On non-bijective tree-questionable-width

Laurent Lyaudet*

June 12, 2025

Abstract

In this note, we show that there are at least 4 types of non-bijective tree-questionable-width. The study of these 4 types comes from an example of tree-questionable-decomposition of unbounded degree and depth 2 for all binary structure. We hierarchise these types and give links with bijective tree-questionable-width.

Initial version: 2025/06/12 Current version: 2025/06/12

Keywords: first difference principle, binary structures, 2-structures, graphs, tree-questionable-width, hierarchical decompositions

1 Introduction

(Balanced) (bijective or non-bijective) tree-questionable-width was introduced in Lyaudet (2019). We give new non-bijective variants, study the links with the maximum degree of the decomposition tree, and the links with the bijective case. What follows comes from the detailed study of:

Example. Let S be a binary structure of cardinality n; in a non-bijective decomposition, we can repeat each vertex ($n-1$ times in the finite case), to make "cherries" for each adjacency between two vertices with two leaves, one for each vertex, plus an internal node that links the two leaves/vertices with the correct adjacency type. Then, we link all these "cherries" to a unique root node that collects all the adjacencies.

This example works or not depending on the meaning we give to first difference principle on a tree. Our example dates from the end of 2019 or the beginning of 2020, but the clarification and what follows is more recent; because in a first time, we just had the reflex to focus on the bijective case with bounded degree, as we did with the binary trees in Lyaudet (2022), Lyaudet (2025b) and Lyaudet (2025a).

2 Definitions

Let V be a set of vertices, a (V,k)-*mapping-run* is a sequence of mappings from V to vertices of binary structures of cardinality at most k (the image binary structure is fixed per mapping).

*`https://lyaudet.eu/laurent/`, laurent.lyaudet@gmail.com

It is possible in tree-questionable-decompositions to have nodes of the decompositions that are mapped to an empty mapping-run (of length 0), but the writing gets more complicated. In this article, we will consider that there is always at least one identity mapping on each node. This identity mapping sends each vertex to a unique vertex of a binary structure of cardinality 1; it simplifies the notion of first difference and will help the reader to form her intuition.

We start by technical details on trees intersections.

Definition 2.1 (Common tree, Junction point, Extended common tree)**.** *Let T be a rooted tree of root r, T_1 be a rooted sub-tree of T of same root r, and T_2 be a rooted sub-tree of T of same root r.*

We call common tree *of T_1 and T_2, denoted by $\mathrm{CT}(T_1, T_2)$, the tree induced by the intersection of nodes of T_1 and T_2.*

We say that a node of $\mathrm{CT}(T_1, T_2)$ is a junction point *of $\mathrm{CT}(T_1, T_2)$ if this node has a son in T_1 or a son in T_2 that is not in $\mathrm{CT}(T_1, T_2)$.*

We call extended common tree *of T_1 and T_2, denoted by $\mathrm{ECT}(T_1, T_2)$, the tree obtained from $\mathrm{CT}(T_1, T_2)$ by adding to each junction point of $\mathrm{CT}(T_1, T_2)$ an adjacent leaf.*

In a similar way, we define a leaf-root path as a path from a leaf of $\mathrm{CT}(T_1, T_2)$ to the root of $\mathrm{CT}(T_1, T_2)$ for some pair $\{T_1, T_2\}$; and we define an extended leaf-root path as a path from a leaf of $\mathrm{ECT}(T_1, T_2)$ to the root of $\mathrm{ECT}(T_1, T_2)$ for some pair $\{T_1, T_2\}$.

Definition 2.2 (Tree-questionable-decomposition)**.** *Let S be a binary structure. A* (k, α, β)-tree-questionable-decomposition *of S is a triplet (T, ll, nl) (T as tree, ll as leaf labels et nl as node labels) such that:*

- *T is a rooted tree;*
- *the function ll is a surjective mapping (a bijection in the bijective case) from the leaves of T to the vertices of S;*
- *hence, each internal node node is associated to the subset of vertices of S union of the values $ll(l)$ for all leaves l under the node node; it defines $ll(\mathrm{node})$;*
- *nl is a mapping with domain the internal nodes of T, such that $nl(\mathrm{node})$ is an $(ll(\mathrm{node}), k)$-mapping-run;*
- *as a consequence, each vertex x of S corresponds to a sub-tree (that is a path in the bijective case) of T, we denote by T_x this sub-tree, resp. path;*
- *for every pair of vertices $\{x, y\}$, we will look at the first difference principle on $\mathrm{CT}(T_x, T_y)$ or $\mathrm{ECT}(T_x, T_y)$;*
- *let C be an, extended or not, leaf-root path relatively to $\{x, y\}$; we can define the $(\{x, y\}, k)$-mapping-run obtained by concatenating the $(ll(\mathrm{node}), k)$-mapping-runs restricted to $\{x, y\}$ when we take the nodes node of C from the leaf to the root; if a first difference between the images of x and of y exists in this mapping-run, it is the* question *of C; when the path C has a question, it must correspond to two vertices of same adjacency type as between x and y to be valid;*

- *for the tree-questionable-decompositions of type 1, we ask that all extended leaf-root paths have a valid question;*
- *for the tree-questionable-decompositions of type 2, we ask that all leaf-root paths have a valid question;*
- *for the tree-questionable-decompositions of type 3, we ask that at least one leaf-root path has a question, and that all the questions of leaf-root paths are valid;*
- *for the tree-questionable-decompositions of type 4, we ask that at least one extended leaf-root path has a question, and that all the questions of extended leaf-root paths are valid;*
- α *is the depth of the tree* T*;*
- β *is the depth of the extended tree* T' *obtained by replacing each internal node by a path of nodes (one for each mapping of the mapping-run associated to the original node).*

k *is called the width of the decomposition;* α *is called the structural depth of the decomposition;* β *is called the logical depth of the decomposition. In the finite case, the degree of the decomposition is the maximum degree of nodes of* T*.*

The original definition given in Lyaudet (2019) corresponds to tree-questionable-decompositions of type 2.

The introductory example can be reworded:

Lemma 2.3. *Let* S *be an (infinite) binary structure, it has a* $(2, 2, 2)$*-tree-questionable-decomposition of types 2, 3 and 4 of unbounded degree.*

3 Comparisons

Lemma 3.1. *A tree-questionable-decomposition of type 1 is also of type 2, type 3 and type 4.*

Lemma 3.2. *A tree-questionable-decomposition of type 2 is also of type 3.*

Lemma 3.3. *A tree-questionable-decomposition of type 4 is also of type 3.*

Proof:

(i) If all the questions of the extended leaf-root paths are valid, a fortiori, all the questions of the leaf-root paths are valid.

(ii) Moreover, there is always at least one leaf-root path that contains a given extended leaf-root path except for its leaf; hence, it contains its (valid) question.

(iii) Thus the existence of an extended leaf-root path with a valid question implies the existence of a leaf-root path with a valid question. Either this is the same question and we conclude by (ii), or this is a difference/question before in the mapping-run and we conclude by (i).

■

All the other inclusions are false; here are counter-examples.

The example decomposition that motivated this article is of types 2, 3 and 4 but not 1.

The following decomposition is of types 2 and 3 but neither 4 nor 1. Consider a binary structure with 3 vertices a, b and c, such that a is adjacent to c, and otherwise everything is non-adjacent. We make a cherry with two leaves for a and b; and we fix the non-adjacency between a and b. We link the root of this cherry to the true root of the decomposition. This true root has also a repeated leaf son a and a leaf son c. The mapping-run of the true root fixes a adjacent to b and c (but this is covered by the cherry between a and b for the types 2 and 3), then fixes b (and a) non-adjacent to c.

The following decomposition is of types 3 and 4 but neither 2 nor 1. Consider a binary structure with two vertices a, b, such that a is adjacent to b. We make a cherry with two leaves for a and b; and we fix the adjacency between a and b. We make a second cherry with two leaves for a and b; and we fix nothing (identity mapping). We link the roots of the two cherries to the true root of the decomposition that has also just an identity mapping.

4 Results on the degree

Lemma 4.1. *If the tree-questionable-decomposition is bijective, it is at the same time of type 1, type 2, type 3, and type 4.*

Proof:

> $\mathrm{CT}(T_x, T_y)$ is a path. $\mathrm{ECT}(T_x, T_y)$ is a path. The two mapping-runs are almost identical modulo an identity mapping as a prefix. ■

Lemma 4.2. *Let S be a finite binary structure. A (k, α, β)-tree-questionable-decomposition of type 1, resp. of type 4, with maximum degree Δ can be converted into a $(k, \alpha \times \lceil \lg(\Delta) \rceil, \beta \times \lceil \lg(\Delta) \rceil)$-tree-questionable-decomposition of type 1, resp. of type 4, with degree two.*

Proof:

> We can separate each node of degree more than two into a cherry in a balanced way. We put an identity mapping on the two leaves of the cherry. The questions will thus not be on these leaves. We do not create any question, but we create/move junction points toward the leaves if and only if the separated node was a junction point. Hence, the new extended leaf-root paths have a question if and only if there was a question starting from the original junction point. Thus, the condition to always have a question on each extended leaf-root path and that it is valid is fulfilled for type 1. And the condition to have a question on at least one extended leaf-root path and that it is valid is fulfilled for type 4. ■

Corollary 4.3. *Let S be a finite binary structure of cardinality n, it has a $(2, \lceil 2 \times \lg(n) \rceil + 1, \lceil 2 \times \lg(n) \rceil + 1)$-tree-questionable-decomposition of types 3 and 4, with degree two.*

Corollary 4.4. *Modulo a logarithmic factor on the depth, a bijective tree-questionable-decomposition of a finite binary structure can be converted into a bijective tree-questionable-decomposition, (binary/)with degree two.*

Thus, the degree doesn't matter much when we search for bijective tree-questionable-decompositions that are balanced in a polylogarithmic way.

5 Pruning

We start by stating something obvious:

Lemma 5.1. *If a node $node$ of a tree-questionable-decomposition of type 1 or 2 has 2 sons $node_1$ and $node_2$, such that $ll(node_1) \subseteq ll(node_2)$, then we can delete all the sub-tree rooted at $node_1$.*

Lemma 5.2. *If a tree-questionable-decomposition of type 1 has two leaves mapped to the same vertex, we can delete one and maybe merge its parent node $node$ with its second son, if $node$ ends up being of degree 1.*

Proof:

> A leaf pruning doesn't create any junction point. And since the leaf is redundant, there is still at least one junction point for each pair of vertices. Since with type 1, every junction point generates a valid question, the property to still have a valid question is fulfilled. ■

Corollary 5.3. *The non-bijective tree-questionable-width of type 1 is equal to the bijective tree-questionable-width.*

6 Conclusion

After this study, only the balanced non-bijective tree-questionable-width of type 2 with bounded degree is still completely open, in the non-bijective case. The types 3 and 4 are too powerful and decomposes everything, no matter the additional constraints. There also remains the more classical open problem of the bijective tree-questionable-width. A lot of things are configurable with the tree-questionable-width, particularly in the non-bijective case. We could complexify the problem further: by looking at various bounds on the degree (logarithmic bounds for example); or by looking at bounds on the "surjectivity", like asking that a vertex of a finite binary structure is the image of at most a logarithmic number of leaves.

Thanks God! Thanks Father! Thanks Jesus! Thanks Holy-Spirit!

References

L. Lyaudet. On finite width questionable representations of orders. *CoRR*, abs/1903.02028, 2019. URL http://arxiv.org/abs/1903.02028.

L. Lyaudet. Diviser n'est pas régner ? *preprint*, abs/202205, 2022. URL https://lyaudet.eu/laurent/Publi/Journaux/LL2022DiviserNestPasRegner/LL2022DiviserNestPasRegner_v6.pdf.

L. Lyaudet. On tree-width and tree-questionable-width. *preprint*, abs/202506-2, 2025a. URL https://lyaudet.eu/laurent/Publi/Journaux/LL2025TreewidthTreequestionablewidth/LL2025TreewidthTreequestionablewidth_en_v1.pdf.

L. Lyaudet. Sur la largeur arborescente et la largeur arborescente questionnable. *preprint*, abs/202506-1, 2025b. URL https://lyaudet.eu/laurent/Publi/Journaux/LL2025TreewidthTreequestionablewidth/LL2025TreewidthTreequestionablewidth_fr_v1.pdf.