



HAL
open science

Normalization properties of a computation process on colored finite trees

Karim Nour, Ayman El Zein

► **To cite this version:**

Karim Nour, Ayman El Zein. Normalization properties of a computation process on colored finite trees. 2024. hal-04765327

HAL Id: hal-04765327

<https://hal.science/hal-04765327v1>

Preprint submitted on 4 Nov 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Normalization properties of a computation process on colored finite trees

Karim Nour* and Ayman El Zein†

November 4, 2024

Abstract

In this paper, we introduce a concept of computation process on colored finite trees, which symbolize an inheritance process between fathers and sons, aiming to investigate their termination properties. We develop formal techniques to analyze their behavior and we present various results regarding the termination of computations performed on these trees. We show that these results can be generalized to digraphs through a simple transformation.

1 Introduction

A significant amount of research has been conducted on reductions in finite trees, most notably a famous work by mathematicians L. Kirby and J. Paris, which has been extensively studied by many researchers and is known as the "Hydra Battle" problem. The game essentially involves cutting a branch from a finite tree, then arbitrarily duplicating the neighboring tree at the branch's cut node. Figure 1 provides an idea of this type of game. We do not wish to provide more details here. It is not at all obvious that this game terminates. The termination proof employs ordinal numbers and is not formalizable within Peano arithmetic. The difficulty of the proof arises from the fact that a part of the game played on a simple and small tree can have a duration that far exceeds our intuition.

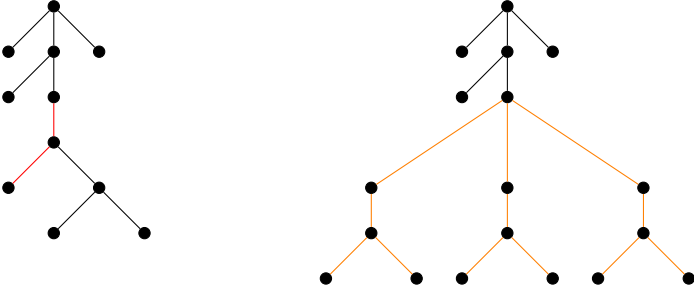


Figure 1: Hydra battle

It is quite natural to think of another game and ask a similar question: what can be done with a game rule that involves cutting branches from the tree but reattaching them (possibly duplicating them) at lower levels (unlike the previous game, where this was done only at the same level)? This game can be seen as a

*LAMA - Équipe LIMD, Université Savoie Mont Blanc, 73376 Le Bourget du Lac, France
karim.nour@univ-smb.fr

†Lebanese University, KALMA Laboratory, Beirut, Lebanon aymanelzein@live.com

process of transferring information from a higher level to lower levels, as shown in Figure 2. Clearly, this type of game can diverge indefinitely if no additional restrictions are imposed.

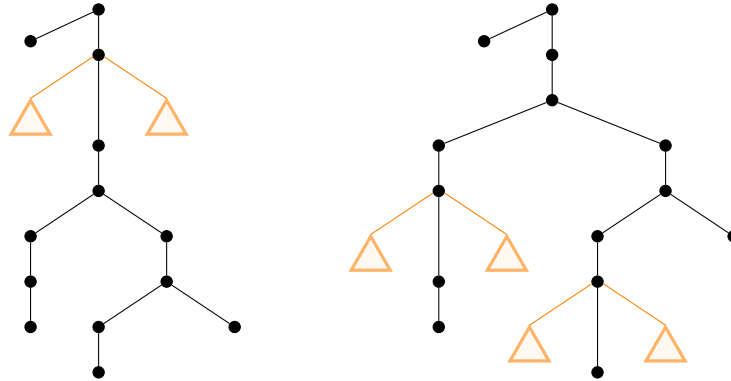


Figure 2: Another game

The idea would therefore be to indicate on the tree the specific locations where branches can be cut as well as the locations where they can be reattached.

In this paper, we introduce a game played on finite trees, focusing solely on the results related to termination. Here's an overview of the game.

- We start with a finite tree drawn from top to bottom, with the initial node at the top and branches leading to terminal nodes. Branch nodes are depicted in green, and terminal nodes in black. This tree serves as the skeleton of the game.
- To facilitate reduction on the tree, we introduce new nodes along the edges, colored red ("father" nodes) and blue ("son" nodes). Multiple additional nodes can be placed in a desired order along the edges, with each subtree's name at the base of the node.
- We establish the "father/son" relationship by adding oriented edges from a "father" node to one or more "son" nodes, provided the latter are within the subtree rooted at the father node. While a father node may have zero, one, or more son nodes, a son node can have at most one father.
- Each addition of father and son nodes, along with their associations, defines a game. This game simulates an inheritance process, where the father node aims to pass on surrounding subtrees to its sons via green branches below the red node.

Figure 3 represents the game shown in Figure 2.

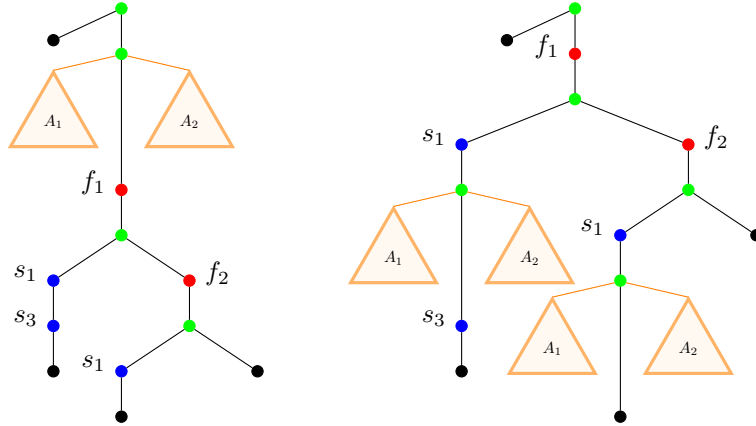


Figure 3: Representation of the new game

In summary, the game involves passing subtrees from one level of the tree to different levels. However, several phenomena can arise.

- A game step can create new situations, causing the tree size to grow arbitrarily.
- A son node can lead a subtree containing other sons with the same father, resulting in recursive inheritance.
- At each step of the game, there is a choice to be made for the reduction, which makes the game non-deterministic.

The termination of this reduction process is unclear. We pose two questions regarding termination.

- 1- **Weak normalization:** Is there an algorithm to complete this inheritance process, meaning to make the right choice at each step to ensure termination?
- 2- **Strong normalization:** Is the inheritance process terminating independent of the choice of reduction to be performed at each step?

In this paper, we prove that the inheritance game presented on trees possesses the property of strong normalization, meaning that the reduction process always terminates on any tree. Although our proof is non-constructive, as it relies on proof by contradiction, we also present a weak normalization result to obtain a particular termination algorithm.

Moreover, we introduce alternative reduction techniques to anticipate and simplify the main dynamics of the game. To our surprise, these techniques lead to the loss of the strong normalization property, resulting only in a weak normalization result for the trees.

We formalize trees completely colored by words over a vocabulary with a countable number of symbols. Terminal nodes are represented by black dots (\bullet), branching nodes by brackets ($[.]$), red "father" nodes by the letter f indexed by integers (f_i), and blue "son" nodes by the letter s indexed similarly (s_i). The indices represent father-son relationships. Note that these indices have no other significance and thus can be renamed to new indices.

The proofs presented in this paper are adaptations and generalizations of results obtained in a different context (see [1, 2]). Viewing the problem as a game on trees allows us to have two perspectives.

- Highlight arithmetic and rigorous demonstration techniques applicable in graph theory.
- Pose and attempt to solve questions within a broader framework, bridging techniques from graph theory and proof theory.

At the end of this paper, we present two distinct applications. The first focuses on a particular class of trees, demonstrating that the normalization results in Parigot's $\lambda\mu$ -calculus are special cases of our findings. The second application is quite surprising, as it extends our results to digraphs using the concept of a module. The results we obtain can be derived from those of the trees through a simple transformation.

2 Definitions

In this section, we will introduce colored trees as well as the main reduction rule, which will be studied later. We also provide the intuitions behind these concepts through examples, which will simplify the understanding of the concepts and proofs.

Definition 2.1 *Recursively, we may define the colored trees, the set of free sons $FS(\cdot)$ and the size $|\cdot|$ of the colored tree as follows.*

1. *We consider a set of symbols.*

- *A black dot : \bullet ,*
- *A bracket : $[\cdot]$,*
- *A set $\mathcal{F} = \{f_i / i \in \mathbb{N}\}$ (each f_i is called "father"),*
- *A set $\mathcal{S} = \{s_i / i \in \mathbb{N}\}$ (each s_i is called "son").*

2. *We define the colored tree A , the set $FS(A)$ and the integer $|A|$ as follows.*

- *$A = \bullet$ is a colored tree, with $|A| = 0$ and $FS(A) = \emptyset$.*
- *If A_1, \dots, A_n are colored trees, then $A = [A_1, \dots, A_n]$ is a colored tree, with $|A| = \sum_{i=1}^n |A_i|$ and $FS(A) = \bigcup_{i=1}^n FS(A_i)$. This new colored tree $[A_1, \dots, A_n]$ is considered as the concatenation of the colored trees A_1, \dots, A_n .*
- *If B is a colored tree, then $A = f_i B$ is a colored tree, with $|A| = |B| + 1$ and $FS(A) = FS(B) \setminus \{s_i\}$. The new colored tree $f_i B$ is seen as an assignment of a name f_i to the colored tree B .*
- *If B is a colored tree, then $A = s_i B$ is a colored tree, with $|A| = |B| + 1$ and $FS(A) = FS(B) \cup \{s_i\}$. The new colored tree $s_i B$ is seen as an assignment of a name s_i to the colored tree B .*

We denote by \mathbb{T} the set of colored trees.

3. *Note that when we say "induction on the tree A ", it means that it is by induction on the natural integer $|A|$.*

We consider the set \mathbb{T} modulo an equivalence, which allows changing the index of a father i to a new integer j (which does not appear in the tree), by replacing all the indices of its sons i , located in the subtree having f_i as the principal node, with the same integer j . In other words, we work with equivalence classes of trees where fathers are considered as binders for the sons that are below them and have the same indices. We do not wish to formally define this notion, which is quite intuitive. A

son s_i that is not under the influence of a father f_i (no tree containing s_i has f_i as its principal node) is said to be free. Therefore, the set $FS(A)$ represents the set of free sons of A . It is clear that we will not be able to rename the indices of free sons.

Example 2.2 We consider the colored tree

$$A = f_1 s_7 [f_2 f_4 [f_6 s_6 \bullet, s_2 s_2 s_4 \bullet], f_3 \bullet, s_1 f_5 s_5 [\bullet, \bullet]].$$

Note the difficulty in reading and understanding the structure of a colored tree in this example, particularly in identifying the open and closed brackets. The colored tree A intuitively represents the following tree. To better understand the structure of the colored tree, we have chosen red for the father nodes, blue for the son nodes, and green for the branching of its subtrees.

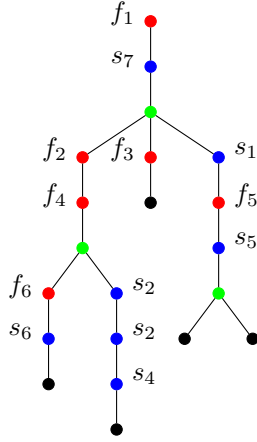


Figure 4: Representation of the colored tree A

To define a notion of computation over colored trees, we begin with the following definition.

Definition 2.3 Let $A, A_1, \dots, A_n \in \mathbb{T}$ and $s_{i_1}, \dots, s_{i_k} \notin \bigcup_{i=1}^n FS(A_i)$. Let $\sigma = \langle (s_{i_1}, \dots, s_{i_k}) := [A_1, \dots, A_j, *, A_{j+1}, \dots, A_n] \rangle$ that we call “substitution”. We define $A\sigma$ by induction on A as follows.

- If $A = \bullet$, then $A\sigma = \bullet$.
- If $A = [B_1, \dots, B_l]$, then $A\sigma = [B_1\sigma, \dots, B_l\sigma]$.
- If $A = f_m B$ with $m \notin \{i_1, \dots, i_k\}$, then $A\sigma = f_m B\sigma$.
Note that, by equivalence, we can assume that $m \notin \{i_1, \dots, i_k\}$, since otherwise, we can rename m by another new integer.
- If $A = s_m B$ with $m \notin \{i_1, \dots, i_k\}$, then $A\sigma = s_m B\sigma$.
- If $A = s_m B$ with $m \in \{i_1, \dots, i_k\}$, then $A\sigma = s_m [A_1, \dots, A_j, B\sigma, A_{j+1}, \dots, A_n]$.

The substitution we have just defined corresponds to the addition of the subtrees A_1, \dots, A_n around all the subtrees named by the sons s_{i_1}, \dots, s_{i_k} . More precisely, we place A_1, \dots, A_j to the left and A_{j+1}, \dots, A_n to the right of the subtrees named by s_{i_1}, \dots, s_{i_k} . Note that the notion of substitution, which will be used to define our

first reduction rule, will only involve a single son. Generalizing this definition will be useful for proving some results. It is clear that if additionally $s_{i_1}, \dots, s_{i_k} \notin FS(A)$, then $A\sigma = A$.

Example 2.4 Let $B = [f_1 s_2 [\bullet, \bullet], s_3 s_3 f_4 \bullet]$ and $\sigma = \langle (s_2, s_3) := [A_1, *, A_2] \rangle$. We have $B\sigma = [f_1 s_2 [A_1, [\bullet, \bullet], A_2], s_3 [A_1, s_3 [A_1, f_4 \bullet, A_2], A_2]]$. It is very easy to visualize this substitution as in Figure 5, which will also help to understand the concept of reduction clearly.

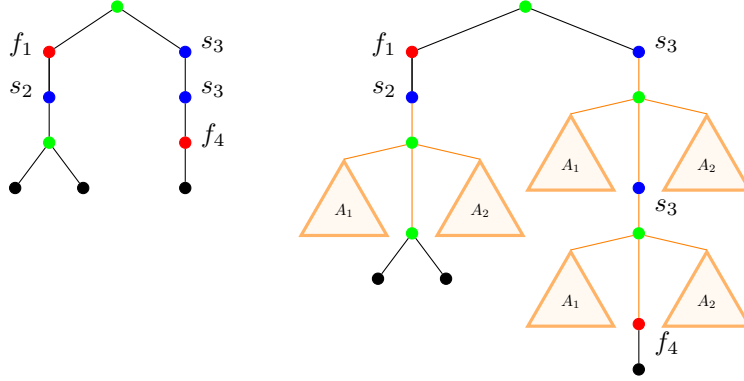


Figure 5: An example of a substitution

Definition 2.5 Let $A \in \mathbb{T}$ and $s_i, s_j \in \mathcal{S}$ such that f_j does not appear in A (we can always assume this). We define $A[s_i := s_j]$ being the colored tree obtained by replacing in A the free sons s_i by s_j .

Now, we have to define a reduction over the colored trees. The first one is called the inheritance and symbolized by “ h ”.

Definition 2.6 An h -redex is a colored tree under the form

$$[A_1, \dots, A_k, f_i B, A_{k+1}, \dots, A_n],$$

where $n \neq 0$. Its h -reduced via f_i is the following colored tree

$$f_i B \langle (s_i) := [A_1, \dots, A_k, *, A_{k+1}, \dots, A_n] \rangle.$$

Note that in this case, we can assume that $s_i \notin \bigcup_{i=1}^n FS(A_i)$, thus the substitution is well-defined. Intuitively, this operation consists of cutting the colored trees located above the colored tree named by f_i , then attaching them around all colored trees named by an s_i .

It is very easy to recognize an h -redex, as this is characterized by finding a father tree within a bracket $[.]$. A bracket can be considered as containing several h -redexes if it contains multiple father trees. Figure 6 shows how the h -reduced of an h -redex is obtained. The situation is much more complex than what we observe in Figure 6, as we may have multiple independent or nested sons s_i that will undergo substitution after the reduction.

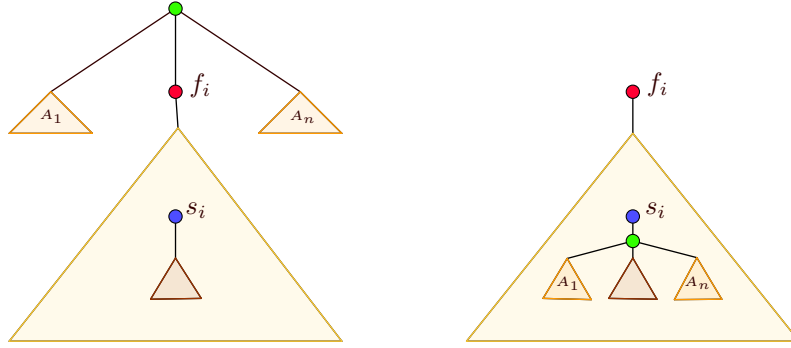


Figure 6: h -redex and h -reduced

Example 2.7 The colored tree A of Example 2.2 has three h -redexes $[f_2 f_4 [f_6 s_6 \bullet, s_2 s_2 s_4 \bullet], f_3 \bullet, s_1 f_5 s_5 [\bullet, \bullet]]$ (this redex counts as two) and $[f_6 s_6 \bullet, s_2 s_2 s_4 \bullet]$.

Definition 2.8 If $A, B \in \mathbb{T}$, we define $A \rightarrow_h B$ recursively on A as follows.

- If $A = \bullet$, then it is impossible.
- If $A = f_i A_1$, then $B = f_i B_1$ and $A_1 \rightarrow_h B_1$.
- If $A = s_i A_1$, then $B = s_i B_1$ and $A_1 \rightarrow_h B_1$.
- If $A = [A_1, \dots, A_n]$, then there are two cases
 - $B = [A_1, \dots, A_{k-1}, B_k, A_{k+1}, \dots, A_n]$ where $A_k \rightarrow_h B_k$.
 - There is an $A_i = f_j B_i$ and B is the h -reduced colored tree of A via f_j .

Example 2.9 The colored tree A of Example 2.2 can be h -reduced via f_2 , f_3 and f_6 . Besides, the three possible h -reductions:

- Via f_2 : $A \rightarrow_h f_1 s_7 f_2 f_4 [f_6 s_6 \bullet, s_2 [s_2 [s_4 \bullet, f_8 \bullet, s_1 f_9 s_9 [\bullet, \bullet]], f_3 \bullet, s_1 f_5 s_5 [\bullet, \bullet]]]$.
We give the name f_8 (resp. f_9) to the second f_3 (resp. f_5) in order to avoid confusion. Notice the appearance of other h -redexes in the new colored tree.
- Via f_3 : $A \rightarrow_h f_1 s_7 f_3 \bullet$.
- Via f_6 : $A \rightarrow_h f_1 s_7 [f_2 f_4 f_6 s_6 [\bullet, s_2 s_2 s_4 \bullet], f_3 \bullet, s_1 f_5 s_5 [\bullet, \bullet]]$.

We leave it to the readers to create the tree representations of these three h -reductions.

Our main study focuses on the normalization of colored trees. We will provide answers to the following two questions:

- Is there a finite sequence of h -reductions for any colored tree?
- If so, is every sequence of h -reductions finite?

To answer these questions, we define the following notions.

Definition 2.10

1. If $A, B \in \mathbb{T}$, we write $A \rightarrow_h B$ if there exist a sequence $A_1, \dots, A_n \in \mathbb{T}$ such that $A \rightarrow_h A_1 \rightarrow_h A_2 \rightarrow_h \dots \rightarrow_h A_n = B$. The reduction \rightarrow_h is therefore the reflexive and transitive closure of the reductio \rightarrow_h .

2. A colored tree is said to be *h-normal* if it contains no *h-redex* i.e. there is no possible *h-reduction*.
3. A colored tree A is said to be *weakly h-normalizable* if there exists an *h-normal* colored tree B such that $A \rightarrow_h B$.
4. A colored tree A is said to be *strongly h-normalizable* if any sequence of *h-reduction* of A ends.

We observe that after an *h-reduction* in a colored tree, two phenomena may arise that hinder the termination of the computation and the elimination of all redexes.

- The creation of new redexes that were not present in the initial colored tree.
- The arbitrary duplication of subtrees, leading to an increase in the size of the initial tree.

This raises the problem of choosing the best redex to reduce at each step in order to eventually reach an *h-normal* tree. In Sections 3 and 4, we will prove the weak *h-normalization* and strong *h-normalization* properties for all colored trees.

To simplify the notation, we denote

- $[A_1, \dots, A_k, B, A_{k+1}, \dots, A_n]$ by $[A_1, \dots, B, \dots, A_n]$,
- $[A_1, \dots, A_k, *, A_{k+1}, \dots, A_n]$ by $[A_1, \dots, *, \dots, A_n]$.

3 The weak normalization of the *h-reduction*

Although the property of weak *h-normalization* follows from that of strong *h-normalization*, in this section, we will provide a constructive proof of weak *h-normalization*, which will allow us to develop a particular algorithm. Moreover, this proof could offer useful insights in other types of calculations. In particular, we will use it in Section 5 to prove the weak normalization property of a set of reductions for which we will not have a strong normalization result.

Intuitively, the following lemma shows that it is impossible to create a father through a substitution followed by an *h-reduction*.

Lemma 3.1 *Let $A, B_1, \dots, B_n \in \mathbb{T}$ and $\sigma = \langle (s_{i_1}, \dots, s_{i_m}) := [B_1, \dots, *, \dots, B_n] \rangle$. If $A\sigma \rightarrow_h f_i B$, then $A \rightarrow_h f_i C$.*

Proof By induction on A .

- If $A = f_i A_1$, then we obtain the result immediately.
- If $A = s_i A_1$, then $A\sigma = s_i A_1\sigma$. So there is no colored tree B such that $A\sigma \rightarrow_h f_j B$.
- If $A = [A_1, \dots, A_k]$, then $A\sigma = [A_1\sigma, \dots, A_k\sigma]$. As $A\sigma \rightarrow_h f_i B$, there exists $1 \leq j \leq k$ such that $A_j\sigma \rightarrow_h f_i B'$. By the induction hypothesis, $A_j \rightarrow_h f_i C'$, thus $A \rightarrow_h [A_1, \dots, f_i C', \dots, A_k] \rightarrow_h f_i C$.

□

The following definitions help us to prove our result.

Definition 3.2

1. If $A, B \in \mathbb{T}$, we say that B is a subtree of A if B is used in the inductive definition of A .
2. Let $[A_1, \dots, A_m]$ be an h -redex of $A \in \mathbb{T}$. The subtree $[A_1, \dots, A_m]$ is said to be protected by a son if it is preceded by a son s_i . In this case, $s_i[A_1, \dots, A_m]$ is a subtree of A .
3. The order of an h -redex $[A_1, \dots, A_n]$ is $o([A_1, \dots, A_n]) = (p, n)$, where p is the number of A_i that start by a father.
4. Let $n, k \in \mathbb{N}$ such that $0 \leq k \leq n$. A colored tree is said to be (k, n) -good if all its h -redex are of order (k, n) and are protected by a son. By convention, a colored tree is said to be $(0, n)$ -good if it is h -normal.

The following lemma is very intuitive and describes the form of a colored tree after two substitutions.

Lemma 3.3 Let $A, B_1, \dots, B_n \in \mathbb{T}$,
 $\sigma = \langle (s_{i_1}, \dots, s_{i_m}) := [B_1, \dots, B_k, *, B_{k+1}, \dots, B_n] \rangle$,
 $\sigma_1 = \langle (s_i) := [B_1, \dots, B_k, *, B_{k+1}, \dots, B_n] \rangle$, and
 $\sigma_2 = \langle (s_{i_1}, \dots, s_{i_m}, s_i) := [B_1, \dots, B_k, *, B_{k+1}, \dots, B_n] \rangle$.
We have $A\sigma\sigma_1 = A\sigma_2$

Proof By induction on A .

- If $A = f_i A_1$, then $A\sigma = f_i A_1\sigma$ and $A\sigma\sigma_1 = f_i A_1\sigma\sigma_1$. By the induction hypothesis, $A_1\sigma\sigma_1 = A_1\sigma_2$, thus $A\sigma\sigma_1 = f_i A_1\sigma_2 = A\sigma_2$.
- If $A = s_j A_1$, we have tree subcases.
 - If $s_j \notin \{s_{i_1}, \dots, s_{i_m}\}$, then $A\sigma = s_j A_1\sigma$ and $A\sigma\sigma_1 = s_j A_1\sigma\sigma_1$. By the induction hypothesis, $A_1\sigma\sigma_1 = A_1\sigma_2$, thus $A\sigma\sigma_1 = A\sigma_2$.
 - If $j \in \{i_1, \dots, i_m\}$, then $A\sigma = s_j [B_1, \dots, A_1\sigma, \dots, B_n]$ and $A\sigma\sigma_1 = s_j [B_1\sigma_1, \dots, A_1\sigma\sigma_1, \dots, B_n\sigma_1]$. By the induction hypothesis, $A_1\sigma\sigma_1 = A_1\sigma_2$, and, since $s_{i_1}, \dots, s_{i_m} \notin \bigcup_{i=1}^n FS(B_i)$, for all $1 \leq i \leq n$, $B_i\sigma_1 = B_i$, thus $A\sigma\sigma_1 = s_j [B_1, \dots, A_1\sigma_2, \dots, B_n] = A\sigma_2$.
 - If $j = i$, then $A\sigma = s_j A_1\sigma$ and $A\sigma\sigma_1 = s_j [B_1, \dots, A_1\sigma\sigma_1, \dots, B_n]$. By the induction hypothesis, $A_1\sigma\sigma_1 = A_1\sigma_2$, thus $A\sigma\sigma_1 = s_j [B_1, \dots, A_1\sigma_2, \dots, B_n] = A\sigma_2$.
- If $A = [A_1, \dots, A_k]$, then $A\sigma = [A_1\sigma, \dots, A_k\sigma]$ and $A\sigma\sigma_1 = [A_1\sigma\sigma_1, \dots, A_k\sigma\sigma_1]$. By the induction hypothesis, for all $1 \leq r \leq k$, $A_r\sigma\sigma_1 = A_r\sigma_2$, thus, $A\sigma\sigma_1 = [A_1\sigma_2, \dots, A_k\sigma_2] = A\sigma_2$. □

Below are the two key lemmas to prove our main result. First, we show (Lemma 3.4) that starting from an h -normal colored tree and a substitution that only uses h -normal colored trees, we can perform an h -reduction and stop at a colored tree containing the same type of h -redexes. Then, we will demonstrate (Lemma 3.5) that such a colored tree can be weakly normalized.

Lemma 3.4 *Let A, B_1, \dots, B_n be h -normal colored trees such that k of B_1, \dots, B_n starts by a father and $\sigma = \langle (s_{i_1}, \dots, s_{i_m}) := [B_1, \dots, *, \dots, B_n] \rangle$. There exists a $(k, n+1)$ -good colored tree B such that $A\sigma \rightarrow_h B$.*

Proof By induction on A .

- If $A = f_i A_1$, then $A\sigma = f_i A_1 \sigma$. By the induction hypothesis, $A_1 \sigma \rightarrow_h C$ with C is $(k, n+1)$ -good, therefore $A\sigma \rightarrow_h B = f_i C$ with B is $(k, n+1)$ -good.
- If $A = s_i A_1$ with $i \notin \{i_1, \dots, i_m\}$, then $A\sigma = s_i A_1 \sigma$. By the induction hypothesis, $A_1 \sigma \rightarrow_h C$ with C is $(k, n+1)$ -good, therefore $A\sigma \rightarrow_h B = s_i C$ with B is $(k, n+1)$ -good.
- If $A = s_i A_1$ with $i \in \{i_1, \dots, i_m\}$, then $A\sigma = s_i [B_1, \dots, A_1 \sigma, B \dots, B_n]$. By the induction hypothesis, $A_1 \sigma \rightarrow_h C$ with C is $(k, n+1)$ -good. If C does not start by a father, then $s_i [B_1, \dots, C, \dots, B_n]$ is $(k, n+1)$ -good i.e. $A\sigma \rightarrow_h B$ with B is $(k, n+1)$ -good. Otherwise, C starts by a father f_j , by Lemma 3.1, $A_1 \rightarrow_h f_j D$, but A is h -normal, so $A_1 = f_j D$. If we reduce $A\sigma$ via f_j , then, by Lemma 3.3, $A\sigma \rightarrow_h s_i f_j D \sigma'$ where $\sigma' = \langle (s_{i_1}, \dots, s_{i_m}, s_j) := [B_1, \dots, *, \dots, B_n] \rangle$. By the induction hypothesis, $D\sigma' \rightarrow_h E$ where E is $(k, n+1)$ -good. Thus $A\sigma \rightarrow_h B$ with B is $(k, n+1)$ -good.
- If $A = [A_1, \dots, A_l]$, then $A\sigma = [A_1 \sigma, \dots, A_l \sigma]$. By the induction hypothesis, $A_i \sigma \rightarrow_h C_i$ with C_i is $(k, n+1)$ -good. If there exists C_i starting by a father, then, by Lemma 3.1, $A_i \rightarrow_h f_j D_i$. But A_i is h -normal, so $A_i = f_j D_i$ and then A has a h -redex, a contradiction. So $A\sigma \rightarrow_h B$ with B is $(k, n+1)$ -good. \square

Note that in the proof of Theorem 3.6, we only need one free son in the preceding lemma, but to prove the lemma, it was necessary to state it with one or more free sons.

Lemma 3.5 *A (k, m) -good colored tree A is weakly h -normalizable and if $A \rightarrow_h B$ with B is h -normal, then B starts by a father if and only if A starts by a father.*

Proof By induction on $(k, |A|)$ using the lexicographic order. The case $k = 0$ is trivial since A will be h -normal. Suppose that $k \geq 1$.

- If $A = f_i A_1$, then A_1 is (k, m) -good and $|A_1| < |A|$, therefore, by the induction hypothesis, $A_1 \rightarrow_h B_1$ with B_1 is h -normal, therefore, $A \rightarrow_h B = f_i B_1$ with B is h -normal.
- If $A = s_i A_1$, then A_1 is (k, m) -good and $|A_1| < |A|$, therefore, by the induction hypothesis, $A_1 \rightarrow_h B_1$ with B_1 is h -normal, therefore, $A \rightarrow_h B = s_i B_1$ with B is h -normal and $B \neq f_j C$.
- If $A = s_j [A_1, \dots, A_m]$ with k of the A_i start by a father, then for all $1 \leq i \leq m$, A_i is (k, m) -good with $|A_i| < |A|$ so, by the induction hypothesis, $A_i \rightarrow_h B_i$ with B_i is h -normal and k of the B_i start by a father. Let $1 \leq i_0 \leq m$ such that B_{i_0} starts by a father f_{i_0} . Reducing via f_{i_0} , $A \rightarrow_h s_j f_{i_0} B_{i_0} \sigma$ where $\sigma = \langle (s_{i_0}) := [B_1, \dots, *, \dots, B_m] \rangle$. By Lemma 3.4, $B_{i_0} \sigma \rightarrow_h C_{i_0}$ with C_{i_0} is $(k-1, m)$ -good and, by the induction hypothesis, since we are using the lexicographic order, $C_{i_0} \rightarrow_h D_{i_0}$ with D_{i_0} is h -normal. Hence, $A \rightarrow_h B = s_j f_{i_0} D_{i_0}$ where B is h -normal and $B \neq f_i C$.

- If $A = [A_1, \dots, A_n]$, then A_i doesn't start by a father for all $1 \leq i \leq n$, since, otherwise, A is not (k, m) -good. For all $1 \leq i \leq n$, A_i is (k, m) -good with $|A_i| < |A|$, so, by the induction hypothesis, $A_i \rightarrow_h B_i$ with B_i is h -normal and doesn't start by a father, therefore, $A \rightarrow_h B = [B_1, \dots, B_n]$ with B is h -normal and $B \neq f_l C$. □

Using the above lemmas, we can now establish the property of weak h -normalization.

Theorem 3.6 *Every colored tree A is weakly h -normalizable.*

Proof By induction on A .

- If $A = f_i A_1$, then, by the induction hypothesis, $A_1 \rightarrow_h B_1$ with B_1 is h -normal. Hence, $A \rightarrow_h B = f_i B_1$, where B is h -normal. The result follows.
- If $A = s_i A_1$, then, by the induction hypothesis, $A_1 \rightarrow_h B_1$ with B_1 is h -normal. Hence, $A \rightarrow_h B = s_i B_1$, where B is h -normal. The result follows.
- If $A = [A_1, \dots, A_m]$, then, by the induction hypothesis, $A_i \rightarrow_h B_i$ with B_i is h -normal, for all $1 \leq i \leq n$. If B_i doesn't start by a father for all $1 \leq i \leq n$, then $A \rightarrow_h B = [B_1, \dots, B_m]$ where B is h -normal. Otherwise, let $1 \leq i_0 \leq m$ such that B_{i_0} starts by a father which is denoted by f_{i_0} . Reducing via f_{i_0} , $B \rightarrow_h f_{i_0} B_{i_0} \sigma$, where $\sigma = \langle (s_{i_0}) := [B_1, \dots, *, \dots, B_m] \rangle$. By Lemma 3.4, $B_{i_0} \sigma \rightarrow_h C_{i_0}$ where C_{i_0} is $(k, m + 1)$ -good, hence, by Lemma 3.5, $C_{i_0} \rightarrow_h D_{i_0}$ where D_{i_0} is h -normal. The result follows. □

We summarize what we did to present a normalization algorithm.

1. Lemma 3.4 gives an algorithm to transform a colored tree of the form $A\sigma$ where A is a h -normal colored tree, $\sigma = \langle (s_{i_1}, \dots, s_{i_m}) := [B_1, \dots, *, \dots, B_n] \rangle$ and B_1, \dots, B_n are h -normal colored trees, to a (k, m) -good colored tree.
2. Lemma 3.5 gives an algorithm (using the first one) to h -normalize a (k, m) -good colored tree.
3. Theorem 3.6 gives an algorithm to h -normalize a colored tree A (by induction on A). In the case where $A = [A_1, \dots, A_n]$, the first two algorithms are successively used. Case 3 in the proof of Theorem 3.6 leaves us with a non-deterministic choice concerning the process of finding a h -normal form of A . This means that instead of one uniquely determined h -normal form we end up with one of the possible h -normal forms of A .

4 The strong normalization of the h -reduction

The proof we presented in the previous section allows us to establish normalization algorithms to transform a colored tree into an h -normal form. In this section, we will establish a stronger result, namely the strong h -normalization of any colored tree. This means that no matter which redex is chosen for reduction, after several h -reductions, we will eventually reach an h -normal colored tree. The proof of this result is non-constructive as it relies on a proof by contradiction.

To establish the property of strong h -normalization, several preliminary lemmas are proven. The first lemma provides a characterization of the phenomenon of creating a father at the head of a colored tree after a sequence of h -reductions on a colored tree of the form $[A_1, \dots, A_n]$.

Lemma 4.1 *If $A_1, \dots, A_m, A \in \mathbb{T}$ and $[A_1, \dots, A_m] \twoheadrightarrow_h f_i A$, then there exists $1 \leq j \leq m$ such that $A_j \twoheadrightarrow_h f_i B_j$, $A_j \twoheadrightarrow_h A'_j$ for all $1 \leq i \leq m$ and $i \neq j$, and $[A_1, \dots, A_m] \twoheadrightarrow_h [A'_1, \dots, f_i B_j, \dots, A'_m] \twoheadrightarrow_h f_i B_j \sigma \twoheadrightarrow_h f_i A$.*

Proof By induction on the number of h -reductions between $[A_1, \dots, A_m]$ and $f_i A$. If there is only one h -reduction, the result follows by the definition of the h -reduction. So, assume that there are two or more h -reductions. Let A' be a colored tree such that $[A_1, \dots, A_m] \twoheadrightarrow_h A' \twoheadrightarrow_h f_i A$. If $A' = f_i C$, we obtain the result by the induction hypothesis. Otherwise, $A' = [A_1, \dots, B_k, \dots, A_m]$ such that $A_k \twoheadrightarrow_h B_k$ for $1 \leq j \leq m$. By the induction hypothesis, there exists $1 \leq j \leq m$ such that $j \neq k$ and $A_j \twoheadrightarrow_h f_i B_j$ or $A_k \twoheadrightarrow_h B_k \twoheadrightarrow_h f_i B_j$ otherwise. The result follows. \square

The following lemma provides a condition for permuting substitutions. It will be used in the proof of Lemma 4.3.

Lemma 4.2 *Let $B, A_1, \dots, A_m, B_1, \dots, B_n \in \mathbb{T}$, $s_i, s_j \in \mathcal{S}$ such that $s_i \neq s_j$ and $s_j \notin \bigcup_{i=1}^n FS(B_i)$, $\sigma = \langle (s_i) := [B_1, \dots, *, \dots, B_n] \rangle$, $\delta = \langle (s_j) := [A_1, \dots, *, \dots, A_m] \rangle$ and $\rho = \langle (s_j) := [A_1 \sigma, \dots, *, \dots, A_m \sigma] \rangle$. We have $A \sigma \rho = A \delta \sigma$.*

Proof By induction on A .

- If $A = f_k B$, then, by the induction hypothesis, $A \sigma \rho = f_k(B \sigma \rho) = f_k(B \delta \sigma) = A \delta \sigma$.
- If $A = s_k B$ with $k \neq i$ and $k \neq j$, then, by the induction hypothesis, $A \sigma \rho = s_k(B \sigma \rho) = s_k(B \delta \sigma) = A \delta \sigma$.
- If $A = s_i B$, then $A \sigma \rho = s_i[B_1, \dots, B \sigma, \dots, B_n] \rho = s_i[B_1 \rho, \dots, B \sigma \rho, \dots, B_n \rho] = s_i[A_1, \dots, B \sigma \rho, \dots, A_n]$ and $A \delta \sigma = (s_i B \delta) \sigma = s_i[B_1, \dots, B \delta \sigma, \dots, B_n]$, hence, by the induction hypothesis, we obtain the result.
- If $A = s_j B$, then $A \sigma \rho = (s_j B \sigma) \rho = s_j[A_1 \sigma, \dots, B \sigma \rho, \dots, A_m \sigma]$ and $A \delta \sigma = (s_j[A_1, \dots, B \delta, \dots, A_m]) \sigma = s_j[A_1 \sigma, \dots, B \delta \sigma, \dots, A_m \sigma]$, hence, by the induction hypothesis, we obtain the result.
- If $A = [A'_1, \dots, A'_k]$, then, by the induction hypothesis, $A \sigma \rho = [A'_1 \sigma \rho, \dots, A'_k \sigma \rho] = [A'_1 \delta \sigma, \dots, A'_k \delta \sigma] = A \delta \sigma$. \square

The following lemma is intuitive and expresses a useful property about the reduction of a colored tree affected by a substitution.

Lemma 4.3 *Let $A, A', B_1, \dots, B_n, B'_1, \dots, B'_n \in \mathbb{T}$ such that $A \twoheadrightarrow_h A'$ and $B_i \twoheadrightarrow_h B'_i$ for all $1 \leq i \leq n$. Let $\sigma = \langle (s_i) := [B_1, \dots, *, \dots, B_n] \rangle$ and $\sigma' = \langle (s_i) := [B'_1, \dots, *, \dots, B'_n] \rangle$. We have : 1. $A \sigma \twoheadrightarrow_h A \sigma'$. 2. $A \sigma \twoheadrightarrow_h A' \sigma$. 3. $A \sigma \twoheadrightarrow_h A' \sigma'$.*

Proof

1. By induction on A .

- If $A = f_j B$, then $A \sigma = f_j B \sigma$ and $A \sigma' = f_j B \sigma'$. By the induction hypothesis, $B \sigma \twoheadrightarrow_h B \sigma'$, thus $A \sigma \twoheadrightarrow_h A \sigma'$.
- If $A = s_j B$ with $j \neq i$, then $A \sigma = s_j B \sigma$ and $A \sigma' = s_j B \sigma'$. By the induction hypothesis, $B \sigma \twoheadrightarrow_h B \sigma'$, thus $A \sigma \twoheadrightarrow_h A \sigma'$.
- If $A = s_i B$ with $1 \leq i \leq n$, then $A \sigma = s_i[B_1, \dots, B \sigma, \dots, B_n]$ and $A \sigma' = s_i[B_1, \dots, B \sigma', \dots, B_n]$. By the induction hypothesis, $B \sigma \twoheadrightarrow_h B \sigma'$, thus $A \sigma \twoheadrightarrow_h A \sigma'$.

- If $A = [A_1, \dots, A_m]$, then $A\sigma = [A_1\sigma, \dots, A_m\sigma]$ and $A\sigma' = [A_1\sigma', \dots, A_m\sigma']$. By the induction hypothesis, $A_i\sigma \twoheadrightarrow_h A_i\sigma'$ for all $1 \leq i \leq m$, thus $A\sigma \twoheadrightarrow_h A\sigma'$.

2. It is sufficient to prove the result for a one step of reduction, i.e. we suppose $A \rightarrow_h A'$ and we prove, by induction on A , $A\sigma \rightarrow_h A'\sigma$.

- If $A = f_j B$, then $A\sigma = f_j B\sigma$ and $A' = f_j B'$ where $B \rightarrow_h B'$. By the induction hypothesis, $B\sigma \rightarrow_h B'\sigma$, thus $A\sigma \rightarrow_h f_j B'\sigma = A'\sigma$.
- If $A = s_j B$ with $j \neq i$, then $A\sigma = s_j B\sigma$ and $A' = s_j B'$ where $B \rightarrow_h B'$. By the induction hypothesis, $B\sigma \rightarrow_h B'\sigma$, thus $A\sigma \rightarrow_h s_j B'\sigma = A'\sigma$.
- If $A = s_i B$ with $1 \leq i \leq n$, then $A\sigma = s_i[B_1, \dots, B\sigma, \dots, B_n]$ and $A' = s_i B'$ where $B \rightarrow_h B'$. By the induction hypothesis, $B\sigma \rightarrow_h B'\sigma$, thus $A\sigma \rightarrow_h s_i[B_1, \dots, B'\sigma, \dots, B_n] = A'\sigma$.
- If $A = [A_1, \dots, A_m]$, then $A\sigma = [A_1\sigma, \dots, A_m\sigma]$ and we have two cases to study.
 - If $A' = [A_1, \dots, A'_i, \dots, A_m]$ where $A_i \rightarrow_h A'_i$, then, by the induction hypothesis, $A_i\sigma \rightarrow_h A'_i\sigma$, thus $A\sigma \rightarrow_h [A_1\sigma, \dots, A'_i\sigma, \dots, A_m\sigma] = A'\sigma$.
 - If $A = [A_1, \dots, f_j B_i, \dots, A_m]$ and $A' = f_j B_i \delta$ where $\delta = \langle (s_j) := [A_1, \dots, *, \dots, A_m] \rangle$, then $A\sigma = [A_1\sigma, \dots, f_j B_i\sigma, \dots, A_m\sigma] \rightarrow_h f_j B_i\sigma\rho$ where $\rho = \langle (s_j) := [A_1\sigma, \dots, *, \dots, A_m\sigma] \rangle$, thus, by Lemma 4.2, $A\sigma \rightarrow_h A'\sigma$.

3. It is sufficient to use parts 1 and 2 of this lemma. □

The following lemma provides a characterization of the phenomenon of creating a father at the head of a colored tree after a sequence of h -reductions on a substitution of a colored tree.

Lemma 4.4 *Let $A, B_1, \dots, B_n \in \mathbb{T}$ and $\sigma = \langle (s_j) := [B_1, \dots, *, \dots, B_n] \rangle$. If $A\sigma \twoheadrightarrow_h f_i B$, then $A \twoheadrightarrow_h f_i A'$ and $A'\sigma \twoheadrightarrow_h B$.*

Proof By induction on A .

- If $A = f_i C$, we have the result immediately.
- If $A = s_i C$, then A cannot be h -reduced to $f_i B$.
- If $A = [A_1, \dots, A_k]$, then $A\sigma = [A_1\sigma, \dots, A_k\sigma]$, and, by Lemma 4.1, there exists $1 \leq j \leq k$ such that $A_j\sigma \twoheadrightarrow_h f_i C$, $A_r\sigma \twoheadrightarrow_h A'_r$ for all $1 \leq r \leq k$ and $k \neq j$, and $[A_1, \dots, A_k] \twoheadrightarrow_h [A'_1, \dots, f_i C, \dots, A'_k] \rightarrow_h f_i C\sigma' \twoheadrightarrow_h f_i B$ where $\sigma' = \langle (s_i) := [A'_1, \dots, *, \dots, A'_k] \rangle$. By the induction hypothesis, $A_j \twoheadrightarrow_h f_i D$ and $D\sigma \twoheadrightarrow_h C$, thus $A \twoheadrightarrow_h [A_1, \dots, f_i D, \dots, A_k] \rightarrow_h f_i D\delta$ where $\delta = \langle (s_i) := [A_1, \dots, *, \dots, A_k] \rangle$. By Lemma 4.2, $D\delta\sigma = D\sigma\rho$ where $\rho = \langle (s_i) := [A_1\sigma, \dots, *, \dots, A_k\sigma] \rangle$, thus, by Lemma 4.3, $D\delta\sigma \twoheadrightarrow_h C\sigma' \twoheadrightarrow_h B$. □

The following König's lemma is essential for defining a measure on strongly h -normalizable colored trees. We state the lemma without providing precise definitions of the terms used, especially since we only need this lemma to define a notion of measure.

Lemma 4.5 (König's lemma) *Every connected, locally finite, infinite undirected graph contains an infinite path (ray).*

Definition 4.6 *Let A be a strongly h -normalizable colored tree. Since the reduction tree of A is locally finite by König's Lemma, we can denote the length of the longest reduction sequence of A by $\eta(A)$. We immediately deduce the following properties.*

1. *If $C \rightarrow_h D$, then $\eta(C) > \eta(D)$.*
2. *To prove that a colored tree C is strongly h -normalizable, it suffices to show that if $C \rightarrow_h D$, then D is strongly h -normalizable.*

We obtain also the following property.

- $\eta(C) = 0$ if and only if C is h -normal.
- $\eta(C) = \max_{C \rightarrow_h D} \eta(D) + 1$ if C is not h -normal.

The following definitions are useful in the proof of the main result.

Definition 4.7 *If $A, B \in \mathbb{T}$, we write*

- $B \sqsubseteq A$ if B is a colored subtree of A ,
- $B \preceq A$ if $B \sqsubseteq A'$ and $A \rightarrow_h A'$.

The two lemmas below explain why a colored tree may not be strongly h -normalizable even if some of its subtrees are.

Lemma 4.8 *Let A_1, \dots, A_n be strongly h -normalizable colored trees. If $[A_1, \dots, A_n]$ is not strongly h -normalizable, then there exists $1 \leq i \leq n$ such that $A_i \rightarrow_h f_j A'_i$ and $A'_i \sigma$ is not strongly h -normalizable, where $\sigma = \langle (s_j) := [A_1, \dots, *, \dots, A_n] \rangle$.*

Proof By induction on $\sum_{i=1}^n \eta(A_i)$. We consider an infinite reduction of $[A_1, \dots, A_n]$ and distinguish two cases for the first reduct.

- If $[A_1, \dots, A_n] \rightarrow_h [A'_1, \dots, A'_n]$, then $[A'_1, \dots, A'_n]$ is not strongly h -normalizable and $\sum_{i=1}^n \eta(A'_i) < \sum_{i=1}^n \eta(A_i)$, then, by the induction hypothesis, there exists $1 \leq i \leq n$ such that $A'_i \rightarrow_h f_j A''_i$ and $A''_i \sigma'$ is not strongly h -normalizable where $\sigma' = \langle (s_j) := [A'_1, \dots, *, \dots, A'_n] \rangle$, thus, $A_i \rightarrow_h f_j A'_i$ and, by Lemma 4.3, $A'_i \sigma \rightarrow_h A''_i \sigma'$, therefore $A'_i \sigma$ is not strongly h -normalizable.
- If there exists $1 \leq i \leq n$ such that $A_i = f_j A'_i$ and $[A_1, \dots, f_i A'_i, \dots, A_n] \rightarrow_h f_i A'_i \sigma$ where $\sigma = \langle (s_j) := [A_1, \dots, *, \dots, A_n] \rangle$, then $A'_i \sigma$ is not strongly h -normalizable. □

Lemma 4.9 *Let A, B_1, \dots, B_n be strongly h -normalizable colored trees and $\sigma = \langle (s_i := [B_1, \dots, *, \dots, B_n]) \rangle$. If $A\sigma$ is not strongly h -normalizable, then there exists $s_i B \preceq A$ such that $B\sigma$ is strongly h -normalizable and $[B_1, \dots, B\sigma, \dots, B_n]$ is not strongly h -normalizable.*

Proof By induction on $(\eta(A), |A|)$ using the lexicographic order.

- If $A = f_j A'$, then, by the induction hypothesis, there exists $s_i B \preceq A' \preceq A$ such that $B\sigma$ is strongly h -normalizable and $[B_1, \dots, B\sigma, \dots, B_n]$ is not strongly h -normalizable.
- If $A = s_j A'$ with $j \neq i$, then, by the induction hypothesis, there exists $s_i B \preceq A' \preceq A$ such that $B\sigma$ is strongly h -normalizable and $[B_1, \dots, B\sigma, \dots, B_n]$ is not strongly h -normalizable.
- If $A = s_i A'$, then $A\sigma = s_i[B_1, \dots, A'\sigma, \dots, B_n]$. If $A'\sigma$ is not strongly h -normalizable, then, by the induction hypothesis, there exists $s_i B \preceq A' \preceq A$ such that $B\sigma$ is strongly h -normalizable and $[B_1, \dots, B\sigma, \dots, B_n]$ is not strongly h -normalizable. Otherwise, $s_i A' \preceq A$ such that $A'\sigma$ is strongly h -normalizable and $[B_1, \dots, A'\sigma, \dots, B_n]$ is not strongly h -normalizable.
- If $A = [A_1, \dots, A_m]$, we have $A\sigma = [A_1\sigma, \dots, A_m\sigma]$. By Lemma 4.8, there exists $1 \leq j \leq m$ such that $A_j\sigma \rightarrow_h f_k A'_j$ and $A'_j\sigma'$ is not strongly h -normalizable, where $\sigma' = \langle (s_k := [[A_1\sigma, \dots, *, \dots, A_m\sigma]]) \rangle$. By Lemma 4.4, $A_j \rightarrow_h f_k C_j$ and $C_j\sigma \rightarrow_h A'_j$. By Lemma 4.2, $C_j\sigma\sigma' = C_j\delta\sigma$, and $C_j\sigma\sigma' \rightarrow_h A'_j\sigma'$, then $C_j\delta\sigma$ is not strongly h -normalizable. We have $A \rightarrow_h C_j\delta$, then $\eta(C_j\delta) < \eta(A)$ and, by the induction hypothesis, there exists $s_i B \preceq C_j\delta \preceq A$ such that $B\sigma$ is strongly h -normalizable and $[B_1, \dots, B\sigma, \dots, B_n]$ is not strongly h -normalizable. □

We define now a norm on colored trees.

Definition 4.10 *The norm $\|\cdot\|$ of a colored tree is defined as follows.*

- $\|\bullet\| = 0$.
- $\|f_i A\| = 0$ if $s_i \notin FS(A)$ and $\|f_i A\| = \max_{s_i B \sqsubseteq A} \|B\| + 1$ if $s_i \in FS(A)$.
- $\|s_i A\| = 0$.
- $\|[A_1, \dots, A_n]\| = \sum_{i=1}^n \|A_i\|$.

The first intuition about the defined norm is that it does not change after “adding branches” to the colored trees.

Lemma 4.11 *Let $A, B_1, \dots, B_n \in \mathbb{T}$ and $\sigma = \langle (s_i := [B_1, \dots, *, \dots, B_n]) \rangle$.*

We have $\|A\sigma\| = \|A\|$.

Proof By induction on A .

- If $A = f_i A'$, then $A\sigma = f_i A'\sigma$. If $s_i \in FS(A)$, then $s_i \in FS(A')$ and $s_i B \sqsubseteq A'\sigma$ if and only if $s_i B \sqsubseteq A$. We have $\|A\sigma\| = \|f_i A'\sigma\| = \max_{s_i B \sqsubseteq A'\sigma} \|B\| + 1 = \max_{s_i B \sqsubseteq A} \|B\| + 1 = \|A\|$. Otherwise, $s_i \notin FS(A)$ and $\|A\| = \|A'\sigma\| = 0$.
- If $A = s_i A'$ with $i \notin \{i_1, \dots, i_m\}$, then $A\sigma = s_i B$, where $B = A'\sigma$. So, $\|A\| = \|A\sigma\| = 0$.

- If $A = s_i A'$ with $i \in \{i_1, \dots, i_m\}$, then $B = [B_1, \dots, A' \sigma, \dots, B_n]$. So, $\|A\| = \|A \sigma\| = 0$.
- If $A = [A_1, \dots, A_k]$, then $A \sigma = [A_1 \sigma, \dots, A_k \sigma]$ and $\|A \sigma\| = \sum_{i=1}^k \|A_i \sigma\|$. By the induction hypothesis, $\|A_i \sigma\| = \|A_i\|$, so $\|A \sigma\| = \sum_{i=1}^k \|A_i\| = \|A\|$.

□

The following lemma is quite intuitive.

Lemma 4.12 *Let $A, B_1, \dots, B_n \in \mathbb{T}$ and $\sigma = \langle (s_i) := [B_1, \dots, *, \dots, B_n] \rangle$. If $s_i A' \sqsubseteq A$, then $\|f_i A\| > \|A' \sigma\|$.*

Proof If $s_i A' \sqsubseteq A$, then, by Lemma 4.11, $\|A' \sigma\| = \|A'\|$, thus $\|f_i A\| = \max_{s_i B \sqsubseteq A} \|B\| + 1 > \|A'\| = \|A' \sigma\|$. □

We will now prove that after an h -reduction on the head of a colored tree, the norm cannot increase.

Lemma 4.13 *Let $A, B_1, \dots, B_n \in \mathbb{T}$ and $\sigma = \langle (s_i) := [B_1, \dots, *, \dots, B_n] \rangle$. We have $\|[B_1, \dots, f_i A, \dots, B_n]\| \leq \|f_i A \sigma\|$.*

Proof We have $s_i B \sqsubseteq A$ if and only if $s_i [B_1, \dots, B, \dots, B_n] \sqsubseteq A \sigma$. If $s_i \notin FS(A)$, then $\|f_i A \sigma\| = 0$ and so $\|[B_1, \dots, f_i A, \dots, B_n]\| \leq \|f_i A \sigma\|$. Otherwise, we have $\|f_i A \sigma\| = \max_{s_i C \sqsubseteq A \sigma} \|C\| + 1 = \max_{s_i B \sqsubseteq A} \|[B_1, \dots, B, \dots, B_n]\| + 1 = \max_{s_i B \sqsubseteq A} \|B\| + 1 + \sum_{i=1}^n \|B_i\| = \sum_{i=1}^n \|B_i\| + \max_{s_i B \sqsubseteq A} \|B\| + 1 = \|[B_1, \dots, f_i A, \dots, B_n]\|$. □

We observe that any sequence of h -reductions on a given colored tree produces a non-increasing sequence of the norm.

Lemma 4.14 *If $A, B \in \mathbb{T}$ and $A \rightarrow_h B$, then $\|A\| \geq \|B\|$.*

Proof Without loss of generality, we may suppose that $A \rightarrow_h B$. We proceed by induction on $(\eta(A), |A|)$ using the lexicographic order.

- If $A = s_i A'$, then $B = s_i B'$ with $A' \rightarrow_h B'$, thus $\|A\| = \|B\| = 0$.
- If $A = f_i A'$, then $B = f_i B'$. If $s_i \notin FS(A')$, then $s_i \notin FS(B')$ and so $\|A\| = \|B\| = 0$. Otherwise, let $s_i C \sqsubseteq B$, then there exists $s_i D \sqsubseteq A$ such that $C = D \sigma$ for some σ . By Lemma 4.11, $\|C\| = \|D\|$, thus, $\|A\| = \max_{s_i A' \sqsubseteq A} \|A'\| + 1 = \max_{s_i B' \sqsubseteq B} \|B'\| + 1 = \|B\|$.
- Let $A = [A_1, \dots, A_n]$. If $A \rightarrow_h [A_1, \dots, A'_i, \dots, A_n] = B$ with $A_i \rightarrow_h A'_i$, then, by the induction hypothesis, $\|A_i\| \geq \|A'_i\|$, thus $\|A\| = \sum_{k=1}^n \|A_k\| \geq \sum_{k=1}^n \|A'_k\| = \|[A_1, \dots, A'_i, \dots, A_n]\| = \|B\|$. Otherwise, $A \rightarrow_h f_i A_j \sigma = B$ with $A_j = f_i A'_j$ and $\sigma = \langle (s_i) := [A_1, \dots, *, \dots, A_n] \rangle$. By Lemma 4.13, $\|A\| \geq \|f_i A_j \sigma\| = \|B\|$.

□

The colored tree $[A_1, \dots, A_n]$ will be strongly h -normalizable if all A_i are strongly h -normalizable. The following theorem, prove by contradiction, prevents us from devising an algorithm for strong h -normalization.

Theorem 4.15 *If A_1, \dots, A_n are strongly h -normalizable, then $[A_1, \dots, A_n]$ is strongly h -normalizable.*

Proof Suppose, to the contrary, that there exist A_1, \dots, A_n which are strongly h -normalizable, but $[A_1, \dots, A_n]$ is not strongly h -normalizable. Let A_1, \dots, A_n be chosen as the smallest counter-example with respect to the pair

$\left(\|[A_1, \dots, A_n]\|, \sum_{i=1}^n \eta(A_i) \right)$ using the lexicographic order. As $[A_1, \dots, A_n]$ is not strongly h -normalizable, by Lemma 4.8, there exists $1 \leq i \leq n$ such that $A_i \rightarrow_h f_j B_i$ and $B_i \sigma$ is not strongly h -normalizable, where $\sigma = \langle (s_j) := [A_1, \dots, *, \dots, A_n] \rangle$. By Lemma 4.14, $\|B_i\| \leq \|A_i\|$. Now, $[A_1, \dots, A_n] \rightarrow_h [A_1, \dots, f_j B_i, \dots, A_n]$, then if $A_i \neq f_j B_i$, we obtain $\eta(B_i) < \eta(A_i)$ so $[A_1, \dots, A_{i-1}, f_j B_i, A_{i+1}, \dots, A_n]$ is a counter-example smaller than $[A_1, \dots, A_n]$, a contradiction. Consequently, $A_i = f_j B_i$. As $B_i \sigma$ is not strongly h -normalizable, by Lemma 4.9, there exists $s_j C_i \preceq B_i$ such that $C_i \sigma$ is strongly h -normalizable and $[A_1, \dots, C_i \sigma, \dots, A_n]$ is not strongly h -normalizable. Let B'_i be the colored tree such that $B_i \rightarrow_h B'_i$ and $s_j C_i \sqsubseteq B'_i$, we have, by Lemma 4.12, $\|f_j B_i\| > \|C_i \sigma\|$. Thus, $[A_1, \dots, C_i \sigma, \dots, A_n]$ is not strongly h -normalizable and smaller than $[A_1, \dots, A_n]$, a contradiction. The result follows. \square

After proving all preceding lemmas and theorems, the main result is established below.

Theorem 4.16 *Any colored tree A is strongly h -normalizable.*

Proof By induction on A .

- If $A = f_i A'$, then, by the induction hypothesis, A' is strongly h -normalizable, so A is strongly h -normalizable.
- If $A = s_i A'$, then, by the induction hypothesis, A' is strongly h -normalizable, so A is strongly h -normalizable.
- If $A = [A_1, \dots, A_n]$, then, by the induction hypothesis, A_1, \dots, A_n are strongly h -normalizable, so, by theorem 4.15, A is strongly h -normalizable. \square

This strong normalization result states that any h -reduction of a colored tree terminates without specifying how to perform it, so there is no need to define a normalization algorithm. We reiterate that the presentation of the weak normalization proof in Section 3 aims to familiarize the reader with this type of proof technique, which will be used a second time in Section 5.

5 A weak normalization result for several reduction rules

We will propose additional reduction rules on colored trees to simplify the use of h -reduction. We will examine the succession of father and son nodes. First, we will explain the intuitive ideas behind these reductions before defining them formally. Let $A \in \mathbb{T}$.

- If there are two consecutive fathers f_i and f_j in A , the father f_j will never undergo an h -reduction. To simplify the reduction process, we will delete the father f_j along with its sons from A . We will call this the ff -reduction.

- If a father f_i has a unique son that comes directly below it, any h -reduction involving f_i will not change the shape of A . Therefore, we will delete the father along with its son. We will call this the sf -reduction.
- If a father f_j appears before a son s_i , the father f_i will only undergo an h -reduction if the father of s_i undergoes an h -reduction that places f_j in an h -redex. Consequently, the h -reduction via f_j will be equivalent to that via f_i but will use the son s_j instead of s_i . Therefore, we will delete the father f_j and substitute its sons with the sons s_i . We will call this the fs -reduction.

We recall that the goal of these three rules is to eliminate undesirable phenomena for our main rule, the h -reduction. We cannot achieve this by restricting the syntax to colored trees, as we could create such phenomena in an h -reduction. The intuition suggests that these elementary rules cannot affect the normalization results already obtained. We will show that this is actually false, as we can construct a colored tree that is not strongly normalizable.

In order to introduce these simplification rules, we define, for any $A \in \mathbb{T}$ and any $i \in \mathbb{N}$, the colored tree $\lceil A \rceil^i$ to be the colored tree obtained by erasing all free sons s_i from A .

Definition 5.1 *Let $A \in \mathbb{T}$ and $i \in \mathbb{N}$. The colored tree $\lceil A \rceil^i$ is defined as follows.*

- If $A = \bullet$, then $\lceil A \rceil^i = \bullet$.
- If $A = f_j B$, then $\lceil A \rceil^i = f_j \lceil B \rceil^i$.
- If $A = s_j B$ with $j \neq i$, then $\lceil A \rceil^i = s_j \lceil B \rceil^i$.
- If $A = s_i B$, then $\lceil A \rceil^i = \lceil B \rceil^i$.
- If $A = [A_1, \dots, A_m]$, then $\lceil A \rceil^i = [\lceil A_1 \rceil^i, \dots, \lceil A_m \rceil^i]$.

The following two definitions introduce the three new simplification rules.

Definition 5.2

1. An ff -redex is a colored tree under the form $f_i f_j B$. Its ff -reduced is $f_i \lceil B \rceil^j$.
2. An fs -redex is a colored tree under the form $f_i s_i B$ with $s_i \notin FS(B)$. Its fs -reduced is B .
3. An sf -redex is a colored tree under the form $s_i f_j B$. Its sf -reduced is $B[s_j := s_i]$.

Definition 5.3 *Let $A, B \in \mathbb{T}$.*

1. We define $A \rightarrow_{ff} B$, by induction on A , as follows.
 - If $A = \bullet$, then it is impossible.
 - If $A = s_i A_1$, then $B = s_i B_1$ and $A_1 \rightarrow_{ff} B_1$.
 - If $A = [A_1, \dots, A_n]$, then $B = [A_1, \dots, B_k, \dots, A_n]$ and $A_k \rightarrow_{ff} B_k$.
 - If $A = f_i A_1$, then there are two cases.
 - $B = f_i B_1$ and $A_1 \rightarrow_{ff} B_1$.
 - $A_1 = f_j A_2$ and $B = f_i \lceil A_2 \rceil^j$.
2. We define $A \rightarrow_{fs} B$, by induction on A , as follows.

- If $A = \bullet$, then it is impossible.
- If $A = s_i A_1$, then $B = s_i B_1$ and $A_1 \rightarrow_{fs} B_1$.
- If $A = [A_1, \dots, A_n]$, then $B = [A_1, \dots, B_k, \dots, A_n]$ and $A_k \rightarrow_{fs} B_k$.
- If $A = f_i A_1$, then there are two cases.
 - $B = f_i B_1$ and $A_1 \rightarrow_{fs} B_1$.
 - $A_1 = s_i A_2$ with $s_i \notin FS(A_2)$ and $B = A_2$.

3. We define $A \rightarrow_{sf} B$, by induction on A , as follows.

- If $A = \bullet$, then it is impossible.
- If $A = f_i A_1$, then $B = f_i B_1$ and $A_1 \rightarrow_{sf} B_1$.
- If $A = [A_1, \dots, A_n]$, then $B = [A_1, \dots, B_k, \dots, A_n]$ and $A_k \rightarrow_{sf} B_k$.
- If $A = s_i A_1$, then there are two cases.
 - $B = s_i B_1$ and $A_1 \rightarrow_{sf} B_1$.
 - $A_1 = f_j A_2$ and $B = A_2[s_j := s_i]$.

As in the case of the h -reduction rule, we can define normalization properties with respect to each reduction as well as to the combined ones.

Definition 5.4 Let $R \subseteq \{h, ff, fs, sf\}$.

1. A colored tree is said to be R -normal if it contains no r -redex for all $r \in R$.
2. If $A, B \in \mathbb{T}$, we write $A \rightarrow_R B$ if there exist a sequence $A_1, \dots, A_n \in \mathbb{T}$ such that $A \rightarrow_{r_1} A_1 \rightarrow_{r_2} A_2 \rightarrow_{r_3} \dots \rightarrow_{r_n} A_n = B$ where $r_i \in R$ for all $1 \leq i \leq n$.
3. A colored tree A is said to be weakly R -normalizable if there exists an R -normal colored tree B such that $A \rightarrow_R B$.
4. A colored tree A is said to be strongly R -normalizable if any sequence of R -reductions of A ends.

Example 5.5 The following counter-example, negates that every colored tree is strongly $\{h, ff, fs, sf\}$ -normalizable.

$$\begin{aligned}
C &= [f_1 f_2 s_2 s_2 \bullet, f_3 s_3 s_3 \bullet] \rightarrow_h f_3 s_3 [f_1 f_2 s_2 s_2 \bullet, s_3 [f_4 f_5 s_5 s_5 \bullet, \bullet]] \\
&\rightarrow_h f_3 s_3 [f_1 f_2 s_2 s_2 \bullet, s_3 f_4 f_5 s_5 s_5 \bullet] \\
&\rightarrow_{sf} f_3 s_3 [f_1 f_2 s_2 s_2 \bullet, f_5 s_5 s_5 \bullet] \\
&\rightarrow_{fs} A.
\end{aligned}$$

Figure 7 shows in the tree representation of C how the reduction leads to a loop. Note that we used f_4 and f_5 instead of f_1 and f_2 under the second son of f_3 . As mentioned earlier, the numbering has no intrinsic meaning; it is given to highlight the relationships between sons and fathers. However, this example prevents us from considering a proof of strong $\{h, ff, fs, sf\}$ -normalization for any colored tree. We observe that if we reduce the h -redex in C via f_1 , we immediately obtain the $\{h, ff, fs, sf\}$ -normal colored tree $f_1 f_2 s_2 s_2 \bullet$. Therefore, we will proceed to prove weak $\{h, ff, fs, sf\}$ -normalization for any colored tree. Note also that this phenomenon of non-normalization does not arise from having two consecutive sons. Indeed, we can easily verify that the colored tree $D = [f_1 f_2 s_2 [\bullet, s_2 \bullet], f_3 s_3 [\bullet, s_3 \bullet]]$ is also non-normalizable.

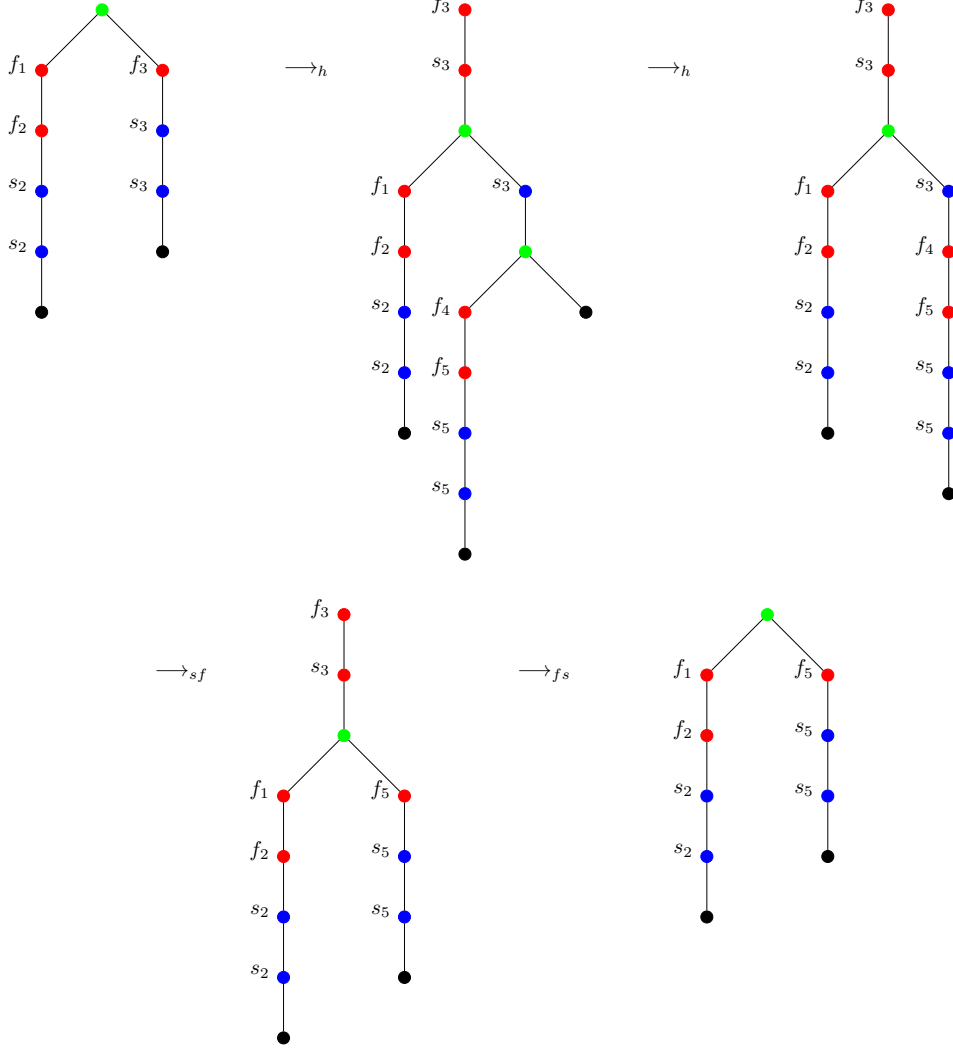


Figure 7: A counter-example for the $\{h, ff, fs, sf\}$ -strong normalization property

For abbreviation, let $R_1 = \{h, ff, sf\}$ and $R_2 = \{h, ff, fs, sf\}$. First, let's define a (k, n) -super good colored tree.

Definition 5.6 *Let $n, k \in \mathbb{N}$ such that $0 \leq k \leq n$. A colored tree is said to be (k, n) -super good if it is $\{ff, sf\}$ -normal and all its h -redex are of order (k, n) and are protected by a son. By convention, a colored tree is said to be $(0, n)$ -good if it is R_1 -normal.*

We will start with the weak R_1 -normalization property, and then we will prove that any colored tree is fs -normalizable; consequently, any tree will be R_2 -normalizable. The following lemmas, which will be useful in the proof, are similar to those used in the proof of the weak h -normalization property; however, here, due to the new reductions, the proofs will be simpler.

Lemma 5.7 *Let A, B_1, \dots, B_n be colored R_1 -normal trees such that k colored tree of B_1, \dots, B_n starts by a father and $\sigma = \langle (s_{i_1}, \dots, s_{i_m}) := [B_1, \dots, *, \dots, B_n] \rangle$. Then $A\sigma$ is $(k, n+1)$ -super good.*

Proof By induction on A .

- If $A = f_i A_1$, then $A\sigma = f_i A_1\sigma$ and, by the induction hypothesis, $A_1\sigma$ is $(k, n+1)$ -super good, thus $A\sigma$ is $(k, n+1)$ -super good.
- If $A = s_i A_1$ with $i \notin \{i_1, \dots, i_m\}$, then $A\sigma = s_i A_1\sigma$. By the induction hypothesis, $A_1\sigma$ is $(k, n+1)$ -super good, thus $A\sigma$ is $(k, n+1)$ -super good.
- If $A = s_i A_1$ with $i \in \{i_1, \dots, i_m\}$, then $A\sigma = s_i[B_1, \dots, A_1\sigma, \dots, B_n]$. By the induction hypothesis, $A_1\sigma$ is $(k, n+1)$ -super good and A_1 doesn't start by a father since A is *sf*-normal, thus $A\sigma$ is $(k, n+1)$ -super good.
- If $A = [A_1, \dots, A_l]$, then $A\sigma = [A_1\sigma, \dots, A_l\sigma]$. By the induction hypothesis, $A_i\sigma$ is $(k, n+1)$ -super good for all $1 \leq i \leq l$. Clearly, A_i doesn't start by a father for all $1 \leq i \leq l$ and so it is the case for $A_i\sigma$, thus A is $(k, n+1)$ -super good.

□

The following lemma explains how to R_1 -normalize a super good colored tree.

Lemma 5.8 *A (k, m) -super good colored tree A is weakly R_1 -normalizable and if $A \rightarrow_{R_1} B$ with B is R_1 -normal, then $B = f_i C$ for some colored tree C if and only if $A = f_i D$ for some colored tree D .*

Proof By induction on $(k, |A|)$ using the lexicographic order.

- If $A = f_i A_1$, then A_1 is (k, m) -super good and $|A_1| < |A|$, therefore, by the induction hypothesis, $A_1 \rightarrow_{R_1} B_1$ with B_1 is R_1 -normal. But A is *ff*-normal, so A_1 doesn't start by a father, then, by the induction hypothesis, B_1 doesn't start by a father, thus, $A \rightarrow_{R_1} B = f_i B_1$ with B is R_1 -normal.
- If $A = s_i A_1$, then A_1 is (k, m) -super good and $|A_1| < |A|$, then, by the induction hypothesis, $A_1 \rightarrow_{R_1} B_1$ with B_1 is R_1 -normal. But A is *sf*-normal, so A_1 doesn't start by a father, then, by the induction hypothesis, B_1 doesn't start by a father, thus $A \rightarrow_{R_1} B = s_i B_1$ with B is R_1 -normal.
- If $A = s_j [A_1, \dots, A_m]$ with k of the A_i start by a father, then or all $1 \leq i \leq m$, A_i is (k, m) -super good with $|A_i| < |A|$ so, by the induction hypothesis, $A_i \rightarrow_{R_1} B_i$ with B_i is R_1 -normal and B_i starts by a father if and only if A_i starts by a father. Let $1 \leq p \leq m$ such that $B_p = f_p C_p$, then $A \rightarrow_{R_1} s_j [B_1, \dots, B_m] \rightarrow_h s_j f_p C_p \sigma \rightarrow_{sf} C_p \sigma [s_p := s_j] = B$ where $\sigma = \langle (s_p) := [B_1, \dots, *, \dots, B_m] \rangle$. By Lemma 5.7, $C_p \sigma$ is $(k-1, m)$ -super good, then, by the induction hypothesis, $C_p \sigma \rightarrow_{R_1} D_p$ with D_p is R_1 -normal. Since $C_p \sigma$ doesn't start by a father, then D_p does not start by a father. Hence, $A \rightarrow_{R_1} B$ where B is R_1 -normal and B doesn't start by a father.
- If $A = [A_1, \dots, A_m]$ and A_i doesn't start by a father for all $1 \leq i \leq m$, then A_i is (k, m) -super good with $|A_i| < |A|$, so, by the induction hypothesis, $A_i \rightarrow_{R_1} B_i$ with B_i is R_1 -normal and doesn't start by a father. Therefore, $A \rightarrow_{R_1} [B_1, \dots, B_m] = B$ with B is R_1 -normal and B does not start by a father.

□

The following lemma is intuitive and simple to prove.

Lemma 5.9 *Let $A \in \mathbb{T}$ and $i \in \mathbb{N}$. If A is R_1 -normal, then $[A]^i$ is R_1 -normal and $[A]^i$ does not start by a father whenever A does not start by a father.*

Proof By induction on A .

- If $A = f_j A_1$, then $[A]^i = f_j [A_1]^i$ and, by the induction hypothesis, $[A_1]^i$ is R_1 -normal and does not start by a father, thus $[A]^i$ is R_1 -normal.
- If $A = s_j A_1$ with $j \neq i$, then $[A]^i = s_j [A_1]^i$ and, by the induction hypothesis, $[A_1]^i$ is R_1 -normal and doesn't start by a father, thus $[A]^i$ is R_1 -normal.
- If $A = s_i A_1$, then $[A]^i = [A_1]^i$, and, by the induction hypothesis, $[A_1]^i$ is R_1 -normal and does not start by a father.
- If $A = [A_1, \dots, A_n]$, then $[A]^i = [[A_1]^i, \dots, [A_n]^i]$ and, by the induction hypothesis, $[A_k]^i$ is R_1 -normal and does not start by a father for all $1 \leq k \leq n$, thus, $[A]^i$ is R_1 -normal and does not start by a father. □

We now achieve the following normalization result.

Theorem 5.10 *Every colored tree A is weakly R_1 -normalizable.*

Proof By induction on A .

- If $A = f_i A_1$, then, by the induction hypothesis, $A_1 \rightarrow_{R_1} B_1$ with B_1 is R_1 -normal. If B_1 does not start by a father, then $A \rightarrow_{R_1} f_i B_1$ and $f_i B_1$ is R_1 -normal. Otherwise, let $B_1 = f_j C_1$ with C_1 R_1 -normal does not start by a father, then $A \rightarrow_{R_1} f_i B_1 \rightarrow_{ff} f_i [C_1]^j$ and, by Lemma 5.9, $f_i [C_1]^j$ is R_1 -normal.
- If $A = s_i A_1$, then, by the induction hypothesis, $A_1 \rightarrow_{R_1} B_1$ with B_1 is R_1 -normal. If B_1 does not start by a father, then $A \rightarrow_{R_1} s_i B_1$ and $s_i B_1$ is R_1 -normal. Otherwise, let $B_1 = f_j C_1$ with C_1 R_1 -normal, then $A \rightarrow_{R_1} s_i B_1 \rightarrow_{ff} C_1 [s_j := s_i]$ and $C_1 [s_j := s_i]$ is R_1 -normal.
- If $A = [A_1, \dots, A_m]$, then, by the induction hypothesis, for all $1 \leq i \leq m$, $A_i \rightarrow_{R_1} B_i$ with B_i is R_1 -normal, thus $A \rightarrow_{R_1} [B_1, \dots, B_m] = B$. If all B_i does not start by a father, then B is R_1 -normal. Otherwise, let $1 \leq j \leq m$ such that $B_j = f_j C_j$. We have $B \rightarrow_h f_j C_j \sigma$, where C_j does not start by a father and $\sigma = \langle (s_j) := [B_1, \dots, *, \dots, B_m] \rangle$. By Lemma 5.7, $C_j \sigma$ is (k, m) -super good, for some k , and by Lemma 5.8, it is weakly R_1 -normalizable and its R_1 -normal form does not start by a father, thus $f_j C_j \sigma$ is weakly R_1 -normalizable. □

The following two lemmas examine the fs -reduction to subsequently achieve our desired result.

Lemma 5.11 *Let A be an R_1 -normal colored tree. If $A \rightarrow_{fs} B$, then B is R_1 -normal with B doesn't start by a father whenever A does not start by a father.*

Proof By induction on A .

- If $A = f_i A_1$, $B = f_i B_1$ and $A_1 \rightarrow_{fs} B_1$, then, by the induction hypothesis, B_1 is R_1 -normal and does not start by a father, thus B is R_1 -normal.
- If $A = f_i s_i B$ with $s_i \notin FS(B)$, then B is R_1 -normal.
- If $A = s_i A_1$, $B = s_i B_1$ and $A_1 \rightarrow_{fs} B_1$, then, by the induction hypothesis, B_1 is R_1 -normal and does not start by a father, thus B is R_1 -normal, thus B is R_1 -normal.
- If $A = [A_1, \dots, A_m]$, $B = [A_1, \dots, A'_i, \dots, A_m]$ and $A_i \rightarrow_{fs} A'_i$, then, by the induction hypothesis, A'_i is R_1 -normal and does not start by a father, thus B is R_1 -normal. □

Lemma 5.12 *Every R_1 -normal colored tree A is weakly R_2 -normalizable.*

Proof By induction on A . If A is not R_2 -normal, then A contains an fs -redex. By Lemma 5.11, $A \rightarrow_{fs} B$ with B is R_1 -normal and $|B| < |A|$, so, by the induction hypothesis, B is weakly R_2 -normalizable. The result follows. \square

We can now deduce our result.

Theorem 5.13 *Every colored tree A is weakly R_2 -normalizable.*

Proof By Theorem 5.10 and Lemma 5.12. \square

Although we used the method from Section 3 to obtain our result of weak normalization for a broader set of reduction rules, we find that the proof is simpler. This is mainly due to the fact that there are fewer normal forms, and thus fewer cases to consider. As in the weak h -normalization property, the proof is constructive, so we may write a similar algorithm.

We observe that the ff -reduction is not utilized in Example 5.5. This raises the question of whether the set $\{h, sf, fs\}$ is weakly normalizable. We have identified a colored tree that is not weakly $\{h, ff, fs\}$ -normalizable; however, we have chosen not to include this example in the paper to maintain clarity and avoid overwhelming the reader. It is quite striking that by considering all the rules, we can still achieve a weak normalization result.

We could also consider a reduction rule concerning the succession of two sons (either with the same or different indices). We could introduce an ss -reduction, but without a solid motivation to simplify the application of our main h -rrreduction. In other types of calculus (such as that in Subsection 6.1), this type of rule might be justified. We prefer not to include it in this paper.

6 Two applications of our results

In this section, we will first present an application of our results to Parigot's μ -calculus, which originally motivated our paper. Subsequently, we will propose a generalization to other types of binary relations, thereby extending our results to more complex structures.

6.1 Parigot's μ -calculus

The $\lambda\mu$ -calculus was introduced by Parigot ([7]) in 1990 to capture the algorithmic content of proofs in classical logic presented in a natural deduction style system. The resulting calculus includes the λ -calculus, which allows for the representation of all computable functions. It also encompasses techniques from the Exit programming style, which enables exiting a program in case of an error and displaying a message. The computation or program execution rules correspond to cut eliminations (reductions on proofs), which simplify certain parts of proofs. Here, we present only the μ -calculus part, which corresponds to reasoning by contradiction in proofs. Towards the end of this subsection, we will see that the execution rules of this calculus exactly correspond to the rules presented in the preceding sections on colored trees, which is surprising given that the motivations for the two calculi are completely different.

Definition 6.1 *We consider two new countable sets of variable $\mathcal{V} = \{x, y, z, \dots\}$ and $\mathcal{W} = \{a_i \mid i \in \mathbb{N}\}$. The set of μ -terms \mathbb{P} of the μ -calculus and the function Fv is defined as follows.*

- If $x \in \mathcal{V}$, then $x \in \mathbb{P}$ and $Fv(x) = \emptyset$.
- If $M \in \mathbb{P}$ and $a_i \in \mathcal{W}$, then $\mu a_i.M \in \mathbb{P}$ and $Fv(\mu a_i.M) = Fv(M) \setminus \{a_i\}$,
- If $M \in \mathbb{P}$ and $a_i \in \mathcal{W}$, then $[a_i]M \in \mathbb{P}$ and $Fv([a_i]M) = Fv(M) \cup \{a_i\}$,
- If $M_1, M_2 \in \mathbb{P}$, then $(M_1)M_2 \in \mathbb{P}$ and $Fv((M_1)M_2) = Fv(M_1) \cup Fv(M_2)$.

In the μ -calculus, similar to colored trees, the symbol μ is considered a binder for the variables a_i . The set $Fv(M)$ represents the set of free a_i in M . We read the term $(M_1)M_2$ as M_1 applied to M_2 , so M_1 is considered as a function and M_2 as an argument.

To define reductions on μ -terms, we first need to define two kinds of substitutions and a translation.

Definition 6.2

1. A μ -substitution is an expression of the form $\sigma = [a_i :=_s N]$ where $s \in \{l, r\}$, $a_i \in \mathcal{W}$ and N is a μ -term.
2. Let $M, N \in \mathbb{P}$ and $\sigma = [a_i :=_s N]$. We define by induction the μ -term $M\sigma$. We can assume that the free variables of the μ -term N are not linked to a μ -abstraction in the μ -term M .
 - If $M = x$, then $M\sigma = x$.
 - If $M = (M_1)M_2$, then $M\sigma = (M_1\sigma)M_2\sigma$.
 - If $M = \mu a_j.M'$, then $M\sigma = \mu a_j.M'\sigma$.
 - If $M = [a_j]M'$ and $j \neq i$, then $M\sigma = [a_j]M'\sigma$.
 - If $M = [a_i]M'$ and $s = r$, then $M\sigma = [a_i](M'\sigma)N$.
 - If $M = [a_i]M'$ and $s = l$, then $M\sigma = [a_i](N)M'\sigma$.
3. Let $M \in \mathbb{P}$ and $i \in \mathbb{N}$. We define the i -translation $\lceil M \rceil^i$ of M by induction on M .
 - If $M = x$, then $\lceil M \rceil^i = x$.
 - If $M = (P)Q$, then $\lceil M \rceil^i = (\lceil P \rceil^i)\lceil Q \rceil^i$.
 - If $M = \mu a_j.N$, then $\lceil M \rceil^i = \mu a_j.\lceil N \rceil^i$.
 - If $M = [a_j]N$ and $j \neq i$, then $\lceil M \rceil^i = [a_j]\lceil N \rceil^i$.
 - If $M = [a_i]N$, then $\lceil M \rceil^i = \lceil N \rceil^i$.

In effect, $\lceil M \rceil^i$ is the result of replacing, starting from the innermost ones, every subterm $[a_i]N$ in M by N .

We now define the redexes of a μ -term and how to reduce them.

Definition 6.3

1. A μ -redex is a μ -term of the form $(\mu a_i.M)N$ and we call $\mu a_i.M[a_i :=_r N]$ its contractum.
2. A μ' -redex is a μ -term of the form $(N)\mu a_i.M$ and we call $\mu a_i.M[a_i :=_l N]$ its contractum.
3. A ρ -redex is a μ -term of the form $[a_j]\mu a_i.M$ and we call $M[a_i := a_j]$ its contractum, where $M[a_i := a_j]$ represents the substitution in M of each a_i by a_j .

4. An ε -redex is a μ -term of the form $\mu a_i.\mu a_j.M$ and we call $\mu a_i.[M]^j$ its contractum.
5. A θ -redex is a μ -term of the form $\mu a_i.[a_i]M$ where $a_i \notin Fv(M)$ and we call M its contractum.

Finally, we define the notions of reductions and normalizations.

Definition 6.4 Let $R \subseteq \{\mu, \mu', \rho, \theta, \varepsilon\}$.

1. If $M, M' \in \mathbb{P}$ and $r \in R$, we write $M \rightarrow_r M'$, if M' is obtained from M by replacing a r -redex in M by its contractum. The reductions (on the redexes) take the following forms (the θ -redex has an additional condition).

$$\begin{array}{lll}
(\mu a_i.M)N & \rightarrow_\mu & \mu a_i.M[a_i :=_r N] \\
(N)\mu a_i.M & \rightarrow_{\mu'} & \mu a_i.M[a_i :=_l N] \\
[a_j]\mu a_i.M & \rightarrow_\rho & M[a_i := a_j] \\
\mu a_i.\mu a_j.M & \rightarrow_\varepsilon & \mu a_i.[M]^j \\
\mu a_i.[a_i]M & \rightarrow_\theta & M
\end{array}$$

2. If $M, M' \in \mathbb{P}$, we write $M \twoheadrightarrow_R M'$ if there exist a sequence $M_1, \dots, M_n \in \mathbb{P}$ such that $M \rightarrow_{r_1} M_1 \rightarrow_{r_2} M_2 \rightarrow_{r_3} \dots \rightarrow_{r_n} M_n = M'$ where $r_i \in R$ for all $1 \leq i \leq n$.
3. A μ -term is said to be R -normal form if it does not contain r -redex for $r \in R$.
4. A μ -term M is said to be weakly R -normalizable if there exists an R -normal form M' such that $M \twoheadrightarrow_R M'$.
5. A μ -term M is said to be strongly R -normalizable, if there exists no infinite R -reduction paths starting from M .

First, we note that the nodes \bullet do not appear in the normalization results we presented in the previous sections. Therefore, we can replace them with the countable set $\mathcal{V} = \{x, y, z, \dots\}$. We will restrict ourselves to the cases where the colored trees contain only subtrees of the form $[A_1, \dots, A_n]$ with $n = 2$. Let \mathbb{T}_2 denote the set of colored trees thus obtained. We will define a bijective translation from \mathbb{T}_2 to \mathbb{P} . Lemma 6.7 shows that the reduction rules we presented in the previous sections correspond to those of the μ -calculus, thereby automatically providing normalization results.

Definition 6.5 We define the translation $\mathcal{M} : \mathbb{T}_2 \longrightarrow \mathbb{P}$ by induction as follows.

- $\mathcal{M}(x) = x$,
- $\mathcal{M}(f_i A) = \mu a_i.\mathcal{M}(A)$,
- $\mathcal{M}(s_i A) = [a_i]\mathcal{M}(A)$,
- $\mathcal{M}([A_1, A_2]) = (\mathcal{M}(A_1))\mathcal{M}(A_2)$.

It is easy to see that this application is bijective and it is very easy to define the inverse bijection.

The following lemma presents the correspondence between the reduction rules of \mathbb{T}_2 and those of \mathbb{P} . This correspondence is a true bijection and works in both directions. We will not provide a proof of this result, as our goal is merely to present this application in a completely different domain from our own. Note that the inheritance rule h transforms into two reductions in the μ -calculus. Indeed, the colored tree $[f_i A, B]$ transforms into a μ -redex, and the colored tree $[B, f_i A]$ transforms into a μ' -redex.

Example 6.6 Let $A = f_1 s_7 [f_2 f_4 [f_6 s_6 x, s_2 s_2 s_4 y], s_1 f_5 s_5 [y, z]]$, we have $\mathcal{M}(A) = \mu a_1 \cdot [f] (\mu a_2 \cdot \mu a_3 \cdot (\mu a_5 \cdot [a_5] x) [a_2] [a_2] [a_3] y) [a_1] \mu a_4 \cdot [a_4] (y) z$.

Lemma 6.7 Let $A, B \in \mathbb{T}_2$.

1. $A \rightarrow_h B$ if and only if $\mathcal{M}(A) \rightarrow_\mu \mathcal{M}(B)$ or $\mathcal{M}(A) \rightarrow_{\mu'} \mathcal{M}(B)$.
2. $A \rightarrow_{sf} B$ if and only if $\mathcal{M}(A) \rightarrow_\rho \mathcal{M}(B)$.
3. $A \rightarrow_{ff} B$ if and only if $\mathcal{M}(A) \rightarrow_\varepsilon \mathcal{M}(B)$.
4. $A \rightarrow_{fs} B$ if and only if $\mathcal{M}(A) \rightarrow_\theta \mathcal{M}(B)$.

Since the translation transforms a reduction in one calculus into a reduction in the other, the normalization results obtained in the previous sections are preserved in the μ -calculus.

Theorem 6.8

1. Every μ -term is strongly $\{\mu, \mu'\}$ -normalizable.
2. Every μ -term is weakly $\{\mu, \mu', \rho, \varepsilon, \theta\}$ -normalizable.

Proof It suffices to apply Lemma 6.7 and Theorems 4.16 and 5.13. \square

The second author of this paper, along with another collaborator, presented direct proofs of Theorem 6.17 in the μ -calculus in [1, 2]. The proofs presented in the preceding sections generalize and adapt these proofs to a more general framework where trees can branch arbitrarily. Our demonstrations not only enhance understanding of the earlier work but also demonstrate that to obtain normalization results in the μ -calculus, the reduction rules μ and μ' play equivalent roles, despite their differing origins and properties.

6.2 Reductions on digraphs

If the previous subsection presents an application of the preceding sections to a very particular case, in this subsection, we will provide a generalization to a much broader class of binary relations. We will see how the proofs introduced in the earlier sections can be adapted to this more general case by simply omitting certain details of these complex structures. The intuitions underlying this general case stem from the modular decomposition of a binary relation. Indeed, this result allows us to define a notion of reduction by coloring certain modules with two distinct colors and establishing father-son relationships between them, akin to our work on trees. We will introduce fairly detailed definitions, although we will not provide full proofs of all the stated results, as some are both intuitive and technically similar to what has already been established in this paper. We will focus on digraphs, although our results also apply to d -multigraphs.

Definition 6.9

1. A digraph is a pair $G = (|G|, A_G)$ where
 - $|G|$ is a finite set of vertices (or nodes), called base of G ,
 - A_G is a set of arcs (or directed edges), which are ordered pairs of vertices, i.e., $A_G \subseteq |G| \times |G|$.
2. A module M of a digraph G is a subset of $|G|$ such that, $\forall u, v \in M$, u and v have the same neighbors outside of M , i.e., $\forall x \in |G| \setminus M$, and $\forall u, v \in M$, we have $((x, u) \in A_G \text{ if and only if } (x, v) \in A_G)$ and $((u, x) \in A_G \text{ if and only if } (v, x) \in A_G)$.

The modular decomposition of a digraph consists of partitioning the set of vertices into modules. This decomposition simplifies the study of the digraph by reducing each module to a single vertex (one can think of "contracting" each module into a single vertex), yielding a simplified digraph on which algorithms can be more easily applied. We will provide the necessary definitions to explain the most famous decomposition due to Gallai [3]. This will be useful for understanding what follows.

Definition 6.10 *Let G be a digraph such that $|G| = \{x_1, \dots, x_n\}$ and M_1, \dots, M_n be digraphs with disjoint base. We define the digraph $\mathcal{G} = G[[M_1, \dots, M_n]]$ as the digraph obtained by replacing each vertex x_i with the digraph M_i , and connecting every vertex of M_i to every vertex of M_j if and only if $(x_i, x_j) \in A_G$. In other words, \mathcal{G} is constructed by taking the digraphs M_1, \dots, M_n and adding edges between M_i and M_j based on the edges between the corresponding vertices x_i and x_j in the digraph G . Formally, the digraph \mathcal{G} is defined by*

- $|\mathcal{G}| = \bigcup_{i=1}^n |M_i|$,
- $(u, v) \in |A_{\mathcal{G}}|$ if and only if $(\exists 1 \leq i \leq n, u, v \in |M_i| \text{ and } (u, v) \in A_{M_i})$ or $(\exists 1 \leq i \neq j \leq n, u \in |M_i|, v \in |M_j|, \text{ and } (x_i, x_j) \in A_G)$.

It is clear that the M_i 's are modules of the digraph \mathcal{G} . In this case, we say that the digraph \mathcal{G} is decomposable into modules M_1, \dots, M_n via the digraph G which is called the basis of the decomposition.

Gallai's modular decomposition of a digraph involves partitioning the vertices into specific modules: "proper, strong and maximal." This decomposition is unique, and its basis has a well-characterized specific form. We prefer not to provide all the definitions and details necessary to properly state Gallai's theorem because, on the one hand, it would require a lot of space, and on the other hand, we will not need its result in the following sections to establish our results. We are only interested in the idea of modular decomposition.

We will now define a game on digraphs, starting from a modular decomposition that is not necessarily the one from Gallai. In fact, the reductions we will define on the digraphs will not preserve such a decomposition. The idea is to assign names (father/son) to certain modules, as we did in the first part of the paper, with links between them. The goal of this game is that each son module receives all the connections associated with the corresponding father module. For this reason, we will use the sets \mathcal{F} and \mathcal{S} from the first part, as well as the notations $[[\dots]]$, which is useful for defining the modular decomposition. We begin by defining colored digraphs, and then introduce reduction concepts on these objects.

Definition 6.11 *Let \mathcal{B} be a countable set of variables. We define the colored digraph \mathcal{G} and the set $FS(\mathcal{G})$ as follows.*

- If $\mathcal{G} \in \mathcal{B}$, then \mathcal{G} is a colored digraph and $FS(\mathcal{G}) = \emptyset$.
- If G is a digraph such that $|G| = \{x_1, \dots, x_n\}$ and $\mathcal{G}_1, \dots, \mathcal{G}_n$ are colored digraphs, then $\mathcal{G} = G[[\mathcal{G}_1, \dots, \mathcal{G}_n]]$ is a colored digraph, with $FS(\mathcal{G}) = \bigcup_{i=1}^n FS(\mathcal{G}_i)$.

The new colored digraph is seen as a decomposition into the modules $\mathcal{G}_1, \dots, \mathcal{G}_n$ via G .

- If \mathcal{G}' is a colored digraph, then $\mathcal{G} = f_i \mathcal{G}'$ is a colored digraph, with $FS(\mathcal{G}) = FS(\mathcal{G}') \setminus \{s_i\}$. The new colored digraph $f_i \mathcal{G}'$ is seen as an assignment of a name f_i to the colored digraph \mathcal{G}' .

- If \mathcal{G}' is a colored digraph, then $\mathcal{G} = s_i\mathcal{G}'$ is a colored digraph, with $FS(\mathcal{G}) = FS(\mathcal{G}') \cup \{s_i\}$. The new colored digraph $s_i\mathcal{G}'$ is seen as an assignment of a name s_i to the colored digraph \mathcal{G}' .

We denote by \mathbb{G} the set of colored digraph.

We now define the main reduction on colored digraphs.

Definition 6.12

1. Let $\mathcal{G}, \mathcal{G}_1, \dots, \mathcal{G}_n \in \mathbb{G}$, G a digraph such that $|G| = \{x_1, \dots, x_n\}$ and $s_k \notin \bigcup_{i=1}^n FS(\mathcal{G}_i)$. Let $\sigma = \langle (s_k) := G \llbracket \mathcal{G}_1, \dots, \mathcal{G}_j, *, \mathcal{G}_{j+1}, \dots, \mathcal{G}_n \rrbracket \rangle$ that we call “substitution”. We define $\mathcal{G}\sigma$ by induction on \mathcal{G} as follows:

- If \mathcal{G} is a digraph, then $\mathcal{G}\sigma = \mathcal{G}$.
- If $\mathcal{G} = G' \llbracket \mathcal{G}'_1, \dots, \mathcal{G}'_l \rrbracket$, then $\mathcal{G}\sigma = G' \llbracket \mathcal{G}'_1\sigma, \dots, \mathcal{G}'_l\sigma \rrbracket$.
- If $\mathcal{G} = f_m\mathcal{G}'$ with $m \notin \{i_1, \dots, i_k\}$, then $\mathcal{G}\sigma = f_m\mathcal{G}'\sigma$.
Note that, by equivalence, we can assume that $m \notin \{i_1, \dots, i_k\}$, since otherwise, we can rename m by another new integer.
- If $\mathcal{G} = s_m\mathcal{G}'$ with $m \notin \{i_1, \dots, i_k\}$, then $\mathcal{G}\sigma = s_m\mathcal{G}'\sigma$.
- If $\mathcal{G} = s_m\mathcal{G}'$ with $m \in \{i_1, \dots, i_k\}$, then
 $\mathcal{G}\sigma = s_m G \llbracket \mathcal{G}_1, \dots, \mathcal{G}_j, \mathcal{G}'\sigma, \mathcal{G}_{j+1}, \dots, \mathcal{G}_n \rrbracket$.

2. An h -redex is a colored digraph under the form

$$G \llbracket \mathcal{G}_1, \dots, \mathcal{G}_k, f_i\mathcal{G}, \mathcal{G}_{k+1}, \dots, \mathcal{G}_n \rrbracket,$$

where $n \neq 0$. Its h -reduced via f_i is the following colored digraph

$$f_i\mathcal{G} \langle (s_i) := G \llbracket \mathcal{G}_1, \dots, \mathcal{G}_k, *, \mathcal{G}_{k+1}, \dots, \mathcal{G}_n \rrbracket \rangle.$$

Note that in this case, we can assume that $s_i \notin \bigcup_{i=1}^n FS(\mathcal{G}_i)$, thus the substitution is well-defined. Intuitively, this operation consists of cutting all connections to the colored digraph named by f_i , and then redirecting them to all colored digraphs named by s_i .

Figure 8 presents a specific example of how to obtain the h -reduced of an h -redex. We have a father module (labeled in red as f_i) connected to modules outside. This module contains three submodules, one of which, M , is its son (labeled in blue as s_i). The reduction only affects the module M , which inherits the connections from its father. Note that the red and blue colors remain, as they can be reused later. The situation is much more complex than what we observe in Figure 8, as we may have multiple independent or nested sons s_i that will undergo substitution after the reduction. Here, we aim to provide further motivation and justification for this notion of reduction. All elements within the module labeled by the father f_i interact uniformly with the external environment, particularly with respect to the son submodules labeled s_i . This reduction enables the father module to offload these connections, transferring them exclusively to its son modules.

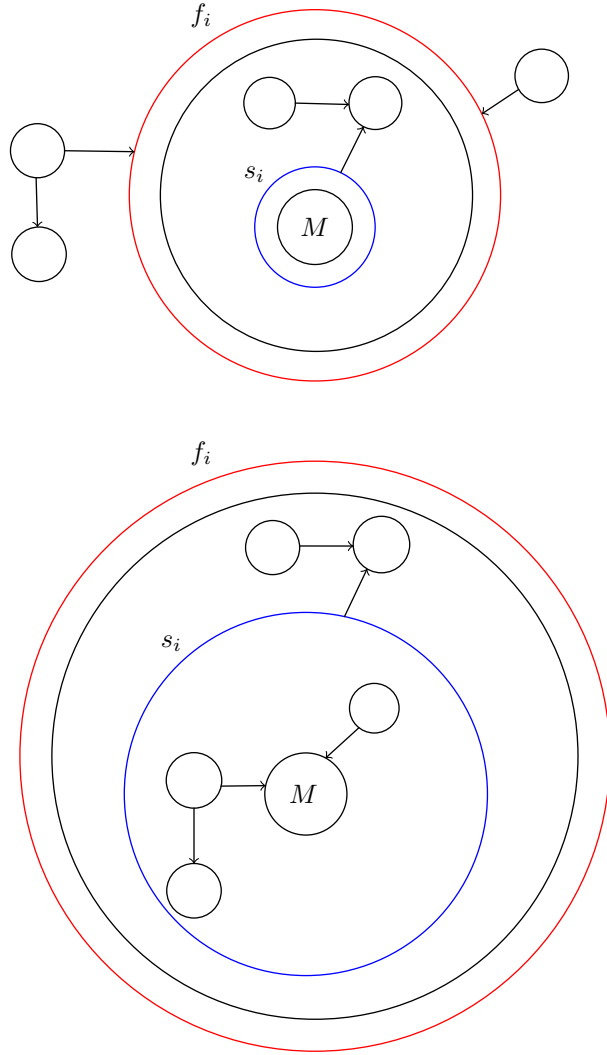


Figure 8: h -redex and h -reduced

We now define three other notions of redex and their reducts.

Definition 6.13

1. Let $\mathcal{G} \in \mathbb{G}$ and $i \in \mathbb{N}$. The colored digraph $[\mathcal{G}]^i$ is defined in the same way as for colored trees, by removing all instances of s_i from the colored digraph \mathcal{G} .
2.
 - An ff -redex is a colored digraph under the form $f_i f_j \mathcal{G}$. Its ff -reduced is $f_i [\mathcal{G}]^j$.
 - An fs -redex is a colored digraph under the form $f_i s_i \mathcal{G}$ with $s_i \notin FS(\mathcal{G})$. Its fs -reduced is \mathcal{G} .
 - An sf -redex is a colored digraph under the form $s_i f_j \mathcal{G}$. Its sf -reduced is $\mathcal{G}[s_j := s_i]$.

Finally, we define the notions of reductions and normalizations.

Definition 6.14 Let $R \subseteq \{h, ff, fs, sf\}$.

1. A colored digraph is said to be R -normal if it contains no r -redex for all $r \in R$.

2. If $\mathcal{G}, \mathcal{G}' \in \mathbb{G}$, we write $\mathcal{G} \rightarrow_R \mathcal{G}'$ if there exist a sequence $\mathcal{G}_1, \dots, \mathcal{G}_n \in \mathbb{G}$ such that $\mathcal{G} \rightarrow_{r_1} \mathcal{G}_1 \rightarrow_{r_2} \mathcal{G}_2 \rightarrow_{r_3} \dots \rightarrow_{r_n} \mathcal{G}_n = \mathcal{G}'$ where $r_i \in R$ for all $1 \leq i \leq n$.
3. A colored digraph \mathcal{G} is said to be weakly R -normalizable if there exists an R -normal colored digraph \mathcal{G}' such that $\mathcal{G} \twoheadrightarrow_R \mathcal{G}'$.
4. A colored digraph \mathcal{G} is said to be strongly R -normalizable if any sequence of R -reductions of \mathcal{G} ends.

We now define a notion of translation from colored digraphs to colored trees that preserves all reductions.

Definition 6.15 We define the translation $\mathcal{T} : \mathbb{G} \rightarrow \mathbb{T}$ by induction as follows.

- If $\mathcal{G} \in \mathcal{B}$, then $\mathcal{T}(\mathcal{G}) = \bullet$.
- If $\mathcal{G} = G \llbracket \mathcal{G}'_1, \dots, \mathcal{G}'_i \rrbracket$, then $\mathcal{T}(\mathcal{G}) = [\mathcal{T}(\mathcal{G}_1), \dots, \mathcal{T}(\mathcal{G}_i)]$.
- If $\mathcal{G} = f_m \mathcal{G}'$, then $\mathcal{T}(\mathcal{G}) = f_m \mathcal{T}(\mathcal{G}')$.
- If $\mathcal{G} = s_m \mathcal{G}'$, then $\mathcal{T}(\mathcal{G}) = s_m \mathcal{T}(\mathcal{G}')$.

It is easy to verify the following reduction preservation properties.

Lemma 6.16 Let $\mathcal{G}, \mathcal{G}' \in \mathbb{G}$ and $r \in \{h, ff, fs, sf\}$. If $\mathcal{G} \rightarrow_r \mathcal{G}'$, then $\mathcal{T}(\mathcal{G}) \rightarrow_r \mathcal{T}(\mathcal{G}')$.

Since the translation transforms a reduction in one calculus into a reduction in the other, the normalization results obtained in the previous sections are preserved on colored digraphs.

Theorem 6.17

1. Every colored digraph is strongly h -normalizable.
2. Every colored digraph is weakly $\{h, ff, fs, sf\}$ -normalizable.

Proof It suffices to apply Lemma 6.16 and Theorems 4.16 and 5.13. □

We will conclude this subsection with important remarks that highlight the significance of our results on digraphs.

- We motivated the definition of colored digraphs through modular decomposition; however, as we will see, the definition of a module does not play a role in our results. Thus, we can reinterpret the object $G \llbracket \mathcal{G}_1, \dots, \mathcal{G}_n \rrbracket$ in a different way and obtain termination results for other forms of digraphs.
- The translation \mathcal{T} that we have defined eliminates all bases of modular decompositions. This means that, after this translation, the h -reduction can be reinterpreted by varying these bases after the reductions, and also in a non-uniform manner. This mechanism leads to a much more general result, and it will be worthwhile to explore it further to derive additional applications, particularly in contexts where modular structures play a key role.
- Before an h -reduction, the submodules of a father module are connected in the same way with respect to their external environment. However, after the reduction, only the son modules of this father retain these connections, while the others become disconnected. This observation reveals another perspective on the h -reduction: it functions as a "cut" of connections for certain submodules of a father module, a behavior not observed in h -reductions on colored trees.

These remarks enhance our understanding of reduction mechanisms in different contexts and open up new avenues for future research.

Acknowledgments. We would like to thank Y. Boudabbous, C. Delhomme and A. El Sahili for the many discussions on this work, which helped to better present and generalize our results.

References

- [1] Battyányi, P. and Nour, K. *Normalization proofs for the un-typed $\mu\mu'$ -calculus*, Special Issue: LICMA'19 Lebanese International Conference on Mathematics and Applications., AIMS Mathematics, 5(4), pp. 3702-3713, 2020.
- [2] Battyányi, P. and Nour, K. *Normalization in the simply typed $\lambda\mu\mu'\rho\theta\varepsilon$ -calculus*. Mathematical Structures in Computer Science, 32(8), pp. 1066-1098, 2022.
- [3] Gallai, T. *Transitiv orientierbare graphen*. Acta Math. Acad. Sci. Hungar. 18, pp. 25-66, 1967.
- [4] David, R. and Nour, K. *Arithmetical proofs of strong normalization results for the symmetric $\lambda\mu\mu'$ -calculus*, In: P. Urzyczyn (editor), Typed Lambda Calculi and Applications, TLCA '05, Lecture Notes in Computer Science (3461), pp. 162-178, Springer Verlag, 2005.
- [5] El Zein, A. *About Cycles and Out-Domination in Tournaments and Rewriting on Trees*, PhD thesis, Lebanese University & Université Savoie Mont Blanc, 2023.
- [6] Kirby, L. and Paris, J. *Accessible independence results for Peano arithmetic*, Bulletin of the London Mathematical Society, 14, pp. 725-731, 1982.
- [7] Parigot, M. *$\lambda\mu$ -calculus: an algorithmic interpretation of classical natural deduction*, Lecture Notes in Computer Science (624), Springer Verlag, Berlin, pp. 190-201, 1992.