottom leaf

While solution search strategies considered so far are nondeterministic, and apply to pairs of general binary trees, in the rest of this work deterministic strategies are investigated, that prove suitable to *solve* specific classes of path tree pairs. The first such class consists of those pairs of path trees which share a bottom leaf. Pairs that share both bottom leaves are reducible, since they are decomposable (by an instance of Proposition 9 in [3]), while pairs that share only one bottom leaf are known to be solvable, by Proposition 8 in [3], where the existence of a common parse word for the subject path trees is shown. The translation of that proof in the present framework proves very instructive and fruitful, since it

- uncovers two related deterministic strategies that always find a successful reassociation sequence, for any pair in the subject class,
- enables a general characterization of the solutions found by the aforementioned strategies, that only depends on path tree structure,
- provides a basis for the investigation of other, related deterministic strategies, that prove suitable to solve certain classes of path tree pairs which share no bottom leaf (as it is shown in the next section).

These goals justify the fact that, in this section, neither decomposable pairs nor otherwise reducible pairs are excluded from consideration, since strategies and solutions to be dealt with, apply to the whole class of equally sized path tree pairs with a shared bottom leaf. The second item in the previous list is first addressed, by proposing the following definition.

Definition 3. A (partial) assignment of (in)equality constraints to backbone edges of a path tree is **sequential** if every nonbottom edge in the backbone is either unconstrained or constrained by

- (i) “=” if it is contravariant with the adjacent edge below it,

(ii) “ $\neq$ ” if it is covariant with the adjacent edge below it.

The following lemma is a first, useful tool to prove the present reformulation of Proposition 8 of [3].

Lemma 5.1. *Let s be a path tree with sequential constraint assignment, and e either the backbone bottom edge not constrained by “ $\neq$ ”, or a nonbottom edge that is contravariant with the adjacent edge below it. Then the e -derivative of s is a path tree with sequential constraint assignment.*

Proof.

Immediate from Def. 3 and the properties of reassociation under the stated assumptions about edge e . $\square$

Any constraint assignment to the bottom edge of the backbone path plays no rôle in the previous definition, but it does play an essential one in the characterization of those sequential constraint assignments that warrant solvability of a pair (s, t) of path trees that share exactly one bottom leaf, and where either tree in the pair may be constrained. A trivial remark is a useful introduction to the forthcoming definition, viz. that if two path trees of equal size share just one bottom leaf, then this lies on opposite sides in the two trees.

Definition 4. *A pair of path trees of equal size, with contravariant top edges and that share exactly one bottom leaf, is*

- (i) *a **majority pair** if the shared bottom leaf lies on the same side as the top leaf, in both trees;*
- (ii) *a **minority pair** if the shared bottom leaf lies on the opposite side to that of the top leaf, in both trees.*

Top leaves in a pair of path trees with contravariant top edges lie on opposite sides, therefore, by the previous remark, the single shared bottom leaf must lie either on the same side as the top leaf in both trees, or on the opposite side, again in both trees. The terminology adopted in the previous definition is justified by the fact that, for any fixed path tree size, the set of pairs of path trees of that size, that have contravariant top edges and share a single bottom leaf, is partitioned into the two subclasses which are resp. characterized by the conditions stated in Def. 4, and that the first class is always larger than the second one (even in the smallest case, viz. of 1-edge backbone, where the majority class is populated by a single pair, up to chiral symmetry, while the second class is empty).

The following *bottom edge rule* characterizes those sequential constraint assignments which warrant solvability of path tree pairs with contravariant top edges, a single shared bottom leaf, and where either tree in the pair may be constrained, thanks to a forthcoming result. It is just amazing that the dichotomy drawn by this rule is essentially determined by the relative orientation of the two farthest edges in the backbone path. Please note that there is no loss of generality in the contravariance assumption about top edges, since moveless top edge contraction applies to pairs with covariant top edges, and any partial constraint assignment that solves the pair of so reduced path trees does also solve the original pair, with the top edge left unconstrained. This preserves sequentiality of the constraint assignment, according to Def. 3.

Definition 5. A sequential constraint assignment to backbone edges of either path tree, in a pair of equal-size path trees that have contravariant top edges and share a single bottom leaf, satisfies the **bottom edge rule** if the bottom edge is either unconstrained, or else

- (i) constrained by “=” iff it is covariant with the top edge in the backbone path, when the given pair is a majority pair;
- (ii) constrained by “=” iff it is contravariant with the top edge in the backbone path, when the given pair is a minority pair.

The forthcoming terminology is based on the notion of path segment, introduced above. The following acronyms (kept lowercase for typographical convenience) will designate two deterministic strategies for path tree pairs, as defined below:

tdbus: top-down (strategy with) bottom-up (tactics for multimove reassociation of) segments;

butds: bottom-up (strategy with) top-down (tactics for multimove reassociation of) segments.

Definition 6. Let (s, t) be a pair of path trees of the same size. Then two deterministic **sequential strategies** are defined as follows:

- (i) the **tdbus** strategy is the deterministic specialization of the TD strategy for path trees, where tactics only select multimove reassociation of edges in bottom-up order, when this is allowed, otherwise the selection is undefined;
- (ii) the **butds** strategy is the deterministic specialization of the BU strategy for path trees, where tactics only select multimove reassociation of edges in top-down order, when this is allowed, otherwise the selection is undefined.

The next lemma tells in what sense do sequential strategies prove effective under, and preserve, validity of the bottom edge rule.

Lemma 5.2. Let (s, t) be a pair of equal-size path trees that have contravariant top edges and share a single bottom leaf, with a sequential constraint assignment in s that satisfies the bottom edge rule. Let α be the edge reassociation sequence selected on s by the tactics of either sequential strategy as per Def. 6. Then α is allowed by the constraints in s , and the path trees in the pair (s', t) , with s' the α -derivative of s , either have identical tree structure or reduce without further reassociation to a pair (s'', t') of equally sized, smaller path trees that either have identical tree structure, or have contravariant top edges, share a single bottom leaf, and with a sequential constraint assignment in s'' that satisfies the bottom edge rule.

Proof.

Each subsequent move of the α sequence satisfies the hypothesis of lemma 5.1, so, if allowed, α preserves sequentiality of the constraint assignment. Figure 13 displays multimove reassociations by the two kinds of sequential strategy defined above, in the case when the sequence of reassociated edges does not include the bottom edge; it is apparent that, in this case, α is always allowed, thanks to sequentiality of the constraint assignment in the source path tree and because α starts with a nonbottom edge that is contravariant with the adjacent edge below it, and this condition is preserved for all subsequent edges in α in the corresponding derivative trees. Note that, when $k=1$, the optional constraint on edge e_1 in the source tree of Figure 13(a) is “=”, and so is the resulting constraint on edge e_1 in the derivative tree in Figure 13(b).

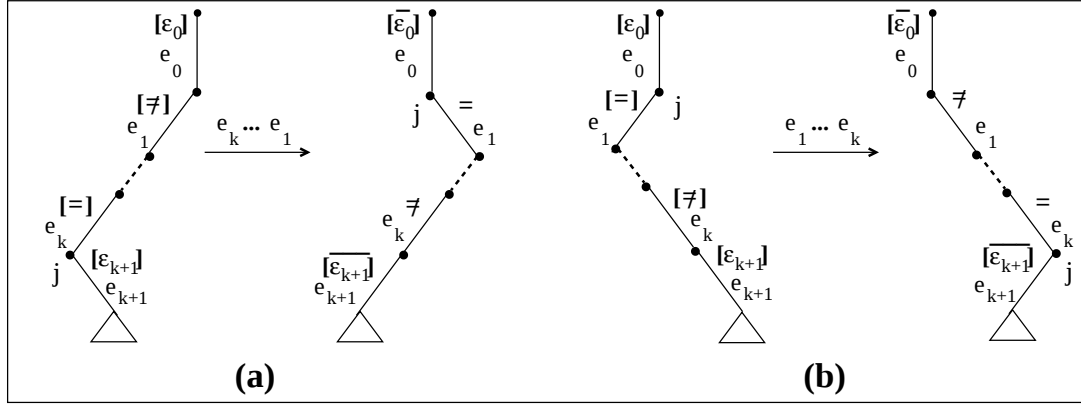


Figure 13: Multimove reassociations by sequential strategies: (a) *tdbus*, (b) *butds*.

Similar pictures represent multimove reassociations by a sequence that includes the backbone bottom edge, by dropping edge e_{k+1} and taking edge e_k to be the bottom edge. In this case, however, the sequence is allowed when e_k is either unconstrained or constrained as shown in the picture, but the latter condition is not warranted by the sequentiality hypothesis alone, rather also by that of validity of the bottom edge rule. This goes as follows. For the *tdbus* strategy, according to Figure 13(a) without edge e_{k+1} , leaf $j+1$ is the shared bottom leaf, hence validity of the bottom edge rule in the constrained source tree entails that the constraint on e_k is “=” both in the case of top edge covariant with the bottom one, since the tree pair is a majority pair, and in the opposite case, where the tree pair is a minority pair. Again, this also holds when $k=1$, as noted above. For the *butds* strategy, as in Figure 13(b) without edge e_{k+1} , leaf $j-1$ is the shared bottom leaf, and a similar reasoning leads to the same conclusion. Moreover, in both strategies, by chiral symmetry the validity of the conclusion does not depend on the displayed orientation of the reassociated edge sequences.

It remains to be shown that validity of the bottom edge rule is preserved as stated. It is convenient, to this purpose, to consider the combined outcome of the α reassociation sequence together with the subsequent, nonempty sequence of moveless edge contractions, which are top edge contractions in the *tdbus* strategy, bottom edge contractions in the other strategy. The two strategies are dealt with separately, as follows.

In the *tdbus* sequence of multimove reassociations, the first bottom-up multimove ends at the top edge of the source tree, since the given trees have contravariant top edges, and each subsequent bottom-up multimove ends at the top edge of the reduced source tree which results from the sequence of moveless top edge contractions that follows the previous multimove, where the reduced tree pair also has contravariant top edges. Let α be the bottom-up sequence of edges selected by the *tdbus* tactics at any step of the strategy, with moveless top edge contractions as already mentioned. Then, α always includes the top edge, as just argued. If it also includes the bottom edge, then the α multimove is the last one in the *tdbus* sequence, the source tree it acts upon is turn-free, and the resulting derivative has the same tree structure as the target, so the statement holds. If α does not include the bottom edge, but it includes the edge just above it, then the source tree has a single turn, placed between this edge and the bottom one, and the α

multimove changes both orientation and constraint of the bottom edge, but not the bottom leaf pair. The top edge of the reduced source tree which results from the subsequent moveless top edge contractions has the same orientation as the top edge of the source tree before the multimove, therefore the majority/minority status of the tree pair is not changed, thus validity of the bottom edge rule is preserved. Finally, consider the case when α includes neither the bottom edge nor that just above it. Then neither orientation nor constraint of the bottom edge change after the multimove. Depending on where do the subsequent moveless top edge contractions end up in the source path, the top edge orientation of the resulting reduced source tree derivative may have the same or opposite orientation as the top edge of the source tree before the multimove, viz. the majority/minority status of the tree pair may stay the same or may change, but neither orientation nor constraint of the bottom edge change, so validity of the bottom edge rule is preserved in both cases.

In the *butds* sequence of multimove reassociations one has an almost dual situation, but not exactly, so a detailed analysis is badly needed. The first top-down multimove ends at the bottom edge of the source tree, and moveless bottom edge contraction after the multimove makes the abstraction leaf to be a shared bottom leaf in the reduced tree pair. If also its sibling bottom leaf is shared, viz. the abstraction leaf is on the same side in both trees, then further moveless bottom edge contraction takes place, until the bottom leaf pairs in the two progressively reduced trees share one leaf only, that is the abstraction leaf. This happens as soon as the bottom edges in the two trees are contravariant, while sharing both bottom leaves; then, moveless bottom edge contraction still takes place, but the resulting abstraction leaves are on opposite sides in the so reduced tree pair, which thus only features the abstraction leaf as a shared bottom leaf. Let α be the top-down sequence of edges selected by the *butds* tactics at any step of the strategy, with moveless bottom edge contractions as already mentioned. Then, α always includes the bottom edge, as just argued. If it also includes the top edge, then the α multimove is the last one in the *butds* sequence, the target tree is turn-free, and the resulting derivative has the same tree structure as the target, so the statement holds. If α does not include the top edge, then contravariant top edges in the reduced tree pair have the same orientations as in the tree pair before the multimove, but orientation and/or constraint of the source bottom edge may differ, depending on where do the moveless bottom edge contractions after the multimove end up in the source path, and the majority/minority status of the tree pair may or may not change, depending on whether does or does not change the side of the shared bottom leaf (in both trees). Validity of the bottom edge rule in the reduced tree pair holds iff either the bottom edge is unconstrained, or else either none or (exactly) two of the following three properties hold: the abstraction leaf is on the same side as the top leaf, top edge and bottom edge are covariant, the bottom edge constraint is “ $\neq$ ”. This turns out to be the case indeed, when the bottom edge in the reduced source tree is constrained, as a consequence of the constraint sequentiality hypothesis (and already shown preservation), and because the last moveless bottom edge contraction, yielding the subject reduced tree pair, refers to contravariant bottom edges, as argued above, while the tree pair before the last moveless bottom edge contraction shares both bottom leaves. Figure 14 illustrates the case analysis that completes the proof, under the assumption that the source top leaf is on the right side. The assumption is harmless, since validity of the statement only depends on the relative orientation of top edge and bottom edge in the source tree, hence a similar

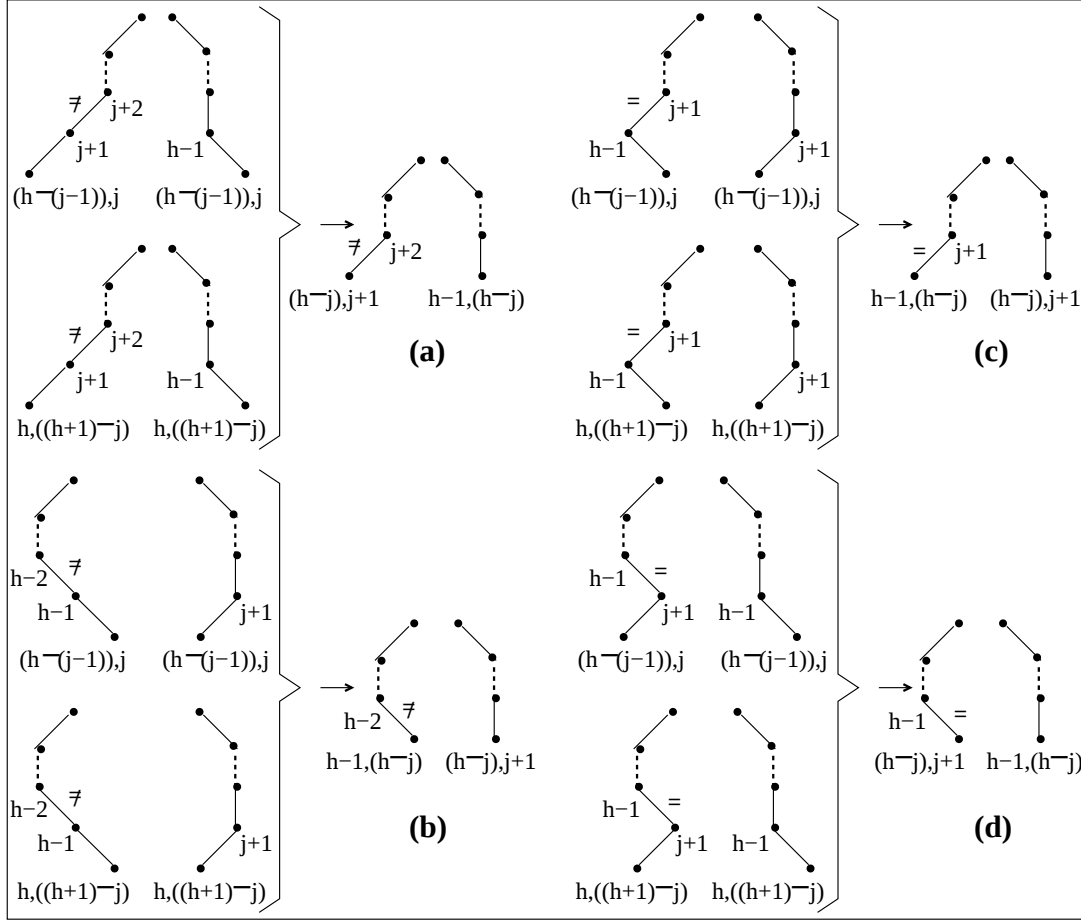


Figure 14: Bottom edge rule validity under the *butds* strategy.

case analysis applies under the opposite assumption, by chiral symmetry. As it may be seen in the picture, eight subcases are to be considered for the tree pair before the last moveless bottom edge contraction, but these can be paired, each pair of subcases yielding the same reduced tree pair. With the notation established in Section 2, $(h-j)$ denotes the abstraction leaf after the last moveless bottom edge contraction, with $h < j$. The bottom edge of the resulting reduced source tree is assumed to be constrained (otherwise the bottom edge rule is satisfied), conforming to the constraint sequentiality hypothesis, while neither any other constraint, nor the orientation of the target bottom edge after the last moveless bottom edge contraction, are displayed, since this information is irrelevant to validity of the bottom edge rule in the resulting, reduced tree pair. For the sake of simplicity, neither edge names nor the contraction type subscript (B, of course) are displayed in the picture. $\square$

And here is the main justification for the title of the present section.

Theorem 4. *Let (s, t) be a pair of path trees of equal size, with contravariant top edges, leaf i a bottom leaf in both trees, and a sequential constraint assignment in s . Then the pair is solvable, and both sequential strategies as per Def. 6, applied to s , find a solution for it, if either of the*

following conditions holds:

- (i) s and t share both bottom leaves;
- (ii) i is the only bottom leaf shared by s and t , and the constraint assignment in s satisfies the bottom edge rule.

Proof.

Under condition (i), the two path trees have the same number of left leaves (hence also the same of right leaves), thus no reassociation of the backbone bottom edge is needed. Multimove reassociations by either sequential strategy that exclude the bottom edge are allowed by the constraints in s , as shown in the first paragraph of the proof of lemma 5.2, and preserve both the number of left leaves and constraint assignment sequentiality, by lemma 5.1. Under condition (ii), the statement follows from lemma 5.2 by induction. $\square$

An easy corollary of the previous theorem is the solvability of every pair of unconstrained path trees that share (exactly) one bottom leaf, viz. the exact translation of Proposition 8 of [3] in the present framework; this holds because the empty constraint assignment is sequential and, for path trees with contravariant top edges, it satisfies the bottom edge rule. More can be said about the class of path tree pairs that share either or both bottom leaves, however, as well as about the examined sequential strategies for them.

Consistently with the concept in [3], say that a level below the top one is a *decomposition level* for a pair of path trees if these have the same subsequence of leaves under that level. In particular, the backbone bottom level is a decomposition level iff the two path trees share both bottom leaves. By the following result, one gets an easy way to count the number of distinct sequential solutions found by either sequential strategy for any given path tree pair that shares at least one bottom leaf; the proof uncovers a very easy way to draw the partial constraint assignment that specifies the set of solutions found by either sequential strategy, with no need to perform any reassociation whatsoever.

Theorem 5. *Both sequential strategies, $tdbus$ and $butds$, are effective for the class of path tree pairs that share either or both bottom leaves. They yield the same partial constraint assignment for any given pair in the class, that thus specifies the same (nonempty) set of solutions. This set has cardinality 2^k , where k is the number of decomposition levels for the given path tree pair.*

Proof.

It is enough to show that 1) the partial constraint assignment found by any strategy for a path tree pair in the subject class, may leave unconstrained only those backbone edges which are located just above a decomposition level, and 2) either sequential strategy leaves all those edges unconstrained. These two statements together entail that the partial constraint assignments found by the two sequential strategies coincide, since either of them delivers a partial constraint assignment that is sequential and, in the case of single shared bottom leaf, that satisfies the bottom edge rule; such an assignment is then uniquely determined by these properties outside the set of unconstrained edges.

The truth of statement 1) is easy to see from the fact that an edge is left unconstrained in the partial constraint assignment delivered by a reassociation sequence iff it does not occur in that sequence; in such a case there is no flow of leaves across the level just below that edge, therefore the reassociation sequence is a successful one only if the subject level is a decomposition level.

Statement 2) may be proven by induction on the backbone length. The two sequential strategies coincide in the basis case, where the backbone is empty, and so is the constraint assignment, of course. For the inductive step, if the tree pair has no decomposition level, then both strategies deliver a partial constraint assignment where all edges are constrained, by statement 1), so assume the tree pair has at least one decomposition level. Then, by decomposition at this level, both component source trees have smaller backbones than the original, composite one, since the decomposition level lies below the top by definition, and the source backbone edge just above it is a terminal edge in the upper component tree, so it belongs to neither component source backbones. The induction hypothesis thus applies to both component tree pairs, and the set of edges of the original source backbone that are left unconstrained by either strategy is the union of those left unconstrained for the smaller component backbones, together with the edge just above the decomposition level. $\square$

Finally, we conjecture that both sequential strategies are not only effective but also complete for the class of path tree pairs that share either or both bottom leaves. A proof of this statement would amount to show that every solution must be a sequential constraint assignment, satisfying the bottom edge rule in the case of single shared bottom leaf, since no other edge may be left unconstrained by any solution strategy besides those so left by either sequential strategy, according to the two statements in the proof of Theorem 5.

6 A class of solvable pairs of weakly mutually crooked path trees

The results presented in Section 5 prove also useful to solve certain pairs of path trees that share no bottom leaf. The basic idea here is to attack the problem by a *two-phase strategy*, where the first phase is aimed at yielding a reduced path tree pair with a shared bottom leaf and a sequential constraint assignment that satisfies the bottom edge rule in the reduced source tree derivative. A successful outcome of this kind from the first phase, enables adoption of a sequential strategy in the second phase, or just direct inference of the remaining part of the final, partial constraint assignment, by the latest remark before Theorem 5.

Path tree pairs of present interest are indecomposable, weakly mutually crooked (*i.e.* share no triplication), have at least three segments each (thus not solvable by Theorems 11 and 12 of [3], which have essentially the same proofs in the present framework), with contravariant top segments of length at least 2 (thus not reducible by Prop. 2.1) and second segments from the top of length at most 2 (thus not reducible by Theorem 2), share no bottom leaf, and neither pair of bottom leaves is duplicated in the other tree. Within this class of path tree pairs, a subclass is isolated that proves solvable by a two-phase, deterministic strategy of the aforementioned kind. The hypotheses which characterize this class give it a fairly narrow outlook, but it may be interesting to anticipate that it even includes a subclass of mutually crooked path tree pairs (*i.e.* sharing no duplication).

Before embarking on the characterization of the subject class, it may be useful to give a flavour of the method by a contrived example, where a mutually crooked path tree pair is solved.

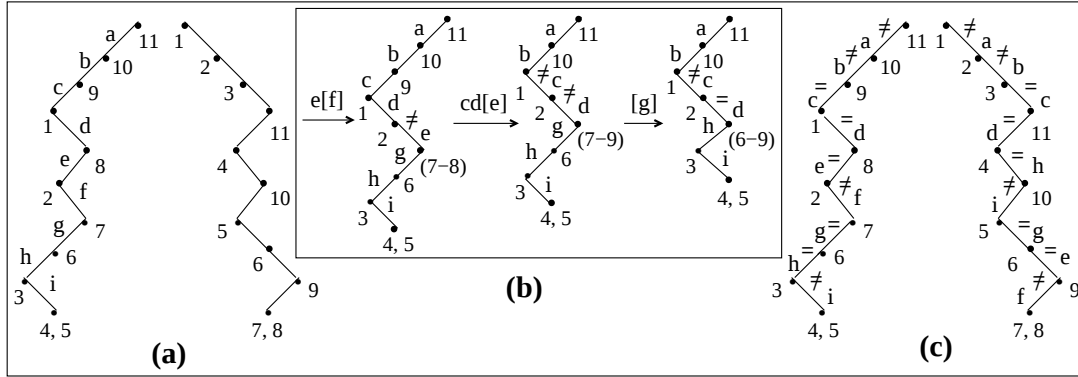


Figure 15: A solvable pair of mutually crooked path trees sharing no bottom leaf

The path tree pair in Figure 15(a) meets all aforementioned assumptions, and shares no duplication. The first phase sequence of reassociations and reductions, as displayed in Figure 15(b), is determined by the *butds* strategy, in this example. In the general case of forthcoming interest, the first phase strategy will be BU, but with a deterministic tactics that selects either the top-down or the bottom-up sequence, throughout the whole phase, depending on which alternative hypotheses are met. Not every intermediate reduced source tree derivative gets a sequential constraint assignment, since there is no shared bottom leaf in the given pair, but the final outcome of the displayed phase does so, it satisfies the bottom edge rule (as the bottom edge is unconstrained, in this case), and it shares a bottom leaf, viz. leaf 5, with the reduced target tree, hence the given pair is solvable. The constraint assignment inferred from the first phase outcome, by constraint sequentiality and bottom edge rule, is displayed in Figure 15(c). Note that only edges that are contracted in the first phase may eventually get constraints that violate constraint sequentiality.

As it will be shown, the following definition characterizes constraint assignments in the reduced source tree derivatives throughout the first phase, for path tree pairs of the intended, solvable class.

Definition 7. A (partial) constraint assignment to backbone edges of a path tree is **near sequential** if at most one of the constrained edges violates constraint sequentiality as per Def. 3.

The next definition generalizes the terminology introduced with Def. 4.

Definition 8. A pair of path trees of equal size, with contravariant top edges and that do not share both bottom leaves, is

- (i) a **majority pair** if either tree has more leaves on its top leaf side than the other tree on the same side;
- (ii) a **minority pair** if either tree has fewer leaves on its top leaf side than the other tree on the same side.

One of the reasons which justify the term “sequential” for the strategies introduced in Section 5 is that all tree derivatives they determine, for a path tree pair sharing a bottom leaf, are

path trees. This is no longer the case with first phase strategies considered here, not even in the case of top-down tactics (that is the *butds* strategy), because of the lack of a shared bottom leaf. However, the *reduced* tree derivatives they determine are path trees. In other words, each tree derivative resulting from a multimove reassociation has at most one edge that violates “pathness” of the internal edge backbone, but this undergoes bottom edge contraction in the subsequent reduction.

It is fairly easy to see that, whichever of the two sequential tactics, viz. bottom-up or top-down, is chosen for the first phase BU strategy, the reduced source tree derivative resulting from the first multimove reassociation and bottom edge contraction has a near sequential constraint assignment. Since neither of the target bottom leaves is a bottom leaf in the given source tree, the abstraction leaf out of the first reduction is not a bottom leaf in the reduced source tree derivative, and this has one edge violating sequentiality of the constraint assignment iff the multimove consists of at least two moves, *i.e.* the target bottom leaves are not duplicated in the source tree, which is one of the nonreducibility assumptions stated above. This one edge is the sibling of the terminal edge ending in the abstraction leaf if the tactics is bottom-up, otherwise it is the parent edge of that terminal edge.

Now, in order to motivate the choice of tactics under appropriate hypotheses, and to justify the hypotheses themselves, it is useful to consider conditions which preserve near sequentiality of constraint assignment throughout the first phase. It turns out that each of the two relative orientations of target tree segments, with respect to the top segment orientation, viz. covariant and contravariant, has a *default tactics*, which also depends on whether the path tree pair is a majority or minority one, that ensures the aforementioned preservation, as long as the target abstraction leaf moves up along a segment with that orientation.

More precisely, with the target abstraction leaf on a segment that is covariant with the target top segment in a majority pair, or contravariant with it in a minority pair, bottom-up tactics preserve near sequentiality of constraint assignment in reduced source tree derivatives, with possible violation of constraint sequentiality in the sibling edge of the source abstraction leaf, provided this condition holds when the target abstraction leaf is at the lower end of the segment, and as long as it keeps staying on that segment. Conversely, with the target abstraction leaf on a segment that is contravariant with the target top segment in a majority pair, or covariant with it in a minority pair, top-down tactics preserve near sequentiality of constraint assignment in reduced source tree derivatives, with possible violation of constraint sequentiality in the parent edge of the source abstraction leaf, provided this condition holds when the target abstraction leaf is at the lower end of the segment, and as long as it keeps staying on that segment.

Turns between adjacent target segments present the following problem. The condition stated above, that is preserved along the lower segment by its default tactics, yields possible violation of constraint sequentiality at an edge, either parent or sibling of the abstraction leaf, that is other than that where it would be preserved by the default tactics for the upper segment. In other words, the idea of switching tactics at backbone turns, to adopt default tactics for each segment, does not work, that is, it does not preserve near sequentiality of constraint assignments in reduced source tree derivatives.

However, it so happens that, if the sibling leaf of the lowest edge in the upper segment and its consecutive leaf at the next lower turn (at the bottom, iff the lower segment is the bottom

one in the target backbone) are duplicated in the source tree, then, after reduction along the lower target segment with default tactics, either tactics select the same, 1-edge move, and this reassociation “recovers” the constraint sequentiality violation in the reduced source tree derivative. Thereafter, if the upper segment has more than one edge, and the next pair of consecutive leaves are not duplicated in the source tree, then the nondefault tactics, that is the same tactics as for the lower segment, reintroduces constraint sequentiality violation in the reduced source tree derivative on the same sort of edge (parent or sibling of the abstraction leaf) where it is found before the aforementioned 1-edge move. Iterated occurrence of this condition, meaning that subsequent pairs of consecutive leaves on the upper target segment are alternately duplicated in the source tree, preserves near sequentiality until the segment end. A situation whereby a successful exit out of the first phase is achieved, is that where the last first phase reduction that yields a reduced source tree derivative sharing a bottom leaf with the reduced target, follows a 1-edge move that, together with the subsequent contraction of the reassociated edge “recover” the single violation of constraint sequentiality.

Once armed with the intuitions exposed above, the reader may hopefully get a grasp of the admittedly complex formulation of the hypotheses for the forthcoming theorem. Further justification is provided in its proof, as it were. For the sake of simplicity, we assume definite (contravariant) orientations of top segments in the given path trees, but this is no harm to generality, by reversibility of successful reassociation sequences solving the converse pair of path trees.

Theorem 6. *Let (s, t) be an indecomposable pair of path trees of equal size, say with n backbone edges, that share no bottom leaf, have at least three segments each, with contravariant top segments of length at least 2, segments just below the top ones of length at most 2, no shared triplication, and where neither pair of bottom leaves is duplicated in the other path tree. Assume leaf 1 is the top leaf in target tree t , thus leaf $n+2$ is the top leaf in source tree s . Let $h, h+1$ be the bottom leaves in s , and $i, i+1$ be the bottom leaves in t . Such a path tree pair is solvable if either occurrence of a source bottom leaf $l \in \{h, h+1\}$ in t , together with the given tree structures satisfy one out of the following four sets of alternative hypotheses:*

- (i) (s, t) is a majority pair, leaf l is not duplicated with leaf $l+1$ in target tree t , and
 - every segment contravariant with the top one in t , below the level of leaf l has even length, except for the top such segment, ending with right leaf $g > i+1$ at the level of (necessarily left) leaf l , that has odd length;
 - every pair of consecutive leaves $(i+2p-1, i+2p)$ is duplicated in source tree s , for $1 \leq p \leq \frac{g-i}{2}$;
- (ii) (s, t) is a minority pair, leaf l is not duplicated with leaf $l-1$ in target tree t , and
 - every segment covariant with the top one in t , below the level of leaf l has even length, except for the top such segment, ending with left leaf $g < i$ at the level of (necessarily right) leaf l , that has odd length;
 - every pair of consecutive leaves $(i-2p+1, i-2p+2)$ is duplicated in source tree s , for $1 \leq p \leq \frac{i-g+1}{2}$;
- (iii) (s, t) is a majority pair, leaf $l = h+1$ and it is duplicated with leaf $h+2$ in target tree t , and

- every segment covariant with the top one in t , between the bottom and the parent vertex of leaf l , with the top such segment cut at that vertex, has even length;
 - every pair of consecutive leaves $(i-2p+1, i-2p+2)$ is duplicated in source tree s , for $1 \leq p \leq \frac{i-l}{2}$;
- (iv) (s, t) is a minority pair, leaf $l = h$ and it is duplicated with leaf $h-1$ in target tree t , and
- every segment contravariant with the top one in t , between the bottom the parent vertex of leaf l , with the top such segment cut at that vertex, has even length;
 - every pair of consecutive leaves $(i+2p-1, i+2p)$ is duplicated in source tree s , for $1 \leq p \leq \frac{l-i-1}{2}$.

Proof.

The claim is that a first phase BU strategy, with bottom-up tactics under hypotheses (i) or (ii), top-down tactics under hypotheses (iii) or (iv), yields a reduced source tree derivative that only shares leaf l as a bottom leaf with the reduced target tree, and has a sequential constraint assignment that satisfies the bottom edge rule. Validity of the claim then entails solvability of (s, t) , thanks to Theorem 4. Figure 16 depicts the situation near (the parent vertex of) leaf l in target tree t , in the four cases considered in the hypotheses.

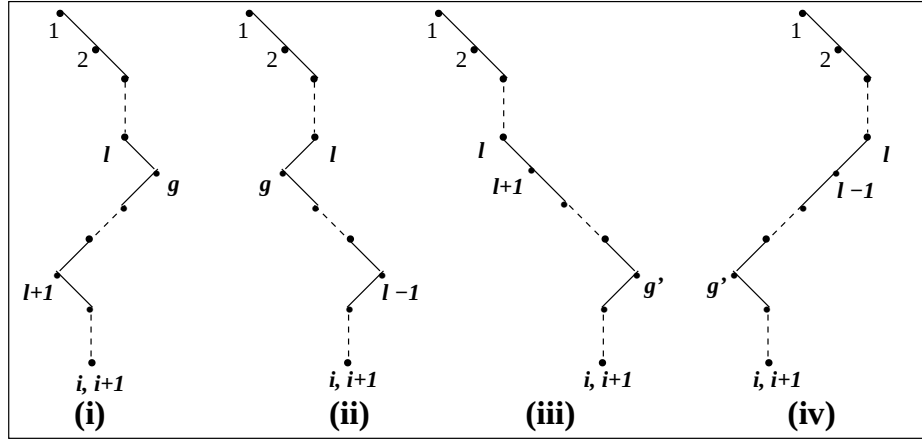


Figure 16: Target tree case analysis for first phase exit with shared bottom leaf l

The choice of tactics in the specified cases only depends on whether or not is leaf l duplicated with its consecutive leaf below it in target tree t . This is motivated as follows. For the claim to hold, by the duplication assumptions on source tree s , the target leaf at the level just below that of l must get joined with the abstraction leaf, in the reduced source tree derivative, by a 1-edge last move of the first phase that recovers the constraint sequentiality violation. Therefore, the nondefault tactics are to be adopted for the segment where its sibling lives. This, together with the definition of default tactics and a quick inspection at Figure 16 entail bottom-up tactics when leaf l is not duplicated with its consecutive leaf below it, as it happens under hypotheses (i) or (ii), top-down tactics otherwise. This also explains the parity requirements on the length of segments where these tactics are the nondefault ones.

The reader may have noticed that the stated hypotheses require duplication of alternating pairs

of consecutive leaves, within a specified interval, in source tree s , but do not require the other pairs of consecutive leaves in the same interval not to be duplicated. This is not necessary, by the following argument. If such a leaf pair has sibling edges in a segment of length greater than 2 in target tree t , then it is part of a triplication in this tree, that would be a shared triplication if that pair were also duplicated in source tree s , against the assumption of absence of shared triplications in the given path tree pair (s, t) . On the contrary, when the subject leaf pair has sibling edges in a segment of length 2 in target tree t , then it is not part of a triplication in this tree, so it may well be part of a triplication in source tree s . However, the net effect of this in the reassociation with reduction process, is the preservation of the single constraint sequentiality violation on the same edge where it occurs before the two 1-edge moves which join the leaves out of the subject leaf pair with the abstraction leaf.

Let the *proper tactics* be bottom-up in cases (i) and (ii), top-down in the other two cases. Let the *first-phase final leaf* f be target leaf g in cases (i) and (ii), $l+1$ in case (iii), $l-1$ in case (iv). The proof is carried out by showing that

- (I) every first phase BU multimove reassociation determined by the proper tactics is allowed, and yields a reduced source tree derivative with near sequential constraint assignment;
- (II) the reduced tree pair obtained by contraction of a bottom edge that has the first-phase final leaf f as a bottom leaf, has leaf l as the only shared bottom leaf, and its reduced source tree derivative has a sequential constraint assignment that satisfies the bottom edge rule.

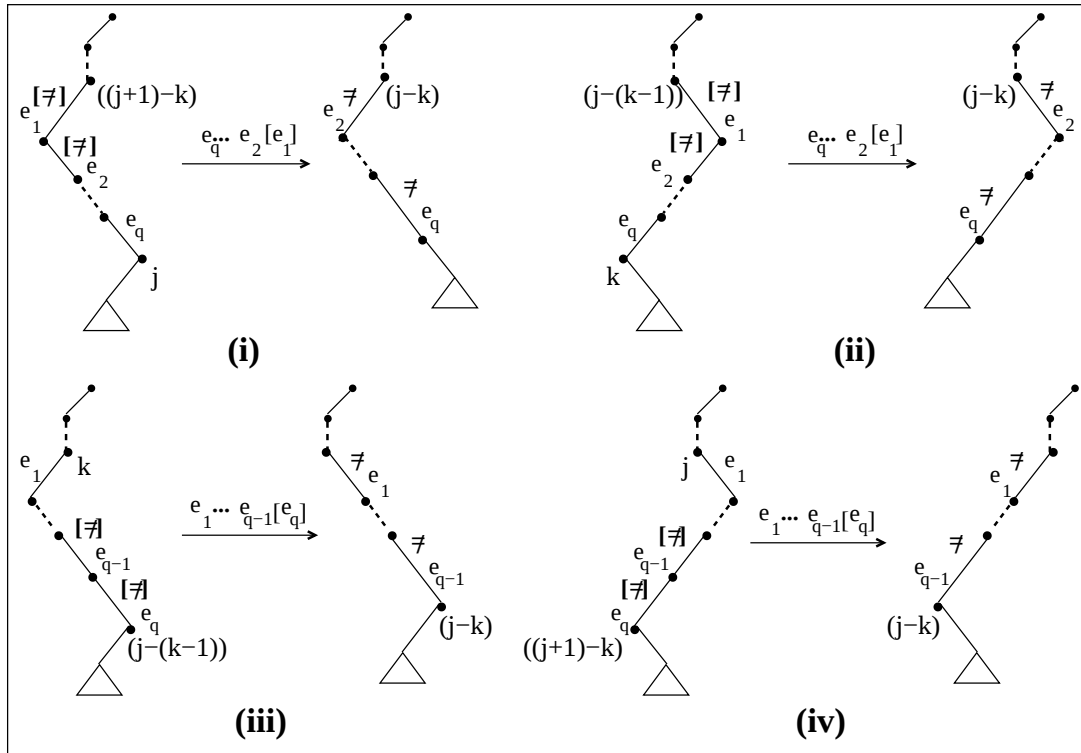


Figure 17: First phase source tree reductions with default tactics: (i,ii) bottom-up, (iii,iv) top-down.

Figure 17 illustrates multimove reassociations with subsequent bottom edge contraction in the source tree derivative, when the proper tactics is the default one on the target segment where the corresponding contraction takes place. With reference to Figure 16, let $l' = l$ in cases (i,ii), $l' = g'+1$ in case (iii), $l' = g'-1$ in case (iv). Clearly, one has $l' < j \leq i < k < f$ in cases (i,iv), $f < j \leq i < k < l'$ in cases (ii,iii). When $j = i$ and $k = i+1$ in Figure 17, then all edges are unconstrained in source tree s before the (first) multimove reassociation, and the generic abstraction leaf $((j+1)-k)$ in Figures 17(i,iv) is actually leaf $i+1$, while $(j-(k-1))$ in Figures 17(ii,iii) is actually leaf i ; $q \geq 2$, in this case because the target bottom leaves are not duplicated in source tree s by hypothesis.

However, $q \geq 2$ for every multimove reassociation that has the proper tactics as the default one on the target segment, not just for the first one, and regardless of whether leaves j and $j+1$ in cases (i,iv), or leaves $k-1$ and k in cases (ii,iii), are duplicated in source tree s . This is shown as follows. From Figure 17 it is apparent that, when the multimove reassociation is not the first one, in all cases the abstraction leaf gets joined to the other target bottom leaf in the reduced source tree derivative by moving the latter leaf toward the former. This fact has three consequences. First, even if the moved leaf is duplicated with its consecutive leaf in the opposite vertical direction with respect to the abstraction leaf, the new abstraction leaf will not be duplicated with it, which fact proves $q \geq 2$ as stated. Second, constrained edges after the multimove reassociation and subsequent bottom edge contraction, all have “ $\neq$ ” constraints and are on the same vertical side, *i.e.* above or below, with respect to the abstraction leaf; only one of them violates constraint sequentiality, and that is the sibling edge of the abstraction leaf with bottom-up default tactics, whereas it is the parent edge of (the parent vertex of) the abstraction leaf with top-down default tactics. Third, on condition that the moved leaf is not a bottom leaf with bottom-up default tactics, nor the top leaf with top-down default tactics, then the first edge in the multimove edge sequence is unconstrained before the multimove; this fact, together with the previous one entail that the reassociation sequence is allowed, under the stated condition. Actually, this also holds when the position of the leaf to be moved reaches the bottom, with bottom-up default tactics, or the top, with top-down default tactics, for the first time.

With top-down default tactics, the lower bound on j in case (iv), and the upper bound on k in case (iii) entail multimove reassociations with default tactics are allowed, provided the first of them for the target segment is allowed. With bottom-up default tactics, the lower bound on j in case (i), and the upper bound on k in case (ii) entail multimove reassociations with default tactics are allowed, provided the first of them for the target segment is allowed. Now, the first multimove reassociation for a target segment with default proper tactics is obviously allowed when the segment is the bottom one, since all source edges are initially unconstrained, but for a nonbottom such segment, the first multimove reassociation is allowed if near sequentiality of constraint assignment holds after the last multimove reassociation for the previous target segment, with nondefault proper tactics, and provided the constraint assignment violates sequentiality on the sibling edge of the abstraction leaf, with bottom-up tactics, or on its parent edge, with top-down tactics. The proof of statement (I) is thus completed by showing that multimove reassociations for target segments with nondefault proper tactics, except for the top one, reaching the level of leaf l in cases (i) or (ii), or cut at its parent vertex in cases (iii) and (iv), do deliver a near sequential constraint assignment that violates constraint sequentiality on the

required edge, when the target abstraction leaf reaches the next turn on the target backbone, *i.e.* as outcome of the last bottom edge contraction for the subject target segment.

Figure 18 illustrates multimove reassociations with subsequent bottom edge contraction in the source tree derivative, when the proper tactics is not the default one on the target segment where the corresponding contraction takes place, for consecutive leaf pairs that are not duplicated in source tree s . Actually, the first reassociation on such a target segment involves, by hypothesis, a pair of consecutive leaves that are not duplicated in target tree t , but are so in source tree s . The 1-edge move on the reduced source tree reassociates and contracts the parent edge of (the parent vertex of) the abstraction leaf with bottom-up proper tactics, the sibling edge of the abstraction leaf with top-down tactics. Therefore the “ $\neq$ ” constraint changes to “ $=$ ” on the edge incident with (the parent vertex of) the abstraction leaf that violated constraint sequentiality. This is the situation depicted in Figure 18 before multimove reassociation, where one sees that reduced source tree derivatives in all cases feature again a single violation of constraint sequentiality, on the same sort of edge (sibling or parent of the abstraction leaf) where it was found before the two-multimove sequence. Note that this also holds when the displayed reassociation is a 1-edge one, *viz.* one may have $q = 1$ in Figure 18, when the target segment has length 2, as

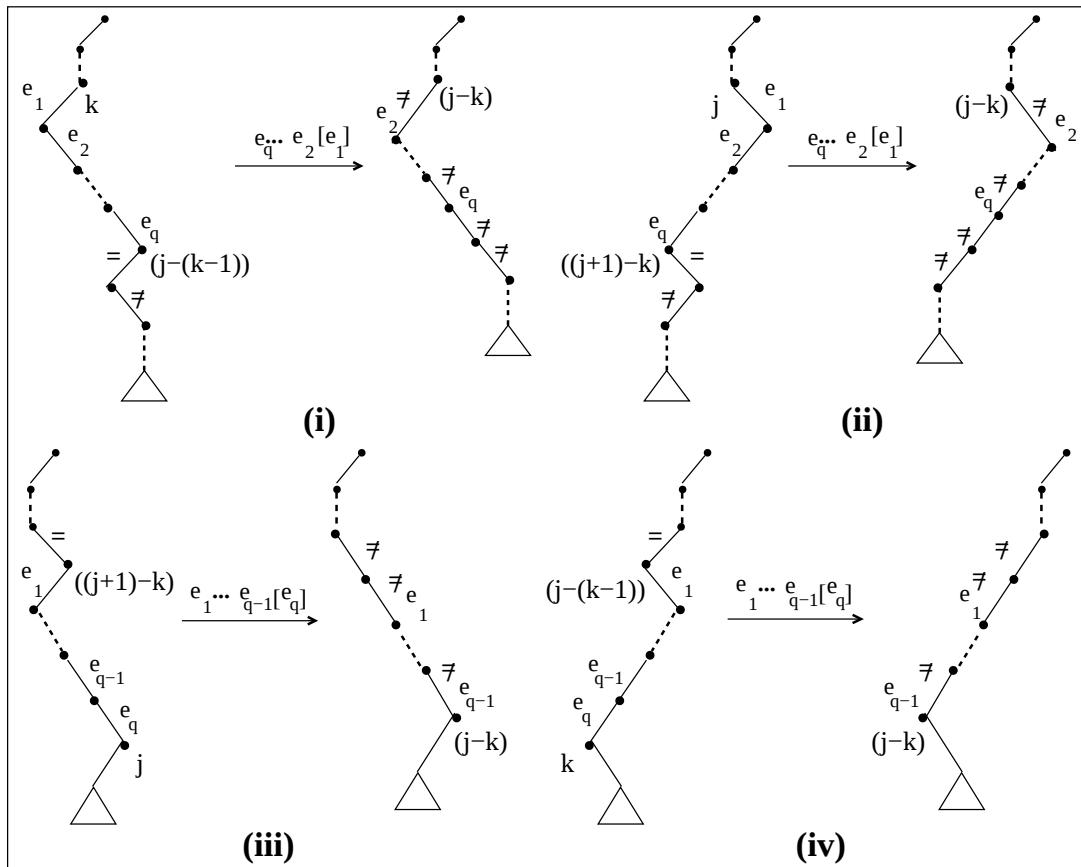


Figure 18: First phase source tree reductions with nondefault tactics: (i,ii) bottom-up, (iii,iv) top-down.

earlier argued. Finally, note that when proper tactics is not the default one, reassociation with subsequent contraction moves the abstraction leaf toward the leaf with which it gets joined, and all reassociated edges are unconstrained before the multimove, which is thus always allowed.

Statement (I) is thus proven, by the parity hypothesis about the length of target segments with nondefault proper tactics, together with the hypotheses about alternating duplications in source tree s . However, also part of statement (II) immediately follows, since the number of reassociations on the top segment with nondefault tactics is odd by hypothesis (this also holds in cases (iii,iv), since the sibling edge of target leaf l is not contracted), which entails sequentiality of the constraint assignment in the reduced source tree that has leaf l as a shared bottom leaf with the reduced target tree.

It remains to be seen that leaf l is the only shared bottom leaf, and that the constraint assignment satisfies the bottom edge rule. The former statement holds because leaf l is on opposite sides in the reduced tree pair. This is immediate in cases (iii) and (iv), see Figure 16, since $l = h+1$ by hypothesis in case (iii) is a right leaf in the source tree, while $l = h$ by hypothesis in case (iv) is a left leaf in the source tree. The same holds in case (i) when $l = h+1$ and in case (ii) when $l = h$. In all these cases, l is on the same side in the reduced source tree as it is in source tree s , since the bottom edge incident with it is not reassociated. This also entails that the constraint assignment satisfies the bottom edge rule, in these cases, because the bottom edge is unconstrained.

The situation is a little different in case (i) when $l = h$, and in case (ii) when $l = h+1$. However, the only differences are that leaf l in the reduced source tree is on the opposite side to that where it is in source tree s , since the bottom edge incident with l is reassociated once in these cases, but again Figure 16 tells that leaf l in target tree t and in source tree s are on the same side in these cases, therefore the shared bottom leaf l is on opposite sides in the reduced tree pair. Finally, Figure 17(i) with $j = l+1 = h+1$, and Figure 17(ii) with $k = l-1 = h$, tell that, in these last two cases, when the bottom leaf l changes side in the reduced source tree derivative, the bottom edge incident with it gets a “ $\neq$ ” constraint and remains contravariant with the top edge in case (i), where the tree pair is a majority pair, covariant with it in case (ii), where the tree pair is a minority pair, therefore the bottom edge rule is satisfied in both cases at this stage, and it remains satisfied throughout the rest of the first phase, since by Figure 16 it is apparent that the remaining multimoves relate to the top segment reaching the level of l in the target tree, with nondefault bottom-up tactics, and therefore they will move up the abstraction leaf even further, hence never include the bottom edge that is incident with l in the reduced source tree derivatives. $\square$

The reader may like to verify that the path tree pair displayed in Figure 15(a) satisfies case (iii) of the hypotheses stated in Theorem 6; this justifies the choice of top-down tactics for the first phase. A necessary condition for a path tree pair, that satisfies the hypotheses of Theorem 6, to be mutually crooked, like the exemplified one, is that all segments which are required to have even length by those hypotheses, actually have length 2.

7 Conclusions

The reassociative approach toward an alternative proof of the 4CT has been explored in this paper by recasting concepts and results from recent work by Cooper, Rowland and Zeilberger, geared toward a language theoretic proof. The dynamical nature of the reassociative approach, whereby solutions are delivered by reassociation sequences, has enabled us both to extend the aforementioned results, viz. to characterize new classes of reducible or solvable binary tree pairs, and to open new questions for further investigation. Some of the new classes have fairly wide extent and are obtained by simple proofs, notably Prop. 2.1, Theorem 2, and Prop. 4.1. Perhaps amazingly, some others are of narrower extent and required more substantial deductive effort, such as Theorem 4 (with the bulk of its proof confined to Lemma 5.2) and especially Theorem 6. Theorem 5 showcases how, also in the reassociative approach, may one count the number of solutions for any given problem instance in a certain class—a valuable feature of the aforementioned language theoretic approach. However, so far we may only claim having found a lower bound to the count for path tree pairs that share a bottom leaf. To turn this into a definite count, a positive answer is needed to the completeness question about the sequential strategies employed for the subject class.

If one evaluates the strength of a theorem as inversely proportional to the number of its hypotheses, and its cost by the size of its proof, then surely Theorem 6 scores low on the benefit/cost ratio, to the point of making one wonder whether the outcome deserves the effort. We believe it does, for a methodological reason which leads to the final subject of these concluding notes, relating to the new questions which arise from the present work, and the further research directions they open.

The main open questions relate to the nondeterministic strategies and tactics introduced in Section 4. In summary, they are: effectiveness of the BU and/or TD strategy, completeness of the BU|TD strategy. While these questions are independent of the validity of the EK Conjecture, settling them in the positive (the answer we expect in all three cases), would yield a solid foundation of solution search algorithms.

Further work of ours is under way, aimed at a fast decision procedure to characterize the subset of a given set of permutations of a set of edges, that comprises those edge reassociation sequences that are allowed by any given (partial) constraint assignment to edges in the given set. This, together with the result provided in Theorem 3 for the TD strategy and with a decision procedure for the map kernel equivalence on reassociation sequences, would deliver useful tools to develop effective parallel implementations of algorithms for nondeterministic strategies. A positive answer to the BU|TD completeness question would then provide an effective means to count the number of solutions to any given problem instance.

As testified to by the work presented in Section 5, specific classes of binary tree pairs are solvable by deterministic specializations of the aforementioned strategies. This actually holds for specific classes of path tree pairs, so far. In this respect, we have shown both the effectiveness and the equivalence of sequential strategies for path tree pairs sharing a bottom leaf, where the only residual open question relates to their completeness, as recalled above (and where, again, a positive answer is our expectation). The work presented in Section 6 is to be viewed as a first attempt to find other effective, deterministic strategies that could expand the class of

solvable tree pairs. The near-sequential weakening of the previously introduced concept of sequential constraint assignment, proves useful to identify deterministic strategies that solve a fairly narrow class of path tree pairs, as that which satisfies the hypotheses of Theorem 6. Yet this should be viewed as a first sortie into a wide open exploration field, not necessarily limited to path tree pairs. Other deterministic blends of strategies and tactics might prove capable of solving further classes thereof, especially blends where the choice of tactics is not fixed for each pair in the class, but rather takes a dynamic character, viz. it may change throughout the reassociation sequence.

References

- [1] K. Appel and W. Haken, Every planar map is four colorable. Part I. Discharging, *Illinois J. Math.* **21** (1977), 429–490.
- [2] K. Appel, W. Haken and J. Koch, Every planar map is four colorable. Part II. Reducibility, *Illinois J. Math.* **21** (1977), 491–567.
- [3] B.J. Cooper, E. Rowland, D. Zeilberger, Toward a Language Theoretic Proof of the Four Color Theorem, June 7, 2010. To appear in *Adv. Appl. Math.* <http://www.math.rutgers.edu/~zeilberg/mamarim/mamarimhtml/4ct.html>
- [4] S. Eliahou, Signed Diagonal Flips and the Four Color Theorem, *European J. Combin.* **20** (1999) 641–647. <http://dx.doi.org/10.1006/eujc.1999.0312>
- [5] L.H. Kauffman, Map coloring and the vector cross product, *J. Combin. Theory, Series B* **48** (1990) 145–154.
- [6] L.H. Kauffman, Reformulating the map color theorem, *Discrete Math.* **302** (2005) 145–172.
- [7] S.I. Kryuchkov, The four color theorem and trees, Preprint, I.V. Kurchatov Institute of Atomic Energy IAE-5537/1, Moscow, 1992.
- [8] R. Penrose, Applications of negative dimensional tensors, in: D.J.A. Welsh (Ed.), *Combinatorial Mathematics and Its Applications*, Academic Press, New York, 1971.
- [9] N. Robertson, D.P. Sanders, P.D. Seymour and R. Thomas, The four colour theorem, *J. Combin. Theory Ser. B* **70** (1997), 2–44.
- [10] P.G. Tait, Note on a theorem in the geometry of position, *Trans. Roy. Soc. Edinburgh* **29** (1880), 657–660, printed in *Scientific Papers* **1**, 408–411.
- [11] P.G. Tait, On Listing’s topology, *Phil. Mag. V. Ser.* **17** (1884), 30–46, printed in *Scientific Papers* **2**, 85–98.
- [12] D. Zeilberger, Enumerative and algebraic combinatorics, In: T. Gowers, (Ed.), *The Princeton companion to mathematics*, Princeton University Press, USA, 2008, pp. 550–561. <http://www.math.rutgers.edu/~zeilberg/mamarim/mamarimhtml/enu.html>