Title	Left-normal Translation for Applicative Term Rewriting Systems
Author(s)	鈴木，裕佑
Citation	
Issue Date	2021-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/17136
Rights	
Description	Supervisor: Nao Hirokawa, Graduate School of Advanced Science and Technology, Master of Science (Information Science)

Master's Thesis

Left-normal Translation for Applicative Term Rewriting Systems

1910251 Yusuke Suzuki

Supervisor	Nao Hirokawa
Main Examiner	Nao Hirokawa
Examiners	Hajime Ishihara
	Kazuhiro Ogata
	Mizuhito Ogawa

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
(Information Science)

February, 2021

Abstract

Term rewriting is a computation model based on directed equations. Sets of such equations are called term rewriting systems (TRSs); especially, TRSs over constant symbols and a single binary function symbol $\circ$, an application symbol, are called applicative term rewriting systems (ATRSs). ATRSs underlie functional programming languages and proof assistants and enable them to model higher-order functions.

Consider the following ATRS $\mathcal{R}$ that computes Fibonacci numbers.

$$
\begin{array}{lrl}
1: & +\ \mathsf{0}\ x & \to x \\
2: & +\ (\mathsf{s}\ x)\ y & \to \mathsf{s}\ (+\ x\ y) \\
3: & \mathsf{tail}\ (:\ x\ xs) & \to xs \\
4: & \mathsf{nth}\ (:\ x\ xs)\ \mathsf{0} & \to x \\
5: & \mathsf{nth}\ (:\ x\ xs)\ (\mathsf{s}\ y) & \to \mathsf{nth}\ xs\ y \\
6: & \mathsf{zip}\ f\ (:\ x\ xs)\ (:\ y\ ys) & \to\ :\ (f\ x\ y)\ (\mathsf{zip}\ f\ xs\ ys) \\
7: & \mathsf{fibs} & \to\ :\ \mathsf{0}\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \mathsf{fibs}\ (\mathsf{tail}\ \mathsf{fibs})))
\end{array}
$$

For instance, the term $\mathsf{nth}\ \mathsf{fibs}\ (\mathsf{s}\ (\mathsf{s}\ \mathsf{0}))$ is rewritten to the second Fibonacci number. However, a naive computation may cause an infinite rewrite sequence like:

$$
\begin{array}{l}
\mathsf{nth}\ \underline{\mathsf{fibs}}\ (\mathsf{s}\ (\mathsf{s}\ \mathsf{0})) \to \mathsf{nth}\ (:\ \mathsf{0}\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \underline{\mathsf{fibs}}\ (\mathsf{tail}\ \mathsf{fibs}))))\ (\mathsf{s}\ (\mathsf{s}\ \mathsf{0})) \\
\to \mathsf{nth}\ (:\ \mathsf{0}\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ (:\ \mathsf{0}\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \underline{\mathsf{fibs}}\ (\mathsf{tail}\ \mathsf{fibs}))))\ (\mathsf{tail}\ \mathsf{fibs}))))\ (\mathsf{s}\ (\mathsf{s}\ \mathsf{0})) \\
\to \cdots
\end{array}
$$

whilst another computation yields the finite rewrite sequence:

$$
\begin{array}{l}
\mathsf{nth}\ \underline{\mathsf{fibs}}\ (\mathsf{s}\ (\mathsf{s}\ \mathsf{0})) \to \underline{\mathsf{nth}\ (:\ \mathsf{0}\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \mathsf{fibs}\ (\mathsf{tail}\ \mathsf{fibs}))))\ (\mathsf{s}\ (\mathsf{s}\ \mathsf{0}))} \\
\to \underline{\mathsf{nth}\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \mathsf{fibs}\ (\mathsf{tail}\ \mathsf{fibs})))\ (\mathsf{s}\ \mathsf{0})} \to \mathsf{nth}\ (\mathsf{zip}\ +\ \underline{\mathsf{fibs}}\ (\mathsf{tail}\ \mathsf{fibs}))\ \mathsf{0} \\
\to \mathsf{nth}\ (\mathsf{zip}\ +\ (:\ \mathsf{0}\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \mathsf{fibs}\ (\mathsf{tail}\ \mathsf{fibs}))))\ (\mathsf{tail}\ \underline{\mathsf{fibs}}))\ \mathsf{0} \\
\to \mathsf{nth}\ (\underline{\mathsf{zip}\ +\ (:\ \mathsf{0}\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \mathsf{fibs}\ (\mathsf{tail}\ \mathsf{fibs}))))} \\
\qquad \underline{(\mathsf{tail}\ (:\ \mathsf{0}\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \mathsf{fibs}\ (\mathsf{tail}\ \mathsf{fibs})))))})\ \mathsf{0} \\
\to \underline{\mathsf{nth}\ (:\ (+\ \mathsf{0}\ (\mathsf{s}\ \mathsf{0}))\ (\mathsf{zip}\ +\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \mathsf{fibs}\ (\mathsf{tail}\ \mathsf{fibs})))} \\
\qquad \underline{(:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \mathsf{fibs}\ (\mathsf{tail}\ \mathsf{fibs})))))\ \mathsf{0}} \\
\to \underline{+\ \mathsf{0}\ (\mathsf{s}\ \mathsf{0})} \\
\to \mathsf{s}\ \mathsf{0}
\end{array}
$$

Underlines indicate rewrite positions. Comparing the two rewrite sequences, they rewrite the same position at their first step, whilst different positions thereafter. In the latter, subterms are *needed* to be rewritten at every underlined position in order to obtain the second Fibonacci number. On the other hand, in the former, subterms are not *needed* to be rewritten at underlined positions but ones at the first step. Such positions that are needed for results of computations are called needed positions, and the strategy that rewrites subterms at needed positions are the needed strategy. However, it is known that needed positions are uncomputable in general [5], and thus we cannot use the strategy readily.

As for another strategy, O'Donnell [9] showed the Normalization Theorem: in left-normal TRSs, the leftmost-outermost strategy always leads a term to a computational result. Left-normality is the property that no function symbols occur on the right of variables in a term. If every left-hand side of a TRS is left-normal, also the TRS is called so. Recalling the ATRS $\mathcal{R}$, the rules 4, 5 and 6 are not left-normal because of the underlined subterms, neither is $\mathcal{R}$.

$$\begin{array}{rrl}
4: & \mathsf{nth}\ (:\ x\ xs)\ \underline{\mathsf{0}} & \to x \\
5: & \mathsf{nth}\ (:\ x\ xs)\ \underline{(\mathsf{s}\ y)} & \to \mathsf{nth}\ xs\ y \\
6: & \mathsf{zip}\ f\ (:\ x\ xs)\ \underline{(:\ y\ ys)} & \to :\ (f\ x\ y)\ (\mathsf{zip}\ f\ xs\ ys)
\end{array}$$

In this thesis, we propose left-normal translation for ATRSs, which translates non-left-normal ATRSs into left-normal ATRSs and enables us to obtain results of computations with the translated ATRSs. For example, the previous TRS $\mathcal{R}$ is translated into the following left-normal ATRSs.

$$\begin{array}{rrl}
1: & +\ \mathsf{0}\ x & \to x \\
2: & +\ (\mathsf{s}\ x)\ y & \to \mathsf{s}\ (+\ x\ y) \\
3: & \mathsf{tail}\ (:\ x\ xs) & \to xs \\
4: & \mathsf{fibs} & \to :\ \mathsf{0}\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \mathsf{fibs}\ (\mathsf{tail}\ \mathsf{fibs}))) \\
5: & \mathsf{nth}\ (:\ x\ xs)\ y & \to \mathsf{nth}_1\ y\ x\ xs \\
6: & \mathsf{zip}\ f\ xs\ ys & \to \mathsf{zip}_1\ xs\ f\ ys \\
7: & \mathsf{nth}_1\ \mathsf{0}\ x\ xs & \to x \\
8: & \mathsf{nth}_1\ (\mathsf{s}\ y)\ x\ xs & \to \mathsf{nth}\ xs\ y \\
9: & \mathsf{zip}_1\ (:\ x\ xs)\ f\ ys & \to \mathsf{zip}_2\ ys\ x\ xs\ f \\
10: & \mathsf{zip}_2\ (:\ y\ ys)\ x\ xs\ f & \to :\ (f\ x\ y)\ (\mathsf{zip}\ f\ xs\ ys)
\end{array}$$

With this left-normal ATRS, we get obtain the term $\textsf{s 0}$ from $\textsf{nth fibs (s (s 0))}$ by the leftmost-outermost strategy.

$$
\begin{aligned}
&\textsf{nth } \underline{\textsf{fibs}} \textsf{ (s (s 0))} \\
&\to \underline{\textsf{nth (: 0 (: (s 0) (zip + fibs (tail fibs)))) (s (s 0))}} \\
&\to \underline{\textsf{nth}_1 \textsf{ (s (s 0)) 0 (: (s 0) (zip + fibs (tail fibs)))}} \\
&\to \underline{\textsf{nth (: (s 0) (zip + fibs (tail fibs))) (s 0)}} \\
&\to \underline{\textsf{nth}_1 \textsf{ (s 0) (s 0) (zip + fibs (tail fibs))}} \\
&\to \textsf{nth (}\underline{\textsf{zip + fibs (tail fibs)}}\textsf{) 0} \\
&\to \textsf{nth (zip}_1 \ \underline{\textsf{fibs}} \textsf{ + (tail fibs)) 0} \\
&\to \textsf{nth (}\underline{\textsf{zip}_1 \textsf{ (: 0 (: (s 0) (zip + fibs (tail fibs)))) + (tail fibs)}}\textsf{) 0} \\
&\to \textsf{nth (zip}_2 \textsf{ (tail } \underline{\textsf{fibs}}\textsf{) 0 (: (s 0) (zip + fibs (tail fibs))) +) 0} \\
&\to \textsf{nth (zip}_2 \textsf{ (}\underline{\textsf{tail (: 0 (: (s 0) (zip + fibs (tail fibs)))))}} \\
&\qquad \textsf{0 (: (s 0) (zip + fibs (tail fibs))) +) 0} \\
&\to \textsf{nth (}\underline{\textsf{zip}_2 \textsf{ (: (s 0) (zip + fibs (tail fibs))) 0 (: (s 0) (zip + fibs (tail fibs))) +}}\textsf{) 0} \\
&\to \underline{\textsf{nth (: (+ 0 (s 0))}} \\
&\qquad \underline{\textsf{(zip + (: (s 0) (zip + fibs (tail fibs))) (zip + fibs (tail fibs)))) 0}} \\
&\to \underline{\textsf{nth}_1 \textsf{ 0 (+ 0 (s 0))}} \\
&\qquad \underline{\textsf{(zip + (: (s 0) (zip + fibs (tail fibs))) (zip + fibs (tail fibs)))}} \\
&\to \underline{\textsf{+ 0 (s 0)}} \\
&\to \textsf{s 0}
\end{aligned}
$$

Meanwhile, this translation moves needed positions to leftmost-outermost positions, which the leftmost-outermost strategy rewrites. Hence, by the translation, we can simulate the needed strategy by the leftmost-outermost strategy.

There is an existing work: left-normal translation was originally developed by Hashida [4]. Hashida's translation translates constructor systems to left-normal constructor systems. This is the pioneering work using an approach that simulates the needed strategy by the leftmost-outermost strategy. Our study is aimed to extend Hashida's translation to ATRSs and to enlarge the class of TRSs that can be left-normal.

Our contribution is two-fold. Firstly, we establish left-normal translation for ATRSs, and then realise simulating the needed strategy by the leftmost-outermost strategy in computations of applicative terms. Secondly, we show that our translation for ATRSs includes Hashida's translation [4]: functional TRSs that can be translated by left-normal translation for functional TRSs can be translated by left-normal translation for ATRSs after currying, a procedure that translates functional TRSs into ATRSs. The translation also can handle ATRSs including higher-order function. We therefore succeed in extending the class of TRSs with which terms can be computed by the needed strategy.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the beginning of this thesis, we give an overview of our study and its results. Term rewriting is a computational model that replaces terms by directed equations, called rewrite rules. We call the set of rewrite rules a term rewriting systems (TRS).

1.1 Motivation

Consider the TRS $\mathcal{R}$ with the following seven rules

$$
\begin{aligned}
1:&& + \ \mathsf{0}\ x &\to x\\
2:&& + \ (\mathsf{s}\ x)\ y &\to \mathsf{s}\ (+\ x\ y)\\
3:&& \mathsf{tail}\ (:\ x\ xs) &\to xs\\
4:&& \mathsf{nth}\ (:\ x\ xs)\ \mathsf{0} &\to x\\
5:&& \mathsf{nth}\ (:\ x\ xs)\ (\mathsf{s}\ y) &\to \mathsf{nth}\ xs\ y\\
6:&& \mathsf{zip}\ f\ (:\ x\ xs)\ (:\ y\ ys) &\to\ :\ (f\ x\ y)\ (\mathsf{zip}\ f\ xs\ ys)\\
7:&& \mathsf{fibs} &\to\ :\ \mathsf{0}\ (:\ (\mathsf{s}\ \mathsf{0})\ (\mathsf{zip}\ +\ \mathsf{fibs}\ (\mathsf{tail}\ \mathsf{fibs})))
\end{aligned}
$$

and the term $t = \mathsf{nth}\ \mathsf{fibs}\ (\mathsf{s}\ (\mathsf{s}\ \mathsf{0}))$.

The TRS $\mathcal{R}$ computes the Fibonacci numbers and t is a term to take the second element of the numbers, namely $\mathsf{s}\ \mathsf{0}$. In these terms there exist the unique application symbols $\circ$ between two subterms. For example, notation $t_1\ t_2$ stands for $t_1 \circ t_2$, moreover it is the infix notation of $\circ(t_1, t_2)$. Hence the term t equals to $\circ(\mathsf{nth}, \circ(\mathsf{fibs}, \circ(\mathsf{s}, \circ(\mathsf{s}, \mathsf{0}))))$. Such TRSs, which consists of constant symbols and the unique binary function $\circ$ called the application symbol, are called applicative term rewriting systems (ATRSs), compared with the other ones called functional term rewriting systems. ATRSs are adopted by most of functional programming languages like Haskell [11] and proof assistants like Coq [2], and have advantage of dealing with higher-order functions as the sixth rule of the above ATRS $\mathcal{R}$.

Computation of the term t is a problem of which subterms will be rewritten. To obtain the second Fibonacci number, it is enough to look at the first two elements of Fibonacci numbers, and the remains do not matter. This policy yields the following rewrite sequence.

$$
\begin{aligned}
&\textsf{nth } \underline{\textsf{fibs}}\textsf{ (s (s 0))} \to \underline{\textsf{nth (: 0 (: (s 0) (zip + fibs (tail fibs)))) (s (s 0))}}\\
&\to \underline{\textsf{nth (: (s 0) (zip + fibs (tail fibs))) (s 0)}} \to \textsf{nth (zip + }\underline{\textsf{fibs}}\textsf{ (tail fibs)) 0}\\
&\to \textsf{nth (zip + (: 0 (: (s 0) (zip + fibs (tail fibs)))) (tail }\underline{\textsf{fibs}}\textsf{)) 0}\\
&\to \textsf{nth (}\underline{\textsf{zip + (: 0 (: (s 0) (zip + fibs (tail fibs))))}}\\
&\qquad \underline{\textsf{(tail (: 0 (: (s 0) (zip + fibs (tail fibs)))))}}\textsf{) 0}\\
&\to \underline{\textsf{nth (: (+ 0 (s 0)) (zip + (: (s 0) (zip + fibs (tail fibs)))}}\\
&\qquad \underline{\textsf{(: (s 0) (zip + fibs (tail fibs))))) 0}}\\
&\to \underline{\textsf{+ 0 (s 0)}}\\
&\to \textsf{s 0}
\end{aligned}
$$

In the above example only subterms needed for reaching $\textsf{s 0}$ are evaluated. Such a strategy is called the needed strategy.

However, not all strategies bring the result; the following strategy fails to obtain the second Fibonacci number, which stems from repeated evaluations of subterms $\textsf{fibs}$ causing an infinite rewrite sequence.

$$
\begin{aligned}
&\textsf{nth } \underline{\textsf{fibs}}\textsf{ (s (s 0))}\\
&\to \textsf{nth (: 0 (: (s 0) (zip + }\underline{\textsf{fibs}}\textsf{ (tail fibs)))) (s (s 0))}\\
&\to \textsf{nth (: 0 (: (s 0) (zip + (: 0 (: (s 0) (zip + }\underline{\textsf{fibs}}\textsf{ (tail fibs)))) (tail fibs)))) (s (s 0))}\\
&\to \cdots
\end{aligned}
$$

The needed strategy rewrites subterms only at positions where subterms need to be evaluated in order to obtain a result of a calculation, called a normal form. We call those positions needed positions. It is known that if a TRS has the property called orthogonality, repeated rewriting at needed positions leads a term to a normal form [5]. Our study is aimed to compute applicative terms by the needed strategy.

1.2 Existing work

Needed positions are, however, uncomputable in general. Huet and Lévy [5] defined strong sequentiality for TRSs, and showed that needed positions are efficiently computable in strongly sequential TRSs. Another study by O'Donnell showed the Normalization Theorem, which claims that repeated rewriting at the leftmost-outermost positions, called the leftmost-outermost strategy, leads a term to a normal form in orthogonal TRSs if the term has a normal form. This theorem requires TRSs one more condition: left-normality, defined as the property that for every rule $\ell \to r$ in a TRS, any function symbol in ℓ occurs on the right of variables.

Example 1. Consider the TRS $\mathcal{R}$:

$$\begin{array}{rrlrrl} 1: & \mathsf{take}(:(x, xs), \underline{\mathsf{0}}) & \to \mathsf{nil} & 3: & \mathsf{from}(x) & \to :(x, \mathsf{from}(\mathsf{s}(x))) \\ 2: & \mathsf{take}(:(x, xs), \underline{\mathsf{s}(y)}) & \to :(x, \mathsf{take}(xs, y)) & 4: & +(\mathsf{0}, x) & \to x \end{array}$$

The TRS $\mathcal{R}$ is not left-normal because $\mathsf{0}$ in the first rule and $\mathsf{s}(y)$ in the second rule with the underlines occur on the right of variables.

Hashida [4] introduced left-normal translation, which translates non-left-normal TRSs into left-normal TRSs, and then enables us to apply the Normalization Theorem and obtain normal forms by the leftmost-outermost strategy. Meanwhile, this procedure moves needed positions from the right of variables to the leftmost-outermost positions. That is to say, the leftmost-outermost strategy repeatedly rewrites subterms at needed positions, and consequently it simulates the needed strategy.

1.3 Subjects

Unfortunately, this approach is not suitable for ATRSs. In order to explain the reason, we need to describe a little more background on which the translation relies. Needless to say, to move needed positions, there must be needed positions to move. Strong sequentiality is a property that ensures existence of positions called indices, which can be needed positions in orthogonal TRSs. Hence the translation requires a non-left-normal TRS to be strongly sequential. Moreover, in case translation does not finish at a time, the translated TRS, which has been non-left-normal, needs to be strongly sequential. Hashida and we have recourse to a refined method by Klop and Middeldorp [6] to show strong sequentiality, which requires orthogonal TRSs to be *constructor systems.*

Function symbols in TRSs are divided into two sets: defined symbols and constructor symbols. The set of defined symbols $\mathcal{D}_\mathcal{R}$ of a TRS $\mathcal{R}$ consists of function symbols occurring on the left of left-hand sides. Such a position is called the root position. If a function symbol is not a defined symbol, it is a constructor symbol; the set of constructor symbols of a TRS $\mathcal{R}$ is denoted by $\mathcal{C}_\mathcal{R}$. A TRS $\mathcal{R}$ is called a constructor system if for every rule $\ell \to r \in \mathcal{R}$, no defined symbols occur at positions in ℓ excepting the root position.

Example 2. Consider the TRS $\mathcal{R}$:

$$\begin{array}{rrrr} 1: & \underline{\mathsf{take}}(:(x, xs), \mathsf{0}) \to \mathsf{nil} & 3: & \underline{\mathsf{from}}(x) \to :(x, \mathsf{from}(\mathsf{s}(x))) \\ 2: & \underline{\mathsf{take}}(:(x, xs), \mathsf{s}(y)) \to :(x, \mathsf{take}(xs, y)) & 4: & \underline{+}(\mathsf{0}, x) \to x \end{array}$$

The underlined symbols are defined symbols and the remaining symbols are constructor symbols. Hence $\mathcal{D}_\mathcal{R} = \{\mathsf{take}, \mathsf{from}, +\}$ and $\mathcal{C}_\mathcal{R} = \{:, \mathsf{0}, \mathsf{s}, \mathsf{nil}\}$, and $\mathcal{R}$ is a constructor system.

As for ATRSs, the secret is to recall that $\circ$ is the unique binary symbol. It is apparent from the next example that ATRSs hardly form constructor systems.

Example 3. Consider the ATRS $\mathcal{R}$:

$$\begin{array}{rr} 1: & \underline{\circ}(\circ(\mathsf{take}, \circ(\circ(:, x), xs)), \mathsf{0}) \to \mathsf{nil} \\ 2: & \underline{\circ}(\circ(\mathsf{take}, \circ(\circ(:, x), xs)), \circ(\mathsf{s}, y)) \to \circ(\circ(:, x), \circ(\circ(\mathsf{take}, xs), y)) \\ 3: & \underline{\circ}(\mathsf{from}, x) \to \circ(\circ(:, x), \circ(\mathsf{from}, \circ(\mathsf{s}, x))) \\ 4: & \underline{\circ}(\circ(+, \mathsf{0}), x) \to x \end{array}$$

The underlined positions are the root positions. Hence we have $\mathcal{D}_\mathcal{R} = \{\circ\}$ and $\mathcal{C}_\mathcal{R} = \{\mathsf{take}, \mathsf{from}, +, :, \mathsf{0}, \mathsf{s}, \mathsf{nil}\}$. Since the application symbols $\circ$ occur in positions other than the root positions, $\mathcal{R}$ is not a constructor system.

Moreover, the lemma on strong sequentiality of TRSs in [6] is based on several other lemmata. One of them, which mentions transitivity of indices of two terms, requires

TRSs to be constructor systems. Because the form of applicative terms is different from the one of functional terms, index transitivity in ATRSs cannot be proved by the lemma.

Those two reasons, (1) ATRSs are rarely constructor systems and (2) strong sequentiality of ATRSs cannot be shown by existing methods, have prevented left-normal translation for ATRSs. This thesis is an endeavour to overcome the problems and establish left-normal translation for ATRSs.

1.4 Outline and results

Out contribution is two-fold: firstly, as aforementioned, the original left-normal translation by Hashida does not fit into ATRSs on account of constructor systems. In order to address the problem, we introduce *applicative* constructor systems (ACSs). This notion enables handling function symbols at the leftmost positions of left-hand sides in ATRSs as if they were defined symbols in functional TRSs, regardless of the application symbols $\circ$. By the concept of ACSs, we propose left-normal translation for ATRSs, and then we realise simulation of the needed strategy by the leftmost-outermost strategy in ATRSs.

Secondly, we show that left-normal translation for ATRSs includes the original translation for functional TRSs through currying, a procedure to translate functional TRSs into ATRSs. More precisely, for every functional TRS that can be translated by left-normal translation for functional TRSs, the curried ATRS from the functional TRS can be translated by left-normal translation for ATRSs. In addition to these TRSs, the translation for ATRSs can deal with ACSs using higher-order functions. That is to say, the translation for ATRSs has increases a range of TRSs with which terms can be calculated by the needed strategy.

After this introduction, we give fundamental definitions on term rewriting in the next chapter. An applicative version of left-normal translation is presented in Chapter 3. In Chapter 4, we show that the translation for ATRS includes the original translation by Hashida. From the arguments in the previous chapters, we conclude our study in Chapter 5.

Chapter 2

Preliminaries

We shall begin by introducing fundamental definitions necessary for this thesis. In this chapter, integers of 0 and above are called *natural numbers*, and we write $\mathbb{Z}$, $\mathbb{N}$ and $\mathbb{N}_+$ for the whole sets of integers, natural numbers and positive integers, respectively. The difference set $\{a \in A \mid a \notin B\}$ for sets A and B, and the empty set are denoted by $A \setminus B$ and $\varnothing$, respectively. We refer to most definitions below from [1].

2.1 Abstract rewriting systems

A binary relation on A is a subset of the Cartesian product $A \times A$. We call the pair $(A, \to)$ of a set A and a binary relation $\to$ over A an *abstract reduction system (ARS)*. For an ARS $(A, \to)$ and elements $a, b \in A$, we write $a \to b$ for $(a, b) \in \to$ and $a \not\to b$ for $(a, b) \notin \to$. Let us describe a binary relation $\to$ on a set A.

Definition 4. The binary relation $\leftarrow$ defined by $\leftarrow = \{(b, a) \in A \times A \mid a \to b\}$ is called the *inverse* of $\to$.

Definition 5. For binary relations $\to_1$ and $\to_2$ on a set A, we define the relation $\to_1 \cdot \to_2$ as follows:

$$\to_1 \cdot \to_2 = \{(a, c) \in A \times A \mid \text{there exists } b \in A \text{ such that } a \to_1 b \text{ and } b \to_2 c\}$$

We call it the *composition* of $\to_1$ and $\to_2$.

Definition 6. For a natural number n, the binary relation $\to^n$ is defined as follows.

$$\to^n = \begin{cases} \{(x,x) \mid x \in A\} & \text{if } n = 0 \\ \to \cdot \to^{n-1} & \text{if } n \geq 1 \end{cases}$$

Definition 7. For a binary relation $\to$, the following relations are defined.

- $\to^= \; = \; \to^0 \cup \to$ is the *reflexive closure*.
- $\leftrightarrow \; = \; \leftarrow \cup \to$ is the *symmetric closure*.
- $\to^+ = \bigcup_{n \in \mathbb{N}_+} \to^n$ is the *transitive closure*.
- $(\to^=)^+$ is denoted by $\to^*$.

Definition 8. Let a be an element in a set A. If there exists no element $b \in A$ such that $a \to b$, then the element a is called a *normal form* with respect to $\to$. We write $\mathsf{NF}(\to)$ for the set of all normal forms with respect to $\to$. If $a \to^* b$ and $b \in \mathsf{NF}(\to)$ then, the normal form b is called a *normal form of* a with respect to $\to$. The unique normal form of c with respect to $\to$ is denoted by $c{\downarrow}$.

Definition 9. The relation $a \to^! b$ holds if $a \to^+ b$ and $b \notin \mathsf{NF}(\to)$.

Definition 10. A binary relation $\to$ is called *normalising* if every element in a set A has a normal form.

Definition 11. For a binary relation $\to$, we define the binary relation $\downarrow$ by $\to^* \cdot \leftarrow^*$. We say that two elements a and b in a set A *join* under $\to$ if $a \downarrow b$.

Definition 12. Let $\to$ be a binary relation.

- The binary relation $\to$ is *terminating* if there exists no infinite chain $a_n \to a_{n+1} \to \cdots$ such that $a_n \to a_{n+1}$ holds for every natural number n and the elements $a_n, a_{n+1} \in A$.
- We say that $\to$ is *confluent* if $\leftarrow^* \cdot \to^* \subseteq \downarrow$.
- The binary relation $\to$ is *complete* if $\to$ is terminating and confluent.

2.2 Terms

We fix a countable finite set $\mathcal{V}$ and call every element in $\mathcal{V}$ a *variable*. Moreover we fix a set $\mathcal{F}$ called a *signature* such that $\mathcal{V} \cap \mathcal{F} = \varnothing$ and a mapping $\mathsf{ar} : \mathcal{F} \to \mathbb{N}$ to call every element in $\mathcal{F}$ a *function symbol* and $\mathsf{ar}(f)$ the *arity of* f for every function symbol f in $\mathcal{F}$. For a subset $\mathcal{F}_0$ in $\mathcal{F}$ and a natural number n, we define the set $\mathcal{F}_0^{(n)}$ by $\mathcal{F}_0^{(n)} = \{f \in \mathcal{F}_0 \mid \mathsf{ar}(f) = n\}$. Every element in $\mathcal{F}^{(0)}$ is called a *constant symbol*.

Definition 13. For a subset $\mathcal{F}_0$ in $\mathcal{F}$ and a subset $\mathcal{V}_0$ in $\mathcal{V}$, we define the set $\mathcal{T}(\mathcal{F}_0, \mathcal{V}_0)$ as the minimum set satisfying the following two conditions.

- $\mathcal{V}_0 \subseteq \mathcal{T}(\mathcal{F}_0, \mathcal{V}_0)$, and
- letting n be a natural number, f a function symbol with an arity n, if $t_1, \ldots, t_n \in \mathcal{T}(\mathcal{F}_0, \mathcal{V}_0)$ then $f(t_1, \ldots, t_n) \in \mathcal{T}(\mathcal{F}_0, \mathcal{V}_0)$.

The set $\mathcal{T}(\mathcal{F}_0, \varnothing)$ is usually denoted by $\mathcal{T}(\mathcal{F}_0)$. Every element in $\mathcal{T}(\mathcal{F}_0, \mathcal{V}_0)$ is called a *term*, and especially every element in $\mathcal{T}(\mathcal{F}_0)$ is called a *ground term*.

Definition 14. We write $\mathbb{N}^*$ for the set of all finite sequences over $\mathbb{N}$, and ε for the sequence with length 0. A position $p \in \mathbb{N}^*$ of a term t is recursively defined as follows.

- If $p = \varepsilon$, then ε is a position of t.
- If $p = iq$ and t is denoted by $f(t_1, \ldots, t_n)$, then iq is a position of t if q is a position of t_i.

The position ε is called the *root position*.

Definition 15. We write $Pos(t)$ for the set of all positions of t, and $Pos_{\mathcal{F}}(t)$ for the set of all positions of function symbols, called function positions. The *subterm* of a term t at a position p is denoted by $t|_p$, recursively defined as follows.

$$t|_p = \begin{cases} t & \text{if } p = \varepsilon \\ t|_p = t_i|_q & \text{if } p = iq \text{ and } t = f(t_1, \ldots, t_n) \end{cases}$$

If a term u is a subset of a term t, it is denoted by $t \trianglerighteq u$. Especially if $t \trianglerighteq u$ and $t \neq u$, then u is a *strict subterm* of t, which is denoted by $t \triangleright u$. Given terms t, u and a position p of t, the term $t[u]_p$ is recursively defined as follows.

$$t[u]_p = \begin{cases} u & \text{if } p = \varepsilon \\ f(t_1, \ldots, t_{i-1}, t_i[u]_q, t_{i+1}, \ldots, t_n) & \text{if } p = iq \text{ and } t = f(t_1, \ldots, t_n) \end{cases}$$

Definition 16. The set $\mathit{Var}(t)$ of all variables occurring in a term t is defined as follows.

$$\mathit{Var}(t) = \{x \in \mathcal{V} \mid p \in \mathit{Pos}(t) \text{ and } t|_p = x\}$$

A position p such that the term $t|_p$ is a variable is called a variable position. We write $\mathit{Pos}_{\mathcal{V}}(t)$ for the set of all variable positions of t.

Example 17. Consider a signature $\mathcal{F} = \{\mathsf{f}, \mathsf{a}\}$ with $\mathsf{f} \in \mathcal{F}^{(2)}$ and $\mathsf{a} \in \mathcal{F}^{(0)}$, and a term $t = \mathsf{f}(\mathsf{a}, x)$. Firstly we have $\mathit{Pos}(t) = \{\varepsilon, 1, 2\}$, and thus $t \rhd \mathsf{a}$ and $t \rhd x$. Since $t|_1 = \mathsf{a}$ and $t|_2 = x$, we have $\mathit{Var}(t) = \{x\}$, and then $\mathit{Pos}_{\mathcal{F}}(t) = \{\varepsilon, 1\}$ and $\mathit{Pos}_{\mathcal{V}}(t) = \{2\}$. The term $t[\mathsf{a}]_2$ is $\mathsf{f}(\mathsf{a}, \mathsf{a})$.

Definition 18. The *size of a term t* is recursively defined as follows.

$$|t| = \begin{cases} 1 + |t_1| + \cdots + |t_n| & \text{if } t = f(t_1, \ldots, t_n) \\ 1 & \text{otherwise} \end{cases}$$

The *size of the function symbols in a term t* is defined by:

$$||t|| = \begin{cases} 0 & \text{if } t \text{ is a variable} \\ 1 & \text{if } t \text{ is a constant} \\ 1 + ||t_1|| + \cdots + ||t_n|| & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

Example 19. Consider a term $t = \mathsf{f}(\mathsf{a}, x)$. We have:

$$\begin{aligned} |t| &= |\mathsf{f}(\mathsf{a}, x)| = 1 + |\mathsf{a}| + |x| \\ &= 1 + 1 + 1 \\ &= 3 \\ ||t|| &= ||\mathsf{f}(\mathsf{a}, x)|| = 1 + ||\mathsf{a}|| + ||x|| \\ &= 1 + 1 + 0 \\ &= 2 \end{aligned}$$

Definition 20. A mapping $\sigma : \mathcal{V} \to \mathcal{T}(\mathcal{F}, \mathcal{V})$ such that the set $\{x \in \mathcal{V} \mid \sigma(x) \neq x\}$ is finite is called a *substitution*. Given a term t and a substitution σ, the term $t\sigma$ is recursively defined as follows.

$$t\sigma = \begin{cases} \sigma(t) & \text{if } t \text{ is a variable} \\ t\sigma = f(t_1\sigma, \ldots, t_n\sigma) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

The *domain* of σ, denoted by $Dom(\sigma)$, is defined by $Dom(\sigma) = \{x \in \mathcal{V} \mid \sigma(x) \notin x\}$. The *range* of σ, denoted by $Ran(\sigma)$. defined by $Ran(\sigma) = \{\sigma(x) \mid x \in Dom(\sigma)\}$. For two substitutions σ, τ and a variable x, the *composition* of two substitutions $\sigma\tau$ is defined by $(\sigma\tau)x = (x\sigma)\tau$. Letting $\mathcal{V}_0$ be a set consisting of variables, a substitution σ is called a *ground substitution* over $\mathcal{V}_0$ if $x\sigma$ is a ground term for every variable x in $\mathcal{V}_0$.

Definition 21. We say that a term t is an *instance* of a term s if there exists a substitution σ such that $s\sigma = t$.

Definition 22. Given two substitutions σ and τ, we say that σ is *more general* than τ if there exists a substitution δ such that $\sigma\delta = \tau$. A substitution σ is called an *unifier* of two terms s and t if $s\sigma = t\sigma$, and especially σ is a *most general unifier (mgu)* if σ is more general than any other unifiers of s and t.

2.3 Term rewriting systems

Furthermore, we prepare definitions on term rewriting systems.

Definition 23. A pair of two terms (ℓ, r) with $Var(\ell) \supseteq Var(r)$ and $\ell \notin \mathcal{V}$ is a *rewrite rule*, denoted by $\ell \to r$. We call ℓ the *left-hand side* and r the *right-hand side* of the rewrite rule. A set of rewrite rules is called a *term rewriting system (TRS)*. An instance of the left-hand side of a rewrite rule is a reducible expression, which is called *redex* for short.

Definition 24. We denote $\mathcal{R}^{-1}$ as the TRS obtained by swapping the left-hand sides and the right-hand sides of every rule in a TRS.

Definition 25. Given a TRS $\mathcal{R}$ and terms t, u, the relation $t \to_{\mathcal{R}} u$ holds if there exist a rewrite rule $l \to r \in \mathcal{R}$, a position $p \in Pos(s)$ and a substitution σ such that $t|p = l\sigma$ and $u = t[r\sigma]_p$. We call a *rewrite step* for the binary relation $\to_{\mathcal{R}}$ and such a position p a *rewrite position.* Especially, if every strict subterms in $\ell\sigma$ is a normal form, the binary relation is called an *innermost rewrite step*, denoted by $\xrightarrow{i}_{\mathcal{R}}$.

Definition 26. Let $\mathcal{R}$ be a TRS. We write $\mathsf{NF}(\mathcal{R})$ for the set of all normal forms with respect to $\to_{\mathcal{R}}$. The unique normal form of t with respect to $\to_{\mathcal{R}}$ is denoted by $t{\downarrow_{\mathcal{R}}}$.

Definition 27. We say that a TRS $\mathcal{R}$ is *terminating*, *confluent* and *complete* if the rewrite step $\to_{\mathcal{R}}$ is terminating, confluent and complete, respectively.

Definition 28. A term t is *linear* if no variable occurs twice in t. A TRS $\mathcal{R}$ is *left-linear* if every left-hand side of the rules in $\mathcal{R}$ is linear.

Definition 29. Terms s and t are *unifiable* if there exists σ such that $s\sigma = t\sigma$.

Definition 30. A substitution ρ is called a *renaming* if every image of ρ is a variable and it is bijective on $\mathcal{V}$. For a rewrite rule $\ell \to r$ and a renaming ρ, we call $\ell\rho \to r\rho$ a *variant* of the rule $\ell \to r$. We sometimes write $\ell_1 \to r_1 \doteq \ell_2 \to r_2$ for $\ell_2 \to r_2$ being a renamed variant of $\ell_1 \to r_1$.

Definition 31. A TRS $\mathcal{R}$ is *overlapping* if there exist $\ell_1 \to r_1, \ell_2 \to r_2 \in \mathcal{R}$, renaming substitutions ρ_1, ρ_2 and $p \in \mathit{Pos}(\ell_2)$ such that:

- $\ell_1\rho_1$ and $\ell_2\rho_2|_p$ are unifiable,
- $p \in \mathit{Pos}_{\mathcal{F}}(\ell_2)$, and
- if $\ell_2 \to r_2$ is a renamed variant of $\ell_1 \to r_1$ then $p \neq \varepsilon$.

Definition 32. A TRS $\mathcal{R}$ is *orthogonal* if $\mathcal{R}$ is left-linear and non-overlapping.

Definition 33. Let p and q be positions. The *leftmost-outermost order* $p >_{\mathsf{lo}} q$ holds if either of the following conditions holds.

- $p \neq \varepsilon$ and $q = \varepsilon$, or
- $p = ip'$, $q = jq'$, and
  - $i > j$ or
  - $i = j$ and $p' >_{\mathsf{lo}} q'$

Definition 34. We say that a term t is *left-normal* if, for all positions $p \in \mathit{Pos}_{\mathcal{V}}(t)$

and $q \in Pos(t)$, if $p <_{\mathsf{lo}} q$ then $q \in Pos_{\mathcal{V}}(t)$. A TRS is *left-normal* if every left-hand side of its rules is left-normal.

2.4 Needed strategy

At the end of this chapter, we introduce definitions and properties related to the needed strategy and the leftmost-outermost strategy in this section, and strong sequentiality in the next section. These topics constitute the centre of this thesis.

Definition 35. Let R be a TRS over a signature $\mathcal{F}$ and $\mathcal{R}_\bullet$ be a TRS over the signature $\mathcal{F} \uplus \bullet$ defined by $\mathcal{R}_\bullet = \mathcal{R} \cup \{\bullet \to \bullet\}$. A position p is *needed* in t if there exists no term u such that $t[\bullet]_p \to^!_{\mathcal{R}_\bullet} u$.

Example 36. Consider the TRS $\mathcal{R}$

$$
\begin{array}{rrlrr}
1: & \mathsf{take}(x : xs, \mathsf{0}) \to \mathsf{nil} & & 3: & \mathsf{from}(x) \to x : \mathsf{from}(\mathsf{s}(x)) \\
2: & \mathsf{take}(x : xs, \mathsf{s}(y)) \to x : \mathsf{take}(xs, y) & & 4: & \mathsf{0} + x \to x
\end{array}
$$

and the term $t = \mathsf{take}(\mathsf{from}(x), \mathsf{0} + \mathsf{0})$. The positions 1 and 2 of t are needed because we have infinite rewrite sequences

$$
\begin{aligned}
t[\bullet]_1 &= \mathsf{take}(\bullet, \mathsf{0} + \mathsf{0}) \to \mathsf{take}(\bullet, \mathsf{0}) \to \cdots \\
t[\bullet]_2 &= \mathsf{take}(\mathsf{from}(x), \bullet) \to \mathsf{take}(x : \mathsf{from}(\mathsf{s}(x)), \bullet) \to \cdots
\end{aligned}
$$

whilst the position 1.1 is not needed because we have:

$$
\begin{aligned}
t[\bullet]_{1.1} &= \mathsf{take}(\mathsf{from}(\bullet), \mathsf{0} + \mathsf{0}) \to \mathsf{take}(\bullet : \mathsf{from}(\mathsf{s}(\bullet)), \mathsf{0} + \mathsf{0}) \\
&\to \mathsf{take}(\bullet : \mathsf{from}(\mathsf{s}(\bullet)), \mathsf{0}) \\
&\to \mathsf{nil}
\end{aligned}
$$

Definition 37. The *needed strategy* $\xrightarrow{\mathsf{N}}$ is a rewrite step at a needed position.

Example 38. Consider the TRS $\mathcal{R}$ and the term from Example 36. The needed strategy yields the following rewrite sequence, rewriting the underlined subterms.

$$
\begin{aligned}
t = \mathsf{take}(\underline{\mathsf{from}(x)}, \mathsf{0} + \mathsf{0}) &\xrightarrow{\mathsf{N}} \mathsf{take}(x : \mathsf{from}(\mathsf{s}(x)), \underline{\mathsf{0} + \mathsf{0}}) \\
&\xrightarrow{\mathsf{N}} \underline{\mathsf{take}(x : \mathsf{from}(\mathsf{s}(x)), \mathsf{0})} \\
&\xrightarrow{\mathsf{N}} \mathsf{nil} \in \mathsf{NF}(\mathcal{R})
\end{aligned}
$$

We prepare two more definitions about rewrite steps.

Definition 39. A rewrite step $\rightsquigarrow$ is a *rewrite strategy* of $\to$ if $\rightsquigarrow \subseteq \to$ and $\mathsf{NF}(\rightsquigarrow) = \mathsf{NF}(\to)$ holds.

The needed strategy is not always a rewrite strategy.

Example 40. Consider the TRS $\mathcal{R}$

$$\begin{aligned} 1: &\quad \mathsf{or}(\mathsf{T}, y) \to \mathsf{T} \\ 2: &\quad \mathsf{or}(x, \mathsf{T}) \to \mathsf{T} \end{aligned}$$

and the term $t = \mathsf{or}(\mathsf{or}(\mathsf{T},\mathsf{F}),\mathsf{or}(\mathsf{F},\mathsf{T}))$. We easily have a rewrite steps $t = \mathsf{or}(\mathsf{or}(\mathsf{T},\mathsf{F}),\mathsf{or}(\mathsf{F},\mathsf{T})) \to \mathsf{or}(\mathsf{T},\mathsf{or}(\mathsf{F},\mathsf{T})) \to \mathsf{T} \in \mathsf{NF}(\mathcal{R})$. However, the needed strategy yields $t = \mathsf{or}(\mathsf{or}(\mathsf{T},\mathsf{F}),\mathsf{or}(\mathsf{F},\mathsf{T})) \in \mathsf{NF}(\mathcal{R})$, and thus $\mathsf{NF}(\to) \neq \mathsf{NF}(\xrightarrow{\mathsf{N}})$. It is because no position of t is needed:

$$\begin{aligned} t[\bullet]_1 &= \mathsf{or}(\bullet,\mathsf{or}(\mathsf{F},\mathsf{T})) \to \mathsf{or}(\bullet,\mathsf{T}) \to \mathsf{T} \\ t[\bullet]_2 &= \mathsf{or}(\mathsf{or}(\mathsf{T},\mathsf{F}),\bullet) \to \mathsf{or}(\mathsf{T},\bullet) \to \mathsf{T} \\ t[\bullet]_{1.1} &= \mathsf{or}(\mathsf{or}(\bullet,\mathsf{F}),\mathsf{or}(\mathsf{F},\mathsf{T})) \to \mathsf{or}(\mathsf{or}(\bullet,\mathsf{F}),\mathsf{T})) \to \mathsf{T} \\ t[\bullet]_{1.2} &= \mathsf{or}(\mathsf{or}(\mathsf{T},\bullet),\mathsf{or}(\mathsf{F},\mathsf{T})) \to \mathsf{or}(\mathsf{T},\mathsf{or}(\mathsf{F},\mathsf{T})) \to \mathsf{T} \\ t[\bullet]_{2.1} &= \mathsf{or}(\mathsf{or}(\mathsf{T},\mathsf{F}),\mathsf{or}(\bullet,\mathsf{T})) \to \mathsf{or}(\mathsf{or}(\mathsf{T},\mathsf{F}),\mathsf{T}) \to \mathsf{T} \\ t[\bullet]_{2.2} &= \mathsf{or}(\mathsf{or}(\mathsf{T},\mathsf{F}),\mathsf{or}(\mathsf{F},\bullet)) \to \mathsf{or}(\mathsf{T},\mathsf{or}(\mathsf{F},\bullet)) \to \mathsf{T} \end{aligned}$$

Hence we may obtain different results, depending on whether we use the needed strategy or other strategies. However, orthogonality prevents such situations.

Theorem 41 ([5])**.** *The needed strategy is a rewrite strategy for every orthogonal TRS.*

Definition 42. A rewrite strategy $\rightsquigarrow$ is a *normalising strategy* if every rewrite step starting from a normalising term is terminating with $\rightsquigarrow$.

Theorem 43 ([5])**.** *The needed strategy is a normalising strategy for every orthogonal TRS.*

Next, we introduce the leftmost-outermost strategy.

Definition 44. The leftmost-outermost strategy $\xrightarrow{\mathsf{lo}}$ is a rewrite step at the minimum rewrite position with respect to $>_{\mathsf{lo}}$.

Example 45. Consider the TRS $\mathcal{R}$ and the term t from Example 36. The term t is normalising because it has a normal form nil, and the needed strategy is a normalising strategy. However the leftmost-outermost strategy is not:

$$\begin{aligned} t = \mathsf{take}(\underline{\mathsf{from}(x)}, \mathsf{0}+\mathsf{0}) &\xrightarrow{\mathsf{lo}} \mathsf{take}(x : \underline{\mathsf{from}(\mathsf{s}(x))}, \mathsf{0}+\mathsf{0}) \\ &\xrightarrow{\mathsf{lo}} \mathsf{take}(x : (\mathsf{s}(x) : \underline{\mathsf{from}(\mathsf{s}(\mathsf{s}(x)))}), \mathsf{0}+\mathsf{0}) \xrightarrow{\mathsf{lo}} \cdots \end{aligned}$$

The leftmost-outermost strategy leads a term to a normal form in *left-normal* and orthogonal TRSs, provided that a given term has a normal form.

Theorem 46 (The Normalization Theorem [9])**.** *For every left-normal orthogonal TRS $\mathcal{R}$, if t has a normal form u then $t \xrightarrow{\mathsf{lo}}{}^{*}_{\mathcal{R}}\, u$.*

2.5 Strong sequentiality

Huet and Lévy [5] showed that every term has a needed position in an orthogonal TRS, however, and needed positions are uncomputable in general. In order to compute needed positions, they introduced strong sequentiality. Hereafter, we see how to show strong sequentiality of orthogonal TRSs. Firstly we refer [5] for Ω-terms, a fundamental material on strong sequentiality.

Definition 47. Terms over the signature $\mathcal{F} \uplus \{\Omega^{(0)}\}$ are called Ω*-terms.* An Ω*-NF* is an Ω-term without any redexes.

Definition 48. The *prefix order* $\leq_\Omega$ on Ω-terms is defined as follows.

- $x \leq_\Omega x$ for every $x \in \mathcal{V}$
- $\Omega \leq_\Omega t$ for every Ω-term t
- $f(s_1, \ldots, s_n) \leq_\Omega f(t_1, \ldots, t_n)$ if $s_1 \leq_\Omega t_1, \ldots, s_n \leq_\Omega t_n$

Definition 49. A position p of a term t is called an Ω*-position* if $t|_p = \Omega$. We write $\mathit{Pos}_\Omega(t)$ for the set of all Ω-positions of t.

Durand and Middeldorp [3] proposed an efficient way to find indices, which are required to show strong sequentiality, introducing an extended TRS called strong approximation and a predicate nf_{s}.

Definition 50. The *strong approximation* $\mathcal{R}_{\mathsf{s}}$ of a TRS $\mathcal{R}$ is obtained by replacing the right-hand sides of every rule in $\mathcal{R}$ by a fresh variable.

Example 51. Consider the TRS $\mathcal{R}$ from Example 36. The strong approximation $\mathcal{R}_{\mathsf{s}}$ consists of the following rules.

$$\begin{array}{rrrr} 1\colon & \mathsf{take}(x : xs, \mathsf{0}) \to z & 3\colon & \mathsf{from}(x) \to z \\ 2\colon & \mathsf{take}(x : xs, \mathsf{s}(y)) \to z & 4\colon & \mathsf{0} + x \to z \end{array}$$

Definition 52. A predicate $nf_{\mathsf{s}}(t)$ on Ω-terms holds if $t \to^*_{\mathcal{R}_{\mathsf{s}}} u$ for some normal form $u \in \mathcal{T}(\mathcal{F}, \mathcal{V})$.

Indices with respect to nf_{s} is defined as follows.

Definition 53. An Ω-position p of an Ω-term t is an *index* if $u|_p \neq \Omega$ for every Ω-term u such that $t \leq_\Omega u$ and $nf_{\mathsf{s}}(u)$ hold.

Example 54. Consider the TRS $\mathcal{R}$ from Example 36, the strong approximation $\mathcal{R}_{\mathsf{s}}$ and the Ω-term $t = \mathsf{take}(\Omega, \mathsf{0} + \Omega)$. The Ω-position 1 is an index, whilst 2.2 is not, because we have $\mathsf{take}(x : xs, \mathsf{0} + \underline{\Omega}) \to_{\mathcal{R}_{\mathsf{s}}} \mathsf{take}(x : xs, \mathsf{0}) \to_{\mathcal{R}_{\mathsf{s}}} \mathsf{0}$.

Moreover, they defined s-sequentiality and showed that the class of s-sequential TRSs coincides the class of strongly sequential TRSs.

Definition 55. A TRS $\mathcal{R}$ is s-*sequential* if every Ω-normal form has an index with respect to nf_{s} in $\mathcal{R}$.

Theorem 56. *An orthogonal TRS $\mathcal{R}$ is* strongly sequential *if every Ω-NF has an index with respect to nf_{s}.*

Strong approximation enables us to indicate indices without considering the right hand sides of TRSs, and thus we can find them easily.

Needed positions and indices look similar, but differ. Nevertheless, in orthogonal TRSs there is an important relation between them.

Theorem 57 ([5])**.** *Let $\mathcal{R}$ be a left-linear TRS. If an Ω-position p in an Ω-term t is an index with respect to nf_{s}, the Ω-position p is a needed position in a term u for every term u such that $t \leq_\Omega u$ and p is a rewrite position of u.*

Indices, that is to say, can be needed positions in left-linear TRSs. This theorem holds also in orthogonal TRSs since they are left-linear. However, it has been still hard to show strong sequentiality because we need to find indices of *every* Ω-NF. Klop and Middeldorp [6] proposed another technique to show strong sequentiality, adding one more condition to orthogonal TRSs: being constructor system.

Definition 58. Let $\mathcal{R}$ be a TRS.

- The set $\Sigma_{\mathcal{R}}$ consists of all function symbols in $\mathcal{R}$.
- The set of *defined symbols* $\mathcal{D}_{\mathcal{R}}$ of $\mathcal{R}$ is defined by $\mathcal{D}_{\mathcal{R}} = \{\mathsf{root}(\ell) \mid \ell \to r \in \mathcal{R}\}$.
- The set of *constructor symbols* $\mathcal{C}_{\mathcal{R}}$ of $\mathcal{R}$ is defined by $\mathcal{C}_{\mathcal{R}} = \Sigma_{\mathcal{R}} \setminus \mathcal{D}_{\mathcal{R}}$.

A term that consists of constructor symbols and variables is called a *constructor term*.

Definition 59. A TRS $\mathcal{R}$ is a *constructor system* if the term $\ell|_p$ is a constructor term for every rule $\ell \to r \in \mathcal{R}$ and every position $p \in \mathit{Pos}_{\mathcal{F}}(\ell) \setminus \{\varepsilon\}$.

Example 60. Consider the TRS $\mathcal{R}$ from Example 36:

$$\begin{array}{rrcl}
1: & \mathsf{take}(x : xs, \mathsf{0}) \to \mathsf{nil} & 3: & \mathsf{from}(x) \to x : \mathsf{from}(\mathsf{s}(x)) \\
2: & \mathsf{take}(x : xs, \mathsf{s}(y)) \to x : \mathsf{take}(xs, y) & 4: & \mathsf{0} + x \to x
\end{array}$$

We have $\Sigma_{\mathcal{R}} = \{\mathsf{take}, \mathsf{from}, +, :, \mathsf{0}, \mathsf{s}, \mathsf{nil}\}$, $\mathcal{D}_{\mathcal{R}} = \{\mathsf{take}, \mathsf{from}, +\}$ and $\mathcal{C}_{\mathcal{R}} = \{:, \mathsf{0}, \mathsf{s}, \mathsf{nil}\}$. No defined symbols occur at positions in every rule excepting ε, and thus $\mathcal{R}$ is a constructor system.

A key feature of the technique is that we need to consider only limited Ω-terms, which is called preredexes and obtained by a given TRS.

Definition 61.

- A *redex scheme* is a term that all variables occurring in the left-hand sides of rewrite rules are replaced by Ω.
- An Ω-term t is a *preredex* if there exists a redex scheme u such that $t \leq_\Omega u$.
- A preredex is *proper* if it is neither a redex scheme nor Ω.

We refer to [6] for a little more definitions and properties about Ω-terms.

Definition 62. The Ω-*reduction* $\to_\Omega$ is defined as follows.

$$t \to_\Omega t[\Omega]_p$$

for every redex compatible subterm u and position p of t such that $t|_p = u$.

Definition 63 ([5])**.** The *direct approximant* $\omega(t)$ of an Ω-term t is the normal form of t with respect to Ω-reduction. An Ω-term t is called *soft* if $\omega(t) = \Omega$.

Lemma 64. *Let s and t be Ω-terms, and p an Ω-position of t.*

- $\omega(t) \leq_\Omega t$.
- $\omega(t) = \omega(t[\omega(t|_p)]_p)$.
- $\omega(\omega(t)) = \omega(t)$.
- *If t is redex compatible then $\omega(t) = \Omega$.*

Lemma 65. *Let $\mathcal{R}$ be a TRS, t an Ω-term and p an Ω-position of t. The following three statements are equivalent.*

1. $p \in \mathcal{I}_\mathcal{R}(t)$.
2. $\omega(t[\bullet]_p) \neq \omega(t)$.
3. $p \in Pos_\Omega(\omega(t[\bullet]_p))$.

Definition 66. Let t be a soft Ω-term and

$$t = t_0 \to_\Omega t_1 \to_\Omega \ldots \to_\Omega t_n = \Omega$$

an arbitrary Ω-reduction from t to Ω, and suppose that in step $t_i \to_\Omega t_{i+1}$ the redex compatible term at position p_i is replaced by Ω. The set defined by $\{(p_i, t_i|p_i) \mid 0 \leq i \leq n-1\}$ is called a *decomposition* of t.

Example 67. Consider the ATRS $\mathcal{R}$

1:	$\mathsf{map}\ f\ \mathsf{nil} \to \mathsf{nil}$	5:	$+\ \mathsf{0} \to \mathsf{id}$
2:	$\mathsf{map}\ f\ (:\ x\ xs) \to\ :\ (f\ x)\ (\mathsf{map}\ f\ xs)$	6:	$+\ (\mathsf{s}\ x)\ y \to \mathsf{s}\ (+\ x\ y)$
3:	$\mathsf{take}\ \mathsf{0}\ xs \to \mathsf{nil}$	7:	$\mathsf{id}\ x \to x$
4:	$\mathsf{take}\ (\mathsf{s}\ x)\ (:\ y\ ys) \to\ :\ y\ (\mathsf{take}\ y\ xs)$		

and the Ω-term $t = \mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ \mathsf{0}))\ (\mathsf{map}\ \mathsf{id}\ \Omega)$. For example, we have the following Ω-reduction sequence, in which the underlined subterms are reduced.

$$t = \mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ \mathsf{0}))\ (\underline{\mathsf{map}\ \mathsf{id}\ \Omega}) \xrightarrow{2}_\Omega \mathsf{take}\ (\underline{+\ \Omega\ (\mathsf{s}\ \mathsf{0})})\ \Omega \xrightarrow{1.2}_\Omega \underline{\mathsf{take}\ \Omega\ \Omega} \xrightarrow{\epsilon}_\Omega \Omega$$

As $\omega(t) = \Omega$, and thus t is soft. We obtain the following decomposition D of t.

$$D = \{(\mathsf{map}\ \mathsf{id}\ \Omega, 2), (+\ \Omega\ (\mathsf{s}\ 0), 1.2), (\mathsf{take}\ \Omega\ \Omega, \varepsilon)\}$$

Note that decompositions vary by Ω-reduction sequences which we obtain. Here are an Ω-reduction and the decomposition D' in another case.

$$t = \underline{\mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ 0))\ (\mathsf{map}\ \mathsf{id}\ \Omega)} \xrightarrow{\epsilon}_{\Omega} \Omega$$

$$D' = \{(\mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ 0))\ (\mathsf{map}\ \mathsf{id}\ \Omega), \varepsilon)\}$$

Definition 68. Let t be an Ω-term and suppose that t is soft. *Procrustes cut* $\mathsf{cut}(t)$ and the set of positions Pos_{cut} where subterms are cut by $\mathsf{cut}(t)$ are defined as follows.

- $\mathsf{cut}(t) = t \cap r_1 \cap \cdots \cap r_n$
- $\overline{Pos}(t) = \{p \mid p \in Pos(t) \text{ and } t|_p \neq \Omega\}$
- $Pos_{cut}(t) = \overline{Pos}(t) \cap Pos_{\Omega}(\mathsf{cut}(t))$

where $\{r_1, \ldots, r_n\}$ is the set of all redex schemes compatible with t.

Example 69. Consider the ATRS $\mathcal{R}$ from Example 67 and the Ω-term $t = \mathsf{map}\ \mathsf{id}\ \Omega$. The redex schemes of $\mathcal{R}$ are the following seven Ω-terms.

$$\begin{array}{ll}
\mathsf{map}\ \Omega\ \mathsf{nil} & +\ 0 \\
\mathsf{map}\ \Omega\ (:\ x\ \Omega) & +\ (\mathsf{s}\ \Omega)\ \Omega \\
\mathsf{take}\ 0\ \Omega & \mathsf{id}\ \Omega \\
\mathsf{take}\ (\mathsf{s}\ \Omega)\ (:\ \Omega\ \Omega) &
\end{array}$$

The Ω-term $t = \mathsf{map}\ \mathsf{id}\ \Omega$ is compatible with the redex schemes $\mathsf{map}\ \Omega\ \mathsf{nil}$ and $\mathsf{map}\ \Omega\ (:\ x\ \Omega)$. We obtain $\mathsf{cut}(t) = (\mathsf{map}\ \mathsf{id}\ \Omega) \cap (\mathsf{map}\ \Omega\ \mathsf{nil}) \cap (\mathsf{map}\ \Omega\ (:\ x\ \Omega))$ as follows.

$$\begin{array}{cccc}
 & \mathsf{map} & \mathsf{id} & \Omega \\
\cap & \mathsf{map} & \Omega & \mathsf{nil} \\
\hline
 & \mathsf{map} & \Omega & \Omega \\
\cap & \mathsf{map} & \Omega & (:\ x\ \Omega) \\
\hline
 & \mathsf{map} & \Omega & \Omega
\end{array}$$

Hence $\mathsf{cut}(t) = \mathsf{map}\ \Omega\ \Omega$, which yields $Pos_{cut}(t) = \{1.2\}$ with $\overline{Pos}(t) = \{1.2\}$.

Definition 70. Let D be a decomposition of a soft Ω-term t. *Procrustes procedure* $t \rightarrow_{cut} t'$ holds if $t' = t[\Omega]_{pq}$ for some $q \in Pos_{cut}(s)$ and $(s, p) \in D$ such that

$\mathsf{cut}(s) \neq s$. The set of all normal forms with respect to $\rightarrow_{cut}$ excepting Ω is denoted by NF_{cut}.

Example 71. Consider the ATRS $\mathcal{R}$ from Example 67 and the Ω-term $t = \mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ 0))\ (\mathsf{map}\ \mathsf{id}\ \Omega)$. As we have seen in Example 67, we can obtain the following Ω-reduction sequence and the decomposition D.

$$t = \mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ 0))\ (\underline{\mathsf{map}\ \mathsf{id}\ \Omega}) \xrightarrow{2}_{\Omega} \mathsf{take}\ (\underline{+\ \Omega\ (\mathsf{s}\ 0)})\ \Omega \xrightarrow{1.2}_{\Omega} \underline{\mathsf{take}\ \Omega\ \Omega} \xrightarrow{\epsilon}_{\Omega} \Omega$$

$$D = \{(\mathsf{map}\ \mathsf{id}\ \Omega, 2), (+\ \Omega\ (\mathsf{s}\ 0), 1.2), (\mathsf{take}\ \Omega\ \Omega, \varepsilon)\}$$

In this case, the Procrustes procedure from the decomposition D is obtained as follows.

$$\begin{aligned} t &= \mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ 0))\ (\mathsf{map}\ \underline{\mathsf{id}}\ \Omega) \\ &\rightarrow_{cut} \mathsf{take}\ (+\ \Omega\ (\underline{\mathsf{s}\ 0}))\ (\mathsf{map}\ \Omega\ \Omega) \\ &\rightarrow_{cut} \mathsf{take}\ (+\ \Omega\ \Omega)\ (\mathsf{map}\ \Omega\ \Omega) \in NF_{cut} \end{aligned}$$

The next two lemmata claim that strong sequentiality of TRSs can be shown by finding an index of every element t of NF_{cut} and every decomposition of t is a proper preredex of a TRS. That is, we can show existence of an index of t if every proper preredex has an index.

Lemma 72. *A TRS is strongly sequential if and only if every NF_{cut} has index.*

Lemma 73. *If $t \in NF_{cut}$ and u is a decomposition of t then u is a proper preredex.*

It is known that $pq \in \mathcal{I}_{\mathcal{R}}(t[u]_p)$ does not always hold even if $p \in \mathcal{I}_{\mathcal{R}}(t)$ and $q \in \mathcal{I}_{\mathcal{R}}(u)$ for a TRS $\mathcal{R}$, and terms s and t. However, it *definitely* holds in constructor systems.

Lemma 74. *Let $\mathcal{R}$ be a constructor system and s, t Ω-terms. If $p \in \mathcal{I}_{\mathcal{R}}(s)$ and $q \in \mathcal{I}_{\mathcal{R}}(t)$ then $pq \in \mathcal{I}_{\mathcal{R}}(s[t]_p)$.*

Since decompositions of $t \in NF_{cut}$ are proper preredexes, t is a 'composition' of them. By this lemma on index transitivity, they showed the following key lemma.

Lemma 75. *An orthogonal constructor system is strongly sequential if and only if every proper preredex has index.*

Lemma 75 suggests that in orthogonal *constructor* TRSs, all we need to do is find indices of their proper preredexes, not of every Ω-NF.

Furthermore, they proposed another method to find indices of Ω-terms for constructor systems, which is based on compatibility of Ω-terms.

Definition 76. Two Ω-terms t_1 and t_2 are *compatible* if there exists an Ω-term t such that $t_1 \leq_\Omega t$ and $t_2 \leq_\Omega t$. We write $t_1 \uparrow_\Omega t_2$ for such compatibility.

Definition 77. An Ω-term t is *redex compatible* if $t \leq_\Omega u$ for some redex u.

Lemma 78. *Let t be a proper preredex in a constructor system. An Ω-position p is an index if and only if $t[\bullet]_p$ is not redex compatible.*

Therefore, in orthogonal constructor systems, we can determine whether an Ω-position p of an proper preredex t is an index or not by checking whether $t[\bullet]_p$ is redex compatible. That is to say, if $t[\bullet]_p$ is an instance of any left-hand side ℓ in a TRS, which is equivalent to $t[\bullet]_p \uparrow_\Omega \ell$, the Ω-position p is an index.

Example 79. Consider the TRS $\mathcal{R}$ from Example 36. The proper preredex are $\mathsf{take}(\Omega : \Omega, \Omega)$, $\mathsf{take}(\Omega, \mathsf{s}(\Omega))$, $\mathsf{take}(\Omega, \Omega)$ and $\Omega + \Omega$.

- In $\mathsf{take}(\Omega : \Omega, \Omega)$, the Ω-position 2 is an index because $\mathsf{take}(\Omega : \Omega, \bullet)$ is not redex compatible. However, the Ω-positions 1.1 and 1.2 is not an index because $\mathsf{take}(\bullet : \Omega, \Omega)$ and $\mathsf{take}(\Omega : \bullet, \Omega)$ can be refined to redexes $\mathsf{take}(\bullet : \Omega, \mathsf{0})$ and $\mathsf{take}(\Omega : \bullet, \mathsf{0})$, respectively.
- In $\mathsf{take}(\Omega, \mathsf{s}(\Omega))$, the Ω-position 1 is an index.
- In $\mathsf{take}(\Omega, \Omega)$, the Ω-positions 1 and 2 are indices because $\mathsf{take}(\bullet, \Omega)$ and $\mathsf{take}(\Omega, \bullet)$ are not redex compatible.
- In $\Omega + \Omega$, the Ω-position 1 is index because $\bullet + \Omega$ is not redex compatible.

Since every proper preredex has an index, $\mathcal{R}$ is strongly sequential.

As we have seen, even though finding indices is difficult, in orthogonal constructor systems it is relatively easy to find indices and show strong sequentiality. It is also notable that strong sequentiality ensures existence of indices of every proper preredex.

Chapter 3

Left-normal translation for ATRSs

In this chapter, we first discuss strong sequentiality in ATRSs. After, we shall define left-normal translation for ATRSs and show its correctness, including strong sequentiality of translated ATRSs. .

3.1 Strong sequentiality in ATRSs

Let us begin by preparing the definitions on ATRSs.

Definition 80. *Applicative term* is terms over a signature consisting of constants and a single binary function symbol $\circ$. *Applicative term rewrite system (ATRS)* are TRSs consisting of applicative terms.

TRSs other than ATRSs are called *functional TRSs*. The binary function symbol $\circ$ is usually denoted by left-associative infix notation like $t_1 \circ t_2 \circ t_3$ for $\circ(\circ(t_1, t_2), t_3)$.

The fact that the set of ATRSs is a subset of the set of TRSs is likely to fit Lemma 75 from [6] for determination of strong sequentiality of ATRSs. However, we cannot use the method. It stems from the definition of constructor system.

Example 81. Consider the ATRS $\mathcal{R}$.

$$
\begin{array}{rrclrrcl}
1: & \mathsf{map}\ f\ \mathsf{nil} & \to & \mathsf{nil} & 5: & +\ 0 & \to & \mathsf{id} \\
2: & \mathsf{map}\ f\ (:\ x\ xs) & \to & :\ (f\ x)\ (\mathsf{map}\ f\ xs) & 6: & +\ (\mathsf{s}\ x)\ y & \to & \mathsf{s}\ (+\ x\ y) \\
3: & \mathsf{take}\ 0\ xs & \to & \mathsf{nil} & 7: & \mathsf{id}\ x & \to & x \\
4: & \mathsf{take}\ (\mathsf{s}\ x)\ (:\ y\ ys) & \to & :\ y\ (\mathsf{take}\ y\ xs) & & & &
\end{array}
$$

The set of defined symbols $\mathcal{D}_{\mathcal{R}}$ apparently consists of map, take, $+$ and id that occur on the 'left' of the left-hand sides, but it does not. Recalling the definition of defined and constructor symbols, we have the following sets $\mathcal{D}_{\mathcal{R}}$ and $\mathcal{C}_{\mathcal{R}}$.

$$
\begin{aligned}
\mathcal{D}_{\mathcal{R}} &= \{\circ\} \\
\mathcal{C}_{\mathcal{R}} &= \{\mathsf{map}, \mathsf{nil}, :, \mathsf{take}, \mathsf{0}, \mathsf{s}, +, \mathsf{id}\}
\end{aligned}
$$

For example, $\mathsf{map}\ f\ \mathsf{nil}$ is denoted by $\circ(\circ(\mathsf{map}, f), \mathsf{nil})$ without left-associative infix notation, and thus its root symbol is $\circ$. The symbol may occur at positions excepting the root positions as we can see the subterm $\circ(\mathsf{map}, f)$. Therefore, the ATRS $\mathcal{R}$ is not a constructor system, and also we cannot show strong sequentiality of $\mathcal{R}$ even if we could find indices of every proper preredex.

In order to overcome this problem, we prepare a concept with two parts: (1) defining a way to handle efficiently symbols in ATRSs excepting $\circ$ and (2) providing a method to show strong sequentiality that fits the concept.

Firstly, we introduce an applicative version of constructor systems.

Definition 82. The *head symbol* of an applicative term t is defined as follows.

$$
\mathsf{head}(t) = \begin{cases} \mathsf{head}(u) & \text{if } t = u \circ v \\ t & \text{otherwise} \end{cases}
$$

Example 83. Consider the applicative term t from Example 97.

$$
\begin{aligned}
\mathsf{head}(t) &= \mathsf{head}(\mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ \mathsf{0}))\ (\mathsf{map}\ \mathsf{id}\ \Omega)) \\
&= \mathsf{head}(\mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ \mathsf{0}))) \\
&= \mathsf{head}(\mathsf{take}) \\
&= \mathsf{take}
\end{aligned}
$$

Definition 84. Let $\mathcal{R}$ be an ATRS.

- The set of signatures $\mathcal{A}\Sigma_{\mathcal{R}}$ consists of all function symbols in $\mathcal{R}$.
- The set of *applicative defined symbols* $\mathcal{AD}_{\mathcal{R}}$ is defined by $\mathcal{AD}_{\mathcal{R}} = \{\mathsf{head}(\ell) \mid \ell \to r \in \mathcal{R}\}$.
- The set of *applicative constructor symbols* $\mathcal{AC}_{\mathcal{R}}$ is defined by $\mathcal{AC}_{\mathcal{R}} = \mathcal{A}\Sigma_{\mathcal{R}} \setminus \mathcal{AD}_{\mathcal{R}}$.

An applicative term t is called an *applicative constructor term* if $t \in \mathcal{T}(\mathcal{AC}_{\mathcal{R}}, \mathcal{V})$.

Example 85. Consider the ATRS $\mathcal{R}$ from Example 81.

$$\begin{array}{lrlr}
1: & \mathsf{map}\ f\ \mathsf{nil} \to \mathsf{nil} & 5: & +\ \mathsf{0} \to \mathsf{id} \\
2: & \mathsf{map}\ f\ (:\ x\ xs) \to :\ (f\ x)\ (\mathsf{map}\ f\ xs) & 6: & +\ (\mathsf{s}\ x)\ y \to \mathsf{s}\ (+\ x\ y) \\
3: & \mathsf{take}\ \mathsf{0}\ xs \to \mathsf{nil} & 7: & \mathsf{id}\ x \to x \\
4: & \mathsf{take}\ (\mathsf{s}\ x)\ (:\ y\ ys) \to :\ y\ (\mathsf{take}\ y\ xs) & &
\end{array}$$

For example, in the first rule we have $\mathsf{head}(\mathsf{map}\ f\ \mathsf{nil}) = \circ(\circ(\mathsf{map}, f), \mathsf{nil}) = \mathsf{map}$. Hence the symbol map is in the set of applicative defined symbols. Finally, we have the following sets $\mathcal{AC}_{\mathcal{R}}$ and $\mathcal{AD}_{\mathcal{R}}$.

$$\mathcal{AD}_{\mathcal{R}} = \{\mathsf{map}, \mathsf{take}, +, \mathsf{id}\}$$
$$\mathcal{AC}_{\mathcal{R}} = \{\mathsf{nil}, :, \mathsf{0}, \mathsf{s}\}$$

Definition 86. Let $\mathcal{R}$ be an ATRS and t be an applicative term. The term t is called an *applicative basic term* if $\mathsf{head}(t) \in \mathcal{AD}_{\mathcal{R}}$ and every applicative argument of t is an applicative constructor term.

Definition 87. A TRS $\mathcal{R}$ is an *applicative constructor system (ACS)* if ℓ is an applicative basic term for every rule $\ell \to r \in \mathcal{R}$.

Example 88. Consider the ATRS $\mathcal{R}$ from Example 81. As aforementioned, we have $\mathcal{AD}_{\mathcal{R}} = \{\mathsf{map}, \mathsf{take}, +, \mathsf{id}\}$ and $\mathcal{AC}_{\mathcal{R}} = \{\mathsf{nil}, :, \mathsf{0}, \mathsf{s}\}$. For every rule $\ell \to r$, we can see that ℓ is an applicative basic term, and thus $\mathcal{R}$ is an ACS.

Secondly, we propose a way to show strong sequentiality of orthogonal ACSs. Our method is basically based on [6] seen in the previous chapter. However, recalling that Lemmata 74 and 75 assume that orthogonal TRSs are constructor systems, we cannot say that these lemmata fit orthogonal *applicative* constructor systems. Hence the mission is proposing two lemmata: applicative versions of Lemmata 74 and 75.

Lemma 89. *Let $\mathcal{R}$ be an orthogonal ACS and s, t Ω-terms. If $p \in \mathcal{I}_{\mathcal{R}}(s)$, $q \in \mathcal{I}_{\mathcal{R}}(t)$ and t is redex compatible, then $pq \in \mathcal{I}_{\mathcal{R}}(s[t]_p)$.*

Proof. Assume that $p \in \mathcal{I}_{\mathcal{R}}(s)$, $q \in \mathcal{I}_{\mathcal{R}}(t)$ and t is redex compatible. We show $pq \in \mathcal{I}_{\mathcal{R}}(s[t]_p)$ by contradiction. Assume that $pq \notin \mathcal{I}_{\mathcal{R}}(s[t]_p)$, then we have $pq \notin \mathcal{P}os(\omega((s[t]_p)[\bullet]_{pq}))$. There exists an Ω-reduction

$$(s[t]_p)[\bullet]_{pq} \to_{\Omega}^{*} t_1 \to_{\Omega} t_2 \to_{\Omega}^{*} \omega((s[t]_p)[\bullet]_{pq})$$

such that $t_1|_{pq} = \bullet$ and $pq \notin Pos(t_2)$. Let $t_1|_{p'}$ be a redex compatible subterm contracted in the step $t_1 \to_\Omega t_2$. Clearly $p' < pq$. We distinguish two cases by p'.

- If $p \le p' < pq$, as $p \in Pos(t_2)$ holds, $((s[t]_p)[\bullet]_{pq})|_p \to_\Omega^* t_1 \to_\Omega t_2$ can be translated into

$$((s[t]_p)[\bullet]_{pq})|_p = ((s[t]_p)|_p)[\bullet]_{(pq|_p)} \to_\Omega^* t_1|_p \to_\Omega t_2|_p$$

  As $pq \notin Pos(t_2)$, $pq|_p \notin Pos(t_2|_p)$, we have $q \notin Pos(t_2|_p)$. By the definition of Ω-reduction, we have:

$$q \notin Pos(\omega(t_2|_p)) = Pos(\omega(((s[t]_p)|_p)[\bullet]_{(pq|_p)})) = Pos(\omega(t[\bullet]_q))$$

  From this it follows that $q \notin \mathcal{I}_\mathcal{R}(t)$, which contradicts the assumption $q \in \mathcal{I}_\mathcal{R}(t)$.
- If $p' < p$, let u be a redex scheme of $\mathcal{R}$ compatible with $t_1|_{p'}$. Firstly, as $\mathcal{R}$ is an ACS, for every position r in u other than the position corresponding to $\mathsf{head}(u)$, we have $u|_r \notin \mathcal{AD}_\mathcal{R}$. Secondly, the assumption $p' < p$ implies that $t_1|_p$ is redex compatible. Let v be a redex scheme of $\mathcal{R}$ compatible with t and $t_1|_p$. As $\mathcal{R}$ is an ACS, $\mathsf{head}(v) \in \mathcal{AD}_\mathcal{R}$. As $\mathcal{R}$ is orthogonal, there exists no $u' \in \mathsf{Arg}(u)$ such that $\mathsf{head}(u') = \mathsf{head}(v)$. It follows that $p|_{p'} \notin Pos(u)$ or $u|_{(p|_{p'})} = \Omega$, and then $(t_1[\bullet]_p)|_{p'}$ is compatible with r. Considering that the position p is preserved in the step $s[t]_p \to_\Omega^* t_1$, the Ω-reduction $(s[t]_p)[\bullet]_{pq} \to_\Omega^* t_1 \to_\Omega t_2$ can be translated into:

$$(s[t]_p)[\bullet]_{pq} \to_\Omega^* t_1[\bullet]_{pq} \to_\Omega (t_1[\bullet]_{pq})[\Omega]_{p'} = t_2$$

  From $p' < p$, we have $p \notin Pos(t_2)$. Hence $p \notin Pos(\omega(t_2)) = Pos(\omega((s[t]_p)[\bullet]_p)) = Pos(\omega(s[\bullet]_p))$. It therefore follows that $p \notin \mathcal{I}_\mathcal{R}(s)$, which contradicts the assumption $p \in \mathcal{I}_\mathcal{R}(s)$.

□

Lemma 90. *An orthogonal ACS is strongly sequential if and only if every proper preredex has index.*

Proof. For the 'if' direction, the claim follows by Lemmata 72 and 73. For the 'only if' direction, assume that every proper preredex has an index. By Lemma 72, to show strong sequentiality of $\mathcal{R}$, it suffices to show every Ω-term in NF_{cut} has an index. Let

t be an arbitrary Ω-term in NF_{cut} and D the decomposition of t. By the definition of decomposition, there exist an Ω-reduction and Ω-terms $t_1, \ldots, t_n$ such that

$$t = t_1 \xrightarrow{p_1}_\Omega t_2 \xrightarrow{p_2}_\Omega t_3 \xrightarrow{p_3}_\Omega \cdots \xrightarrow{p_{n-1}}_\Omega t_n \xrightarrow{p_n}_\Omega \Omega$$

and thus t can be denoted by:

$$t = t_1 = ((\Omega[t_n]_{p_n}) \ldots [t_2]_{p_2})$$

By Lemma 73, $t_1, \ldots, t_n$ are proper preredexes of $\mathcal{R}$. The assumption implies that each of them has an index. By Lemma 89, t has an index. $\square$

Together with our lemmata 89, 90 and lemmata 72, 73 by [6], a method to show strong sequentiality of orthogonal ACSs has been set up.

Moreover, we introduce additional concepts uniquely available for ATRSs in order to find indices more practically.

Definition 91. A term t is *head-variable-free* if there exist no terms t_1, t_2 such that $t \trianglerighteq \circ(t_1, t_2)$ and $t_1 \in \mathcal{V}$.

Definition 92. A TRS $\mathcal{R}$ is *left-head-variable-free* if the left-hand side ℓ is head-variable-free for every $\ell \to r \in \mathcal{R}$.

We usually write ℓ-ATRSs and ℓ-ACSs for left-head-variable-free ATRSs and left-head-variable-free ACSs, respectively.

Example 93. Consider the ATRS $\mathcal{R}$ from Example 81.

1:	$\underline{\mathsf{map}}\ f\ \mathsf{nil} \to \mathsf{nil}$	5:	$\underline{\mathsf{from}}\ x \to :\ x\ (\mathsf{from}\ (\mathsf{s}\ x))$
2:	$\underline{\mathsf{map}}\ f\ (\underline{:}\ x\ xs) \to :\ (f\ x)\ (\mathsf{map}\ f\ xs)$	6:	$\underline{+}\ 0\ x \to x$
3:	$\underline{\mathsf{take}}\ (\underline{:}\ x\ xs)\ 0 \to \mathsf{nil}$	7:	$\underline{\mathsf{id}}\ x \to x$
4:	$\underline{\mathsf{take}}\ (\underline{:}\ x\ xs)\ (\underline{\mathsf{s}}\ y) \to :\ x\ (\mathsf{take}\ xs\ y)$		

Since the underlined subterms are not variables, $\mathcal{R}$ is an ℓ-ATRS.

Definition 94. A term t is *head-Ω-free* if there exist no terms t_1 and t_2 such that $t \trianglerighteq \circ(t_1, t_2)$ and $t_1 = \Omega$.

ATRSs yield much more non-head-Ω-free proper preredexes than head-Ω-free ones, and then the number of indices which we need to find tends to be larger. It is because function symbols are constants, which can be replaced by Ω.

Example 95. Consider the ATRS $\mathcal{R}$:

$$
\begin{array}{rlrl}
1: & \mathsf{take}\ xs\ \mathsf{0} \to \mathsf{nil} & 3: & \mathsf{from}\ x \to\ :\ x\ (\mathsf{from}\ (\mathsf{s}\ x)) \\
2: & \mathsf{take}\ (:\ x\ xs)\ (\mathsf{s}\ y) \to\ :\ x\ (\mathsf{take}\ xs\ y) & 4: & +\ \mathsf{0}\ x \to x
\end{array}
$$

The redex schemes and the proper preredexes are as follows; only underlined proper preredexes are head-Ω-free.

redex scheme	$\mathsf{take}\ \Omega\ \mathsf{0}$	$\mathsf{take}\ (:\ \Omega\ \Omega)\ (\mathsf{s}\ \Omega)$	$\mathsf{from}\ \Omega$	$+\ \mathsf{0}\ \Omega$
proper preredex	$\underline{\mathsf{take}\ \Omega\ \mathsf{0}}$	$\underline{\mathsf{take}\ (:\ \Omega\ \Omega)\ (\mathsf{s}\ \Omega)}$	$\underline{\mathsf{from}\ \Omega}$	$\underline{+\ \mathsf{0}\ \Omega}$
	$\underline{\mathsf{take}\ \Omega\ \Omega}$	$\mathsf{take}\ (:\ \Omega\ \Omega)\ (\Omega\ \Omega)$	$\Omega\ \Omega$	$\underline{+\ \Omega\ \Omega}$
	$\Omega\ \Omega\ \mathsf{0}$	$\mathsf{take}\ (\Omega\ \Omega\ \Omega)\ (\mathsf{s}\ \Omega)$		$\Omega\ \mathsf{0}\ \Omega$
		$\Omega\ (:\ \Omega\ \Omega)\ (\mathsf{s}\ \Omega)$		$\Omega\ \Omega\ \Omega$
		$\mathsf{take}\ (\Omega\ \Omega\ \Omega)\ (\Omega\ \Omega)$		
		$\Omega\ (:\ \Omega\ \Omega)\ (\Omega\ \Omega)$		
		$\Omega\ (\Omega\ \Omega\ \Omega)\ (\Omega\ \Omega)$		
		$\underline{\mathsf{take}\ (:\ \Omega\ \Omega)\ \Omega}$		
		$\mathsf{take}\ (\Omega\ \Omega\ \Omega)\ \Omega$		
		$\Omega\ (:\ \Omega\ \Omega)\ \Omega$		
		$\Omega\ (\Omega\ \Omega\ \Omega)\ \Omega$		
		$\mathsf{take}\ (\Omega\ \Omega)\ \Omega$		
		$\Omega\ (\Omega\ \Omega)\ \Omega$		
		$\mathsf{take}\ (\Omega\ \Omega)\ (\mathsf{s}\ \Omega)$		
		$\mathsf{take}\ (\Omega\ \Omega)\ (\Omega\ \Omega)$		
		$\Omega\ (\Omega\ \Omega)\ (\mathsf{s}\ \Omega)$		
		$\Omega\ (\Omega\ \Omega)\ (\Omega\ \Omega)$		
		$\underline{\mathsf{take}\ \Omega\ (\mathsf{s}\ \Omega)}$		
		$\mathsf{take}\ \Omega\ (\Omega\ \Omega)$		
		$\Omega\ \Omega\ (\mathsf{s}\ \Omega)$		
		$\Omega\ \Omega\ (\Omega\ \Omega)$		
		$\Omega\ (\mathsf{s}\ \Omega)$		
		$\Omega\ (\Omega\ \Omega)$		

Fortunately, every non-head-Ω-free proper preredex definitely has an index if an orthogonal ℓ-ACS satisfies one more condition: being non-variadic.

Definition 96. Let t be an applicative term. The set of *applicative arguments* $\mathsf{Arg}(t)$ is defined as follows.

$$\mathsf{Arg}(t) = \begin{cases} \mathsf{Arg}(u) \cup \{v\} & \text{if } t = u \circ v \\ \varnothing & \text{otherwise} \end{cases}$$

Example 97. Consider the applicative term $t = \mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ 0))\ (\mathsf{map}\ \mathsf{id}\ \Omega)$.

$$\begin{aligned}
\mathsf{Arg}(t) &= \mathsf{Arg}(\mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ 0))\ (\mathsf{map}\ \mathsf{id}\ \Omega)) \\
&= \mathsf{Arg}(\mathsf{take}\ (+\ \Omega\ (\mathsf{s}\ 0))) \cup \{\mathsf{map}\ \mathsf{id}\ \Omega\} \\
&= \mathsf{Arg}(\mathsf{take}) \cup \{+\ \Omega\ (\mathsf{s}\ 0)\} \cup \{+\ 0\ 0\} \\
&= \varnothing \cup \{+\ \Omega\ (\mathsf{s}\ 0)\} \cup \{+\ 0\ 0\} \\
&= \{+\ \Omega\ (\mathsf{s}\ 0), +\ 0\ 0\}
\end{aligned}$$

Definition 98. An ATRS $\mathcal{R}$ is *non-variadic* if, for every two rules $\ell \to r$ and $\ell' \to r'$ in $\mathcal{R}$, if $\mathsf{head}(\ell) = \mathsf{head}(\ell')$ then $||\mathsf{Arg}(\ell)|| = ||\mathsf{Arg}(\ell')||$.

Example 99. Consider the ATRS $\mathcal{R}$ from Example 81. The rules 1 and 2, and the rules 3 and 4 are sets of rules whose left-hand sides have a same head symbol. Since $||\mathsf{Arg}(\mathsf{map}\ f\ \mathsf{nil})|| = ||\mathsf{Arg}(\mathsf{map}\ f\ (:\ x\ xs))|| = 2$ and $||\mathsf{Arg}(\mathsf{take}\ 0\ xs)|| = ||\mathsf{Arg}(\mathsf{take}\ (\mathsf{s}\ x)\ (:\ y\ ys))|| = 2$, the ATRS $\mathcal{R}$ is non-variadic.

In order to show that non-head-Ω-free proper preredexes have indices, we introduce a non-variadic orthogonal ℓ-ACS version of Lemma 78.

Lemma 100. *Let t be a proper preredex in a non-variadic orthogonal ℓ-ACS $\mathcal{R}$. An Ω-position p of t is an index if and only if $t[\bullet]_p$ is not redex compatible.*

Proof.

- For the 'if' direction, assume that $p \in \mathcal{I}_\mathcal{R}(t)$. We show the claim holds by contradiction. Assuming that $t[\bullet]_p$ is redex compatible with ℓ_Ω for some $\ell \to r \in \mathcal{R}$, we have $\omega(t[\bullet]_p) = \Omega$. Since t is a proper preredex, $\omega(t) = \Omega$. Hence it follows that $\omega(t) = \omega(t[\bullet]_p) = \Omega$. By contraposition of Lemma 65, we can conclude that $p \notin \mathcal{I}_\mathcal{R}(t)$, contradicting the assumption.
- For the 'only if' direction, assume that $t[\bullet]_p$ is not redex compatible. We have $\omega(t[\bullet]_p) = t[\bullet]_p$. On the other hand, t is a proper preredex, and thus $\omega(t) = \Omega$. Hence it follows that $\omega(t) \neq \omega(t[\bullet]_p)$, which is equivalent to $p \in \mathcal{I}_\mathcal{R}(t)$ by Lemma 65.

$\square$

Furthermore, we obtain the following lemma.

Lemma 101. *Let $\mathcal{R}$ be a non-variadic orthogonal ℓ-ACS over $\mathcal{F}$, t an Ω-term and p an Ω-position of t. The following two statements are equivalent.*

1. *For every rule $\ell \to r \in \mathcal{R}$, if $t <_\Omega \ell_\Omega$ then $p \in Pos_{\mathcal{F}}(\ell)$.*
2. *$p \in \mathcal{I}_{\mathcal{R}}(t)$.*

Proof. For the 'only if' direction, assume that for every rule $\ell \to r \in \mathcal{R}$, if $t <_\Omega \ell_\Omega$ then $p \in Pos_{\mathcal{F}}(\ell)$. Let $\ell \to r$ be an arbitrary rule in $\mathcal{R}$. Assume that $t <_\Omega \ell_\Omega$. It suffices to show that $t[\bullet]_p \uparrow_\Omega \ell_\Omega$ does not hold. Assume to the contrary. Since $p \in Pos_{\mathcal{F}}(\ell)$ and $p \in Pos_{\mathcal{F}}(t[\bullet]_p)$, the assumption $t[\bullet]_p \uparrow_\Omega \ell_\Omega$ requires that the function of ℓ at p should be $\bullet$. This contradicts the assumption that $\bullet$ is a fresh symbol.

For the 'if' direction, assume that $p \in \mathcal{I}_{\mathcal{R}}(t)$. We show that the claim holds by contradiction. As $p \in \mathcal{I}_{\mathcal{R}}(t)$, it follows that $t[\bullet]_p$ is not redex compatible in $\mathcal{R}$. Hence $t[\bullet]_p \uparrow_\Omega \ell_\Omega$ for no rule $\ell \to r$. Let $\ell \to r$ be an arbitrary rule in $\mathcal{R}$, and assume that $t <_\Omega \ell_\Omega$. As $t[\bullet]_p \uparrow_\Omega \ell_\Omega$ does not hold, there exists no term u such that $t[\bullet]_p \leq_\Omega u$ and $\ell_\Omega \leq_\Omega u$. Assume $p \in Pos_{\mathcal{V}}(\ell)$ further. Considering that $t \leq_\Omega \ell_\Omega$ and $p \in Pos_{\mathcal{V}}(\ell)$, we can take u such that $u = (\ell_\Omega)[\bullet]_p$, and then we obtain $t[\bullet]_p \leq_\Omega u$ and $\ell_\Omega \leq_\Omega u$. This contradicts the previous argument, and thus $p \in Pos_{\mathcal{F}}(\ell)$. Since $\ell \to r$ is arbitrary, we conclude that for every rule $\ell \to r \in \mathcal{R}$, if $t <_\Omega \ell_\Omega$ then $p \in Pos_{\mathcal{F}}(\ell)$. $\square$

We can give an constructor system version of the above lemma by Lemma 78.

Lemma 102. *Let $\mathcal{R}$ be a constructor system over $\mathcal{F}$, t an Ω-term and p an Ω-position of t. The following two statements are equivalent.*

1. *For every rule $\ell \to r \in \mathcal{R}$, if $t <_\Omega \ell_\Omega$ then $p \in Pos_{\mathcal{F}}(\ell)$.*
2. *$p \in \mathcal{I}_{\mathcal{R}}(t)$.*

Positions of non-head-Ω-free proper preredex are denoted like $p1$. The next lemma shows that such positions always become indices in non-variadic orthogonal ℓ-ACSs.

Lemma 103. *Let $\mathcal{R}$ be a non-variadic orthogonal ℓ-ACS and t a proper preredex of $\mathcal{R}$. If $p1 \in Pos_\Omega(t)$ then $p1 \in \mathcal{I}_{\mathcal{R}}(t)$.*

Proof. Assume that $p1 \in Pos_\Omega(t)$. Let $\ell \to r$ an arbitrary rule in $\mathcal{R}$, and assume that $t <_\Omega \ell_\Omega$. As $t <_\Omega \ell_\Omega$, we have $Pos(t) \subseteq Pos(\ell_\Omega) \subseteq Pos(\ell)$, and thus $p1 \in Pos(\ell)$. Since $\mathcal{R}$ is left-head-variable-free, $p1 \in Pos_\mathcal{F}(t)$. As $\ell \to r$ is arbitrary, by Lemma 101, it follows that $p \in \mathcal{I}_\mathcal{R}(t)$. □

Owing to this property, in non-variadic orthogonal ℓ-ACSs, actually we should consider only head-Ω-free proper preredexes to show their strong sequentiality.

3.2 Definitions

We define left-normal translation for ATRSs.

Definition 104. Let $\mathsf{L} = \{t_\Omega \mid t \text{ is a left-normal term}\}$. We write t^L for $\bigvee\{u \in \mathsf{L} \mid u \leq_\Omega t\}$. An arbitrary but fixed linear term u such that $u_\Omega = t$ is denoted by $t^\mho$.

Lemma 105. *Let u be an Ω-term. The set $\{t \in \mathsf{L} \mid t \leq_\Omega u\}$ is finite and totally ordered with respect to $\leq_\Omega$.*

Definition 106. Let $\mathcal{R}$ be a non-variadic ATRS. We define $\mathsf{M}(\mathcal{R})$ as follows.

$$\mathsf{M}(\mathcal{R}) = \min_{\leq_\Omega}(\{\ell^\mathsf{L} \mid \ell \to r \in \mathcal{R} \text{ and } \mathcal{I}_\mathcal{R}({\ell_\Omega}^\mathsf{L}) \neq \varnothing\})$$

The ATRS $\mathsf{A}(\mathcal{R})$ is defined by

$$\mathsf{A}(\mathcal{R}) = \{t^\mho \to f(t^\mho|_p)x_1 \ \cdots \ x_n \mid t \in \mathsf{M}(\mathcal{R})\}$$

where $p = \min_{<_\mathsf{lo}}(\mathcal{I}_\mathcal{R}(t))$, f is a fresh constant symbol, and $x_1, \ldots, x_n$ are the variables in $Var(t^\mho) \setminus \{t^\mho|_p\}$ in a fixed order.

Example 107. Consider the ATRS $\mathcal{R}$ from Example 99. The rules 1, 2 and 4 are not left-normal: for example, we can see that in the 4th left-hand side $\circ(\circ(\mathsf{take}, \circ(\mathsf{s}, x)), \underline{\circ(\circ(:, y), ys)})$, the underlined subterm is on the right of the variable x. We obtain $\mathsf{M}(\mathcal{R}) = \{\mathsf{map}\ f\ x_1, \mathsf{take}\ (\mathsf{s}\ x)\ x_1\}$ where x_1 is a fresh variable, and $\mathsf{A}(\mathcal{R})$:

$$\mathsf{map}\ f\ x_1 \to \mathsf{map}_1\ x_1\ f$$
$$\mathsf{take}\ (\mathsf{s}\ x)\ x_1 \to \mathsf{take}_1\ x_1\ x$$

Lemma 108. *Let $\mathcal{R}$ be a non-variadic ATRS. For every two Ω-terms t and u in $\mathsf{M}(\mathcal{R})$, $\mathsf{head}(t) \neq \mathsf{head}(u)$.*

Definition 109. Let $\mathcal{R}$ be a non-variadic ATRS. The one-step left-normal translation $\mathsf{AB}(\mathcal{R})$ is defined by $\mathsf{AB}(\mathcal{R}) = \mathsf{A}(\mathcal{R}) \cup \mathsf{B}(\mathcal{R})$, where $\mathsf{B}(\mathcal{R}) = \{\ell{\downarrow_{\mathsf{A}(\mathcal{R})}} \to r \mid \ell \to r \in \mathcal{R}\}$.

Example 110. Consider the ATRS from Example 107. The original TRS $\mathcal{R}$ is rewritten to be $\mathsf{B}(\mathcal{R})$ by $\mathsf{A}(\mathcal{R})$, and then we obtain $\mathsf{AB}(\mathcal{R}) = \mathsf{A}(\mathcal{R}) \cup \mathsf{B}(\mathcal{R})$ with the following rules.

$$
\begin{array}{rr@{\;}l}
1: & \mathsf{map}\ f\ x_1 & \to \mathsf{map_1}\ x_1\ f \\
2: & \mathsf{take}\ (\mathsf{s}\ x)\ x_1 & \to \mathsf{take_1}\ x_1\ x \\
3: & \mathsf{map_1}\ \mathsf{nil}\ f & \to \mathsf{nil} \\
4: & \mathsf{map_1}\ (:\ x\ xs)\ f & \to\ :\ (f\ x)\ (\mathsf{map}\ f\ xs) \\
5: & \mathsf{take}\ \mathsf{0}\ xs & \to \mathsf{nil} \\
6: & \mathsf{take_1}\ (:\ y\ ys)\ x & \to\ :\ y\ (\mathsf{take}\ y\ xs) \\
7: & +\ \mathsf{0} & \to \mathsf{id} \\
8: & +\ (\mathsf{s}\ x)\ y & \to \mathsf{s}\ (+\ x\ y) \\
9: & \mathsf{id}\ x & \to x
\end{array}
$$

When generating $\mathsf{M}(\mathcal{R})$, subterms in non-left-normal left-hand sides that are not variables and on the right of variables are replaced by Ω. For every element t in $\mathsf{M}(\mathcal{R})$, t is a proper preredex of a TRS $\mathcal{R}$ because there exists a rule $\ell \to r$ such that $t <_\Omega \ell_\Omega$. Hence positions of such subterms become indices of elements in $\mathsf{M}(\mathcal{R})$ by Lemma 100.

Next, we discuss finiteness of the translation. Whenever left-normal translation yields the same ATRS as a strongly sequential input ATRS, the input one is already left-normal and the translation terminates at the time. In the remaining part of the subsection, we assume that $\mathcal{R}$ is a *finite* non-variadic orthogonal ℓ-ACS.

Lemma 111. *Let t be an applicative term and σ a substitution.* $||t\sigma|| \geq \sum_{x \in Var(t)} ||x\sigma||$.

Proof. By structural induction on t.

- If $t \in \mathcal{V}$, then we have:

$$
\begin{aligned}
(lhs) &= ||t\sigma|| \\
(rhs) &= \sum_{x \in Var(t)} ||x\sigma|| = ||t\sigma|| = (lhs)
\end{aligned}
$$

- If $t \in t_1 \circ t_2$, then we have:

$$(lhs) = 1 + ||t_1\sigma|| + ||t_2\sigma||$$
$$(rhs) = \sum_{x \in Var(t)} ||x\sigma|| = \sum_{x \in Var(t_1)} ||x\sigma|| + \sum_{x \in Var(t_2)} ||x\sigma||$$

  By the induction hypothesis, $||t_1\sigma|| \geq \sum_{x \in Var(t_1)} ||x\sigma||$ and $||t_2\sigma|| \geq \sum_{x \in Var(t_2)} ||x\sigma||$. Therefore, it follows that:

$$\begin{aligned}(lhs) &= 1 + ||t_1\sigma|| + ||t_2\sigma|| \\ &\geq 1 + \sum_{x \in Var(t_1)} ||x\sigma|| + \sum_{x \in Var(t_2)} ||x\sigma|| \\ &> \sum_{x \in Var(t_1)} ||x\sigma|| + \sum_{x \in Var(t_2)} ||x\sigma|| = (rhs)\end{aligned}$$

□

Lemma 112. *Let t be an applicative term such that $t \notin \mathcal{V}$ and σ a substitution. For every $y \in Var(t)$, we have $||t\sigma|| > ||y\sigma||$.*

Proof. By Lemma 111, $||t\sigma|| > \sum_{x \in Var(t)} ||x\sigma||$. As $y \in Var(t)$, we have $\sum_{x \in Var(t)} ||x\sigma|| \geq ||y\sigma||$, and thus $||t\sigma|| > ||y\sigma||$. □

Lemma 113. $\mathsf{AB}^n(\mathcal{R}) = \mathsf{AB}^{n+1}(\mathcal{R})$ *for some $n \geq 1$.*

Proof.

- If $\mathcal{R}$ is left-normal, then $\mathsf{M}(\mathcal{R}) = \varnothing$. Hence we have $\mathsf{A}(\mathcal{R}) = \varnothing$ and $\mathsf{B}(\mathcal{R}) = \mathcal{R}$, which imply $\mathcal{R} = \mathsf{AB}(\mathcal{R})$.
- If $\mathcal{R}$ is not left-normal, it suffices to show that $||\ell|| > ||\ell{\downarrow_{\mathsf{A}(\mathcal{R})}}||$ for an arbitrary non-left-normal rule $\ell \to r \in \mathcal{R}$. Let $\ell \to r$ be an arbitrary non-left-normal rule in $\mathcal{R}$. As $\ell \to r$ is not left-normal, there exists $t \to t' \in \mathsf{A}(\mathcal{R})$ and σ such that $\ell = t\sigma$. If $\ell{\downarrow_{\mathsf{A}(\mathcal{R})}}$ is not left-normal, there exists $u \to u' \in \mathsf{A}(\mathsf{AB}(\mathcal{R}))$ and τ such that $\ell{\downarrow_{\mathsf{A}(\mathcal{R})}} = u\tau$. By Lemma 111, we have $||\ell|| - ||\ell{\downarrow_{\mathsf{A}(\mathcal{R})}}|| = ||\ell\sigma|| - ||u\tau|| \geq \sum_{x \in Var(t)} ||x\sigma|| - \sum_{x \in Var(u)} ||x\tau||$. It suffices to show $\sum_{x \in Var(t)} ||x\sigma|| > \sum_{x \in Var(u)} ||x\tau||$. By the definition of $\mathsf{A}(\mathcal{R})$, we have $\ell{\downarrow_{\mathsf{A}(\mathcal{R})}} = (f\ t|_p\ x_1\ \ldots\ x_m)\sigma$ where p is an index of $\mathcal{R}$ and $\{x_1, \ldots, x_m\} = Var(t) \setminus \{t|_p\}$. Hence $Var(t) = t|_p, x_1, \ldots, x_m$, and thus it

follows that:

$$\sum_{x \in Var(t)} ||x\sigma|| = ||(t|_p)\sigma|| + ||x_1\sigma|| + \cdots + ||x_m\sigma||$$

By the definition of $\mathsf{M}(\mathcal{R})$, since $u \to u' \in \mathsf{A}(\mathsf{AB}(\mathcal{R}))$, the position p is an index, we have:

$$\begin{aligned} \ell{\downarrow}_{\mathsf{A}(\mathcal{R})} &= f\tau \ (t|_p)\sigma \ x_1\sigma \ \ldots \ x_m\sigma \\ &= f\tau \ (g \ s_1 \ \cdots \ s_n)\tau \ x_1\tau \ \ldots \ x_m\tau = u\tau \end{aligned}$$

Hence $x_1\sigma = x_1\tau, \ldots, x_m\sigma = x_m\tau$ and $(t|_p)\sigma = (g \ s_1 \ \ldots \ s_n)\tau$ hold, and thus we obtain:

$$\sum_{x \in Var(u)} ||x\tau|| = \sum_{1 \le i \le m} \sum_{x \in Var(x_i)} ||x\tau|| + \sum_{1 \le j \le n} \sum_{x \in Var(s_j)} ||x\tau||$$

By Lemma 111, for arbitrary $1 \le i \le m$, $||x_i\sigma|| = ||x_i\tau|| \ge \sum_{x \in Var(x_i)} ||x\tau||$. By Lemma 112, we have $||(t|_p)\sigma|| = ||(g \ s_1 \ \cdots \ s_n)\tau|| > \sum_{1 \le j \le n} \sum_{x \in Var(s_j)} ||x\tau||$. Consequently it follows that $\sum_{x \in Var(t)} ||x\sigma|| - \sum_{x \in Var(u)} ||x\tau|| > 0$.

□

Definition 114. We define $\mathsf{L}(\mathcal{R})$ by $\mathsf{L}(\mathcal{R}) = \mathsf{AB}^n(\mathcal{R})$ such that $\mathsf{AB}^n(\mathcal{R}) = \mathsf{AB}^{n+1}(\mathcal{R})$ for some $n \ge 1$.

In the case of Example 110, we have $\mathsf{L}(\mathcal{R}) = \mathsf{AB}^1(\mathcal{R})$; the ATRS $\mathsf{AB}^1(\mathcal{R})$ is left-normal, and thus $\mathsf{AB}^1(\mathcal{R}) = \mathsf{AB}^2(\mathcal{R})$.

3.3 Correctness

Hereafter, we show correctness of left-normal translation for ATRSs. In order to show this, we need to proof that the translation satisfies the following properties.

1. Being an ℓ-ACS. If an input ATRS is an ℓ-ACS, the translated ATRS is an ℓ-ACS.
2. Orthogonality. If an input ATRS is orthogonal, the translated ATRS is orthogonal.
3. Reachability. If we can reach a term u from a term t by an input ATRS, we can reach there also by the translated ATRS.

4. Strong sequentiality. If an input ATRS is strongly sequential, the translated ATRS is strongly sequential.

In the remaining part of this section, we suppose that an ATRS $\mathcal{R}$ is non-variadic orthogonal ℓ-ACS in order to ensure orthogonality of $\mathsf{A}(\mathcal{R})$, $\mathsf{B}(\mathcal{R})$ and $\mathsf{AB}(\mathcal{R})$. Necessity of being non-variadic, left-head-variable-free and ACSs is shown in the following counterexamples lacking each property.

Example 115. Consider the variadic ATRS $\mathcal{R} = \{1 : \mathsf{F}\ x\ \mathsf{a} \to \mathsf{a}, 2 : \mathsf{F}\ x\ \mathsf{b}\ \mathsf{a} \to \mathsf{b}\}$ We have $\mathsf{A}(\mathcal{R}) = \{3 : \mathsf{F}\ x\ y \to \mathsf{F}_1\ y\ x, 4 : \mathsf{F}\ x\ y\ z \to \mathsf{F}'_1\ y\ x\ z\}$, which is overlapping.

Example 116. Consider the non-head-variable-free ATRS $\mathcal{R} = \{1 : \mathsf{F}\ \mathsf{a} \to \mathsf{a}, 2 : \mathsf{F}\ (x\ \mathsf{b}) \to \mathsf{b}\}$. We obtain $\mathsf{A}(\mathcal{R}) = \{3 : \mathsf{F}\ (x\ y) \to \mathsf{F}_1\ y\ x\}$ and then $\mathsf{B}(\mathcal{R}) = \{1 : \mathsf{F}\ \mathsf{a} \to \mathsf{a}, 5 : \mathsf{F}_1\ \mathsf{b}\ x \to \mathsf{b}\}$. The rules 1 and 3 are overlapping.

Example 117. Consider the non-ACS ATRS $\mathcal{R} = \{1 : \mathsf{F}\ (\mathsf{F}\ x\ \mathsf{a})\ \mathsf{b} \to x, 2 : \mathsf{F}\ \mathsf{a}\ \mathsf{b} \to x\}$. We obtain $\mathsf{A}(\mathcal{R}) = \{3 : \mathsf{F}\ (\mathsf{F}\ x\ y)\ z \to \mathsf{F}_1\ y\ x\ z\}$ and then $\mathsf{B}(\mathcal{R}) = \{2 : \mathsf{F}\ \mathsf{a}\ \mathsf{b} \to x, 4 : \mathsf{F}_1\ \mathsf{a}\ x\ \mathsf{b} \to x\}$. The rules 2 and 3 are overlapping.

3.3.1 Applicative Constructor Systems

In this part, we show that if an input ATRS $\mathcal{R}$ is a non-variadic orthogonal ℓ-ACS then both of $\mathsf{A}(\mathcal{R})$ and $\mathsf{B}(\mathcal{R})$ are ℓ-ACSs, and consequently $\mathsf{AB}(\mathcal{R})$ is so. We prepare one more definition on ACSs.

Definition 118. A substitute σ is called an *applicative constructor substitute* if t is an applicative constructor term for every mapping $x \mapsto t$ in σ.

Firstly we show that $\mathsf{A}(\mathcal{R})$ is an ℓ-ACS.

Lemma 119. *Let t an applicative basic term. $t^{\mathsf{L}^{\mho}}$ is an applicative basic term.*

Proof. By structural induction on t.

- If t is a constant, then $t^{\mathsf{L}^{\mho}} = t$, and thus t is applicatively basic.
- If $t = t_1 \circ t_2$, then t_1 is applicatively basic such that $\mathsf{head}(t_1) \in \mathcal{AD}_{\mathcal{R}}$ and t_2 is an applicative constructor term. If t_1 contains variables, we have $t^{\mathsf{L}^{\mho}} = {t_1}^{\mathsf{L}^{\mho}} \circ x$

where x is a fresh variable. By the induction hypothesis ${t_1}^{\mathsf{L}\mho}$ is applicatively basic, so is $t^{\mathsf{L}\mho}$. Otherwise, $t^{\mathsf{L}\mho} = t_1 \circ {t_2}^{\mathsf{L}\mho}$. Since ${t_1}^{\mathsf{L}\mho}$ is trivially an applicative constructor system, t is applicatively basic.

$\square$

Lemma 120. $\mathsf{A}(\mathcal{R})$ *is an ℓ-ACS.*

Proof. Let $\ell^\bullet \to r^\bullet$ be an arbitrary rule in $\mathsf{A}(\mathcal{R})$. By the definition of $\mathsf{A}(\mathcal{R})$, there exists $t \in \mathsf{M}(\mathcal{R})$ such that $\ell^\bullet = t^\mho$. By Lemma 119, $\ell^\bullet$ is an applicative basic term. $\square$

Secondly, we show that $\mathsf{B}(\mathcal{R})$ is an ℓ-ACS.

Lemma 121. *Let t be an applicative term and σ a substitution. If t is an applicative constructor term and σ is an applicative constructor substitution, then $t\sigma$ is an applicative constructor term.*

Proof. Let t be an arbitrary applicative constructor term and σ an arbitrary applicative constructor substitution. We show $t\sigma$ is an applicative constructor term by induction on t.

- If $t \in \mathcal{V}$, as σ is an applicative constructor substitution, $t\sigma = \sigma(t)$ is an applicative constructor term.
- If $t \in \mathcal{F}^{(0)}$, $t\sigma = t$ is an applicative constructor term.
- If $t = u \circ v$, $t\sigma = u\sigma \circ v\sigma$. By the I.H., $u\sigma$ and $v\sigma$ are applicative constructor terms. Thus $u\sigma \circ v\sigma$ is an applicative constructor terms.

$\square$

Lemma 122. *Let t an applicative term. If t is an applicative constructor term then $t \in \mathsf{NF}(\mathcal{R})$.*

Proof. Let t be an arbitrary applicative constructor term. We show $t \in \mathsf{NF}(\mathcal{R})$ by contradiction. Assume $t \notin \mathsf{NF}(\mathcal{R})$. There exists $\ell \to r \in \mathcal{R}$, p and σ such that $t|_p = \ell\sigma$. As $\mathcal{R}$ is an ℓ-ACS, $\mathsf{head}(\ell) \in \mathcal{AD}_\mathcal{R}$. As t is an applicative constructor term, $\mathsf{head}(t|_p) \notin \mathcal{AD}_\mathcal{R}$. These contradict $t|_p = \ell\sigma$. $\square$

Lemma 123. $\mathsf{B}(\mathcal{R})$ *is an ℓ-ACS.*

Proof. Let $\ell^\circ \to r^\circ$ be an arbitrary rule in $\mathsf{B}(\mathcal{R})$. By the definition of $\mathsf{B}(\mathcal{R})$, there exists $\ell \to r \in \mathcal{R}$ such that $\ell^\circ = \ell{\downarrow}_{\mathsf{A}(\mathcal{R})}$ and $r^\circ = r$. We distinguish two cases by $\ell{\downarrow}_{\mathsf{A}(\mathcal{R})}$.

- If $\ell{\downarrow}_{\mathsf{A}(\mathcal{R})} = \ell$, as $\mathcal{R}$ is an ℓ-ACS, ℓ is an applicative basic term, so is ℓ°.
- If $\ell{\downarrow}_{\mathsf{A}(\mathcal{R})} \neq \ell$, then $\ell \xrightarrow{\varepsilon}_{\mathsf{A}(\mathcal{R})} \ell'$ holds for some ℓ' and there exists $\ell^\bullet \to r^\bullet \in \mathsf{A}(\mathcal{R})$ and σ such that $\ell = \ell^\bullet\sigma$ and $\ell' = r^\bullet\sigma$. By the definition of $\mathsf{A}(\mathcal{R})$, $\mathsf{head}(r^\bullet) \in \mathcal{AC}_{\mathsf{A}(\mathcal{R})}$ and every argument of $r^\bullet$ is a variable. Thus $r^\bullet$ is an applicative constructor term with respect to $\mathsf{A}(\mathcal{R})$. Since $\ell = \ell^\bullet\sigma$ and ℓ is an applicative basic term, σ is an applicative constructor substitution. By Lemma 121, $r^\bullet\sigma$ is an applicative constructor term. By Lemma 120, as $\mathsf{A}(\mathcal{R})$ is an ℓ-ACS, $r^\bullet \in \mathsf{NF}(\mathsf{A}(\mathcal{R}))$ by Lemma 122. Hence we have $\ell' = r^\bullet\sigma = \ell{\downarrow}_{\mathsf{A}(\mathcal{R})} = \ell^\circ$. By the definition of $\mathsf{A}(\mathcal{R})$, $\mathsf{head}(\ell^\circ) = \mathsf{head}(r^\bullet\sigma)$ is a fresh symbol. Therefore, for every $t \in \mathsf{Arg}(\ell^\circ)$, we have $\mathsf{head}(t) \notin \mathcal{AD}_{\mathsf{B}(\mathcal{R})}$, and $\mathsf{head}(\ell^\circ) \in \mathcal{AD}_{\mathsf{B}(\mathcal{R})}$. Hence every argument of ℓ° is an applicative constructor term.

□

We can conclude that $\mathsf{AB}(\mathcal{R}) = \mathsf{A}(\mathcal{R}) \cup \mathsf{B}(\mathcal{R})$ is an ℓ-ACS.

Theorem 124. $\mathsf{AB}(\mathcal{R})$ *is an ℓ-ACS.*

3.3.2 Orthogonality

In this part we show that the translated ATRS $\mathsf{AB}(\mathcal{R})$ obtained from an orthogonal ATRS $\mathcal{R}$ is orthogonal. Orthogonality requires left-linearity and being non-overlapping. Firstly, we show orthogonality of $\mathsf{AB}(\mathcal{R})$, noting the following property on $\mathsf{M}(\mathcal{R})$.

Lemma 125. *Let $t \in \mathsf{M}(\mathcal{R})$. The term $t^\mho$ is left-normal and linear.*

Lemma 126. $\mathsf{A}(\mathcal{R})$ *is left-linear.*

Proof. Let $\ell^\bullet \to r^\bullet$ be an arbitrary rule in $\mathsf{A}(\mathcal{R})$. By the definition of $\mathsf{A}(\mathcal{R})$, there exists $t \in \mathsf{M}(\mathcal{R})$ such that $t^\mho = \ell^\bullet$. Lemma 125 entails that $t^\mho$ is linear. □

Lemma 127. *Every non-variadic ℓ-ACS is an overlay system.*

Proof. Let $\ell_1 \to r_1$ and $\ell_2 \to r_2$ be arbitrary rules in a non-variadic ℓ-ACS $\mathcal{R}$, and assume they are overlapping: there exist ρ_1, ρ_2 and $p \in Pos_{\mathcal{F}}(\ell_2)$ such that $\ell_1\rho_1$ and $\ell_2\rho_2$ are unifiable. We show that $p = \varepsilon$ by contradiction. Assume that $p \neq \varepsilon$. Since $\mathcal{R}$ is an ℓ-ACS, $\mathsf{head}(\ell_2|_p) \in \mathcal{AC}_{\mathcal{R}}$. As $\mathsf{head}(\ell_1) \in \mathcal{AD}_{\mathcal{R}}$, $\ell_1\rho_1$ and $\ell_2\rho_2$ are not unifiable. $\square$

Lemma 128. $\mathsf{A}(\mathcal{R})$ *is non-overlapping.*

Proof. We show that the claim holds by contradiction. Let $\ell_1 \to r_1$ and $\ell_2 \to r_2$ be arbitrary rules in $\mathsf{A}(\mathcal{R})$, and assume they are overlapping: there exist renaming substitutions ρ_1, ρ_2 and $p \in Pos_{\mathcal{F}}(\ell_2)$ such that $\ell_1\rho_1$ and $\ell_2\rho_2|_p$ are unifiable. By Lemma 127, $p = \varepsilon$. By the definition of $\mathsf{A}(\mathcal{R})$, there exist t and u such that $\ell_1 = t^{\mho}$ and $\ell_2 = u^{\mho}$. By Lemma 108, $\mathsf{head}(t) \neq \mathsf{head}(u)$, and thus $\mathsf{head}(\ell_1) \neq \mathsf{head}(\ell_2)$. This contradicts the assumption. $\square$

We obtain orthogonality of $\mathsf{A}(\mathcal{R})$.

Lemma 129. $\mathsf{A}(\mathcal{R})$ *is orthogonal.*

Moreover, we show completeness of $\mathsf{A}(\mathcal{R})$ and its inverse.

Lemma 130. $\mathsf{A}(\mathcal{R})$ *is complete.*

Proof. Let $\mathcal{F} = \mathcal{AD}_{\mathcal{R}}$. Termination of $\mathsf{A}(\mathcal{R})$ is obtained from the fact that $|t|_{\mathcal{F}} > |u|_{\mathcal{F}}$ whenever $t \to_{\mathsf{A}(\mathcal{R})} u$. Confluence of $\mathsf{A}(\mathcal{R})$ is shown by Lemma 132. $\square$

Lemma 131. $\mathsf{A}(\mathcal{R})^{-1}$ *is complete.*

Proof. Let $\mathcal{G} = \mathcal{AD}_{\mathsf{A}(\mathcal{R})^{-1}}$. Termination of $\mathsf{A}(\mathcal{R})^{-1}$ is obtained from the fact that $|t|_{\mathcal{G}} > |u|_{\mathcal{G}}$ whenever $t \to_{\mathsf{A}(\mathcal{R})^{-1}} u$. As orthogonality implies confluence, we show that $\mathsf{A}(\mathcal{R})^{-1}$ is left-linear and non-overlapping. The former is trivial, and then the remaining task is to show the latter. Suppose $\ell_1\sigma = \ell_2|_p\tau$ for some rules $\ell_1 \to r_1, \ell_2 \to r_2 \in \mathsf{A}(\mathcal{R})^{-1}$, substitutions σ and τ, and a position $p \in Pos_{\mathcal{F}}(\ell_2)$. By Lemma 127, we have $p = \varepsilon$. As ℓ_1 and ℓ_2 are applicative basic terms,

$$\mathsf{head}(\ell_1) = \mathsf{head}(\ell_1\sigma) = \mathsf{head}(\ell_2\tau) = \mathsf{head}(\ell_2)$$

and thus $\ell_1 \to r_1$ and $\ell_2 \to r_2$ are identical. Therefore a critical overlap does not exist. $\square$

By the above lemmata, we further have the following lemmata.

Lemma 132. *Let t and u be terms over $\Sigma_{\mathcal{R}}$. $t = u$ if and only if $t{\downarrow_{\mathsf{A}(\mathcal{R})}} = u{\downarrow_{\mathsf{A}(\mathcal{R})}}$.*

Next, we discuss properties of $\mathsf{B}(\mathcal{R})$.

Lemma 133. $\mathsf{B}(\mathcal{R})$ *is left-linear.*

Proof. Let $\ell \to r$ be an arbitrary rule in $\mathcal{R}$. It suffices to show that $\ell{\downarrow_{\mathsf{A}(\mathcal{R})}}$ is linear. As $\mathcal{R}$ is left-linear, ℓ is linear and $\mathit{Var}(r) \subseteq \mathit{Var}(\ell)$. Since $\mathsf{A}(\mathcal{R})$ is linear and non-erasing, $\mathit{Var}(\ell) = \mathit{Var}(\ell{\downarrow_{\mathsf{A}(\mathcal{R})}})$, and thus it follows that $\ell{\downarrow_{\mathsf{A}(\mathcal{R})}}$ is linear. □

Lemma 134. $\mathsf{B}(\mathcal{R})$ *is non-overlapping.*

Proof. We show that the claim holds by contradiction. Let $\alpha : \ell_1 \to r_1$ and $\beta : \ell_2 \to r_2$ be rules in $\mathsf{B}(\mathcal{R})$, and $p \in \mathit{Pos}_{\mathcal{F}}(\ell_2)$. Assume that $\ell_1\rho_1\sigma = \ell_2|_p\rho_2\sigma$ for some substitution σ and remaining substitutions ρ_1 and ρ_2. By Lemma 127, $p = \varepsilon$, and thus $\ell_1\rho_1\sigma = \ell_2|_p\rho_2\sigma$. Since $\alpha, \beta \in \mathsf{B}(\mathcal{R})$, there exist $\ell'_1 \to r_1, \ell'_2 \to r_2 \in \mathcal{R}$ such that $\ell'_1{\downarrow_{\mathsf{A}(\mathcal{R})}} = \ell_1$ and $\ell'_2{\downarrow_{\mathsf{A}(\mathcal{R})}} = \ell_2$. Moreover, as ρ_1 and ρ_2 are renaming substitutions, and σ is an applicative constructor substitution, we have:

$$(\ell'_1\rho_1\sigma){\downarrow_{\mathsf{A}(\mathcal{R})}} = \ell_1\rho_1\sigma = \ell_2\rho_2\sigma = (\ell'_2\rho_2\sigma){\downarrow_{\mathsf{A}(\mathcal{R})}}$$

By Lemma 132, it follows that $\ell'_1\rho_1\sigma = \ell'_2\rho_2\sigma$, which shows that ℓ'_1 and ℓ'_2 are unifiable. This contradicts the assumption that $\mathcal{R}$ is non-overlapping. □

Finally, we show that $\mathsf{AB}(\mathcal{R})$ is orthogonal. Left-linearity of $\mathsf{AB}(\mathcal{R})$ is obtained by Lemmata 126 and 133.

Lemma 135. $\mathsf{AB}(\mathcal{R})$ *is left-linear.*

Lemma 136. $\mathsf{AB}(\mathcal{R})$ *is non-overlapping.*

Proof. Let $\alpha : \ell_1 \to r_1$ and $\beta : \ell_2 \to r_2$ be rules in $\mathsf{AB}(\mathcal{R})$, and $p \in \mathit{Pos}_{\mathcal{F}}(\ell_2)$. Assume that $\ell_1\rho_1\sigma = \ell_2|_p\rho_2\sigma$ for some substitution σ and remaining substitutions ρ_1 and ρ_2. By Lemma 127, we have $p = \varepsilon$, which implies $\ell_1\rho_1\sigma = \ell_2|\rho_2\sigma$. Moreover by Lemmata 128 and 134, $\alpha \in \mathsf{A}(\mathcal{R})$ and $\beta \in \mathsf{B}(\mathcal{R})$. Hence we have

$$\mathsf{head}(\ell_1) = \mathsf{head}(\ell_1\rho_1\sigma) = \mathsf{head}(\ell_2\rho_2\sigma) = \mathsf{head}(\ell_2) \in \mathcal{AD}_{\mathcal{R}}$$

and thus $\ell_2 \in \mathsf{NF}(\mathsf{A}(\mathcal{R}))$. As $\ell_1\rho_1\sigma = \ell_2\rho_2\sigma$, we have $\ell_{1\Omega} \uparrow_\Omega \ell_{2\Omega}$ and thus $\ell_{1\Omega} \uparrow_\Omega {\ell_2}^{\mathsf{L}}$, which implies that $\ell_{1\Omega} \leq_\Omega {\ell_2}^{\mathsf{L}}$ or $\ell_{1\Omega} >_\Omega {\ell_2}^{\mathsf{L}}$. By the definition of $\mathsf{A}(\mathcal{R})$, there exists $t \in \mathsf{M}(\mathcal{R})$ such that $t^{\mho} = \ell_1$. By minimality of $\mathsf{M}(\mathcal{R})$, $t \leq_\Omega {\ell_2}^{\mathsf{L}}$, and thus $t \leq_\Omega {\ell_2}^{\mathsf{L}} \leq_\Omega \ell_{2\Omega}$ follows, contradicting $\ell_2 \in \mathsf{NF}(\mathsf{A}(\mathcal{R}))$. □

Theorem 137. $\mathsf{AB}(\mathcal{R})$ *is orthogonal.*

3.3.3 Reachability

In this part, we show that if we can reach from a term t to u with a non-variadic orthogonal ℓ-ATRS $\mathcal{R}$ then we can reach there with the translated ATRS $\mathsf{AB}(\mathcal{R})$.

Lemma 138. *If $s \to_\mathcal{R} t$ then $s \to^k_{\mathsf{AB}(\mathcal{R})} t$ for some $k \in \{1, 2\}$.*

Proof. Assume that $s \to_\mathcal{R} t$. Let $\ell \to r$ be an arbitrary rule in $\mathcal{R}$. It suffices to show that there exists $k \in \{1, 2\}$ such that $s \to^k_{\mathsf{AB}(\mathcal{R})} t$.

- If there exists no term $u \in \mathsf{M}(\mathcal{R})$ such that $u \leq t$, then we have $\ell{\downarrow}_{\mathsf{A}(\mathcal{R})} = \ell$, and thus $\ell \to_{\mathsf{B}(\mathcal{R})} r$.
- Otherwise, there exists $\ell^\bullet \to r^\bullet \in \mathsf{A}(\mathcal{R})$ and an applicative constructor substitution σ such that $\ell \to_{\mathsf{A}(\mathcal{R})} \ell'$ with $\ell = \ell^\bullet\sigma$ and $\ell' = r^\bullet\sigma$. Since $r^\bullet$ is an applicative constructor term, by Lemma 121, $\ell' = r^\bullet\sigma$ is an applicative constructor term. By Lemma 122, $\ell' = \mathsf{NF}(\mathsf{A}(\mathcal{R}))$, and thus $\ell' = \ell{\downarrow}_{\mathsf{A}(\mathcal{R})}$. By the definition of $\mathsf{B}(\mathcal{R})$, we have $\ell{\downarrow}_{\mathsf{A}(\mathcal{R})} \to_{\mathsf{B}(\mathcal{R})} r$. Consequently, it follows that $\ell^\bullet \to_{\mathsf{A}(\mathcal{R})} \ell{\downarrow}_{\mathsf{A}(\mathcal{R})} \to_{\mathsf{B}(\mathcal{R})} r$.

□

Moreover, completeness of $\mathsf{A}(\mathcal{R})^{-1}$ shown by Lemma 131 yields the following lemma.

Remark 139. We write $t \uparrow_\mathcal{R}$ for $t{\downarrow}_{\mathcal{R}^{-1}}$.

Lemma 140. *Let s and t be terms. If $s \to_{\mathsf{A}(\mathcal{R})} t$ then $s{\uparrow}_{\mathsf{A}(\mathcal{R})} = t{\uparrow}_{\mathsf{A}(\mathcal{R})}$.*

We shall introduce two properties: (1) if $s \to_\mathcal{R} t$ then $s \to_{\mathsf{L}(\mathcal{R})} t{\downarrow}_{\mathsf{A}(\mathcal{R})}$ and (2) if $s \to_{\mathsf{L}(\mathcal{R})} t$ then $s \to_\mathcal{R} t{\uparrow}_{\mathsf{A}(\mathcal{R})}$ in order to show the main theorem of this part.

Lemma 141. *If $s \xrightarrow{\varepsilon}_{\mathsf{B}(\mathcal{R})} t$ then $s{\uparrow}_{\mathsf{A}(\mathcal{R})} \xrightarrow{\varepsilon}_\mathcal{R} t{\uparrow}_{\mathsf{A}(\mathcal{R})}$.*

Proof. Assume that $s \xrightarrow{\varepsilon}_{\mathsf{B}(\mathcal{R})} t$. There exists $\ell^\circ \to r^\circ \in \mathsf{B}(\mathcal{R})$ and σ such that $s = \ell^\circ\sigma$ and $t = r^\circ\sigma$. We show that there exist $\ell \to r \in \mathcal{R}$ and τ such that $s{\uparrow}_{\mathsf{A}(\mathcal{R})} = \ell\tau$ and $t{\uparrow}_{\mathsf{A}(\mathcal{R})} = r\tau$. Firstly, we can say that r° is an applicative constructor term with respect to $\mathsf{A}(\mathcal{R})^{-1}$ by the definition of $\mathsf{A}(\mathcal{R})$, and thus $r^\circ{\uparrow}_{\mathsf{A}(\mathcal{R})} = r$. Secondly, ℓ° can be denoted by

$$\ell^\circ = f\ \ell_1^\circ\ \cdots\ \ell_n^\circ$$

with $f \in \mathcal{AD}_{\mathsf{B}(\mathcal{R})}$ and $\ell_1^\circ, \dots, \ell_n^\circ \in \mathcal{T}(\mathcal{AC}_{\mathsf{B}(\mathcal{R})}, \mathcal{V})$.

- If $f \notin \mathcal{AD}_{\mathsf{A}(\mathcal{R})^{-1}}$, then ℓ° is an applicative constructor term with respect to $\mathsf{A}(\mathcal{R})^{-1}$. By Lemma 122, $\ell^\circ{\uparrow}_{\mathsf{A}(\mathcal{R})} = \ell^\circ$. By the definition of $\mathsf{A}(\mathcal{R})$, it follows that $\ell_1^\circ, \dots, \ell_n^\circ \in \mathcal{T}(\mathcal{AC}_{\mathsf{A}(\mathcal{R})^{-1}}, \mathcal{V})$. Hence we have

  $$s{\uparrow}_{\mathsf{A}(\mathcal{R})} = (\ell^\circ\sigma){\uparrow}_{\mathsf{A}(\mathcal{R})} = f\ \ell_1^\circ\sigma{\uparrow}_{\mathsf{A}(\mathcal{R})}\ \dots\ \ell_n^\circ\sigma{\uparrow}_{\mathsf{A}(\mathcal{R})} = \ell^\circ\sigma{\uparrow}_{\mathsf{A}(\mathcal{R})}$$

  and likewise $t{\uparrow}_{\mathsf{A}(\mathcal{R})} = (r^\circ\sigma){\uparrow}_{\mathsf{A}(\mathcal{R})} = r^\circ\sigma{\uparrow}_{\mathsf{A}(\mathcal{R})}$. Take $\ell \to r \in \mathcal{R}$ such that $\ell = \ell^\circ$ and $r = r^\circ$, and take $\tau = \sigma{\uparrow}_{\mathsf{A}(\mathcal{R})}$.
- If $f \in \mathcal{AD}_{\mathsf{A}(\mathcal{R})^{-1}}$, then ℓ° is an applicative basic term with respect to $\mathsf{A}(\mathcal{R})^{-1}$. There exist $u \to u' \in \mathsf{A}(\mathcal{R})$ and μ such that $\ell^\circ{\uparrow}_{\mathsf{A}(\mathcal{R})} = u\mu$, and also there exists $\ell \to r \in \mathcal{R}$ such that $\ell = u\mu$ and $r = r^\circ$. Hence we have:

  $$s{\uparrow}_{\mathsf{A}(\mathcal{R})} = (\ell^\circ\sigma){\uparrow}_{\mathsf{A}(\mathcal{R})} = (u\mu)\sigma{\uparrow}_{\mathsf{A}(\mathcal{R})} = \ell\sigma{\uparrow}_{\mathsf{A}(\mathcal{R})}$$

  It suffices to take $\tau = \sigma{\uparrow}_{\mathsf{A}(\mathcal{R})}$.

$\square$

Lemma 142. *Let t be an applicative constructor term, u an applicative term and σ a substitution. If t is linear and $t\sigma \to_{\mathcal{R}} u$ then there exists $p \in Pos_{\mathcal{V}}(t)$ such that $u = t(\{t|_p \mapsto u|_p\} \cup \{y \mapsto y\sigma \mid t|_p \neq y \in \mathcal{V}\})$.*

Proof. Assume that t is linear and $t\sigma \to_{\mathcal{R}} u$. We show that the claim holds by structural induction on t.

- If $t \in \mathcal{V}$, take $p = \varepsilon$.
- If $t = f\ t_1\ \cdots\ t_n$ for some $f \in \mathcal{AC}_{\mathcal{R}}$ and $t_1, \dots, t_n \in \mathcal{T}(\mathcal{AC}_{\mathcal{R}}, \mathcal{V})$, then $t\sigma = f\ t_1\sigma\ \dots\ t_n\sigma$. By linearity of t, there exists i and a tern v such that $t_i\sigma \to_{\mathcal{R}} v$. By the induction hypothesis, there exists $q \in Pos_{\mathcal{V}}(t_i)$ such that $v = t_i(\{t_i|_q \mapsto v|_q\} \cup \{y \mapsto y\sigma \mid t_i|_q \neq y \in \mathcal{V}\})$. It suffices to take $p = iq$.

$\square$

Here we refer to [1] for two lemmata in order to proof a proposition below.

Lemma 143 (Parallel moves lemma [1])**.** *Let $\mathcal{R}$ and $\mathcal{S}$ be orthogonal TRSs such that $\mathcal{R}$ and $\mathcal{S}$ are mutually orthogonal and $\mathcal{S}$ is non-erasing and linear. If $t\,{}_{\mathcal{R}}\!\leftarrow s \rightarrow_{\mathcal{S}} u$ then $t \rightarrow_{\mathcal{S}} \cdot^{*}_{\mathcal{R}} \leftarrow u$.*

Lemma 144 (Critical pair lemma [1])**.** *Let $\mathcal{R}$ and $\mathcal{S}$ be left-linear TRSs such that ${}_{\mathcal{R}}\!\leftarrow \rtimes \rightarrow_{\mathcal{S}} \subseteq \rightarrow_{\mathcal{S}}$, and p and q positions such that $p \parallel q$ or $p \geq q$. If $t\,{}_{\mathcal{R}}\!\xleftarrow{p} s \xrightarrow{q}_{\mathcal{S}} u$ then $t \rightarrow_{\mathcal{S}} \cdot^{*}_{\mathcal{R}} \leftarrow u$.*

Lemma 145. *Let s, t, u be applicative terms. If $t\,{}_{\mathsf{A}(\mathcal{R})^{-1}}\!\xleftarrow{i} s \rightarrow_{\mathsf{B}(\mathcal{R})} u$ then $t \rightarrow_{\mathsf{B}(\mathcal{R})\cup\mathcal{R}} \cdot^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \xleftarrow{i} u$.*

Proof. Assume $t\,{}_{\mathsf{A}(\mathcal{R})^{-1}}\!\xleftarrow{i\ p} s \xrightarrow{q}_{\mathsf{B}(\mathcal{R})} u$ for some positions p and q. There exists $\ell^{\circ} \rightarrow r^{\circ} \in \mathsf{B}(\mathcal{R})$ and σ such that $s|_q = \ell^{\circ}\sigma$ and $u = \ell^{\circ}[r^{\circ}\sigma]_p$. We distinguish two cases.

- If $\ell^{\circ} \rightarrow r^{\circ} \in \mathcal{R}$, by the definition of $\mathsf{A}(\mathcal{R})$ we can say that $\mathcal{R}$ and $\mathsf{A}(\mathcal{R})^{-1}$ are mutually orthogonal, besides $\mathsf{A}(\mathcal{R})^{-1}$ is non-erasing and linear. By Lemma 143 (parallel moves lemma), $t \rightarrow_{\mathsf{B}(\mathcal{R})\cup\mathcal{R}} \cdot^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \xleftarrow{i} u$.
- If $\ell^{\circ} \rightarrow r^{\circ} \notin \mathcal{R}$, there exists $\ell^{\bullet} \rightarrow r^{\bullet} \in \mathsf{A}(\mathcal{R})$ and τ such that $\ell^{\circ} = r^{\bullet}\tau$. Assume $p < q$, and let $q = pq'$. As $\mathsf{head}(\ell^{\circ}) \in \mathcal{AD}_{\mathcal{R}}$ and $\ell^{\circ} = r^{\bullet}\tau$, there exists no position q', and thus $p \geq q$ or $p \parallel q$. By Lemma 144 (critical pair lemma) $t \rightarrow_{\mathsf{B}(\mathcal{R})} \cdot^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \xleftarrow{i} u$ follows. Therefore, we obtain $t \rightarrow_{\mathsf{B}(\mathcal{R})\cup\mathcal{R}} \cdot^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \xleftarrow{i} u$.

$\square$

Lemma 146. *Let s, t, u be applicative terms. If $t^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \xleftarrow{i} s \rightarrow_{\mathsf{B}(\mathcal{R})} u$ then $t \rightarrow_{\mathsf{B}(\mathcal{R})\cup\mathcal{R}} \cdot^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \xleftarrow{i} u$.*

Proof. Assume that $t^{n}_{\mathsf{A}(\mathcal{R})^{-1}} \xleftarrow{i} s \rightarrow_{\mathsf{B}(\mathcal{R})} u$ for some n. We show that $t \rightarrow_{\mathsf{B}(\mathcal{R})\cup\mathcal{R}} \cdot^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \xleftarrow{i} u$ by induction on n.

- If $n = 0$, then $t = s$, and thus the claim holds.
- If $n = n' + 1$, we have $t^{n'}_{\mathsf{A}(\mathcal{R})^{-1}} \xleftarrow{i} t'\,{}_{\mathsf{A}(\mathcal{R})^{-1}}\!\xleftarrow{i} s \rightarrow_{\mathsf{B}(\mathcal{R})} u$ for some t'. By Lemma 145, $t' \rightarrow_{\mathsf{B}(\mathcal{R})\cup\mathcal{R}} u'^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \xleftarrow{i} u$ for some u'. Since $t' \notin \mathsf{NF}(\mathsf{A}(\mathcal{R})^{-1})$ and $t \in \mathsf{NF}(\mathsf{A}(\mathcal{R})^{-1})$, we have $t' \rightarrow_{\mathsf{B}(\mathcal{R})} u'$, and thus $t^{n'}_{\mathsf{A}(\mathcal{R})^{-1}} \xleftarrow{i} t' \rightarrow_{\mathsf{B}(\mathcal{R})} u'$. By

the induction hypothesis we have $t \to_{\mathsf{B}(\mathcal{R})\cup\mathcal{R}} \cdot^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \overset{i}{\leftarrow} u'$, and consequently $t \to_{\mathsf{B}(\mathcal{R})\cup\mathcal{R}} \cdot^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \overset{i}{\leftarrow} u$ follows.

□

Lemma 147. *Let s, t be applicative terms. If $s \to_{\mathsf{B}(\mathcal{R})} t$ then $s\!\uparrow_{\mathsf{A}(\mathcal{R})} \to_{\mathcal{R}} t\!\uparrow_{\mathsf{A}(\mathcal{R})}$.*

Proof. Assume that $s \to_{\mathsf{B}(\mathcal{R})} t$. By Lemma 131, as $\mathsf{A}(\mathcal{R})^{-1}$ is complete, it follows that $s \xrightarrow{i}{}^{*}_{\mathsf{A}(\mathcal{R})^{-1}} s\!\uparrow_{\mathsf{A}(\mathcal{R})}$ and $t \xrightarrow{i}{}^{*}_{\mathsf{A}(\mathcal{R})^{-1}} t\!\uparrow_{\mathsf{A}(\mathcal{R})^{-1}}$. Hence we have:

$$s\!\uparrow_{\mathsf{A}(\mathcal{R})}{}^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \overset{i}{\leftarrow} s \to_{\mathsf{B}(\mathcal{R})} t \xrightarrow{i}{}^{*}_{\mathsf{A}(\mathcal{R})^{-1}} t\!\uparrow_{\mathsf{A}(\mathcal{R})^{-1}}$$

By Lemma 146, $s\!\uparrow_{\mathsf{A}(\mathcal{R})} \to_{\mathsf{B}(\mathcal{R})\cup\mathcal{R}} t'{}^{*}_{\mathsf{A}(\mathcal{R})^{-1}} \overset{i}{\leftarrow} t$ for some t'. As $s\!\uparrow_{\mathsf{A}(\mathcal{R})} \in \mathsf{NF}(\mathsf{A}(\mathcal{R})^{-1})$, we have $\mathsf{head}(s\!\uparrow_{\mathsf{A}(\mathcal{R})}) \in \mathcal{AD}_{\mathsf{B}(\mathcal{R})} \setminus \mathcal{AD}_{\mathcal{R}}$, and thus $s\!\uparrow_{\mathsf{A}(\mathcal{R})} \to t'$. By the definition of $\mathsf{A}(\mathcal{R})$, it follows that $t' \in \mathsf{NF}(\mathsf{A}(\mathcal{R})^{-1})$. Consequently, we have $s\!\uparrow_{\mathsf{A}(\mathcal{R})} \to_{\mathcal{R}} t\!\uparrow_{\mathsf{A}(\mathcal{R})}$. □

Lemma 148. *Let s, t be applicative terms. If $s \to_{\mathsf{AB}(\mathcal{R})} t$ then $s\!\uparrow_{\mathsf{A}(\mathcal{R})} \to^{=}_{\mathcal{R}} t\!\uparrow_{\mathsf{A}(\mathcal{R})}$.*

Proof. Assume that $s \to_{\mathsf{AB}(\mathcal{R})} t$. From $\mathsf{AB}(\mathcal{R}) = \mathsf{A}(\mathcal{R}) \cup \mathsf{B}(\mathcal{R})$, we distinguish two cases.

- If $s \to_{\mathsf{A}(\mathcal{R})} t$, by Lemma 140, we have $s\!\uparrow_{\mathsf{A}(\mathcal{R})} = t\!\uparrow_{\mathsf{A}(\mathcal{R})}$, and thus $s\!\uparrow_{\mathsf{A}(\mathcal{R})} \to^{0}_{\mathcal{R}} t\!\uparrow_{\mathsf{A}(\mathcal{R})}$.
- If $s \to_{\mathsf{B}(\mathcal{R})} t$, by Lemma 147, we have $s\!\uparrow_{\mathsf{A}(\mathcal{R})} \to_{\mathcal{R}} t\!\uparrow_{\mathsf{A}(\mathcal{R})}$.

□

Lemma 149. *Let s, t, u be applicative terms. If $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V}) \cap \mathsf{NF}(\mathcal{R})$ holds and $s \to^{*}_{\mathsf{A}(\mathcal{R})} t \to_{\mathsf{AB}(\mathcal{R})} u$ then $t \to_{\mathsf{A}(\mathcal{R})} u$.*

Proof. Assume that $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V}) \cap \mathsf{NF}(\mathcal{R})$ holds and $s \to^{*}_{\mathsf{A}(\mathcal{R})} t \to_{\mathsf{AB}(\mathcal{R})} u$. Assume $s \to^{*}_{\mathsf{A}(\mathcal{R})} t \to_{\mathsf{B}(\mathcal{R})} u$. By Lemma 147, $t\!\uparrow_{\mathsf{A}(\mathcal{R})} \to_{\mathcal{R}} u\!\uparrow_{\mathsf{A}(\mathcal{R})}$, and thus $s \to^{*}_{\mathsf{A}(\mathcal{R})} t \to^{*}_{\mathsf{A}(\mathcal{R})^{-1}} t\!\uparrow_{\mathsf{A}(\mathcal{R})} \to_{\mathcal{R}} u\!\uparrow_{\mathsf{A}(\mathcal{R})}$. As $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V})$ and $\mathsf{A}(\mathcal{R})^{-1}$ is complete, $s = t\!\uparrow_{\mathsf{A}(\mathcal{R})}$ follows, and thus $s = t\!\uparrow_{\mathsf{A}(\mathcal{R})} \to_{\mathcal{R}} u\!\uparrow_{\mathsf{A}(\mathcal{R})}$. This contradicts the assumption $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V}) \cap \mathsf{NF}(\mathcal{R})$. □

Lemma 150. *Let s, t be applicative terms. If $s \to^{*}_{\mathsf{AB}(\mathcal{R})} t$ and $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V}) \cap \mathsf{NF}(\mathcal{R})$ then $s \to^{*}_{\mathsf{A}(\mathcal{R})} t$.*

Proof. Assume that $s \to^n_{\mathsf{AB}(\mathcal{R})} t$ and $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V}) \cap \mathsf{NF}(\mathcal{R})$ for some n. We show that $s \to^*_{\mathsf{A}(\mathcal{R})} t$ by induction on n.

- If $n = 0$, then $s \to^0_{\mathsf{AB}(\mathcal{R})} t$, and thus $s \to^0_{\mathsf{A}(\mathcal{R})} t$.
- If $n = n' + 1$, then $s \to^{n'}_{\mathsf{AB}(\mathcal{R})} u \to_{\mathsf{AB}(\mathcal{R})} t$ for some u. By the induction hypothesis, $s \to^*_{\mathsf{A}(\mathcal{R})} u$, which yields $s \to^*_{\mathsf{A}(\mathcal{R})} u \to_{\mathsf{AB}(\mathcal{R})} t$. By Lemma 149, we have $u \to_{\mathsf{A}(\mathcal{R})} t$, and thus $s \to^*_{\mathsf{A}(\mathcal{R})} t$.

$\square$

Lemma 151. *Let s be an applicative term. If $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V})$ and $s \to^!_{\mathcal{R}} t$ then $s \to^!_{\mathsf{AB}(\mathcal{R})} t{\downarrow}_{\mathsf{A}(\mathcal{R})}$.*

Proof. Assume that $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V})$ and $s \to^!_{\mathcal{R}} t$. It suffices to show that $t{\downarrow}_{\mathsf{A}(\mathcal{R})} \in \mathsf{NF}(\mathsf{AB}(\mathcal{R}))$. By the assumption, we have a rewrite sequence

$$s \to_{\mathcal{R}} s_1 \to_{\mathcal{R}} \cdots \to_{\mathcal{R}} s_n = t \in \mathsf{NF}(\mathcal{R})$$

for some $n > 0$. By Lemma 138, we have:

$$s \to^*_{\mathsf{AB}(\mathcal{R})} s_1 \to^*_{\mathsf{AB}(\mathcal{R})} \cdots \to^*_{\mathsf{AB}(\mathcal{R})} s_n = t$$

Since $s \to^*_{\mathcal{R}} t$ and $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V})$, it follows that $t \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V})$ and thus $t \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V}) \cap \mathsf{NF}(\mathcal{R})$. By Lemma 150, $t \to^*_{\mathsf{AB}(\mathcal{R})} t{\downarrow}_{\mathsf{AB}(\mathcal{R})}$ implies $t \to^*_{\mathsf{A}(\mathcal{R})} t{\downarrow}_{\mathsf{AB}(\mathcal{R})}$. As $\mathsf{A}(\mathcal{R}) \subset \mathsf{AB}(\mathcal{R})$, we have $t{\downarrow}_{\mathsf{AB}(\mathcal{R})} = t{\downarrow}_{\mathsf{A}(\mathcal{R})}$. Therefore, we conclude that $t{\downarrow}_{\mathsf{A}(\mathcal{R})} \in \mathsf{NF}(\mathsf{AB}(\mathcal{R}))$. $\square$

Lemma 152. *Let s be an applicative term. If $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V})$ and $s \to^!_{\mathsf{AB}(\mathcal{R})} t$ then $s \to^!_{\mathcal{R}} t{\uparrow}_{\mathsf{A}(\mathcal{R})}$.*

Proof. Assume that $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V})$ and $s \to^!_{\mathsf{AB}(\mathcal{R})} t$. It suffices to show that $t{\uparrow}_{\mathsf{A}(\mathcal{R})} \in \mathsf{NF}(\mathcal{R})$. We show this by contradiction. By the assumption, we have a rewrite sequence

$$s \to_{\mathsf{AB}(\mathcal{R})} s_1 \to_{\mathsf{AB}(\mathcal{R})} \cdots \to_{\mathsf{AB}(\mathcal{R})} s_n = t \in \mathsf{NF}(\mathsf{AB}(\mathcal{R}))$$

for some $n > 0$. By Lemma 148, we have:

$$s{\uparrow}_{\mathsf{A}(\mathcal{R})} \to^=_{\mathcal{R}} s_1{\uparrow}_{\mathsf{A}(\mathcal{R})} \to^=_{\mathcal{R}} \cdots \to^=_{\mathcal{R}} s_n{\uparrow}_{\mathsf{A}(\mathcal{R})} = t{\uparrow}_{\mathsf{A}(\mathcal{R})}$$

Assume that $t{\uparrow}_{\mathsf{A}(\mathcal{R})} \notin \mathsf{NF}(\mathcal{R})$, then there exists u such that $t{\uparrow}_{\mathsf{A}(\mathcal{R})} \to_{\mathcal{R}} u$. This entails $(t{\uparrow}_{\mathsf{A}(\mathcal{R})}){\downarrow}_{\mathsf{A}(\mathcal{R})} \to_{\mathsf{AB}(\mathcal{R})} u{\downarrow}_{\mathsf{A}(\mathcal{R})}$. Since $t \in \mathsf{NF}(\mathsf{AB}(\mathcal{R}))$ and $\mathsf{A}(\mathcal{R}) \subset \mathsf{AB}(\mathcal{R})$, we have $(t{\uparrow}_{\mathsf{A}(\mathcal{R})}){\downarrow}_{\mathsf{A}(\mathcal{R})} = t{\downarrow}_{\mathsf{A}(\mathcal{R})} = t$, and thus $t \to_{\mathsf{AB}(\mathcal{R})} u{\downarrow}_{\mathsf{A}(\mathcal{R})}$ follows. This contradicts $t \in \mathsf{NF}(\mathsf{AB}(\mathcal{R}))$ in the first assumption. □

Finally we obtain the following theorem by Lemmata 151 and 152.

Theorem 153. *Let s, t be applicative terms such that $s \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V})$ and $t \in \mathcal{T}(\mathcal{A}\Sigma_{\mathcal{R}}, \mathcal{V}) \cap \mathsf{NF}(\mathcal{R})$ hold. $s \to^*_{\mathcal{R}} t$ if and only if $s \to^*_{\mathsf{AB}(\mathcal{R})} t$.*

3.3.4 Strong sequentiality

In the end of this chapter, we show that strong sequentiality of a non-variadic ℓ-ACS $\mathcal{R}$ is preserved in the translated ATRS $\mathsf{AB}(\mathcal{R})$.

Lemma 154. *Let t be a preredex of $\mathsf{AB}(\mathcal{R})$. If t is redex compatible in $\mathsf{A}(\mathcal{R})$ then t is redex compatible in $\mathcal{R}$.*

Proof. It suffices to show that $t \uparrow_\Omega \ell$ for some $\ell \to r \in \mathcal{R}$. Assume that t is redex compatible in $\mathsf{A}(\mathcal{R})$. There exists $\ell^\bullet \to r^\bullet \in \mathsf{A}(\mathcal{R})$ such that $t \uparrow_\Omega \ell^\bullet$ and $t \leq_\Omega {\ell^\bullet}_\Omega$. By the definition of $\mathsf{M}(\mathcal{R})$, there exists $\ell \to r \in \mathcal{R}$ such that ${\ell^\bullet}_\Omega \leq_\Omega \ell_\Omega$. Hence $t \leq_\Omega \ell_\Omega$, and thus $t \uparrow_\Omega \ell$. □

By this lemma and lemmata in the previous section, we can conclude the strong sequentiality.

Theorem 155. *If $\mathcal{R}$ is strongly sequential then $\mathsf{AB}(\mathcal{R})$ is strongly sequential.*

Proof. It suffices to show that every proper preredex of $\mathsf{AB}(\mathcal{R})$ has index. Let t an arbitrary proper preredex of $\mathsf{AB}(\mathcal{R})$. We distinguish two cases by whether t is a proper preredex of $\mathsf{A}(\mathcal{R})$ or $\mathsf{B}(\mathcal{R})$.

1. If t is a proper preredex of $\mathsf{A}(\mathcal{R})$, there exists $\ell^\bullet \to r^\bullet \in \mathsf{A}(\mathcal{R})$ such that $t <_\Omega {\ell^\bullet}_\Omega$. By the definition of $\mathsf{M}(\mathcal{R})$, there exists $\ell \to r \in \mathcal{R}$ such that ${\ell^\bullet}_\Omega <_\Omega \ell_\Omega$. Hence $t <_\Omega {\ell^\bullet}_\Omega <_\Omega \ell_\Omega$. As $t \neq \Omega$, t is also a proper preredex of $\mathcal{R}$. As $\mathcal{R}$ is strongly sequential, there exists $p \in \mathit{Pos}_\Omega(t)$ such that $p \in \mathcal{I}_{\mathcal{R}}(t)$. By Lemma 101, $t[\bullet]_p$ is not redex compatible in $\mathcal{R}$. By contraposition of Lemma 154, $t[\bullet]_p$

is not redex compatible in $\mathsf{A}(\mathcal{R})$. By Lemma 101, we have $p \in \mathcal{I}_{\mathsf{A}(\mathcal{R})}(t)$, and consequently $p \in \mathcal{I}_{\mathsf{AB}(\mathcal{R})}(t)$.

2. If t is a proper preredex of $\mathsf{B}(\mathcal{R})$, there exists $\ell^\circ \to r^\circ \in \mathsf{B}(\mathcal{R})$ such that $t <_\Omega \ell^\circ{}_\Omega$, and there exists $\ell \to r \in \mathcal{R}$ such that $\ell^\circ = \ell{\downarrow_{\mathsf{A}(\mathcal{R})}}$ and $r^\circ = r$.
   - If $\ell{\downarrow_{\mathsf{A}(\mathcal{R})}} = \ell$, we have $t <_\Omega \ell^\circ{}_\Omega = \ell_\Omega$. Hence t is a proper preredex of $\mathcal{R}$. As $\mathcal{R}$ is strongly sequential, there exists $p \in Pos_\Omega(t)$ such that $p \in \mathcal{I}_\mathcal{R}(t)$. By Lemma 101, $t[\bullet]_p$ is not redex compatible in $\mathcal{R}$.
     (a) If $\mathsf{head}(t) = \Omega$, as $\mathsf{AB}(\mathcal{R})$ is an ACS, the Ω-position of $\mathsf{head}(t)$ is an index by Lemma 103.
     (b) If $\mathsf{head}(t) \neq \Omega$, as $\mathsf{head}(t) \notin \mathcal{AD}_{\mathsf{A}(\mathcal{R})^{-1}}$, it follows that $t[\bullet]_p$ is not redex compatible in $\mathsf{B}(\mathcal{R}) \setminus \mathcal{R}$. Moreover, as $t[\bullet]_p$ is not redex compatible in $\mathcal{R}$, by contraposition of Lemma 154, $t[\bullet]_p$ is not redex compatible in $\mathsf{A}(\mathcal{R})$. Hence $t[\bullet]_p$ is not redex compatible in $\mathsf{AB}(\mathcal{R})$. By Lemma 101, $p \in \mathcal{I}_{\mathsf{AB}(\mathcal{R})}(t)$.
   - If $\ell{\downarrow_{\mathsf{A}(\mathcal{R})}} \neq \ell$, then $\ell \xrightarrow{\varepsilon}_{\mathsf{A}(\mathcal{R})} \ell^\circ$. There exist $\ell^\bullet \to r^\bullet \in \mathsf{A}(\mathcal{R})$ and σ such that $\ell = \ell^\bullet\sigma$ and $\ell^\circ = r^\bullet\sigma$. By the definition of $\mathsf{A}(\mathcal{R})$, there exists $p \in Pos_\Omega(\ell^\bullet{}_\Omega)$ such that $p \in \mathcal{I}_\mathcal{R}(\ell^\bullet{}_\Omega)$. Moreover, we have:

$$r^\bullet = f\ x_1\ x_2\ \ldots\ x_n$$

     where $x_1 = \ell^\bullet|_p$, $p \in \min_{<_{\mathsf{lo}}}(\mathcal{I}_\mathcal{R}(t))$, f is a fresh variable and $\{x_2, \ldots, x_n\} = Var(\ell^\bullet) \setminus \{\ell^\bullet|_p\}$.
     (a) If $\mathsf{head}(t) = \Omega$, by Lemma 103, the Ω-position of $\mathsf{head}(t)$ is an index.
     (b) If $\mathsf{head}(t) \neq \Omega$, as $t \leq_\Omega \ell^\circ{}_\Omega = (r^\bullet\sigma)_\Omega$, the Ω-term t can be denoted by $t = f\ t_1\ t_2\ \ldots\ t_n$. Taking $\tau = \{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$, we have $t = r^\bullet\tau$. Since $t = r^\bullet\tau$ is a proper preredex of $\mathsf{B}(\mathcal{R})$, $\ell^\bullet\tau$ is a proper preredex of $\mathcal{R}$. As $\mathcal{R}$ is strongly sequential, there exists $q \in Pos(\ell^\bullet\tau)$ such that $q \in \mathcal{I}_\mathcal{R}(\ell^\bullet\tau)$. Letting p_i be a position in $\ell^\bullet$ corresponding to x_i in $r^\bullet$, we have $q = p_jq'$ for some j and q'. To show the claim holds, it suffices to show $jq' \in \mathcal{I}_{\mathsf{AB}(\mathcal{R})}(t)$. Assume $jq' \notin \mathcal{I}_{\mathsf{AB}(\mathcal{R})}(t)$. By Lemma 78, $t[\bullet]_{jq'}$ is redex compatible in $\mathsf{AB}(\mathcal{R})$, and by the definition of $\mathsf{A}(\mathcal{R})$, it follows that $t[\bullet]_{jq'}$ is redex compatible in $\mathsf{B}(\mathcal{R}) \setminus \mathcal{R}$. Hence there exists $\ell' \to r' \in \mathsf{B}(\mathcal{R}) \setminus \mathcal{R}$ with which $t[\bullet]_{jq'}$ is compatible, and ℓ' can be written as $\ell' = f\ u_1\ \ldots\ u_n$. By taking $\mu = \{x_1 \mapsto u_1, \ldots, x_n \mapsto$

$u_n\}$, we have $r^\bullet\mu \to r' \in \mathcal{R}$. Since $t[\bullet]_{jq'} = (f\ x_1\ \ldots\ x_n)[\bullet]_{jq'}$ and $\ell' = f\ u_1\ \ldots\ u_n$ are compatible, for every i with $1 \leq i \leq n$, t_i and u_i have maximum on $\leq_\Omega$. Let v_i be such maximum. Taking $\nu = \{x_1 \mapsto v_1, \ldots, x_n \mapsto v_n\}$, $\ell^\bullet\nu$ is the maximum of $\ell^\bullet\tau[\bullet]_q$ and $\ell^\bullet\mu$. Hence $\ell^\bullet\tau[\bullet]_q$ is redex compatible in $\mathcal{R}$. By Lemma 78, we have $q \notin \mathcal{I}_\mathcal{R}(\ell^\bullet\tau)$. This contradicts $q \in \mathcal{I}_\mathcal{R}(\ell^\bullet\tau)$.

$\square$

Chapter 4

Currying and Left-normal translation

Up to here, we have seen that every finite non-variadic orthogonal strongly sequential ℓ-ACS can be translated into an left-normal one by left-normal translation for ATRSs. In this chapter, we show that the translation for ATRSs is available also for functional TRSs after a procedure called currying.

4.1 Currying

First of all, we introduce currying and related definitions, referring to [8]. Currying translates functional TRSs into ATRSs with specialised TRS called a currying system.

Definition 156. Let $\mathcal{F}$ be a signature with $\circ \notin \mathcal{F}$. The *currying system* $\mathcal{C}(\mathcal{F})$ consists of the rewrite rules $f_i(x_1, \ldots, x_{i-1}, x_i) \to f_{i-1}(x_1, \ldots, x_{i-1}) \circ x_i$ for every n-ary function symbol $f \in \mathcal{F}$ and $0 < i \leq n$. Here $f_n = f$ and f_i is a fresh i-ary function symbol.

In a currying procedure, both sides in every rule of a functional TRS are rewritten by a currying system.

Definition 157. Let $\mathcal{R}$ be a TRS over the signature $\mathcal{F}$. The curried system $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is the ATRS consisting of the rules $\ell{\downarrow_{\mathcal{C}(\mathcal{F})}} \to r{\downarrow_{\mathcal{C}(\mathcal{F})}}$ for every $\ell \to r \in \mathcal{R}$.

Example 158. Consider the TRS $\mathcal{R}$:

$$
\begin{array}{rrrr}
1: & \mathsf{take}(:(x, xs), \mathsf{0}) \to \mathsf{nil} & 3: & \mathsf{from}(x) \to :(x, \mathsf{from}(\mathsf{s}(x))) \\
2: & \mathsf{take}(:(x, xs), \mathsf{s}(y)) \to :(x, \mathsf{take}(xs, y)) & 4: & +(\mathsf{0}, x) \to x
\end{array}
$$

We have $\mathcal{F} = \{\mathsf{take}, \mathsf{from}, +, :, 0, \mathsf{s}, \mathsf{nil}\}$. The currying system $\mathcal{C}(\mathcal{F})$ is obtained from the signature $\mathcal{F}$ of $\mathcal{R}$.

$$\begin{array}{llll} 1: & \mathsf{take}(x_1, x_2) \to \mathsf{take}_1(x_1) \circ x_2 & 5: & +(x_1, x_2) \to +_1(x_1) \circ x_2 \\ 2: & \mathsf{take}_1(x_1) \to \mathsf{take}_0 \circ x_1 & 6: & +_1(x_1) \to +_0 \circ x_1 \\ 3: & :(x_1, x_2) \to :_1(x_1) \circ x_2 & 7: & \mathsf{from}(x_1) \to \mathsf{from}_0 \circ x_1 \\ 4: & :_1(x_1) \to :_0 \circ x_1 & 8: & \mathsf{s}(x_1) \to \mathsf{s}_0 \circ x_1 \end{array}$$

The curried ATRS $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ obtained from $\mathcal{R}$ and by $\mathcal{C}(\mathcal{F})$ is:

$$\begin{array}{ll} 1: & \mathsf{take}_0 \circ (:_0 \circ x \circ xs) \circ \mathsf{0} \to \mathsf{nil} \\ 2: & \mathsf{take}_0 \circ (:_0 \circ x \circ xs) \circ (\mathsf{s}_0 \circ y) \to :_0 \circ x \circ (\mathsf{take}_0 \circ xs \circ y) \\ 3: & \mathsf{from}_0 \circ x \to :_0 \circ x \circ (\mathsf{from}_0 \circ (\mathsf{s}_0 \circ x)) \\ 4: & +_0 \circ \mathsf{0} \circ x \to x \end{array}$$

Note that every functional TRS can be translated into an ATRS, and currying systems are terminating and confluent [8].

4.2 Left-normal translation

Hereafter, we show that TRSs that can be translated by left-normal translation for functional TRSs can be translated by left-normal translation for ATRSs after currying unless the translation is not necessary. Hashida [4] introduced left-normal translation for functional TRSs. The translation requires functional TRSs to be finite, strongly sequential and constructor systems.

Example 159. Consider the TRS from Example 158. The rule 1 and 2 is not left-normal, and then the TRS $\mathsf{A}(\mathcal{R})$ is generated as follows.

$$\mathsf{A}(\mathcal{R}) = \{\mathsf{take}(:(x, xs), x_1) \to \mathsf{F}(x_1, x, xs)\}$$

The TRS $\mathcal{R}$ is rewritten into $\mathsf{B}(\mathcal{R})$ by $\mathsf{A}(\mathcal{R})$, and finally the translated TRS $\mathsf{L}(\mathcal{R})$ is obtained by $\mathsf{A}(\mathcal{R}) \cup \mathsf{B}(\mathcal{R})$.

$$\begin{array}{llll} 1: & \mathsf{take}(:(x, xs), x_1) \to \mathsf{F}(x_1, x, xs) & & \\ 2: & \mathsf{F}(\mathsf{0}, x, xs) \to \mathsf{nil} & 3: & \mathsf{from}(x) \to :(x, \mathsf{from}(\mathsf{s}(x))) \\ 4: & \mathsf{F}(\mathsf{s}(y), x, xs) \to :(x, \mathsf{take}(xs, y)) & 5: & +(\mathsf{0}, x) \to x \end{array}$$

On the other hand, left-normal translation for ATRSs requires ATRSs to be non-variadic, orthogonal, left-head-variable-free, ACSs and strongly sequential. Hence curried ATRSs need to satisfy these properties to be left-normal through the translation. Moreover, we need to ensure that if a functional TRS is already left-normal, the curried ATRS should be left-normal, and thus it cannot be translated any more. It is trivial that curried ATRSs are non-variadic; our task is to proof the following properties.

1. Orthogonality of $\mathcal{R}$ is preserved in $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$.
2. $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is left-head-variable-free for any functional TRS $\mathcal{R}$.
3. If $\mathcal{R}$ is a constructor system, then $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is an applicative constructor system.
4. $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ succeeds to left-normality of $\mathcal{R}$.
5. If $\mathcal{R}$ is strongly sequential, so is $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$.

4.2.1 Orthogonality

Firstly, we show that if a TRS $\mathcal{R}$ is left-linear then $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is left-linear.

Lemma 160. *Let $\mathcal{R}$ be a left-linear TRS over $\mathcal{F}$ and $\ell \to r$ a rule in $\mathcal{R}$. If there exists ℓ' such that $\ell \to_{\mathcal{C}(\mathcal{F})} \ell'$ then ℓ' is linear.*

Proof. Let $\ell \to r$ be an arbitrary rule in $\mathcal{R}$. As $\mathcal{R}$ is left-linear, ℓ is linear. Assume that there exists ℓ' such that $\ell \to_{\mathcal{C}(\mathcal{F})} \ell'$. Since $\mathcal{C}(\mathcal{F})$ is right-linear, we conclude that ℓ' is linear by Lemma 160. □

Lemma 161. *Let $\mathcal{R}$ be a left-linear TRS over $\mathcal{F}$ and $\ell \to r$ a rule in $\mathcal{R}$. If $\ell \to^*_{\mathcal{C}(\mathcal{F})} \ell'$ for some ℓ', then ℓ' is linear.*

Proof. By induction on n of $\ell \to^n_{\mathcal{C}(\mathcal{F})} \ell'$. □

By Lemma 161, we have the following lemma on left-linearity.

Lemma 162. *Let $\mathcal{R}$ be a TRS. If $\mathcal{R}$ is left-linear then $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is left-linear.*

Secondly, we show that if a TRS $\mathcal{R}$ is non-overlapping then $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is non-overlapping. Let us begin by showing general properties on currying and substitutions.

Definition 163. Let σ an arbitrary substitution. We denote $\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}$ for the set $\{(\sigma(x)){\downarrow_{\mathcal{C}(\mathcal{F})}} \mid x \in \mathcal{V}\}$.

Lemma 164. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$, and let t a term and σ a substitution. $(t{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) = (t\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}}$.*

Proof. We show that the claim holds by structural induction on t.

- If t is a constant symbol, we have:

$$\begin{aligned}(lhs) &= (t{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) = t(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) = t \\ (rhs) &= (t\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}} = t{\downarrow_{\mathcal{C}(\mathcal{F})}} = t \\ &= (lhs)\end{aligned}$$

- If $t = x \in \mathcal{V}$, we have:

$$\begin{aligned}(lhs) &= (x{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) = x(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) = \sigma(x){\downarrow_{\mathcal{C}(\mathcal{F})}} \\ (rhs) &= (x\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}} = (\sigma(x)){\downarrow_{\mathcal{C}(\mathcal{F})}} = (lhs)\end{aligned}$$

- If $t = f(t_1, \ldots, t_n)$, we have:

$$\begin{aligned}(lhs) &= (f(t_1, \ldots, t_n){\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) \\ &= (f_0 \circ t_1{\downarrow_{\mathcal{C}(\mathcal{F})}} \circ \cdots \circ t_n{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) \\ &= f_0(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) \circ (t_1{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) \circ \cdots \circ (t_n{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) \\ &= f_0 \circ (t_1\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}} \circ \cdots \circ (t_n\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}} \\ (rhs) &= (f(t_1, \ldots, t_n)\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}} \\ &= (f(t_1\sigma, \ldots, t_n\sigma)){\downarrow_{\mathcal{C}(\mathcal{F})}} \\ &= f_0 \circ (t_1\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}} \circ \cdots \circ (t_n\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}} \\ &= (lhs)\end{aligned}$$

□

Lemma 165. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and s, t terms. If $s{\downarrow_{\mathcal{C}(\mathcal{F})}} = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$ then $s = t$.*

Proof. Assume $s{\downarrow_{\mathcal{C}(\mathcal{F})}} = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$. We show that $s = t$ by structural induction on s.

- If $s \in \mathcal{V}$, then $s = s{\downarrow_{\mathcal{C}(\mathcal{F})}}$. As $s{\downarrow_{\mathcal{C}(\mathcal{F})}} = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$, we have $s = t{\downarrow_{\mathcal{C}(\mathcal{F})}} \in \mathcal{V}$, and thus $t = t{\downarrow_{\mathcal{C}(\mathcal{F})}} \in \mathcal{V}$.
- If s is a constant symbol, then $s = s{\downarrow_{\mathcal{C}(\mathcal{F})}}$. As $s{\downarrow_{\mathcal{C}(\mathcal{F})}} = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$, it follows that $t{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is a constant symbol, and thus $t = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$.

- If $s = f(s_1, \ldots, s_n)$, then $s{\downarrow_{\mathcal{C}(\mathcal{F})}} = f_0 \circ s_1{\downarrow_{\mathcal{C}(\mathcal{F})}} \circ \cdots \circ s_n{\downarrow_{\mathcal{C}(\mathcal{F})}}$. As $s{\downarrow_{\mathcal{C}(\mathcal{F})}} = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$, we have $t{\downarrow_{\mathcal{C}(\mathcal{F})}} = f_0 \circ t_1{\downarrow_{\mathcal{C}(\mathcal{F})}} \circ \cdots \circ t_n{\downarrow_{\mathcal{C}(\mathcal{F})}}$. As $s{\downarrow_{\mathcal{C}(\mathcal{F})}} = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$, we have $s_1{\downarrow_{\mathcal{C}(\mathcal{F})}} = t_1{\downarrow_{\mathcal{C}(\mathcal{F})}}, \ldots, s_n{\downarrow_{\mathcal{C}(\mathcal{F})}} = t_n{\downarrow_{\mathcal{C}(\mathcal{F})}}$. By the induction hypotheses, we have $s_1 = t_1, \ldots, s_n = t_n$, and consequently $s = t$ follows.

□

Lemma 166. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$, and let s, t terms and σ a substitution. $s\sigma = t\sigma$ if and only if $(s{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) = (t{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}})$.*

Proof.

- For the 'if' direction, assume that $(s{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) = (t{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}})$. By Lemma 164, $(s\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}} = (t\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}}$, and then by Lemma 165 we obtain $s\sigma = t\sigma$.
- For the 'only if' direction, assume that $s\sigma = t\sigma$. By Lemma 165, we have $(s\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}} = (t\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}}$. By Lemma 164, it follows that:

$$(s{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}}) = (s\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}} = (t\sigma){\downarrow_{\mathcal{C}(\mathcal{F})}} = (t{\downarrow_{\mathcal{C}(\mathcal{F})}})(\sigma{\downarrow_{\mathcal{C}(\mathcal{F})}})$$

□

Next, we define a function to decide the position q of $t{\downarrow_{\mathcal{C}(\mathcal{F})}}$ such that $(t|_p){\downarrow_{\mathcal{C}(\mathcal{F})}} = (t{\downarrow_{\mathcal{C}(\mathcal{F})}})|_q$, and show a property related to this function.

Definition 167. Let t be a term, p a position in t, and $\mathsf{root}(t) \in \mathcal{F}^{(n)}$. We define the function $\mathsf{C}(t, p)$ that indicates the position of $(t|_p){\downarrow_{\mathcal{C}(\mathcal{F})}}$ in $t{\downarrow_{\mathcal{C}(\mathcal{F})}}$.

$$C(t, p) = \begin{cases} \varepsilon & (\text{if } p = \varepsilon) \\ 1^{(n-i)} \cdot 2 \cdot C(t|_i, q) & (\text{if } p = iq) \end{cases}$$

The following lemma proposes that a position $q2$ of a term $t{\downarrow_{\mathcal{C}(\mathcal{F})}}$ always has the position p of the term t such that $\mathsf{C}(t, p) = q2$.

Lemma 168. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$. If $q2 \in Pos(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$ then there exists $p \in Pos(t)$ such that $\mathsf{C}(t, p) = q2$.*

Proof. Assume that $q2 \in Pos(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$. We show that there exists $p \in Pos(t)$ such that $\mathsf{C}(t, p) = q2$ by structural induction on t.

- If t is a constant, then $Pos(t) = \{\varepsilon\}$, and thus $q2 \notin Pos(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$

- If $t = f(t_1, \ldots, t_n)$, then $q2 \neq \varepsilon$, and thus $p \neq \varepsilon$. We distinguish two cases by q.
  1. If $q = \varepsilon$, then $2 \in Pos(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$. Taking $p = n$, we obtain $\mathsf{C}(t, n) = 1^{(n-n)} \cdot 2 \cdot \mathsf{C}(t_n, \varepsilon) = 2$.
  2. If $q = iq'$, then $iq'2 \in Pos(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$. Let $p = jp'$ where $1 \leq j \leq n$, then we have $\mathsf{C}(t, p) = 1^{(n-j)} \cdot 2 \cdot \mathsf{C}(t_j, p')$. By the induction hypothesis, there exists $\bar{p} \in Pos(t_j)$ such that $\mathsf{C}(t_j, \bar{p}) = q'2$. Hence taking $p' = \bar{p}$ yields $\mathsf{C}(t, p) = 1^{(n-j)} \cdot 2 \cdot q'2$. We shall take j such that $i = 1^{(n-j)} \cdot 2$.

$\square$

Example 169. Consider the TRS from Example 158 and a term $t = \mathsf{take}(\mathsf{from}(x), \mathsf{0})$. We have $Pos(t) = \{\varepsilon, 1, 2, 1.1\}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}} = \mathsf{take}_0 \circ (\mathsf{from}_0 \circ x) \circ \mathsf{0}$. As the following table shows, for every position p in $Pos(t)$ there is the position $\mathsf{C}(t, p)$ in $Pos(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$.

p	$t\vert_p$	$\mathsf{C}(t, p)$	$(t{\downarrow_{\mathcal{C}(\mathcal{F})}})\vert_{\mathsf{C}(t,p)}$
ε	$\mathsf{take}(\mathsf{from}(x), \mathsf{0})$	ε	$\mathsf{take}_0 \circ (\mathsf{from}_0 \circ x) \circ \mathsf{0}$
1	$\mathsf{from}(x)$	1.2	$\mathsf{from}_0 \circ x$
2	$\mathsf{0}$	2	$\mathsf{0}$
1.1	x	1.2.2	x

Moreover, we prepare three lemmata on being overlapping.

Lemma 170. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$, and let t be a term such that $t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ and ρ a renaming. $(t\rho){\downarrow_{\mathcal{C}(\mathcal{F})}} = t{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho$.*

Proof. By Lemma 164, $(t\rho){\downarrow_{\mathcal{C}(\mathcal{F})}} = (t{\downarrow_{\mathcal{C}(\mathcal{F})}})(\rho{\downarrow_{\mathcal{C}(\mathcal{F})}})$. Since ρ is a renaming, every image of ρ is a variable. Hence $\rho{\downarrow_{\mathcal{C}(\mathcal{F})}} = \rho$, and thus $(t{\downarrow_{\mathcal{C}(\mathcal{F})}})(\rho{\downarrow_{\mathcal{C}(\mathcal{F})}}) = t{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho$. $\square$

In order to proof lemmata below, we refer to the standard inference rules for syntactic unification and related lemmata from [1].

$$\text{Eliminate} \quad \frac{\{x \approx t\} \uplus \mathcal{E}}{\{x \approx t\} \cup \mathcal{E}\{x \mapsto t\}} \quad \text{if } x \in \mathit{Var}(t)$$

$$\text{Orient} \quad \frac{\{t \approx x\} \uplus \mathcal{E}}{\{x \approx t\} \cup \mathcal{E}} \quad \text{if } t \notin \mathcal{V}$$

$$\text{Delete} \quad \frac{\{t \approx t\} \uplus \mathcal{E}}{\mathcal{E}}$$

$$\text{Decompose} \quad \frac{\{f(s_1, \ldots, s_n) \approx f(t_1, \ldots, t_n)\} \uplus \mathcal{E}}{\{s_1 \approx t_1, \ldots, s_n \approx t_n\} \cup \mathcal{E}}$$

We write $\Longrightarrow$ for a derivation by the inferences.

Lemma 171 ([1])**.** *A derivation $\Longrightarrow$ is terminating.*

Lemma 172 ([1])**.** *Let $\mathcal{E}$ be a unification problem. $\mathcal{E} \Longrightarrow^* \mathcal{S}$ for some solved form $\mathcal{S}$ if and only if $\mathcal{E}$ is unifiable.*

Lemma 173 ([1])**.** *Let $\mathcal{E}$ and $\mathcal{E}'$ be unification problems. The following two claims are equivalent.*

- *If $\mathcal{E} \Longrightarrow \mathcal{E}'$ then $\mathcal{E}$ is unifiable.*
- *$\mathcal{E}'$ is unifiable.*

Definition 174. Let $\mathcal{S}$ be a unification problem such that $\mathcal{S} = \{s_1 \approx t_1, \ldots, s_n \approx t_n\}$. The *curried unification problem* $\mathcal{S}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is defined by $\mathcal{S}{\downarrow_{\mathcal{C}(\mathcal{F})}} = \{s_1{\downarrow_{\mathcal{C}(\mathcal{F})}} \approx t_1{\downarrow_{\mathcal{C}(\mathcal{F})}}, \ldots, s_n{\downarrow_{\mathcal{C}(\mathcal{F})}} \approx t_n{\downarrow_{\mathcal{C}(\mathcal{F})}}\}$.

Lemma 175. *Let $\mathcal{E}$ be a unification problem. If $\mathcal{E}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is a unifiable then $\mathcal{E}$ is a unifiable.*

Proof. By well-founded induction on $\mathcal{E}$ with respect to $\Longrightarrow$. Assume that $\mathcal{E}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is a unifiable. There exists a solved form $\mathcal{S}'$ such that $\mathcal{E}{\downarrow_{\mathcal{C}(\mathcal{F})}} \Longrightarrow^* \cdot \Longrightarrow^* \mathcal{S}'$. We distinguish four cases by the derivation $\mathcal{E}'{\downarrow_{\mathcal{C}(\mathcal{F})}} \Longrightarrow^* \cdot$.

- If the derivation is due to Delete rule, we have:

$$\mathcal{E} = \{t \approx t\} \uplus \mathcal{E}_0 \Longrightarrow \mathcal{E}_0$$
$$\mathcal{E}{\downarrow_{\mathcal{C}(\mathcal{F})}} = \{t{\downarrow_{\mathcal{C}(\mathcal{F})}} \approx t{\downarrow_{\mathcal{C}(\mathcal{F})}}\} \uplus \mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}} \Longrightarrow \mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}}$$

  By Lemma 173, $\mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is unifiable, which implies that $\mathcal{E}_0$ is unifiable by the induction hypothesis. Hence by Lemma 173, $\mathcal{E}$ is unifiable.

- If the derivation is due to Orient rule, we have:

$$\mathcal{E} = \{t \approx x\} \uplus \mathcal{E}_0 \Longrightarrow \{x \approx t\} \cup \mathcal{E}_0$$
$$\mathcal{E}{\downarrow_{\mathcal{C}(\mathcal{F})}} = \{t{\downarrow_{\mathcal{C}(\mathcal{F})}} \approx x\} \uplus \mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}} \Longrightarrow \{x \approx t{\downarrow_{\mathcal{C}(\mathcal{F})}}\} \cup \mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}}$$

  By Lemma 173, $\{x \approx t{\downarrow_{\mathcal{C}(\mathcal{F})}}\} \cup \mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is unifiable, so is $\mathcal{E}_0$ by the induction hypothesis. By Lemma 173, $\mathcal{E}$ is unifiable.

- If the derivation is due to Eliminate rule, we have:

$$\mathcal{E} = \{x \approx t\} \uplus \mathcal{E}_0 \Longrightarrow \{x \approx t\} \cup \mathcal{E}_0\{x \mapsto t\}$$
$$\mathcal{E}{\downarrow_{\mathcal{C}(\mathcal{F})}} = \{x \approx t{\downarrow_{\mathcal{C}(\mathcal{F})}}\} \uplus \mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}} \Longrightarrow \{x \approx t{\downarrow_{\mathcal{C}(\mathcal{F})}}\} \cup \mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}}\{x \mapsto t{\downarrow_{\mathcal{C}(\mathcal{F})}}\}$$

  By Lemma 173, $\mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}}\{x \mapsto t{\downarrow_{\mathcal{C}(\mathcal{F})}}\} = (\mathcal{E}_0\{x \mapsto t\}){\downarrow_{\mathcal{C}(\mathcal{F})}}$, and thus, by the induction hypothesis $\{x \approx t\} \cup \mathcal{E}_0\{x \mapsto t\}$ are unifiable. By Lemma 173, so is $\mathcal{E}$.

- If the derivation is due to Decompose rule, we have

$$\mathcal{E} = \{f(s_1, \ldots, s_n) \approx f(t_1, \ldots, t_n)\} \uplus \mathcal{E}_0$$
$$\mathcal{E}{\downarrow_{\mathcal{C}(\mathcal{F})}} = \{f \circ s_1{\downarrow_{\mathcal{C}(\mathcal{F})}} \circ \cdots \circ s_n{\downarrow_{\mathcal{C}(\mathcal{F})}} \approx g \circ t_1{\downarrow_{\mathcal{C}(\mathcal{F})}} \circ \cdots \circ t_n{\downarrow_{\mathcal{C}(\mathcal{F})}}\} \uplus \mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}}$$

  As $\mathcal{E}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is unifiable, $f = g$. Hence we have

$$\mathcal{E} \Longrightarrow \{s_1 \approx t_1, ..., s_n \approx t_n\} \cup \mathcal{E}_0$$

  and:

$$\begin{aligned}\mathcal{E}{\downarrow_{\mathcal{C}(\mathcal{F})}} &\Longrightarrow^n \{f \approx g, s_1{\downarrow_{\mathcal{C}(\mathcal{F})}} \approx t_1{\downarrow_{\mathcal{C}(\mathcal{F})}}, \ldots, s_n{\downarrow_{\mathcal{C}(\mathcal{F})}} \approx t_n{\downarrow_{\mathcal{C}(\mathcal{F})}}\} \cup \mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}} \\ &\Longrightarrow \{s_1{\downarrow_{\mathcal{C}(\mathcal{F})}} \approx t_1{\downarrow_{\mathcal{C}(\mathcal{F})}}, \ldots, s_n{\downarrow_{\mathcal{C}(\mathcal{F})}} \approx t_n{\downarrow_{\mathcal{C}(\mathcal{F})}}\} \cup \mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}}\end{aligned}$$

  By Lemma 173, $\{s_1{\downarrow_{\mathcal{C}(\mathcal{F})}} \approx t_1{\downarrow_{\mathcal{C}(\mathcal{F})}}, \ldots, s_n{\downarrow_{\mathcal{C}(\mathcal{F})}} \approx t_n{\downarrow_{\mathcal{C}(\mathcal{F})}}\} \cup \mathcal{E}_0{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is unifiable, and then the induction hypothesis implies that $\{s_1 \approx t_1, ..., s_n \approx t_n\} \cup \mathcal{E}_0$ is unifiable. We conclude that $\mathcal{E}$ is unifiable by Lemma 173.

□

Whether two terms s and t are unifiable depends on unification of the singleton $\{s \approx t\}$. Hence we obtain the following lemma.

Lemma 176. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and s, t terms such that $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$. If $s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}$ are unifiable then s and t are unifiable.*

Lemma 177. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and s, t terms such that $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$, and let q a position of $t{\downarrow_{\mathcal{C}(\mathcal{F})}}$. If $s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable then $q = \varepsilon$ or $q = q'2$.*

Proof. Assume that $s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable. We show that the claim holds by contradiction. Assume that $q = q'1$ further. Since $t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$, it follows that $t{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is head-variable-free, which yields $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q \notin \mathcal{V}$. Hence, as $s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable, $\mathsf{head}(s{\downarrow_{\mathcal{C}(\mathcal{F})}}) = \mathsf{head}(t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q)$. Moreover, the assume that $q = q'1$ implies $|\mathsf{Arg}(t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q)| < |\mathsf{Arg}(s{\downarrow_{\mathcal{C}(\mathcal{F})}})|$. For any substitution σ, we have also $|\mathsf{Arg}((t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q)\sigma)| < |\mathsf{Arg}(s{\downarrow_{\mathcal{C}(\mathcal{F})}}\sigma)|$, and thus $(t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q)\sigma \neq s{\downarrow_{\mathcal{C}(\mathcal{F})}}\sigma$. This contradicts the assumption. □

Now we are ready to show that $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is non-overlapping. Instead of showing this, we propose the contraposition: if $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is overlapping then $\mathcal{R}$ is overlapping. We recall the definition of being overlapping.

Definition 178. A TRS $\mathcal{R}$ over $\mathcal{F}$ is *overlapping* if there exist $\ell_1 \to r_1, \ell_2 \to r_2 \in \mathcal{R}$, renaming substitutions ρ_1, ρ_2 and a position $p \in \mathit{Pos}(\ell_2)$ such that:

- $\ell_1\rho_1$ and $(\ell_2\rho_2)|_p$ are unifiable,
- $p \in \mathit{Pos}_{\mathcal{F}}(\ell_2)$, and
- if $\ell_2 \to r_2$ is a renamed variant of $\ell_1 \to r_1$ then $p \neq \varepsilon$.

We show that these conditions are satisfied.

Lemma 179. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and s, t terms such that $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$, and let ρ be a renaming. If $s\rho = t$ then $s{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$.*

Proof. Assume that $s\rho = t$. As $s\rho = t$, we have $(s\rho){\downarrow_{\mathcal{C}(\mathcal{F})}} = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$. By Lemma 170, it follows that $(s\rho){\downarrow_{\mathcal{C}(\mathcal{F})}} = s{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho$, and thus $s{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$. □

Lemma 180. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and s, t terms such that $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$, and let q a position $\mathit{Pos}(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$. If $s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable then s and $t|_p$ are unifiable for some p.*

Proof. Assume that $s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable. By Lemma 177, $q = \varepsilon$ or $q = q'2$ holds. Hence there exists p such that $\mathsf{C}(t, p) = q$. This implies that $(t|_p){\downarrow_{\mathcal{C}(\mathcal{F})}} = (t{\downarrow_{\mathcal{C}(\mathcal{F})}})|_q$, and thus $s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $(t|_p){\downarrow_{\mathcal{C}(\mathcal{F})}}$ are unifiable. By Lemma 176, s and $t|_p$ are unifiable. $\square$

Lemma 181. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and s, t terms such that $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$, and let q a position $Pos(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$. If the following condition holds:*

- *$s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable, and*
- *$q \in Pos_{\mathcal{F}}(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$*

then, s and $t|_p$ are unifiable for some p and $p \in Pos_{\mathcal{F}}(t)$.

Proof. Assume that $s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable and $q \in Pos_{\mathcal{F}}(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$. As $s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable, by Lemma 180, s and $t|_p$ are unifiable for some p. Taking such a position p, we show that $p \in Pos_{\mathcal{F}}(t)$. Since $q \in Pos_{\mathcal{F}}(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$, we have $(t{\downarrow_{\mathcal{C}(\mathcal{F})}})|_q \notin \mathcal{V}$. This implies $t|_p \notin \mathcal{V}$, and thus $p \in Pos_{\mathcal{F}}(t)$. $\square$

Lemma 182. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and s, t terms such that $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$, and let ρ be a renaming and q a position $Pos(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$. If the following condition holds:*

- *$s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable, and*
- *if $s{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$ then $q \neq \varepsilon$*

then, if $s\rho = t$ then s and $t|_p$ are unifiable for some p and $p \neq \varepsilon$.

Proof. Assume that $s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable, and if $s{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$ then $q \neq \varepsilon$. As $s{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable, by Lemma 180, s and $t|_p$ are unifiable for some p. Take such a position p. Assume that $s\rho = t$. It suffices to show that $p \neq \varepsilon$. By Lemma 179 it follows that $s{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$, which results in $q \neq \varepsilon$ by the second assumption that if $s{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho = t{\downarrow_{\mathcal{C}(\mathcal{F})}}$ then $q \neq \varepsilon$. Assuming further that $p = \varepsilon$, we have $q = \mathsf{C}(t, \varepsilon) = \varepsilon$. This contradicts $q \neq \varepsilon$, and consequently $p \neq \varepsilon$ follows. $\square$

Lemma 183. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and s_1, s_2, t_1, t_2 terms such that $s_1, s_2, t_1, t_2 \in \mathcal{T}(\mathcal{F}, \mathcal{V})$, and let ρ be a renaming and q a position $Pos(t{\downarrow_{\mathcal{C}(\mathcal{F})}})$. If the following condition holds:*

- *$s_1{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t_1{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable, and*
- *if $s_1{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho = t_1{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $s_2{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho = t_2{\downarrow_{\mathcal{C}(\mathcal{F})}}$ then $q \neq \varepsilon$*

then, it follows that: if $s_1\rho = t_1$ and $s_2\rho = t_2$ then s_1 and $t_1|_p$ are unifiable for some

p and $p \neq \varepsilon$.

Proof. Assume that $s_1{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t_1{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable, and if $s_1{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho = t_1{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $s_2{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho = t_2{\downarrow_{\mathcal{C}(\mathcal{F})}}$ then $q \neq \varepsilon$. Assume further that $s_1\rho = t_1$ and $s_2\rho = t_2$. By Lemma 179, we have $s_1{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho = t_1{\downarrow_{\mathcal{C}(\mathcal{F})}}$ As $s_1{\downarrow_{\mathcal{C}(\mathcal{F})}}$ and $t_1{\downarrow_{\mathcal{C}(\mathcal{F})}}|_q$ are unifiable and by Lemma 182, we can conclude that s_1 and $t_1|_p$ are unifiable for some p and $p \neq \varepsilon$. □

From these Lemmata, we have the following lemma on overlapping.

Lemma 184. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$. If $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is overlapping then $\mathcal{R}$ is overlapping.*

Proof. Assume that $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is overlapping. This implies that there exist rules $\ell_1 \to r_1$ and $\ell_2 \to r_2$ in $\mathcal{R}$, renaming substitutions ρ_1, ρ_2 and a position p such that:

- $\ell_1{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho_1$ and $(\ell_2{\downarrow_{\mathcal{C}(\mathcal{F})}}\rho_2)|_q$ are unifiable,
- $q \in Pos_{\mathcal{F}}(\ell_2{\downarrow_{\mathcal{C}(\mathcal{F})}})$, and
- if $(\ell_1{\downarrow_{\mathcal{C}(\mathcal{F})}} \to r_1{\downarrow_{\mathcal{C}(\mathcal{F})}}) \doteq (\ell_2{\downarrow_{\mathcal{C}(\mathcal{F})}} \to r_2{\downarrow_{\mathcal{C}(\mathcal{F})}})$ then $q \neq \varepsilon$.

We show that there exists p such that:

- $\ell_1\rho_1$ and $(\ell_2\rho 2)|_p$ are unifiable,
- $p \in Pos_{\mathcal{F}}(\ell_2)$ and
- if $(\ell_1 \to r_1) \doteq (\ell_2 \to r_2)$ then $p \neq \varepsilon$.

The first condition follows by Lemma 180, and thus there exists a position p such that $\ell_1\rho_1$ and $(\ell_2\rho_2)|_p$ are unifiable. Take such a position p. The second and the third condition follows by Lemma 181 and 183, respectively. □

Finally by Lemmata 162 and 184, orthogonality of $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is shown.

Theorem 185. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$. If $\mathcal{R}$ is orthogonal then $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is orthogonal.*

4.2.2 Left-head-variable-freeness

Clearly, any functional TRS is left-head-variable-free. We show that the left-head-variable-freeness is preserved in ATRSs obtained by currying functional TRSs.

Definition 186. Let t be a term. The currying function $\mathsf{c}(t)$ is defined by $\mathsf{c}(t) = t{\downarrow}_{\mathcal{C}(\mathcal{F})}$.

Lemma 187. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$. If t is head-variable-free and $t \to_{\mathcal{C}(\mathcal{F})} u$ then u is head-variable-free.*

Proof. Assume that t is head-variable-free. There exist a rule $\ell \to r \in \mathcal{C}(\mathcal{F})$, a position p and a substitution σ such that $t|_p = \ell\sigma$ and $u = t[r\sigma]_p$. As t is head-variable-free, so is $t|_p$. It suffices to show that $r\sigma$ is head-variable-free. Since $t|_p = \ell\sigma$, by the definition of $\mathcal{C}(\mathcal{F})$, it follows that $\sigma(x)$ is head-variable-free for every $x \in \mathit{Var}(\ell)$. Since $\mathit{Var}(\ell) = \mathit{Var}(r)$, we conclude that $r\sigma$ is head-variable-free. □

Lemma 188. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$. If t is head-variable-free and $t \to^*_{\mathcal{C}(\mathcal{F})} u$ then u is head-variable-free.*

Proof. By induction on n of $t \to^n_{\mathcal{C}(\mathcal{F})} u$. □

By Lemma 188, we have the following theorem.

Theorem 189. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$. If $\mathcal{R}$ is left-head-variable-free then $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ is left-head-variable-free.*

Proof. Assume that $\mathcal{R}$ is left-head-variable-free. Let $\ell' \to r'$ be an arbitrary rule in $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$. It suffices to show that $\ell' \to r'$ is left-head-variable-free. There exists $\ell \to r \in \mathcal{R}$ such that $\mathsf{c}(\ell) = \ell'$ and $\mathsf{c}(r) = r'$. As $\mathcal{R}$ is left-head-variable-free, ℓ is head-variable-free, and by Lemma 188, $\mathsf{c}(\ell)$ is head-variable-free. Consequently, $\ell' \to r'$ is left-head-variable-free. □

4.2.3 Applicative Constructor Systems

In this part, we show that if a functional TRS $\mathcal{R}$ is a constructor system then $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ is an ACS.

Lemma 190. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and f an n-ary function symbol in $\Sigma_{\mathcal{R}}$. It holds that $f_0 \in \mathcal{AD}_{\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}}$ if and only if $f \in \mathcal{D}_{\mathcal{R}}$.*

Proof.

- For 'if' direction, assume $f \in \mathcal{D}_\mathcal{R}$. Let $\ell \to r \in \mathcal{R}$ be an arbitrary rule such that $\mathsf{root}(\ell) = f$.
  - If ℓ is a constant, then $\ell = \ell{\downarrow_{\mathcal{C}(\mathcal{F})}} = f = f_0$. Hence $\mathsf{head}(\ell{\downarrow_{\mathcal{C}(\mathcal{F})}}) = f_0$, and thus $f_0 \in \mathcal{AD}_{\mathcal{R}\downarrow_{\mathcal{C}(\mathcal{F})}}$.
  - If $\ell = f(t_1, \ldots, t_n)$, then we have

    $$\ell = f(t_1, \ldots, t_n) \to^!_{\mathcal{C}(\mathcal{F})} f_0 \circ t_1{\downarrow_{\mathcal{C}(\mathcal{F})}} \circ \cdots \circ t_n{\downarrow_{\mathcal{C}(\mathcal{F})}} = \ell{\downarrow_{\mathcal{C}(\mathcal{F})}}$$

    which implies $\mathsf{head}(\ell{\downarrow_{\mathcal{C}(\mathcal{F})}}) = f_0$. Since $\ell{\downarrow_{\mathcal{C}(\mathcal{F})}} \to r{\downarrow_{\mathcal{C}(\mathcal{F})}} \in \mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$, it follows that $f_0 \in \mathcal{AD}_{\mathcal{R}\downarrow_{\mathcal{C}(\mathcal{F})}}$.
- For 'only if' direction, assume $f_0 \in \mathcal{AD}_\mathcal{R}$. Let $\ell' \to r'$ be an arbitrary rule in $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ such that $\mathsf{head}(\ell') = f_0$.
  - If ℓ' is a constant, then $\ell' = f_0$. There exists $\ell \to r \in \mathcal{R}$ such that $\ell{\downarrow_{\mathcal{C}(\mathcal{F})}} = \ell'$. As $\ell{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is a constant, so is ℓ. Hence $f = f_0$. Since $\mathsf{root}(\ell) = f$ and $\ell \to r \in \mathcal{R}$, it follows that $f \in \mathcal{D}_\mathcal{R}$.
  - If $\ell' = f_0 \circ t'_1 \circ \cdots \circ t'_n$, as $\ell' \to r' \in \mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$, there exists $\ell \to r \in \mathcal{R}$ such that $\ell{\downarrow_{\mathcal{C}(\mathcal{F})}} = \ell'$. The definition of currying entails that $\ell = f(t_1, \ldots, t_n)$ with $t_1{\downarrow_{\mathcal{C}(\mathcal{F})}} = t'_1, \ldots, t_n{\downarrow_{\mathcal{C}(\mathcal{F})}} = t'_n$. As $\mathsf{root}(\ell) = f$, we obtain $f \in \mathcal{D}_\mathcal{R}$.

□

Lemma 191. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and f an n-ary function symbol in $\Sigma_\mathcal{R}$. If $f \in \mathcal{C}_\mathcal{R}$ then $f \in \mathcal{AC}_{\mathcal{R}\downarrow_{\mathcal{C}(\mathcal{F})}}$.*

Proof. Since $\mathcal{C}_\mathcal{R} = \Sigma_\mathcal{R} \setminus \mathcal{D}_\mathcal{R}$ and $\mathcal{AC}_{\mathcal{R}\downarrow_{\mathcal{C}(\mathcal{F})}} = \mathcal{A}\Sigma_{\mathcal{R}\downarrow_{\mathcal{C}(\mathcal{F})}} \setminus (\mathcal{AD}_{\mathcal{R}\downarrow_{\mathcal{C}(\mathcal{F})}} \uplus \{\circ\})$, it suffices to show the contraposition: if $f_0 \in \mathcal{AD}_{\mathcal{R}\downarrow_{\mathcal{C}(\mathcal{F})}}$ then $f \in \mathcal{D}_\mathcal{R}$. This follows from Lemma 190. □

From Lemmata 190 and 191, the following theorem follows.

Theorem 192. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$. If $\mathcal{R}$ is a constructor system, then $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is an ACS.*

4.2.4 Left-normality

We show that a functional TRS $\mathcal{R}$ is (already) left-normal then $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ is left-normal.

Lemma 193. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and t, u terms. If every left-hand side and right-hand side in $\mathcal{R}$ are left-normal, t is left-normal and $t \xrightarrow{\varepsilon}_{\mathcal{R}} u$ then u is left-normal.*

Proof. As $t \xrightarrow{\varepsilon}_{\mathcal{R}} u$, there exists $\ell \to r \in \mathcal{R}$ and σ such that $t = \ell\sigma$ and $u = r\sigma$. As t is left-normal, $\sigma(x)$ is left-normal for every variable $x \in \mathit{Var}(\ell)$. As r is left-normal, so is $r\sigma$. □

Lemma 194. *Let $\mathcal{R}$ be a TRS and t, u terms. If every left-hand side and right-hand side in $\mathcal{R}$ are left-normal, t is left-normal and $t \to_{\mathcal{R}} u$ then u is left-normal.*

Proof. Assume $t \xrightarrow{p}_{\mathcal{R}} u$. We show that u is left-normal by induction on p.

- If $p = \varepsilon$, the claim holds by Lemma 193.
- If $p = iq$, then t can be denoted by $t = f(t_1, \ldots, t_{i-1}, t_i, t_{i+1}, \ldots, t_n)$ for $1 \leq i \leq n$, where $f \in \mathcal{F}^{(n)}$ and $\{t_1, \ldots, t_n\}$ is the set of its arguments. There exists $\ell \to r \in \mathcal{R}$ and σ such that $t|_{iq} = \ell\sigma$ and $u = t[r\sigma]_{iq}$. Hence we have:

  $$u = f(t_1, \ldots, t_{i-1}, t_i[r\sigma]_q, t_{i+1}, \ldots, t_n)$$

  By the induction hypothesis, $t_i[r\sigma]_q$ is left-normal. Consequently, $u = t[r\sigma]_{iq} = t[t_i[r\sigma]_q]_i$ is left-normal.

□

Lemma 195. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$, and t, u terms. If t is left-normal $t \to_{\mathcal{C}(\mathcal{F})} u$ then u is left-normal.*

Proof. Assume that $t \to_{\mathcal{C}(\mathcal{F})} u$ and t is left-normal. By the definition of $\mathcal{C}(\mathcal{F})$, for every rule $\ell \in r \in \mathcal{C}(\mathcal{F})$, both of ℓ and r are left-normal. As t is left-normal, by Lemma 194, it follows that u is left-normal. □

Lemma 196. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$, and t, u terms. If t is left-normal and $t \to^*_{\mathcal{C}(\mathcal{F})} u$ then u is left-normal.*

Proof. By induction on n of $t \to^n_{\mathcal{C}(\mathcal{F})} u$. □

Finally, we obtain the following theorem by Lemma 196.

Theorem 197. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$. If $\mathcal{R}$ is left-normal then $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ is left-normal.*

4.2.5 Strong-sequentiality

The climax of this chapter is showing strong sequentiality of $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ under the assumption that a TRS $\mathcal{R}$ is strongly sequential. In order to show this we refer to Lemma 90, which requires $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ to be an ACS and orthogonal, and then we suppose that $\mathcal{R}$ is an orthogonal constructor system. Lemmata 185 and 192 ensures that $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ is an orthogonal ACS under this supposition.

We give an overview of our plan. By Lemma 90, we need to show that every proper preredex in $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ has an index. The set of proper preredexes in $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ can be divided into two subsets consisting of (1) head-Ω-free proper preredexes and (2) the others, non-head-Ω-free proper preredexes. For the latter, we have already known existence of their indices by Lemma 103 in Chapter 3. Hence the main theme in this part is showing indices of the former, that is, how to show indices of *all* head-Ω-free proper preredexes. Firstly, we prepare fundamental lemmata on terms and currying.

Lemma 198. *Let s and t be Ω-terms. If $s <_\Omega t$ then $\mathsf{c}(s) <_\Omega \mathsf{c}(t)$.*

Proof. Assume that $s <_\Omega t$. We show that $\mathsf{c}(s) <_\Omega \mathsf{c}(t)$ by structural induction on t.

- If t is a constant, then $s = \Omega$. As $s = \mathsf{c}(s) = \Omega$ and $t = \mathsf{c}(t) \neq \Omega$, it follows that $\mathsf{c}(s) <_\Omega \mathsf{c}(t)$.
- IF $t = f(t_1, \ldots, t_n)$, we distinguish two cases by s.
  1. If $s = \Omega$, then $\mathsf{c}(s) = s = \Omega$. As $\mathsf{c}(t) \neq \Omega$, we obtain $\mathsf{c}(s) <_\Omega \mathsf{c}(t)$.
  2. If $s = f(s_1, \ldots, s_n)$, then we have:

$$\mathsf{c}(t) = f_0 \circ \mathsf{c}(t_1) \circ \cdots \circ \mathsf{c}(t_n)$$
$$\mathsf{c}(s) = f_0 \circ \mathsf{c}(s_1) \circ \cdots \circ \mathsf{c}(s_n)$$

  Since $s <_\Omega t$, there exists i such that $s_i <_\Omega t_i$ and $s_j \leq_\Omega t_j$ for $j \neq i$. By the induction hypothesis, we have $\mathsf{c}(s_i) <_\Omega \mathsf{c}(t_i)$, and consequently $\mathsf{c}(s) <_\Omega \mathsf{c}(t)$ follows.

□

Lemma 199. *Let t a term.* $\mathsf{c}(t_\Omega) = (\mathsf{c}(t))_\Omega$.

Proof. We show that the claim holds by structural induction on t.

- If $t \in \mathcal{V}$, then $t_\Omega = \mathsf{c}(t_\Omega) = \Omega$. From $\mathsf{c}(t) = t$, we have $(\mathsf{c}(t))_\Omega = t_\Omega = \Omega$.
- If t is a constant, then $\mathsf{c}(t_\Omega) = t_\Omega = t = \mathsf{c}(t) = (\mathsf{c}(t))_\Omega$.
- If $t = f(t_1, \dots, t_n)$, then $t_\Omega = f(t_{1\Omega}, \dots, t_{n\Omega})$. Hence $\mathsf{c}(t_\Omega) = f_0 \circ \mathsf{c}(t_{1\Omega}) \circ \cdots \circ \mathsf{c}(t_{n\Omega})$. We also have $\mathsf{c}(t) = f_0 \circ \mathsf{c}(t_1) \circ \cdots \circ \mathsf{c}(t_n)$, and thus $(\mathsf{c}(t))_\Omega = f_0 \circ (\mathsf{c}(t_1))_\Omega \circ \cdots \circ (\mathsf{c}(t_n))_\Omega$. By the induction hypotheses $\mathsf{c}(t_{1\Omega}) = (\mathsf{c}(t_1))_\Omega, \dots, \mathsf{c}(t_{n\Omega}) = (\mathsf{c}(t_n))_\Omega$ hold, and thus $\mathsf{c}(t_\Omega) = (\mathsf{c}(t))_\Omega$ follows.

□

Secondly, we show that a proper preredex t of a TRS $\mathcal{R}$ becomes a proper preredex $\mathsf{c}(t)$ of the curried ATRS $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ and $\mathsf{c}(t)$ has an index.

Lemma 200. *Let t be an Ω-term. If t is a proper preredex in $\mathcal{R}$ then $\mathsf{c}(t)$ is a proper preredex in $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$.*

Proof. Assume that t is a proper preredex in $\mathcal{R}$. This implies that there exists $\ell \to r \in \mathcal{R}$ such that $\ell <_\Omega \ell_\Omega$. As $\mathsf{c}(\ell) \to \mathsf{c}(r) \in \mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$, it suffices to show that $\mathsf{c}(\ell) <_\Omega (\mathsf{c}(\ell))_\Omega$. By Lemma 198, $t <_\Omega \ell_\Omega$ yields $\mathsf{c}(t) <_\Omega \mathsf{c}(\ell_\Omega)$. By Lemma 199, we have $\mathsf{c}(\ell_\Omega) = (\mathsf{c}(\ell))_\Omega$, and thus $\mathsf{c}(t) <_\Omega (\mathsf{c}(\ell))_\Omega$ follows. □

The above lemma says that part of proper preredexes of $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ can be obtained from proper preredexes of $\mathcal{R}$. We shall continue to show that they are in the set of head-Ω-free proper preredexes.

Lemma 201. *Let t be an Ω-term. If t is head-Ω-free then $\mathsf{c}(t)$ is head-Ω-free.*

Proof. Assume that t is head-Ω-free. We show that $\mathsf{c}(t)$ is head-Ω-free by structural induction on t.

- If $t = \Omega$, then $\mathsf{c}(t) = t = \Omega$, and thus $\mathsf{c}(t)$ is head-Ω-free.
- If $t = f(t_1, \dots, t_n)$, then $\mathsf{c}(t) = f_0 \circ \mathsf{c}(t_1) \circ \cdots \mathsf{c}(t_n)$. Since t is head-Ω-free, so are its subterms $t_1, \dots, t_n$. By the induction hypotheses, $\mathsf{c}(t_1), \dots, \mathsf{c}(t_n)$ are head-Ω-free, and thus $\mathsf{c}(t)$ is head-Ω-free.

□

Lemma 202. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and t be an Ω-term. If t is a proper preredex in $\mathcal{R}$ then $\mathsf{c}(t)$ is head-Ω-free.*

Proof. Assume that t is a proper preredex in $\mathcal{R}$. There exists $\ell \to r \in \mathcal{R}$ such that $t \leq_\Omega \ell_\Omega$. Since ℓ does not contain $\circ$, neither do t and ℓ_Ω. Hence t is head-Ω-free. By Lemma 201, it follows that $\mathsf{c}(t)$ is head-Ω-free. $\square$

Lemma 202 suggests that (at least) part of head-Ω-free proper preredexes of $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ originate proper preredexes of $\mathcal{R}$.

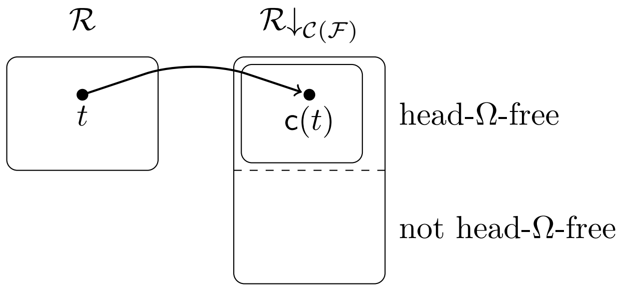

Figure 4.1 Proper preredexes in TRSs

In order to show that these head-Ω-free proper preredexes have indices, we focus on relation between an Ω-position of an proper preredex and a redex scheme.

Lemma 203. *Let t be a term and p a position. If $p \in Pos_{\mathcal{F}}(t)$ then $\mathsf{C}(t,p) \in Pos_{\mathcal{F}}(\mathsf{c}(t))$.*

Proof. We show that the claim holds by the contraposition: if $\mathsf{C}(t,p) \in Pos_{\mathcal{F}}(\mathsf{c}(t))$ then $p \in Pos_{\mathcal{F}}(t)$. Assume that $\mathsf{C}(t,p) \notin Pos_{\mathcal{F}}(\mathsf{c}(t))$. This is equivalent to $\mathsf{C}(t,p) \in Pos_{\mathcal{V}}(\mathsf{c}(t))$, and thus $\mathsf{c}(t)|_{\mathsf{C}(t,p)} \in \mathcal{V}$. Since $\mathsf{c}(t)|_{\mathsf{C}(t,p)} = \mathsf{c}(t|_p)$, it follows that $t|_p \in \mathcal{V}$. This yields $p \in Pos_{\mathcal{V}}(t)$, and therefore $p \notin Pos_{\mathcal{F}}(t)$. $\square$

This lemma says that if a subterm of t at a position p is in $\mathcal{T}(\mathcal{F}, \mathcal{V})$, then the subterm of $\mathsf{c}(t)$ at the corresponding position $\mathsf{C}(t,p)$ to p after currying is in $\mathcal{T}(\mathcal{F} \uplus \{\circ\}, \mathcal{V})$, i.e. $\mathsf{c}(t)|_{\mathsf{C}(t,p)}$ cannot be a variable. Moreover, we show that an Ω-position p of a proper preredex t is an index if $r|_p$ is not a variable for every redex scheme r such that $t <_\Omega r$. Recall Lemma 101 and 102 from Chapter 3.

Lemma 204 (from Lemma 101)**.** *Let $\mathcal{R}$ be an orthogonal ℓ-ACS over $\mathcal{F}$, t an Ω-term and p an Ω-position of t. The following two statements are equivalent.*

1. *For every rule $\ell \to r \in \mathcal{R}$, if $t <_\Omega \ell_\Omega$ then $p \in Pos_{\mathcal{F}}(\ell)$.*

2. $p \in \mathcal{I}_{\mathcal{R}}(t)$.

From observation of $Pos_{\mathcal{F}}$ and indices in Lemmata 203 and 204, now we can say that, for every proper preredex t in $\mathcal{R}$ and redex scheme r such that $t <_{\Omega} r$, if $p \in Pos_{\Omega}(t)$ is an index in $\mathcal{R}$ then $\mathsf{C}(t,p) \in Pos_{\Omega}(\mathsf{c}(t))$ is an index in $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$. The subterm $\mathsf{c}(r)|_{\mathsf{C}(t,p)}$ is never a variable as $r|_p$ is not a variable.

Lemma 205. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$, t be an Ω-term and p a position of t. If $p \in \mathcal{I}_{\mathcal{R}}(t)$ then $\mathsf{C}(t,p) \in \mathcal{I}_{\mathcal{R}\downarrow_{\mathcal{C}(\mathcal{F})}}(\mathsf{c}(t))$.*

Proof. Assume that $p \in \mathcal{I}_{\mathcal{R}}(t)$, and let $\ell \to r$ be an arbitrary rule in $\mathcal{R}$. Since $\mathsf{c}(\ell) \to \mathsf{c}(r) \in \mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$, by Lemma 204, it suffices to show that if $\mathsf{c}(t) <_{\Omega} (\mathsf{c}(\ell))_{\Omega}$ then $\mathsf{C}(t,p) \in Pos_{\mathcal{F}}(\mathsf{c}(\ell))$.

Assume that $\mathsf{c}(t) <_{\Omega} (\mathsf{c}(\ell))_{\Omega}$. Since $(\mathsf{c}(\ell))_{\Omega} = \mathsf{c}(\ell_{\Omega})$ by Lemma 199, we have $\mathsf{c}(t) <_{\Omega} \mathsf{c}(\ell_{\Omega})$, and thus it follows that $t <_{\Omega} \ell_{\Omega}$ by Lemma 198. As $p \in \mathcal{I}_{\mathcal{R}}(t)$, by Lemma 204 for every $\ell \to r \in \mathcal{R}$, if $t <_{\Omega} \ell_{\Omega}$ then $p \in Pos_{\mathcal{F}}(\ell)$. Hence we obtain $p \in Pos_{\mathcal{F}}(\ell)$, which implies $\mathsf{C}(t,p) \in Pos_{\mathcal{F}}(\mathsf{c}(\ell))$ by Lemma 203. Since $p \in Pos_{\mathcal{F}}(t)$ and $Pos(t) \subseteq Pos(\ell_{\Omega}) = Pos(\ell)$ from $t <_{\Omega} \ell_{\Omega}$, we have $\mathsf{C}(\ell,p) = \mathsf{C}(t,p)$. Consequently, it follows that $\mathsf{C}(t,p) \in Pos_{\mathcal{F}}(\mathsf{c}(\ell))$. □

Finally, we give the following lemma to show that *all* head-Ω-free proper preredexes of $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$ originate the proper preredexes in $\mathcal{R}$.

Lemma 206. *Let $\mathcal{R}$ be a TRS over $\mathcal{F}$ and t a proper preredex of $\mathcal{R}{\downarrow_{\mathcal{C}(\mathcal{F})}}$. If t is head-Ω-free then there exists s such that $\mathsf{c}(s) = t$.*

Proof. Assume that t is head-Ω-free. We show that the claim holds by structural induction on t.

- If $t = \Omega$, then $\mathsf{c}(t) = t$. Take $s = t$.
- If $t = f_0 \circ t_1 \circ \cdots \circ t_n$, since t is head-Ω-free, its subterms $t_1, \ldots, t_n$ are head-Ω-free. By the induction hypotheses, there exist s_i such that $\mathsf{c}(s_i) = t_i$ for $1 \leq i \leq n$. Hence it suffices to take $s = f(s_1, \ldots, s_n)$, and then we shall obtain $\mathsf{c}(s) = f_0 \circ \mathsf{c}(s_1) \circ \cdots \circ \mathsf{c}(s_n) = f_0 \circ t_1 \circ \cdots \circ t_n$.

□

With the arguments which we have seen above, we can explain that the set of proper preredexes of $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ consists of head-Ω-free proper preredexes, all of which are obtained by currying proper preredexes of $\mathcal{R}$, and non-head-Ω-free proper preredexes. Eventually, all proper preredexes have indices.

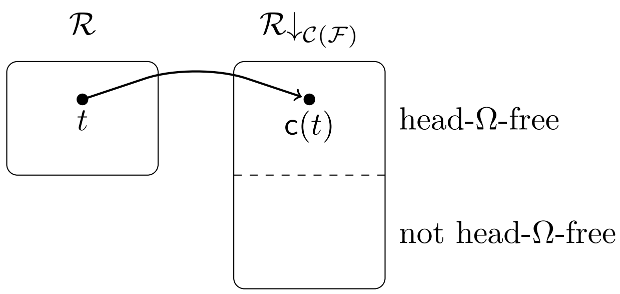

Figure 4.2 Proper preredexes in TRSs

We oversee this part with the next theorem concluding strong sequentiality of $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$.

Theorem 207. *Let $\mathcal{R}$ an orthogonal constructor system over $\mathcal{F}$. If $\mathcal{R}$ is strongly sequential then $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ is strongly sequential.*

Proof. Assume that $\mathcal{R}$ is strongly sequential. Orthogonality of $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ and being an ACS for $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ are ensured by Theorems 185 and 192, respectively. By Lemma 90, it suffices to show that every proper preredex in $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ has an index. As $\mathcal{R}$ is strongly sequential, by Lemma 75 we can say that every proper preredex t in $\mathcal{R}$ has an index. For such a proper preredex t, Lemmata 200 and 205 show that $\mathsf{c}(t)$ is a proper preredex in $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ and it has an index. We consider two sets of proper preredexes of $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$.

- For the set of head-Ω-free proper preredexes, Lemma 206 shows that this set coincides the set consisting of $\mathsf{c}(t)$ for every proper preredex t of $\mathcal{R}$. As we have seen, every element of the set has an index.
- For the set of non-head-Ω-free proper preredexes, Lemma 103 promises existence of an index of every non-head-Ω-free proper preredex.

Therefore, every proper preredex in $\mathcal{R}{\downarrow}_{\mathcal{C}(\mathcal{F})}$ has an index. $\square$

Chapter 5

Conclusion

In this thesis, we have developed a technique for left-normal translation for ATRSs. By introducing the concepts of applicative constructor systems and left-head-variable-freeness, we overcame the difference between the forms of terms in functional TRSs and ATRSs. In addition, the fact that the ATRS $\mathsf{AB}(\mathcal{R})$ obtained by one-step translation has strong sequentiality mentions that repeated one-step translations of $\mathsf{AB}(\mathcal{R})$ result in the left-normal ATRS $\mathsf{L}(\mathcal{R})$. Hence we can say that every strongly sequential and orthogonal non-variadic ACS is a preimage of some orthogonal left-normal ACS.

In Chapter 4, we showed that left-normal translation can deal with all functional TRSs that can be left-normal by the translation for functional TRSs. This means that our new result includes the result of left-normal translation by Hashida [4], and contributes to extending a class of TRSs that can be translated into left-normal systems.

Future work

One of what we have left undone is to moderate restrictions on non-left-normal ATRSs to be translated. Left-normal translation for ATRSs requires non-left-normal ATRSs to be strongly sequential, orthogonal, non-variadic, left-head-variable-free and ACSs. In spite of a number of conditions, we have already succeeded in handling orthogonal ACSs which most functional programming languages and proof assistants support, and including all functional TRSs with which the Hashida's translation can deal. However, in order to make the translation generalised to ATRSs further, we need to decrease the restrictions. Strong sequentiality is considered indispensable because there hardly exists an alternative concept to compute needed positions, whilst necessity of the other properties is not conclusive.

Moreover, we asked currying to translate functional TRSs in order to include the result of Hashida [4] because of different ways of showing index transitivity and strong sequentiality between functional and applicative TRSs. Absolutely functional TRSs are left-head-variable-free and non-variadic; if left-normal translation does not require TRSs to be (applicative) constructor systems, such a bypass is no longer needed. We anticipate a more general condition on which non-left normal TRSs can be translated without the division between the two classes of TRSs.

As for another sequentiality using tree automata, various classes have been proposed for TRSs. We are uncertain of whether translated ATRSs are in these classes. Especially, the class of NV-sequentiality, which is introduced by Oyamaguchi [10], is known to coincide the class of strong sequentiality in orthogonal constructor systems [7]. We conclude the thesis by expecting the coincidence in orthogonal *applicative* constructor systems.

Acknowledgement

At the end of this thesis, I would like to express my gratitude to my supervisor Prof. Hirokawa for continual support and guidance. Furthermore I would like to thank Prof. Ishihara and Prof. Ogawa who gave me a lot of advice on my study in the mid-term defence, and colleagues at Hirokawa Laboratory in JAIST.

Bibliography

[1] F. Baader and T. Nipkow. *Term rewriting and all that.* Cambridge university press, 1999.

[2] Y. Bertot. Coq in a hurry. *arXiv preprint cs/0603118*, 2006.

[3] I. Durand and A. Middeldorp. Decidable call-by-need computations in term rewriting. *Information and Computation*, 196(2):95–126, 2005.

[4] A. Hashida. Transformation-based normalization analysis for term rewriting. Master's thesis, JAIST, 2019.

[5] G. Huet and J.-J. Lévy. Computations in orthogonal rewriting systems, II. In *Computational Logic – Essays in Honor of Alan Robinson*, pages 395–414. The MIT Press, 1991.

[6] J. W. Klop and A. Middeldorp. Sequentiality in orthogonal term rewriting systems. *Journal of Symbolic Computation*, 12(2):161–195, 1991.

[7] S. Lucas. Strong and NV-sequentiality of constructor systems. *Information processing letters*, 89(4):191–201, 2004.

[8] A. Middeldorp N. Hirokawa and H. Zankl. Uncurrying for termination and complexity. *Journal of Automated Reasoning*, 50(3):279–315, 2013.

[9] M. J. O'Donnell. Computing in systems described by equations. 1977.

[10] M. Oyamaguchi. NV-sequentiality: a decidable condition for call-by-need computations in term-rewriting systems. *SIAM Journal on Computing*, 22(1):114–135, 1993.

[11] S. P. Jones P. Hudak, J. Hughes and P. Wadler. A history of haskell: being lazy with class. In *Proceedings of the third ACM SIGPLAN conference on History of programming languages*, pages 12–1, 2007.