Term Rewriting

LVA 703141 (26W)

Aart Middeldorp
University of Innsbruck

Do Not Distribute

February 28, 2026

Contents

Part I

Introduction

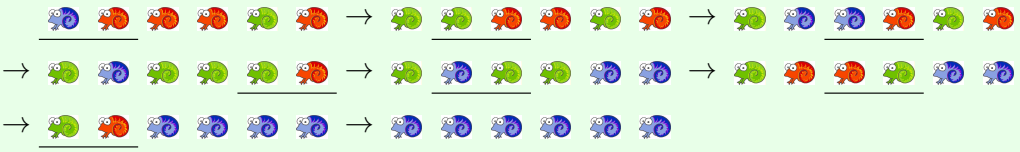
Examples

Rewriting is a pervasive concept in mathematics, computer science, and other areas; simplification of expressions constitutes rewriting, the execution of a program can be seen as a rewrite sequence on program states, and in fact probably almost any development according to a set of fixed rules can be considered rewriting. In *term rewriting*, we assume that the objects which are rewritten are terms. This yields a powerful formalism which is crucial for simplification in automated theorem proving, it provides tools to analyze security protocols, it can be used to model the development of RNA structures, but it is also a versatile method in program verification, to name only a few application areas. In fact, term rewriting is a Turing-complete model of computation, and provides methods to investigate important properties of computation and simplification processes on an abstract level. To give an idea of the problems addressed in term rewriting, in this preliminary chapter we present a couple of examples.

Example 1. A colony of chameleons includes 20 red, 18 blue, and 16 green individuals. Whenever two chameleons of different color meet, each changes to the third color. Some time passes during which no chameleons are born or die nor do any enter or leave the colony. Is it possible that at the end of this period, all 54 chameleons are the same color? To simplify matters, we represent a colony as a string of chameleons, and chamelons meet only direct neighbours. So the following meeting rules are used:

For instance, consider the following colony consisting of 4 red, 1 blue, and 1 green chameleons:

In several steps this colony is transformed into an all-blue colony:

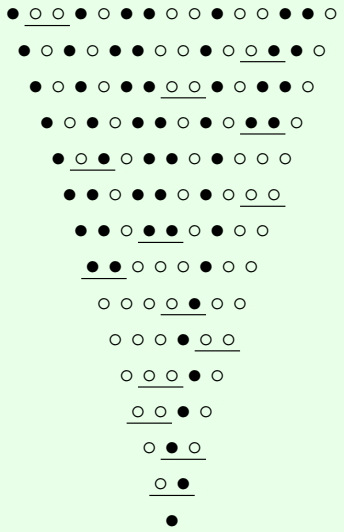


How to find such a sequence? Is it decidable whether initial colonies can be transformed into monochrome colonies? What is the answer to the original question?

Example 2. Coffee beans come in two kinds called black (●) and white (○). A two-player game starts with a random sequence of black and white beans. In a move, a player must take two adjacent beans and put back one bean, according to the following set of rules:

$$●● \to ○ \qquad ○○ \to ○ \qquad ●○ \to ● \qquad ○● \to ●$$

The player who puts the last black bean wins. For instance, the following is a valid game:



In this case the player who started lost, since the last black bean was put in the 14th move. A number of interesting questions can be asked about such a game: Which moves should the respective players perform to win? Are there game states which are equivalent in the sense that they offer the same opportunities to each of the players? In short, is there a winning strategy for one of the players? While it is obvious that the above game terminates, is this still the case for the modified game using the rules

$$●● \to ○○○○ \qquad ○○ \to ○ \qquad ●○ \to ○○○● \qquad ○● \to ●$$

and if yes, how many steps are needed?

The next example originates from a Dutch popular science magazin.

Example 3. A team of genetic engineers decides to create cows that produce cola instead of milk. To that end they have to transform the DNA of the milk gene

TAGCTAGCTAGCT

in every fertilized egg into the cola gene

CTGACTGACT

Techniques exist to perform the following DNA substitutions

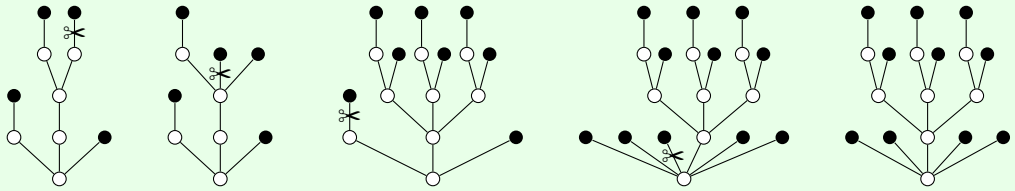
$$\mathsf{TCAT} \leftrightarrow \mathsf{T} \qquad \mathsf{GAG} \leftrightarrow \mathsf{AG} \qquad \mathsf{CTC} \leftrightarrow \mathsf{TC} \qquad \mathsf{AGTA} \leftrightarrow \mathsf{A} \qquad \mathsf{TAT} \leftrightarrow \mathsf{CT}$$

Recently it has been discovered that the mad cow disease is caused by a retrovirus with the following DNA sequence CTGCTACTGACT. What now, if accidentally cows with this virus are created? According to the engineers there is little risk because this never happened in their experiments, but various action groups demand absolute assurance.

The following questions naturally arise. How to tranform TAGCTAGCTAGCT into CTGACTGACT? How does one show that TAGCTAGCTAGCT cannot be transformed into CTGCTACTGACT?

In the first three examples the objects to rewrite were strings of symbols. In the next example, known as the Battle of Hercules and Hydra, rewriting is applied to (unordered) trees.

Example 4. The mythological monster Hydra is a dragon-like creature with multiple heads. Whenever Hercules in his fight chops off a head, more and more new heads can grow instead, since the beast gets increasingly angry. Here we model a Hydra as an unordered tree. If Hercules cuts off a leaf corresponding to a head, the tree is modified in the following way: If the cut-off node h has a grandparent n, then the branch from n to the parent of h gets multiplied, where the number of copies depends on the number of decapitations so far. Hydra dies if there are no heads left, in that case Hercules wins. The following sequence shows an example fight:



0 1 2 3 4

Though the number of heads can grow considerably in one step, it turns out that the fight always terminates, and Hercules will win independent of his strategy. This can be shown by an argument based on ordinals.

The final two examples are more down to earth. They address the question how to compute the addition of natural numbers, represented as terms.

Example 5. Suppose we represent natural numbers in unary term notation. So the objects that we will rewrite are expressions built from a constant symbol $\mathsf{0}$ and a unary function symbol s, representing the successor function. We add a binary function symbol a to perform addition. Some valid expressions are $\mathsf{s}(\mathsf{s}(\mathsf{0}))$, representing the number 2, and $\mathsf{a}(\mathsf{s}(\mathsf{0}),\mathsf{s}(\mathsf{0}))$, representing $1+1$. Just two rules are required to transform the latter into the former:

$$\mathsf{a}(\mathsf{0},y) \to y \qquad\qquad \mathsf{a}(\mathsf{s}(x),y) \to \mathsf{s}(\mathsf{a}(x,y))$$

The rules contain the placeholders x and y. When applying the rules, these can be instantiated by arbitrary terms:

$$\underline{\mathsf{a}(\mathsf{s}(\mathsf{0}),\mathsf{s}(\mathsf{0}))} \to \mathsf{s}(\underline{\mathsf{a}(\mathsf{0},\mathsf{s}(\mathsf{0}))}) \to \mathsf{s}(\mathsf{s}(\mathsf{0}))$$

In the first rewrite step we use the second rule with the variable binding $x \mapsto \mathsf{0}$ and $y \mapsto \mathsf{s}(\mathsf{0})$. In the second step the first rule is applied (with $y \mapsto \mathsf{s}(\mathsf{0})$) to the underlined subexpression $\mathsf{a}(\mathsf{0},\mathsf{s}(\mathsf{0}))$. To the final expression $\mathsf{s}(\mathsf{s}(\mathsf{0}))$ no rule can be applied. Such expressions are called normal forms and play an important role in term rewriting.

By changing the radix of the representation, addition can be computed more efficiently.

Example 6. The rules below implement addition on natural numbers in decimal notation:

$$
\begin{array}{llll}
0+0\to 0 & 1+0\to 1 & \cdots & 9+0\to 9 \qquad 0:x\to x \\
0+1\to 1 & 1+1\to 2 & \cdots & 9+1\to 1:0 \qquad x+(y:z)\to y:(x+z) \\
0+2\to 2 & 1+2\to 3 & \cdots & 9+2\to 1:1 \qquad (x:y)+z\to x:(y+z) \\
0+3\to 3 & 1+3\to 4 & \cdots & 9+3\to 1:2 \qquad x:(y:z)\to (x+y):z \\
0+4\to 4 & 1+4\to 5 & \cdots & 9+4\to 1:3 \\
0+5\to 5 & 1+5\to 6 & \cdots & 9+5\to 1:4 \\
0+6\to 6 & 1+6\to 7 & \cdots & 9+6\to 1:5 \\
0+7\to 7 & 1+7\to 8 & \cdots & 9+7\to 1:6 \\
0+8\to 8 & 1+8\to 9 & \cdots & 9+8\to 1:7 \\
0+9\to 9 & 1+9\to 1:0 & \cdots & 9+9\to 1:8
\end{array}
$$

Here numbers are built from the constants $\mathsf{0}-\mathsf{9}$ and the binary function symbol : which is used in infix notation. So the number 12 is represented by $\mathsf{1}:\mathsf{2}$ and the term $(\mathsf{3}:\mathsf{4}):\mathsf{5}$ represents 345. For addition the infix symbol + is used. Besides the 100 rules from the standard addition table and the rule $\mathsf{0}:x\to x$ to remove leading zeros, the three remaining rules in the last column take care of adding numbers greater than 10. For instance, $23+77$ can be evaluated as follows (the applied rule is given on the right):

$$
\begin{array}{lll}
\underline{(2:3)+(7:7)} & \to 7:(\underline{(2:3)+7}) & x+(y:z)\to y:(x+z) \\
& \to 7:(2:(\underline{3+7})) & (x:y)+z\to x:(y+z) \\
& \to 7:(\underline{2:(1:0)}) & 3+7\to 1:0 \\
& \to 7:((\underline{2+1}):0) & x:(y:z)\to (x+y):z \\
& \to \underline{7:(3:0)} & 2+1\to 3 \\
& \to (\underline{7+3}):0 & x:(y:z)\to (x+y):z \\
& \to (1:0):0 & 7+3\to 1:0
\end{array}
$$

As this is not the only possibility to evaluate $23+77$, the question arises whether the outcome (normal form) depends on the evaluation strategy to determine which rule to apply to which subexpression. Also, do computations always terminate?

The first three exercises below are challenging. Do not despair! In this book you learn techniques that readily apply to solve them.

Exercises

1 Solve the chameleon puzzle in Example 1.

2 Consider the coffee bean game in Example 2 with initial configuration ● ○ ○ ● ○ ● ● ○ ○ ● ○ ○ ● ● ○. Is it possible that the first player wins the game?

3 Consider the DNA substitutions in Example 3.

a Transform the milk gene TAGCTAGCTAGCT into the cola gene CTGACTGACT.

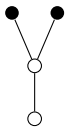
b Show that the milk gene TAGCTAGCTAGCT cannot be transformed into the mad cow retrovirus CTGCTACTGACT.

4 ***a*** Compute a normal form of the term $2 + (4 + (6 + (4 + 2)))$ with respect to the rewrite rules in Example 6.

b Extend the rewrite rules in Example 6 with rules that compute the multiplication of natural numbers. Illustrate your rules on 11×46.

c Repeat parts (a) and (b) in the setting of Example 5.

5 Consider the battle of Hercules and Hydra in Example 4. Present two different battles with the Hydra on the right:



Bibliographic Notes

Example 2 describes variants in [33] of the Coffee Can Problem from Gries [49]. Example 3 is a puzzle contest from a Dutch science magazin [1]. Kirby and Paris [70] introduced the Hydra battle and showed that its termination cannot be proved in Peano arithmetic. The rewrite rules in Example 6 are taken from Walters and Zantema [135].

Part II

Basic Topics

Chapter 1

Abstract Rewrite Systems

Several important notions and results in term rewriting do not depend on the term structure, but can already be stated in the more general framework of abstract rewrite systems. In this chapter we introduce basic concepts and properties of abstract rewrite systems. This avoids having to repeat definitions and results when later on we study conditional rewriting (in Chapter 11) and polynomial rewriting (in Chapter 13), which go beyond the pure term rewriting format presented in Chapter 3. In the first two sections we define the concept of abstract rewriting and introduce several properties. In Section 1.3 we establish the relationship—known as Newman's Lemma—between three important properties. Section 1.4 presents some easy abstract "divide and conquer" results and in Section 1.5 we give an abstract account of strategies.

1.1 Definitions

Our first definition introduces the object that will be studied in this chapter.

Definition 1.1.1. An *abstract rewrite system* (ARS for short) is a pair $\mathcal{A} = \langle A, \to \rangle$ consisting of a set A and a binary relation $\to$ on A. Instead of $(a, b) \in \to$ we write $a \to b$ and we say that $a \to b$ is a *rewrite step*. The set A is called the *domain* of $\mathcal{A}$.

It is instructive to view an ARS $\langle A, \to \rangle$ as a directed (possibly infinite) graph. The nodes of the graph are the elements of A and there is an arrow from node a to node b if and only if $a \to b$. An example of this representation is given in Figure 1.1.

Example 1.1.2. The ARS depicted in Figure 1.1 is formally defined as $\mathcal{A} = \langle A, \to \rangle$ with $A = \{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d}, \mathsf{e}, \mathsf{f}, \mathsf{g}\}$ and $\to = \{(\mathsf{a}, \mathsf{e}), (\mathsf{b}, \mathsf{a}), (\mathsf{b}, \mathsf{c}), (\mathsf{c}, \mathsf{d}), (\mathsf{c}, \mathsf{f}), (\mathsf{e}, \mathsf{b}), (\mathsf{e}, \mathsf{g}), (\mathsf{f}, \mathsf{e}), (\mathsf{f}, \mathsf{g})\}$.

Sequences of rewrite steps yield rewrite sequences.

Definition 1.1.3. Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS. A (finite) *rewrite sequence* is a non-empty sequence $(a_0, \ldots, a_n)$ of elements in A such that $a_i \to a_{i+1}$ for all $0 \leqslant i < n$. We usually write $a_0 \to \cdots \to a_n$ for $(a_0, \ldots, a_n)$. The number n is called the *length* of the rewrite sequence. Given elements $a, b \in A$ we say that a *rewrites* to b and we write $a \to^* b$ if there exists a rewrite sequence $(a_0, \ldots, a_n)$ with $a = a_0$ and $b = a_n$. If $a \to^* b$ then we call b a *reduct* or *descendant* of a.

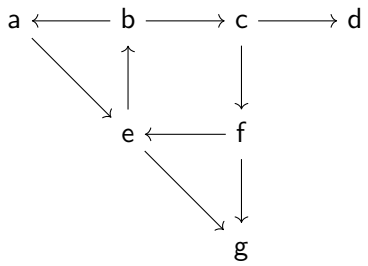


Figure 1.1: An example ARS.

Example 1.1.4. In the ARS of Figure 1.1 we have $\mathsf{a} \to^* \mathsf{c}$ as witnessed by the rewrite sequence $\mathsf{a} \to \mathsf{e} \to \mathsf{b} \to \mathsf{c}$. This is not the only rewrite sequence from a to c. Another one is $\mathsf{a} \to \mathsf{e} \to \mathsf{b} \to \mathsf{c} \to \mathsf{f} \to \mathsf{e} \to \mathsf{b} \to \mathsf{a} \to \mathsf{e} \to \mathsf{b} \to \mathsf{c}$.

One can prove that $\to^*$ is the transitive and reflexive closure of $\to$, i.e., the least extension of $\to$ that is both transitive and reflexive. In particular, every element rewrites to itself, that is, we consider also rewrite sequences without rewrite steps. Infinite rewrite sequences are also of interest.

Definition 1.1.5. Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS. An *infinite rewrite sequence* is an infinite sequence $(a_i)_{i \in \mathbb{N}}$ of elements in A such that $a_i \to a_{i+1}$ for all $i \in \mathbb{N}$.

In the next few definitions several other derived relations of the basic relation $\to$ are introduced.

Definition 1.1.6. Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS. We write $a \to^+ b$ if there exists a rewrite sequence from a to b containing at least one rewrite step. We write $a \to^= b$ if $a \to b$ or $a = b$.

The relation $\to^+$ is the transitive closure of $\to$ and $\to^=$ is its reflexive closure. We write $a \leftarrow b$ if $b \to a$. The relations ${}^*\!\leftarrow$, ${}^+\!\leftarrow$, and ${}^=\!\leftarrow$ are defined similarly.

When establishing properties of ARSs we frequently use induction on the length of rewrite sequences. For that purpose it is convenient to define approximations of $\to^*$ and $\to^+$, as follows.

Definition 1.1.7. Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS. We write $a \to^n b$ ($n \geqslant 0$) if there exists a rewrite sequence from a to b of length n. We write $a \, {}^n\!\leftarrow b$ if $b \to^n a$.

An inductive definition of $\to^n$ is not hard to give (cf. Definition A.1.6). We have $\to^* = \bigcup \{ \to^n \mid n \geqslant 0 \}$ and $\to^+ = \bigcup \{ \to^n \mid n \geqslant 1 \}$.

Definition 1.1.8. Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS. Two elements $a, b \in A$ are *joinable*, denoted by $a \downarrow b$, if there exists an element $c \in A$ such that $a \to^* c \, {}^*\!\leftarrow b$. Here $a \to^* c \, {}^*\!\leftarrow b$ stands for $a \to^* c$ and $c \, {}^*\!\leftarrow b$. Put differently, a and b are joinable if and only if they have a common reduct. The relation $\downarrow$ is called *joinability*. We write $a \uparrow b$ if a and b have a common *ancestor*, i.e., $a \, {}^*\!\leftarrow c \to^* b$ for some $c \in A$. The relation $\uparrow$ is

called *meetability*.

If $\to_1$ and $\to_2$ are binary relations on a set A then $\to_1 \cdot \to_2$ denotes their composition, i.e., $a \to_1 \cdot \to_2 b$ if and only if there exists a $c \in A$ such that $a \to_1 c$ and $c \to_2 b$. So $\downarrow$ simply abbreviates $\to^* \cdot {}^*\!\leftarrow$ and $\uparrow$ stands for ${}^*\!\leftarrow \cdot \to^*$.

Example 1.1.9. The ARS of Figure 1.1 admits infinite rewrite sequences, e.g.

$$\mathsf{a} \to \mathsf{e} \to \mathsf{b} \to \mathsf{a} \to \mathsf{e} \to \cdots$$

We have $\mathsf{d} \downarrow \mathsf{e}$. The elements d and g are not joinable but $\mathsf{d} \uparrow \mathsf{g}$ holds.

Definition 1.1.10. Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS. We write $a \leftrightarrow b$ if $a \to b$ or $a \leftarrow b$. A *conversion* between elements $a, b \in A$ consists of a non-empty sequence of elements $c_0, \ldots, c_n \in A$ $(n \geqslant 0)$ such that $a = c_0$, $b = c_n$, and $c_i \leftrightarrow c_{i+1}$ for all $0 \leqslant i < n$. The number n is the length of the conversion. We write $a \leftrightarrow^* b$ and we say that a and b are *convertible* if there exists a conversion between a and b. The relation $\leftrightarrow^*$ is called *conversion*. We write $a \leftrightarrow^m b$ if there exists a conversion between a and b of length m.

In diagrams we usually draw a conversion with arrows pointing downwards, as in Figure 1.2. It can be shown that $\leftrightarrow^*$ is the transitive, reflexive, and symmetric closure of $\to$. We have $\leftrightarrow^* = \bigcup \{ \leftrightarrow^n \mid n \geqslant 0 \}$.

Of special interest are the elements which cannot be reduced. They model the result of a computation.

Definition 1.1.11. Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS. An element $a \in A$ is *reducible* if there exists an element $b \in A$ with $a \to b$. A *normal form* is an element that is not reducible. The set of normal forms of $\mathcal{A}$ is denoted by $\mathsf{NF}(\mathcal{A})$ or $\mathsf{NF}(\to)$ when A can be inferred from the context. An element $a \in A$ *has a normal form* if $a \to^* b$ for some normal form b. In that case we write $a \to^! b$. A *normalizing* rewrite sequence is a (finite) rewrite sequence ending in a normal form.

Note that the relation $\to^!$ can be defined more succinctly as the intersection of $\to^*$ and $A \times \mathsf{NF}(\mathcal{A})$.

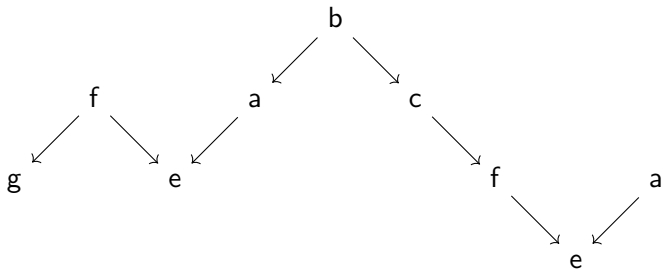


Figure 1.2: A conversion in the ARS of Figure 1.1.

Example 1.1.12. The ARS of Figure 1.1 has two normal forms: d and g. We have $\mathsf{a} \to^! \mathsf{d}$ but also $\mathsf{d} \to^! \mathsf{d}$. The normal forms d and g are convertible but neither $\mathsf{d} \to^! \mathsf{g}$ nor $\mathsf{g} \to^! \mathsf{d}$ holds.

Exercises

1.1 What are the normal forms of the ARS in Figure 1.1?

1.2 Compute the relations $\to^n$ $(n \geqslant 0)$, $\downarrow$, and $\uparrow$ for the ARS in Figure 1.1.

1.3 Construct ARSs $\mathcal{A} = \langle A, \to \rangle$ with non-empty A such that

a $\mathsf{NF}(\mathcal{A}) = A$

b $\mathsf{NF}(\mathcal{A}) = \varnothing$

1.4 Construct ARSs $\mathcal{A} = \langle A, \to \rangle$ with non-empty $\to$ such that

a $\to^+ = \to^*$

b $\leftrightarrow^* \neq \uparrow$

c $\to^* \cdot \leftrightarrow \neq \downarrow$

d $\to^2 \neq \to^3 \cdot \leftarrow$

e ${}^!\leftarrow \neq \leftarrow^!$

Here ${}^!\leftarrow$ denotes the inverse of $\to^!$, conforming to the notational convention stated on page 211.

1.5 Define the ARS $\mathcal{A} = \langle A, \to \rangle$ as follows: $A = \mathbb{N} \setminus \{0, 1\}$, the set of natural numbers greater than 1, and $a \to b$ if there exists an element $c \in A$ such that $a = b \times c$.

a Give all rewrite sequences starting at 12.

b Describe the set of normal forms of an element $a \in A$.

c What are the normal forms of $\mathcal{A}$?

d Show $a \leftrightarrow^* b$ for all elements $a, b \in A$. Which elements $a, b \in A$ satisfy $a \downarrow b$?

e Compute the number of rewrite sequences starting at 2^n, for all $n \geqslant 1$.

1.6 Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS.

a Show $\mathsf{NF}(\to^+) = \mathsf{NF}(\to)$. Here $\mathsf{NF}(\to^+)$ denotes the set of normal forms of the ARS $\langle A, \to^+ \rangle$.

b Let $\rightarrowtail \subseteq \to$. Show $\mathsf{NF}(\to) \subseteq \mathsf{NF}(\rightarrowtail)$.

c Give a simple expression for $\mathsf{NF}(\downarrow)$.

1.2 Properties

In this section we define various properties of ARSs. If an ARS $\mathcal{A} = \langle A, \to \rangle$ has a certain property $\mathcal{P}$, we also say the relation $\to$ has the property $\mathcal{P}$, and we write $\mathcal{P}(\mathcal{A})$, $\mathcal{P}(\to)$, or simply $\mathcal{P}$ when $\mathcal{A}$ can be inferred from the context. Most properties that we consider make also sense for individual elements of ARSs. When an element $a \in A$ of an ARS $\mathcal{A} = \langle A, \to \rangle$ satisfies $\mathcal{P}$, we write $\mathcal{P}(a)$.

Definition 1.2.1. Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS. An element $a \in A$ is called *terminating* or *strongly normalizing* (SN) if there are no infinite rewrite sequences starting at a. The ARS $\mathcal{A}$ is terminating or strongly normalizing if all its elements are terminating. An element $a \in A$ has *unique normal forms* (UN) if it does not have different normal forms (for all $b, c \in A$, if $a \to^! b$ and $a \to^! c$ then $b = c$). The ARS $\mathcal{A}$ has unique normal forms if all its elements have unique normal forms.

In this book we favor the terminology termination above strong normalization but we use the acronym SN as abbreviation.

Example 1.2.2. In the ARS of Figure 1.1 the element a is not terminating since we have the infinite rewrite sequence $\mathsf{a} \to \mathsf{e} \to \mathsf{b} \to \mathsf{a} \to \cdots$. The element c does not have unique normal forms as $\mathsf{c} \to^! \mathsf{d}$ and $\mathsf{c} \to^! \mathsf{g}$.

The usual way to show that an ARS has unique normal forms is to show the stronger property defined below.

Definition 1.2.3. Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS. An element $a \in A$ is *confluent* if for all elements $b, c \in A$ with $b \,{}^*\!\leftarrow a \to^* c$ we have $b \downarrow c$. The ARS $\mathcal{A}$ is confluent if all its elements are confluent.

Confluence is illustrated in Figure 1.3(i). In diagrams like the ones in Figure 1.3 dashed arrows are existentially quantified; they depend on the solid arrows. Confluence of an ARS $\langle A, \to \rangle$ can be expressed more succinctly by means of the inclusion $\uparrow \subseteq \downarrow$. By the same token we could have defined UN for ARSs by the inclusion ${}^!\!\leftarrow \cdot \to^! \subseteq =$. In the following we frequently employ such inclusions to define properties of ARSs.

Lemma 1.2.4. *Every confluent* ARS *has unique normal forms.*

Proof Let $\mathcal{A} = \langle A, \to \rangle$ be a confluent ARS. Consider rewrite sequences $a \to^! b$ and $a \to^! c$. Confluence yields $b \downarrow c$ and because b and c are normal forms we obtain $b = c$. Therefore $\mathcal{A}$ has unique normal forms. □

The next proposition states that confluence is equivalent to the property illustrated in Figure 1.3(ii), stating that every two convertible elements are joinable. This latter property is known as the *Church–Rosser* property (CR).

Lemma 1.2.5. *An* ARS $\mathcal{A} = \langle A, \to \rangle$ *is confluent if and only if* $\leftrightarrow^* \subseteq \downarrow$.

Proof

$\Longrightarrow$ By induction on n we show $\leftrightarrow^n \subseteq \downarrow$. If $n = 0$ then $\leftrightarrow^n$ is the identity on A, and clearly $a \downarrow a$ for every $a \in A$. Suppose $n > 0$ and let $a \leftrightarrow a' \leftrightarrow^{n-1} b$. From the induction hypothesis we obtain $a' \downarrow b$. If $a \to a'$ then clearly $a \downarrow b$. Assume $a \leftarrow a'$. Because $a' \downarrow b$ there exists a common reduct c of a' and b. We have $a \leftarrow a' \to^* c$. Confluence yields $a \downarrow c$ and since $b \to^* c$ we also have $a \downarrow b$.

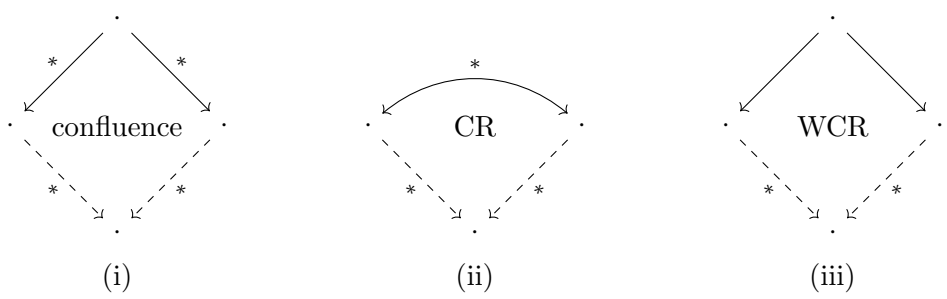


Figure 1.3: Properties related to confluence.

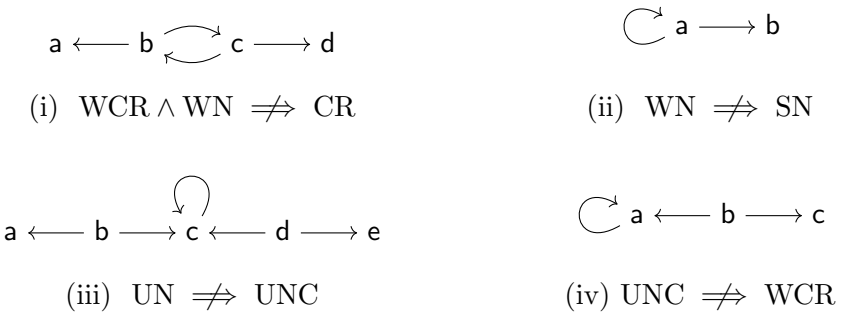


Figure 1.4: Four abstract counterexamples.

$\Longleftarrow$ Combining the assumption $\leftrightarrow^* \subseteq \downarrow$ with the obvious $\uparrow \subseteq \leftrightarrow^*$ yields $\uparrow \subseteq \downarrow$. □

Since we clearly have $\downarrow \subseteq \leftrightarrow^*$, the above proposition states that in a confluent ARS conversion and joinability coincide. This observation will be used freely in the sequel. We will also use confluence and CR interchangeably, except in Chapter 12 where we study rewriting modulo equivalence relations. In particular, we write $\mathrm{CR}(a)$ to denote that the element a is confluent.

ARSs that are not confluent do not satisfy $\leftrightarrow^* \subseteq \downarrow$ but the weaker inclusion given in the following lemma always holds. It states the easy but useful fact that every conversion that is not of the form $\rightarrow^* \cdot {}^*\leftarrow$, a so-called *valley*, contains a *local peak* $\leftarrow \cdot \rightarrow$. The proof is left to the reader (Exercise 1.12).

Lemma 1.2.6. *For every* ARS $\langle A, \rightarrow \rangle$ *the inclusion* $\leftrightarrow^* \subseteq \downarrow \cup \leftrightarrow^* \cdot \leftarrow \cdot \rightarrow \cdot \leftrightarrow^*$ *holds.*

In order to show that an ARS $\mathcal{A} = \langle A, \rightarrow \rangle$ is confluent, we have to consider all diverging situations $b \,{}^*\!\leftarrow a \rightarrow^* c$. Matters would be considerably easier if we only had to consider local peaks.

Definition 1.2.7. Let $\mathcal{A} = \langle A, \rightarrow \rangle$ be an ARS. An element $a \in A$ is *locally confluent* if for all elements $b, c \in A$ with $b \leftarrow a \rightarrow c$ we have $b \downarrow c$. The ARS $\mathcal{A}$ is locally confluent if all its elements are locally confluent. Local confluence will also be referred to as the *weak Church–Rosser* property (WCR).

Local confluence is illustrated in Figure 1.3(iii). Confluent ARSs are obviously locally confluent, but the converse does not hold in general. The ARS of Figure 1.4(i) constitutes a simple counterexample. However, in the presence of termination, local confluence implies confluence. This important result, known as Newman's Lemma, will be proved in the next section.

In the following definition three further properties are introduced.

Definition 1.2.8. Let $\mathcal{A} = \langle A, \rightarrow \rangle$ be an ARS. An element $a \in A$ is called *weakly normalizing* (WN) or simply *normalizing* if it has a normal form. An element $a \in A$ is called *complete* if it is both terminating and confluent. An element $a \in A$ is said to be *semi-complete* if it is both normalizing and confluent. The ARS $\mathcal{A}$ is normalizing (complete, semi-complete) if all its elements are normalizing (complete, semi-complete).

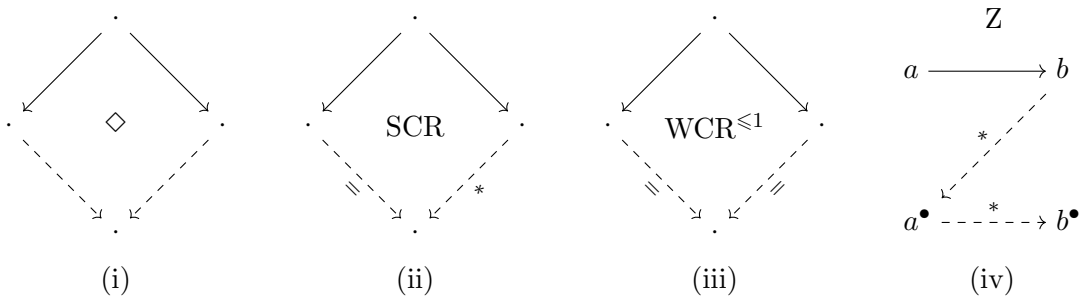


Figure 1.5: Some properties of ARSs.

Every terminating ARS is normalizing, but the converse does not hold in general. The ARS of Figure 1.4(ii) constitutes the simplest counterexample against the implication WN $\implies$ SN. (An implication $P \implies Q$ between properties P and Q of ARSs holds if every ARS with the property P also has the property Q.)

If an element $a \in A$ in an ARS $\langle A, \to \rangle$ has exactly one normal form then we write $a{\downarrow}$ to denote this normal form. Semi-completeness is a sufficient condition for $a{\downarrow}$ to exist for all elements $a \in A$.

Lemma 1.2.9. *Let $\mathcal{A} = \langle A, \to \rangle$ be a semi-complete* ARS*. Two elements $a, b \in A$ are convertible if and only if $a{\downarrow} = b{\downarrow}$.*

Proof If $a{\downarrow} = b{\downarrow}$ then $a \to^! \cdot \,{}^!\!\leftarrow b$ and thus $a \leftrightarrow^* b$. Conversely, if $a \leftrightarrow^* b$ then $a \downarrow b$ by Lemma 1.2.5. Let c be a common reduct of a and b. Because $\mathcal{A}$ is normalizing, $c \to^! d$ for some normal form d. Hence $a \to^! d$ and $b \to^! d$ and thus $a{\downarrow} = d = b{\downarrow}$. □

The property we introduce next is a variation of unique normal forms.

Definition 1.2.10. An ARS $\mathcal{A} = \langle A, \to \rangle$ has *unique normal forms with respect to conversion* (UNC) if different normal forms are not convertible (for all $a, b \in \mathsf{NF}(\mathcal{A})$, if $a \leftrightarrow^* b$ then $a = b$).

So in an ARS with the property UNC every equivalence class of convertible elements contains at most one normal form. Clearly UNC implies UN. Somewhat surprisingly, UN and UNC are not equivalent, as witnessed by the ARS of Figure 1.4(iii). In this ARS the normal forms a and e are convertible, notwithstanding the fact that UN holds.

Using Lemma 1.2.5, it is easy to show that CR implies UNC. The ARS of Figure 1.4(iv) shows that the converse does not hold. Observe that this ARS is not normalizing. Under the additional assumption of normalization, UNC does imply CR. Slightly stronger, we have the following result.

Lemma 1.2.11. *Every normalizing* ARS *with unique normal forms is confluent.*

Proof Let $\mathcal{A} = \langle A, \to \rangle$ be a normalizing ARS with unique normal forms. Consider rewrite sequences $a \to^* b$ and $a \to^* c$. Let b' and c' be normal forms of b and c, respectively. We have $a \to^! b'$ and $a \to^! c'$. Because $\mathcal{A}$ has unique normal forms we obtain $b' = c'$ and hence b and c have a common reduct. So $\mathcal{A}$ is confluent. □

The next result of this section gives a sufficient condition for confluence that does not rely on any normalization assumption. First a definition.

Definition 1.2.12. An ARS $\mathcal{A} = \langle A, \to \rangle$ has the *diamond property* ($\diamond$) if $\leftarrow \cdot \to \subseteq \to \cdot \leftarrow$, see Figure 1.5(i).

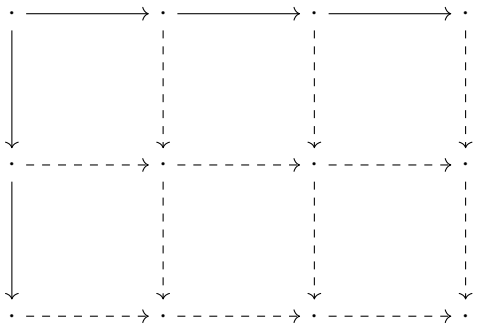
Lemma 1.2.13. *Let $\mathcal{A} = \langle A, \to \rangle$ be an* ARS.

1. *If $\mathcal{A}$ has the diamond property then $\mathcal{A}$ is confluent.*
2. *If there exists a relation $\rightarrowtail$ on A such that $\rightarrowtail^* = \to^*$ then $\mathcal{A}$ is confluent if and only if $\langle A, \rightarrowtail \rangle$ is confluent.*

The condition $\rightarrowtail^* = \to^*$ in part 2 of Lemma 1.2.13 is typically established by means of the inclusions $\to \subseteq \rightarrowtail \subseteq \to^*$.

Proof

1. An easy induction proof (Exercise 1.14(a)) shows ${}^m\!\leftarrow \cdot \to^n \subseteq \to^n \cdot {}^m\!\leftarrow$ for all $m, n \geqslant 0$. For example, for $m = 2$ and $n = 3$:



Hence $\mathcal{A}$ is confluent.

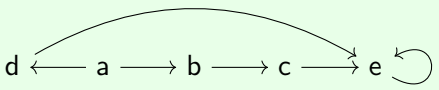
2. From $\rightarrowtail^* = \to^*$ we obtain both $\upharpoonright = \uparrow$ and $\downharpoonright = \downarrow$. Hence $\upharpoonright \subseteq \downharpoonright$ if and only if $\uparrow \subseteq \downarrow$. □

We present a further sufficient condition for confluence. This one depends on the existence of a suitable function on the set of elements.

Definition 1.2.14. An ARS $\mathcal{A} = \langle A, \to \rangle$ has the *Z-property* if there exists a function $\bullet$ on A such that both $b \to^* \bullet(a)$ and $\bullet(a) \to^* \bullet(b)$ whenever $a \to b$, see Figure 1.5(iv). The function $\bullet$ is called *bullet function*.

We say that $\mathcal{A}$ has the Z-property *for* $\bullet$ if we want to indicate the bullet function that satisfies the properties of Definition 1.2.14. We often write $a^\bullet$ for $\bullet(a)$.

Example 1.2.15. Consider the ARS



Define the function $\bullet$ as follows: $\mathsf{a}^\bullet = \mathsf{b}^\bullet = \mathsf{c}^\bullet = \mathsf{d}^\bullet = \mathsf{e}^\bullet = \mathsf{e}$. Since every element of the ARS rewrites to e, the Z-property is trivially satisfied.

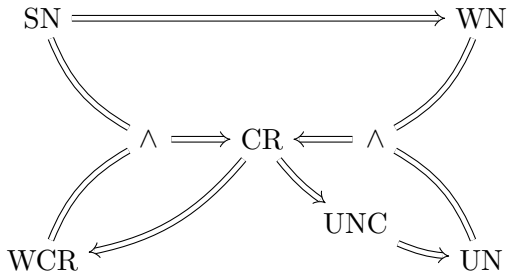


Figure 1.6: The relationships between six basic properties of ARSs.

We now show that the Z-property implies confluence.

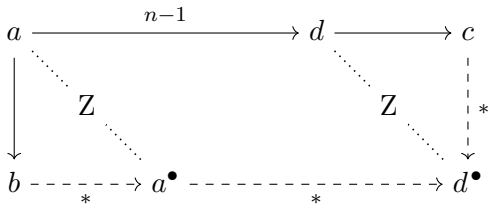
Lemma 1.2.16. *Let $\mathcal{A} = \langle A, \to \rangle$ be an* ARS *with the Z-property for $\bullet$. If $a \to^* b$ then $a^\bullet \to^* b^\bullet$.*

Proof Straightforward induction on the number of rewrite steps in $a \to^* b$. □

In concrete applications of the following result, the challenge is to find a suitable bullet function.

Lemma 1.2.17. *Every* ARS *with the* Z*-property is confluent.*

Proof Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS with the Z-property for $\bullet$. We prove that $\mathcal{A}$ satisfies the inclusion $\leftarrow \cdot \to^* \subseteq \downarrow$. According to the result of Exercise 1.17(c) this ensures confluence. So let $b \leftarrow a \to^n c$. We prove $b \downarrow c$ by induction on n. If $n = 0$ then $c = a \to b$. Suppose $n > 0$ and let d be an element such that $a \to^{n-1} d \to c$. The Z-property yields $b \to^* a^\bullet$ and $c \to^* d^\bullet$. Moreover, $a^\bullet \to^* d^\bullet$ follows from $a \to^{n-1} d$ by Lemma 1.2.16. Hence $d^\bullet$ is a common reduct of b and c. The diagram



summarizes the reasoning in the induction step. □

The relationships between the properties termination, normalization, confluence, local confluence, and the two unique normal form properties are visualized in Figure 1.6. We stress that these relationships hold for ARSs as a whole, but not necessarily for individual elements (cf. Exercise 1.13). In the exercises of this and the next section several other properties of ARSs are introduced. Some of them are illustrated in Figure 1.5.

We conclude this section with two useful notions of equivalence. They play an important role in the theory of completion (Section 5.3).

Definition 1.2.18. Two ARSs $\mathcal{A}$ and $\mathcal{B}$ are *(conversion) equivalent* if $\leftrightarrow^*_{\mathcal{A}} = \leftrightarrow^*_{\mathcal{B}}$ and *normalization equivalent* if $\to^!_{\mathcal{A}} = \to^!_{\mathcal{B}}$.

Example 1.2.19. The ARSs

$\mathcal{A}_1$: a ⟶ b $\qquad$ $\mathcal{B}_1$: a ⟵ b

are conversion equivalent but not normalization equivalent. The ARSs

$\mathcal{A}_2$: a ⟶ b ↻ $\qquad$ $\mathcal{B}_2$: ↻ a b ↻

are normalization equivalent but not conversion equivalent.

The easy proof of the following result is left to the reader (Exercise 1.23).

Lemma 1.2.20. *Normalization equivalent terminating* ARS*s are equivalent.*

We conclude this section with a sufficient condition for normalization equivalence.

Lemma 1.2.21. *Let $\mathcal{A}$ and $\mathcal{B}$ be* ARS*s over the same domain such that $\to_{\mathcal{B}} \subseteq \to_{\mathcal{A}}^+$ and $\mathsf{NF}(\mathcal{B}) \subseteq \mathsf{NF}(\mathcal{A})$. If $\mathcal{A}$ is complete then $\mathcal{B}$ is complete and normalization equivalent to $\mathcal{A}$.*

Proof From the inclusion $\to_{\mathcal{B}} \subseteq \to_{\mathcal{A}}^+$ we infer that $\mathcal{B}$ is terminating. Moreover, $\to_{\mathcal{B}}^* \subseteq \to_{\mathcal{A}}^*$ and, since $\mathsf{NF}(\mathcal{B}) \subseteq \mathsf{NF}(\mathcal{A})$, also $\to_{\mathcal{B}}^! \subseteq \to_{\mathcal{A}}^!$. For the reverse inclusion we reason as follows. Let $a \to_{\mathcal{A}}^! b$. Because $\mathcal{B}$ is terminating, $a \to_{\mathcal{B}}^! c$ for some $c \in \mathsf{NF}(\mathcal{B})$. So $a \to_{\mathcal{A}}^! c$ and thus $b = c$ from the confluence of $\mathcal{A}$. It follows that $\mathcal{A}$ and $\mathcal{B}$ are normalization equivalent. It remains to show that $\mathcal{B}$ is locally confluent. This follows from the sequence of inclusions

$$ {}_{\mathcal{B}}\!\leftarrow \cdot \to_{\mathcal{B}} \;\subseteq\; {}_{\mathcal{A}}^{+}\!\leftarrow \cdot \to_{\mathcal{A}}^{+} \;\subseteq\; \to_{\mathcal{A}}^{!} \cdot {}_{\mathcal{A}}^{!}\!\leftarrow \;\subseteq\; \to_{\mathcal{B}}^{!} \cdot {}_{\mathcal{B}}^{!}\!\leftarrow $$

where we use the inclusion $\to_{\mathcal{B}} \subseteq \to_{\mathcal{A}}^+$, the confluence of $\mathcal{A}$, the termination of $\mathcal{A}$, and the normalization equivalence of $\mathcal{A}$ and $\mathcal{B}$. □

Exercises

1.7 Consider the ARS of Figure 1.1.

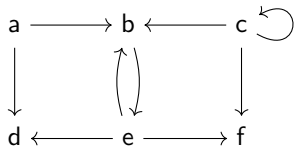
a Complete the following table:

	SN	WN	CR	WCR	UN
a				✓	
b	×				
c					
d					
e					×
f					
g		✓			

b Which arrow in Figure 1.1 has to be eliminated in order to make the ARS terminating?

c Add a single arrow to Figure 1.1 such that the resulting ARS has unique normal forms without being confluent.

1.8 Construct a table similar to the one in Exercise 1.7(a) for the following ARS:



1.9 Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS.

a Show that every normal form of $\mathcal{A}$ is complete.

b Show the equivalence of the following statements:

- ▷ $\mathcal{A}$ is semi-complete
- ▷ every element of A has a unique normal form
- ▷ $\leftrightarrow^* = \to^! \cdot {}^!\leftarrow$

c Show that $\mathcal{A}$ is normalizing if and only if $\mathsf{NF}(\to^!) = \varnothing$.

1.10 Consider the ARS $\mathcal{A}$ of Exercise 1.5.

a Show that $\mathcal{A}$ is terminating.

b Does $\mathcal{A}$ have unique normal forms?

c Describe the confluent elements of $\mathcal{A}$.

1.11 Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS.

a Suppose $a \in A$ is terminating. Show that the reducts of a are terminating.

b Which of the properties CR, UN, WCR, WN, completeness, and semi-completeness satisfy the behavior expressed in (a)?

c Repeat parts (a) and (b) for ancestors instead of reducts.

1.12 Prove Lemma 1.2.6.

1.13 Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS and let a be an arbitrary element of A.

a Which of the following implications hold?

- ▷ $\mathrm{SN}(a) \implies \mathrm{WN}(a)$
- ▷ $\mathrm{CR}(a) \implies \mathrm{WCR}(a)$
- ▷ $\mathrm{CR}(a) \implies \mathrm{UN}(a)$

b Show that the implication $\mathrm{WN}(a) \wedge \mathrm{UN}(a) \implies \mathrm{CR}(a)$ does not hold in general.

c Show the implication $\mathrm{WN}(\mathcal{A}) \wedge \mathrm{UN}(a) \implies \mathrm{CR}(a)$.

d Does the implication $\mathrm{WN}(a) \wedge \mathrm{UN}(\mathcal{A}) \implies \mathrm{CR}(a)$ hold?

1.14 ***a*** Complete the proof of Lemma 1.2.13 **1**.

b Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS with the diamond property. Show $\mathrm{SN}(a)$ if and only if $a \in \mathsf{NF}(\mathcal{A})$, for all $a \in A$. Conclude that $\mathcal{A}$ is SN if and only if $\to = \varnothing$.

1.15 Consider the ARS $\mathcal{A} = \langle A, \to \rangle$ with $A = \mathbb{N}_+ \cup (\mathbb{N}_+ \times \mathbb{N}_+)$ and

$$\begin{array}{ll} (x,x) \to x & \text{for all } x \in \mathbb{N}_+ \\ (x,y) \to (x-y,y) & \text{for all } x,y \in \mathbb{N}_+ \text{ with } x > y \\ (x,y) \to (y,x) & \text{for all } x,y \in \mathbb{N}_+ \text{ with } x \leqslant y \end{array}$$

Here $\mathbb{N}_+$ denotes the set of positive integers.

a Compute a normal form of the element $(12, 18)$.

b Is $\mathcal{A}$ normalizing?

c Is $\mathcal{A}$ terminating?

d Is $\mathcal{A}$ confluent?

1.16 An ARS $\mathcal{A} = \langle A, \to \rangle$ has the *normal form property* (NFP) if all normalizing elements are confluent ($\forall\, a \in A$ if $\mathrm{WN}(a)$ then $\mathrm{CR}(a)$).

a Show that every confluent ARS has the normal form property.

b Does the converse also hold?

c Show the equivalence of the following statements for every ARS $\mathcal{A} = \langle A, \to \rangle$:

- ▷ $\mathcal{A}$ has the normal form property
- ▷ $\leftarrow \cdot \to^! \subseteq \to^!$
- ▷ every element convertible to a normal form rewrites to that normal form

d Establish the relationship between NFP, CR, UN, and UNC.

1.17 An ARS $\mathcal{A} = \langle A, \to \rangle$ is *strongly confluent* (SCR) if $\leftarrow \cdot \to \subseteq \to^= \cdot\, {}^*\!\leftarrow$, see Figure 1.5(ii).

a Show that every strongly confluent ARS is confluent.

b Does the converse also hold?

c Show that an ARS $\mathcal{A} = \langle A, \to \rangle$ is confluent if and only if $\leftarrow \cdot \to^* \subseteq \to^* \cdot\, {}^*\!\leftarrow$.

1.18 Consider the ARS $\mathcal{A} = \langle A, \to \rangle$ with $A = \mathbb{N}_+$ and $x \to y$ if $x > 1$ and

$$y = \begin{cases} \frac{x}{2} & \text{if } x \text{ is even} \\ 3x + 1 & \text{if } x \text{ is odd} \end{cases}$$

a Compute a normal form of 23.

b Is $\mathcal{A}$ confluent?

c Is $\mathcal{A}$ terminating?

1.19 Let $\mathcal{A} = \langle A, \to \rangle$ and $\mathcal{B} = \langle A, \rightarrowtail \rangle$ be ARSs such that $\to \subseteq \rightarrowtail^* \subseteq \downarrow$.

a Show that $\mathcal{B}$ is confluent whenever $\mathcal{A}$ is confluent.

b Show that $\mathcal{A}$ is confluent whenever $\mathcal{B}$ is confluent.

c Is the result of part (b) still true if we replace the assumption $\to \subseteq \rightarrowtail^* \subseteq \downarrow$ by $\to \subseteq \rightarrowtail^* \subseteq \leftrightarrow^*$?

1.20 ***a*** One of the ARSs $\mathcal{A} = \langle \mathbb{N}, < \rangle$ and $\mathcal{B} = \langle \mathbb{N}, <_1 \rangle$ has the Z-property. Which one? Give a proof and counterexample. Here $<_1$ is the predecessor relation on $\mathbb{N}$ (cf. Example A.1.7).

b Show that the bullet function in Example 1.2.15 is the only one for which the Z-property holds.

c Construct an ARS that has the Z-property for different bullet functions.

d Does the converse of Lemma 1.2.17 hold?

1.21 ***a*** Let $\mathcal{A}$ be an ARS with the Z-property for $\bullet$. Does $a^\bullet = a$ hold for every normal form $a \in \mathsf{NF}(\mathcal{A})$?

b Show that a terminating ARS $\mathcal{A}$ has the Z-property if and only if $\mathcal{A}$ is locally confluent.

c An ARS $\mathcal{A} = \langle A, \to \rangle$ has the *angle property* if there a function $\bullet$ on A such that $b \to a^\bullet$ whenever $a \to b$. Show that $\mathcal{A}$ has the Z-property if and only if there exists a relation $\rightarrowtail$ on A such that $\to \subseteq \rightarrowtail \subseteq \to^*$ and $\langle A, \rightarrowtail \rangle$ has the angle property.

1.22 Consider ARSs $\mathcal{A} = \langle A, \to_\alpha \rangle$ and $\mathcal{B} = \langle B, \to_\beta \rangle$ such that $\mathcal{A}$ is normalizing and $\mathcal{B}$ is confluent. Let ϕ be a mapping from A to B with the following properties:

- ▷ if $a \to_\alpha a'$ then $\phi(a) \leftrightarrow_\beta^* \phi(a')$,
- ▷ if $a \in \mathsf{NF}(\mathcal{A})$ then $\phi(a) \in \mathsf{NF}(\mathcal{B})$,
- ▷ ϕ is injective on $\mathsf{NF}(\mathcal{A})$.

Show that $\mathcal{A}$ is confluent.

1.23 ***a*** Prove Lemma 1.2.20.

b Let $\mathcal{A}$ and $\mathcal{B}$ be normalization equivalent ARSs. Which of the following properties guarantee that $\mathcal{A}$ and $\mathcal{B}$ are (conversion) equivalent?

- ▷ $\mathcal{A}$ and $\mathcal{B}$ are confluent
- ▷ $\mathcal{A}$ and $\mathcal{B}$ are normalizing

c Show the necessity of the condition that $\mathcal{A}$ and $\mathcal{B}$ have the same domain in Lemma 1.2.21.

d Does Lemma 1.2.21 remain true if we replace completeness by semi-completeness?

1.3 Newman's Lemma

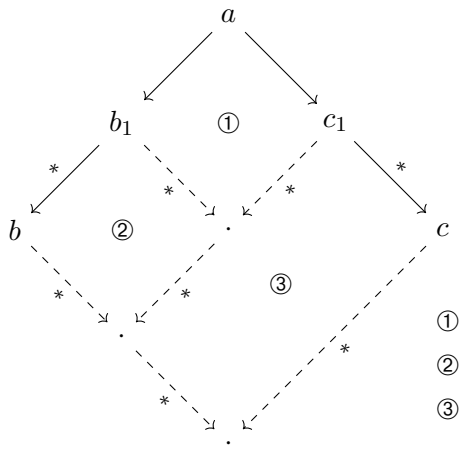
In this section we present two proofs of the following result.

Newman's Lemma. *Every terminating and locally confluent* ARS *is confluent.*

Newman's Lemma forms the theoretical basis for the completion procedure of Knuth and Bendix, to be explained in Chapter 5. The first proof illustrates the useful technique of *well-founded induction* (cf. Section A.2). The second proof is by contradiction.

The proof technique of well-founded induction states that a property $\mathcal{P}$ of elements of a terminating ARS $\mathcal{A} = \langle A, \to \rangle$ holds for all elements in A if the following condition is satisfied: An element $a \in A$ has the property $\mathcal{P}$ if all elements b with $a \to b$ have the property $\mathcal{P}$. In particular every normal form has to satisfy the property $\mathcal{P}$.

First Proof of Newman's Lemma Let $\mathcal{A} = \langle A, \to \rangle$ be a terminating and locally confluent ARS. We will show that every element of A is confluent by means of well-founded induction. Let a be an arbitrary element of A. The induction hypothesis states that all one-step reducts of a are confluent. We have to show that a itself is confluent. So consider two rewrite sequences $a \to^* b$ and $a \to^* c$. If one of these sequences is empty then b or c is a common reduct of b and c. So we may assume that both sequences consist of at least one rewrite step. Drawing the diagram



① WCR

② induction hypothesis ($a \to b_1 \implies b_1$ is CR)

③ induction hypothesis ($a \to c_1 \implies c_1$ is CR)

completes the proof. □

Our second proof uses a minimal counterexample argument and is illustrated in Figure 1.7.

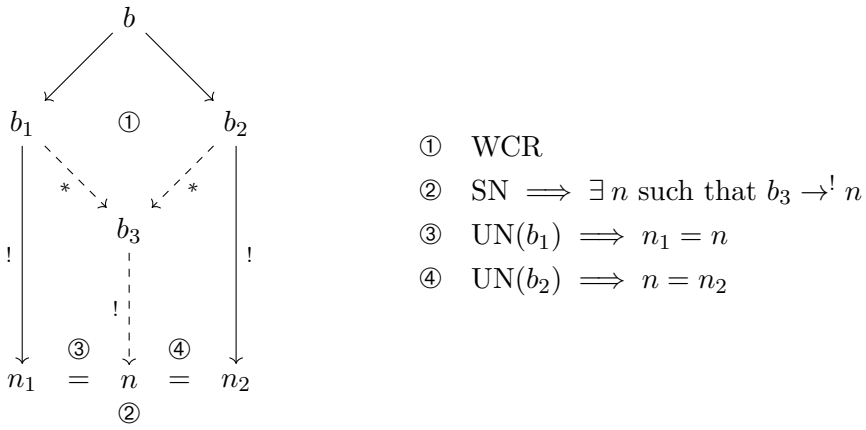


Figure 1.7: Second proof of Newman's Lemma.

Second Proof of Newman's Lemma Let $\mathcal{A} = \langle A, \to \rangle$ be a terminating and locally confluent ARS. According to Lemma 1.2.11 it suffices to show that every element has unique normal forms. For a proof by contradiction, suppose the set $B = \{a \in A \mid \neg \mathrm{UN}(a)\}$ is non-empty. Because $\mathcal{A}$ is terminating, the relation $\to$ is well-founded. Hence B contains a minimal element with respect to $\to$ (cf. Theorem A.2.3), say b. By definition of B, $b \to^! n_1$ and $b \to^! n_2$ with $n_1 \neq n_2$. Clearly these sequences contain at least one step, so we may write $n_1 \,{}^!\!\leftarrow b_1 \leftarrow b \to b_2 \to^! n_2$. Local confluence yields a common reduct of b_1 and b_2, say $b_1 \to^* b_3 \,{}^*\!\leftarrow b_2$. Termination guarantees the existence of a normal form n of b_3. We have $b_1 \to^! n \,{}^!\!\leftarrow b_2$. Because b is a minimal element in B, $b_1, b_2 \notin B$. So b_1 and b_2 have unique normal forms. This implies $n_1 = n = n_2$, contradicting the assumption that n_1 and n_2 are different normal forms. □

In the next lemma we present a slight generalization of Newman's Lemma.

Lemma 1.3.1. *Every terminating element in a locally confluent* ARS *is confluent.*

Proof Let $\mathcal{A} = \langle A, \to \rangle$ be a locally confluent ARS and let $a \in A$ be terminating. Let $B = \{b \mid a \to^* b\}$ be the set of all reducts of a. Let $\rightarrowtail$ be the restriction of $\to$ to $B \times B$. One easily verifies that the ARS $\mathcal{B} = \langle B, \rightarrowtail \rangle$ is terminating and locally confluent. According to Newman's Lemma $\mathcal{B}$ is confluent. Because $a \in B$, we conclude that a is confluent (in $\mathcal{A}$). □

Exercises

1.24 Can we strengthen Lemma 1.3.1 further by localizing also local confluence, i.e., is it true that every terminating and locally confluent element in an arbitrary ARS is confluent?

1.25 An ARS $\mathcal{A} = \langle A, \to \rangle$ is *finitely branching* (FB) if for every element $a \in A$ the set $\{b \mid a \to b\}$ of its one-step reducts is finite. We call $\mathcal{A}$ *bounded* if for every element $a \in A$ there exists a natural number n such that the length of every rewrite sequence starting at a is at most n.

a Show that every bounded ARS is terminating.

b Show by well-founded induction that every finitely branching and terminating ARS is bounded.

c Is every terminating ARS bounded?

1.26 An ARS $\mathcal{A} = \langle A, \to \rangle$ is *cyclic* if there exists a cycle $a \to^+ a$ for some $a \in A$.

a Let $\mathcal{A} = \langle A, \to \rangle$ be an acyclic ARS with the property that for every element $a \in A$ the set $\{b \mid a \to^* b\}$ of its reducts is finite. Show that $\mathcal{A}$ is terminating.

b Prove the implication UN $\implies$ UNC for finite acyclic ARSs. (An ARS $\langle A, \to \rangle$ is called finite if A is finite.)

c Is finiteness essential in part (b)?

1.27 Given an ARS $\mathcal{A} = \langle A, \to \rangle$ and a well-founded order $>$ on A, we say that b is *connected* to c *below* a if there exists a conversion $b = a_0 \leftrightarrow \cdots \leftrightarrow a_n = c$ with $n \geqslant 0$ such that $a > a_i$ for all $0 \leqslant i \leqslant n$. We call $\mathcal{A}$ *connected* (with respect to $>$) if b and c are connected below a whenever $b \leftarrow a \to c$, for all $a, b, c \in A$.

a Show that every connected ARS is complete.

b Does the converse also hold?

1.28 An ARS $\mathcal{A} = \langle A, \to \rangle$ has the property $\mathrm{WCR}^{\leqslant 1}$ if $\leftarrow \cdot \to \subseteq \to^{=} \cdot {}^{=}\!\leftarrow$, see Figure 1.5(iii). We say that $\mathcal{A}$ has the property WCR^1 if $\leftarrow \cdot \to \subseteq \to \cdot \leftarrow \cup =$, i.e., for all elements $a, b, c \in A$ with $b \leftarrow a \to c$ and $b \neq c$ there exists a $d \in A$ such that $b \to d \leftarrow c$.

a What is the relationship between WCR^1 and $\diamond$?

b Construct a terminating ARS that is $\mathrm{WCR}^{\leqslant 1}$ but not WCR^1.

c Show that every ARS with the property $\mathrm{WCR}^{\leqslant 1}$ is confluent.

d Prove that every normalizing ARS with the property WCR^1 is terminating.

e Does the implication $\mathrm{WN} \wedge \mathrm{WCR}^{\leqslant 1} \implies \mathrm{SN}$ hold for every ARS?

1.29 An ARS $\mathcal{A} = \langle A, \to \rangle$ is *inductive* (IND) if for every infinite rewrite sequence $a_1 \to a_2 \to \cdots$ there exists an element $a \in A$ such that $a_i \to^* a$ for every $i \geqslant 1$.

a Show that every ARS $\mathcal{A} = \langle A, \to \rangle$ with finite A is inductive.

b Prove that every semi-complete ARS is inductive.

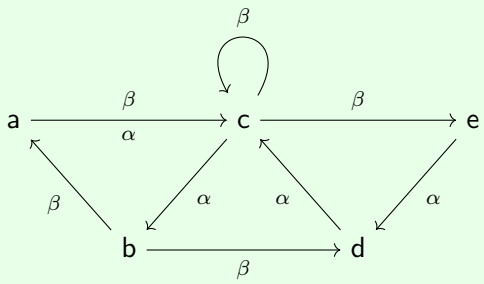
c Construct a confluent ARS that is not inductive.

d Construct a locally confluent and normalizing ARS that is not inductive.

1.4 Commutation

Termination and confluence are important properties of ARSs. In later chapters we develop a great many techniques for proving these properties in the concrete setting of term rewriting. In this section we present a few abstract results which are based on decomposition. We show how confluence and termination of an ARS $\mathcal{A} = \langle A, \to \rangle$, whose rewrite relation $\to$ is divided into smaller relations $\to_\alpha$ with α belonging to some index set I, can be inferred from properties of the individual relations $\to_\alpha$, by making suitable assumptions about the way these relations interact. In Chapter 6 many of the confluence results presented here are generalized, using the important *decreasing diagrams* technique.

Example 1.4.1. We illustrate the concepts defined in this section on the following ARS $\mathcal{A} = \langle A, \to \rangle$, where $\to$ is divided in the two (non-disjoint) relations $\to_\alpha$ and $\to_\beta$:



Let us first fix some notation. We write $\langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ to denote the ARS $\langle A, \to\rangle$ where $\to = \bigcup\{\to_\alpha \mid \alpha \in I\}$, so $\langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ can be viewed as a *presentation* of $\langle A, \to\rangle$. In the sequel we use the phrase "let $\mathcal{A} = \langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ be an ARS" as a shorthand for "let $\langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ be a presentation of the ARS $\mathcal{A}$". In order to reduce the number of arrows, we denote the individual relations $\to_\alpha$ of an ARS $\langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ simply by α and we denote ${}_\alpha\!\leftarrow$ by α^-. The next definition expresses a simple kind of interaction between relations α and β.

Definition 1.4.2. Let $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ be an ARS. The relations α and β *commute* if $(\alpha^-)^*\beta^* \subseteq \beta^*(\alpha^-)^*$, see Figure 1.8(i).

Observe that an ARS $\mathcal{A} = \langle A, \to\rangle$ is confluent if and only if its rewrite relation $\to$ is *self*-commuting.

Example 1.4.3. Consider the ARS $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ of Example 1.4.1. One easily checks that α and β commute. For instance, we have $\mathsf{b} \,{}^*_\alpha\!\leftarrow \mathsf{a} \to^*_\beta \mathsf{e}$ but also $\mathsf{b} \to^*_\beta \mathsf{e}$.

The following elementary result is sketched in Figure 1.9. By employing the diamond

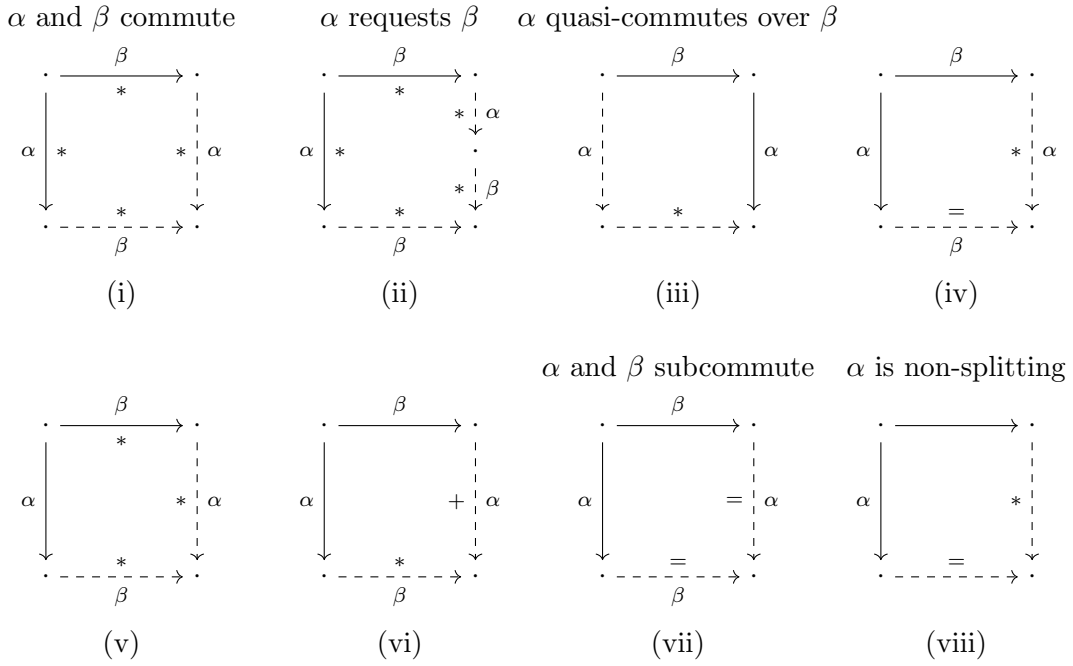


Figure 1.8: Commutation properties.

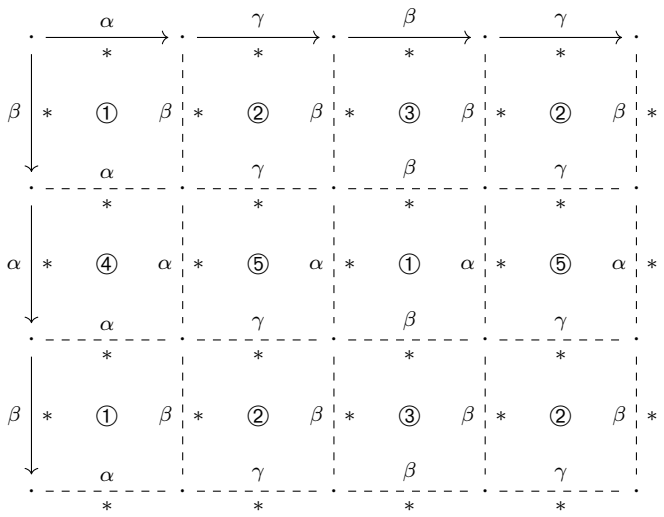


Figure 1.9: Commutation implies confluence.

property we can avoid well-founded induction in the formal proof.

Lemma 1.4.4. *Let $\mathcal{A} = \langle A, \{\alpha\}_{\alpha \in I} \rangle$ be an* ARS. *If α and β commute for all $\alpha, \beta \in I$ then $\mathcal{A}$ is confluent.*

Proof The relation $\rightarrowtail = \bigcup \{ \rightarrow_\alpha^* \mid \alpha \in I \}$ obviously satisfies $\rightarrow \subseteq \rightarrowtail \subseteq \rightarrow^*$. It has the diamond property by the commutation assumption. Hence Lemma 1.2.13 applies. □

Sufficient conditions for commutation are given in Exercise 1.30.

Example 1.4.5. Consider the ARS $\mathcal{A} = \langle A, \{\alpha, \beta\} \rangle$ of Example 1.4.1. Lemma 1.4.4 cannot be used to infer the confluence of $\mathcal{A}$, notwithstanding the fact that α and β commute. The reason is that β does not commute with itself: we have $\mathsf{d} \ {}_\beta\!\leftarrow \mathsf{b} \rightarrow_\beta^* \mathsf{e}$ but d and e are different β-normal forms.

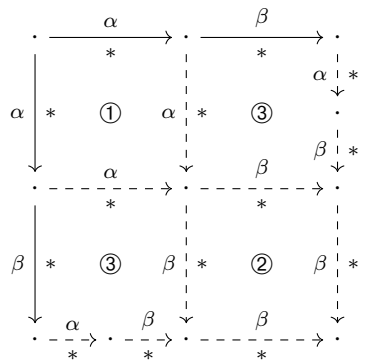
Commutation is a rather strong condition. The following lemma shows that a weaker condition suffices for confluence.

Definition 1.4.6. Let $\mathcal{A} = \langle A, \{\alpha, \beta\} \rangle$ be an ARS. We say that α *requests* β if $(\alpha^-)^* \beta^* \subseteq \beta^* (\beta^-)^* (\alpha^-)^*$, see Figure 1.8(ii).

Clearly, if α and β commute then α requests β and β requests α, but the converse is not true in general.

Lemma 1.4.7. *Let $\mathcal{A} = \langle A, \{\alpha, \beta\} \rangle$ be an* ARS *such that α and β are confluent. If α requests β then $\mathcal{A}$ is confluent.*

Proof The diagram



① α is confluent
② β is confluent
③ α requests β

shows that the relation $\alpha^*\beta^*$ has the diamond property. Since $\alpha \cup \beta \subseteq \alpha^*\beta^* \subseteq (\alpha \cup \beta)^*$, confluence of $\mathcal{A}$ follows from Lemma 1.2.13. □

We present another confluence result. This easy result will be used in the correctness proof of completion in Section 5.4.

Definition 1.4.8. An ARS $\mathcal{A} = \langle A, \{\alpha\}_{\alpha \in I} \rangle$ is *peak decreasing* if there exists a well-founded order $>$ on I such that for all $\alpha, \beta \in I$ the inclusion

$$\alpha\!\leftarrow \cdot \rightarrow_\beta \;\subseteq\; \xleftarrow[\vee\alpha\beta]{*}\!\rightarrow$$

holds. Here $\vee\alpha\beta$ denotes the set $\{\gamma \in I \mid \alpha > \gamma \text{ or } \beta > \gamma\}$ and if $J \subseteq I$ then $\rightarrow_J$ denotes the union of $\rightarrow_\gamma$ with $\gamma \in J$. We write $\vee\alpha$ for $\vee\alpha\alpha$.

In the proof of the following lemma we make use of *multiset orders* (cf. Section A.3).

Lemma 1.4.9. *Every peak decreasing* ARS *is confluent.*

Proof Let $>$ be a well-founded order on I which shows that the ARS $\mathcal{A} = \langle A, \{\alpha\}_{\alpha \in I} \rangle$ is peak decreasing. With every conversion C in $\mathcal{A}$ we associate the multiset M_C consisting of the labels of its steps. These multisets are compared by the multiset extension $>_{\mathsf{mul}}$ of $>$, which is a well-founded order on $\mathcal{M}(I)$ by Theorem A.3.8. We prove $\leftrightarrow^* \subseteq \downarrow$ by well-founded induction on $>_{\mathsf{mul}}$. Consider a conversion C between a and b. According to Lemma 1.2.6 we either have $a \downarrow b$ or $a \leftrightarrow^* \cdot \leftarrow \cdot \rightarrow \cdot \leftrightarrow^* b$. In the former case we are done. In the latter case there exist labels $\alpha, \beta \in I$ and multisets $\Gamma_1, \Gamma_2 \in \mathcal{M}(I)$ such that $M_C = \Gamma_1 \uplus \{\alpha, \beta\} \uplus \Gamma_2$. By the peak decreasingness assumption there exists a conversion C' between a and b such that $M_{C'} = \Gamma_1 \uplus \Gamma \uplus \Gamma_2$ with $\Gamma \in \mathcal{M}(\vee\alpha\beta)$. We obviously have $\{\alpha, \beta\} >_{\mathsf{mul}} \Gamma$ and hence $M_C >_{\mathsf{mul}} M_{C'}$ by Exercise A.29. We obtain $a \downarrow b$ from the induction hypothesis. □

Example 1.4.10. The ARS $\mathcal{A} = \langle A, \{\alpha, \beta\} \rangle$ of Example 1.4.1 is peak decreasing. For *trivial* local peaks (i.e., local peaks consisting of two identical steps) like $\mathsf{d}\ {}_\alpha\!\leftarrow \mathsf{e} \rightarrow_\alpha \mathsf{d}$ and $\mathsf{c}\ {}_\beta\!\leftarrow \mathsf{a} \rightarrow_\alpha \mathsf{c}$ there is nothing to check. There are four non-trivial local peaks:

$$\mathsf{a} \xleftarrow{\beta} \mathsf{b} \xrightarrow{\beta} \mathsf{d} \qquad \mathsf{b} \xleftarrow{\alpha} \mathsf{c} \xrightarrow{\beta} \mathsf{c} \qquad \mathsf{b} \xleftarrow{\alpha} \mathsf{c} \xrightarrow{\beta} \mathsf{e} \qquad \mathsf{c} \xleftarrow{\beta} \mathsf{c} \xrightarrow{\beta} \mathsf{e}$$

Take $\beta > \alpha$. We have

$$\mathsf{a} \xrightarrow{\alpha} \mathsf{c} \xleftarrow{\alpha} \mathsf{d} \qquad \mathsf{b} \xleftarrow{\alpha} \mathsf{c} \qquad \mathsf{b} \xleftarrow{\alpha} \mathsf{c} \xleftarrow{\alpha} \mathsf{d} \xleftarrow{\alpha} \mathsf{e} \qquad \mathsf{c} \xleftarrow{\alpha} \mathsf{d} \xleftarrow{\alpha} \mathsf{e}$$

We continue this section with an easy result which states that also the task of showing termination of an ARS $\langle A, \to \rangle$ can be simplified. It is based on *relative termination*.

Definition 1.4.11. Let $\mathcal{A} = \langle A, \{\alpha, \beta\} \rangle$ be an ARS. The relation α/β is defined as $\beta^*\alpha\beta^*$. If α/β is well-founded then we call α *relatively terminating* with respect to β. In this case we also say that α/β is terminating.

If α is relatively terminating with respect to β then any infinite sequence in $\mathcal{A}$ contains only finitely many α-steps.

Example 1.4.12. The relation α in the ARS $\mathcal{A} = \langle A, \{\alpha, \beta\} \rangle$ of Example 1.4.1 is terminating but not relatively terminating with respect to β. The infinite rewrite sequence

$$\mathsf{c} \xrightarrow{\alpha} \mathsf{b} \xrightarrow{\beta} \mathsf{d} \xrightarrow{\alpha} \mathsf{c} \xrightarrow{\alpha} \cdots$$

contains infinitely many α-steps. The same sequence shows that β is not relatively terminating with respect to α.

The easy proof of the following lemma is left to the reader (Exercise 1.34).

Lemma 1.4.13. *The statements*
1. *α is relatively terminating with respect to β,*
2. *$\beta^*\alpha$ is terminating, and*
3. *$\alpha\beta^*$ is terminating*

are equivalent for every ARS $\mathcal{A} = \langle A, \{\alpha, \beta\} \rangle$.

The next lemma states that relative termination is helpful for easing the task of showing termination by decomposition. In Chapter 6 we will encounter confluence criteria based on relative termination.

Lemma 1.4.14. *Let $\mathcal{A} = \langle A, \{\alpha, \beta\} \rangle$ be an* ARS. *If α/β and β are terminating then $\mathcal{A}$ is terminating.*

Proof Suppose $\mathcal{A}$ is not terminating. So there exists an infinite rewrite sequence $a_0 \to a_1 \to a_2 \to \cdots$. Because α is relatively terminating with respect to β, this sequence contains only finitely many α-steps. Thus there is an $i \geqslant 0$ such that the tail $a_i \to a_{i+1} \to a_{i+2} \to \cdots$ consists entirely of β-steps, contradicting the termination of β. □

Next we consider a special case of relative termination.

Definition 1.4.15. Let $\mathcal{A} = \langle A, \{\alpha, \beta\} \rangle$ be an ARS. We say that α *quasi-commutes over* β if $\beta\alpha \subseteq \alpha(\alpha \cup \beta)^*$, see Figure 1.8(iii).

Example 1.4.16. The relation α in the ARS $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ of Example 1.4.1 does not quasi-commute over β because $\mathsf{b} \to_\beta \mathsf{a} \to_\alpha \mathsf{c}$ and b is an α-normal form. The sequence $\mathsf{d} \to_\alpha \mathsf{c} \to_\beta \mathsf{e}$ shows that β does not quasi-commute over α.

Lemma 1.4.17. *Let $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ be an* ARS. *If α quasi-commutes over β then every α-terminating element is α/β-terminating.*

Proof From the quasi-commutation assumption we obtain $\beta^*\alpha \subseteq \alpha(\alpha\cup\beta)^*$ by a straightforward induction argument. So α quasi-commutes over β^*. We prove that every α-terminating element $a \in A$ is α/β-terminating by well-founded induction on the restriction of $\to_\alpha$ to α-terminating elements, which is a well-founded relation. If $a \in \mathsf{NF}(\alpha/\beta)$ then the claim is trivial. Consider an arbitrary step $a \to_{\alpha/\beta} b$, i.e., $a \to_\beta^* \cdot \to_\alpha \cdot \to_\beta^* b$. Using the quasi-commutation of α over β^*, the latter sequence can be written as $a \to_\alpha a' \to^* b$. The element a' is α-terminating because a is α-terminating and $a \to_\alpha a'$. Hence the induction hypothesis yields the α/β-termination of a'. Since $(\alpha \cup \beta)^* = \beta^* \cup (\alpha/\beta)^*$ and α/β-terminating elements are preserved under $\to_\beta$, it follows that b is α/β-terminating. Because this holds for any step $a \to_{\alpha/\beta} b$, element a is α/β-terminating. □

Example 1.4.18. Consider the ARS $\mathcal{A} = \langle\{\mathsf{a}, \mathsf{b}, \mathsf{c}\}, \{\alpha, \beta\}\rangle$ with

$$\mathsf{a} \xrightarrow{\beta} \mathsf{b} \xrightarrow{\alpha} \mathsf{c}$$

Clearly α is relatively terminating with respect to β and β is terminating. Thus $\mathcal{A}$ is terminating by Lemma 1.4.14. Note that relative termination cannot be shown by Lemma 1.4.17 since α does not quasi-commute over β.

The following statement is an immediate consequence of Lemmata 1.4.14 and 1.4.17.

Corollary 1.4.19. *Let $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ be an* ARS *such that α quasi-commutes over β. If α and β are terminating then $\mathcal{A}$ is terminating.*

Exercises

1.30 Let $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ be an ARS. Show that each of the following conditions is sufficient for the commutation of α and β, see Figure 1.8(iv,v,vi):

a $\alpha^-\beta \subseteq \beta^=(\alpha^-)^*$

b $\alpha^-\beta^* \subseteq \beta^*(\alpha^-)^*$

c $\alpha^-\beta \subseteq \beta^*(\alpha^-)^+$ and α is terminating

1.31 Let $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ be an ARS. The relations α and β *subcommute* if $\alpha^-\beta \subseteq \beta^=(\alpha^-)^=$, see Figure 1.8(vii). Show that α and β commute if and only if α^+ and β^+ subcommute.

1.32 Suppose $\mathcal{A} = \langle A, \{\alpha\}_{\alpha\in I}\rangle$ is an ARS and let $\alpha \in I$. We call α *non-splitting* if ${}_\alpha\!\leftarrow \cdot \to \subseteq \to^= \cdot {}^*\!\leftarrow$, see Figure 1.8(viii). Now consider an ARS $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ such that α is non-splitting and β is locally confluent and relatively terminating with respect to α. Prove that $\mathcal{A}$ is confluent.

1.33 ***a*** Prove that every connected ARS (Exercise 1.27) is peak decreasing.

b Construct a peak decreasing ARS that is not connected.

c Is every terminating peak decreasing ARS connected?

1.34 ***a*** Prove Lemma 1.4.13.

b Let $\mathcal{A} = \langle A, \{\alpha, \beta, \gamma\}\rangle$ be an ARS such that $\alpha/(\beta \cup \gamma)$ and β/γ are terminating. Prove that $(\alpha \cup \beta)/\gamma$ is terminating.

1.35 Let $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ be an ARS. We say that β *preserves* α*-normal forms* if $b \in \mathsf{NF}(\alpha)$ whenever $a \to_\beta b$ with $a \in \mathsf{NF}(\alpha)$.

a Show that β preserves α-normal forms if α quasi-commutes over β.

b Suppose $(\alpha \cup \beta)^! \subseteq \alpha^!\beta^!$. Show that $\mathcal{A}$ has unique normal forms whenever both α and β have unique normal forms.

c Does part (b) hold for the other properties displayed in Figure 1.6?

d Show that part (b) holds for confluence if we additionally require that $\mathcal{A}$ is normalizing.

e Can normalization of $\mathcal{A}$ in part (d) be replaced by normalization of both α and β?

f One of the following conditions implies $(\alpha \cup \beta)^! \subseteq \alpha^!\beta^!$:

- ▷ $\mathcal{A}$ is confluent, α is normalizing, and β preserves α-normal forms
- ▷ $\mathcal{A}$ is complete and $\beta\alpha \subseteq \alpha^*\beta^*$

Which one? Give a proof and a counterexample.

1.36 Let $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ be an ARS such that α and β are complete. Show that $\mathcal{A}$ is complete whenever $\alpha \subseteq \beta^!\alpha^+(\beta^-)^!$.

1.37 Let $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ be an ARS such that both α and β are terminating.

a Prove that $\mathcal{A}$ is terminating whenever $\alpha \cup \beta$ is transitive.

b Prove that $\mathcal{A}$ is terminating whenever $\beta\alpha \subseteq \alpha(\alpha \cup \beta)^* \cup \beta$.

c Show that the result of part (b) generalizes both part (a) and Corollary 1.4.19.

1.38 Let $\mathcal{A} = \langle A, \{\alpha, \beta, \gamma\}\rangle$ be an ARS such that $\alpha\beta \subseteq \beta^+\alpha^*$, $\alpha\gamma \subseteq \gamma(\alpha \cup \beta)^*$, and β is terminating. Prove the following inclusions:

a $(\alpha \cup \beta)^* \subseteq \beta^*\alpha^*$

b $\alpha^*\gamma^* \subseteq (\beta \cup \gamma)^*\alpha^*$

c $(\alpha \cup \beta \cup \gamma)^* \subseteq (\beta \cup \gamma)^*\alpha^*$

1.39 Let $\mathcal{A} = \langle A, \{\alpha_1, \ldots, \alpha_n\}\rangle$ be an ARS such that $\alpha_1 \cup \cdots \cup \alpha_n$ is transitive.

a Prove that $\mathcal{A}$ is terminating if and only if α_i is terminating for all $1 \leqslant i \leqslant n$.

b Show that the result of part (a) fails if $\to_\mathcal{A}$ is divided into an infinite union of relations.

1.5 Strategies

In this section we give an abstract account of strategies. In Chapter 7 we apply the definitions and results presented here to the concrete setting of term rewriting.

Definition 1.5.1. A (many-step) *rewrite strategy* $\mathcal{S}$ for an ARS $\mathcal{A} = \langle A, \to_\mathcal{A}\rangle$ is a relation $\to_\mathcal{S}$ such that $\to_\mathcal{S} \subseteq \to_\mathcal{A}^+$ and $\mathsf{NF}(\to_\mathcal{S}) = \mathsf{NF}(\mathcal{A})$. If $\to_\mathcal{S} \subseteq \to_\mathcal{A}$ then $\mathcal{S}$ is called a *one-step* strategy. We say that a strategy $\mathcal{S}$ is *deterministic* if $a = b$ whenever $a \,{}_\mathcal{S}\!\leftarrow \cdot \to_\mathcal{S} b$.

Example 1.5.2. Consider the ARS $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ of Example 1.4.1. Neither α nor β is a strategy for $\mathcal{A}$ because $\mathsf{b} \in \mathsf{NF}(\alpha) \setminus \mathsf{NF}(\mathcal{A})$ and $\mathsf{d} \in \mathsf{NF}(\beta) \setminus \mathsf{NF}(\mathcal{A})$. The relation $\beta \cup \{(\mathsf{d}, \mathsf{c}), (\mathsf{e}, \mathsf{d})\}$ is a strategy for $\mathcal{A}$. It is not deterministic as $(\mathsf{c}, \mathsf{c}), (\mathsf{c}, \mathsf{e}) \in \beta$.

A major aim of strategies is to compute normal forms. To ensure that this is always possible, the normalization property defined below is required.

Definition 1.5.3. A rewrite strategy $\mathcal{S}$ for an ARS $\mathcal{A}$ is *normalizing* if every normalizing element is $\mathcal{S}$-terminating. We call $\mathcal{S}$ *hyper-normalizing* if every normalizing element is $\mathcal{S}/\mathcal{A}$-terminating (cf. Definition 1.4.11).

Normalization is the property that by repeatedly performing steps according to the strategy a normal form will be computed, provided the starting term has a normal form. Hyper-normalization is a much stronger property. It guarantees that normal forms will still be computed even if between successive strategy steps arbitrary but finitely many other steps are performed.

Example 1.5.4. All strategies for the ARS $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ of Example 1.4.1 are vacuously normalizing because $\mathsf{NF}(\mathcal{A}) = \varnothing$. Consider the ARS $\mathcal{B} = \langle A, \{\beta, \gamma\}\rangle$ with $\gamma = (\alpha \setminus \{(\mathsf{e}, \mathsf{d})\}) \cup \{(\mathsf{b}, \mathsf{e})\}$. The relation γ is a deterministic normalizing strategy for $\mathcal{B}$. It is not hyper-normalizing as $\mathsf{a} \to_\gamma \mathsf{c} \to_\gamma \mathsf{b} \to_\beta \mathsf{a}$.

Quasi-commutation is a sufficient condition for hyper-normalization of a normalizing strategy.

Lemma 1.5.5. *A normalizing rewrite strategy $\mathcal{S}$ for an* ARS *$\mathcal{A}$ is hyper-normalizing if $\mathcal{S}$ quasi-commutes over $\mathcal{A}$.*

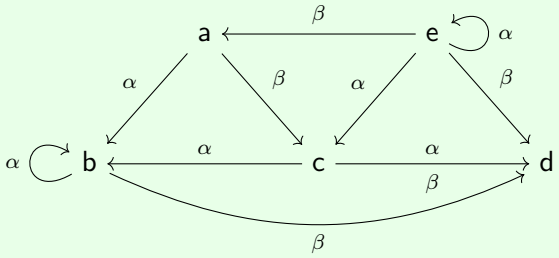
Proof Since every $\mathcal{S}$-terminating element is $\mathcal{S}/\mathcal{A}$-terminating according to Lemma 1.4.17, the result follows from the definitions of normalization and hyper-normalization. □

Normalization of a strategy follows from the stronger property introduced below.

Definition 1.5.6. A rewrite strategy $\mathcal{S}$ for an ARS $\mathcal{A}$ is *cofinal* if for every $a \to_{\mathcal{A}}^* b$ and maximal sequence $a = a_0 \to_{\mathcal{S}} a_1 \to_{\mathcal{S}} a_2 \to_{\mathcal{S}} \cdots$ there is a $k \geqslant 0$ such that $b \to_{\mathcal{A}}^* a_k$.

A maximal sequence is either a finite sequence ending in a normal form or an infinite sequence.

Example 1.5.7. Consider the ARS $\mathcal{A} = \langle A, \{\alpha, \beta\}\rangle$ with



Both α and β are strategies for $\mathcal{A}$. The first one is not cofinal since $\mathsf{a} \to_\alpha \mathsf{b} \to_\alpha \mathsf{b} \to_\alpha \cdots$ is a maximal α-rewrite sequence and $\mathsf{a} \to^* \mathsf{d}$ but neither $\mathsf{d} \to^* \mathsf{a}$ nor $\mathsf{d} \to^* \mathsf{b}$ holds. Because all maximal β-rewrite sequences end in d and all elements rewrite to d, β is cofinal for $\mathcal{A}$.

Lemma 1.5.8. *Cofinal strategies are normalizing.*

Proof Let $\mathcal{S}$ be a cofinal strategy for an ARS $\mathcal{A}$. Consider a maximal sequence $a = a_0 \to_{\mathcal{S}} a_1 \to_{\mathcal{S}} a_2 \to_{\mathcal{S}} \cdots$ and let $a \to_{\mathcal{A}}^{!} b$. By definition, there is a $k \geqslant 0$ such that $b \to_{\mathcal{A}}^{*} a_k$. Since b is a normal form, $b = a_k$. Hence the sequence ends in a normal form. It follows that $\mathcal{S}$ is normalizing. □

The opposite of a normalizing strategy is a *perpetual* strategy.

Definition 1.5.9. A strategy $\mathcal{S}$ is called *perpetual* if every maximal $\mathcal{S}$-rewrite sequence starting from a non-terminating element with respect to the underlying ARS is infinite.

The following result explains why the study of strategies is especially of interest for non-terminating ARSs.

Lemma 1.5.10. *Every strategy is hyper-normalizing and perpetual for terminating* ARS*s.*

Proof Immediate consequence of the definitions. □

Next we define a normalizing strategy for ARSs with the Z-property.

Definition 1.5.11. Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS that has the Z-property with respect to $\bullet$. The strategy $\mathcal{S}_\bullet$ is defined as follows: $a \mathrel{-\!\!\bullet\!\!\to} b$ if $a \notin \mathsf{NF}(\mathcal{A})$ and $b = a^\bullet$.

Note that $\mathrel{-\!\!\bullet\!\!\to}$ is a deterministic many-step strategy.

Theorem 1.5.12. *The strategy $\mathcal{S}_\bullet$ is normalizing for every* ARS *that has the* Z*-property with respect to $\bullet$.*

Proof First we show $b \to^* \bullet^n(a)$ and $a \mathrel{-\!\!\bullet\!\!\to}^{\leqslant n} \bullet^n(a)$ whenever $a \to^n b$ and $n > 0$, by induction on n. Suppose $a \to^{n-1} c \to b$. The Z-property yields $b \to^* \bullet(c)$. Moreover, $c \mathrel{-\!\!\bullet\!\!\to} \bullet(c)$ by the definition of $\mathcal{S}_\bullet$. If $n = 1$ then $a = c$ and we are done. If $n > 1$ then we obtain $c \to^* \bullet^{n-1}(a)$ and $a \mathrel{-\!\!\bullet\!\!\to}^{\leqslant n-1} \bullet^{n-1}(a)$ from the induction hypothesis. From $c \to^* \bullet^{n-1}(a)$ we obtain $\bullet(c) \to^* \bullet^n(a)$ by Lemma 1.2.16. Hence also $b \to^* \bullet^n(a)$. Since a is not a normal form and $\mathrel{-\!\!\bullet\!\!\to}$ is a many-step strategy, we infer $a \to^+ \bullet^{n-1}(a)$ from $a \mathrel{-\!\!\bullet\!\!\to}^{\leqslant n-1} \bullet^{n-1}(a)$. Hence $a \to^* d \to \bullet^{n-1}(a)$ for some d. The Z-property yields $\bullet^{n-1}(a) \to^* \bullet(d) \to^* \bullet^n(a)$. We distinguish two cases. If $\bullet^{n-1}(a) \in \mathsf{NF}(\to)$ then $\bullet^n(a) = \bullet^{n-1}(a)$ and thus $a \mathrel{-\!\!\bullet\!\!\to}^{\leqslant n-1} \bullet^n(a)$. If $\bullet^{n-1}(a) \notin \mathsf{NF}(\to)$ then $\bullet^{n-1}(a) \mathrel{-\!\!\bullet\!\!\to} \bullet^n(a)$ and thus $a \mathrel{-\!\!\bullet\!\!\to}^{\leqslant n} \bullet^n(a)$. Now let $a \to^n b$ with $n > 0$ and $b \in \mathsf{NF}(\to)$. We have $a \mathrel{-\!\!\bullet\!\!\to}^{\leqslant n} \bullet^n(a) \mathrel{{}^*\!\!\leftarrow} b$ and thus $a \mathrel{-\!\!\bullet\!\!\to}^{\leqslant n} b$ as b is a normal form. So $\mathrel{-\!\!\bullet\!\!\to}$ reaches b in at most n steps. Since $\mathrel{-\!\!\bullet\!\!\to}$ is deterministic, we conclude the normalization of $\mathcal{S}_\bullet$. □

Theorem 1.5.12 can be strengthened (Exercise 1.48) to hyper-normalization. We conclude this section with another sufficient condition for normalization.

Definition 1.5.13. An ARS $\mathcal{A} = \langle A, \to \rangle$ has *random descent* (RD) if for every conversion $a \leftrightarrow^* b$ with normal form b we have $a \to^n b$ with $n + l = r$. Here l (r) denotes the number of $\leftarrow$ ($\to$) steps in the conversion $a \leftrightarrow^* b$. A strategy $\mathcal{S}$ for $\mathcal{A}$ has random descent if $\langle A, \to_{\mathcal{S}} \rangle$ has random descent.

Example 1.5.14. The ARS $\mathcal{A}$ of Example 1.5.7 has no random descent because there exists a conversion $\mathsf{b} \leftarrow \mathsf{c} \rightarrow \mathsf{d}$ with normal form d and $l = r = 1$ but $\mathsf{b} \rightarrow^0 \mathsf{d}$ obviously does not hold. The strategies α and β also lack random descent as witnessed by the conversions $\mathsf{b}\ {}_\alpha\!\leftarrow \mathsf{c} \rightarrow_\alpha \mathsf{d}$ and $\mathsf{a}\ {}_\beta\!\leftarrow \mathsf{e} \rightarrow_\beta \mathsf{d}$. The relation $\gamma = \beta \setminus \{(\mathsf{e},\mathsf{a})\}$ is a deterministic strategy for $\mathcal{A}$ with random descent. A non-deterministic strategy for $\mathcal{A}$ with random descent is provided by the relation $\delta = (\beta \setminus \{(\mathsf{e},\mathsf{d})\}) \cup \{(\mathsf{a},\mathsf{b})\}$.

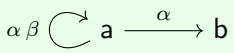
Theorem 1.5.15. *Let $\mathcal{A}$ be an* ARS *with random descent. If $a \leftrightarrow^* b$ with normal form b then a is complete and all rewrite sequences from a to b have the same length.*

Proof Let l (r) be the number of $\leftarrow$ ($\rightarrow$) steps in the conversion from a to b. We have $l \leqslant r$ since $n + l = r$ for some n by random descent. First we prove termination of a. For a proof by contradiction, suppose $a = a_0 \rightarrow a_1 \rightarrow a_2 \rightarrow \cdots$ is an infinite rewrite sequence. Clearly, $a \rightarrow^{r-l} a_{r-l}$ and thus there exists a conversion $a_{r-l}\ {}^*\!\leftarrow a \leftrightarrow^* b$ with r backwards and r forwards steps. Hence $a_{r-l} = b$ by another application of random descent and therefore $b \rightarrow a_{r-l+1}$, contradicting the fact that b is a normal form. Next we prove confluence of a. Suppose $c\ {}^*\!\leftarrow a \rightarrow^* d$. We obtain the two conversions $c \leftrightarrow^* b$ and $d \leftrightarrow^* b$, which are transformed into $c \downarrow d$ by two applications of random descent. Finally, assume there are two rewrite sequences $a \rightarrow^m b$ and $a \rightarrow^n b$ from a to b of length m and n. Reversing the first sequence and appending the second one yields a conversion $b \leftrightarrow^* b$ with m backwards and n forwards steps. A final application of random descent yields $b \rightarrow^k b$ for some k with $k + m = n$. Since b is a normal form, $k = 0$ and thus $m = n$ as desired. □

Corollary 1.5.16. *Any strategy for an* ARS *with random descent is normalizing.*

Note that random descent is a requirement on the underlying ARS in the above corollary. The strategy need not have random descent.

Example 1.5.17. Consider the ARS $\mathcal{A} = \langle \{a,b\}, \{\alpha,\beta\} \rangle$ with



The relation β is a strategy for $\mathcal{A}$ with random descent but clearly not normalizing.

Exercises

1.40 ***a*** Construct a normalizing strategy for the ARS of Exercise 1.15.

b Show that every ARS admits a normalizing one-step strategy.

c Does every ARS admit a hyper-normalizing strategy?

1.41 Show that a deterministic rewrite strategy $\mathcal{S}$ for an ARS $\mathcal{A}$ is normalizing if $\rightarrow_{\mathcal{A}}^* \subseteq \rightarrow_{\mathcal{S}}^* \cdot \rightarrow_{\mathcal{B}}^*$ and $\mathsf{NF}(\mathcal{A}) \subseteq \mathsf{NF}(\mathcal{B}^{-1})$ for some ARS $\mathcal{B}$.

1.42 Is quasi-commutation a necessary condition for hyper-normalization of a normalizing strategy?

1.43 Can Lemma 1.5.8 be strengthened to hyper-normalization?

1.44 ***a*** Show that the converse of Lemma 1.5.8 does not hold.

b Show that every ARS admitting a cofinal strategy is confluent.

c Does every confluent ARS admit a cofinal strategy?

1.45 An ARS $\mathcal{A} = \langle A, \to \rangle$ is *balanced weakly Church–Rosser* (BWCR) if for all elements $a, b, c \in A$ with $b \leftarrow a \to c$ there exist an $n \geqslant 0$ and an element $d \in A$ such that $b \to^n d \,{}^n\!\leftarrow c$. Note that WCR^1 (Exercise 1.28) implies BWCR.

a Show that BWCR is sufficient but not necessary for random descent.

b Is every normalizing element with the BWCR property in an arbitrary ARS confluent?

1.46 Are strategies for ARSs with random descent hyper-normalizing?

1.47 ***a*** Let $\mathcal{S}$ be a one-step strategy for an ARS $\mathcal{A}$. Show that $\mathcal{S}/\mathcal{A}$ is a many-step strategy for $\mathcal{A}$.

b Show that a strategy $\mathcal{S}$ is hyper-normalizing if and only if $\mathcal{S}/\mathcal{A}$ is normalizing.

1.48 A strategy $\mathcal{S}$ for an ARS $\mathcal{A}$ is called *hyper-cofinal* if $\mathcal{S}/\mathcal{A}$ is cofinal.

a Show that hyper-cofinal strategies are hyper-normalizing.

b Let $\mathcal{A} = \langle A, \to \rangle$ be an ARS that has the Z-property with respect to $\bullet$. Show that $\mathcal{S}_\bullet$ is a hyper-cofinal strategy for $\mathcal{A}$.

Bibliographic Notes

The origins of abstract rewriting can be traced back to early work on combinatory logic and λ-calculus. An influential paper by Huet [57] and surveys by Klop [73, 74] and Dershowitz and Jouannaud [33] popularized the independent study of abstract rewriting.

The counterexample of Figure 1.4(i) against the implication WCR $\implies$ CR is attributed to Kleene in Hindley [52]. The Z-property is due to Dehornoy and van Oostrom [104]. Exercise 1.17 comes from Curien and Ghelli [19]. Newman's Lemma is from Newman [97]. Exercise 1.25 is from the same paper. The first proof of Newman's Lemma in Section 1.3 by well-founded induction originates from Huet [57]. The second proof is from Barendregt [11]. Strong confluence (Exercise 1.17) was introduced in the same paper by Huet. The property in Exercise 1.17(c) originates from [97] and is known as *semi-confluence* [5]. Exercise 1.27 is from Winkler and Buchberger [137]. Lemma 1.4.4 comes from Hindley [52]. It is sometimes referred to as the Lemma of Hindley – Rosen. Lemma 1.4.7 is from Rosen [115]. Peak decreasingness was introduced in [55]. Lemmata 1.4.14 and 1.4.17 are from Bachmair and Dershowitz [7]. Parts (a) and (b) of Exercise 1.30 are based on Hindley [52] and Staples [121], part (c) is from Geser [44, p. 38]. Exercises 1.32 and 1.37(a) come from Geser [44]. Exercise 1.35 is based on Prehofer [112] and Alfons Geser (personal communication). Exercise 1.37(b,c) is from Doornbos *et al.* [37]. Variations of the result of part (b) are presented in Dershowitz [32]. Exercise 1.38 is from Accattoli [2]. Exercise 1.39 is based on Podelski and Rybalchenko [108] and Vincent van Oostrom (personal communication). Theorem 1.5.12 and Exercise 1.48 are based on [104]. Theorem 1.5.15 and Exercise 1.46 are based on van Oostrom and Toyama [102, 105, 131].

In the literature one often encounters different terminology. Abstract rewrite systems are called *abstract reduction systems* in Klop [72, 75], *replacement systems* in Staples [121], and *general replacement systems* in Rosen [115]. Normal forms are sometimes called *irreducible*, a terminating ARS is said to be *noetherian*, a complete ARS is often called *convergent*, and semi-completeness is sometimes referred to as *unique normalization*. Some authors use the phrase *unique normal forms* and the notation UN for the property UNC introduced in Definition 1.2.10. Occasionally one encounters UN$^\to$ to denote the property UN of Definition 1.2.1. Various (term) rewriting 'schools' adopt different notations. The most important differences are $\twoheadrightarrow$ instead of $\to^*$, $=$ instead of $\leftrightarrow^*$, and $\equiv$ instead of $=$.

Chapter 2

Equational Logic

In the previous chapter we considered abstract rewriting. By adding structure to the objects and the rewrite relation of ARSs we obtain term rewrite systems. In this chapter we study this term structure. The syntax of terms is presented in Section 2.1. Important concepts like positions and substitutions are introduced and a simple algorithm for term matching is presented. Algebras give meaning to terms and are the topic of Section 2.2. In Section 2.4 the theory of substitutions is developed further and the subsumption and encompassment relations are introduced. We define the syntax and semantics of equational systems, a generalization of term rewrite systems, in Section 2.3 and we show the soundness and completeness of equational reasoning. In the final section of this chapter we introduce unification. Unification plays a central role in the study of confluence of term rewrite systems, as we will see in Chapter 6.

2.1 Terms

Term rewrite systems, the main topic of this book, are ARSs where the objects have a term structure and rewrite steps are generated by rules that operate on these structured objects. To motivate the development in this section, consider the rules

$$\mathsf{0} + y \to y \qquad\qquad \mathsf{s}(x) + y \to \mathsf{s}(x + y)$$

which can be viewed as a recursive definition of addition of natural numbers in unary notation. An expression like $\mathsf{s}(\mathsf{s}(\mathsf{0}) + \mathsf{s}(\mathsf{s}(\mathsf{0})))$ is simplified to $\mathsf{s}(\mathsf{s}(\mathsf{0} + \mathsf{s}(\mathsf{s}(\mathsf{0}))))$ by applying the rule $\mathsf{s}(x)+y \to \mathsf{s}(x+y)$ to the subexpression $\mathsf{s}(\mathsf{0})+\mathsf{s}(\mathsf{s}(\mathsf{0}))$ after binding the variables x and y in the rule to the terms $\mathsf{0}$ and $\mathsf{s}(\mathsf{s}(\mathsf{0}))$, respectively. The effect of this rule application is the replacement of the subexpression by the instantiated right-hand side $\mathsf{s}(\mathsf{0} + \mathsf{s}(\mathsf{s}(\mathsf{0})))$ of the rule. In this section we provide formal definitions of the operations required to manipulate terms by applying rules or equations.

Definition 2.1.1. A *signature* is a set $\mathcal{F}$ of *function symbols*. Associated with every $f \in \mathcal{F}$ is a natural number denoting its *arity*, i.e., the number of arguments it is supposed to have. A function symbol of arity n is called n-ary. We use *unary* for 1-ary, *binary* for 2-ary, and *ternary* for 3-ary function symbols. Function symbols of arity 0 are called *constants*.

Definition 2.1.2. Let $\mathcal{F}$ be a signature and $\mathcal{V}$ a countably infinite set of *variables* disjoint from $\mathcal{F}$. The set $\mathcal{T}(\mathcal{F},\mathcal{V})$ of *terms* built from $\mathcal{F}$ and $\mathcal{V}$ is the smallest set such that every variable is a term, every constant is a term, and if $f \in \mathcal{F}$ is a function symbol of arity $n > 0$ and $t_1, \ldots, t_n$ are terms then $f(t_1, \ldots, t_n)$ is a term. In the latter case, the terms $t_1, \ldots, t_n$ are called the *arguments* of $f(t_1, \ldots, t_n)$.

Function symbols of arity greater than 0 are typically denoted by f, g, h, constants by a, b, c, variables by x, y, z, and terms by s, t, u (and their derivatives, like s' and t_1). In examples we use sans serif font to denote function symbols.

Example 2.1.3. Consider the signature $\mathcal{F}$ consisting of a binary function symbol $+$, a unary function symbol s, and a constant $\mathsf{0}$. Suppose x and y are variables in $\mathcal{V}$. The term $t = +(\mathsf{s}(+(\mathsf{0},\mathsf{s}(x))),+(y,\mathsf{s}(x)))$ belongs to $\mathcal{T}(\mathcal{F},\mathcal{V})$. Figure 2.1 shows a tree representation of t.

Sometimes we use infix notation for binary function symbols and prefix or postfix notation for unary function symbols to enhance readability. For instance, using infix notation for $+$, the term t of Example 2.1.3 is written as $\mathsf{s}(\mathsf{0}+\mathsf{s}(x))+(y+\mathsf{s}(x))$.

Definition 2.1.4. Let s and t be terms in $\mathcal{T}(\mathcal{F},\mathcal{V})$. We say that s is a *subterm* of t and we write $s \trianglelefteq t$ if either $s = t$ or $t = f(t_1, \ldots, t_n)$ and s is a subterm of t_i for some $1 \leqslant i \leqslant n$. In the latter case we say that s is a *proper* subterm of t. We write $\triangleleft$ to denote the proper subterm relation.

Example 2.1.5. The subterms of the term $t = \mathsf{s}(\mathsf{0}+\mathsf{s}(x))+(y+\mathsf{s}(x))$ are t itself, $\mathsf{s}(\mathsf{0}+\mathsf{s}(x))$, $\mathsf{0}+\mathsf{s}(x)$, $\mathsf{0}$, $\mathsf{s}(x)$, x, $y+\mathsf{s}(x)$, and y. Note that x and $\mathsf{s}(x)$ occur twice in t.

Lemma 2.1.6. *The relation $\triangleright$ is a well-founded order on terms. The relation $\trianglelefteq$ is a partial order on terms.*

Proof Irreflexivity of $\triangleright$ holds by definition. Let us write $t \triangleright_1 u$ if u is an argument of

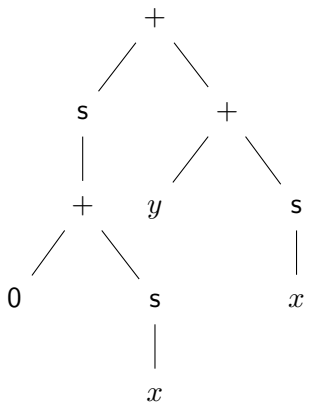


Figure 2.1: Tree representation of a term.

t. The relation $\rhd_1$ is well-founded by the inductive definition of terms. Since $\rhd$ is the transitive closure of $\rhd_1$, it inherits well-foundedness from $\rhd_1$ (Exercise A.15).

The relation $\trianglelefteq$ is reflexive by definition. Next we show its transitivity. So let $s \trianglelefteq t$ and $t \trianglelefteq u$. We use induction on the derivation of $t \trianglelefteq u$. If $t = u$ then $s \trianglelefteq u$ trivially holds. Otherwise, $u = f(u_1, \ldots, u_n)$ and $t \trianglelefteq u_i$ for some $1 \leqslant i \leqslant n$. The induction hypothesis yields $s \trianglelefteq u_i$ and thus $s \trianglelefteq u$ by definition. For anti-symmetry, suppose $s \trianglelefteq t$ and $t \trianglelefteq s$. If $s \neq t$ then $t \rhd s$ and $s \rhd t$, contradicting the well-foundedness of $\rhd$. Hence $s = t$. This concludes the proof that $\trianglelefteq$ is a partial order. It remains to establish transitivity of $\rhd$. So let $s \rhd t$ and $t \rhd u$. We have $t \trianglelefteq s$ and $u \trianglelefteq t$ with $s \neq t$ and $t \neq u$. Transitivity of $\trianglelefteq$ yields $u \trianglelefteq s$. Since $s = u$ contradicts the well-foundedness of $\rhd$, we have $s \neq u$ and thus $s \rhd u$ as desired. □

Proving properties by induction with respect to $\rhd$ is known as (strong) *structural induction*. In Section 2.4 we will introduce two other well-founded orders on terms.

Definition 2.1.7. Let $t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$. We inductively define the set $\mathcal{V}\mathsf{ar}(t)$ of variables occurring in t as follows:

$$\mathcal{V}\mathsf{ar}(t) = \begin{cases} \{t\} & \text{if } t \text{ is a variable} \\ \varnothing & \text{if } t \text{ is a constant} \\ \bigcup_{i=1}^{n} \mathcal{V}\mathsf{ar}(t_i) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

A term t is called *ground* if $\mathcal{V}\mathsf{ar}(t) = \varnothing$. The set of all ground terms is denoted by $\mathcal{T}(\mathcal{F})$.

It is not difficult to show that $\mathcal{T}(\mathcal{F})$ is the smallest set such that every constant belongs to $\mathcal{T}(\mathcal{F})$ and if $f \in \mathcal{F}$ is a function symbol of arity $n > 0$ and $t_1, \ldots, t_n$ belong to $\mathcal{T}(\mathcal{F})$ then so does $f(t_1, \ldots, t_n)$. This characterization is convenient when proving properties of ground terms by structural induction.

Definition 2.1.8. Let $t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$. We inductively define the set $\mathcal{F}\mathsf{un}(t)$ of function symbols occurring in t as follows:

$$\mathcal{F}\mathsf{un}(t) = \begin{cases} \varnothing & \text{if } t \text{ is a variable} \\ \{t\} & \text{if } t \text{ is a constant} \\ \{f\} \cup \bigcup_{i=1}^{n} \mathcal{F}\mathsf{un}(t_i) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

Example 2.1.9. The term t of Example 2.1.3 is not ground as $\mathcal{V}\mathsf{ar}(t) = \{x, y\}$. We have $\mathcal{F}\mathsf{un}(t) = \{+, \mathsf{s}, 0\}$.

In the third clause of the above definitions of $\mathcal{V}\mathsf{ar}(t)$ and $\mathcal{F}\mathsf{un}(t)$ the arity n of function symbol f is assumed to be greater than 0. If we allow for $n = 0$ then we get $\varnothing$ and $\{f\} = \{t\}$ respectively. So the third clause subsumes the second clause if we adopt the convention that $f(t_1, \ldots, t_n)$ denotes the constant f when $n = 0$. Hence when defining operations on terms we often give two instead of three clauses and when proving properties of ground terms by structural induction we have no base case.

Definition 2.1.10. Let $t \in \mathcal{T}(\mathcal{F},\mathcal{V})$. The *root symbol* of t is defined as follows:

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

In the following definitions we associate various measures to terms, which are convenient for inductive proofs.

Definition 2.1.11. The *size* $|t|$ of t is the number of variables and function symbols occurring in it, inductively defined as follows:

$$|t| = \begin{cases} 1 & \text{if } t \text{ is a variable} \\ 1 + \sum_{i=1}^{n} |t_i| & \text{if } t = f(t_1,\ldots,t_n) \end{cases}$$

The number of occurrences of a symbol $a \in \mathcal{F} \cup \mathcal{V}$ in t is denoted by $|t|_a$. We denote by $\|t\|$ the number of function symbols occurring in t:

$$\|t\| = \begin{cases} 0 & \text{if } t \text{ is a variable} \\ 1 + \sum_{i=1}^{n} \|t_i\| & \text{if } t = f(t_1,\ldots,t_n) \end{cases}$$

Definition 2.1.12. The *height* $\mathsf{height}(t)$ of t is inductively defined as follows:

$$\mathsf{height}(t) = \begin{cases} 0 & \text{if } t \text{ is a variable or a constant} \\ 1 + \max\{\mathsf{height}(t_i) \mid 1 \leqslant i \leqslant n\} & \text{if } t = f(t_1,\ldots,t_n) \text{ with } n > 0 \end{cases}$$

Example 2.1.13. Consider again the term t of Example 2.1.3. We have $\mathsf{root}(t) = +$, $|t| = 10$, $|t|_+ = 3$, $\|t\| = 7$, and $\mathsf{height}(t) = 4$.

We now introduce a formalism which enables us to easily distinguish multiple occurrences of the same subterm.

Definition 2.1.14. A *position* is a finite sequence of positive integers. The *root* position is the empty sequence and denoted by ϵ and pq denotes the concatenation of positions p and q. We define binary relations $\leqslant$, $<$, and $\parallel$ on positions as follows. We say that position p is *above* position q if there exists a (necessarily unique) position r such that $pr = q$. In that case we define $q \setminus p$ as the position r. If p is above q we also say that q is *below* p or p is a *prefix* of q, and we write $p \leqslant q$. We write $p < q$ if $p \leqslant q$ and $p \neq q$. If $p < q$ we say that p is a *proper* prefix of q. Positions p, q are *parallel*, denoted by $p \parallel q$, if neither $p \leqslant q$ nor $q \leqslant p$.

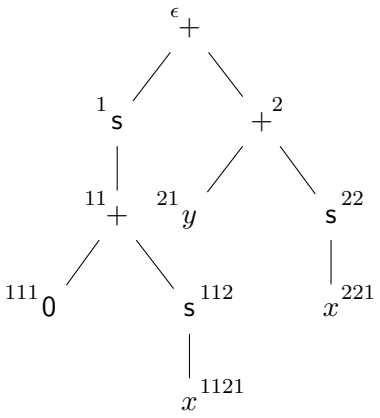


Figure 2.2: Positions in a term.

Example 2.1.15. The position 13 is a proper prefix of the position 132. We have $132 \setminus 13 = 2$. The positions 13 and 21 are parallel. If a position contains a number larger than 9, we use parentheses. So position 1 is a prefix of 1(11) but not of (11).

Definition 2.1.16. The set $\mathcal{P}\mathsf{os}(t)$ of positions in a term $t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ is inductively defined as follows:

$$\mathcal{P}\mathsf{os}(t) = \begin{cases} \{\epsilon\} & \text{if } t \text{ is a variable} \\ \{\epsilon\} \cup \{ip \mid 1 \leqslant i \leqslant n \text{ and } p \in \mathcal{P}\mathsf{os}(t_i)\} & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

Let $p \in \mathcal{P}\mathsf{os}(t)$. The subterm of t at position p is denoted by $t|_p$, i.e.,

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

The symbol $t(p)$ at position p in t is defined as $t(p) = \mathsf{root}(t|_p)$. We partition the set $\mathcal{P}\mathsf{os}(t)$ into $\mathcal{P}\mathsf{os}_{\mathcal{V}}(t) = \{p \in \mathcal{P}\mathsf{os}(t) \mid t|_p \in \mathcal{V}\}$ and $\mathcal{P}\mathsf{os}_{\mathcal{F}}(t) = \mathcal{P}\mathsf{os}(t) \setminus \mathcal{P}\mathsf{os}_{\mathcal{V}}(t)$.

Example 2.1.17. Figure 2.2 shows the positions in the term $t = \mathsf{s}(0 + \mathsf{s}(x)) + (y + \mathsf{s}(x))$. We have $\mathcal{P}\mathsf{os}_{\mathcal{V}}(t) = \{1121, 21, 221\}$ and $\mathcal{P}\mathsf{os}_{\mathcal{F}}(t) = \{\epsilon, 1, 11, 111, 112, 2, 22\}$. The two occurrences of the subterm $\mathsf{s}(x)$ are distinguished by their respective positions: 112 and 22.

An important operation on terms is the replacement of subterms. We decompose this operation into two steps. First the subterm that we want to replace is removed, resulting in a term with a hole.

Definition 2.1.18. We introduce a fresh constant symbol $\Box$, named *hole*, and we denote the subset of $\mathcal{T}(\mathcal{F} \cup \{\Box\}, \mathcal{V})$ consisting of all terms which contain exactly one occurrence

of $\Box$ by $\mathcal{C}(\mathcal{F},\mathcal{V})$. Terms in $\mathcal{C}(\mathcal{F},\mathcal{V})$ will be called *contexts*. The context $\Box$ is called the *empty* context.

It is not difficult to see that $\mathcal{C}(\mathcal{F},\mathcal{V})$ is the smallest set such that $\Box \in \mathcal{C}(\mathcal{F},\mathcal{V})$ and if $f \in \mathcal{F}$ has arity $n > 0$, $t_1,\ldots,t_{i-1},t_{i+1},\ldots,t_n \in \mathcal{T}(\mathcal{F},\mathcal{V})$ with $1 \leqslant i \leqslant n$, and $C \in \mathcal{C}(\mathcal{F},\mathcal{V})$ then $f(t_1,\ldots,t_{i-1},C,t_{i+1},\ldots,t_n) \in \mathcal{C}(\mathcal{F},\mathcal{V})$. This characterization of contexts facilitates proofs by induction.

We adopt the convention that 'term' denotes an element of $\mathcal{T}(\mathcal{F},\mathcal{V})$ and 'context' an element of $\mathcal{C}(\mathcal{F},\mathcal{V})$, for some fixed signature $\mathcal{F}$ that does not contain the hole $\Box$.

Definition 2.1.19. If t is a term then $t[\,]_p$ denotes the context that is obtained from t by replacing the subterm at position p by the hole. Formally:

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

The second step in the replacement operation is filling the hole with the new subterm.

Definition 2.1.20. If C is a context and t a term then $C[t]$ denotes the term that is obtained from C by replacing the hole by t:

$$C[t] = \begin{cases} t & \text{if } C = \Box \\ f(t_1,\ldots,C'[t],\ldots,t_n) & \text{if } C = f(t_1,\ldots,C',\ldots,t_n) \end{cases}$$

Furthermore, if p is a position in a term s then $s[t]_p = (s[\,]_p)[t]$ denotes the term that is obtained from s by replacing the subterm at position p by the term t.

Example 2.1.21. For the term t in Example 2.1.17 we have $t|_{1121} = t|_{221} = t(221) = x$ and $t|_{11} = 0 + \mathsf{s}(x) = t|_2[t|_{111}]_1$.

Lemma 2.1.22. *Let s and t be terms. The following statements are equivalent.*

1. *The term s is a subterm of t.*
2. *There exists a position $p \in \mathcal{P}\mathsf{os}(t)$ such that $t|_p = s$.*
3. *There exists a context C such that $t = C[s]$.*

Proof

1 $\Longrightarrow$ 2 We use induction on the structure of t. If t is a variable then $s = t$ and thus $t|_p = s$ for the position $p = \epsilon$. Let $t = f(t_1,\ldots,t_n)$. If $s = t$ then we take $p = \epsilon$ as before. Otherwise, s is a subterm of t_i for some $1 \leqslant i \leqslant n$. According to the induction hypothesis $t_i|_q = s$ for some position $q \in \mathcal{P}\mathsf{os}(t_i)$. Let $p = iq$. We clearly have $p \in \mathcal{P}\mathsf{os}(t)$ and $t|_p = t_i|_q = s$.

2 $\Longrightarrow$ 3 Let $C = t[\,]_p$. We have $C[s] = (t[\,]_p)[s] = t[s]_p = t[t|_p]_p = t$. The last equality requires an easy induction proof (Exercise 2.13(a)).

3 $\Longrightarrow$ **1** We use induction on the structure of C. If $C = \Box$ then $t = \Box[s] = s$ and thus s is a subterm of t. If $C = f(t_1, \ldots, C', \ldots, t_n)$ then $t = C[s] = f(t_1, \ldots, C'[s], \ldots, t_n)$. According to the induction hypothesis s is a subterm of $C'[s]$ and thus of t. $\Box$

By adding 'non-empty' to statements **2** and **3** we get a similar characterization of proper subterms (Exercise 2.4). In the following we use both contexts and positions to describe and manipulate subterms. The context framework is preferred if the exact positions do not matter.

Definition 2.1.23. A binary relation R on terms is *closed under contexts* if $C[s]\ R\ C[t]$ for all contexts C and terms s, t with $s\ R\ t$.

Example 2.1.24. The relation $\rhd$ is not closed under contexts. We have $\mathsf{f}(\mathsf{a}) \rhd \mathsf{a}$ but $\mathsf{g}(\mathsf{f}(\mathsf{a})) \rhd \mathsf{g}(\mathsf{a})$ does not hold. The relation $|\cdot| > |\cdot|$ is closed under contexts: If $|s| > |t|$ then $|C[s]| - |C[t]| = |s| - |t| > 0$ as a consequence of Exercise 2.3(c).

The following lemma states a simple but useful result.

Lemma 2.1.25. *Let R be a relation on terms that is closed under contexts. If R is well-founded then $R \cup \rhd$ is well-founded.*

Proof We show $\rhd \cdot R \subseteq R \cdot \rhd$. This implies that R quasi-commutes over $\rhd$ and hence the result follows from Corollary 1.4.19. Let $s \rhd t$ and $t\ R\ u$. So $s = C[t]$ for some non-empty context C. Because R is closed under contexts we infer $C[t]\ R\ C[u]$ from $t\ R\ u$. We clearly have $C[u] \rhd u$. Hence $s\ R \cdot \rhd u$. $\Box$

A key operation in equational reasoning and term rewriting is the *instantiation* of variables by terms.

Definition 2.1.26. A *substitution* is a mapping σ from $\mathcal{V}$ to $\mathcal{T}(\mathcal{F}, \mathcal{V})$ with the property that its *domain* $\mathcal{D}\mathsf{om}(\sigma) = \{x \mid \sigma(x) \neq x\}$ is finite. We write $t\sigma$ for the result of applying the substitution σ to the term t:

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

We call $t\sigma$ an *instance* of t. A substitution σ is often presented as a set of *variable bindings* $\{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$. Here $\mathcal{D}\mathsf{om}(\sigma) = \{x_1, \ldots, x_n\}$ and $t_i = x_i\sigma$ for $1 \leqslant i \leqslant n$. We denote by ε the (unique) substitution with empty domain. The set of all substitutions from $\mathcal{V}$ to $\mathcal{T}(\mathcal{F}, \mathcal{V})$ is denoted by $\Sigma(\mathcal{F}, \mathcal{V})$.

Example 2.1.27. Consider the substitutions $\sigma = \{x \mapsto x + z, y \mapsto x\}$ and $\tau = \{x \mapsto y, z \mapsto y\}$. We have $\mathcal{D}\mathsf{om}(\sigma) = \{x, y\}$ and $\mathcal{D}\mathsf{om}(\tau) = \{x, z\}$. If $t = x + (\mathsf{s}(y) + (z + x))$ then $t\sigma = (x + z) + (\mathsf{s}(x) + (z + (x + z)))$ and $t\tau = y + (\mathsf{s}(y) + (y + y))$.

Definition 2.1.28. A term s *matches* a term t if t is an instance of s.

There is a simple procedure to decide the matching problem.

Theorem 2.1.29. *The* matching problem *is decidable:*

instance: *terms s, t*

question: *does there exist a substitution σ such that $s\sigma = t$?*

Proof We start with the problem $\{s \mapsto t\}$ and apply the following transformation rules as long as possible:

$$\begin{aligned}
\{f(s_1,\ldots,s_n) \mapsto f(t_1,\ldots,t_n)\} \uplus S &\Longrightarrow \{s_1 \mapsto t_1,\ \ldots,\ s_n \mapsto t_n\} \cup S \\
\{f(s_1,\ldots,s_n) \mapsto g(t_1,\ldots,t_m)\} \uplus S &\Longrightarrow \bot \quad \text{if } f \neq g \\
\{f(s_1,\ldots,s_n) \mapsto x\} \uplus S &\Longrightarrow \bot \\
\{x \mapsto t\} \uplus S &\Longrightarrow \bot \quad \text{if } S \text{ contains } x \mapsto t' \text{ with } t \neq t'
\end{aligned}$$

Here $\uplus$ stands for disjoint union, so the selected 'binding' $s \mapsto t$ does not appear in S. Termination of this transformation process follows from the observation that in each step the number of function symbols and variables strictly decreases. A substitution σ is said to be a solution of a problem $\{s_1 \mapsto t_1,\ \ldots,\ s_n \mapsto t_n\}$ if $s_i\sigma = t_i$ for all $1 \leqslant i \leqslant n$. The problem $\bot$ has no solutions. A straightforward induction proof shows that S and T have the same solutions whenever $S \Longrightarrow^* T$. Consider now a maximal derivation $\{s \mapsto t\} \Longrightarrow^! T$. If $T = \bot$ then $\{s \mapsto t\}$ has no solutions and hence s does not match t. If $T \neq \bot$ then $T = \{x_1 \mapsto t_1,\ \ldots,\ x_n \mapsto t_n\}$ with pairwise different variables $x_1,\ldots,x_n$, for otherwise a transformation rule can be applied to T. After eliminating trivial variable bindings $x_i \mapsto x_i$ from T, we obtain a substitution that solves T and thus also S. Hence s matches t. □

Example 2.1.30. The term $x + \mathsf{s}(y + z)$ matches $\mathsf{s}(y) + \mathsf{s}((x + \mathsf{s}(\mathsf{0})) + z)$:

$$\begin{aligned}
&\{x + \mathsf{s}(y + z) \mapsto \mathsf{s}(y) + \mathsf{s}((x + \mathsf{s}(\mathsf{0})) + z)\} \\
&\quad\Longrightarrow \{x \mapsto \mathsf{s}(y),\ \mathsf{s}(y + z) \mapsto \mathsf{s}((x + \mathsf{s}(\mathsf{0})) + z)\} \\
&\quad\Longrightarrow \{x \mapsto \mathsf{s}(y),\ y + z \mapsto (x + \mathsf{s}(\mathsf{0})) + z\} \\
&\quad\Longrightarrow \{x \mapsto \mathsf{s}(y),\ y \mapsto x + \mathsf{s}(\mathsf{0}),\ z \mapsto z\}
\end{aligned}$$

The term $\mathsf{s}(x) + (x + y)$ does not match $\mathsf{s}(\mathsf{0} + x) + ((\mathsf{0} + \mathsf{0}) + x)$:

$$\begin{aligned}
&\{\mathsf{s}(x) + (x + y) \mapsto \mathsf{s}(\mathsf{0} + x) + ((\mathsf{0} + \mathsf{0}) + x)\} \\
&\quad\Longrightarrow \{\mathsf{s}(x) \mapsto \mathsf{s}(\mathsf{0} + x),\ x + y \mapsto (\mathsf{0} + \mathsf{0}) + x\} \\
&\quad\Longrightarrow \{x \mapsto \mathsf{0} + x,\ x + y \mapsto (\mathsf{0} + \mathsf{0}) + x\} \\
&\quad\Longrightarrow \{x \mapsto \mathsf{0} + x,\ x \mapsto \mathsf{0} + \mathsf{0},\ y \mapsto x\} \\
&\quad\Longrightarrow \bot
\end{aligned}$$

Definition 2.1.31. A binary relation R on terms is *closed under substitutions* if $s\sigma \; R \; t\sigma$ for all substitutions σ and terms s, t with $s \; R \; t$.

Lemma 2.1.32. *The relation $\trianglelefteq$ is closed under substitutions.*

Proof We use induction on the definition of $\trianglelefteq$. Let $s \trianglelefteq t$ and let σ be an arbitrary substitution. If $s = t$ then $s\sigma = t\sigma$ and thus also $s\sigma \trianglelefteq t\sigma$. Otherwise $t = f(t_1, \ldots, t_n)$ and $s \trianglelefteq t_i$ for some $1 \leqslant i \leqslant n$. The induction hypothesis yields $s\sigma \trianglelefteq t_i\sigma$. Since $t\sigma = f(t_1\sigma, \ldots, t_i\sigma, \ldots, t_n\sigma)$, we have $s\sigma \trianglelefteq t\sigma$ by definition. □

This section is concluded with a simple but important definition.

Definition 2.1.33. A term t is called *linear* if it does not contain multiple occurrences of the same variable.

Exercises

2.1 Consider the term $t = (\mathsf{s}(\mathsf{0}) + x) + \mathsf{s}(\mathsf{s}(\mathsf{0}))$.

a Determine $\mathcal{V}\mathsf{ar}(t)$, $\mathcal{F}\mathsf{un}(t)$, $\mathsf{root}(t)$, $|t|$, $\|t\|$, and $\mathsf{height}(t)$.

b Is t linear?

c Determine all subterms of t.

d Which subterms have multiple occurrences in t?

2.2 Show that $\mathcal{T}(\mathcal{F}) \neq \varnothing$ if and only if $\mathcal{F}$ contains a constant.

2.3 ***a*** Let a be a function symbol or variable. Give an inductive definition of $|t|_a$ and show $|t|_a = 0$ if $a \notin \mathcal{F}\mathsf{un}(t) \cup \mathcal{V}\mathsf{ar}(t)$.

b Show the following equality for every term t:

$$|t| \; = \; \sum \big\{ \, |t|_a \; \big| \; a \in \mathcal{F}\mathsf{un}(t) \cup \mathcal{V}\mathsf{ar}(t) \, \big\}$$

c Show that $|C[t]| = |C| - 1 + |t|$, for all contexts C and terms t.

2.4 Let s and t be terms. Show that the following statements are equivalent:

▷ The term s is a proper subterm of t.

▷ There exists a non-empty position $p \in \mathcal{P}\mathsf{os}(t)$ such that $t|_p = s$.

▷ There exists a non-empty context C such that $t = C[s]$.

2.5 Consider Lemma 2.1.25.

a Show that the requirement that R is closed under contexts is essential.

b Show that $R \cdot \rhd$ is a well-founded order if we additionally require that R is transitive.

2.6 Consider the term $t = (\mathsf{0} + \mathsf{s}(\mathsf{0})) + (\mathsf{s}(\mathsf{s}(\mathsf{0})) + (\mathsf{0} + \mathsf{0}))$. Which terms are denoted by the following expressions?

a $t|_{21}$

b $t[\mathsf{0} + \mathsf{s}(\mathsf{0})]_{121}$

c $(t|_2[t|_1[t|_{22}]_{21}]_{11})|_1[t|_{211}[t|_{121}]_1]_{12}$

2.7 Let t be a term.

a Show that $|t| > \mathsf{height}(t)$.

b Define $\mathcal{P}\mathsf{os}_{\mathcal{V}}(t)$ by induction on t.

c Show that $|t| = \|t\| + |\mathcal{P}\mathsf{os}_{\mathcal{V}}(t)|$. Conclude that t is ground if and only if $|t| = \|t\|$.

d Show that $|\mathcal{P}\mathsf{os}_{\mathcal{V}}(t)| \geqslant |\mathcal{V}\mathsf{ar}(t)|$. Conclude that t is linear if and only if $|\mathcal{P}\mathsf{os}_{\mathcal{V}}(t)| = |\mathcal{V}\mathsf{ar}(t)|$.

2.8 Prove that a binary relation on terms is closed under contexts if and only if it is closed under *shallow* contexts, which are contexts where the hole is an argument of the root symbol.

2.9 ***a*** Show that $<$ is a proper order on positions.

b Let p and q be positions. Show that either $p = q$, $p < q$, $p \parallel q$, or $q < p$.

c Is $\parallel$ a proper order on positions?

2.10 A set of positions P is called *prefix closed* if $p \in P$ whenever $p < q$ and $q \in P$.

a Show that $\mathcal{P}\mathsf{os}(t)$ is prefix closed, for any term t.

b Are $\mathcal{P}\mathsf{os}_{\mathcal{F}}(t)$ and $\mathcal{P}\mathsf{os}_{\mathcal{V}}(t)$ prefix closed?

2.11 ***a*** Prove that $p \parallel q$ if and only there exist positions r, r_1, r_2 and positive integers $i \neq j$ such that $p = rir_1$ and $q = rjr_2$.

b Let p and q be parallel positions. Show that $pr \parallel qr$ and $rp \parallel rq$ for all positions r.

2.12 We say that a position p is to the *left* of a position q, denoted by $p <_{\mathsf{left}} q$, if there exist positions r, r_1, r_2 and positive integers $i < j$ such that $p = rir_1$ and $q = rjr_2$.

a Show that positions p, q are parallel whenever p is to the left of q.

b Is $<_{\mathsf{left}}$ a proper order on positions?

c Show that the union of $<_{\mathsf{left}}$ and $<$ is a total order on positions.

2.13 Let s, t, and u be terms. Prove the following identities for all positions $p \in \mathcal{P}\mathsf{os}(s)$ and $q \in \mathcal{P}\mathsf{os}(t)$.

a $s[s|_p]_p = s$

b $(s[t]_p)|_{pq} = t|_q$

c $(s[t]_p)[u]_{pq} = s[t[u]_q]_p$

2.14 Let s, t, and u be terms. Prove the following identities for all parallel positions $p, q \in \mathcal{P}\mathsf{os}(s)$.

a $(s[t]_p)|_q = s|_q$

b $(s[t]_p)[u]_q = (s[u]_q)[t]_p$

2.15 Let $t = x + (y + (x + y))$. Determine $t\sigma$ and $\mathcal{D}\mathsf{om}(\sigma)$ for the following substitutions σ.

a $\{x \mapsto y\}$

b $\{x \mapsto y + x,\ y \mapsto y + y,\ z \mapsto x\}$

c $\{x \mapsto \mathsf{0} + z,\ y \mapsto \mathsf{s}(\mathsf{0}),\ z \mapsto x + x\}$

2.16 Which of the following terms match the term $t = (\mathsf{s}(x) + x) + \mathsf{s}(\mathsf{s}(\mathsf{0} + y))$?

a $\mathsf{s}(x) + y$

b $x + \mathsf{s}(y)$

c $(x + y) + x$

2.2 Algebras

In the previous section we introduced the syntax of terms. In this section we are concerned with their semantics.

Definition 2.2.1. Let $\mathcal{F}$ be a signature. An *$\mathcal{F}$-algebra* $\mathcal{A}$ is a set A equipped with operations $f_{\mathcal{A}} \colon A^n \to A$ for every n-ary function symbol $f \in \mathcal{F}$. The underlying set A is

called the *carrier* of $\mathcal{A}$ and $f_{\mathcal{A}}$ is called the *interpretation* of f.

The two $\mathcal{F}$-algebras introduced in the following example will be used to illustrate subsequent developments.

Example 2.2.2. Consider a signature $\mathcal{F}$ consisting of a constant $\mathsf{0}$, a unary function symbol s, and a binary function symbol $+$. The set $\mathbb{N}$ of natural numbers can be turned into an $\mathcal{F}$-algebra $\mathcal{A}$ by defining $\mathsf{0}_{\mathcal{A}} = 0$, $\mathsf{s}_{\mathcal{A}}$ as the successor function, and $+_{\mathcal{A}}$ as addition. The carrier of the $\mathcal{F}$-algebra $\mathcal{B}$ is the set $\{\oplus, \ominus, \otimes, \oslash\}$. Constant $\mathsf{0}$ is interpreted as $\oplus$ and the interpretations $\mathsf{s}_{\mathcal{B}}$ and $+_{\mathcal{B}}$ of the function symbols s and $+$ are given in the following tables (the first argument of $+_{\mathcal{B}}$ is written in the left column and the second argument in the top row):

$\mathsf{s}_{\mathcal{B}}$	
$\oplus$	$\ominus$
$\ominus$	$\otimes$
$\otimes$	$\oslash$
$\oslash$	$\oplus$

$+_{\mathcal{B}}$	$\oplus$	$\ominus$	$\otimes$	$\oslash$
$\oplus$	$\oplus$	$\ominus$	$\otimes$	$\oslash$
$\ominus$	$\ominus$	$\ominus$	$\oplus$	$\oplus$
$\otimes$	$\otimes$	$\oslash$	$\otimes$	$\ominus$
$\oslash$	$\oslash$	$\oplus$	$\otimes$	$\oslash$

In the following we make the notational convention that A (B, C, ...) denotes the carrier of the $\mathcal{F}$-algebra $\mathcal{A}$ ($\mathcal{B}$, $\mathcal{C}$, ...). Furthermore, if the signature $\mathcal{F}$ can be inferred from the context or is irrelevant, we often write algebra instead of $\mathcal{F}$-algebra.

Let $\mathcal{A}$ be an arbitrary algebra. Every ground term t can be interpreted in $\mathcal{A}$ by simply replacing every function symbol f in t by its interpretation $f_{\mathcal{A}}$ and evaluating the resulting expression.

Example 2.2.3. For instance, in the algebra $\mathcal{A}$ of Example 2.2.2 above the ground term $\mathsf{s}(\mathsf{0} + \mathsf{s}(\mathsf{0}))$ is interpreted as $\mathsf{s}_{\mathcal{A}}(\mathsf{0}_{\mathcal{A}} +_{\mathcal{A}} \mathsf{s}_{\mathcal{A}}(\mathsf{0}_{\mathcal{A}})) = 2$. Its interpretation in the algebra $\mathcal{B}$ is $\mathsf{s}_{\mathcal{B}}(\mathsf{0}_{\mathcal{B}} +_{\mathcal{B}} \mathsf{s}_{\mathcal{B}}(\mathsf{0}_{\mathcal{B}})) = \otimes$. This is formalized below.

Definition 2.2.4. Let $\mathcal{A}$ be an arbitrary algebra. We inductively define a mapping $[\cdot]_{\mathcal{A}}$ from the set of ground terms to A as follows: $[f(t_1, \ldots, t_n)]_{\mathcal{A}} = f_{\mathcal{A}}([t_1]_{\mathcal{A}}, \ldots, [t_n]_{\mathcal{A}})$. In particular, if t is a constant then $[t]_{\mathcal{A}} = t_{\mathcal{A}}$.

Example 2.2.5. So $[\mathsf{s}(\mathsf{0}+\mathsf{s}(\mathsf{0}))]_{\mathcal{A}} = 2$ and $[\mathsf{s}(\mathsf{0}+\mathsf{s}(\mathsf{0}))]_{\mathcal{B}} = \otimes$ in the example algebras. The interpretation of non-ground terms depends on the values that we assign to the variables. Consider for instance the term $\mathsf{s}(x + \mathsf{s}(y))$ and the example algebra $\mathcal{A}$. If we assign 2 to x and 3 to y then we get $\mathsf{s}_{\mathcal{A}}(2 +_{\mathcal{A}} \mathsf{s}_{\mathcal{A}}(3))$, which evaluates to 7. Assigning the value 49 to both x and y results in $\mathsf{s}_{\mathcal{A}}(49 +_{\mathcal{A}} \mathsf{s}_{\mathcal{A}}(49)) = 100$.

Definition 2.2.6. Let $\mathcal{A}$ be an algebra. A mapping from $\mathcal{V}$ to A is called an *assignment*. The set of all assignments (from $\mathcal{V}$ to A) is denoted by $A^{\mathcal{V}}$. Let $\alpha \in A^{\mathcal{V}}$ be an assignment.

We inductively define a mapping $[\alpha]_{\mathcal{A}}(\cdot)$ from the set of terms to A as follows:

$$[\alpha]_{\mathcal{A}}(t) = \begin{cases} \alpha(t) & \text{if } t \text{ is a variable} \\ f_{\mathcal{A}}([\alpha]_{\mathcal{A}}(t_1), \ldots, [\alpha]_{\mathcal{A}}(t_n)) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

The following result states that the values $\alpha(x)$ for $x \notin \mathcal{V}\mathsf{ar}(t)$ are irrelevant for the value $[\alpha]_{\mathcal{A}}(t)$.

Lemma 2.2.7. *Let $\mathcal{A}$ be an algebra, t a term, and α an assignment.*
1 *If β is an assignment such that $\alpha(x) = \beta(x)$ for all $x \in \mathcal{V}\mathsf{ar}(t)$ then $[\alpha]_{\mathcal{A}}(t) = [\beta]_{\mathcal{A}}(t)$.*
2 *If t is a ground term then $[\alpha]_{\mathcal{A}}(t) = [t]_{\mathcal{A}}$.*

Proof Both statements follow by a routine induction on the structure of the term t. □

Example 2.2.8. Consider the algebra $\mathcal{A}$ of Example 2.2.2. Let α be any assignment satisfying $\alpha(x) = 2$ and $\alpha(y) = 1$. We have $[\alpha]_{\mathcal{A}}(\mathsf{s}(x) + (y + \mathsf{0})) = 4$. If β is any assignment (from $\mathcal{V}$ to the carrier of the example algebra $\mathcal{B}$) satisfying $\beta(x) = \ominus$ and $\beta(y) = \oslash$ then $[\beta]_{\mathcal{B}}(\mathsf{s}(x) + (y + \mathsf{0})) = \ominus$.

In the sequel the subscript $\mathcal{A}$ in $[\cdot]_{\mathcal{A}}$ and $[\cdot]_{\mathcal{A}}(\cdot)$ is often dropped when $\mathcal{A}$ can be inferred from the context.

Definition 2.2.9. Let $\mathcal{A}$ be an algebra. The binary relation $=_{\mathcal{A}}$ on terms is defined as follows: $s =_{\mathcal{A}} t$ if $[\alpha]_{\mathcal{A}}(s) = [\alpha]_{\mathcal{A}}(t)$ for every assignment $\alpha \in A^{\mathcal{V}}$.

Example 2.2.10. Consider the example algebras $\mathcal{A}$ and $\mathcal{B}$. We have $\mathsf{s}(x) + (y + \mathsf{0}) =_{\mathcal{A}} \mathsf{s}(y + x)$ because $[\alpha]_{\mathcal{A}}(\mathsf{s}(x) + (y + \mathsf{0})) = m + n + 1 = [\alpha]_{\mathcal{A}}(\mathsf{s}(y + x))$ for any assignment α satisfying $\alpha(x) = m$ and $\alpha(y) = n$. We do not have $\mathsf{s}(x) + (y + \mathsf{0}) =_{\mathcal{B}} \mathsf{s}(y + x)$ because $[\beta]_{\mathcal{A}}(\mathsf{s}(x) + (y + \mathsf{0})) = \otimes \neq \oplus = [\beta]_{\mathcal{A}}(\mathsf{s}(y + x))$ when $\beta(x) = \ominus$ and $\beta(y) = \otimes$.

Definition 2.2.11. A *congruence relation* is an equivalence relation $\sim$ on terms such that $f(s_1, \ldots, s_n) \sim f(t_1, \ldots, t_n)$ for all n-ary function symbols f and terms $s_1, \ldots, s_n, t_1, \ldots, t_n$ with $s_i \sim t_i$ for $1 \leqslant i \leqslant n$.

Lemma 2.2.12. *Let $\mathcal{A}$ be an algebra. The relation $=_{\mathcal{A}}$ is a congruence relation.*

Proof Reflexivity, symmetry, and transitivity of $=_{\mathcal{A}}$ are obvious. Let $s = f(s_1, \ldots, s_n)$ and $f(t_1, \ldots, t_n)$ such that $s_i =_{\mathcal{A}} t_i$ for all $1 \leqslant i \leqslant n$ and let α be an arbitrary assignment. We have $[\alpha]_{\mathcal{A}}(s_i) = [\alpha]_{\mathcal{A}}(t_i)$ for all $1 \leqslant i \leqslant n$ and thus also

$$[\alpha]_{\mathcal{A}}(s) = f_{\mathcal{A}}([\alpha]_{\mathcal{A}}(s_1), \ldots, [\alpha]_{\mathcal{A}}(s_n)) = f_{\mathcal{A}}([\alpha]_{\mathcal{A}}(t_1), \ldots, [\alpha]_{\mathcal{A}}(t_n)) = [\alpha]_{\mathcal{A}}(t)$$

Hence $s =_{\mathcal{A}} t$ and therefore $=_{\mathcal{A}}$ is a congruence relation. □

Congruence relations are closed under contexts. Moreover, equivalence relations that are closed under contexts are congruence relations. In Exercise 2.19(a,b) the reader is asked to prove these easy facts. At the end of this section we prove that $=_{\mathcal{A}}$ is also closed under substitutions.

Next we define two very important algebras.

Definition 2.2.13. Let $\mathcal{F}$ be a signature. The carrier of the *term algebra* $\bar{\mathcal{T}}(\mathcal{F},\mathcal{V})$ is the set $\mathcal{T}(\mathcal{F},\mathcal{V})$ of terms. Every n-ary function symbol f is interpreted as the operation $f_{\bar{\mathcal{T}}(\mathcal{F},\mathcal{V})}$ that maps terms $t_1,\ldots,t_n$ to $f(t_1,\ldots,t_n)$. We write $\bar{\mathcal{T}}$ instead of $\bar{\mathcal{T}}(\mathcal{F},\mathcal{V})$ if the signature $\mathcal{F}$ can be inferred from the context or is irrelevant. The carrier of the *ground term algebra* $\bar{\mathcal{T}}(\mathcal{F})$ is the set $\mathcal{T}(\mathcal{F})$ of ground terms. The interpretation $f_{\bar{\mathcal{T}}(\mathcal{F})}$ of an n-ary function symbol f is defined by $f_{\bar{\mathcal{T}}(\mathcal{F})}(t_1,\ldots,t_n) = f(t_1,\ldots,t_n)$ for all ground terms $t_1,\ldots,t_n$.

It is easy to prove that the interpretation of a ground term in both the term algebra and the ground term algebra is the term itself, i.e., $[t]_{\bar{\mathcal{T}}(\mathcal{F},\mathcal{V})} = [t]_{\bar{\mathcal{T}}(\mathcal{F})} = t$ for all ground terms t.

We now explain how to compare different algebras.

Definition 2.2.14. Let $\mathcal{A}$ and $\mathcal{B}$ be $\mathcal{F}$-algebras. A *homomorphism* from $\mathcal{A}$ to $\mathcal{B}$ is a mapping ϕ from A to B such that $\phi(f_{\mathcal{A}}(a_1,\ldots,a_n)) = f_{\mathcal{B}}(\phi(a_1),\ldots,\phi(a_n))$ for all n-ary function symbols f and elements $a_1,\ldots,a_n \in A$.

Example 2.2.15. Consider the signature $\mathcal{F}$ and the algebra $\mathcal{A}$ of Example 2.2.2. We define an $\mathcal{F}$-algebra $\mathcal{C}$ as follows: $C = \{\mathsf{F},\mathsf{T}\}$, $0_{\mathcal{C}} = \mathsf{T}$, and the interpretation functions $\mathsf{s}_{\mathcal{C}}$ and $+_{\mathcal{C}}$ are given in the following tables:

$\mathsf{s}_{\mathcal{C}}$	
F	T
T	F

$+_{\mathcal{C}}$	F	T
F	T	F
T	F	T

Consider the mapping $\phi\colon \mathbb{N} \to C$ defined as $\phi(x) = \mathsf{T}$ if x is even and $\phi(x) = \mathsf{F}$ if x is odd. We have $\phi(0_{\mathcal{A}}) = \phi(0) = \mathsf{T} = 0_{\mathcal{C}}$. Furthermore, $\phi(\mathsf{s}_{\mathcal{A}}(x)) = \phi(x+1) = \mathsf{F} = \mathsf{s}_{\mathcal{C}}(\mathsf{T}) = \mathsf{s}_{\mathcal{C}}(\phi(x))$ for even x and $\phi(\mathsf{s}_{\mathcal{A}}(x)) = \phi(x+1) = \mathsf{T} = \mathsf{s}_{\mathcal{C}}(\mathsf{F}) = \mathsf{s}_{\mathcal{C}}(\phi(x))$ for odd x. By a similar case distinction we obtain $\phi(+_{\mathcal{A}}(x,y)) = +_{\mathcal{C}}(\phi(x),\phi(y))$ for all $x,y \in \mathbb{N}$. Therefore ϕ is a homomorphism from $\mathcal{A}$ to $\mathcal{C}$.

Lemma 2.2.16. *Let $\mathcal{A}$ be an $\mathcal{F}$-algebra.*

1. *The mapping $[\cdot]_{\mathcal{A}}$ is a homomorphism from $\bar{\mathcal{T}}(\mathcal{F})$ to $\mathcal{A}$.*
2. *The mapping $[\alpha]_{\mathcal{A}}(\cdot)$ is a homomorphism from $\bar{\mathcal{T}}(\mathcal{F},\mathcal{V})$ to $\mathcal{A}$ for every $\alpha \in A^{\mathcal{V}}$.*

Proof Let f be an arbitrary n-ary function symbol.

1. Let $t_1,\ldots,t_n$ be ground terms. We have

$$[f_{\bar{\mathcal{T}}(\mathcal{F})}(t_1,\ldots,t_n)]_{\mathcal{A}} = [f(t_1,\ldots,t_n)]_{\mathcal{A}} = f_{\mathcal{A}}([t_1]_{\mathcal{A}},\ldots,[t_n]_{\mathcal{A}})$$

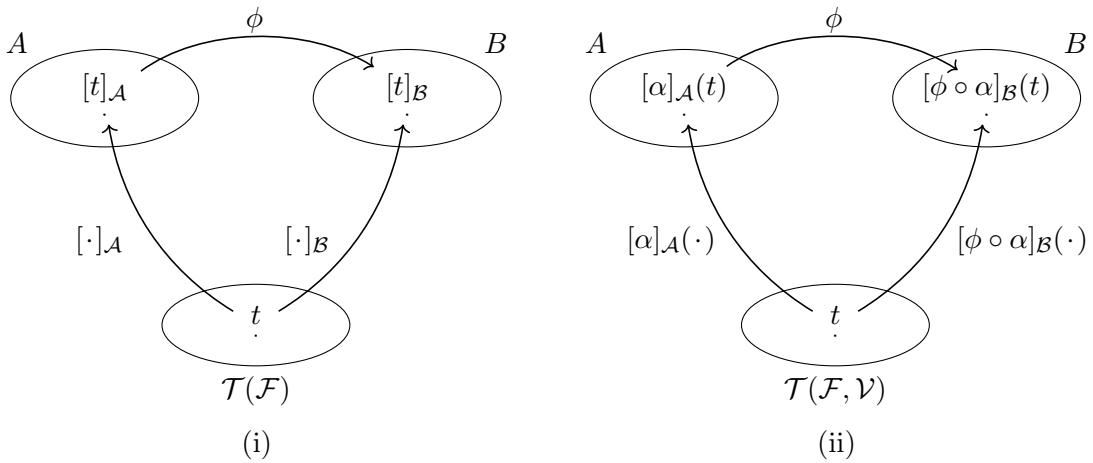


Figure 2.3: Homomorphisms preserve the interpretation of terms.

2 Let $t_1, \ldots, t_n$ be terms. We have

$$[\alpha]_{\mathcal{A}}(f_{\bar{\mathcal{T}}}(t_1, \ldots, t_n)) = [\alpha]_{\mathcal{A}}(f(t_1, \ldots, t_n)) = f_{\mathcal{A}}([\alpha]_{\mathcal{A}}(t_1), \ldots, [\alpha]_{\mathcal{A}}(t_n)) \qquad \square$$

The following result states that homomorphisms preserve the interpretation of ground terms. This property is illustrated in Figure 2.3(i).

Lemma 2.2.17. *Let $\mathcal{A}$ and $\mathcal{B}$ be algebras. If ϕ is a homomorphism from $\mathcal{A}$ to $\mathcal{B}$ then $\phi \circ [\cdot]_{\mathcal{A}} = [\cdot]_{\mathcal{B}}$.*

Proof We show $\phi([t]_{\mathcal{A}}) = [t]_{\mathcal{B}}$ by induction on the structure of ground terms t. Let $t = f(t_1, \ldots, t_n)$. We have

$$\begin{aligned} \phi([t]_{\mathcal{A}}) &= \phi(f_{\mathcal{A}}([t_1]_{\mathcal{A}}, \ldots, [t_n]_{\mathcal{A}})) = f_{\mathcal{B}}(\phi([t_1]_{\mathcal{A}}), \ldots, \phi([t_n]_{\mathcal{A}})) \\ &= f_{\mathcal{B}}([t_1]_{\mathcal{B}}, \ldots, [t_n]_{\mathcal{B}}) = [t]_{\mathcal{B}} \end{aligned}$$

The induction hypothesis is used in the third equality. $\square$

Example 2.2.18. Consider the example algebras $\mathcal{A}$ and $\mathcal{B}$. We claim that there are no homomorphisms from $\mathcal{A}$ to $\mathcal{B}$. This can be seen as follows. Consider the ground terms $\mathsf{s}(\mathsf{s}(0))$ and $\mathsf{s}(0) + \mathsf{s}(0)$. Both terms have the value 2 in $\mathcal{A}$, but in $\mathcal{B}$ their values differ: $[\mathsf{s}(\mathsf{s}(0))]_{\mathcal{B}} = \otimes$ and $[\mathsf{s}(0) + \mathsf{s}(0)]_{\mathcal{B}} = \ominus$. Any homomorphism ϕ from $\mathcal{A}$ to $\mathcal{B}$ must satisfy $\phi(2) = \otimes$ and $\phi(2) = \ominus$, which is clearly impossible.

From Lemma 2.2.17 we easily infer that there is no other homomorphism from the ground term algebra to $\mathcal{A}$ than $[\cdot]_{\mathcal{A}}$.

Lemma 2.2.19. *For every $\mathcal{F}$-algebra $\mathcal{A}$ the mapping $[\cdot]_{\mathcal{A}}$ is the only homomorphism from $\bar{\mathcal{T}}(\mathcal{F})$ to $\mathcal{A}$.*

Proof Let ϕ be a homomorphism from $\bar{\mathcal{T}}(\mathcal{F})$ to $\mathcal{A}$. Lemma 2.2.17 yields $\phi \circ [\cdot]_{\bar{\mathcal{T}}(\mathcal{F})} = [\cdot]_{\mathcal{A}}$. Since $[\cdot]_{\bar{\mathcal{T}}(\mathcal{F})}$ is the identity mapping on $\bar{\mathcal{T}}(\mathcal{F})$ we obtain $\phi = [\cdot]_{\mathcal{A}}$. So $[\cdot]_{\mathcal{A}}$ is the only homomorphism from $\bar{\mathcal{T}}(\mathcal{F})$ to $\mathcal{A}$. $\square$

The converse of Lemma 2.2.17 does not hold in general (Exercise 2.23). However, the extension of Lemma 2.2.17 to non-ground terms stated below turns out to be a complete characterization of homomorphisms. This is stated in the next lemma and illustrated in Figure 2.3(ii).

Lemma 2.2.20. *Let $\mathcal{A}$ and $\mathcal{B}$ be algebras. A mapping ϕ from A to B is a homomorphism if and only if $\phi \circ [\alpha]_{\mathcal{A}} = [\phi \circ \alpha]_{\mathcal{B}}$ for all assignments $\alpha \in A^{\mathcal{V}}$.*

Proof The direction from left to right is proved by an easy induction. For the reverse direction assume that $\phi \circ [\alpha]_{\mathcal{A}} = [\phi \circ \alpha]_{\mathcal{B}}$ for all assignments $\alpha \in A^{\mathcal{V}}$. Let f be an n-ary function symbol and let $a_1, \ldots, a_n \in A$. We have to show $\phi(f_{\mathcal{A}}(a_1, \ldots, a_n)) = f_{\mathcal{B}}(\phi(a_1), \ldots, \phi(a_n))$. Choose pairwise different variables $x_1, \ldots, x_n \in \mathcal{V}$ and let α be the assignment that maps x_i to a_i for $1 \leqslant i \leqslant n$, and all other variables to an arbitrary element in A. Note that if $n = 0$ then A cannot be empty as it contains $f_{\mathcal{A}}$. We have

$$\begin{aligned}\phi(f_{\mathcal{A}}(a_1, \ldots, a_n)) &= \phi(f_{\mathcal{A}}([\alpha]_{\mathcal{A}}(x_1), \ldots, [\alpha]_{\mathcal{A}}(x_n))) = (\phi \circ [\alpha]_{\mathcal{A}})(f(x_1, \ldots, x_n)) \\ &= [\phi \circ \alpha]_{\mathcal{B}}(f(x_1, \ldots, x_n)) = f_{\mathcal{B}}(\phi(\alpha(x_1)), \ldots, \phi(\alpha(x_n))) \\ &= f_{\mathcal{B}}(\phi(a_1), \ldots, \phi(a_n))\end{aligned}$$

□

From Lemma 2.2.20 we easily infer that there are no homomorphisms between the term algebra and $\mathcal{A}$ besides the ones identified in the second part of Lemma 2.2.16.

Lemma 2.2.21. *Let $\mathcal{A}$ be an $\mathcal{F}$-algebra. For every assignment α the mapping $[\alpha]_{\mathcal{A}}$ is the only homomorphism from $\overline{\mathcal{T}}$ to $\mathcal{A}$ that extends α.*

Proof Suppose ϕ is a homomorphism from $\overline{\mathcal{T}}$ to $\mathcal{A}$ that extends α, so $\phi(x) = \alpha(x)$ for all $x \in \mathcal{V}$. Let β be the identity assignment in $\overline{\mathcal{T}}$. Lemma 2.2.20 yields $\phi \circ [\beta]_{\overline{\mathcal{T}}} = [\phi \circ \beta]_{\mathcal{A}}$. It is not difficult to see that $[\beta]_{\overline{\mathcal{T}}}$ is the identity mapping on terms. The assignment $\phi \circ \beta$ is simply the restriction of ϕ to $\mathcal{V}$, which by assumption is the same as α. So we obtain the desired $\phi = [\alpha]_{\mathcal{A}}$. □

Since the restriction of a homomorphism $\phi\colon \overline{\mathcal{T}} \to \mathcal{A}$ to $\mathcal{V}$ is evidently an assignment, the set $A^{\mathcal{V}}$ of assignments precisely characterizes the set of homomorphisms from $\overline{\mathcal{T}}$ to $\mathcal{A}$. Writing $\hat{\phi}$ for the assignment associated with the homomorphism ϕ, we have $[\hat{\phi}]_{\mathcal{A}} = \phi$.

Lemma 2.2.22. *Let $\mathcal{A}$ and $\mathcal{B}$ be algebras. If ϕ is a homomorphism from $\overline{\mathcal{T}}$ to $\mathcal{A}$ and ψ a homomorphism from $\mathcal{A}$ to $\mathcal{B}$ then $\psi \circ \phi = [\psi \circ \hat{\phi}]_{\mathcal{B}}$.*

Proof Taking $\alpha = \hat{\phi}$ in Lemma 2.2.20 yields $\psi \circ \phi = \psi \circ [\hat{\phi}]_{\mathcal{A}} = [\psi \circ \hat{\phi}]_{\mathcal{B}}$. □

Definition 2.2.23. Let $\mathcal{A}$ be an algebra. An algebra $\mathcal{B}$ is called a *homomorphic image* of $\mathcal{A}$ if there exists a surjective homomorphism from $\mathcal{A}$ to $\mathcal{B}$. We say that $\mathcal{A}$ has *no junk* if $\mathcal{A}$ is a homomorphic image of the ground term algebra.

Intuitively, an algebra has no junk if every element of its carrier is the interpretation of some ground term.

Example 2.2.24. The example algebra $\mathcal{A}$ has no junk because for every $n \in \mathbb{N}$ there is a ground term t_n such that $[t_n]_{\mathcal{A}} = n$. For instance,

$$t_n = \underbrace{\mathsf{s}(\cdots(\mathsf{s}}_{n}(\mathsf{0}))\cdots)$$

Also the example algebra $\mathcal{B}$ has no junk: $[\mathsf{0}]_{\mathcal{B}} = \oplus$, $[\mathsf{s}(\mathsf{0})]_{\mathcal{B}} = \ominus$, $[\mathsf{s}(\mathsf{s}(\mathsf{0}))]_{\mathcal{B}} = \otimes$, and $[\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{0})))]_{\mathcal{B}} = \oslash$. The algebra $\mathcal{C}$ obtained from $\mathcal{A}$ by extending the carrier from $\mathbb{N}$ to $\mathbb{Z}$ (and the operations accordingly) has junk since there is no ground term t such that $[t]_{\mathcal{C}} \in \mathbb{Z} \setminus \mathbb{N}$.

Definition 2.2.25. An *isomorphism* is a bijective homomorphism. Two algebras $\mathcal{A}$ and $\mathcal{B}$ are *isomorphic* if there exists an isomorphism from $\mathcal{A}$ to $\mathcal{B}$.

We continue this section with the important quotient construction on algebras.

Definition 2.2.26. Let $\mathcal{A}$ be an algebra. An equivalence relation $\sim$ on its carrier A is called a *congruence* if $f_{\mathcal{A}}(a_1, \ldots, a_n) \sim f_{\mathcal{A}}(b_1, \ldots, b_n)$ for all n-ary function symbols f and elements $a_1, \ldots, a_n, b_1, \ldots, b_n \in A$ with $a_i \sim b_i$ for $i = 1, \ldots, n$.

Note the difference with the definition of congruence relation in Definition 2.2.11. The latter is defined on terms, whereas congruence relations in the above definition are defined on algebras.

Definition 2.2.27. Let $\mathcal{A}$ be an algebra equipped with a congruence $\sim$. The carrier of the *quotient algebra* $\mathcal{A}/\!\sim$ is the set $A/\!\sim$ of all equivalence classes. Every n-ary function symbol f is interpreted as the operation $f_{\mathcal{A}/\sim}$ that maps equivalence classes $c_1, \ldots, c_n$ of $A/\!\sim$ to the equivalence class $[f_{\mathcal{A}}(a_1, \ldots, a_n)]_\sim \in A/\!\sim$ where $a_i \in c_i$ for $1 \leqslant i \leqslant n$.

The above definition of $f_{\mathcal{A}/\sim}$ does not depend on the representatives $a_1, \ldots, a_n$ because $\sim$ is a congruence: If $a_i \sim b_i$ for $1 \leqslant i \leqslant n$ then $f_{\mathcal{A}}(a_1, \ldots, a_n) \sim f_{\mathcal{A}}(b_1, \ldots, b_n)$ and thus $[f_{\mathcal{A}}(a_1, \ldots, a_n)]_\sim = [f_{\mathcal{A}}(b_1, \ldots, b_n)]_\sim$. So $\mathcal{A}/\!\sim$ is a well-defined algebra.

Example 2.2.28. Consider again the example algebra $\mathcal{A}$. Define a relation $\sim$ on the carrier $\mathbb{N}$ of $\mathcal{A}$ as follows: $m \sim n$ if and only if m and n are both even or m and n are both odd, i.e., $m \equiv n \pmod 2$. This clearly defines an equivalence relation. It is a congruence on $\mathcal{A}$ as $\mathsf{s}_{\mathcal{A}}(m) = m + 1 \sim n + 1 = \mathsf{s}_{\mathcal{A}}(n)$ whenever $m \sim n$ and $m_1 +_{\mathcal{A}} m_2 = m_1 + m_2 \sim n_1 + n_2 = n_1 +_{\mathcal{A}} n_2$ whenever $m_1 \sim n_1$ and $m_2 \sim n_2$. The set $\mathbb{N}/\!\sim$ consists of two equivalence classes, one containing all even numbers and one containing all odd numbers. These will be denoted by even and odd. In the algebra $\mathcal{A}/\!\sim$ function symbol $\mathsf{0}$ is interpreted as even and the interpretations of s and $+$ are given in the following tables:

$\mathsf{s}_{\mathcal{A}/\sim}$	
even	odd
odd	even

$+_{\mathcal{A}/\sim}$	even	odd
even	even	odd
odd	odd	even

Note that the algebra $\mathcal{A}/\!\sim$ in the above example is isomorphic to the algebra $\mathcal{C}$ of Example 2.2.15.

Lemma 2.2.29. *Let $\mathcal{A}$ be an algebra equipped with a congruence $\sim$. The mapping $[\cdot]_\sim$ is a homomorphism from $\mathcal{A}$ to $\mathcal{A}/\!\sim$.*

Proof Let f be an arbitrary n-ary function symbol and $a_1, \ldots, a_n \in A$. We have

$$[f_{\mathcal{A}}(a_1, \ldots, a_n)]_\sim \;=\; f_{\mathcal{A}/\sim}([a_1]_\sim, \ldots, [a_n]_\sim)$$

by the definition of $f_{\mathcal{A}/\sim}$. □

Substitutions are assignments in the term algebra $\bar{\mathcal{T}}$ and $t\sigma$ is the result of applying the homomorphism $[\sigma]_{\bar{\mathcal{T}}}(\cdot)$ to t (Exercise 2.27). This allows us to use Lemma 2.2.22 to obtain the closure under substitutions of $=_{\mathcal{A}}$ for an arbitrary algebra $\mathcal{A}$.

Lemma 2.2.30. *Let $\mathcal{A}$ be an algebra. The relation $=_{\mathcal{A}}$ is closed under substitutions.*

Proof Suppose $s =_{\mathcal{A}} t$. Let σ be an arbitrary substitution and $\alpha \in A^{\mathcal{V}}$ an arbitrary assignment. From Lemma 2.2.22 we infer $[\alpha](s\sigma) = [[\alpha] \circ \sigma](s)$ and $[\alpha](t\sigma) = [[\alpha] \circ \sigma](t)$. Observe that $[\alpha] \circ \sigma \in A^{\mathcal{V}}$. Hence the equality of $[[\alpha] \circ \sigma](s)$ and $[[\alpha] \circ \sigma](t)$ follows from $s =_{\mathcal{A}} t$. □

Relations that are closed under contexts and substitutions play an important rule in term rewriting.

Definition 2.2.31. A binary relation on terms is called a *rewrite relation* if it is closed under contexts and substitutions.

Exercises

2.17 Consider the algebra $\mathcal{B}$ of Example 2.2.2. Construct for every $\odot \in \{\oplus, \ominus, \otimes, \oslash\}$ an assignment $\alpha_\odot$ such that $[\alpha_\odot]_{\mathcal{B}}((\mathsf{0} + x) + (y + \mathsf{s}(y))) = \odot$.

2.18 Consider the signature $\mathcal{F}$ consisting of a constant a, a binary function symbol f, and a unary function symbol g. We define an $\mathcal{F}$-algebra $\mathcal{A}$ as follows. Its carrier is the set $\{\square, \diamond, \triangle\}$. Constant a is interpreted as $\square$ and the interpretations $\mathsf{f}_{\mathcal{A}}$ and $\mathsf{g}_{\mathcal{A}}$ of the function symbols f and g are given in the following tables:

$\mathsf{f}_{\mathcal{A}}$	$\square$	$\diamond$	$\triangle$
$\square$	$\square$	$\square$	$\square$
$\diamond$	$\diamond$	$\square$	$\triangle$
$\triangle$	$\triangle$	$\square$	$\triangle$

$\mathsf{g}_{\mathcal{A}}$	
$\square$	$\diamond$
$\diamond$	$\square$
$\triangle$	$\triangle$

a Let α be any assignment satisfying $\alpha(x) = \square$ and $\alpha(y) = \triangle$. Which element is denoted by the expression $[\alpha]_{\mathcal{A}}(\mathsf{f}(\mathsf{g}(\mathsf{f}(x, y)), \mathsf{f}(y, \mathsf{g}(\mathsf{a}))))$?

b Does $\mathcal{A}$ have junk?

2.19 ***a*** Show that congruence relations on terms are closed under contexts.

b Show that equivalence relations that are closed under contexts are congruence relations.

c Show that an equivalence relation on terms is a congruence on the term algebra if and only if it is closed under contexts.

2.20 Consider the algebras $\mathcal{A}$ and $\mathcal{B}$ of Example 2.2.2.

a Show that there are no homomorphisms from $\mathcal{B}$ to $\mathcal{A}$.

b Construct an $\mathcal{F}$-algebra $\mathcal{C}$ with junk.

2.21 Let $\mathcal{F}$ be a signature. The carrier A of the $\mathcal{F}$-algebra $\mathcal{A}$ is the set $\mathbb{N}$ of natural numbers and every n-ary function symbol $f \in \mathcal{F}$ is interpreted as follows: $f_{\mathcal{A}}(m_1, \ldots, m_n) = 1 + m_1 + \cdots + m_n$ for all $m_1, \ldots, m_n \in \mathbb{N}$.

a Define the assignment $\alpha \in A^{\mathcal{V}}$ by $\alpha(x) = 1$ for all $x \in \mathcal{V}$. Show that $|t| = [\alpha]_{\mathcal{A}}(t)$ for all terms t. Conclude that the size function $|\cdot|$ is a homomorphism from $\bar{\mathcal{T}}(\mathcal{F}, \mathcal{V})$ to $\mathcal{A}$.

b Show that also $\|\cdot\|$ is a homomorphism from $\bar{\mathcal{T}}(\mathcal{F}, \mathcal{V})$ to $\mathcal{A}$. What is the corresponding assignment?

c Show that the mappings $\mathcal{V}\mathsf{ar}$, $\mathcal{F}\mathsf{un}$, root, and height defined in Definitions 2.1.7, 2.1.8, 2.1.10, and 2.1.12 are homomorphisms from $\bar{\mathcal{T}}(\mathcal{F}, \mathcal{V})$ to some $\mathcal{F}$-algebra.

2.22 Let $\mathcal{A}$ and $\mathcal{B}$ be algebras. Suppose there exists a homomorphism from $\mathcal{A}$ to $\mathcal{B}$. Which of the following inclusions hold?

a $=_{\mathcal{A}} \subseteq =_{\mathcal{B}}$

b $=_{\mathcal{B}} \subseteq =_{\mathcal{A}}$

2.23 Let $\mathcal{A}$ and $\mathcal{B}$ be algebras.

a Suppose $\mathcal{A}$ has no junk. Show that every mapping $\phi\colon A \to B$ satisfying $\phi \circ [\cdot]_{\mathcal{A}} = [\cdot]_{\mathcal{B}}$ is a homomorphism from $\mathcal{A}$ to $\mathcal{B}$.

b Show that it is essential that $\mathcal{A}$ has no junk in part (a).

2.24 ***a*** Let $\phi\colon \mathcal{A} \to \mathcal{B}$ and $\psi\colon \mathcal{B} \to \mathcal{C}$ be homomorphisms. Show that their composition $\psi \circ \phi$ is a homomorphism from $\mathcal{A}$ to $\mathcal{C}$.

b Show that $\mathcal{F}$-algebras together with homomorphisms comprise a *category*, i.e., show that

- ▷ composition of homomorphisms is associative,
- ▷ for every $\mathcal{F}$-algebra $\mathcal{A}$ there exists a homomorphism $\mathrm{id}_{\mathcal{A}}\colon \mathcal{A} \to \mathcal{A}$ such that $\phi \circ \mathrm{id}_{\mathcal{A}} = \mathrm{id}_{\mathcal{B}} \circ \phi = \phi$ for all homomorphisms $\phi\colon \mathcal{A} \to \mathcal{B}$.

Since for every $\mathcal{F}$-algebra $\mathcal{A}$ there is a unique homomorphism from $\bar{\mathcal{T}}(\mathcal{F})$ to $\mathcal{A}$, the ground term algebra $\bar{\mathcal{T}}(\mathcal{F})$ is an initial object in this category.

2.25 Does the algebra $\mathcal{B}$ of Example 2.2.2 admit any non-trivial congruences?

2.26 Let $\mathcal{A}$ be an algebra equipped with a congruence $\sim$.

a Is $\mathcal{A}/{\sim}$ a homomorphic image of $\mathcal{A}$?

b Suppose $\mathcal{A}$ has no junk. Show that $\mathcal{A}/{\sim}$ has no junk.

c Does the converse of part (b) hold?

2.27 Show that $t\sigma = [\sigma]_{\bar{\mathcal{T}}}(t)$ for all terms t and substitutions σ.

2.3 Equational Reasoning

Definition 2.3.1. An *equation* is a pair (s, t) of terms, written as $s \approx t$.

In Section 2.2 we introduced algebras to give meaning to terms. Equations are interpreted by comparing the meaning of the two constituent terms.

Example 2.3.2. Consider for instance the example algebras $\mathcal{A}$ and $\mathcal{B}$ of Example 2.2.2 and the equation $\mathsf{s}(\mathsf{0})+\mathsf{s}(\mathsf{0}) \approx \mathsf{s}(\mathsf{s}(\mathsf{0}+\mathsf{0}))$. Both terms have the same value 2 in $\mathcal{A}$. We say the equation is *valid* in $\mathcal{A}$. The equation $\mathsf{s}(\mathsf{0})+\mathsf{s}(\mathsf{0}) \approx \mathsf{s}(\mathsf{s}(\mathsf{0}+\mathsf{0}))$ is not valid in $\mathcal{B}$ because the values of the two terms differ: $[\mathsf{s}(\mathsf{0})+\mathsf{s}(\mathsf{0})]_{\mathcal{B}} = \ominus$ and $[\mathsf{s}(\mathsf{s}(\mathsf{0}+\mathsf{0}))]_{\mathcal{B}} = \otimes$. To determine the validity of equations involving non-ground terms, we have to take all possible values for the variables into consideration.

Definition 2.3.3. An equation $s \approx t$ is *valid* in an algebra $\mathcal{A}$ if $s =_{\mathcal{A}} t$. We also say that $\mathcal{A}$ is a *model* of the equation $s \approx t$.

Combining equations gives rise to equational systems.

Definition 2.3.4. An *equational system* (ES for short) is a pair $(\mathcal{F}, \mathcal{E})$ consisting of a signature $\mathcal{F}$ and a set $\mathcal{E}$ of equations between terms in $\mathcal{T}(\mathcal{F}, \mathcal{V})$.

Both $\mathcal{F}$ and $\mathcal{E}$ may be infinite. An ES $(\mathcal{F}, \mathcal{E})$ is said to be *finite* if both $\mathcal{F}$ and $\mathcal{E}$ are finite. We often present an ES as a set of equations, without making explicit its signature, assuming that the signature consists of the function symbols occurring in the equations. In the next definition we give semantics to ESs.

Definition 2.3.5. An algebra $\mathcal{A}$ is a *model* of an ES $\mathcal{E}$ if every equation in $\mathcal{E}$ is valid in $\mathcal{A}$. We write $s =_{\mathcal{E}} t$ if $s \approx t$ is valid in all models of $\mathcal{E}$.

Example 2.3.6. Consider the algebras $\mathcal{A}$ and $\mathcal{B}$ of Example 2.2.2 and the ES $\mathcal{E}$ consisting of the two equations

$$\mathsf{0} + x \approx x \qquad\qquad \mathsf{s}(x) + y \approx \mathsf{s}(x + y)$$

The equation $\mathsf{s}(x) + y \approx \mathsf{s}(x + y)$ is valid in $\mathcal{A}$ since for all natural numbers m and n we have $(m + 1) + n = (m + n) + 1$. Since the equation $\mathsf{0} + x \approx x$ is also valid in $\mathcal{A}$ (as $0 + n = n$ for all natural numbers n), $\mathcal{A}$ is a model of $\mathcal{E}$. The equation $\mathsf{s}(x) + y \approx \mathsf{s}(x + y)$ is not valid in $\mathcal{B}$: $\mathsf{s}_{\mathcal{B}}(\oplus) +_{\mathcal{B}} \ominus = \ominus \neq \otimes = \mathsf{s}_{\mathcal{B}}(\oplus +_{\mathcal{B}} \ominus)$. Hence $\mathcal{B}$ is not a model of $\mathcal{E}$. The algebra $\mathcal{C}$ of Example 2.2.15 is a model of $\mathcal{E}$ because it is a homomorphic image of $\mathcal{A}$ (cf. Exercise 2.34).

Lemma 2.3.7. *Let $\mathcal{E}$ be an* ES*. The relation $=_{\mathcal{E}}$ is the smallest equivalence and rewrite relation on terms that contains $\mathcal{E}$.*

Proof It is an immediate consequence of the definition of $=_{\mathcal{E}}$ and Lemmata 2.2.12 and 2.2.30 that $=_{\mathcal{E}}$ is an equivalence and rewrite relation on terms that contains $\mathcal{E}$. It remains to show that $=_{\mathcal{E}}$ is the smallest such relation. Let $\sim$ be an equivalence and rewrite relation on terms that contains $\mathcal{E}$. We have to show $=_{\mathcal{E}} \subseteq \sim$. To this end we construct a model of $\mathcal{E}$ in which all valid equations $s \approx t$ satisfy $s \sim t$.

Closure under contexts of $\sim$ implies that it is a congruence on $\bar{\mathcal{T}}$ (cf. Exercise 2.19(c)). Hence $\bar{\mathcal{T}}/{\sim}$ is a well-defined algebra. We show that $\bar{\mathcal{T}}/{\sim}$ is a model of $\mathcal{E}$. Let $s \approx t \in \mathcal{E}$. We have to show that $[\alpha]_{\bar{\mathcal{T}}/\sim}(s) = [\alpha]_{\bar{\mathcal{T}}/\sim}(t)$ for every assignment α. Let β be any assignment

from $\mathcal{V}$ to terms that satisfies $\beta(x) \in \alpha(x)$ for every $x \in \mathcal{V}$. Clearly $[\cdot]_\sim \circ \beta = \alpha$. We obtain $[\cdot]_\sim \circ [\beta]_{\bar{\mathcal{T}}} = [\alpha]_{\bar{\mathcal{T}}/\sim}$ from Lemma 2.2.22. Let σ be the restriction of β to $\mathcal{V}\mathsf{ar}(s \approx t)$, i.e., $\sigma(x) = \beta(x)$ if $x \in \mathcal{V}\mathsf{ar}(s \approx t)$ and $\sigma(x) = x$ otherwise. Because $\mathcal{D}\mathsf{om}(\sigma)$ is finite, σ is a substitution. We clearly have $[\beta]_{\bar{\mathcal{T}}}(s) = s\sigma$ and $[\beta]_{\bar{\mathcal{T}}}(t) = t\sigma$. Since $s\sigma \sim t\sigma$ we obtain the desired $[\alpha]_{\bar{\mathcal{T}}/\sim}(s) = [[\beta]_{\bar{\mathcal{T}}}(s)]_\sim = [s\sigma]_\sim = [t\sigma]_\sim = [[\beta]_{\bar{\mathcal{T}}}(t)]_\sim = [\alpha]_{\bar{\mathcal{T}}/\sim}(t)$.

Now let $s =_{\mathcal{E}} t$. We have to show $s \sim t$. By definition, $s =_{\mathcal{A}} t$ for all models $\mathcal{A}$ of $\mathcal{E}$. In particular $s =_{\bar{\mathcal{T}}/\sim} t$. Consider the assignment α defined by $\alpha(x) = [x]_\sim$ for every $x \in \mathcal{V}$. An easy structural induction argument shows that $[\alpha]_{\bar{\mathcal{T}}/\sim}(u) = [u]_\sim$ for all terms u. Hence $[s]_\sim = [\alpha]_{\bar{\mathcal{T}}/\sim}(s) = [\alpha]_{\bar{\mathcal{T}}/\sim}(t) = [t]_\sim$, i.e., $s \sim t$. □

We now define *equational reasoning*. The presentation below is based on the inference rules of *equational logic*.

Definition 2.3.8. Let $\mathcal{E}$ be an ES. We write $s \approx_{\mathcal{E}} t$ if the equation $s \approx t$ is derivable from the following inference rules:

[r] ***reflexivity***	$\dfrac{}{t \approx t}$	for all terms t
[s] ***symmetry***	$\dfrac{s \approx t}{t \approx s}$	for all terms s and t
[t] ***transitivity***	$\dfrac{s \approx t \quad t \approx u}{s \approx u}$	for all terms s, t, and u
[a] ***application***	$\dfrac{}{\ell\sigma \approx r\sigma}$	for all $\ell \approx r \in \mathcal{E}$ and substitutions σ
[c] ***congruence***	$\dfrac{s_1 \approx t_1 \quad \cdots \quad s_n \approx t_n}{f(s_1,\ldots,s_n) \approx f(t_1,\ldots,t_n)}$	for all n-ary function symbols f and terms $s_1,\ldots,s_n,t_1,\ldots,t_n$

Example 2.3.9. The following *proof tree* shows that $\mathsf{s}(\mathsf{s}(0) + \mathsf{s}(0)) \approx_{\mathcal{E}} \mathsf{s}(\mathsf{s}(\mathsf{s}(0)))$ with respect to the ES $\mathcal{E}$ of Example 2.3.6:

$$\dfrac{\dfrac{\dfrac{}{\mathsf{s}(0)+\mathsf{s}(0) \approx \mathsf{s}(0+\mathsf{s}(0))}[\mathsf{a}] \quad \dfrac{\dfrac{}{0+\mathsf{s}(0) \approx \mathsf{s}(0)}[\mathsf{a}]}{\mathsf{s}(0+\mathsf{s}(0)) \approx \mathsf{s}(\mathsf{s}(0))}[\mathsf{c}]}{\mathsf{s}(0)+\mathsf{s}(0) \approx \mathsf{s}(\mathsf{s}(0))}[\mathsf{t}]}{\mathsf{s}(\mathsf{s}(0)+\mathsf{s}(0)) \approx \mathsf{s}(\mathsf{s}(\mathsf{s}(0)))}[\mathsf{c}]$$

It turns out that the equation $\mathsf{s}(\mathsf{s}(0)+\mathsf{s}(0)) \approx_{\mathcal{E}} \mathsf{s}(\mathsf{s}(\mathsf{s}(0)))$ is valid in the example algebras $\mathcal{A}$ and $\mathcal{C}$: $[\mathsf{s}(\mathsf{s}(0)+\mathsf{s}(0))]_{\mathcal{A}} = 3 = [\mathsf{s}(\mathsf{s}(\mathsf{s}(0)))]_{\mathcal{A}}$ and $[\mathsf{s}(\mathsf{s}(0)+\mathsf{s}(0))]_{\mathcal{C}} = \mathsf{F} = [\mathsf{s}(\mathsf{s}(\mathsf{s}(0)))]_{\mathcal{C}}$. This is not a coincidence.

Below we show that equational reasoning is *sound*. This means that every equation $s \approx t$ deducible from an ES $\mathcal{E}$ by equational reasoning is valid in all models of $\mathcal{E}$.

Lemma 2.3.10. *Let $\mathcal{E}$ be an* ES *and s and t be terms. If $s \approx_{\mathcal{E}} t$ then $s =_{\mathcal{E}} t$.*

Proof We use induction on the structure of the proof tree of $s \approx t$. In the proof we use the fact that $=_{\mathcal{E}}$ is an equivalence and rewrite relation that contains $\mathcal{E}$ (Lemma 2.3.7). In

the base case $s \approx_{\mathcal{E}} t$ is obtained by the reflexivity or application inference rule. In the former case we have $s = t$ and thus $s =_{\mathcal{E}} t$ because $=_{\mathcal{E}}$ is reflexive. In the latter case $s = \ell\sigma$ and $t = r\sigma$ for some equation $\ell \approx r \in \mathcal{E}$ and substitution σ. Since $=_{\mathcal{E}}$ contains $\mathcal{E}$ and is closed under substitutions, $s =_{\mathcal{E}} t$. In the induction step the last proof step of $s \approx_{\mathcal{E}} t$ is an application of symmetry, transitivity, or congruence.

[s] If symmetry is applied then $t \approx s$ has a simpler proof tree and thus $t =_{\mathcal{E}} s$ by the induction hypothesis. Hence $s =_{\mathcal{E}} t$ because $=_{\mathcal{E}}$ is symmetric.

[t] If transitivity is applied then $s =_{\mathcal{E}} u$ and $u =_{\mathcal{E}} t$ for some intermediate term u follow from the induction hypothesis. We obtain $s =_{\mathcal{E}} t$ by the transitivity of $=_{\mathcal{E}}$.

[c] In this case we may write $s = f(s_1, \ldots, s_n)$ and $t = f(t_1, \ldots, t_n)$. The induction hypothesis yields $s_i =_{\mathcal{E}} t_i$ for $1 \leqslant i \leqslant n$. Since $=_{\mathcal{E}}$ is an equivalence relation closed under contexts, it is a congruence relation (Exercise 2.19(b)). Hence we obtain $s =_{\mathcal{E}} t$ as desired. □

The final result of this section, which is known as the *completeness* of equational reasoning, states that the converse of Lemma 2.3.10 also holds.

Lemma 2.3.11. *Let $\mathcal{E}$ be an* ES *and s and t be terms. If $s =_{\mathcal{E}} t$ then $s \approx_{\mathcal{E}} t$.*

Proof According to Lemma 2.3.7 it suffices to show that $\approx_{\mathcal{E}}$ is an equivalence and rewrite relation that contains $\mathcal{E}$. Reflexivity, symmetry, and transitivity of $\approx_{\mathcal{E}}$ is an immediate consequence of the corresponding inference rules in Definition 2.3.8. The application inference rule with $\sigma = \varepsilon$ shows that $\approx_{\mathcal{E}}$ contains $\mathcal{E}$. Closure under contexts is an easy consequence of the reflexivity and congruence inference rules, using induction on the structure of contexts. Finally, closure under substitutions is easily proved by induction on the definition of $\approx_{\mathcal{E}}$. □

Theorem 2.3.12. *Let $\mathcal{E}$ be an* ES*. The relations $=_{\mathcal{E}}$ and $\approx_{\mathcal{E}}$ coincide.*

Definition 2.3.13. The *equational theory* of an ES $\mathcal{E}$ is the set of all equations $s \approx t$ such that $s =_{\mathcal{E}} t$. The *validity problem* for a given ES $\mathcal{E}$ is the question whether an arbitrary equation $s \approx t$ belongs to the equational theory of $\mathcal{E}$.

Using Theorem 2.3.12, the validity problem for $\mathcal{E}$ is the question whether equations $s \approx t$ are derivable. The validity problem is undecidable in general. A concrete example of an ES with undecidable validity problem is *combinatory logic*, presented in Table 2.1. The signature of combinatory logic consist of three constants S, K, and I, and a binary function symbol $\cdot$, called *application*. So there is no algorithm that decides, given two terms s and t, whether s and t can be proved equal using the equations of Table 2.1. We have more to say about this in Section 3.4.

When the equations of an ES contain no variables, validity is decidable. This will be shown in Section 3.2 and, using more advanced rewrite techniques, again in Chapter 5.

We conclude this section with an important definition.

$$\begin{array}{|c|}\hline ((\mathsf{S}\cdot x)\cdot y)\cdot z \approx (x\cdot z)\cdot(y\cdot z)\\ (\mathsf{K}\cdot x)\cdot y \approx x\\ \mathsf{I}\cdot x \approx x\\ \hline\end{array}$$

Table 2.1: Combinatory logic.

Definition 2.3.14. An ES $\mathcal{E}$ is called *consistent* if $s \neq_{\mathcal{E}} t$ for some equation $s \approx t$.

Exercises

2.28 Determine the validity of the following equations in the algebras $\mathcal{A}$ and $\mathcal{B}$ of Example 2.2.2.

a $x + (x + x) \approx 0 + x$

b $\mathsf{s}(0) + x \approx x + \mathsf{s}(0)$

c $\mathsf{s}(x + \mathsf{s}(y)) \approx (y + \mathsf{s}(0)) + \mathsf{s}(x)$

2.29 Consider the ES $\mathcal{E}$ consisting of the three equations

$$\mathsf{f}(x) \approx x \qquad \mathsf{f}(\mathsf{f}(\mathsf{a})) \approx \mathsf{g}(x,x) \qquad \mathsf{g}(x,\mathsf{f}(x)) \approx \mathsf{b}$$

Which of the following equations belong to the equational theory of $\mathcal{E}$?

a $\mathsf{a} \approx \mathsf{b}$

b $\mathsf{g}(x,y) \approx \mathsf{g}(y,x)$

c $\mathsf{g}(\mathsf{f}(\mathsf{a}),\mathsf{a}) \approx \mathsf{f}(\mathsf{b})$

2.30 Show that the symmetry and transitivity inference rules of Definition 2.3.8 can be replaced by the single inference rule

$$\frac{s \approx t \quad u \approx t}{s \approx u}$$

without affecting the induced relation $\approx_{\mathcal{E}}$.

2.31 Consider the ES $\mathcal{E}$ consisting of the equations

$$\begin{array}{lll}\mathsf{T}(\mathsf{C}(\mathsf{A}(\mathsf{T}(x)))) \approx \mathsf{T}(x) & \mathsf{G}(\mathsf{A}(\mathsf{G}(x))) \approx \mathsf{A}(\mathsf{G}(x)) & \mathsf{C}(\mathsf{T}(\mathsf{C}(x))) \approx \mathsf{T}(\mathsf{C}(x))\\ \mathsf{A}(\mathsf{G}(\mathsf{T}(\mathsf{A}(x)))) \approx \mathsf{A}(x) & \mathsf{T}(\mathsf{A}(\mathsf{T}(x))) \approx \mathsf{C}(\mathsf{T}(x)) & \end{array}$$

Show that

$$\mathsf{T}(\mathsf{A}(\mathsf{G}(\mathsf{C}(\mathsf{T}(\mathsf{A}(\mathsf{G}(\mathsf{C}(\mathsf{T}(\mathsf{A}(\mathsf{G}(\mathsf{C}(\mathsf{T}(x)))))))))))))) \not\approx_{\mathcal{E}} \mathsf{C}(\mathsf{T}(\mathsf{G}(\mathsf{C}(\mathsf{T}(\mathsf{A}(\mathsf{C}(\mathsf{T}(\mathsf{G}(\mathsf{A}(\mathsf{C}(\mathsf{T}(x))))))))))))$$

by constructing a suitable model of $\mathcal{E}$.

2.32 Consider the ES $\mathcal{E}$ consisting of the three equations

$$\mathsf{e}\cdot x \approx x \qquad x^{-}\cdot x \approx \mathsf{e} \qquad (x\cdot y)\cdot z \approx x\cdot(y\cdot z)$$

specifying group theory (which means that its models are precisely the mathematical groups). We assume that the inverse operation $^{-}$ (written in postfix notation) binds stronger than the multiplication operation $\cdot$ (written in infix notation).

a Show that $\mathsf{e}^{-} \approx_{\mathcal{E}} \mathsf{e}$.

b Construct a non-Abelian group, i.e., construct a model $\mathcal{A}$ of $\mathcal{E}$ in which the equation $x\cdot y \approx y\cdot x$ is not valid.

$x + y \approx y + x$	$x \times y \approx y \times x$	$1 \uparrow x \approx 1$
$(x + y) + z \approx x + (y + z)$	$(x \times y) \times z \approx x \times (y \times z)$	$x \uparrow 1 \approx x$
	$x \times 1 \approx x$	$x \uparrow (y + z) \approx x \uparrow y \times x \uparrow z$
	$x \times (y + z) \approx x \times y + x \times z$	$(x \times y) \uparrow z \approx x \uparrow z \times y \uparrow z$
		$(x \uparrow y) \uparrow z \approx x \uparrow (y \times z)$

Table 2.2: High School Identities.

2.33 Let $\mathcal{E}$ be an ES and s and t be terms. Show that $s =_{(\bar{\mathcal{T}}/=_\mathcal{E})} t$ if and only if $s =_\mathcal{E} t$.

2.34 ***a*** Let ϕ be a homomorphism between two algebras $\mathcal{A}$ and $\mathcal{B}$. Suppose $s \approx t$ is an equation between ground terms s and t that is valid in $\mathcal{A}$. Prove that $s \approx t$ is valid in $\mathcal{B}$.

b Show that the restriction to ground terms in part (a) is essential.

c Show that the restriction can be dropped in case $\mathcal{B}$ is a homomorphic image of $\mathcal{A}$.

d Let $\mathcal{A}$ and $\mathcal{B}$ be isomorphic algebras. Show that $s =_\mathcal{A} t$ if and only if $s =_\mathcal{B} t$, for all terms s and t.

2.35 Let $\mathcal{E}$ be an ES. Show that the following statements are equivalent:

- ▷ $\mathcal{E}$ is inconsistent,
- ▷ $x =_\mathcal{E} y$ for some variables $x \neq y$,
- ▷ $x =_\mathcal{E} y$ for all variables x and y,
- ▷ $\mathcal{E}$ admits no model that has more than one element.

2.36 Consider the ES $\mathcal{HSI}$ consisting of the eleven equations about addition, multiplication, and exponentiation on positive integers, presented in Table 2.2. We assume that exponentiation ($\uparrow$) binds stronger than multiplication ($\times$), which binds stronger than addition ($+$). The equation

$$(A \uparrow x + B \uparrow x) \uparrow y \times (C \uparrow y + D \uparrow y) \uparrow x \;\approx\; (A \uparrow y + B \uparrow y) \uparrow x \times (C \uparrow x + D \uparrow x) \uparrow y$$

with $A = x + 1$, $B = x \uparrow 2 + (x + 1)$, $C = x \uparrow 3 + 1$, and $D = x \uparrow 4 + (x \uparrow 2 + 1)$ is known as *Wilkie's identity*. Here 2 stands for $1 + 1$, 3 for $1 + (1 + 1)$, and 4 for $(1 + 1) + (1 + 1)$.

a Prove that the equation $(x + 1) \uparrow 2 \approx x \uparrow 2 + (2 \times x + 1)$ is valid in $\mathcal{HSI}$.

b Prove that Wilkie's identity is true for all positive integers x and y.

c Construct a model for $\mathcal{HSI}$ which violates Wilkie's identity.

2.4 Substitutions

Definition 2.4.1. Let σ be a substitution. The variables *introduced* by σ is the union of the sets $\mathcal{V}\mathsf{ar}(\sigma(x))$ for all $x \in \mathcal{D}\mathsf{om}(\sigma)$ and denoted by $\mathcal{I}(\sigma)$. The *composition* of two substitutions σ and τ is defined as $[\tau]_{\bar{\mathcal{T}}} \circ \sigma$ and denoted by $\sigma\tau$.

One easily checks that $\mathcal{D}\mathsf{om}(\sigma\tau)$ is finite, so the composition of substitutions is again a substitution. If $\sigma = \{x_i \mapsto s_i \mid 1 \leqslant i \leqslant n\}$ and $\tau = \{y_j \mapsto t_j \mid 1 \leqslant j \leqslant m\}$ then

$$\{x_1 \mapsto s_1\tau, \ldots, x_n \mapsto s_n\tau\} \cup \{y_j \mapsto t_j \mid 1 \leqslant j \leqslant m \text{ and } y_j \notin \{x_1, \ldots, x_n\}\}$$

provides a more explicit definition of the composition of σ and τ.

Example 2.4.2. Consider the substitutions $\sigma = \{x \mapsto x + z, y \mapsto x\}$ and $\tau = \{x \mapsto y, z \mapsto y\}$. We have $\mathcal{I}(\sigma) = \{x, z\}$, $\mathcal{I}(\tau) = \{y\}$, $\sigma\tau = \{x \mapsto y + y, z \mapsto y\}$ and $\tau\sigma = \{y \mapsto x, z \mapsto x\}$. Since $\sigma\tau \neq \tau\sigma$, composition of substitutions is not commutative.

The following lemma justifies the terminology *composition*.

Lemma 2.4.3. *If σ and τ are substitutions then $t(\sigma\tau) = (t\sigma)\tau$ for all terms t.*

Proof We have

$$t(\sigma\tau) = [\sigma\tau]_{\bar{\mathcal{T}}}(t) = [[\tau]_{\bar{\mathcal{T}}} \circ \sigma]_{\bar{\mathcal{T}}}(t) = ([\tau]_{\bar{\mathcal{T}}} \circ [\sigma]_{\bar{\mathcal{T}}})(t) = [\tau]_{\bar{\mathcal{T}}}([\sigma]_{\bar{\mathcal{T}}}(t)) = (t\sigma)\tau$$

by applying Lemma 2.2.22 in the third step. □

Definition 2.4.4. Let σ and τ be substitutions and $V \subseteq \mathcal{V}$. We write $\sigma = \tau\ [V]$ if $\sigma(x) = \tau(x)$ for all variables $x \in V$.

The next result states that composition of substitutions is associative. So there is no need to write parentheses in expressions like $\rho\sigma\tau$.

Lemma 2.4.5. *If ρ, σ, and τ are substitutions then $\rho(\sigma\tau) = (\rho\sigma)\tau$.*

Proof We have

$$\begin{aligned}\rho(\sigma\tau) &= [\sigma\tau]_{\bar{\mathcal{T}}} \circ \rho = [[\tau]_{\bar{\mathcal{T}}} \circ \sigma]_{\bar{\mathcal{T}}} \circ \rho = ([\tau]_{\bar{\mathcal{T}}} \circ [\sigma]_{\bar{\mathcal{T}}}) \circ \rho = [\tau]_{\bar{\mathcal{T}}} \circ ([\sigma]_{\bar{\mathcal{T}}} \circ \rho) \\ &= [\tau]_{\bar{\mathcal{T}}} \circ \rho\sigma = (\rho\sigma)\tau\end{aligned}$$

by applying Lemma 2.2.22 in the third step. □

Definition 2.4.6. Let s and t be terms. We write $s \trianglelefteq t$ and we say that s *subsumes* t if t is an instance of s. The relation $\trianglelefteq$ is called *subsumption*.

Example 2.4.7. The term $x + y$ subsumes both $x + x$ and $y + x$. The term $x + x$ does not subsume $x + y$.

Lemma 2.4.8. *Subsumption is a preorder on terms.*

Proof We have to show that $\trianglelefteq$ is a reflexive and transitive relation on terms. Reflexivity is obvious as $t = t\varepsilon$ for every term t. Suppose $s \trianglelefteq t$ and $t \trianglelefteq u$. By definition there exist substitutions σ and τ such that $s\sigma = t$ and $t\tau = u$. Using Lemma 2.4.3 we obtain $s(\sigma\tau) = (s\sigma)\tau = u$, hence $s \trianglelefteq u$. □

Subsumption is not a partial order because antisymmetry does not hold. For instance, we have both $x \trianglelefteq y$ and $y \trianglelefteq x$ for different variables x and y.

Definition 2.4.9. We write $s \lessdot t$ if $s \leq t$ but not $t \leq s$ and $s \doteq t$ if both $s \leq t$ and $t \leq s$. The relation $\doteq$ is called *literal similarity*.

Note that $\lessdot$ is the proper order associated with $\leq$ and $\doteq$ its induced equivalence relation (Lemma A.1.21).

Example 2.4.10. We have $x + y \lessdot x + x$ but not $x + y \lessdot y + x$.

We proceed by giving an alternative characterization of literal similarity.

Definition 2.4.11. A *variable substitution* is a substitution from $\mathcal{V}$ to $\mathcal{V}$. A *renaming* is a bijective variable substitution. A term s is a *variant* of a term t if $s = t\sigma$ for some renaming σ.

Example 2.4.12. The substitution $\sigma = \{x \mapsto y, y \mapsto z, z \mapsto x\}$ is a renaming. Its inverse σ^{-1} is the renaming $\{x \mapsto z, y \mapsto x, z \mapsto y\}$. The term $y + (y + x)$ is a variant of $x + (x + z)$ since $x + (x + z)\sigma = y + (y + x)$.

If a term s is a variant of a term t, say $s = t\sigma$ with renaming σ, then t is also a variant of s because $t = t\sigma\sigma^{-1} = s\sigma^{-1}$. Hence we can say that s and t are variants. According to Lemma 2.4.15 below this is just another way of saying that s and t are literally similar. The proof relies on some preliminary results. The first one states that any bijection between finite sets of variables can be extended to a renaming.

Lemma 2.4.13. *For every bijection f between finite subsets A and B of $\mathcal{V}$ there exists a renaming ρ such that $\rho(x) = f(x)$ for all $x \in A$ and $\rho(x) = x$ for all $x \notin A \cup B$.*

The reader is asked in Exercise 2.38(d) to prove this result. The second property of ρ will be used in Section 2.5. The next result gives an easy sufficient condition for terms to be variants.

Lemma 2.4.14. *If $s = t\sigma$ with σ a variable substitution and $|\mathcal{V}\mathsf{ar}(s)| = |\mathcal{V}\mathsf{ar}(t)|$ then s and t are variants.*

Proof The substitution σ is a bijection from $\mathcal{V}\mathsf{ar}(t)$ to $\mathcal{V}\mathsf{ar}(s)$. According to Lemma 2.4.13 there exists a renaming ρ such that $\rho(x) = \sigma(x)$ for all $x \in \mathcal{V}\mathsf{ar}(t)$. Clearly $t\rho = t\sigma = s$. Hence s and t are variants. □

Lemma 2.4.15. *Two terms s and t are variants if and only if $s \doteq t$.*

Proof

$\Longrightarrow$ If s and t are variants then there exist renamings σ and τ such that $s = t\sigma$ and $t = s\tau$. Hence $s \geq t$ and $t \geq s$, and thus $s \doteq t$ by definition.

$\Longleftarrow$ We have $s \geq t$ and $t \geq s$. So there exist substitutions σ and τ such that $s\sigma = t$ and $t\tau = s$. We may assume that $\mathcal{D}\mathsf{om}(\sigma) \subseteq \mathcal{V}\mathsf{ar}(s)$. We have $s\sigma\tau = t\tau = s$. So $\sigma\tau = \varepsilon\ [\mathcal{V}\mathsf{ar}(s)]$. This implies that σ is a variable substitution and $|\mathcal{V}\mathsf{ar}(t)| \leqslant |\mathcal{V}\mathsf{ar}(s)|$. Applying the same reasoning to $t\tau = s$ yields $|\mathcal{V}\mathsf{ar}(s)| \leqslant |\mathcal{V}\mathsf{ar}(t)|$. Hence $|\mathcal{V}\mathsf{ar}(s)| = |\mathcal{V}\mathsf{ar}(t)|$. It follows from Lemma 2.4.14 that s and t are variants. □

Theorem 2.4.16. *The relation $\succcurlyeq$ is a well-founded order on terms.*

Proof First we show that $\succcurlyeq$ is a proper order. Irreflexivity is obvious. If $s \succcurlyeq t$ and $t \succcurlyeq u$ then $s \trianglerighteq t$ and $t \trianglerighteq u$ and thus $s \trianglerighteq u$ because $\trianglerighteq$ is transitive by Lemma 2.4.8. Now, if $u \trianglerighteq s$ would hold then $t \trianglerighteq s$ by transitivity of $\trianglerighteq$, contradicting $s \succcurlyeq t$. So u is not an instance of s and thus $s \succcurlyeq u$, proving transitivity. Next we show well-foundedness. To this end we show $(|s|, |s| - |\mathcal{V}\mathsf{ar}(s)|)\ (>,>)_{\mathsf{lex}}\ (|t|, |t| - |\mathcal{V}\mathsf{ar}(t)|)$ whenever $s \succcurlyeq t$. So let $s \succcurlyeq t$. We have $s = t\sigma$ for some substitution σ and thus $|s| > |t|$ or $|s| = |t|$. In the former case the result is immediate. If $|s| = |t|$ then σ must be a variable substitution. We have

$$\mathcal{V}\mathsf{ar}(s) \;=\; \bigcup_{x \in \mathcal{V}\mathsf{ar}(t)} \mathcal{V}\mathsf{ar}(\sigma(x)) \;=\; \{\sigma(x) \mid x \in \mathcal{V}\mathsf{ar}(t)\}$$

and thus $|\mathcal{V}\mathsf{ar}(s)| \leqslant |\mathcal{V}\mathsf{ar}(t)|$. Since $|\mathcal{V}\mathsf{ar}(s)| \neq |\mathcal{V}\mathsf{ar}(t)|$ due to Lemma 2.4.14, $|\mathcal{V}\mathsf{ar}(s)| < |\mathcal{V}\mathsf{ar}(t)|$. Therefore $|s| - |\mathcal{V}\mathsf{ar}(s)| = |t| - |\mathcal{V}\mathsf{ar}(s)| > |t| - |\mathcal{V}\mathsf{ar}(t)|$. Since $|u| - |\mathcal{V}\mathsf{ar}(u)| \geqslant 0$ for all terms u, the well-foundedness of $\succcurlyeq$ is a consequence of Theorem A.2.9. □

The measure that is used in the well-foundedness proof is illustrated in the following example.

Example 2.4.17. Consider the terms

$$\begin{array}{lll} t_1 = \mathsf{f}(\mathsf{g}(x), \mathsf{f}(y,y)) & t_3 = \mathsf{f}(y, \mathsf{f}(z,x)) & t_5 = \mathsf{f}(x,y) \\ t_2 = \mathsf{f}(\mathsf{g}(x), \mathsf{f}(y,z)) & t_4 = \mathsf{f}(x, \mathsf{f}(y,z)) & t_6 = x \end{array}$$

We have $t_1 \succcurlyeq t_2 \succcurlyeq t_3 \doteq t_4 \succcurlyeq t_5 \succcurlyeq t_6$. Below we list $|t_i|$ and $|t_i| - |\mathcal{V}\mathsf{ar}(t_i)|$ for $1 \leqslant i \leqslant 6$:

i	1	2	3	4	5	6
$\lvert t_i\rvert$	6	6	5	5	3	1
$\lvert t_i\rvert - \lvert\mathcal{V}\mathsf{ar}(t_i)\rvert$	4	3	2	2	1	0

Subsumption is a special case of *encompassment*, defined below.

Definition 2.4.18. Let s and t be terms. We write $s \trianglerighteq t$ and we say that s *encompasses* t if $s = C[t\sigma]$ for some context C and substitution σ.

Since $C[t\sigma] \trianglerighteq t\sigma \trianglerighteq t$, encompassment is the composition of $\trianglerighteq$ and $\trianglerighteq$.

Example 2.4.19. The term $s = \mathsf{s}(0{+}\mathsf{s}(x))$ encompasses the term $t = 0{+}x$ since $s = C[t\sigma]$ for $C = \mathsf{s}(\Box)$ and $\sigma = \{x \mapsto \mathsf{s}(x)\}$. Note that t is neither a subterm of nor subsumes s. The term $u = \mathsf{s}(\mathsf{s}(0) + \mathsf{s}(0))$ encompasses neither s nor t.

Lemma 2.4.20. *Encompassment is a preorder on terms.*

Proof Reflexivity of encompassment follows directly from the reflexivity of $\trianglerighteq$ and $\trianglerighteq$. For transitivity it suffices to show that $\trianglerighteq \cdot \trianglerighteq \cdot \trianglerighteq \cdot \trianglerighteq$ is included in $\trianglerighteq \cdot \trianglerighteq$. We claim that $\trianglerighteq \cdot \trianglerighteq \subseteq \trianglerighteq \cdot \trianglerighteq$. Suppose $s \trianglerighteq t \trianglerighteq u$. From $s \trianglerighteq t$ we infer the existence of a substitution σ

such that $s = t\sigma$. Because $\trianglerighteq$ is closed under substitutions, we obtain $s = t\sigma \trianglerighteq u\sigma \geq u$. This proves the claim. Now $\trianglerighteq \cdot \geq \cdot \trianglerighteq \cdot \geq \subseteq \trianglerighteq \cdot \trianglerighteq \cdot \geq \cdot \geq \subseteq \trianglerighteq \cdot \geq$. The last inclusion follows from the transitivity of $\trianglerighteq$ and $\geq$. □

The associated equivalence relation is simply literal similarity.

Lemma 2.4.21. *The intersection of encompassment and its converse is literal similarity.*

Proof We have to show $\trianglerighteq \cap \trianglelefteq = \doteq$. Clearly $\trianglerighteq = \trianglerighteq \cdot \geq = (\triangleright \cup =) \cdot \geq = \triangleright \cdot \geq \cup \geq$. We already know that $|s| \geqslant |t|$ whenever $s \geq t$. Because $s \triangleright t$ implies $|s| > |t|$, it follows that $|s| > |t|$ whenever $s \triangleright \cdot \geq t$. Now suppose that $s \trianglerighteq t$ and $t \trianglelefteq s$. This is only possible if $|s| = |t|$. Hence we must have $s \geq t$ and $t \geq s$, i.e., $s \doteq t$. □

So strict encompassment ($\triangleright$) is the difference of encompassment and literal similarity.

Example 2.4.22. The term $\mathsf{s}(\mathsf{s}(x)+y)$ strictly encompasses $y+x$, $\mathsf{s}(x)+y$, and $\mathsf{s}(x+y)$. It does not (strictly) encompass $\mathsf{s}(x+x)$.

Theorem 2.4.23. *Strict encompassment is a well-founded order on terms.*

Proof The strict part of any preorder is a proper order, so transitivity and reflexivity of $\trianglerighteq$ follow from Lemma 2.4.20. Using the observations made in the proof of the preceding lemma, it is easily shown that $\trianglerighteq = \triangleright \cdot \geq \cup \gtrdot$. Moreover, both $\triangleright \cdot \geq$ and $\gtrdot$ (Theorem 2.4.16) are well-founded relations on terms. Hence, according to Corollary 1.4.19, it suffices to show that $\triangleright \cdot \geq$ quasi-commutes over $\gtrdot$. So suppose $s \gtrdot \cdot \triangleright \cdot \geq t$. It is not difficult to prove that $\gtrdot \cdot \triangleright \subseteq \triangleright \cdot \geq$. Hence $s \triangleright \cdot \geq \cdot \geq t$, which can be simplified to $s \triangleright \cdot \geq t$. This concludes the proof. □

The final result in this section can be viewed as an extension of Lemma 2.1.25 for rewrite relations. The proof is left to the reader (Exercise 2.41).

Lemma 2.4.24. *If R is a well-founded rewrite relation then $R \cup \trianglerighteq$ is well-founded.*

Exercises

2.37 ***a*** Show that $\mathcal{D}\mathsf{om}(\sigma\tau) \subseteq \mathcal{D}\mathsf{om}(\sigma) \cup \mathcal{D}\mathsf{om}(\tau)$ for all substitutions σ and τ.

b Does the reverse inclusion also hold?

2.38 ***a*** Which of the following substitutions are renamings?

- $\triangleright$ ε
- $\triangleright$ $\{x \mapsto \mathsf{s}(x)\}$
- $\triangleright$ $\{x \mapsto y\}$
- $\triangleright$ $\{x \mapsto y, y \mapsto x\}$

b Prove that the composition of renamings is again a renaming.

c Show that a variable substitution σ is a renaming if and only if $\mathcal{D}\mathsf{om}(\sigma) = \mathcal{I}(\sigma)$.

d Prove Lemma 2.4.13.

2.39 ***a*** Consider the terms $s = \mathsf{s}(x) + y$, $t = \mathsf{s}(x)$, and $u = \mathsf{s}(\mathsf{s}(x) + x)$. Complete the following tables:

$\rhd$	s	t	u
s		✓	
t	×		
u			

$\gtrdot$	s	t	u
s			
t	×		
u			

$\trianglerighteq$	s	t	u
s			
t	×		
u			

b Which of the following terms are encompassed by the term $(x \times \mathsf{s}(y)) + x$?

- $\rhd$ $x + y$
- $\rhd$ $y \times x$
- $\rhd$ $x + x$

2.40 ***a*** Is the relation $\gtrdot$ closed under contexts?

b Is it closed under substitutions?

c Repeat parts (a) and (b) for strict encompassment.

2.41 Let R be a rewrite relation.

a Prove the inclusion $\trianglerighteq \cdot R \subseteq R \cdot \trianglerighteq$.

b Prove Lemma 2.4.24.

c Prove that $R \cup \trianglerighteq$ quasi-commutes over $\trianglerighteq$.

d Let R be well-founded. Prove that $(R \cup \trianglerighteq) \,/\, \trianglerighteq$ is well-founded.

2.5 Unification

Equation solving is an important topic in mathematics. When restricting to the term algebra, we speak of *unification*. In this section we present the basic theory of unification.

Definition 2.5.1. Let $\mathcal{A}$ be an algebra. An equation $s \approx t$ between terms s and t is *satisfiable* in $\mathcal{A}$ if there exists an assignment $\alpha \in A^{\mathcal{V}}$ such that $[\alpha]_{\mathcal{A}}(s) = [\alpha]_{\mathcal{A}}(t)$. Such an assignment is called a *solution* of the equation $s \approx t$.

Example 2.5.2. Consider again the example algebras $\mathcal{A}$ and $\mathcal{B}$ of Example 2.2.2. The equation $\mathsf{s}(x) + y \approx x + x$ is satisfiable in $\mathcal{A}$ because the assignment α with $\alpha(x) = 1$ and $\alpha(y) = 0$ makes the terms $\mathsf{s}(x) + y$ and $x + x$ equal: $[\alpha]_{\mathcal{A}}(\mathsf{s}(x) + y) = 2 = [\alpha]_{\mathcal{A}}(x + x)$. The equation $\mathsf{s}(x) + y \approx x + x$ is also satisfiable in $\mathcal{B}$ as $[\beta]_{\mathcal{B}}(\mathsf{s}(x) + y) = \oplus = [\beta]_{\mathcal{B}}(x + x)$ for the assignment β with $\beta(x) = \oplus$ and $\beta(y) = \otimes$. The equation $x + x \approx \mathsf{s}(x)$ is satisfiable in $\mathcal{A}$ but not in $\mathcal{B}$. The equation $\mathsf{s}(x) + \mathsf{s}(x) \approx \mathsf{s}(x)$ is satisfiable in $\mathcal{B}$ but not in $\mathcal{A}$.

Definition 2.5.3. Satisfiability in the term algebra is called *unifiability*. If an equation $s \approx t$ is unifiable then we say also that s and t are unifiable. Solutions in the term algebra are called *unifiers*.

Example 2.5.4. The equation $x + (\mathsf{s}(y) + y) \approx \mathsf{s}(z) + z$ is unifiable. The substitution $\sigma = \{x \mapsto \mathsf{s}(\mathsf{s}(y) + y),\ z \mapsto \mathsf{s}(y) + y\}$ is a unifier since

$$(x + (\mathsf{s}(y) + y))\sigma \;=\; \mathsf{s}(\mathsf{s}(y) + y) + (\mathsf{s}(y) + y) \;=\; (\mathsf{s}(z) + z)\sigma$$

Another unifier is $\tau = \{x \mapsto \mathsf{s}(\mathsf{s}(0)+0),\ y \mapsto 0,\ z \mapsto \mathsf{s}(0)+0\}$. The equation $x+x \approx y+\mathsf{s}(y)$ is not unifiable.

Subsumption and literal similarity easily generalize to substitutions.

Definition 2.5.5. A substitution σ is *at least as general* as a substitution τ, denoted by $\sigma \leq \tau$, if there exists a substitution ρ such that $\sigma\rho = \tau$. This relation $\leq$ is also called *subsumption*.

Example 2.5.6. The substitution σ from Example 2.5.4 is at least as general τ because $\sigma\{y \mapsto 0\} = \tau$.

Lemma 2.5.7. *Subsumption is a preorder on substitutions.*

Proof We show that $\leq$ is reflexive and transitive. The former follows from $\sigma\epsilon = \sigma$. For the latter, suppose $\sigma \leq \tau$ and $\tau \leq \rho$. So there exist substitutions σ_1 and τ_1 such that $\sigma\sigma_1 = \tau$ and $\tau\tau_1 = \rho$. Hence $\sigma\sigma_1\tau_1 = \tau\tau_1 = \rho$ and thus $\sigma \leq \rho$ as desired. □

The induced equivalence relation on substitutions, *literal similarity*, is again denoted by $\doteq$. A substitution σ is a *variant* of a substitution τ if $\sigma = \tau\rho$ for some renaming ρ. It is easy to see that τ is a variant of σ whenever σ is a variant of τ.

Example 2.5.8. The substitutions $\{x \mapsto \mathsf{s}(y),\ y \mapsto z\}$ and $\{x \mapsto \mathsf{s}(z),\ z \mapsto y\}$ are variants, the substitutions $\{x \mapsto \mathsf{s}(y)\}$ and $\{x \mapsto \mathsf{s}(z)\}$ are not.

Lemma 2.5.9. *Two substitutions σ and τ are variants if and only if $\sigma \doteq \tau$.*

Proof

$\Longrightarrow$ If σ and τ are variants then there exist renamings ρ and ρ' such that $\sigma = \tau\rho$ and $\tau = \sigma\rho'$. Hence $\tau \leq \sigma$ and $\sigma \leq \tau$, and thus $\sigma \doteq \tau$ by definition.

$\Longleftarrow$ If $\sigma \doteq \tau$ then there exist substitutions σ' and τ' such that $\sigma\sigma' = \tau$ and $\tau\tau' = \sigma$. Let $D = \mathcal{D}\mathsf{om}(\sigma) \cup \mathcal{D}\mathsf{om}(\tau)$. We have $x\sigma' = x\sigma\sigma' = x\tau = x$ and $x\tau' = x\tau\tau' = x\sigma = x$ for all $x \in \mathcal{V} \setminus D$. Hence $\sigma = \sigma' = \tau = \tau' = \varepsilon\ [\mathcal{V} \setminus D]$. We write $\mathcal{I}_D(\sigma)$ for the union of $\mathcal{V}\mathsf{ar}(\sigma(x))$ for all $x \in D$, and similarly for $\mathcal{I}_D(\tau)$. From $\sigma\sigma' = \tau\ [D]$ and $\tau\tau' = \sigma\ [D]$ we obtain $\sigma'(\mathcal{I}_D(\sigma)) = \mathcal{I}_D(\tau)$ and $\tau'(\mathcal{I}_D(\tau)) = \mathcal{I}_D(\sigma)$ and thus $\tau'(\sigma'(\mathcal{I}_D(\sigma))) = \mathcal{I}_D(\sigma)$. Hence σ' is injective on $\mathcal{I}_D(\sigma)$ and thus a bijection from $\mathcal{I}_D(\sigma)$ to $\mathcal{I}_D(\tau)$. According to Lemma 2.4.13 there exists a renaming ρ such that $\rho(x) = \sigma'(x)$ for all $x \in \mathcal{I}_D(\sigma)$ and $\rho(x) = x$ for all $x \notin \mathcal{I}_D(\sigma) \cup \mathcal{I}_D(\tau)$. We claim that $\sigma\rho = \tau$. Let $x \in \mathcal{V}$. We distinguish two cases. If $x \in D$ then $\mathcal{V}\mathsf{ar}(\sigma(x)) \subseteq \mathcal{I}_D(\sigma)$ and thus $x\sigma\rho = x\sigma\sigma' = x\tau$. If $x \notin D$ then $\sigma(x) = \tau(x) = x$ and thus we need to show $\rho(x) = x$. If $x \in \mathcal{I}_D(\sigma)$ then $\rho(x) = \sigma'(x) = x$. If $x \in \mathcal{I}_D(\tau)$ then $x = \tau'(x) \in \mathcal{I}_D(\sigma)$ because $x \notin D$ and $\tau'(\mathcal{I}_D(\tau)) = \mathcal{I}_D(\sigma)$. Hence $\rho(x) = \sigma'(x) = x$ as before. If $x \notin \mathcal{I}_D(\sigma) \cup \mathcal{I}_D(\tau)$ then $\rho(x) = x$ by assumption. □

The proof of the following result is left to the reader (Exercise 2.45).

Theorem 2.5.10. *The relation $\succcurlyeq$ is a well-founded order on substitutions.*

Unifiable terms have infinitely many unifiers, but there is single unifier (modulo renaming) that subsumes every other unifier.

Definition 2.5.11. A *most general* unifier (*mgu* for short) of s and t is at least as general as any other unifier of s and t.

Theorem 2.5.12. *If two terms are unifiable then they have an mgu.*

Definition 2.5.13. The following inference rules constitute a simple procedure for computing mgus:

[d] ***decomposition***

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

[v] ***variable elimination***

$$\frac{\{x \approx t\} \uplus E}{E\sigma} \quad \text{and} \quad \frac{\{t \approx x\} \uplus E}{E\sigma} \quad \text{if } x \notin \mathcal{V}\mathsf{ar}(t) \text{ and } \sigma = \{x \mapsto t\}$$

[t] ***removal of trivial equations***

$$\frac{\{x \approx x\} \uplus E}{E}$$

These rules operate on finite sets of equations. The condition $x \notin \mathcal{V}\mathsf{ar}(t)$ in the variable elimination rule [v] is called the *occurs-check*. If E and E' are the upper and lower goal in the inference rule $[\alpha]$ ($\alpha \in \{\mathsf{d},\mathsf{v},\mathsf{t}\}$), we write $E \Longrightarrow_{[\alpha]} E'$. The substitution σ in the variable elimination rule [v] may be supplied as subscript. A derivation $E \Longrightarrow^* \varnothing$ is said to be *successful*.

Suppose we want to determine whether the terms s and t are unifiable. Starting from $E_1 = \{s \approx t\}$, we repeatedly apply the transformation rules of Definition 2.5.13. Since there are no infinite derivations (Exercise 2.47), this process terminates after n steps:

$$E_1 = \{s \approx t\} \Longrightarrow_{\sigma_1} E_2 \Longrightarrow_{\sigma_2} \cdots \Longrightarrow_{\sigma_{n-1}} E_n$$

(If a step uses [d] or [t] then the corresponding substitution is ε.) Now we can show that s and t are unifiable if and only if $E_n = \varnothing$. Furthermore, if $E_n = \varnothing$ then $\sigma_1\sigma_2\cdots\sigma_{n-1}$ is an mgu of s and t. Moreover, there is no need to compute more than one maximal derivation:

1 if two terms are not unifiable then there are no successful derivations,

2 if two terms are unifiable then all derivations are successful and produce an mgu.

Before proving these statements, we illustrate the algorithm by means of two examples.

Example 2.5.14. The sequence

$$\begin{aligned}
\{x + (\mathsf{0} + y) \approx \mathsf{s}(z) + (\mathsf{0} + x)\} &\Longrightarrow_{[\mathsf{d}]} && \{x \approx \mathsf{s}(z),\ \mathsf{0} + y \approx \mathsf{0} + x\} \\
&\Longrightarrow_{[\mathsf{d}]} && \{x \approx \mathsf{s}(z),\ \mathsf{0} \approx \mathsf{0},\ y \approx x\} \\
&\Longrightarrow_{[\mathsf{v}],\{y \mapsto x\}} && \{x \approx \mathsf{s}(z),\ \mathsf{0} \approx \mathsf{0}\} \\
&\Longrightarrow_{[\mathsf{d}]} && \{x \approx \mathsf{s}(z)\} \\
&\Longrightarrow_{[\mathsf{v}],\{x \mapsto \mathsf{s}(z)\}} && \varnothing
\end{aligned}$$

shows that the terms $x + (\mathsf{0} + y)$ and $\mathsf{s}(z) + (\mathsf{0} + x)$ are unifiable, with mgu

$$\{y \mapsto x\}\{x \mapsto \mathsf{s}(z)\} = \{y \mapsto \mathsf{s}(z),\ x \mapsto \mathsf{s}(z)\}$$

The terms $x + x$ and $y + \mathsf{s}(y)$ are not unifiable since the derivation

$$\begin{aligned}
\{x + x \approx y + \mathsf{s}(y)\} &\Longrightarrow_{[\mathsf{d}]} && \{x \approx y,\ x \approx \mathsf{s}(y)\} \\
&\Longrightarrow_{[\mathsf{v}],\{x \mapsto y\}} && \{y \approx \mathsf{s}(y)\}
\end{aligned}$$

ends in the normal form $\{y \approx \mathsf{s}(y)\}$. Note that the variable elimination rule is not applicable to $\{y \approx \mathsf{s}(y)\}$ due to the occurs-check.

Definition 2.5.15. Given a finite set E of equations, we denote by $\Sigma(E)$ the set of unifiers of E.

Note that $\Sigma(\varnothing)$ is the set of all substitutions over the given signature.

Lemma 2.5.16. *If $E_1 \Longrightarrow_\sigma E_2$ then $\Sigma(E_1) = \sigma \cdot \Sigma(E_2)$.*

Here $\sigma \cdot \Sigma(E_2)$ denotes $\{\sigma\tau \mid \tau \in \Sigma(E_2)\}$.

Proof We perform a case analysis on the employed inference rule. If $E_1 \Longrightarrow_\sigma E_2$ by an application of $[\mathsf{d}]$ or $[\mathsf{t}]$ then $\sigma = \epsilon$ and $\Sigma(E_1) = \Sigma(E_2)$. Consider the case $E_1 \Longrightarrow_{[\mathsf{v}],\sigma} E_2$. We have $E_1 = \{x \approx t\} \uplus E$ or $E_1 = \{t \approx x\} \uplus E$ with $x \notin \mathcal{V}\mathsf{ar}(t)$, $\sigma = \{x \mapsto t\}$, and $E_2 = E\sigma$. First we show $\Sigma(E_1) \subseteq \sigma \cdot \Sigma(E_2)$. Let $\tau \in \Sigma(E_1)$. So $\tau \in \Sigma(E)$ and $x\tau = t\tau$. Consider the substitution $\tau' = \{y \mapsto \tau(y) \mid y \neq x\}$. Since $x \notin \mathcal{V}\mathsf{ar}(t)$, $t\tau = t\tau'$ and thus $x\tau = x\sigma\tau'$. Moreover, if $y \neq x$ then $y\tau = y\tau' = y\sigma\tau'$. Hence $\tau = \sigma\tau'$. Together with $\tau \in \Sigma(E)$ we obtain $\tau' \in \Sigma(E_2)$. Hence $\Sigma(E_1) \subseteq \sigma \cdot \Sigma(E_2)$. Next we show $\sigma \cdot \Sigma(E_2) \subseteq \Sigma(E_1)$. Let $\tau \in \Sigma(E_2)$. So $\sigma\tau \in \Sigma(E)$. Since $\sigma\tau$ is also a solution of $x \approx t$ and $t \approx x$, $\sigma\tau \in \Sigma(E_1)$. □

Theorem 2.5.17. *Let E be a finite list of equations.*

1. *If $\Sigma(E) = \varnothing$ then E admits no successful derivations.*
2. *If $\Sigma(E) \neq \varnothing$ then all maximal derivations starting from E are successful and compute an mgu of E.*

Theorem 2.5.12 is a direct consequence of Theorem 2.5.17.

Proof Consider an arbitrary maximal derivation $E \Longrightarrow_\sigma^* F$ starting from E. Repeated applications of Lemma 2.5.16 yield $\Sigma(E) = \sigma \cdot \Sigma(F)$.

1 We obtain $\Sigma(F) = \varnothing$ from $\Sigma(E) = \varnothing$. Hence $F \neq \varnothing$.

2 Let τ be an arbitrary unifier of E. So $\tau \in \Sigma(E)$. Hence there exists a substitution $\tau' \in \Sigma(F)$ such that $\tau = \sigma\tau'$ and thus $\sigma \leqslant \tau$. From $\Sigma(F) \neq \varnothing$ and the fact that no inference rule is applicable to F we infer $F = \varnothing$. Hence the derivation $E \Longrightarrow_\sigma^* F$ is successful and σ is an mgu of E. □

Exercises

2.42 ***a*** Show that the equation $x \times x \approx \mathsf{s}(\mathsf{s}(0))$ is not unifiable.

b Construct an algebra in which the equation $x \times x \approx \mathsf{s}(\mathsf{s}(0))$ is satisfiable.

2.43 Is the statement

$$\sigma \gtrsim \tau \iff t\sigma \gtrsim t\tau \text{ for every term } t$$

true or false? Give a proof or a counterexample.

2.44 Determine an mgu of the following pairs of terms, if possible.

a $\mathsf{f}(\mathsf{g}(x,y),x,y)$ and $\mathsf{f}(z,\mathsf{g}(y,y),y)$

b $\mathsf{g}(\mathsf{h}(x),\mathsf{g}(x,y))$ and $\mathsf{g}(z,\mathsf{g}(\mathsf{g}(x,x),z))$

c $\mathsf{f}(x,\mathsf{g}(x,y),\mathsf{h}(y))$ and $\mathsf{f}(\mathsf{g}(z,z),x,x)$

2.45 With every substitution σ we associate the natural number $d(\sigma) = |\sigma| - |\mathcal{D}\mathsf{om}(\sigma) \cap \mathcal{I}(\sigma)|$. Here

$$|\sigma| \;=\; \sum \big\{\, |\sigma(x)| \;\big|\; x \in \mathcal{D}\mathsf{om}(\sigma) \,\big\}$$

a Compute $d(\sigma)$ for the following substitutions σ:

- ▷ ε
- ▷ $\{x \mapsto \mathsf{f}(x,y)\}$
- ▷ $\{x \mapsto \mathsf{f}(x,x)\}$
- ▷ $\{x \mapsto \mathsf{f}(y,y),\ y \mapsto x\}$

b Show that $d(\sigma) = 0$ if and only if σ is a renaming.

c Show that $d(\sigma) = d(\tau)$ if σ and τ are variants.

d Prove Theorem 2.5.10 by showing that $d(\sigma) > d(\tau)$ whenever $\sigma \gtrsim \tau$.

e Show that Theorem 2.5.10 fails if substitutions would be allowed to have infinite domain.

2.46 Let s and t be terms.

a Show that all mgus of s and t are variants of each other.

b Suppose σ is an mgu of s and t. Prove that $\sigma\tau$ is an mgu of s and t, for any renaming τ.

2.47 In this exercise we will show that the inference rules of Definition 2.5.13 can be applied only finitely many times to any given finite set of equations. Let $E = \{\, s_1 \approx t_1,\ \ldots,\ s_n \approx t_n \,\}$. Define

$$\mathcal{V}\mathsf{ar}(E) \;=\; \bigcup_{i=1}^{n} \big(\mathcal{V}\mathsf{ar}(s_i) \cup \mathcal{V}\mathsf{ar}(t_i)\big)$$

a Let $E_1 \Longrightarrow E_2$ by an application of the inference rule [d] or [t]. Show that $\mathcal{V}\mathsf{ar}(E_2) \subseteq \mathcal{V}\mathsf{ar}(E_1)$.

b Let $E_1 \Longrightarrow_{[\mathsf{v}],\sigma} E_2$. Show that $\mathcal{V}\mathsf{ar}(E_2) \subsetneq \mathcal{V}\mathsf{ar}(E_1)$.

c Show that there are no infinite derivations consisting of $\Longrightarrow_{[\mathsf{d}]}$ and $\Longrightarrow_{[\mathsf{t}]}$-steps.

d Conclude that there are no infinite derivations.

2.48 Let s and t be unifiable terms. Show that not all mgus of s and t can be obtained by the inference rules of Definition 2.5.13.

2.49 A substitution σ is called *idempotent* if $\sigma\sigma = \sigma$.

a Which of the substitutions in Exercise 2.38(a) are idempotent?

b Show that σ is idempotent if and only if $\mathcal{D}\mathsf{om}(\sigma) \cap \mathcal{I}(\sigma) = \varnothing$.

c Let σ be a unifier of two terms s and t. Prove that σ is an idempotent mgu if and only if $\tau = \sigma\tau$ for every unifier τ of s and t.

d Suppose σ is an idempotent mgu of two terms s, t that have no variables in common. Prove that $\mathcal{D}\mathsf{om}(\sigma) \cup \mathcal{I}(\sigma) = \mathcal{V}\mathsf{ar}(s) \cup \mathcal{V}\mathsf{ar}(t)$.

2.50 Let s and t be linear and unifiable terms without common variables. Let $\mathcal{V}\mathsf{ar}(s) = \{x_1, \ldots, x_n\}$ and $\mathcal{V}\mathsf{ar}(t) = \{y_1, \ldots, y_m\}$. Further assume that p_i (q_j) is the (unique) position of x_i (y_j) in s (t) for $1 \leqslant i \leqslant n$ ($1 \leqslant j \leqslant m$). Prove that the substitution

$$\tau_{s,t} = \{x_i \mapsto t|_{p_i} \mid 1 \leqslant i \leqslant n \text{ and } p_i \in \mathcal{P}\mathsf{os}(t)\} \cup \{y_j \mapsto s|_{q_j} \mid 1 \leqslant j \leqslant m \text{ and } q_j \in \mathcal{P}\mathsf{os}_{\mathcal{F}}(s)\}$$

is an idempotent mgu of s and t.

Bibliographic Notes

Theorem 2.3.12 is from Birkhoff [13]. Exercise 2.41(d) is from Hirokawa *et al.* [55]. Exercise 2.36 presents the (negative) solution of Wilkie [136] to Tarski's High School Algebra problem. This problem asked whether all true equations over the positive integers are valid in $\mathcal{HSI}$. Robinson [114] was the first to present an algorithm for computing mgus. The presentation in Definition 2.5.13 using inference rules is due to Martelli and Montanari [90]. Theorem 2.5.10 is from Eder [38].

Chapter 3

Term Rewrite Systems

By adding more structure to the abstract rewrite systems of the first chapter we obtain term rewrite systems. In Section 3.1 we define the concept of term rewriting. In Section 3.2 we present a few examples to give an idea of the issues addressed in later chapters. In Section 3.3 we show that the behavior of a Turing machine can be simulated by a term rewrite system. Hence term rewrite systems have universal computing power. As a consequence, all interesting properties of term rewrite systems are undecidable. Section 3.4 is devoted to combinatory logic.

3.1 Term Rewriting

We start this section with an alternative presentation of equational reasoning. This presentation, which is based on the slogan "replacing equals by equals", paves the way for the definition of term rewriting.

Definition 3.1.1. For every ES $\mathcal{E}$ we define the relation $\to_{\mathcal{E}}$ on terms as follows: $s \to_{\mathcal{E}} t$ if there exist an equation $\ell \approx r$ in $\mathcal{E}$, a substitution σ, and a context C such that $s = C[\ell\sigma]$ and $t = C[r\sigma]$.

Example 3.1.2. We have $\mathsf{s}(\mathsf{s}(\mathsf{s}(0)) + (0 + 0)) \to_{\mathcal{E}} \mathsf{s}(\mathsf{s}(\mathsf{s}(0) + (0 + 0)))$ with respect to the ES $\mathcal{E}$ of Example 2.3.6 because we can take $\ell = \mathsf{s}(x) + y$, $r = \mathsf{s}(x + y)$, $C = \mathsf{s}(\Box)$, and $\sigma = \{x \mapsto \mathsf{s}(0), y \mapsto 0 + 0\}$ in the above definition.

Lemma 3.1.3. *Let $\mathcal{E}$ be an* ES*. The relation $\to_{\mathcal{E}}$ is the smallest rewrite relation that includes $\mathcal{E}$.*

Proof Let $\ell \approx r$ be an equation in $\mathcal{E}$. We obtain $\ell \to_{\mathcal{E}} r$ by letting $C = \Box$ and $\sigma = \varepsilon$ in Definition 3.1.1. Hence $\to_{\mathcal{E}}$ includes $\mathcal{E}$. Next we show that $\to_{\mathcal{E}}$ is a rewrite relation. Suppose $s = C[\ell\sigma] \to_{\mathcal{E}} C[r\sigma] = t$. Let C' be an arbitrary context. Define $C'' = C'[C]$. Clearly C'' is a context. We have $C'[s] = C'[C[\ell\sigma]] = (C'[C])[\ell\sigma] = C''[\ell\sigma]$ and likewise $C'[t] = C''[r\sigma]$. Therefore $C'[s] \to_{\mathcal{E}} C'[t]$. Let σ' be an arbitrary substitution. Define $C'' = C\sigma'$ and $\sigma'' = \sigma\sigma'$. We have $s\sigma' = (C[\ell\sigma])\sigma' = (C\sigma')[\ell\sigma\sigma'] = C''[\ell\sigma'']$ and likewise $t\sigma' = C''[r\sigma'']$. Therefore $s\sigma' \to_{\mathcal{E}} t\sigma'$. It remains to show that $\to_{\mathcal{E}}$ is included in any other rewrite relation that includes $\mathcal{E}$. Suppose $\rightarrowtail$ is such a relation. Let $s = C[\ell\sigma] \to_{\mathcal{E}}$

$C[r\sigma] = t$. We have to show $s \rightarrowtail t$. Because $\rightarrowtail$ includes $\mathcal{E}$ we have $\ell \rightarrowtail r$. Closure under substitutions of $\rightarrowtail$ yields $\ell\sigma \rightarrowtail r\sigma$. Since $\rightarrowtail$ is also closed under contexts, we obtain the desired $s = C[\ell\sigma] \rightarrowtail C[r\sigma] = t$. □

With every ES $\mathcal{E}$ over a signature $\mathcal{F}$ we associate the ARS $\langle \mathcal{T}(\mathcal{F}, \mathcal{V}), \rightarrow_{\mathcal{E}} \rangle$. Via this connection all notations and concepts defined in Chapter 1 for ARSs carry over to ESs. The relation $\rightarrow_{\mathcal{E}}$ is obtained by using the equations in $\mathcal{E}$ from left to right. Equations are normally used in both directions, so the basic relation of equational reasoning is $\leftrightarrow_{\mathcal{E}}$, the symmetric closure of $\rightarrow_{\mathcal{E}}$.

Lemma 3.1.4. *Let $\mathcal{E}$ be an* ES. *The relations $\approx_{\mathcal{E}}$ and $\leftrightarrow^*_{\mathcal{E}}$ coincide.*

Proof From Lemma 3.1.3 it follows that $\leftrightarrow^*_{\mathcal{E}}$ is the smallest equivalence and rewrite relation that includes $\mathcal{E}$. According to Lemma 2.3.7 $\leftrightarrow^*_{\mathcal{E}}$ and $=_{\mathcal{E}}$ coincide. The desired result follows from Theorem 2.3.12. □

Definition 3.1.5. A *rewrite rule* is an equation $\ell \approx r$ that satisfies the following two conditions:

1. the left-hand side ℓ is not a variable,
2. the variables which occur in the right-hand side r occur also in ℓ, i.e., $\mathcal{V}\mathsf{ar}(r) \subseteq \mathcal{V}\mathsf{ar}(\ell)$.

Rewrite rules $\ell \approx r$ will henceforth be written as $\ell \rightarrow r$. A *term rewrite system* (TRS for short) is an ES with the property that all its equations are rewrite rules.

We denote TRSs by $\mathcal{R}$ and $\mathcal{S}$. Below we give an equivalent definition of $\rightarrow_{\mathcal{R}}$ using the position formalism developed in Section 2.1.

Definition 3.1.6. Let $\mathcal{R}$ be a TRS. We write $s \rightarrow_{\mathcal{R}} t$ if there exist a rewrite rule $\ell \rightarrow r$ in $\mathcal{R}$, a substitution σ, and a position $p \in \mathcal{P}\mathsf{os}(s)$ such that $s|_p = \ell\sigma$ and $t = s[r\sigma]_p$. The subterm $\ell\sigma$ of s (at position p) is called a *redex*—an abbreviation of reducible expression—and we say that s rewrites to t by *contracting* redex $\ell\sigma$. The instance $r\sigma$ of the right-hand side r is called the *contractum* of $\ell\sigma$.

In the following we usually drop the subscript $\mathcal{R}$ from $\rightarrow_{\mathcal{R}}$ and its derivatives like $\leftrightarrow^*_{\mathcal{R}}$ and $\downarrow_{\mathcal{R}}$ when no confusion can arise.

We write $s \rightarrow_{p|\ell\rightarrow r|\sigma} t$ if we want to indicate the position p of the contracted redex, the employed rewrite rule $\ell \rightarrow r$, and the substitution σ. Actually we can omit σ without loss of information. First observe that we may assume that $\mathcal{D}\mathsf{om}(\sigma) \subseteq \mathcal{V}\mathsf{ar}(\ell)$ since it is completely irrelevant which terms σ assigns to variables not in $\mathcal{V}\mathsf{ar}(\ell)$, due to the restriction $\mathcal{V}\mathsf{ar}(r) \subseteq \mathcal{V}\mathsf{ar}(\ell)$ imposed on rewrite rules. Hence we can use matching of the left-hand side ℓ to the subterm $s|_p$ in order to obtain a substitution σ which determines the resulting term t. (This is not true for arbitrary ESs.) This explains why we usually write $s \rightarrow_{p|\ell\rightarrow r} t$. If only the position p of the contracted redex matters, we simply write $s \rightarrow_p t$. We call $s \rightarrow_\epsilon t$ a *root* rewrite step.

From Lemma 3.1.3 we learn that $\rightarrow_{\mathcal{R}}$ is the smallest rewrite relation that includes $\mathcal{R}$, for every TRS $\mathcal{R}$. An easy induction proof yields the following result, which will be freely used in the sequel.

$\mathcal{R}_1$	$\mathcal{R}_2$
$0 + y \to y$	$x + 0 \to x$
$\mathsf{s}(x) + y \to \mathsf{s}(x + y)$	$x + \mathsf{s}(y) \to \mathsf{s}(x + y)$
$0 \times y \to 0$	$x \times 0 \to 0$
$\mathsf{s}(x) \times y \to (x \times y) + y$	$x \times \mathsf{s}(y) \to (x \times y) + x$

Table 3.1: Two TRSs for addition and multiplication.

Lemma 3.1.7. *The relations* $\to_{\mathcal{R}}^*$, $\to_{\mathcal{R}}^+$, $\downarrow_{\mathcal{R}}$, $\uparrow_{\mathcal{R}}$, *and* $\leftrightarrow_{\mathcal{R}}^*$ *are rewrite relations for every* TRS $\mathcal{R}$.

Example 3.1.8. Consider the TRS $\mathcal{R}_1$ of Table 3.1, which can be viewed as a specification of addition and multiplication over natural numbers, and the term $\mathsf{s}(\mathsf{s}(0)) \times \mathsf{s}(\mathsf{s}(0))$. This term is a redex with respect to the rewrite rule $\mathsf{s}(x) \times y \to (x \times y) + y$. The corresponding matching substitution is $\sigma = \{x \mapsto \mathsf{s}(0),\, y \mapsto \mathsf{s}(\mathsf{s}(0))\}$. Hence the term $\mathsf{s}(\mathsf{s}(0)) \times \mathsf{s}(\mathsf{s}(0)) = (\mathsf{s}(x) \times y)\sigma$ rewrites to $((x \times y) + y)\sigma = (\mathsf{s}(0) \times \mathsf{s}(\mathsf{s}(0))) + \mathsf{s}(\mathsf{s}(0))$. This is the first step in the following rewrite sequence from $\mathsf{s}(\mathsf{s}(0)) \times \mathsf{s}(\mathsf{s}(0))$ to the normal form $\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0))))$, where the contracted redexes are underlined:

$$\begin{aligned}\underline{\mathsf{s}(\mathsf{s}(0)) \times \mathsf{s}(\mathsf{s}(0))} &\to (\underline{\mathsf{s}(0) \times \mathsf{s}(\mathsf{s}(0))}) + \mathsf{s}(\mathsf{s}(0)) \to ((\underline{0 \times \mathsf{s}(\mathsf{s}(0))}) + \mathsf{s}(\mathsf{s}(0))) + \mathsf{s}(\mathsf{s}(0)) \\ &\to (\underline{0 + \mathsf{s}(\mathsf{s}(0))}) + \mathsf{s}(\mathsf{s}(0)) \to \underline{\mathsf{s}(\mathsf{s}(0)) + \mathsf{s}(\mathsf{s}(0))} \to \mathsf{s}(\underline{\mathsf{s}(0) + \mathsf{s}(\mathsf{s}(0))}) \\ &\to \mathsf{s}(\mathsf{s}(\underline{0 + \mathsf{s}(\mathsf{s}(0))})) \to \mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0))))\end{aligned}$$

Hence $\mathsf{s}(\mathsf{s}(0)) \times \mathsf{s}(\mathsf{s}(0)) \to^! \mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0))))$. The above sequence is in fact the only rewrite sequence from $\mathsf{s}(\mathsf{s}(0)) \times \mathsf{s}(\mathsf{s}(0))$ to $\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0))))$ as one easily verifies, but in general things are more complicated. Consider for example the TRS $\mathcal{R}_2$ of Table 3.1. This TRS is very similar to $\mathcal{R}_1$, but the number of rewrite sequences from $\mathsf{s}(\mathsf{s}(0)) \times \mathsf{s}(\mathsf{s}(0))$ to $\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0))))$ is greatly increased, as can be seen from Figure 3.1.

Lemma 3.1.9. *Every* TRS *with finitely many rewrite rules is finitely branching.*

Proof Let $\mathcal{R}$ be a TRS with finitely many rewrite rules. Let s be an arbitrary term. We have to show that the set $\{t \mid s \to t\}$ of one-step reducts of s is finite. For every term t in that set there exists a position $p \in \mathcal{P}\mathsf{os}(s)$ and a rewrite rule $\ell \to r \in \mathcal{R}$ such that $s \to_{p|\ell\to r} t$. Given s, the pair of p and $\ell \to r$ completely determines t. Since $\mathcal{P}\mathsf{os}(s)$ and $\mathcal{R}$ are finite sets, the set of one-step reducts of s must be finite. □

This property ensures that normal forms are computable in a breadth-first manner. More formally, if $\mathcal{R}$ is a TRS with finitely many rewrite rules and s is normalizing then we can actually compute a term t such that $s \to_{\mathcal{R}}^! t$. In particular, normal forms are computable for finite normalizing TRSs.

In the next few definitions we introduce a number of syntactical properties that a given TRS may enjoy.

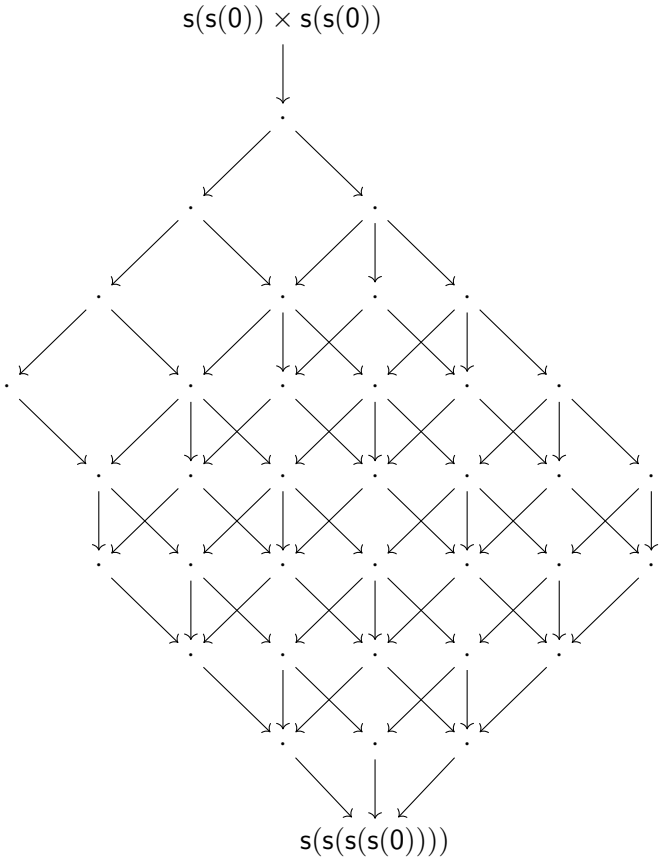


Figure 3.1: All rewrite sequences starting from $\mathsf{s}(\mathsf{s}(0)) \times \mathsf{s}(\mathsf{s}(0))$ in $\mathcal{R}_2$.

Definition 3.1.10. A rewrite rule $\ell \to r$ is called *left-linear* if ℓ is a linear term. The rule $\ell \to r$ is called *right-linear* if r is a linear term. A *linear* rewrite rule is both left and right-linear. A TRS is left-linear (right-linear, linear) if all its rewrite rules are left-linear (right-linear, linear).

Left-linearity will turn out to play an important role when it comes to determining confluence of TRSs.

Definition 3.1.11. A TRS $\mathcal{R}$ is called *left-reduced* if the left-hand side ℓ of every rewrite rule $\ell \to r \in \mathcal{R}$ is a normal form of $\mathcal{R} \setminus \{\ell \to r\}$. We say that $\mathcal{R}$ is *right-reduced* if the right-hand side r of every rewrite rule $\ell \to r \in \mathcal{R}$ is a normal form of $\mathcal{R}$. A TRS that is both left and right-reduced is called *reduced*.

Definition 3.1.12. Let $\mathcal{R}$ be a TRS over a signature $\mathcal{F}$. A function symbol $f \in \mathcal{F}$ is called a *defined symbol* if there exists a rewrite rule $\ell \to r \in \mathcal{R}$ such that $f = \mathsf{root}(\ell)$. Function symbols of $\mathcal{F}$ that are not defined symbols are called *constructors*. The subset of $\mathcal{F}$ consisting of all defined symbols is denoted by $\mathcal{F}_\mathcal{D}$ and the set $\mathcal{F} \setminus \mathcal{F}_\mathcal{D}$ of all constructors

by $\mathcal{F}_\mathcal{C}$. A *constructor system* (CS for short) is a TRS $(\mathcal{F},\mathcal{R})$ with the property that every left-hand side $f(t_1,\ldots,t_n)$ of a rewrite rule of $\mathcal{R}$ satisfies $t_1,\ldots,t_n \in \mathcal{T}(\mathcal{F}_\mathcal{C},\mathcal{V})$.

Example 3.1.13. The TRSs of Table 3.1 are left-linear but not right-linear since the right-hand side $(x \times y) + x$ of the last rewrite rule contains two occurrences of the variable x. The defined symbols are $+$ and $\times$, the constructors are s and $\mathsf{0}$. Both TRSs are reduced CSs. In Chapter 4 we will see that both CSs are terminating. They are also confluent, as we will see in Chapter 5, and hence they are complete.

Definition 3.1.14. An equation $\ell \approx r$ is called *ground* if it does not contain variables. The equation $\ell \approx r$ is called *right-ground* if r is a ground term. An ES is (right-)ground if all its equations are (right-)ground.

Note that every ground ES is a TRS.

Definition 3.1.15. A *string rewrite system* (SRS for short) is a TRS over a signature that contains only unary function symbols.

Every term in an SRS contains exactly one variable. Moreover, both sides of a rewrite rule of an SRS contain the same variable. Since the variable name carries no information, in examples we often omit variables altogether and write terms as strings.

Example 3.1.16. The TRS consisting of the rewrite rules

$$\mathsf{a}(\mathsf{a}(x)) \to \mathsf{b}(\mathsf{c}(x)) \qquad \mathsf{b}(\mathsf{b}(x)) \to \mathsf{a}(\mathsf{c}(x)) \qquad \mathsf{c}(\mathsf{c}(x)) \to \mathsf{a}(\mathsf{b}(x))$$

is an SRS. Writing terms as strings, the rewrite rules are presented as follows:

$$\mathsf{aa} \to \mathsf{bc} \qquad \mathsf{bb} \to \mathsf{ac} \qquad \mathsf{cc} \to \mathsf{ab}$$

We conclude this section by extending the notion of literal similarity to TRSs.

Definition 3.1.17. If $\ell \to r$ is a rewrite rule and σ a renaming then the rewrite rule $\ell\sigma \to r\sigma$ is called a *variant* of $\ell \to r$. A TRS is said to be *variant-free* if it does not contain rewrite rules that are variants of each other.

Throughout the following we assume that **TRSs are variant-free**. This entails no loss of generality, cf. Exercise 3.11(a).

Definition 3.1.18. Two TRSs $\mathcal{R}_1$ and $\mathcal{R}_2$ over the same signature $\mathcal{F}$ are called *literally similar*, denoted by $\mathcal{R}_1 \doteq \mathcal{R}_2$, if every rewrite rule in $\mathcal{R}_1$ has a variant in $\mathcal{R}_2$ and vice-versa.

It is easy to show (Exercise 3.11(b)) that literally similar TRSs induce the same rewrite relation.

Example 3.1.19. The TRSs

$$\mathcal{R}_1\colon \quad 0 + x \to x \qquad\qquad \mathcal{R}_2\colon \quad 0 + y \to y$$

are literally similar. The TRSs

$$\mathcal{R}_3\colon \quad \mathsf{f}(x) \to x \qquad \mathsf{f}(x) \to \mathsf{a} \qquad\qquad \mathcal{R}_4\colon \quad \mathsf{f}(x) \to x \qquad \mathsf{f}(\mathsf{a}) \to \mathsf{a}$$

are not literally similar.

All syntactical notions that we defined for TRSs are preserved under literal similarity.

Exercises

3.1 Determine all terms t such that $\mathsf{f}(\mathsf{a},\mathsf{a}) \leftrightarrow_{\mathcal{E}} t$ for each of the following ESs $\mathcal{E}$.

a $\{\mathsf{f}(x,\mathsf{a}) \approx \mathsf{g}(\mathsf{a}),\ \mathsf{f}(x,x) \approx x\}$

b $\{\mathsf{a} \approx \mathsf{f}(\mathsf{a},\mathsf{g}(\mathsf{a})),\ \mathsf{g}(\mathsf{g}(\mathsf{a})) \approx \mathsf{g}(\mathsf{a})\}$

c $\{\mathsf{a} \approx x\}$

3.2 Which of the following equations are rewrite rules?

a $\mathsf{f}(x,x) \approx \mathsf{g}(\mathsf{f}(x,y))$

b $\mathsf{f}(x,\mathsf{g}(y)) \approx \mathsf{g}(\mathsf{f}(\mathsf{g}(x),\mathsf{f}(x,x)))$

c $\mathsf{g}(x) \approx \mathsf{c}$

d $x \approx \mathsf{g}(x)$

3.3 Consider the TRS $\mathcal{R}_1$ of Table 3.1.

a Show that $\mathsf{s}(\mathsf{s}(\mathsf{0})) \times \mathsf{s}(\mathsf{s}(\mathsf{0}))$ rewrites to $\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{0}))))))$.

b Compute all rewrite sequences starting from the term $\mathsf{s}(\mathsf{0}) \times (\mathsf{0} + \mathsf{s}(\mathsf{0}))$.

c Describe the set of ground normal forms of $\mathcal{R}_1$.

3.4 Consider the TRS $\mathcal{R}_2$ of Table 3.1.

a Complete Figure 3.1 by finding terms which represent the dots.

b How many rewrite sequences are there from $\mathsf{s}(\mathsf{s}(\mathsf{0})) \times \mathsf{s}(\mathsf{s}(\mathsf{0}))$ to $\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{0})))$?

3.5 The TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{array}{lll} \neg(x \wedge y) \to (\neg x) \vee (\neg y) & x \wedge (y \vee z) \to (x \wedge y) \vee (x \wedge z) & \neg(\neg x) \to x \\ \neg(x \vee y) \to (\neg x) \wedge (\neg y) & (x \vee y) \wedge z \to (x \wedge z) \vee (y \wedge z) & \end{array}$$

transforms propositional formulae into disjunctive normal form.

a Show that the term $\neg(x \vee (\neg(y \vee z)))$ is confluent.

b Verify that the term $\neg(x \wedge (y \vee z))$ has different normal forms.

c Compute all normal forms of the term $\neg(x \wedge ((y \vee x) \vee x))$.

d Show that $\mathcal{R}$ is not locally confluent.

3.6 ***a*** Let R and S be rewrite relations. Show that $R \cdot S$, $R \cup S$, R^{-1}, $R^{=}$, R^{+}, and R^{*} are rewrite relations.

b Prove Lemma 3.1.7.

3.7 ***a*** Show that a terminating TRS $\mathcal{R}$ is reduced if and only if both ℓ and r are normal forms of $\mathcal{R} \setminus \{\ell \to r\}$, for all rewrite rules $\ell \to r \in \mathcal{R}$.

b Construct a reduced SRS that is not terminating.

c Construct a reduced SRS that is not confluent.

3.8 ***a*** Show that the TRS $\mathcal{R}$ consisting of the three rewrite rules

$$\mathsf{f}(x) \to x \qquad \mathsf{f}(x) \to \mathsf{a} \qquad \mathsf{a} \to \mathsf{a}$$

has unique normal forms but not with respect to conversion.

b Construct a TRS having only two rewrite rules with the same behavior as in part (a).

c Construct a TRS having a single rewrite rule that is normalizing but not terminating.

3.9 ***a*** Let $\mathcal{E}$ be an ES that is not a TRS. Show that $\mathcal{E}$ is not terminating.

b Show that every finite ES that is not a TRS because it contains equations that violate the second restriction imposed on rewrite rules in Definition 3.1.5 is not finitely branching, cf. Lemma 3.1.9.

3.10 Why does Definition 3.1.14 not contain the notion *left-ground*?

3.11 ***a*** Suppose a TRS $\mathcal{R}$ contains different rewrite rules $\ell_1 \to r_1$ and $\ell_2 \to r_2$ that are variants of each other. Show that $\mathcal{R}$ defines the same rewrite relation as $\mathcal{R} \setminus \{\ell_2 \to r_2\}$.

b Let $\mathcal{R}_1$ and $\mathcal{R}_2$ be literally similar TRSs. Suppose $s \to_{p|\ell_1 \to r_1|\sigma_1} t$ with $\ell_1 \to r_1 \in \mathcal{R}_1$. Show that $s \to_{p|\ell_2 \to r_2|\sigma_2} t$ for some variant $\ell_2 \to r_2 \in \mathcal{R}_2$ of $\ell_1 \to r_1$ and substitution σ_2. Conclude that literally similar TRSs define the same rewrite relation.

c Can we assume that σ_2 is a variant of σ_1 in the preceding item?

d Suppose $\mathcal{R}_1$ and $\mathcal{R}_2$ are TRSs over the same signature such that $\to_{\mathcal{R}_1} = \to_{\mathcal{R}_2}$. Are $\mathcal{R}_1$ and $\mathcal{R}_2$ literally similar?

3.2 Examples

Finite semi-complete TRSs are of special interest since they have a decidable conversion.

Lemma 3.2.1. *Conversion is decidable for finite semi-complete* TRS*s.*

Proof Let $\mathcal{R}$ be a finite semi-complete TRS. Let s and t be terms. According to Lemma 1.2.9, $s \leftrightarrow^*_{\mathcal{R}} t$ if and only if s and t reduce to the same normal form. In the preceding section we observed that normal forms are computable for finite normalizing TRSs. □

Example 3.2.2. In the complete TRS $\mathcal{R}_1$ of Table 3.1 the terms $\mathsf{s}(\mathsf{s}(0)) \times x$ and $x + x$ are convertible since $x + x$ is the normal form of $\mathsf{s}(\mathsf{s}(0)) \times x$. The terms $x + y$ and $y + x$ are different normal forms and hence not convertible.

Definition 3.2.3. A TRS $(\mathcal{F}, \mathcal{R})$ *represents* an ES $(\mathcal{F}, \mathcal{E})$ if $\leftrightarrow^*_{\mathcal{R}} = \leftrightarrow^*_{\mathcal{E}}$.

Combining Lemma 3.2.1 with Lemma 3.1.4 and Theorem 2.3.12 explains why term rewriting can be useful for deciding validity.

Theorem 3.2.4. *The validity problem for an* ES *is decidable if there exists a finite semi-complete* TRS *that represents it.*

$\mathsf{e} \cdot x \to x$	$x \cdot \mathsf{e} \to x$
$x^- \cdot x \to \mathsf{e}$	$x \cdot x^- \to \mathsf{e}$
$(x \cdot y) \cdot z \to x \cdot (y \cdot z)$	$x^{--} \to x$
$\mathsf{e}^- \to \mathsf{e}$	$(x \cdot y)^- \to y^- \cdot x^-$
$x^- \cdot (x \cdot y) \to y$	$x \cdot (x^- \cdot y) \to y$

Table 3.2: A complete TRS for group theory.

In Chapter 5 we explain how to transform an ES $\mathcal{E}$ into a complete TRS $\mathcal{R}$ that represents $\mathcal{E}$. The method presented in Chapter 5 transforms the ES $\mathcal{E}$ of Exercise 2.32 into the complete TRS $\mathcal{R}$ of Table 3.2. At the present stage it is not at all clear that $\mathcal{R}$ is complete, but in subsequent chapters we develop techniques for proving termination and confluence of TRSs which readily apply to $\mathcal{R}$. Hence the validity problem for group theory is decidable. For instance, the equation $(x^- \cdot y)^- \approx (y \cdot \mathsf{e})^- \cdot x$ is valid in every group—a model of the ES $\mathcal{E}$—since both terms have the same normal form $y^- \cdot x$, but not every group is Abelian as $x \cdot y$ and $y \cdot x$ are different normal forms.

The method presented in Chapter 5 does not always succeed in producing finite complete TRSs, simply because many ESs have an undecidable validity problem. Even if an ES has a decidable validity problem, there may not be a finite (semi-)complete TRS representing it. In Chapter 5 we return to these issues.

Below we show that the validity problem is decidable for ground ESs.

Theorem 3.2.5. *The following problem is decidable:*

instance: *a finite ground* ES $\mathcal{E}$ *and two ground terms* s *and* t

question: $s =_{\mathcal{E}} t$?

Proof Let S be the set of all subterms of terms in $\mathcal{E} \cup \{s \approx t\}$. The idea is to partition S into different congruence classes. This is achieved by the following algorithm.

1. Each term in S is placed in a separate set.
2. Different sets $\{\ldots, t_1, \ldots\}$ and $\{\ldots, t_2, \ldots\}$ are merged when one of the following two conditions is met:
   (a) $t_1 \approx t_2$ is an equation in $\mathcal{E}$,
   (b) $t_1 = f(u_1, \ldots, u_m)$ and $t_2 = f(v_1, \ldots, v_m)$ with u_i and v_i belonging to the same set for all $1 \leqslant i \leqslant m$.
3. Step 2 is repeated until no more sets can be merged.

Termination is obvious since S is finite and each application of step 2 reduces the number of sets. Let n be the number of merge operations in step 2. Upon termination, we determine whether the terms s and t belong to the same set. If that is the case, we conclude $s =_{\mathcal{E}} t$. Otherwise, $s =_{\mathcal{E}} t$ does not hold. For the correctness proof we define binary relations $\equiv_i$ on $S \times S$ for $0 \leqslant i \leqslant n$ as follows: $u \equiv_i v$ if and only if u and v belong to the same set after the i-th merge operation in step 2. We claim that $\equiv_n = \leftrightarrow_{\mathcal{E}}^* \cap (S \times S)$. The correctness proof is concluded by an appeal to Theorem 2.3.12 and Lemma 3.1.4.

We prove the inclusion $\equiv_i \subseteq \leftrightarrow_{\mathcal{E}}^* \cap (S \times S)$ by induction on $0 \leqslant i \leqslant n$. Clearly, $u \equiv_0 v$ if and only if $u, v \in S$ and $u = v$. Suppose $\equiv_{i+1}$ is obtained from $\equiv_i$ by merging the sets A and B. So there exist terms $u' \in A$ and $v' \in B$ such that $u' \approx v' \in \mathcal{E}$ or

$u' = f(u_1, \ldots, u_m)$ and $v' = f(v_1, \ldots, v_m)$ with $u_j \equiv_i v_j$ for all $1 \leqslant j \leqslant m$. In the former case we obviously have $u' \to_{\mathcal{E}} v'$. In the latter case we obtain $u_j \leftrightarrow_{\mathcal{E}}^* v_j$ for all $1 \leqslant j \leqslant m$ from the induction hypothesis and thus $u' \leftrightarrow_{\mathcal{E}}^* v'$ because $\leftrightarrow_{\mathcal{E}}^*$ is a congruence relation. Now consider $u \equiv_{i+1} v$. We have $u, v \in S$ by definition. If $u \equiv_i v$ then $u \leftrightarrow_{\mathcal{E}}^* v$ follows from the induction hypothesis. Otherwise, we may assume (without loss of generality) that $u \in A$ and $v \in B$. Hence $u \equiv_i u'$ and $v' \equiv_i v$, and thus $u \leftrightarrow_{\mathcal{E}}^* u'$ and $v' \leftrightarrow_{\mathcal{E}}^* v$ by the induction hypothesis. Combining this with $u' \leftrightarrow_{\mathcal{E}}^* v'$ yields the desired $u \leftrightarrow_{\mathcal{E}}^* v$.

Conversely, suppose $u \leftrightarrow_{\mathcal{E}}^* v$ with $u, v \in S$. We show $u \equiv_n v$ by induction on the conversion $u \leftrightarrow_{\mathcal{E}}^* v$. If $u = v$ then the result obviously holds. Suppose $u \leftrightarrow_{\mathcal{E}}^+ v$. We distinguish two cases. If $u \leftrightarrow_{\mathcal{E}}^+ v$ contains a root step then there exists an equation $\ell \approx r$ or $r \approx \ell$ in $\mathcal{E}$ such that $u \leftrightarrow_{\mathcal{E}}^* \ell \leftrightarrow_{\mathcal{E}} r \leftrightarrow_{\mathcal{E}}^* v$. Because $\ell, r \in S$, we obtain $u \equiv_n \ell$ and $r \equiv_n v$ from the induction hypothesis. According to $\boxed{2}$(a), ℓ and r belong to the same set and thus $\ell \equiv_n r$. Hence also $u \equiv_n v$. If $u \leftrightarrow_{\mathcal{E}}^+ v$ contains no root step then we may write $u = f(u_1, \ldots, u_m)$ and $v = f(v_1, \ldots, v_m)$ with $u_i \leftrightarrow_{\mathcal{E}}^* v_i$ for all $1 \leqslant i \leqslant m$. By construction, $u_1, \ldots, u_m, v_1, \ldots, v_m \in S$. The induction hypothesis yields $u_i \equiv_n v_i$ for all $1 \leqslant i \leqslant m$. According to $\boxed{2}$(b), u and v belong to the same set and thus $u \equiv_n v$ as desired. □

The *congruence closure* algorithm in the proof of Theorem 3.2.5 is illustrated on a simple example.

Example 3.2.6. Consider the ES $\mathcal{E}$ consisting of the equations

$$\mathsf{f(f(f(a))) \approx g(f(g(f(b))))} \qquad \mathsf{f(g(f(b))) \approx f(a)} \qquad \mathsf{g(g(b)) \approx g(f(a))} \qquad \mathsf{g(a) \approx b}$$

and the terms $s = \mathsf{f(a)}$ and $t = \mathsf{g(b)}$. In step $\boxed{1}$ we create the following sets:

1. {a}	5. {f(f(a))}	9. {f(g(f(b)))}	13. {g(a)}
2. {f(a)}	6. {f(f(f(a)))}	10. {g(f(g(f(b))))}	
3. {b}	7. {f(b)}	11. {g(g(b))}	
4. {g(b)}	8. {g(f(b))}	12. {g(f(a))}	

Next we merge sets using the equations in $\mathcal{E}$, i.e., we repeatedly use condition $\boxed{2}$(a):

1. {a}	5. {f(f(a))}	11. {g(g(b)), g(f(a))}
2. {f(a), f(g(f(b)))}	6. {f(f(f(a))), g(f(g(f(b))))}	
3. {b, g(a)}	7. {f(b)}	
4. {g(b)}	8. {g(f(b))}	

Using the terms in set 2, we can use condition $\boxed{2}$(b) to merge sets 6 and 11:

1. {a}	5. {f(f(a))}
2. {f(a), f(g(f(b)))}	6. {f(f(f(a))), g(f(g(f(b)))), g(g(b)), g(f(a))}
3. {b, g(a)}	7. {f(b)}
4. {g(b)}	8. {g(f(b))}

One easily verifies that no more sets can be merged with condition $\boxed{2}$(b). Since f(a) and

$\mathsf{g(b)}$ belong to different sets, the equation $\mathsf{f(a)} \approx \mathsf{g(b)}$ is not valid in $\mathcal{E}$.

Definition 3.2.7. A reduced complete TRS is called *canonical*.

Example 3.2.8. Consider the ground TRS $\mathcal{R}$ consisting of the rewrite rules

$$\mathsf{f(g(f(b))} \xrightarrow{1} \mathsf{f(a)} \qquad \mathsf{g(f(a))} \xrightarrow{2} \mathsf{g(g(b))} \qquad \mathsf{g(a)} \xrightarrow{3} \mathsf{b} \qquad \mathsf{f(f(f(a))} \xrightarrow{4} \mathsf{g(g(b))}$$

and let $\mathcal{E}$ be the ES of Example 3.2.6. The inclusion $\mathcal{E} \subseteq {\leftrightarrow}^*_{\mathcal{R}}$ is easily verified:

$$\mathsf{f(f(f(a)))} \xrightarrow{4} \mathsf{g(g(b))} \xleftarrow{2} \mathsf{g(f(a))} \xleftarrow{1} \mathsf{g(f(g(f(b))))}$$
$$\mathsf{f(g(f(b)))} \xrightarrow{1} \mathsf{f(a)} \qquad\qquad \mathsf{g(g(b))} \xleftarrow{2} \mathsf{g(f(a))} \qquad\qquad \mathsf{g(a)} \xrightarrow{3} \mathsf{b}$$

We also have $\mathcal{R} \subseteq {\leftrightarrow}^*_{\mathcal{E}}$. For the first three rules of $\mathcal{R}$ this is obvious and for rule 4 this can be verified by congruence closure. As a matter of fact, both $\mathsf{f(f(f(a))}$ and $\mathsf{g(g(b))}$ belong to set 6 in Example 3.2.6. Consequently, $\mathcal{R}$ represents $\mathcal{E}$. Note that $\mathcal{R}$ is reduced. In Chapter 5 we will see that reduced ground TRSs are complete. Hence $\mathcal{R}$ is a canonical presentation of $\mathcal{E}$.

The left-linear reduced CS of Table 3.3 specifies *Ackermann's function*, a well-known computable function over the natural numbers that is not primitive recursive. Using methods developed in the next two chapters, completeness of this CS is easily established. Hence for computing normal forms we can adopt any strategy for selecting redexes to contract. This does not imply though that normal forms are easy to compute. For instance, the (unique) rewrite sequence from the term $\mathsf{ack(s(s(0)),s(s(0)))}$ to its normal form $\mathsf{s(s(s(s(s(s(s(0)))))))}$ in which only *innermost* redexes are contracted consists already of 27 rewrite steps, but normalizing $\mathsf{ack(s(s(s(0))),s(s(s(0))))}$ is monkish work (Exercise 3.15).

Definition 3.2.9. A redex s in a term t is called *innermost* if all proper subterms of s are in normal form. A redex s in a term t is called *outermost* if it is not a proper subterm of another redex in t.

The TRS $\mathcal{R}$ of Table 3.4 specifies addition over natural numbers in the usual decimal representation. Its signature $\mathcal{F}$ consists of the digits $\mathsf{0}-\mathsf{9}$ and the binary function symbols $+$ and $:$ with the latter being used to compose digits into numbers. For instance, the number 123 is represented by the term $(\mathsf{1}:\mathsf{2}):\mathsf{3}$. We assume that $:$ binds stronger than $+$, so $x+y:z$ stands for the term $x+(y:z)$. The following rewrite sequence shows how

$$\begin{array}{r@{\;\to\;}l}
\mathsf{ack(0},y) & \mathsf{s}(y) \\
\mathsf{ack(s}(x),\mathsf{0}) & \mathsf{ack}(x,\mathsf{s(0)}) \\
\mathsf{ack(s}(x),\mathsf{s}(y)) & \mathsf{ack}(x,\mathsf{ack(s}(x),y))
\end{array}$$

Table 3.3: Ackermann's TRS.

$0+0 \to 0$	$1+0 \to 1$	$\cdots$	$9+0 \to 9$
$0+1 \to 1$	$1+1 \to 2$	$\cdots$	$9+1 \to 1:0$
$0+2 \to 2$	$1+2 \to 3$	$\cdots$	$9+2 \to 1:1$
$0+3 \to 3$	$1+3 \to 4$	$\cdots$	$9+3 \to 1:2$
$0+4 \to 4$	$1+4 \to 5$	$\cdots$	$9+4 \to 1:3$
$0+5 \to 5$	$1+5 \to 6$	$\cdots$	$9+5 \to 1:4$
$0+6 \to 6$	$1+6 \to 7$	$\cdots$	$9+6 \to 1:5$
$0+7 \to 7$	$1+7 \to 8$	$\cdots$	$9+7 \to 1:6$
$0+8 \to 8$	$1+8 \to 9$	$\cdots$	$9+8 \to 1:7$
$0+9 \to 9$	$1+9 \to 1:0$	$\cdots$	$9+9 \to 1:8$
$x+y:z \to y:(x+z)$		$0:x \to x$	
$x:y+z \to x:(y+z)$		$x:(y:z) \to (x+y):z$	

Table 3.4: A TRS for addition of natural numbers in decimal notation.

the sum of 123 and 77 can be computed in $\mathcal{R}$:

$$\begin{aligned}\underline{(1:2):3+7:7} &\to 7:(\underline{(1:2):3+7}) \to 7:((1:2):(\underline{3+7})) \to 7:(\underline{(1:2):(1:0)})\\ &\to 7:((\underline{1:2+1}):0) \to 7:((1:(\underline{2+1})):0) \to \underline{7:((1:3):0)}\\ &\to (\underline{7+1:3}):0 \to (1:(\underline{7+3})):0 \to (\underline{1:(1:0)}):0\\ &\to ((\underline{1+1}):0):0 \to (2:0):0\end{aligned}$$

This is considerably more efficient than the computation of $123 + 77$ in the TRSs of Table 3.1.

Let $\mathcal{A}$ be the $\mathcal{F}$-algebra with carrier $\mathbb{N}$ and the natural interpretations for the digits $0-9$ and the function symbol $+$, together with $:_{\mathcal{A}}(m,n) = 10m+n$ for all $m, n \in \mathbb{N}$. One easily checks that $\ell =_{\mathcal{A}} r$ for all $\ell \to r \in \mathcal{R}$, so $\mathcal{A}$ is a model of $\mathcal{R}$. It is also not difficult to prove that the mapping $[\cdot]_{\mathcal{A}}$ is a bijection from the ground normal forms of $\mathcal{R}$ to the natural numbers. These two observations imply that a ground term cannot have different normal forms.

Lemma 3.2.10. *If $\mathcal{A}$ is a model of a* TRS *$\mathcal{R}$ such that $t_1 = t_2$ whenever t_1 and t_2 are ground normal forms with $t_1 =_{\mathcal{A}} t_2$, then every ground term has unique normal forms.*

Proof Let s be a ground term with normal forms t_1 and t_2. We show $t_1 = t_2$. We have $t_1 \leftrightarrow^* t_2$. We obtain $t_1 =_{\mathcal{A}} t_2$ from the fact that $\mathcal{A}$ is a model of $\mathcal{R}$. Since rewriting does not introduce variables, t_1 and t_2 are ground terms. The assumption of the lemma yields $t_1 = t_2$. □

In Chapter 4 we will show that $\mathcal{R}$ is terminating. Hence every ground term has a unique normal form. In particular $\mathcal{R}$ satisfies the property defined below. However, $\mathcal{R}$ is not confluent (Exercise 3.18(b)).

Definition 3.2.11. A TRS is *ground-confluent* if all its ground terms are confluent.

The TRS $\mathcal{R}$ of Table 3.5 implements the famous *Sieve of Eratosthenes* for generating the infinite list of all prime numbers.

Definition 3.2.12. Let $\mathcal{R}$ be a TRS. We define a relation $\mathrel{+\!\!\!\rightarrow}_{\mathcal{R}}$ on terms inductively as follows:

1. $x \mathrel{+\!\!\!\rightarrow}_{\mathcal{R}} x$ for all variables x,
2. $f(s_1, \ldots, s_n) \mathrel{+\!\!\!\rightarrow}_{\mathcal{R}} f(t_1, \ldots, t_n)$ if $s_i \mathrel{+\!\!\!\rightarrow}_{\mathcal{R}} t_i$ for all $1 \leqslant i \leqslant n$, and
3. $\ell\sigma \mathrel{+\!\!\!\rightarrow}_{\mathcal{R}} r\sigma$ if $\ell \to r \in \mathcal{R}$.

The relation $\mathrel{+\!\!\!\rightarrow}_{\mathcal{R}}$ is called *parallel rewriting*.

It is not difficult to see that $s \mathrel{+\!\!\!\rightarrow}_{\mathcal{R}} t$ if t can be obtained from s by contracting redexes at parallel positions in s.

Example 3.2.13. Consider the TRS $\mathcal{R}$ of Table 3.5 and the term

$$t = \mathsf{head}(\mathsf{from}(0) : \mathsf{filter}(0, \mathsf{from}(0), \mathsf{head}(0 : \mathsf{from}(0))))$$

We have $t \mathrel{+\!\!\!\rightarrow}_{\mathcal{R}} \mathsf{from}(0)$, $t \mathrel{+\!\!\!\rightarrow}_{\mathcal{R}} t$, and $t \mathrel{+\!\!\!\rightarrow}_{\mathcal{R}} \mathsf{head}((0 : \mathsf{from}(\mathsf{s}(0))) : \mathsf{filter}(0, 0 : \mathsf{from}(\mathsf{s}(0)), 0))$ but $t \mathrel{+\!\!\!\rightarrow}_{\mathcal{R}} 0 : \mathsf{from}(\mathsf{s}(0))$ does not hold.

The proof of the following basic result is left to the reader (Exercise 3.21(b)).

Lemma 3.2.14. *The inclusions* $\to_{\mathcal{R}} \subseteq \mathrel{+\!\!\!\rightarrow}_{\mathcal{R}} \subseteq \to_{\mathcal{R}}^*$ *hold for every* TRS $\mathcal{R}$.

Corollary 3.2.15. *A* TRS $\mathcal{R}$ *is confluent if* $\mathrel{+\!\!\!\rightarrow}_{\mathcal{R}}$ *has the diamond property.*

Proof This is an immediate consequence of Lemma 1.2.13 and Lemma 3.2.14. □

In a later chapter we will see that the above corollary applies to the TRS $\mathcal{R}$ of Table 3.5. Hence $\mathcal{R}$ has unique normal forms. Nevertheless, many terms do not have a normal form. The term $\mathsf{from}(0)$ for instance admits only the infinite rewrite sequence

$$\mathsf{from}(0) \to 0 : \mathsf{from}(\mathsf{s}(0)) \to 0 : (\mathsf{s}(0) : \mathsf{from}(\mathsf{s}(\mathsf{s}(0)))) \to \cdots$$

Some terms have a (unique) normal form but also allow infinite rewrite sequences. A typical example is the term $\mathsf{head}(\mathsf{tail}(\mathsf{tail}(\mathsf{primes})))$ which can be rewritten to the normal form $\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0)))))$—the third prime number—provided we adopt a good recipe for selecting redexes to contract. In Chapter 7 we will show that for a large class of TRSs, including the one of Table 3.5, repeatedly contracting all outermost redexes in parallel is guaranteed to find a normal form (if it exists), whereas always choosing an innermost redex for contraction will produce an infinite rewrite sequence if the term under consideration is not terminating.

$\mathsf{primes} \to \mathsf{sieve}(\mathsf{from}(\mathsf{s}(\mathsf{s}(0))))$	$\mathsf{sieve}(0 : y) \to \mathsf{sieve}(y)$
$\mathsf{from}(x) \to x : \mathsf{from}(\mathsf{s}(x))$	$\mathsf{sieve}(\mathsf{s}(x) : y) \to \mathsf{s}(x) : \mathsf{sieve}(\mathsf{filter}(x, y, x))$
$\mathsf{head}(x : y) \to x$	$\mathsf{filter}(0, y : z, w) \to 0 : \mathsf{filter}(w, z, w)$
$\mathsf{tail}(x : y) \to y$	$\mathsf{filter}(\mathsf{s}(x), y : z, w) \to y : \mathsf{filter}(x, z, w)$

Table 3.5: Sieve of Eratosthenes.

If a term lacks normal forms, infinite rewrite sequences may be meaningful. For instance, constant primes in the TRS of Table 3.5 admits an infinite rewrite sequence whose limit is the infinite term

$$\mathsf{s}(\mathsf{s}(0)) : (\mathsf{s}(\mathsf{s}(\mathsf{s}(0))) : (\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0))))) : (\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0))))))) : \cdots)))$$

representing the list of all prime numbers. In Section 14.3 we have more to say about this.

Exercises

3.12 Which of the following equations belong to the equational theory of the ES of Exercise 2.32?

a $(x \cdot (y^- \cdot x)^-) \cdot y \approx \mathsf{e}$

b $(x \cdot x^-) \cdot ((y^- \cdot (\mathsf{e}^- \cdot x))^- \cdot y^-) \approx (x^- \cdot \mathsf{e})^-$

c $(x^{--} \cdot (x \cdot (x \cdot \mathsf{e})^-))^- \approx x^{-----}$

3.13 Consider the ground ES $\mathcal{E}$ consisting of the equations

$$\mathsf{a} \approx \mathsf{b} \qquad \mathsf{f}(\mathsf{a}) \approx \mathsf{b} \qquad \mathsf{f}(\mathsf{b}) \approx \mathsf{c}$$

a Show that $\mathsf{a} \approx \mathsf{c}$ is valid in $\mathcal{E}$ by using the congruence closure algorithm.

b Show that $\mathcal{E}$ is consistent by giving a model of $\mathcal{E}$ in which the equation $\mathsf{f}(x) \approx x$ is not valid.

3.14 ***a*** Determine whether the equation $\mathsf{f}(\mathsf{a}) \approx \mathsf{a}$ is valid in the ES consisting of the two equations

$$\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{a}))) \approx \mathsf{a} \qquad \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{a}))))) \approx \mathsf{a}$$

b Is the restriction to ground terms s and t in Theorem 3.2.5 essential?

3.15 Consider the TRS of Table 3.3. Rewrite the term $\mathsf{ack}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0))), \mathsf{s}(\mathsf{s}(\mathsf{s}(0))))$ to normal form.

3.16 How many rewrite steps are required to add 123 and 77 in the TRSs of Table 3.1?

3.17 Let $\mathcal{A}$ be a model of a TRS $\mathcal{R}$ such that $t_1 = t_2$ whenever t_1 and t_2 are normal forms with $t_1 =_{\mathcal{A}} t_2$. Prove that $\mathcal{R}$ has unique normal forms with respect to conversion.

3.18 Consider the TRS $\mathcal{R}$ of Table 3.4.

a Is $\mathcal{R}$ a CS?

b Show that $\mathcal{R}$ is not confluent.

c Let $\mathcal{A}$ be the algebra described in the paragraph preceding Lemma 3.2.10. Prove that $[\cdot]_{\mathcal{A}}$ is a bijection from the ground normal forms of $\mathcal{R}$ to $\mathbb{N}$.

3.19 Construct a left-linear right-ground TRS that is ground-confluent but not confluent.

3.20 Consider the TRS $\mathcal{R}$ of Table 3.5.

a Consider the term t in Example 3.2.13. Compute all terms u such that $t \twoheadrightarrow_{\mathcal{R}} u$.

b Show that the term $\mathsf{head}(\mathsf{tail}(\mathsf{primes}))$ is normalizing but not terminating.

3.21 ***a*** Prove that $\twoheadrightarrow_{\mathcal{R}}$ is a reflexive rewrite relation for every TRS $\mathcal{R}$.

b Prove Lemma 3.2.14.

c Does the converse of Corollary 3.2.15 hold?

3.3 Undecidability

In this section we will show that basic properties like termination and confluence are undecidable for finite TRSs. These results are established by reductions from the halting

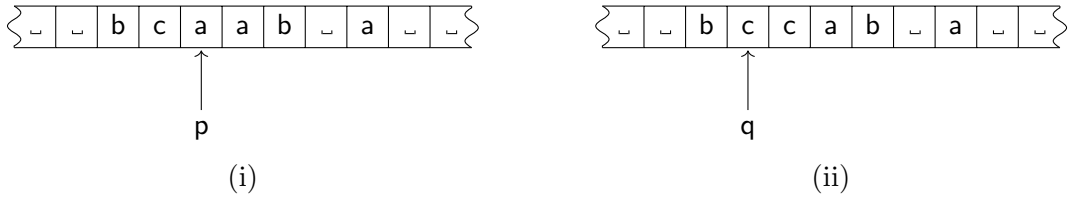


Figure 3.2: Configurations of a Turing machine.

problem for Turing machines and from Post's correspondence problem. In later chapters we give several conditions that ensure the decidability of many properties of TRSs. In the remainder of this section TRS will always mean finite TRS, unless explicitly stated otherwise.

We start with an intuitive description of Turing machines. A Turing machine consists of a two-way infinite tape which is divided into cells, a tape head that scans one cell at a time, and a finite control, see Figure 3.2(i). Each cell of the tape contains one symbol of a finite alphabet. There is one blank tape symbol, denoted by $\textvisiblespace$. At any time there is an infinity of blank symbols both to the left and to the right of the non-blank portion of the tape. A Turing machine operates as follows. Depending on the state of the finite control and the symbol scanned by the tape head, a Turing machine

- ▷ changes state,
- ▷ replaces the symbol scanned by the tape head by another symbol, and
- ▷ moves the tape head one cell to the left or to the right.

It is not required that the new state or the new tape symbol differ from the previous ones. On certain combinations of state and tape symbol, the Turing machine stops operating.

Definition 3.3.1. A *Turing machine* is a septuple $M = (Q, \Sigma, \Gamma, \textvisiblespace, \delta, s, F)$ consisting of a finite set Q of *states*, a finite set Σ of *input symbols*, a finite set $\Gamma \supseteq \Sigma$ of *tape symbols*, disjoint from Q, a special tape symbol $\textvisiblespace \in \Gamma$, named *blank*, a designated *start* state $s \in Q$, a set $F \subseteq Q$ of *final states*, and a *transition function* δ, which is a partial mapping from $(Q \setminus F) \times \Gamma$ to $Q \times \Gamma \times \{\mathsf{L}, \mathsf{R}\}$. A *configuration* is an element of $\Gamma^* Q \Gamma^*$, i.e., a string uqv with q a state and u, v strings of tape symbols.

Given a configuration uqv, the non-blank portion of the tape is contained in the string uv and the Turing machine scans the leftmost symbol of v. If $v = \epsilon$ then the tape head is positioned at a blank tape symbol with only blank symbols to its right. Similarly, if $u = \epsilon$ then the infinite portion of tape to the left of the cell scanned by the tape head is completely blank.

Definition 3.3.2. The transition function δ of a Turing machine $M = (Q, \Sigma, \Gamma, \textvisiblespace, \delta, s, F)$ determines a relation $\vdash_M$ on configurations as follows:

transition step	provided
$upav \vdash_M ubqv$	$\delta(p,a) = (q,b,\mathsf{R})$
$up \vdash_M ubq$	$\delta(p,\sqcup) = (q,b,\mathsf{R})$
$ucpav \vdash_M uqcbv$	$\delta(p,a) = (q,b,\mathsf{L})$
$pau \vdash_M q\sqcup bu$	$\delta(p,a) = (q,b,\mathsf{L})$
$ucp \vdash_M uqcb$	$\delta(p,\sqcup) = (q,b,\mathsf{L})$
$p \vdash_M q\sqcup b$	$\delta(p,\sqcup) = (q,b,\mathsf{L})$

Here $p,q \in Q$, $a,b,c \in \Gamma$ and $u,v \in \Gamma^*$.

Observe that for every configuration α there is at most one configuration β such that $\alpha \vdash_M \beta$. In other words, the transition relation $\vdash_M$ is deterministic.

Example 3.3.3. The situation of Figure 3.2(i) can be described by the configuration $\mathsf{bcpaab\sqcup a}$. If $\delta(\mathsf{p},\mathsf{a}) = (\mathsf{q},\mathsf{c},\mathsf{L})$ then $\mathsf{bcpaab\sqcup a} \vdash \mathsf{bqccab\sqcup a}$, i.e., the situation of Figure 3.2(ii) is obtained.

Associating with every Turing machine $M = (Q,\Sigma,\Gamma,\sqcup,\delta,s,F)$ the ARS $\mathcal{A}_M = \langle \Gamma^*Q\Gamma^*, \vdash_M \rangle$ enables us to use the terminology developed in Chapter 1. Since the transition relation $\vdash_M$ is deterministic, the ARS $\mathcal{A}_M$ is trivially confluent. Moreover, normalization and termination coincide for any configuration α. The well-known *halting problem* can now be rendered as follows.

Theorem 3.3.4. *The following problems are undecidable:*

1 *instance:* *a Turing machine M and a configuration α*
question: *is configuration α terminating?*

2 *instance:* *a Turing machine M*
question: *are all configurations terminating?*

The second problem is known as the *uniform* halting problem. Theorem 3.3.4 will be used to show that termination is an undecidable properties of (terms in) TRSs. The problem is how to view the ARS $\mathcal{A}_M$ as a (finite) TRS. First we add appropriate term structure to the set $\Gamma^*Q\Gamma^*$ of configurations.

Definition 3.3.5. Let $M = (Q,\Sigma,\Gamma,\sqcup,\delta,s,F)$ be a Turing machine. The signature $\mathcal{F}_M$ consists of the following symbols:

- ▷ a binary function symbol q for every state $q \in Q$,
- ▷ a unary function symbol a for every tape symbol $a \in \Gamma$,
- ▷ a constant ∞.

The constant ∞ will represent an infinite blank portion of tape. If $\alpha = a_1 \cdots a_n \in \Gamma^*$ and $t \in \mathcal{T}(\mathcal{F}_M)$ then $\alpha(t)$ denotes the term $a_1(\cdots(a_n(t))\cdots)$. We define a translation ϕ from configurations to terms in $\mathcal{T}(\mathcal{F}_M)$ by means of the equation

$$\phi(uqv) = q(u^-(\infty), v(\infty))$$

Here u^- denotes the reversal of u.

Example 3.3.6. For example, the configuration $\mathsf{q_0}$ corresponds to the term $\mathsf{q_0}(\infty,\infty)$ and configuration $\mathsf{aa}\sqcup\mathsf{qb}\sqcup$ to $\mathsf{q}(\sqcup(\mathsf{a}(\mathsf{a}(\infty))),\mathsf{b}(\sqcup(\infty)))$.

Next we associate a set of rewrite rules with every Turing machine.

Definition 3.3.7. Let $M = (Q,\Sigma,\Gamma,\sqcup,\delta,s,F)$ be a Turing machine. The TRS $\mathcal{R}_M$ over the signature $\mathcal{F}_M$ is defined as follows ($p,q \in Q$ and $a,b,c \in \Gamma$):

rewrite rule	provided
$p(x,a(y)) \to q(b(x),y)$	$\delta(p,a) = (q,b,\mathsf{R})$
$p(x,\infty) \to q(b(x),\infty)$	$\delta(p,\sqcup) = (q,b,\mathsf{R})$
$p(c(x),a(y)) \to q(x,c(b(y)))$	$\delta(p,a) = (q,b,\mathsf{L})$
$p(\infty,a(x)) \to q(\infty,\sqcup(b(x)))$	$\delta(p,a) = (q,b,\mathsf{L})$
$p(c(x),\infty) \to q(x,c(b(\infty)))$	$\delta(p,\sqcup) = (q,b,\mathsf{L})$
$p(\infty,\infty) \to q(\infty,\sqcup(b(\infty)))$	$\delta(p,\sqcup) = (q,b,\mathsf{L})$

Note the similarity between Definitions 3.3.2 and 3.3.7. Lemma 3.3.8 below is an immediate consequence of this observation. Observe that $\mathcal{R}_M$ contains finitely many rules.

Lemma 3.3.8. *Let M be a Turing machine. For all configurations α and β we have $\alpha \vdash_M \beta$ if and only if $\phi(\alpha) \to_{\mathcal{R}_M} \phi(\beta)$.*

Proof Straightforward by comparing the tables in Definitions 3.3.2 and 3.3.7. □

Corollary 3.3.9. *Let M be a Turing machine. For all configurations α the following statements are equivalent.*

1. *The configuration α is terminating.*
2. *The term $\phi(\alpha)$ is terminating.*

Combining Corollary 3.3.9 and the first part of Theorem 3.3.4 yields the following result.

Theorem 3.3.10. *The following problem is undecidable:*

instance: *a* TRS $\mathcal{R}$ *and a term* t

question: *is* t *terminating*?

Combinatory logic (CL) is a concrete example of a TRS in which termination of an arbitrary term is undecidable. This should not be confused with the decidability of termination for CL. (In Section 3.4 we will see that CL is not terminating.) Next we show that it is undecidable whether a given TRS is terminating. The second part of Theorem 3.3.4 cannot be applied directly since $\mathcal{R}_M$ contains terms that do not correspond to configurations. An example of such a term is $\sqcup(\mathsf{q}(\mathsf{a}(\mathsf{p}(\infty,x)),\mathsf{b}(\sqcup(\mathsf{q}(x,\infty)))))$. The next

lemma states that these ill-formed terms cause no problems. The proof is not difficult (Exercise 3.29). Later in this book we present a more general result which enables one to discard ill-formed terms when confronted with termination (or confluence) issues.

Lemma 3.3.11. *Let M be a Turing machine. If $\mathcal{R}_M$ is not terminating then there exists a configuration α such that $\phi(\alpha)$ is not terminating.*

Hence we obtain the following result.

Theorem 3.3.12. *The following problem is undecidable:*

instance: *a* TRS $\mathcal{R}$

question: *is $\mathcal{R}$ terminating?*

Proof Let M be an arbitrary Turing machine. According to Lemma 3.3.11 and Corollary 3.3.9 the TRS $\mathcal{R}_M$ is terminating if and only if every configuration α of the Turing machine M is terminating. Theorem 3.3.4 $\boxed{2}$ shows that this is undecidable. □

In Section 7.2 we prove a general result which, when applied to the present situation, shows that normalization and termination coincide for $\mathcal{R}_M$, both for individual terms and for $\mathcal{R}_M$ as a whole. Hence Theorems 3.3.10 and 3.3.12 apply also to normalization. Below we give an alternative proof of these results.

Lemma 3.3.13. *The* TRS $\mathcal{R}_M$ *is confluent, for any Turing machine M.*

Proof Corollary 3.2.15 applies, as will be shown in Chapter 6. □

So termination and normalization are undecidable properties of confluent TRSs. It is not difficult to modify the construction of $\mathcal{R}_M$ in order to conclude that confluence is an undecidable property of TRSs. Here we prove the same result by a reduction from the Post correspondence problem.

Definition 3.3.14. The *Post correspondence problem* (PCP for short) is the following problem:

instance: a finite non-empty set Γ and a finite subset P of $\Gamma^+ \times \Gamma^+$

question: does P have a solution?

Here a *solution* of P is a non-empty finite sequence of pairs $(\alpha_1, \beta_1), \ldots, (\alpha_n, \beta_n)$ in P such that $\alpha_1 \cdots \alpha_n = \beta_1 \cdots \beta_n$.

Example 3.3.15. Let $\Gamma = \{0, 1\}$. The PCP instance $P_1 = \{(11, 01), (00, 0), (0, 10)\}$ has a solution since the string 00110 can be partitioned as $\underline{00}\,\underline{11}\,\underline{0} = \underline{0}\,\underline{01}\,\underline{10}$. It is not difficult to see that the PCP instance $P_2 = \{(00, 0), (110, 01), (00, 100)\}$ has no solution.

Theorem 3.3.16. PCP *is undecidable.*

We now associate a TRS with every instance of PCP.

Definition 3.3.17. Let $P \subseteq \Gamma^+ \times \Gamma^+$ be an instance of PCP. The signature $\mathcal{F}_P$ consists of a unary function symbol a for every $a \in \Gamma$, a binary function symbol f, and constants start, stop, and c. If $\alpha = a_1 \cdots a_n \in \Gamma^+$ and $t \in \mathcal{T}(\mathcal{F}_P, \mathcal{V})$ then $\alpha(t)$ denotes the term $a_1(\cdots(a_n(t))\cdots)$. We define a TRS $\mathcal{R}_P$ over the signature $\mathcal{F}_P$ to consist of the following rewrite rules:

$$\begin{array}{lll} \mathsf{start} \to \mathsf{f}(\alpha(\mathsf{c}), \beta(\mathsf{c})) & \text{for all } (\alpha,\beta) \in P & \mathsf{f}(x,y) \to \mathsf{start} \\ \mathsf{f}(x,y) \to \mathsf{f}(\alpha(x), \beta(y)) & \text{for all } (\alpha,\beta) \in P & \mathsf{f}(x,x) \to \mathsf{stop} \end{array}$$

Lemma 3.3.18. *Let P be an arbitrary instance of* PCP. *The following statements are equivalent.*

1. *The* TRS $\mathcal{R}_P$ *is confluent.*
2. *The term* $\mathsf{f}(\mathsf{c},\mathsf{c})$ *is confluent.*
3. *The* TRS $\mathcal{R}_P$ *is normalizing.*
4. *The term* start *is normalizing.*
5. *The instance P has a solution.*

Proof

1 $\Longrightarrow$ 2 Trivial.

2 $\Longrightarrow$ 3 We have $\mathsf{start} \leftarrow \mathsf{f}(\mathsf{c},\mathsf{c}) \to \mathsf{stop}$. Since stop is a normal form, we obtain $\mathsf{start} \to^! \mathsf{stop}$ from the confluence of $\mathsf{f}(\mathsf{c},\mathsf{c})$. We show by structural induction that every term has a normal form. For variables and the constants stop and c this is obvious. The constant start rewrites to the normal form stop. Also terms of the form $\mathsf{f}(t_1,t_2)$ can be rewritten to stop: $\mathsf{f}(t_1,t_2) \to \mathsf{start} \to^! \mathsf{stop}$. Finally, a term of the form $a(t)$ with $a \in \Gamma$ rewrites to the normal form $a(t')$ where t' is a normal form of t, whose existence is guaranteed by the induction hypothesis.

3 $\Longrightarrow$ 4 Trivial.

4 $\Longrightarrow$ 5 It is easy to see that stop is the only normal form of start. A shortest rewrite sequence from start to stop must be of the form

$$\mathsf{start} \to \mathsf{f}(\alpha_1(\mathsf{c}), \beta_1(\mathsf{c})) \to^* \mathsf{f}(\alpha_n \cdots \alpha_1(\mathsf{c}), \beta_n \cdots \beta_1(\mathsf{c})) \to \mathsf{stop}$$

for some $n \geqslant 1$ with $(\alpha_i, \beta_i) \in P$ for $1 \leqslant i \leqslant n$. The last step is only possible if $\alpha_n \cdots \alpha_1(\mathsf{c}) = \beta_n \cdots \beta_1(\mathsf{c})$. Hence (α_n, β_n), …, (α_1, β_1) is a solution of P.

5 $\Longrightarrow$ 1 Suppose (α_1, β_1), …, (α_n, β_n) is a solution of P. So $\alpha_1 \cdots \alpha_n = \beta_1 \cdots \beta_n$ Hence start rewrites to stop:

$$\mathsf{start} \to \mathsf{f}(\alpha_n(\mathsf{c}), \beta_n(\mathsf{c})) \to^* \mathsf{f}(\alpha_1 \cdots \alpha_n(\mathsf{c}), \beta_1 \cdots \beta_n(\mathsf{c})) \to \mathsf{stop}$$

It follows that start, stop, and all terms of the form $\mathsf{f}(t_1,t_2)$ are confluent. Confluence of the remaining terms can be concluded by an obvious induction argument. □

Corollary 3.3.19. *The following two problems are undecidable:*

instance:	*a* TRS $\mathcal{R}$ *and a term* t	*instance:*	*a* TRS $\mathcal{R}$
question:	*is* t *confluent?*	*question:*	*is* $\mathcal{R}$ *confluent?*

Corollary 3.3.20. *The following two problems are undecidable:*

instance:	*a* TRS $\mathcal{R}$ *and a term* t	*instance:*	*a* TRS $\mathcal{R}$
question:	*is* t *normalizing?*	*question:*	*is* $\mathcal{R}$ *normalizing?*

It is not difficult to show that the statements in Lemma 3.3.18 are equivalent to the local confluence of $\mathcal{R}_P$ and the local confluence of $\mathsf{f}(\mathsf{c},\mathsf{c})$.

Corollary 3.3.21. *The following two problems are undecidable:*

instance:	*a* TRS $\mathcal{R}$ *and a term* t	*instance:*	*a* TRS $\mathcal{R}$
question:	*is* t *locally confluent?*	*question:*	*is* $\mathcal{R}$ *locally confluent?*

It is interesting to note that termination and confluence do not play a symmetric role with respect to decidability issues. We have seen that both properties are undecidable. Termination is undecidable for confluent TRSs, but confluence is decidable for terminating TRSs, as we will show in Chapter 5. Note that $\mathcal{R}_P$ is terminating if and only if $P = \varnothing$.

Exercises

3.22 For each of the following PCP instances P, determine whether the TRS $\mathcal{R}_P$ is normalizing.

a $\{(1,11),(10111,011)\}$

b $\{(10,101),(101,011),(011,11)\}$

c $\{(0,01),(101,1),(1,0)\}$

3.23 Prove that UN is an undecidable property of TRSs.

3.24 ***a*** Prove that confluence is an undecidable property of TRSs by a reduction from the halting problem for Turing machines.

b Prove that termination is an undecidable property of TRSs by a reduction from Post's correspondence problem.

3.25 Strengthen the undecidability results of Corollaries 3.3.19, 3.3.20, and 3.3.21 to left-linear TRSs by modifying the construction in Definition 3.3.17.

3.26 Prove that confluence is an undecidable property of locally confluent TRSs.

3.27 Prove that termination is an undecidable property of SRSs.

3.28 ***a*** For each of the relations $\to^*$, $\to^+$, $\downarrow$, $\uparrow$, $\leftrightarrow^*$, and $\to^!$ show that the following problem is undecidable:

instance: a TRS $\mathcal{R}$ and terms s and t

question: does the pair (s,t) belong to the relation?

b Do the problems remain undecidable if we restrict attention to terminating TRSs $\mathcal{R}$?

3.29 Prove Lemma 3.3.11.

3.30 In this exercise we show that termination is decidable for (finite) right-ground TRSs.

a Show that a TRS is terminating if and only if all its redexes are terminating.

$\mathsf{S}xyz \to xz(yz)$
$\mathsf{K}xy \to x$
$\mathsf{I}x \to x$

Table 3.6: Combinatory logic.

$\mathsf{B} = \mathsf{S}(\mathsf{KS})\mathsf{K}$	$\mathsf{W} = \mathsf{SS}(\mathsf{KI})$
$\mathsf{C} = \mathsf{S}(\mathsf{BBS})(\mathsf{KK})$	$\mathsf{Y} = \mathsf{B}(\mathsf{SI})(\mathsf{SII})(\mathsf{B}(\mathsf{SI})(\mathsf{SII}))$
	$\Omega = \mathsf{SII}(\mathsf{SII})$

Table 3.7: Some combinators.

b Show that a right-ground TRS is terminating if and only if all its right-hand sides are terminating.

c Let $\mathcal{R}$ be a finite right-ground TRS. Show by induction on the number of rewrite rules that if $\mathcal{R}$ is not terminating then there exist a right-hand side r and a context C such that $r \to^+ C[r]$.

d Conclude that termination is a decidable property of (finite) right-ground TRSs.

e Show that finiteness of $\mathcal{R}$ in part (c) is essential.

f Apply the decision procedure of part (d) to the TRS consisting of the rewrite rules

$$\begin{aligned} \mathsf{f}(\mathsf{f}(\mathsf{f}(x))) &\to \mathsf{g}(\mathsf{f}(\mathsf{g}(\mathsf{f}(\mathsf{a})))) & \mathsf{g}(\mathsf{g}(x)) &\to \mathsf{f}(\mathsf{f}(\mathsf{a})) \\ \mathsf{f}(\mathsf{g}(\mathsf{f}(x))) &\to \mathsf{g}(\mathsf{a}) & \mathsf{g}(\mathsf{a}) &\to \mathsf{g}(\mathsf{f}(\mathsf{a})) \end{aligned}$$

3.31 Is it decidable whether a given term is terminating with respect to a given right-ground TRS?

3.4 Combinatory Logic

In this section we study the ES combinatory logic of Table 2.1. Since its equations are rewrite rules, combinatory logic is a TRS, which we denote by CL. Usually one suppresses the $\cdot$ when writing CL-terms. Moreover, parentheses are omitted under the convention of *association to the left*, which means that missing parentheses are restored by always taking the leftmost possibility. For instance, xyz denotes the term $(x \cdot y) \cdot z$, whereas $x \cdot (y \cdot z)$ is written as $x(yz)$. The abridged notation for CL-terms is usually referred to as *applicative* notation, as opposed to the ordinary *functional* notation. With these conventions in mind, the rules of CL are presented as in Table 3.6.

A ground CL-term is called a *combinator*. Table 3.7 lists some useful combinators.

Example 3.4.1. We have $\mathsf{B}xyz \to^* x(yz)$, $\mathsf{C}xyz \to^* xzy$, and $\mathsf{W}xy \to^* xyy$:

$$\begin{aligned} \mathsf{B}xyz &= \mathsf{S}(\mathsf{KS})\mathsf{K}xyz \\ &\to \mathsf{KS}x(\mathsf{K}x)yz \\ &\to \mathsf{S}(\mathsf{K}x)yz \\ &\to \mathsf{K}xz(yz) \\ &\to x(yz) \end{aligned} \qquad \begin{aligned} \mathsf{C}xyz &= \mathsf{S}(\mathsf{BBS})(\mathsf{KK})xyz \\ &\to \mathsf{BBS}x(\mathsf{KK}x)yz \\ &\to \mathsf{BBS}x\mathsf{K}yz \\ &\to^+ \mathsf{B}(\mathsf{S}x)\mathsf{K}yz \\ &\to^+ \mathsf{S}x(\mathsf{K}y)z \\ &\to xz(\mathsf{K}yz) \\ &\to xzy \end{aligned} \qquad \begin{aligned} \mathsf{W}xy &= \mathsf{SS}(\mathsf{KI})xy \\ &\to \mathsf{S}x(\mathsf{KI}x)y \\ &\to \mathsf{S}x\mathsf{I}y \\ &\to xy(\mathsf{I}y) \\ &\to xyy \end{aligned}$$

In the previous chapter we remarked that combinatory logic has an undecidable validity problem, hence CL cannot be semi-complete (cf. Theorem 3.2.4). Below we show that CL is confluent, so CL is not normalizing. The simplest combinator which admits an infinite

rewrite sequence is Ω:

$$\Omega \;=\; \mathsf{SII}(\mathsf{SII}) \;\to\; \mathsf{I}(\mathsf{SII})(\mathsf{I}(\mathsf{SII})) \;\to\; \mathsf{SII}(\mathsf{I}(\mathsf{SII})) \;\to\; \mathsf{SII}(\mathsf{SII}) \;\to\; \cdots$$

One easily verifies that Ω has no normal form. Observe that Ω is a redex that does not contain other redexes, i.e., an innermost redex. In particular II is not a subterm of Ω, since $\mathsf{SII}(\mathsf{SII})$ stands for $((\mathsf{S}\cdot\mathsf{I})\cdot\mathsf{I})\cdot((\mathsf{S}\cdot\mathsf{I})\cdot\mathsf{I})$ in which $\mathsf{I}\cdot\mathsf{I}$ does not occur.

Theorem 3.4.2. CL *is confluent.*

The proof below uses the Z-property (Definition 1.2.14), in connection with the bullet function defined below. In Chapter 6 other proofs are presented, including one that shows the diamond property of $\twoheadrightarrow$.

Definition 3.4.3. We define the mappings $\diamond$ and $\star$ as follows:

$$t^\diamond \;=\; \begin{cases} u^\diamond \star v^\diamond & \text{if } t = uv \\ t & \text{otherwise} \end{cases} \qquad \text{with} \qquad s \star t \;=\; \begin{cases} t & \text{if } s = \mathsf{I} \\ u & \text{if } s = \mathsf{K}u \\ ut(vt) & \text{if } s = \mathsf{S}uv \\ st & \text{otherwise} \end{cases}$$

The function $\diamond$ contracts all redexes present in a CL term in addition to upward created redexes.

Example 3.4.4. Consider the peak

$$\mathsf{IIS} \;\leftarrow\; \mathsf{K}(\mathsf{IIS})(\mathsf{IK}(\mathsf{IIS})) \;\leftarrow\; \mathsf{SK}(\mathsf{IK})(\mathsf{IIS}) \;\to\; \mathsf{SKK}(\mathsf{IIS}) \;\to\; \mathsf{SKK}(\mathsf{IS})$$

By two applications of the function $\diamond$ to the starting term $\mathsf{SK}(\mathsf{IK})(\mathsf{IIS})$, a common reduct of IIS and $\mathsf{SKK}(\mathsf{IS})$ is obtained:

$$\begin{aligned} (\mathsf{SK}(\mathsf{IK})(\mathsf{IIS}))^{\diamond\diamond} \;&=\; ((\mathsf{SK}(\mathsf{IK}))^\diamond \star (\mathsf{IIS})^\diamond)^\diamond \;=\; (((\mathsf{SK})^\diamond \star (\mathsf{IK})^\diamond) \star ((\mathsf{II})^\diamond \star \mathsf{S}^\diamond))^\diamond \\ &=\; (((\mathsf{S}^\diamond \star \mathsf{K}^\diamond) \star (\mathsf{I}^\diamond \star \mathsf{K}^\diamond)) \star ((\mathsf{I}^\diamond \star \mathsf{I}^\diamond) \star \mathsf{S}))^\diamond \\ &=\; (((\mathsf{S}\star\mathsf{K}) \star (\mathsf{I}\star\mathsf{K})) \star ((\mathsf{I}\star\mathsf{I})\star\mathsf{S}))^\diamond \;=\; ((\mathsf{SK}\star\mathsf{K}) \star (\mathsf{I}\star\mathsf{S}))^\diamond \\ &=\; (\mathsf{SKK}\star\mathsf{S})^\diamond \;=\; (\mathsf{KS}(\mathsf{KS}))^\diamond \;=\; \mathsf{KS}\star\mathsf{KS} \;=\; \mathsf{S} \end{aligned}$$

and $\mathsf{IIS} \to \mathsf{IS} \to \mathsf{S} \leftarrow \mathsf{IS} \leftarrow \mathsf{K}(\mathsf{IS})(\mathsf{K}(\mathsf{IS})) \leftarrow \mathsf{SKK}(\mathsf{IS})$.

Proof (of Theorem 3.4.2) We employ the Z-property with respect to the bullet function of Definition 3.4.3. In Exercise 3.34 the reader is asked to prove the following preliminary statements, for all CL-terms s, t, u and v:

1. $st \to^= s\star t$,
2. $t \to^* t^\diamond$,
3. if $s \to^* t$ and $u \to^* v$ then $s\star u \to^* t\star v$.

We prove the statement

$$s \;\to^=\; t \;\;\Longrightarrow\;\; t \;\to^*\; s^\diamond \;\to^*\; t^\diamond \tag{3.1}$$

by induction on s. If $s = t$ then the conclusion is an immediate consequence of $\boxed{2}$. Suppose $s \to t$. So s is an application $s_1 s_2$. If $s \to t$ is a root step then there are three possibilities for s_1. If $s_1 = \mathsf{S}uv$ then $s^\diamond = (\mathsf{S}uv)^\diamond \star s_2^\diamond = (\mathsf{S}u^\diamond v^\diamond \star s_2^\diamond = u^\diamond s_2^\diamond (v^\diamond s_2^\diamond)$ and $t = us_2(vs_2) = s_1 \star s_2 \to^* s_1^\diamond \star s_2^\diamond = s^\diamond$ by two applications of $\boxed{2}$. Since $u^\diamond s_2^\diamond (v^\diamond s_2^\diamond) \to^* (u^\diamond \star s_2^\diamond) \star (v^\diamond \star s_2^\diamond) = t^\diamond$ by three applications of $\boxed{1}$, we obtain $s^\diamond \to^* t^\diamond$. If $s_1 = \mathsf{K}u$ then $s^\diamond = (\mathsf{K}u)^\diamond \star s_2^\diamond = (\mathsf{K}u^\diamond) \star s_2^\diamond = u^\diamond$ and $t = u$. Hence $t \to^* s^\diamond = t^\diamond$ follows from $\boxed{2}$. If $s_1 = \mathsf{I}$ then $s^\diamond = s_2^\diamond$ and $t = s_2$. Hence $t \to^* s^\diamond = t^\diamond$ follows from $\boxed{2}$. Next we consider the case that $s \to t$ is not a root step. Then $t = t_1 t_2$ with $s_1 \to^= t_1$ and $s_2 \to^= t_2$. We obtain $t_1 \to^* s_1^\diamond \to^* t_1^\diamond$ and $t_2 \to^* s_2^\diamond \to^* t_2^\diamond$ from the induction hypothesis. Hence $t \to^* s_1^\diamond s_2^\diamond \to^= s_1^\diamond \star s_2^\diamond = s^\diamond$ by $\boxed{1}$ and $s^\diamond = s_1^\diamond \star s_2^\diamond \to^* t_1^\diamond \star t_2^\diamond = t^\diamond$ by $\boxed{3}$. This completes the proof of (3.1). It follows that CL has the Z-property with respect to $\diamond$. Hence CL is confluent according to Lemma 1.2.17. □

Like the TRS of Table 3.5, CL admits terms that are normalizing but not terminating. Specializing the strategy $\mathcal{S}_\bullet$ of Definition 1.5.11 to the function $\diamond$ in the proof of Theorem 3.4.2 yields a normalizing strategy for CL according to Theorem 1.5.12.

Example 3.4.5. The CL term $\mathsf{S}(\mathsf{KK})(\mathsf{KI})\Omega$ is not terminating due to the occurrence of Ω. Its (unique) normal form KI is obtained in two $\mathcal{S}_\diamond$ steps:

$$\mathsf{S}(\mathsf{KK})(\mathsf{KI})\Omega \to^* (\mathsf{S}(\mathsf{KK})(\mathsf{KI})\Omega)^\diamond = \mathsf{KK}\Omega^\diamond(\mathsf{KI}\Omega^\diamond) \to^* (\mathsf{KK}\Omega^\diamond(\mathsf{KI}\Omega^\diamond))^\diamond = \mathsf{KI}$$

Repeatedly selecting the leftmost of the outermost redexes in a normalizing CL-term is guaranteed to also result in a normal form. This *leftmost outermost* strategy is actually hyper-normalizing (Definition 1.5.3) for CL, a result that will be proved in Chapter 7.

It may come as a surprise that an apparently simple TRS like CL can represent *all* computable functions. So CL has the same expressive power as (the TRS $\mathcal{R}_M$ of Definition 3.3.7 for) a *universal* Turing machine M. The following definition explains how functions on natural numbers are represented in CL.

Definition 3.4.6. For every $n \in \mathbb{N}$ the *Church numeral* $\underline{n}$ is the combinator inductively defined as follows:

$$\underline{n} = \begin{cases} \mathsf{KI} & \text{if } n = 0 \\ \mathsf{SB}\,\underline{n-1} & \text{if } n > 0 \end{cases}$$

A (partial) function $f\colon \mathbb{N}^n \to \mathbb{N}$ is CL-*representable* if there exists a combinator F such that

$$\begin{aligned} f(x_1, \ldots, x_n) = y \quad &\Longrightarrow \quad F\,\underline{x_1} \cdots \underline{x_n} \to^* \underline{y} \\ f(x_1, \ldots, x_n) \text{ is undefined} \quad &\Longrightarrow \quad F\,\underline{x_1} \cdots \underline{x_n} \text{ has no normal form} \end{aligned}$$

for all $x_1, \ldots, x_n, y \in \mathbb{N}$.

Note that Church numerals are normal forms. In the exercises we will develop the fundamental result that the CL-representable functions coincide with the partial recursive functions. Two important ingredients, abstraction and fixed point combinators, are defined below.

Definition 3.4.7. Let t be a CL-term and x a variable. We define the CL-term $[x]t$ inductively as follows:

$$[x]t = \begin{cases} \mathsf{I} & \text{if } t = x \\ \mathsf{K}t & \text{if } x \notin \mathcal{V}\mathsf{ar}(t) \\ \mathsf{S}([x]u)([x]v) & \text{if } t = uv \text{ and } x \in \mathcal{V}\mathsf{ar}(t) \end{cases}$$

We write $[x_1, \ldots, x_n]t$ for $[x_1](\ldots[x_n]t\ldots)$.

Example 3.4.8. We compute $[x, y, z](xzy)$:

$$\begin{aligned}
[x,y,z](xzy) &= [x,y](\mathsf{S}([z](xz))([z]y)) \\
&= [x,y](\mathsf{S}(\mathsf{S}([z]x)([z]z))(\mathsf{K}y)) \\
&= [x,y](\mathsf{S}(\mathsf{S}(\mathsf{K}x)\mathsf{I})(\mathsf{K}y)) \\
&= [x](\mathsf{S}([y](\mathsf{S}(\mathsf{S}(\mathsf{K}x)\mathsf{I})))([y](\mathsf{K}y))) \\
&= [x](\mathsf{S}(\mathsf{K}(\mathsf{S}(\mathsf{S}(\mathsf{K}x)\mathsf{I})))(\mathsf{S}([y]\mathsf{K})([y]y))) \\
&= [x](\mathsf{S}(\mathsf{K}(\mathsf{S}(\mathsf{S}(\mathsf{K}x)\mathsf{I})))(\mathsf{S}(\mathsf{KK})\mathsf{I})) \\
&= \mathsf{S}([x](\mathsf{S}(\mathsf{K}(\mathsf{S}(\mathsf{S}(\mathsf{K}x)\mathsf{I}))))(\mathsf{K}(\mathsf{S}(\mathsf{KK})\mathsf{I}))) \\
&= \mathsf{S}(\mathsf{S}(\mathsf{KS})([x](\mathsf{K}(\mathsf{S}(\mathsf{S}(\mathsf{K}x)\mathsf{I})))))(\mathsf{K}(\mathsf{S}(\mathsf{KK})\mathsf{I})) \\
&= \mathsf{S}(\mathsf{S}(\mathsf{KS})(\mathsf{S}(\mathsf{KK})([x](\mathsf{S}(\mathsf{S}(\mathsf{K}x)\mathsf{I})))))(\mathsf{K}(\mathsf{S}(\mathsf{KK})\mathsf{I})) \\
&= \mathsf{S}(\mathsf{S}(\mathsf{KS})(\mathsf{S}(\mathsf{KK})(\mathsf{S}(\mathsf{KS})([x](\mathsf{S}(\mathsf{K}x)\mathsf{I})))))(\mathsf{K}(\mathsf{S}(\mathsf{KK})\mathsf{I})) \\
&= \mathsf{S}(\mathsf{S}(\mathsf{KS})(\mathsf{S}(\mathsf{KK})(\mathsf{S}(\mathsf{KS})(\mathsf{S}([x](\mathsf{S}(\mathsf{K}x)))(\mathsf{KI})))))(\mathsf{K}(\mathsf{S}(\mathsf{KK})\mathsf{I})) \\
&= \mathsf{S}(\mathsf{S}(\mathsf{KS})(\mathsf{S}(\mathsf{KK})(\mathsf{S}(\mathsf{KS})(\mathsf{S}(\mathsf{S}(\mathsf{KS})([x](\mathsf{K}x)))(\mathsf{KI})))))(\mathsf{K}(\mathsf{S}(\mathsf{KK})\mathsf{I})) \\
&= \mathsf{S}(\mathsf{S}(\mathsf{KS})(\mathsf{S}(\mathsf{KK})(\mathsf{S}(\mathsf{KS})(\mathsf{S}(\mathsf{S}(\mathsf{KS})(\mathsf{S}(\mathsf{KK})\mathsf{I}))(\mathsf{KI})))))(\mathsf{K}(\mathsf{S}(\mathsf{KK})\mathsf{I}))
\end{aligned}$$

It is easy to prove that the variable x no longer occurs in $[x]t$.

Lemma 3.4.9. *We have $([x]t)x \to^* t$ for every CL-term t and variable x.*

Proof We use induction on t. If $t = x$ then $([x]t)x = \mathsf{I}x \to x = t$. If $x \notin \mathcal{V}\mathsf{ar}(t)$ then $([x]t)x = \mathsf{K}tx \to t$. Finally, if $t = uv$ and $x \in \mathcal{V}\mathsf{ar}(t)$ then we obtain $([x]u)x \to^* u$ and $([x]v)x \to^* v$ from the induction hypothesis and hence $([x]t)x = \mathsf{S}([x]u)([x]v)x \to ([x]u)x(([x]v)x) \to^* uv = t$. □

As illustrated in Example 3.4.8, abstraction produces rather large CL-terms. A more efficient version using the combinators B and C is presented in Exercise 3.40.

Definition 3.4.10. A CL-term s is called a *fixed point* of a CL-term t if $ts \leftrightarrow^* s$. A *fixed point combinator* is any combinator t that satisfies $tx \leftrightarrow^* x(tx)$. Here x is an arbitrary variable.

Lemma 3.4.11. *The combinator Y is a fixed point combinator.*

Proof In Example 3.4.1 we observed that $\mathsf{B}xyz \to^+ x(yz)$. Hence

$$\begin{aligned}\mathsf{Y}x &\to^+ \mathsf{SI}(\mathsf{SII}(\mathsf{B}(\mathsf{SI})(\mathsf{SII})))x \to \mathsf{I}x(\mathsf{SII}(\mathsf{B}(\mathsf{SI})(\mathsf{SII}))x)\\ &\to^+ x(\mathsf{I}(\mathsf{B}(\mathsf{SI})(\mathsf{SII}))(\mathsf{I}(\mathsf{B}(\mathsf{SI})(\mathsf{SII})))x)\\ &\to^+ x(\mathsf{B}(\mathsf{SI})(\mathsf{SII})(\mathsf{B}(\mathsf{SI})(\mathsf{SII}))x) = x(\mathsf{Y}x)\end{aligned}$$

□

From the proof we observe that Y satisfies the stronger property $\mathsf{Y}x \to^+ x(\mathsf{Y}x)$.

Definition 3.4.12. A set A of CL-terms is *conversion-closed* (or *closed under conversion*) if $t \in A$ whenever $s \in A$ and $s \leftrightarrow^* t$.

The following result has several important consequences.

Theorem 3.4.13. *No pair of non-empty conversion-closed sets of* CL*-terms is recursively separable.*

So if A and B are non-empty sets of CL-terms closed under conversion, then there is no decision procedure that can tell A and B apart. More precisely, there is no recursive set C such that $A \subseteq C$ and $B \cap C = \varnothing$. (Here we assume that the set of CL-terms is recursively enumerable, which follows if the set of variables is recursively enumerable.)

Corollary 3.4.14. *Any non-trivial conversion-closed set of* CL*-terms has an undecidable membership problem.*

Non-trivial means that the set is neither empty nor the set of all CL-terms.

Proof Let A be a non-trivial set of conversion-closed CL-terms and let B be its complement. It is easy to see that also B is non-trivial and conversion-closed. According to Theorem 3.4.13 A and B are recursively inseparable. Hence A cannot be recursive and thus the membership problem for A is undecidable. □

Corollary 3.4.15. *The validity problem for* CL *is undecidable.*

Proof Let A be set of all CL-terms that are convertible with I. Because I and K are different normal forms and CL is confluent, $\mathsf{K} \notin A$. Hence A is non-trivial and thus the decision problem

instance: a CL-term t

question: $t \leftrightarrow^* \mathsf{I}$?

is undecidable as a consequence of Corollary 3.4.14. It follows that the more general validity problem for CL is undecidable as well. □

Whereas the previous undecidability results were about conversion, the final one is about rewriting.

Theorem 3.4.16. *The following two problems are undecidable:*

instance:	*a* CL*-term* t	*instance:*	*a* CL*-term* t
question:	*is* t *normalizing?*	*question:*	*is* t *terminating?*

Proof Let A be the set of CL-terms that are convertible to a normal form of CL. We already observed that CL is not normalizing and hence A is non-trivial. Since a CL-term is normalizing if and only if it belongs to A, the first result follows from Corollary 3.4.14. The second result is considerably harder to prove (Exercise 3.47). □

Exercises

3.32 ***a*** Write the following combinators in applicative notation.

- ▷ $(\mathsf{S} \cdot ((\mathsf{S} \cdot \mathsf{S}) \cdot (\mathsf{S} \cdot \mathsf{S}))) \cdot (\mathsf{S} \cdot \mathsf{S})$
- ▷ $(((\mathsf{S} \cdot \mathsf{S}) \cdot (\mathsf{S} \cdot \mathsf{S})) \cdot ((\mathsf{S} \cdot \mathsf{S}) \cdot \mathsf{S})) \cdot \mathsf{S}$

b Write the following combinators in functional notation.

- ▷ $\mathsf{K}(\mathsf{IS})\mathsf{S}$
- ▷ $(\mathsf{SIS}(\mathsf{I}(\mathsf{SI})\mathsf{S})\mathsf{I})\mathsf{SI}(\mathsf{S}(\mathsf{ISI}))$

c Rewrite the following combinators to normal form.

- ▷ SSSS
- ▷ SSSSS
- ▷ $\mathsf{SS}(\mathsf{SS})(\mathsf{SS})$

3.33 ***a*** Show that the combinator Ω has no normal form.

b Find a CL-term that is weakly but not strongly normalizing.

3.34 ***a*** Compute $t^\diamond$ for the following combinators t.

- ▷ IIIIK
- ▷ KISS
- ▷ $\mathsf{SS}(\mathsf{SS})(\mathsf{SS})$

b Show $st \to^= s \star t$ for all CL-terms s and t.

c Show $t \to^* t^\diamond$ for all CL-terms t.

d Suppose $s \to^* t$ and $u \to^* v$. Show $s \star u \to^* t \star v$.

3.35 Which of the following statements hold?

a $\mathsf{SB}(\mathsf{KI}) \leftrightarrow^* \mathsf{KI}$

b $\mathsf{SKK} \leftrightarrow^* \mathsf{I}$

c $\mathsf{SKK}x \leftrightarrow^* \mathsf{I}x$

d $\mathsf{SII}(\mathsf{SII}) \leftrightarrow^* \mathsf{Y}$

3.36 In this exercise we show that CL has the Z-property with respect to the following bullet function:

$$t^\Delta = \begin{cases} u^\Delta w^\Delta (v^\Delta w^\Delta) & \text{if } t = \mathsf{S}uvw \\ u^\Delta & \text{if } t = \mathsf{K}uv \\ u^\Delta & \text{if } t = \mathsf{I}u \\ u^\Delta v^\Delta & \text{if } t = uv \text{ is not a redex} \\ t & \text{otherwise} \end{cases}$$

a Show $s^\Delta t^\Delta \to^= (st)^\Delta$ for all CL-terms s and t.

b Show $t \to^* t^\Delta$ for all CL-terms t.

c Show $t \to^* s^\Delta \to^* t^\Delta$ whenever $s \to^= t$.

3.37 ***a*** Show that every CL-term has a fixed point.

b Show that $\mathsf{SSI(S(S(KS)K)(K(SII)))}$ is a fixed point combinator.

c Suppose t is a combinator that satisfies $t \leftrightarrow^* \mathsf{S}\mathsf{I}t$. Show that t is a fixed point combinator.

3.38 Let t be a CL-term and suppose that $\mathcal{V}\mathsf{ar}(t) \subseteq \{x_1, \ldots, x_n\}$.

a Construct a combinator s such that $s x_1 \cdots x_n \to^* t$.

b Construct a combinator s' such that $s' x_2 \cdots x_n \to^* t\{x_1 \mapsto s'\}$.

3.39 ***a*** Construct a combinator M with the property $\mathsf{M} \to^* \mathsf{MMM}$.

b Construct a combinator Z that satisfies $\mathsf{Z}x_1 \cdots x_n \to^* \mathsf{Z}$ for all $n \geqslant 1$.

c Show that there is no combinator Q satisfying $\mathsf{Q}x(yz) \leftrightarrow^* xyz$.

3.40 Let t be a CL-term and x a variable. We inductively define the CL-term $\langle x \rangle t$ as follows:

$$\langle x \rangle t \;=\; \begin{cases} \mathsf{I} & \text{if } t = x \\ \mathsf{K}t & \text{if } x \notin \mathcal{V}\mathsf{ar}(t) \\ u & \text{if } t = ux \text{ and } x \notin \mathcal{V}\mathsf{ar}(u) \\ \mathsf{B}u(\langle x \rangle v) & \text{if } t = uv \text{ and } x \notin \mathcal{V}\mathsf{ar}(u) \\ \mathsf{C}(\langle x \rangle u)v & \text{if } t = uv \text{ and } x \notin \mathcal{V}\mathsf{ar}(v) \\ \mathsf{S}(\langle x \rangle u)(\langle x \rangle v) & \text{if } t = uv \end{cases}$$

a Compute $\langle x, y, z \rangle (xzy)$.

b Show that $(\langle x \rangle t)x \to^* t$ for every CL-term t and variable x.

3.41 ***a*** Construct a combinator X such that $\mathsf{X}\underline{n} \to^* \underline{n+1}$ for all $n \geqslant 0$.

b Show that the combinator $\mathsf{P} = \mathsf{C(C(CI(B(CI)(CI(SB))))(K(KI)))I}$ satisfies

$$\mathsf{P}\underline{n} \to^* \begin{cases} \underline{0} & \text{if } n = 0 \\ \underline{n-1} & \text{if } n > 0 \end{cases}$$

c Construct a combinator A such that $\mathsf{A}\underline{m}\,\underline{n} \to^* \underline{m+n}$ for all $m, n \geqslant 0$.

d Construct a combinator $\mathsf{zero?}$ such that

$$\mathsf{zero?}\,\underline{n} \to^* \begin{cases} \mathsf{K} & \text{if } n = 0 \\ \mathsf{KI} & \text{otherwise} \end{cases}$$

3.42 Show that the notion of CL-representable function in Definition 3.4.6 is not affected if we replace the two implications by equivalences.

3.43 A class $\mathcal{C}$ of numeric functions is closed under *composition* if the function $f\colon \mathbb{N}^n \to \mathbb{N}$ defined by

$$f(\vec{x}) \;=\; g(h_1(\vec{x}), \ldots, h_m(\vec{x}))$$

belongs to $\mathcal{C}$ whenever the functions $g\colon \mathbb{N}^m \to \mathbb{N}$ and $h_1, \ldots, h_m\colon \mathbb{N}^n \to \mathbb{N}$ belong to $\mathcal{C}$. Prove that the class of total CL-representable functions is closed under composition.

3.44 A class $\mathcal{C}$ of numeric functions is closed under *primitive recursion* if the function $f\colon \mathbb{N}^{n+1} \to \mathbb{N}$ defined by the recursive equations

$$\begin{aligned} f(0, \vec{y}) \;&=\; g(\vec{y}) \\ f(x+1, \vec{y}) \;&=\; h(f(x, \vec{y}), x, y_1, \ldots, y_n) \end{aligned}$$

belongs to $\mathcal{C}$ whenever $g\colon \mathbb{N}^m \to \mathbb{N}$ and $h\colon \mathbb{N}^{m+2} \to \mathbb{N}$ belong to $\mathcal{C}$. Prove that the class of total CL-representable functions is closed under primitive recursion.

3.45 The class of *primitive recursive functions* is the smallest class of numeric functions that contains the following *initial* functions

1. the zero function $x \mapsto 0$
2. the successor function $x \mapsto x + 1$
3. the projection functions $x_1, \ldots, x_n \mapsto x_i$ for all $0 < i \leqslant n$

and is closed under composition and primitive recursion. Prove that primitive recursive functions are CL-representable.

3.46 A class $\mathcal{C}$ of numeric functions is closed under *minimization* if the function $f\colon \mathbb{N}^n \to \mathbb{N}$ defined by

$$f(\vec{x}) \;=\; (\mu\, y)(g(y, \vec{x}) = 0)$$

belongs to $\mathcal{C}$ whenever the function $g\colon \mathbb{N}^{n+1} \to \mathbb{N}$ belongs to $\mathcal{C}$. Here $(\mu\, y)(g(y, \vec{x}) = 0)$ denotes the smallest $y \in \mathbb{N}$ such that $g(y, \vec{x}) = 0$ and $g(z, \vec{x}) > 0$ for all $z < y$. (In particular, $g(z, \vec{x})$ is defined for all $z < y$.) The class of *recursive functions* is the smallest class of total numeric functions that contains the initial functions and is closed under composition, primitive recursion, and minimization. Prove that recursive functions are CL-representable.

3.47 Prove that the second decision problem in Theorem 3.4.16 is undecidable.

3.48 The class of *partial recursive functions* is the smallest class of (partial) numeric functions that contains the initial functions and is closed under composition, primitive recursion, and minimization.

a Prove that partial recursive functions are CL-representable.

b Prove the converse.

Bibliographic Notes

Huet and Lankford [59] proved the undecidability of termination. Dauchet [23] showed that even for TRSs consisting of a single rewrite rule, termination is undecidable. Huet and Lankford [59] showed the decidability of termination for ground TRSs. The stronger result in Exercise 3.30, the decidability of termination for right-ground TRSs, is due to Dershowitz [26]. Also confluence is decidable for right-ground TRSs. This much harder result was independently obtained by Kaiser [64] and Tiwari *et al.* [46]. For (left-linear and right-)ground TRSs, relatively simple tree automata techniques suffice to decide confluence ([24]), which we cover in Chapter 8. Confluence for ground TRSs should not be confused with ground-confluence. Kapur *et al.* [68] showed that ground-confluence is an undecidable property of terminating TRSs. The study of CL was initiated by Schönfinkel [118] in an attempt to eliminate bound variables from predicate logic. CL was rediscovered by Curry [20]. The theory of CL is extensively studied in Curry *et al.* [21, 22]. For a gentle but formal introduction to CL and λ-calculus, the interested reader is referred to Hindley and Seldin [53]. The proof of Theorem 3.4.2 is based on [104]. Theorem 3.4.13 is due to Scott [119]. Exercise 3.38, the so-called *combinatorial completeness* of CL, is due to Curry [20]. The use of (the combinatorial completeness of) CL for implementing functional programming languages was pioneered by Turner [132].

Chapter 4

Termination

One of the main problems in the theory of term rewrite systems is the detection of termination. In this chapter we introduce several methods for establishing termination. In Section 4.1 we equip algebras with well-founded orders to obtain a complete semantic characterization of termination. Polynomial interpretations are an instance of this general framework. In Sections 4.3 and 4.4 we introduce lexicographic path orders and Knuth–Bendix orders. Well-foundedness of these orders follows from the subterm property, which is covered in Section 4.2. In Section 4.5 we present the basics of the dependency pair method, a powerful termination method which is particularly suited for automation.

4.1 Reduction Orders

We start with a simple observation.

Lemma 4.1.1. *A* TRS $\mathcal{R}$ *is terminating if and only if there exists a well-founded order* $>$ *on terms such that* $\to_{\mathcal{R}} \subseteq >$.

Proof

$\Longrightarrow$ If $\mathcal{R}$ is terminating then $\to_{\mathcal{R}}^{+}$ is a well-founded order on terms which contains $\to_{\mathcal{R}}$.

$\Longleftarrow$ If $\mathcal{R}$ is not terminating then there exists an infinite rewrite sequence $t_0 \to_{\mathcal{R}} t_1 \to_{\mathcal{R}} \cdots$ which, because $\to_{\mathcal{R}}$ is included in $>$, gives rise to an infinite descending sequence $t_0 > t_1 > \cdots$, contradicting the well-foundedness of $>$. □

Example 4.1.2. Consider the SRS $\mathcal{R}$ consisting of the single rewrite rule $\mathsf{aa} \to \mathsf{b}$. Define the relation $>$ on terms as follows: $s > t$ if $|s| > |t|$. Clearly $>$ is a well-founded order. Let $s \to_{\mathcal{R}} t$ be an arbitrary rewrite step, so $s = u\mathsf{aa}v$ and $t = u\mathsf{b}v$ for some strings u and v. We have $|s| = |u| + 2 + |v|$ and $|t| = |u| + 1 + |v|$, hence $s > t$. According to the preceding lemma the TRS $\mathcal{R}$ is terminating.

The test $\to_{\mathcal{R}} \subseteq >$ in Lemma 4.1.1, necessary to conclude termination, is not very convenient since even TRSs having only finitely many rewrite rules admit in general infinitely many rewrite steps. According to Theorem 4.1.4 below, to conclude termination it is sufficient to test the orientability of the rewrite rules, provided we require that the well-founded order is compatible with the term structure.

Definition 4.1.3. A proper order on terms that is also a rewrite relation is called a *rewrite order*. A *reduction order* is a well-founded rewrite order. A TRS $\mathcal{R}$ and a binary relation R on terms are called *compatible* if $\ell \mathrel{R} r$ for every rewrite rule $\ell \to r \in \mathcal{R}$. We also say that $\mathcal{R}$ is compatible with R or that R is compatible with $\mathcal{R}$.

Theorem 4.1.4. *A TRS $\mathcal{R}$ is terminating if and only if it is compatible with a reduction order.*

Proof

$\Longrightarrow$ The relation $\to_{\mathcal{R}}^+$ is closed under contexts and substitutions for any TRS $\mathcal{R}$. Hence it is a reduction order for terminating TRSs. Clearly $\ell \to_{\mathcal{R}}^+ r$ for every rewrite rule $\ell \to r \in \mathcal{R}$.

$\Longleftarrow$ According to Lemma 4.1.1 it suffices to show $\to_{\mathcal{R}} \subseteq >$. By assumption $\ell > r$ for every rewrite rule $\ell \to r \in \mathcal{R}$. Since $>$ is a rewrite relation we have $C[\ell\sigma] > C[r\sigma]$ for all contexts C and substitutions σ. Hence $\to_{\mathcal{R}} \subseteq >$. □

Lemma 4.1.5. *A TRS $\mathcal{R}$ and a rewrite order $>$ are compatible if and only if $\to_{\mathcal{R}}^+$ is contained in $>$.*

Proof

$\Longrightarrow$ By definition $\ell > r$ for every rewrite rule $\ell \to r \in \mathcal{R}$. Since $>$ is a rewrite relation we have $C[\ell\sigma] > C[r\sigma]$ for all contexts C and substitutions σ. Hence $\to_{\mathcal{R}} \subseteq >$. Transitivity of $>$ yields the desired $\to_{\mathcal{R}}^+ \subseteq >$.

$\Longleftarrow$ Every rewrite rule $\ell \to r \in \mathcal{R}$ satisfies $\ell \to_{\mathcal{R}}^+ r$ and thus also $\ell > r$. Therefore $\mathcal{R}$ and $>$ are compatible. □

According to Theorem 4.1.4 termination of a TRS $\mathcal{R}$ is equivalent to the existence of a compatible reduction order. We show that $\mathcal{F}$-algebras are a convenient tool for constructing reduction orders on $\mathcal{T}(\mathcal{F}, \mathcal{V})$.

Definition 4.1.6. A *monotone* $\mathcal{F}$-algebra $(\mathcal{A}, >)$ consists of a non-empty $\mathcal{F}$-algebra $\mathcal{A}$ and a proper order $>$ on the carrier A of $\mathcal{A}$ such that every algebra operation is strictly monotone in all its coordinates, i.e., if $f \in \mathcal{F}$ has arity $n \geqslant 1$ then

$$f_{\mathcal{A}}(a_1, \ldots, a_i, \ldots, a_n) > f_{\mathcal{A}}(a_1, \ldots, b, \ldots, a_n)$$

for all $a_1, \ldots, a_n, b \in A$ and $i \in \{1, \ldots, n\}$ with $a_i > b$. We call a monotone $\mathcal{F}$-algebra $(\mathcal{A}, >)$ *well-founded* if $>$ is well-founded.

Definition 4.1.7. Let $(\mathcal{A}, >)$ be a monotone algebra. We define a relation $>_{\mathcal{A}}$ on terms as follows: $s >_{\mathcal{A}} t$ if $[\alpha]_{\mathcal{A}}(s) > [\alpha]_{\mathcal{A}}(t)$ for all assignments α.

We now show that the relation $>_{\mathcal{A}}$ just defined is a rewrite order.

Lemma 4.1.8. *If $(\mathcal{A}, >)$ is a monotone algebra then $>_{\mathcal{A}}$ is a rewrite order. Moreover, if $(\mathcal{A}, >)$ is in addition well-founded then $>_{\mathcal{A}}$ is a reduction order.*

Proof Let $(\mathcal{A}, >)$ be a monotone algebra. The relation $>_\mathcal{A}$ clearly inherits irreflexivity and transitivity from the proper order $>$ on A. Next we show that $>_\mathcal{A}$ is closed under substitutions. Suppose $s >_\mathcal{A} t$ and let σ be an arbitrary substitution. We have $[\alpha]_\mathcal{A}(s\sigma) = [[\alpha]_\mathcal{A} \circ \sigma]_\mathcal{A}(s) > [[\alpha]_\mathcal{A} \circ \sigma]_\mathcal{A}(t) = [\alpha]_\mathcal{A}(t\sigma)$ for all assignments α. Here the equalities follow from Lemma 2.2.22. Hence $s\sigma >_\mathcal{A} t\sigma$. Closure under contexts follows from the strict monotonicity of the operations of $\mathcal{A}$. Hence $>_\mathcal{A}$ is a rewrite order. The second statement of the lemma is obvious since $>_\mathcal{A}$ inherits well-foundedness from $>$. □

Definition 4.1.9. We say that a TRS $\mathcal{R}$ and a monotone algebra $(\mathcal{A}, >)$ are *compatible* if $\mathcal{R}$ and $>_\mathcal{A}$ are compatible.

The main result of this section states that well-founded monotone algebras are powerful enough to capture all possible termination arguments.

Theorem 4.1.10. *A* TRS *is terminating if and only if it is compatible with a well-founded monotone algebra.*

Proof

$\Longrightarrow$ We already observed that $> = \to_\mathcal{R}^+$ is a reduction order for any terminating TRS $\mathcal{R}$. Hence $(\bar{\mathcal{T}}, >)$ is a monotone algebra. (Strict monotonicity of the algebra operations follows from closure of $>$ under contexts.) We claim that $\mathcal{R}$ and $(\bar{\mathcal{T}}, >)$ are compatible. Let $\ell \to r \in \mathcal{R}$. We have to show $\ell >_{\bar{\mathcal{T}}} r$, i.e., $[\alpha]_{\bar{\mathcal{T}}}(\ell) > [\alpha]_{\bar{\mathcal{T}}}(r)$ for every assignment α. Let σ be the restriction of α to $\mathcal{V}\mathsf{ar}(\ell)$, so σ is a substitution such that $[\alpha]_{\bar{\mathcal{T}}}(\ell) = \ell\sigma$ and $[\alpha]_{\bar{\mathcal{T}}}(r) = r\sigma$. We clearly have $\ell\sigma > r\sigma$.

$\Longleftarrow$ Immediate consequence of Lemma 4.1.8 and Theorem 4.1.4. □

Theorem 4.1.10 does not make the task of showing termination of a given TRS easier. It does however provide a useful basis for comparing termination arguments, as we will see in a later chapter. Employing the technique of monotone algebras for obtaining the termination of a given TRS $\mathcal{R}$ over a signature $\mathcal{F}$ amounts to the following:

1. choose a non-empty well-founded order $(A, >)$,
2. turn $(A, >)$ into a well-founded monotone $\mathcal{F}$-algebra $(\mathcal{A}, >)$ by defining for each n-ary function symbol $f \in \mathcal{F}$ an operation $f_\mathcal{A}$ from A^n to A that is strictly monotone in all its n coordinates,
3. show $[\alpha]_\mathcal{A}(\ell) > [\alpha]_\mathcal{A}(r)$ for all rewrite rules $\ell \to r \in \mathcal{R}$ and assignments α.

In the remainder of this section we illustrate the above technique by means of a few examples.

Example 4.1.11. First of all consider the TRS $\mathcal{R}_1$ consisting of the single rewrite rule

$$\mathsf{f}(\mathsf{f}(x, y), z) \to \mathsf{f}(x, \mathsf{f}(y, z))$$

Let $(A, >) = (\mathbb{N}, >_\mathbb{N})$, the set of natural numbers equipped with the usual order, and define $\mathsf{f}_\mathcal{A}(x, y) = 2x + y + 1$ for all $x, y \in \mathbb{N}$. The operation $\mathsf{f}_\mathcal{A}$ is strictly monotone in both coordinates: if $x >_\mathbb{N} x'$ and $y >_\mathbb{N} y'$ then $2x+y+1 >_\mathbb{N} 2x'+y+1$ and $2x+y+1 >_\mathbb{N} 2x+y'+1$.

We have

$$\mathsf{f}_{\mathcal{A}}(\mathsf{f}_{\mathcal{A}}(x,y),z) \;=\; 4x+2y+z+3 \;>_{\mathbb{N}}\; 2x+2y+z+2 \;=\; \mathsf{f}_{\mathcal{A}}(x,\mathsf{f}_{\mathcal{A}}(y,z))$$

for all $x,y,z \in \mathbb{N}$. Hence $[\alpha]_{\mathcal{A}}(\mathsf{f}(\mathsf{f}(x,y),z)) >_{\mathbb{N}} [\alpha]_{\mathcal{A}}(\mathsf{f}(x,\mathsf{f}(y,z)))$ for every assignment α, yielding the termination of $\mathcal{R}_1$.

Example 4.1.12. Next we consider the TRS $\mathcal{R}_2$ of Table 3.1. We take $(A,>) = (\mathbb{N}_+,>_{\mathbb{N}})$, the set of positive integers equipped with the usual order, and we interpret the function symbols $\mathsf{0}$, s, $+$, and $\times$ as follows: $\mathsf{0}_{\mathcal{A}} = 2$, $\mathsf{s}_{\mathcal{A}}(x) = x+3$, $+_{\mathcal{A}}(x,y) = x+2y$, and $\times_{\mathcal{A}}(x,y) = xy$ for all $x,y \geqslant 1$. All algebra operations are strictly monotone in all their coordinates. We have

$$\begin{aligned}
x +_{\mathcal{A}} \mathsf{0}_{\mathcal{A}} &= x+4 &&>_{\mathbb{N}} x \\
x +_{\mathcal{A}} \mathsf{s}_{\mathcal{A}}(y) &= x+2y+6 &&>_{\mathbb{N}} x+2y+2 &&= \mathsf{s}_{\mathcal{A}}(x +_{\mathcal{A}} y) \\
x \times_{\mathcal{A}} \mathsf{0}_{\mathcal{A}} &= 2x &&>_{\mathbb{N}} 2 &&= \mathsf{0}_{\mathcal{A}} \\
x \times_{\mathcal{A}} \mathsf{s}_{\mathcal{A}}(y) &= xy+3x &&>_{\mathbb{N}} xy+2x &&= (x \times_{\mathcal{A}} y) +_{\mathcal{A}} x
\end{aligned}$$

for all $x,y \geqslant 1$, proving the termination of $\mathcal{R}_2$.

Example 4.1.13. The TRS $\mathcal{R}_3$ consisting of the rewrite rules

$$\begin{aligned}
\partial(\alpha) &\to \mathsf{1} & \partial(x-y) &\to \partial(x) - \partial(y) \\
\partial(\beta) &\to \mathsf{0} & \partial(x \times y) &\to (\partial(x) \times y) + (x \times \partial(y)) \\
\partial(x+y) &\to \partial(x) + \partial(y) & \partial(x \div y) &\to ((\partial(x) \times y) - (x \times \partial(y))) \div (y \times y)
\end{aligned}$$

can be viewed as a specification of symbolic differentiation with respect to α. Termination of $\mathcal{R}_3$ can be shown by taking $(A,>) = (\{n \in \mathbb{N} \mid n \geqslant 3\},>_{\mathbb{N}})$ with the following strictly monotone interpretations: $\mathsf{0}_{\mathcal{A}} = \mathsf{1}_{\mathcal{A}} = \alpha_{\mathcal{A}} = \beta_{\mathcal{A}} = 3$, $\partial_{\mathcal{A}}(x) = x^2$, and $+_{\mathcal{A}}(x,y) = -_{\mathcal{A}}(x,y) = \times_{\mathcal{A}}(x,y) = \div_{\mathcal{A}}(x,y) = x+y$ for all $x,y \geqslant 3$. Since

$$\begin{aligned}
\partial_{\mathcal{A}}(\alpha_{\mathcal{A}}) &= 9 &&>_{\mathbb{N}} 3 &&= \mathsf{1}_{\mathcal{A}} \\
\partial_{\mathcal{A}}(\beta_{\mathcal{A}}) &= 9 &&>_{\mathbb{N}} 3 &&= \mathsf{0}_{\mathcal{A}} \\
\partial_{\mathcal{A}}(x +_{\mathcal{A}} y) &= x^2+2xy+y^2 &&>_{\mathbb{N}} x^2+y^2 &&= \partial_{\mathcal{A}}(x) +_{\mathcal{A}} \partial_{\mathcal{A}}(y) \\
\partial_{\mathcal{A}}(x -_{\mathcal{A}} y) &= x^2+2xy+y^2 &&>_{\mathbb{N}} x^2+y^2 &&= \partial_{\mathcal{A}}(x) -_{\mathcal{A}} \partial_{\mathcal{A}}(y) \\
\partial_{\mathcal{A}}(x \times_{\mathcal{A}} y) &= x^2+2xy+y^2 &&>_{\mathbb{N}} x^2+x+y^2+y &&= (\partial_{\mathcal{A}}(x) \times_{\mathcal{A}} y) +_{\mathcal{A}} (x \times_{\mathcal{A}} \partial_{\mathcal{A}}(y)) \\
\partial_{\mathcal{A}}(x \div_{\mathcal{A}} y) &= x^2+2xy+y^2 &&>_{\mathbb{N}} x^2+x+y^2+3y &&= \\
& & & & & ((\partial_{\mathcal{A}}(x) \times_{\mathcal{A}} y) -_{\mathcal{A}} (x \times_{\mathcal{A}} \partial_{\mathcal{A}}(y))) \div_{\mathcal{A}} (y \times_{\mathcal{A}} y)
\end{aligned}$$

for all $x,y \geqslant 3$, the TRS $\mathcal{R}_3$ is terminating.

These three examples share the following characteristics: the carrier of the algebra is $\{n \in \mathbb{N} \mid n \geqslant N\}$ for some $N \in \mathbb{N}$, the well-founded order is the restriction of the standard order on $\mathbb{N}$ to $\{n \in \mathbb{N} \mid n \geqslant N\}$, and every function symbol is interpreted as a polynomial.

Definition 4.1.14. A TRS is called *polynomially terminating over* $\mathbb{N}$ if it is compatible with a monotone algebra $(\mathcal{A}, >)$ having the following three properties:

1. the carrier of $\mathcal{A}$ is $\mathbb{N}$,
2. the proper order $>$ is the standard order $>_{\mathbb{N}}$ on $\mathbb{N}$,
3. the operation $f_{\mathcal{A}}$ is a polynomial for every function symbol f.

In this chapter we drop the qualification "over $\mathbb{N}$." In Chapter 9 we consider polynomial termination over other domains. Taking $\{n \in \mathbb{N} \mid n \geqslant N\}$ for some $N > 0$ instead of $\mathbb{N}$ as carrier in the preceding definition does not change the class of polynomially terminating TRSs (Exercise 4.4).

Example 4.1.15. The TRS $\mathcal{R}_2$ can be interpreted in $(\mathbb{N}, >_{\mathbb{N}})$ by defining $\mathsf{0}_{\mathcal{A}} = 0$, $\mathsf{s}_{\mathcal{A}}(x) = x+3$, $+_{\mathcal{A}}(x,y) = x+2y+2$, and $\times_{\mathcal{A}}(x,y) = xy+x+y$ for all $x, y \in \mathbb{N}$. Likewise, we could have interpreted $\mathcal{R}_3$ in $(\mathbb{N}, >_{\mathbb{N}})$ by letting $\mathsf{0}_{\mathcal{A}} = \mathsf{1}_{\mathcal{A}} = \alpha_{\mathcal{A}} = \beta_{\mathcal{A}} = 0$, $\partial_{\mathcal{A}}(x) = x^2 + 6x + 6$, and $+_{\mathcal{A}}(x,y) = -_{\mathcal{A}}(x,y) = \times_{\mathcal{A}}(x,y) = \div_{\mathcal{A}}(x,y) = x + y + 3$ for all $x, y \in \mathbb{N}$.

Even if a TRS is polynomially terminating, finding the right polynomials can be very difficult. Worse, it is undecidable whether a polynomial over the natural numbers is strictly monotone in all its coordinates. Moreover, the induced reduction order on terms (cf. Definition 4.1.7) is not computable. As a matter of fact, it is undecidable whether a given finite TRS is polynomially terminating, even we restrict ourselves to linear interpretations. These results will be proved in Chapter 9.

Many terminating TRSs are not polynomially terminating. In the next two examples we present two such TRSs.

Example 4.1.16. Consider the SRS $\mathcal{R}_4$ consisting of the rule

$$\mathsf{aa} \to \mathsf{aba}$$

We show that $\mathcal{R}_4$ is not polynomially terminating. Actually we show the stronger statement that $\mathcal{R}_4$ is not compatible with a well-founded monotone algebra $(\mathcal{A}, >_{\mathbb{N}})$ whose carrier A is $\mathbb{N}$. Suppose to the contrary that there exist strictly monotone functions $\mathsf{a}_{\mathcal{A}}, \mathsf{b}_{\mathcal{A}} \colon \mathbb{N} \to \mathbb{N}$ such that $\mathsf{a}_{\mathcal{A}}(\mathsf{a}_{\mathcal{A}}(x)) >_{\mathbb{N}} \mathsf{a}_{\mathcal{A}}(\mathsf{b}_{\mathcal{A}}(\mathsf{a}_{\mathcal{A}}(x)))$ for all $x \in \mathbb{N}$. By induction on x we easily obtain $\mathsf{b}_{\mathcal{A}}(x) \geqslant_{\mathbb{N}} x$ for all $x \in \mathbb{N}$. Fix $x \in \mathbb{N}$. We have $\mathsf{b}_{\mathcal{A}}(\mathsf{a}_{\mathcal{A}}(x)) \geqslant_{\mathbb{N}} \mathsf{a}_{\mathcal{A}}(x)$. Strict monotonicity of $\mathsf{a}_{\mathcal{A}}$ yields $\mathsf{a}_{\mathcal{A}}(\mathsf{b}_{\mathcal{A}}(\mathsf{a}_{\mathcal{A}}(x))) \geqslant_{\mathbb{N}} \mathsf{a}_{\mathcal{A}}(\mathsf{a}_{\mathcal{A}}(x))$, contradicting $\mathsf{a}_{\mathcal{A}}(\mathsf{a}_{\mathcal{A}}(x)) >_{\mathbb{N}} \mathsf{a}_{\mathcal{A}}(\mathsf{b}_{\mathcal{A}}(\mathsf{a}_{\mathcal{A}}(x)))$. So for a termination proof of $\mathcal{R}_4$ by monotone algebras we cannot use natural numbers. Define instead $A = \{0,1\} \times \mathbb{N}$ and $(a,x) > (b,y)$ if and only if $a = b$ and $x >_{\mathbb{N}} y$. It is easy to show that $>$ is a well-founded order on A. Interpret the function symbols a and b as follows: $\mathsf{a}_{\mathcal{A}}((0,x)) = (0, x+1)$, $\mathsf{a}_{\mathcal{A}}((1,x)) = (0,x)$, and $\mathsf{b}_{\mathcal{A}}((0,x)) = \mathsf{b}_{\mathcal{A}}((1,x)) = (1,x)$ for all $x \in \mathbb{N}$. Both operations are strictly monotone, while

$$\begin{aligned} \mathsf{a}_{\mathcal{A}}(\mathsf{a}_{\mathcal{A}}((0,x))) &= (0,x+2) > (0,x+1) = \mathsf{a}_{\mathcal{A}}(\mathsf{b}_{\mathcal{A}}(\mathsf{a}_{\mathcal{A}}((0,x)))) \\ \mathsf{a}_{\mathcal{A}}(\mathsf{a}_{\mathcal{A}}((1,x))) &= (0,x+1) > (0,x) \quad\ = \mathsf{a}_{\mathcal{A}}(\mathsf{b}_{\mathcal{A}}(\mathsf{a}_{\mathcal{A}}((1,x)))) \end{aligned}$$

for all $x \in \mathbb{N}$, proving the termination of $\mathcal{R}_4$.

Example 4.1.17. As a final example we consider the TRS $\mathcal{R}_5$ consisting of the single rewrite rule

$$\mathsf{f}(\mathsf{a},\mathsf{b},x) \to \mathsf{f}(x,x,x)$$

It is not difficult to show that $\mathcal{R}_5$ is not polynomially terminating, cf. Exercise 4.8. Termination can be shown by taking the same well-founded order $(A,>)$ as in the previous example. Define operations $\mathsf{a}_\mathcal{A} = (0,1)$, $\mathsf{b}_\mathcal{A} = (1,1)$, and

$$\mathsf{f}_\mathcal{A}((a,x),(b,y),(c,z)) = \begin{cases} (0,x+y+z) & \text{if } a = b \\ (0,x+y+3z) & \text{if } a \neq b \end{cases}$$

It is easy to show that $\mathsf{f}_\mathcal{A}$ is strictly monotone in all three coordinates. We have

$$\mathsf{f}_\mathcal{A}(\mathsf{a}_\mathcal{A},\mathsf{b}_\mathcal{A},(a,x)) = (0,3x+2) > (0,3x) = \mathsf{f}_\mathcal{A}((a,x),(a,x),(a,x))$$

for all $(a,x) \in A$. Hence also $\mathcal{R}_5$ is terminating.

Exercises

4.1 Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{array}{lll} 0+y \to y & 0 \mathbin{\dot-} y \to 0 & \mathsf{min}(x,y) \to x \mathbin{\dot-} (x \mathbin{\dot-} y) \\ \mathsf{s}(x)+y \to \mathsf{s}(x+y) & \mathsf{s}(x) \mathbin{\dot-} 0 \to \mathsf{s}(x) & \mathsf{max}(x,y) \to (x+y) \mathbin{\dot-} \mathsf{min}(x,y) \\ & \mathsf{s}(x) \mathbin{\dot-} \mathsf{s}(y) \to x \mathbin{\dot-} y & \end{array}$$

a Show that the following interpretation over $\mathbb{N}$ is compatible with $\mathcal{R}$:

$$\begin{array}{lll} 0_\mathbb{N} = 0 & +_\mathbb{N}(x,y) = 2x+y+1 & \mathsf{min}_\mathbb{N}(x,y) = 2x+y+3 \\ \mathsf{s}_\mathbb{N}(x) = x+1 & \mathbin{\dot-}_\mathbb{N}(x,y) = x+y+1 & \mathsf{max}_\mathbb{N}(x,y) = 4x+2y+6 \end{array}$$

b Find natural numbers a, b, c, d, e, and f such that the following interpretation over $\mathbb{N}$ is compatible with $\mathcal{R}$:

$$\begin{array}{lll} 0_\mathbb{N} = 0 & +_\mathbb{N}(x,y) = 2x+y+1 & \mathsf{min}_\mathbb{N}(x,y) = 3x+by+c \\ \mathsf{s}_\mathbb{N}(x) = x+1 & \mathbin{\dot-}_\mathbb{N}(x,y) = x+2y+a & \mathsf{max}_\mathbb{N}(x,y) = dx+ey+f \end{array}$$

c Consider the following partial interpretation over $\mathbb{N}$:

$$\begin{array}{ll} 0_\mathbb{N} = 1 & +_\mathbb{N}(x,y) = 3x+y \\ \mathbin{\dot-}_\mathbb{N}(x,y) = 2x+y & \mathsf{max}_\mathbb{N}(x,y) = 5x^2+5x+5y+5 \end{array}$$

Which of the following interpretations give rise to a correct termination proof of $\mathcal{R}$?

1. $\mathsf{s}_\mathbb{N}(x) = x+1$ and $\mathsf{min}_\mathbb{N}(x,y) = 5x+y$
2. $\mathsf{s}_\mathbb{N}(x) = x+1$ and $\mathsf{min}_\mathbb{N}(x,y) = 4x^2+y+4$
3. $\mathsf{s}_\mathbb{N}(x) = 2x+1$ and $\mathsf{min}_\mathbb{N}(x,y) = 4x+y+1$

4.2 Is the SRS consisting of the single rewrite rule $\mathsf{ab} \to \mathsf{bbaa}$ terminating?

4.3 Prove that the TRS consisting of the three rewrite rules

$$\mathsf{f}(x) \otimes \mathsf{f}(y) \to \mathsf{f}(x \otimes y) \qquad \mathsf{f}(x) \otimes (\mathsf{f}(y) \otimes z) \to \mathsf{f}(x \otimes y) \otimes z \qquad (x \otimes y) \otimes z \to x \otimes (y \otimes z)$$

is polynomially terminating.

4.4 Show that a TRS is polynomially terminating if and only if it is compatible with a monotone algebra $(\mathcal{A}, >)$ such that the carrier of $\mathcal{A}$ is the set $\mathbb{N}_{\geqslant N} = \{n \in \mathbb{N} \mid n \geqslant N\}$ for some $N \in \mathbb{N}$, the proper order $>$ is the restriction of $>_{\mathbb{N}}$ to $\mathbb{N}_{\geqslant N}$, and the operation $f_{\mathcal{A}}$ is a polynomial for every function symbol f.

4.5 Is the TRS consisting of the three rewrite rules

$$\mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x) \qquad \mathsf{g}(x, y) \to x \qquad \mathsf{g}(x, y) \to y$$

terminating?

4.6 Show that Theorem 4.1.4 remains true if we do not require reduction orders to be transitive.

4.7 Is the TRS consisting of the three rewrite rules

$$\begin{aligned} \mathsf{f}(x, \mathsf{g}(y, z)) &\to \mathsf{g}(\mathsf{f}(x, y), \mathsf{f}(x, z)) \\ \mathsf{f}(\mathsf{g}(x, y), z) &\to \mathsf{g}(\mathsf{f}(x, z), \mathsf{f}(y, z)) \\ \mathsf{g}(\mathsf{g}(x, y), z) &\to \mathsf{g}(x, \mathsf{g}(y, z)) \end{aligned}$$

polynomially terminating?

4.8 Show that the TRS consisting of the single rewrite rule $\mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x)$ is not compatible with a well-founded monotone algebra whose underlying well-founded order is $(\mathbb{N}, >_{\mathbb{N}})$.

4.9 Find a well-founded monotone algebra that is compatible with the TRS consisting of the following rewrite rules:

$$\mathsf{f}(\mathsf{g}(x), y) \to \mathsf{g}(\mathsf{f}(x, \mathsf{f}(x, y))) \qquad \mathsf{f}(x, x) \to \mathsf{g}(\mathsf{g}(x))$$

4.10 Find a well-founded monotone algebra that is compatible with the TRS of Exercise 3.5.

4.11 Consider the TRS $\mathcal{R}_1$ of Table 3.1.

a Prove that $\mathcal{R}_1$ is not polynomially terminating.

b Find a compatible well-founded monotone algebra whose carrier is $\mathbb{N} \setminus \{0\}$ equipped with $>_{\mathbb{N}}$.

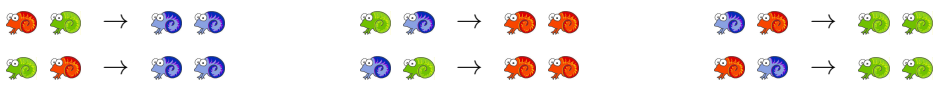
c Discuss the usefulness of the polynomial interpretation

$$\begin{aligned} \mathsf{0}_{\mathbb{N}} &= 2 & +_{\mathbb{N}}(x, y) &= x + y \\ \mathsf{s}_{\mathbb{N}}(x) &= x + 2 & \times_{\mathbb{N}}(x, y) &= xy \end{aligned}$$

over $\mathbb{N} \setminus \{0, 1\}$ for proving the termination of $\mathcal{R}_1$.

4.12 Show that the TRS consisting of the single rewrite rule $\mathsf{f}(\mathsf{a}) \to \mathsf{f}(\mathsf{g}(\mathsf{a}))$ is terminating by constructing a compatible well-founded monotone algebra.

4.13 Is the SRS consisting of the rewrite rules



terminating?

4.14 A TRS $\mathcal{R}$ is called *looping* if it admits a rewrite sequence $s \to_{\mathcal{R}}^{+} t$ such that t encompasses s.

a Show that looping TRSs are not terminating.

b Construct a finite TRS that is neither terminating nor looping.

4.15 Is the TRS consisting of the six rewrite rules

$$\begin{aligned} \mathsf{f}(0) &\to 0 & \mathsf{s}(\mathsf{s}(0)) &\to \mathsf{f}(0) \\ \mathsf{f}(\mathsf{s}(0)) &\to \mathsf{s}(0) & \mathsf{s}(\mathsf{s}(\mathsf{s}(0))) &\to \mathsf{f}(\mathsf{s}(0)) \\ \mathsf{f}(\mathsf{s}(\mathsf{s}(0))) &\to \mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0))))) & \mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0))))))))) &\to \mathsf{f}(\mathsf{s}(\mathsf{s}(0))) \end{aligned}$$

polynomially terminating?

4.2 Simple Termination

In the previous section we have seen that proving termination of a TRS is equivalent to the construction of a compatible reduction order. Later we will define several powerful reduction orders by induction on the structure of terms. Closure under contexts and substitutions of these orders is easy to prove. Proving irreflexivity and transitivity often turns out to be feasible, using some induction and case analysis. The bottleneck is proving well-foundedness. In this section we present a simple syntactic condition which ensures well-foundedness (for terms over a finite signature).

Definition 4.2.1. A binary relation R on terms has the *subterm property* if $C[t]\ R\ t$ for all non-empty contexts C and terms t.

The subterm property of a relation R can be expressed more concisely by the inclusion $\rhd \subseteq R$. The task of showing that a given *transitive* relation R has the subterm property amounts to verifying $f(t_1,\ldots,t_n)\ R\ t_i$ for all function symbols f of arity $n \geqslant 1$, terms $t_1,\ldots,t_n$, and $i \in \{1,\ldots,n\}$; cf. Exercise 4.17. This observation will be used freely in the sequel.

The relation $\rhd$ is the smallest proper order with the subterm property, but it is not a rewrite order as it lacks closure under contexts. For instance a is a proper subterm of $\mathsf{f}(\mathsf{a})$, but $\mathsf{g}(\mathsf{a})$ is not a (proper) subterm of $\mathsf{g}(\mathsf{f}(\mathsf{a}))$. We will show that there is a smallest rewrite order with the subterm property.

Definition 4.2.2. Let $\mathcal{F}$ be a signature. The TRS $\mathcal{E}\mathsf{mb}(\mathcal{F})$ consists of all rewrite rules

$$f(x_1,\ldots,x_n) \to x_i$$

with $f \in \mathcal{F}$ a function symbol of arity $n \geqslant 1$ and $i \in \{1,\ldots,n\}$. Here $x_1,\ldots,x_n$ are pairwise different variables. We abbreviate $\to^*_{\mathcal{E}\mathsf{mb}(\mathcal{F})}$ ($\to^+_{\mathcal{E}\mathsf{mb}(\mathcal{F})}$) to $\trianglerighteq_{\mathsf{emb}}$ ($\rhd_{\mathsf{emb}}$). The converse $\trianglelefteq_{\mathsf{emb}}$ ($\lhd_{\mathsf{emb}}$) of $\trianglerighteq_{\mathsf{emb}}$ ($\rhd_{\mathsf{emb}}$) is called (*proper*) *embedding*. We abbreviate $\mathcal{E}\mathsf{mb}(\mathcal{F})$ to $\mathcal{E}\mathsf{mb}$ when the signature $\mathcal{F}$ can be inferred from the context.

Example 4.2.3. The term $\mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{b})$ is (properly) embedded in $\mathsf{f}(\mathsf{h}(\mathsf{g}(\mathsf{a}),\mathsf{a}),\mathsf{g}(\mathsf{b}))$ since there exists a (non-empty) rewrite sequence in the TRS

$$\mathcal{E}\mathsf{mb}(\{\mathsf{a},\mathsf{b},\mathsf{f},\mathsf{g},\mathsf{h}\}) = \left\{ \begin{array}{lll} \mathsf{f}(x,y) \to x & \mathsf{g}(x) \to x & \mathsf{h}(x,y) \to x \\ \mathsf{f}(x,y) \to y & & \mathsf{h}(x,y) \to y \end{array} \right\}$$

from $\mathsf{f}(\mathsf{h}(\mathsf{g}(\mathsf{a}),\mathsf{a}),\mathsf{g}(\mathsf{b}))$ to $\mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{b})$:

$$\mathsf{f}(\mathsf{h}(\mathsf{g}(\mathsf{a}),\mathsf{a}),\mathsf{g}(\mathsf{b})) \to \mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{g}(\mathsf{b})) \to \mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{b})$$

Observe that the term $\mathsf{f}(\mathsf{a},\mathsf{b})$ is not embedded in the term $\mathsf{f}(\mathsf{h}(\mathsf{a},\mathsf{b}),\mathsf{g}(\mathsf{a}))$.

Lemma 4.2.4. *A rewrite order $>$ has the subterm property if and only if $\rhd_{\mathsf{emb}} \subseteq >$.*

Proof

$\Longrightarrow$ Since x_i is a proper subterm of $f(x_1,\ldots,x_n)$ the TRS $\mathcal{E}\mathsf{mb}$ is compatible with $>$. Lemma 4.1.5 yields the desired $\rhd_{\mathsf{emb}} = \to^+_{\mathcal{E}\mathsf{mb}} \subseteq >$.

$\Longleftarrow$ Because $\rhd_{\mathsf{emb}}$ has the subterm property, so does $>$. □

Lemma 4.2.5. *The relation $\rhd_{\mathsf{emb}}$ is the smallest rewrite order with the subterm property.*

Proof It is easy to see that $\rhd_{\mathsf{emb}}$ is a rewrite order with the subterm property. Lemma 4.2.4 shows that it is the smallest. □

Is it possible to replace the well-foundedness requirement in the definition of reduction order by the subterm property and still be able to conclude the termination of compatible TRSs? The answer is no.

Example 4.2.6. Consider for instance the TRS $\mathcal{R}$ consisting of infinitely many constants a_i and rewrite rules $\mathsf{a}_i \to \mathsf{a}_{i+1}$ for all $i \geqslant 0$. The rewrite order $\to^+_{\mathcal{R}}$ vacuously satisfies the subterm property, but $\mathcal{R}$ is not terminating: $\mathsf{a}_0 \to \mathsf{a}_1 \to \mathsf{a}_2 \to \cdots$.

In the above example the signature is infinite. In this section we show that this is essential. In other words, we will prove that for TRSs over finite signatures compatibility with a rewrite order that has the subterm property suffices for termination. In the remainder of this section we only consider *finite signatures*.

Definition 4.2.7. A *simplification order* is a rewrite order with the subterm property. A TRS is called *simply terminating* if it is compatible with a simplification order.

According to Lemma 4.2.4 a rewrite order is a simplification order if and only if it contains $\rhd_{\mathsf{emb}}$. The main result of this section (Corollary 4.2.9) states that every simply terminating TRS is terminating, justifying the terminology simple *termination*. The proof is based on the beautiful Tree Theorem of Kruskal.

Kruskal's Tree Theorem. *For every infinite sequence t_0, t_1, t_2, ... of ground terms there exist $1 \leqslant i < j$ such that $t_i \trianglelefteq_{\mathsf{emb}} t_j$.*

A proof of Kruskal's Tree Theorem can be found in Appendix B.2.

Lemma 4.2.8. *Every simplification order is well-founded.*

Proof Let $>$ be a simplification order on $\mathcal{T}(\mathcal{F},\mathcal{V})$. First we show $\mathcal{V}\mathsf{ar}(t) \subseteq \mathcal{V}\mathsf{ar}(s)$ whenever $s > t$. Suppose to the contrary that there exists a variable $x \in \mathcal{V}\mathsf{ar}(t) \setminus \mathcal{V}\mathsf{ar}(s)$. Define $\sigma = \{x \mapsto s\}$. Closure under substitutions yields $s = s\sigma > t\sigma$. Since s is a subterm of $t\sigma$, the subterm property yields $t\sigma \geqslant s$, contradicting the fact that $>$ is a proper order. Now consider an infinite sequence $t_0 > t_1 > t_2 > \cdots$ and let $\mathcal{V}\mathsf{ar}(t_0) = \{x_1,\ldots,x_n\}$. According to the above observation we have $\mathcal{V}\mathsf{ar}(t_i) \subseteq \{x_1,\ldots,x_n\}$ for all $i \geqslant 0$. Choose a fresh constant c and define $\mathcal{F}' = \mathcal{F} \cup \{c\}$ and $\tau = \{x_i \mapsto c \mid 1 \leqslant i \leqslant n\}$. We obtain $t_0\tau > t_1\tau > t_2\tau > \cdots$ because $>$ is closed under substitutions. The infinite sequence $t_0\tau$, $t_1\tau$, $t_2\tau$, $\ldots$ contains only terms in $\mathcal{T}(\mathcal{F}')$. From Kruskal's Tree Theorem we learn the existence of indices i, j with $1 \leqslant i < j$ such that $t_i\tau \trianglelefteq_{\mathsf{emb}} t_j\tau$. From Lemma 4.2.4 we obtain $t_i\tau \leqslant t_j\tau$. Since $i < j$ we also have $t_i\tau > t_j\tau$. This is impossible. We conclude that $>$ is well-founded. □

So simplification orders are reduction orders. Hence we obtain the following result.

Corollary 4.2.9. *Every simply terminating* TRS *is terminating.*

The next result is especially useful for showing that a given TRS is *not* simply terminating.

Lemma 4.2.10. *The following statements are equivalent.*

1. *The* TRS $\mathcal{R}$ *is simply terminating.*
2. *The* TRS $\mathcal{R} \cup \mathcal{E}\mathsf{mb}$ *is terminating.*
3. *The* TRS $\mathcal{R} \cup \mathcal{E}\mathsf{mb}$ *is acyclic.*

Proof

1 $\Longrightarrow$ 2 Let $\mathcal{R}$ be compatible with the simplification order $>$. Lemma 4.2.4 shows that $\rhd_{\mathsf{emb}} \subseteq >$ and hence $>$ is compatible with the TRS $\mathcal{E}\mathsf{mb}$. Therefore $\mathcal{R} \cup \mathcal{E}\mathsf{mb}$ is a simply terminating TRS. Corollary 4.2.9 yields its termination.

2 $\Longrightarrow$ 3 Obvious.

3 $\Longrightarrow$ 1 Let $>$ be the transitive closure of the rewrite relation of the TRS $\mathcal{R} \cup \mathcal{E}\mathsf{mb}$. Because $\mathcal{R} \cup \mathcal{E}\mathsf{mb}$ is acyclic, $>$ is irreflexive and hence a rewrite order. Since $\rhd_{\mathsf{emb}} \subseteq >$, $>$ is a simplification order. Since the TRS $\mathcal{R}$ is compatible with $>$, it is simply terminating. □

Example 4.2.11. The SRS $\mathcal{R}_4$ of Example 4.1.16 is not simply terminating because $\mathcal{R}_4 \cup \mathcal{E}\mathsf{mb}$ is cyclic: $\mathsf{aa} \to_{\mathcal{R}_4} \mathsf{aba} \to_{\mathcal{E}\mathsf{mb}} \mathsf{aa}$. Likewise, the TRS $\mathcal{R}_5$ of Example 4.1.17 is not simply terminating:

$$\begin{aligned}\mathsf{f}(\mathsf{a},\mathsf{b},\mathsf{f}(\mathsf{a},\mathsf{b},\mathsf{a})) &\to_{\mathcal{R}_5} \mathsf{f}(\mathsf{f}(\mathsf{a},\mathsf{b},\mathsf{a}),\mathsf{f}(\mathsf{a},\mathsf{b},\mathsf{a}),\mathsf{f}(\mathsf{a},\mathsf{b},\mathsf{a}))\\ &\to_{\mathcal{E}\mathsf{mb}} \mathsf{f}(\mathsf{a},\mathsf{f}(\mathsf{a},\mathsf{b},\mathsf{a}),\mathsf{f}(\mathsf{a},\mathsf{b},\mathsf{a}))\\ &\to_{\mathcal{E}\mathsf{mb}} \mathsf{f}(\mathsf{a},\mathsf{b},\mathsf{f}(\mathsf{a},\mathsf{b},\mathsf{a}))\end{aligned}$$

In the remainder of this section we give a semantic characterization of simple termination.

Definition 4.2.12. A *simple monotone* $\mathcal{F}$-algebra $(\mathcal{A}, >)$ consists of a non-empty $\mathcal{F}$-algebra $\mathcal{A}$ and a well-founded order $>$ on the carrier A of $\mathcal{A}$ such that every algebra operation is both simple and weakly monotone, i.e., if $f \in \mathcal{F}$ has arity $n \geqslant 1$ then

$$f_{\mathcal{A}}(a_1, \ldots, a_i, \ldots, a_n) \geqslant a_i$$

for all $a_1, \ldots, a_n \in A$, $i \in \{1, \ldots, n\}$ and

$$f_{\mathcal{A}}(a_1, \ldots, a_i, \ldots, a_n) \geqslant f_{\mathcal{A}}(a_1, \ldots, b, \ldots, a_n)$$

for all b with $a_i > b$. The induced relation $>_{\mathcal{A}}$ on terms is defined as for monotone algebras and the same holds for the notion of compatibility (Definitions 4.1.7 and 4.1.9).

The final result of this section states that simple monotone algebras characterize simple termination. In particular, unary function symbols may be interpreted as the identity operation on the carrier of the algebra.

Theorem 4.2.13. *A* TRS $\mathcal{R}$ *is simply terminating if and only if it is compatible with a simple monotone algebra.*

Proof

$\Longrightarrow$ Let $\mathcal{R}$ be a simply terminating TRS. According to Lemma 4.2.10 its extension $\mathcal{R} \cup \mathcal{E}\mathsf{mb}$ is terminating. Hence $> \; = \; \to^+_{\mathcal{R} \cup \mathcal{E}\mathsf{mb}}$ is a reduction order and (cf. the proof of Theorem 4.1.10) $\mathcal{R} \cup \mathcal{E}\mathsf{mb}$ is compatible with the well-founded monotone algebra $(\overline{\mathcal{T}}, >)$. Due to the definition of $>$, $(\overline{\mathcal{T}}, >)$ satisfies (the strict versions of) the conditions in Definition 4.2.12.

$\Longleftarrow$ Let $(\mathcal{A}, >)$ be a simple monotone algebra that is compatible with $\mathcal{R}$. We construct an algebra $(\mathcal{B}, \sqsupset)$ as follows. Its carrier B consists of all finite multisets $M \in \mathcal{M}(\mathcal{A})$ that have a maximum (with respect to $>$) element $\mathsf{max}(M)$, $\sqsupset$ is the restriction of $>_{\mathsf{mul}}$ to B, and

$$f_{\mathcal{B}}(M_1, \ldots, M_n) \; = \; \{ f_{\mathcal{A}}(\mathsf{max}(M_1), \ldots, \mathsf{max}(M_n)) \} \uplus M_1 \uplus \ldots \uplus M_n$$

for all function symbols f. Since

$$f_{\mathcal{A}}(\mathsf{max}(M_1), \ldots, \mathsf{max}(M_n)) \; \geqslant \; \mathsf{max}(M_i)$$

for all $1 \leqslant i \leqslant n$, $f_{\mathcal{A}}(\mathsf{max}(M_1), \ldots, \mathsf{max}(M_n))$ is a maximum element in $f_{\mathcal{B}}(M_1, \ldots, M_n)$ and hence $f_{\mathcal{B}}$ is well-defined. In Exercise 4.25 the reader is asked to show that $(\mathcal{B}, \sqsupset)$ is a well-founded monotone algebra compatible with $\mathcal{R} \cup \mathcal{E}\mathsf{mb}$. Hence simple termination follows from Lemma 4.2.10. □

Corollary 4.2.14. *Polynomially terminating* TRS*s are simply terminating.*

Proof Suppose $\mathcal{R}$ is a polynomially terminating TRS. So it is compatible with a well-founded monotone algebra $(\mathcal{A}, >_{\mathbb{N}})$ with carrier $\mathbb{N}$ and polynomials $f_{\mathcal{A}}$ for every function symbol f. An easy induction on the value of a_i shows that $f_{\mathcal{A}}(a_1, \ldots, a_n) \geqslant a_i$ for all n-ary function symbols f, natural numbers $a_1, \ldots, a_n$, and $i \in \{1, \ldots, n\}$. Hence $\mathcal{R}$ is simply terminating according to Theorem 4.2.13. □

Exercises

4.16 Which of the following terms are embedded in the term $\mathsf{f}(\mathsf{g}(\mathsf{f}(\mathsf{a},\mathsf{g}(\mathsf{b}))),\mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{b}))$?

- ▷ $\mathsf{f}(\mathsf{f}(\mathsf{a},\mathsf{a}),\mathsf{b})$
- ▷ $\mathsf{f}(\mathsf{g}(\mathsf{b}),\mathsf{g}(\mathsf{a}))$
- ▷ $\mathsf{f}(\mathsf{f}(\mathsf{a},\mathsf{b}),\mathsf{f}(\mathsf{a},\mathsf{b}))$

4.17 ***a*** Show that a transitive relation R on terms has the subterm property if and only if

$$f(t_1,\ldots,t_n)\ R\ t_i$$

for all function symbols f of arity $n \geqslant 1$, terms $t_1,\ldots,t_n$, and $i \in \{1,\ldots,n\}$.

b Why do we require R to be transitive in part (a)?

4.18 Compute all terms that are embedded in the term $\mathsf{f}(\mathsf{f}(\mathsf{a},\mathsf{f}(\mathsf{a},\mathsf{a})),\mathsf{f}(\mathsf{a},\mathsf{a}))$.

4.19 The equivalence in Lemma 4.2.4 is stated for rewrite orders. Is this essential?

4.20 ***a*** Show that embedding is an antisymmetric relation.

b What is the relationship between embedding and encompassment?

4.21 Let t be an arbitrary term. Show that $\{\,s \mid s \trianglelefteq_{\mathsf{emb}} t\,\}$ is a finite set.

4.22 Let $\mathcal{F}$ be an arbitrary signature.

a What are the normal forms of the TRS $\mathcal{E}\mathsf{mb}(\mathcal{F})$?

b Show that $\mathcal{E}\mathsf{mb}(\mathcal{F})$ is confluent if and only if $\mathcal{F}$ lacks function symbols of arity at least two.

4.23 Is the TRS consisting of the single rewrite rule $\mathsf{f}(x,x) \to \mathsf{f}(\mathsf{a},\mathsf{b})$ simply terminating?

4.24 ***a*** Prove that a TRS $\mathcal{R}$ is simply terminating if and only if the TRS $\mathcal{R} \setminus \{\ell \to r \mid \ell \rhd_{\mathsf{emb}} r\}$ is simply terminating.

b Can we replace simple termination by termination in part (a)?

4.25 Consider the proof of Theorem 4.2.13.

a Prove that $(\mathcal{B},\sqsupset)$ is a well-founded monotone algebra.

b Prove that $(\mathcal{B},\sqsupset)$ is compatible with $\mathcal{R}$.

c Prove that $(\mathcal{B},\sqsupset)$ is compatible with $\mathcal{E}\mathsf{mb}$.

4.26 ***a*** Prove that simple termination is an undecidable property of (finite) TRSs.

b Prove that simple termination is a decidable property of (finite) right-ground TRSs.

4.3 Lexicographic Path Order

In this section we introduce one of the most useful methods for humans to establish (simple) termination.

Definition 4.3.1. A *precedence* is a proper order on a signature.

Definition 4.3.2. Let $>$ be a precedence. We define the *lexicographic path order* (LPO for short) $>_{\mathsf{lpo}}$ on terms inductively as follows: $s = f(s_1,\ldots,s_n) >_{\mathsf{lpo}} t$ if one of the following alternatives holds:

1 $t = f(t_1,\ldots,t_n)$ and there exists an $i \in \{1,\ldots,n\}$ such that

(a) $s_j = t_j$ for all $1 \leqslant j < i$,

(b) $s_i >_{\mathsf{lpo}} t_i$, and

(c) $s >_{\mathsf{lpo}} t_j$ for all $i < j \leqslant n$,

2 $t = g(t_1, \ldots, t_m)$, $f > g$, and $s >_{\mathsf{lpo}} t_i$ for all $1 \leqslant i \leqslant m$, or

3 $s_i \geqslant^{=}_{\mathsf{lpo}} t$ for some $1 \leqslant i \leqslant n$.

Here $\geqslant^{=}_{\mathsf{lpo}}$ denotes the reflexive closure of $>_{\mathsf{lpo}}$.

Observe that $\|u\| + \|v\| < \|s\| + \|t\|$ for all statements of the form $u >_{\mathsf{lpo}} v$ in the three clauses in Definition 4.3.2. Hence $>_{\mathsf{lpo}}$ is well-defined (by induction on $\|s\| + \|t\|$).

Let us write $s >_{\mathsf{lpo}} t \ \langle i \rangle$ if $s >_{\mathsf{lpo}} t$ by clause **i** in Definition 4.3.2. Moreover, in case of $>_{\mathsf{lpo}} \ \langle 1 \rangle$ and $>_{\mathsf{lpo}} \ \langle 3 \rangle$ we often find it convenient to add the index $i \in \{1, \ldots, n\}$ whose existence is guaranteed in clauses **1** and **3**. Before showing that any TRS compatible with $>_{\mathsf{lpo}}$ for some well-founded precedence $>$ is (simply) terminating, we illustrate the strength of LPO on three examples.

Example 4.3.3. First of all consider the TRS $\mathcal{R}_1$ of Table 3.1. This TRS is not polynomially terminating (cf. Exercise 4.11), but its termination is easily shown by means of LPO. Take as precedence $\times > + > \mathsf{s}$. We clearly have $\mathsf{0} + x >_{\mathsf{lpo}} x \ \langle 3, 2 \rangle$ and $\mathsf{0} \times x >_{\mathsf{lpo}} \mathsf{0} \ \langle 3, 1 \rangle$. Since $\mathsf{s}(x) >_{\mathsf{lpo}} x \ \langle 3 \rangle$ and $\mathsf{s}(x) + y >_{\mathsf{lpo}} y \ \langle 3, 2 \rangle$ we have $\mathsf{s}(x) + y >_{\mathsf{lpo}} x + y \ \langle 1, 1 \rangle$. Together with $+ > \mathsf{s}$ this implies $\mathsf{s}(x) + y >_{\mathsf{lpo}} \mathsf{s}(x + y) \ \langle 2 \rangle$. We obtain $\mathsf{s}(x) \times y >_{\mathsf{lpo}} x \times y \ \langle 1, 1 \rangle$ just as $\mathsf{s}(x) + y >_{\mathsf{lpo}} x + y \ \langle 1, 1 \rangle$. Together with $\mathsf{s}(x) \times y >_{\mathsf{lpo}} y \ \langle 3, 2 \rangle$ and $\times > +$, this gives $\mathsf{s}(x) \times y >_{\mathsf{lpo}} (x \times y) + y \ \langle 2 \rangle$. So $>_{\mathsf{lpo}}$ orients all rewrite rules of $\mathcal{R}_1$ from left to right.

Example 4.3.4. Consider the TRS $\mathcal{R}$ of Exercise 3.5. With precedence $\neg > \wedge > \vee$ we easily obtain $\neg(\neg x) >_{\mathsf{lpo}} x \ \langle 3 \rangle$, $\neg(x \wedge y) >_{\mathsf{lpo}} (\neg x) \vee (\neg y) \ \langle 2 \rangle$, $\neg(x \vee y) >_{\mathsf{lpo}} (\neg x) \wedge (\neg y) \ \langle 2 \rangle$, $x \wedge (y \vee z) >_{\mathsf{lpo}} (x \wedge y) \vee (x \wedge z) \ \langle 2 \rangle$, and $(x \vee y) \wedge z >_{\mathsf{lpo}} (x \wedge z) \vee (y \wedge z) \ \langle 2 \rangle$. Hence $\mathcal{R}$ is compatible with $>_{\mathsf{lpo}}$.

Example 4.3.5. Consider the TRS $\mathcal{R}_3$ of Example 4.1.13. Take as precedence $\partial > \mathsf{0}$, $\partial > \mathsf{1}$, $\partial > +$, $\partial > -$, $\partial > \times$, and $\partial > \div$. One easily shows that $\ell >_{\mathsf{lpo}} r \ \langle 2 \rangle$ for every rewrite rule $\ell \to r \in \mathcal{R}_3$.

We show that $>_{\mathsf{lpo}}$ is a rewrite order which extends the relation $\rhd_{\mathsf{emb}}$. First we show that it is a rewrite relation.

Lemma 4.3.6. *Let $>$ be a precedence. The lexicographic path order $>_{\mathsf{lpo}}$ is a rewrite relation.*

Proof First we show that $>_{\mathsf{lpo}}$ is closed under contexts. Assume $s >_{\mathsf{lpo}} t$. It suffices to show $C[s] >_{\mathsf{lpo}} C[t]$ for every context C of the form $f(t_1, \ldots, \Box, \ldots, t_n)$ (cf. Exercise 2.8). Let i be the position of $\Box$ in C. We have $C[s] = f(t_1, \ldots, s, \ldots, t_n) >_{\mathsf{lpo}} t_j \ \langle 3, j \rangle$ for all $i < j \leqslant n$. Hence $C[s] >_{\mathsf{lpo}} f(t_1, \ldots, t, \ldots, t_n) = C[t] \ \langle 1 \rangle$. Next we show that $>_{\mathsf{lpo}}$ is closed under substitutions. So assume $s = f(s_1, \ldots, s_n) >_{\mathsf{lpo}} t$ and let σ be a substitution. By induction on $\|s\| + \|t\|$ we show $s\sigma >_{\mathsf{lpo}} t\sigma$. The base case is $\|s\| = 1$ and $\|t\| = 0$. So $s_1, \ldots, s_n, t \in \mathcal{V}$. Hence we must have $s >_{\mathsf{lpo}} t \ \langle 3, i \rangle$. This clearly implies $s\sigma >_{\mathsf{lpo}} t\sigma \ \langle 3, i \rangle$. For the induction step, suppose $s'\sigma >_{\mathsf{lpo}} t'\sigma$ for all terms s', t' with $s' >_{\mathsf{lpo}} t'$ and $\|s'\| + \|t'\| < k$ $(k > 1)$, and let $\|s\| + \|t\| = k$. We distinguish three cases.

1 If $s >_{\mathsf{lpo}} t\ \langle 1,i\rangle$ then $t = f(t_1,\ldots,t_n)$, $s_j = t_j$ for $1 \leqslant j < i$, $s_i >_{\mathsf{lpo}} t_i$, and $s >_{\mathsf{lpo}} t_j$ for $i < j \leqslant n$. Obviously $s_j\sigma = t_j\sigma$ for $1 \leqslant j < i$. The induction hypothesis yields $s_i\sigma >_{\mathsf{lpo}} t_i\sigma$ and $s\sigma >_{\mathsf{lpo}} t_j\sigma$ for $i < j \leqslant n$. Hence $s\sigma >_{\mathsf{lpo}} t\sigma\ \langle 1,i\rangle$.

2 If $s >_{\mathsf{lpo}} t\ \langle 2\rangle$ then $t = g(t_1,\ldots,t_m)$ with $f > g$ and $s >_{\mathsf{lpo}} t_i$ for all $1 \leqslant i \leqslant m$. From the induction hypothesis we learn that $s\sigma >_{\mathsf{lpo}} t_i\sigma$ for all $1 \leqslant i \leqslant m$ and therefore $s\sigma >_{\mathsf{lpo}} t\sigma\ \langle 2\rangle$.

3 If $s >_{\mathsf{lpo}} t\ \langle 3,i\rangle$ then $s_i >^{=}_{\mathsf{lpo}} t$. If $s_i = t$ then clearly $s_i\sigma = t\sigma$. If $s_i >_{\mathsf{lpo}} t$ then we obtain $s_i\sigma >_{\mathsf{lpo}} t\sigma$ from the induction hypothesis. So in both cases we have $s_i\sigma >^{=}_{\mathsf{lpo}} t\sigma$ and hence $s\sigma >_{\mathsf{lpo}} t\sigma\ \langle 3,i\rangle$. □

Next we show that the lexicographic path order is a proper order. The lengthy proof is quite straightforward.

Lemma 4.3.7. *Let $>$ be a precedence. The lexicographic path order $>_{\mathsf{lpo}}$ is a proper order.*

Proof We have to show that $>_{\mathsf{lpo}}$ is irreflexive and transitive, but first we show

$$s = f(s_1,\ldots,s_n) >_{\mathsf{lpo}} g(t_1,\ldots,t_m) = t \text{ implies } s >_{\mathsf{lpo}} t_i \text{ for all } 1 \leqslant i \leqslant m$$

by induction on $\|s\| + \|t\|$. If $\|s\| + \|t\| = 2$ then $s >_{\mathsf{lpo}} t\ \langle 2\rangle$ because $s_1,\ldots,s_n \in \mathcal{V}$ and thus $s >_{\mathsf{lpo}} t_i$ for all $1 \leqslant i \leqslant m$ by assumption. Suppose $\|s\| + \|t\| > 2$. If $s >_{\mathsf{lpo}} t\ \langle 1,i\rangle$ then $s >_{\mathsf{lpo}} t_j\ \langle 3,j\rangle$ for all $1 \leqslant j < i$, $s >_{\mathsf{lpo}} t_i\ \langle 3,i\rangle$, and $s >_{\mathsf{lpo}} t_j$ for all $i < j \leqslant m$ by assumption. If $s >_{\mathsf{lpo}} t\ \langle 2\rangle$ then $s >_{\mathsf{lpo}} t_i$ for all $1 \leqslant i \leqslant m$ by assumption. Suppose $s >_{\mathsf{lpo}} t\ \langle 3,j\rangle$. Let $1 \leqslant i \leqslant m$. If $s_j = t$ then $s_j >_{\mathsf{lpo}} t_i\ \langle 3,i\rangle$. If $s_j >_{\mathsf{lpo}} t$ then $s_j >_{\mathsf{lpo}} t_i$ by the induction hypothesis. So in both cases we have $s_j >_{\mathsf{lpo}} t_i$ and hence $s >_{\mathsf{lpo}} t_i\ \langle 3,j\rangle$.

We continue with the transitivity of $>_{\mathsf{lpo}}$. Suppose $s = f(s_1,\ldots,s_n) >_{\mathsf{lpo}} t$ and $t = g(t_1,\ldots,t_m) >_{\mathsf{lpo}} u$. We have to show $s >_{\mathsf{lpo}} u$. This will be established by induction on $\|s\| + \|t\| + \|u\|$. The base of the induction is $\|s\| = \|t\| = 1$ and $\|u\| = 0$. Since $u \in \mathcal{V}$, we must have $t >_{\mathsf{lpo}} u\ \langle 3,i\rangle$ and hence $s >_{\mathsf{lpo}} t_i = u$. For the induction step, suppose $s' >_{\mathsf{lpo}} t' >_{\mathsf{lpo}} u'$ implies $s' >_{\mathsf{lpo}} u'$ for all terms s', t', and u' with $\|s'\| + \|t'\| + \|u'\| < k$ $(k > 2)$. We may of course strengthen the induction hypothesis to $s' >^{=}_{\mathsf{lpo}} t' >_{\mathsf{lpo}} u'$ or $s' >_{\mathsf{lpo}} t' >^{=}_{\mathsf{lpo}} u'$ implies $s' >_{\mathsf{lpo}} u'$ whenever $\|s'\| + \|t'\| + \|u'\| < k$. Let $\|s\| + \|t\| + \|u\| = k$. In principle there are nine cases to consider, but the following table shows that we can do considerably better:

	$t >_{\mathsf{lpo}} u\ \langle 1\rangle$	$t >_{\mathsf{lpo}} u\ \langle 2\rangle$	$t >_{\mathsf{lpo}} u\ \langle 3\rangle$
$s >_{\mathsf{lpo}} t\ \langle 1\rangle$	1	2	4
$s >_{\mathsf{lpo}} t\ \langle 2\rangle$	3	3	4
$s >_{\mathsf{lpo}} t\ \langle 3\rangle$	4	4	4

1 Suppose $s >_{\mathsf{lpo}} t\ \langle 1,i\rangle$ and $t >_{\mathsf{lpo}} u\ \langle 1,j\rangle$. Let $k = \min\{i,j\}$. We show $s >_{\mathsf{lpo}} u\ \langle 1,k\rangle$. For that purpose we have to show (a) $s_l = u_l$ for all $1 \leqslant l < k$, (b) $s_k >_{\mathsf{lpo}} u_k$, and (c) $s >_{\mathsf{lpo}} u_l$ for all $k < l \leqslant n$.

(a) Since $l < i$ and $l < j$ we have $s_l = t_l = u_l$.

(b) If $k = i$ and $k < j$ then $s_k >_{\mathsf{lpo}} t_k = u_k$. If $k = j$ and $k < i$ then $s_k = t_k >_{\mathsf{lpo}} u_k$. If $k = i = j$ then $s_k >_{\mathsf{lpo}} t_k >_{\mathsf{lpo}} u_k$. In the first two cases we clearly have $s_k >_{\mathsf{lpo}} u_k$. In the third case $s_k >_{\mathsf{lpo}} u_k$ follows from the induction hypothesis.

(c) Since $t >_{\mathsf{lpo}} u_l$ the desired $s >_{\mathsf{lpo}} u_l$ follows from the induction hypothesis.

2 Suppose $s >_{\mathsf{lpo}} t\ \langle 1\rangle$ and $t >_{\mathsf{lpo}} u\ \langle 2\rangle$. We have $u = h(u_1, \ldots, u_p)$ with $f = g > h$ and $t >_{\mathsf{lpo}} u_i$ for all $1 \leqslant i \leqslant p$. The induction hypothesis yields $s >_{\mathsf{lpo}} u_i$ for all $1 \leqslant i \leqslant p$ and hence $s >_{\mathsf{lpo}} u\ \langle 2\rangle$.

3 If $s >_{\mathsf{lpo}} t\ \langle 2\rangle$ and $t >_{\mathsf{lpo}} u\ \langle 1\rangle$ or $t >_{\mathsf{lpo}} u\ \langle 2\rangle$ then $f > g$ and $u = h(u_1, \ldots, u_m)$ with $g \geqslant h$. Clearly $f > h$. Let $1 \leqslant i \leqslant m$. Since $t >_{\mathsf{lpo}} u_i$ we obtain $s >_{\mathsf{lpo}} u_i$ from the induction hypothesis. Therefore $s >_{\mathsf{lpo}} u\ \langle 2\rangle$.

4 If $s >_{\mathsf{lpo}} t$ and $t >_{\mathsf{lpo}} u\ \langle 3, i\rangle$ then $s >_{\mathsf{lpo}} t_i >^{=}_{\mathsf{lpo}} u$ and thus $s >_{\mathsf{lpo}} u$ by an application of the induction hypothesis. If $s >_{\mathsf{lpo}} t\ \langle 3, i\rangle$ and $t >_{\mathsf{lpo}} u$ then $s_i >^{=}_{\mathsf{lpo}} t$ and thus $s_i >_{\mathsf{lpo}} u$ by the induction hypothesis. Therefore $s >_{\mathsf{lpo}} u\ \langle 3, i\rangle$.

It remains to show that $>_{\mathsf{lpo}}$ is irreflexive. If $>_{\mathsf{lpo}}$ is not irreflexive then there exists a minimal term t such that $t >_{\mathsf{lpo}} t$. Since $t >_{\mathsf{lpo}} t\ \langle 2\rangle$ is clearly impossible, we must have $t >_{\mathsf{lpo}} t\ \langle 1, i\rangle$ or $t >_{\mathsf{lpo}} t\ \langle 3, i\rangle$. In the former case $t_i >_{\mathsf{lpo}} t_i$ and as t_i is a proper subterm of t this contradicts the minimality of t. In the latter case we have $t_i >^{=}_{\mathsf{lpo}} t$. Since $t_i \neq t$ we have $t_i >_{\mathsf{lpo}} t$. Combined with $t >_{\mathsf{lpo}} t_i\ \langle 3, i\rangle$ and the transitivity of $>_{\mathsf{lpo}}$ we obtain $t_i >_{\mathsf{lpo}} t_i$, again contradicting the minimality of t_i. □

Finally we show that the lexicographic path order $>_{\mathsf{lpo}}$ contains the relation $\rhd_{\mathsf{emb}}$.

Lemma 4.3.8. *We have $\rhd_{\mathsf{emb}} \subseteq >_{\mathsf{lpo}}$ for any precedence $>$.*

Proof If $\ell \to r \in \mathcal{E}\mathsf{mb}$ then r is an argument of ℓ and thus $\ell >_{\mathsf{lpo}} r\ \langle 3\rangle$. Since $>_{\mathsf{lpo}}$ is a rewrite order, $\rhd_{\mathsf{emb}} \subseteq >_{\mathsf{lpo}}$ follows. □

Hence, when attempting to show that a given TRS is compatible with LPO, we can discard every rewrite rule whose right-hand side is properly embedded in its left-hand side, irrespective of the used precedence (cf. also Exercise 4.24).

Example 4.3.9. Consider for example the TRS of Table 3.2. Six of the ten rewrite rules can be ignored. The remaining four are $x^- \cdot x \to \mathsf{e}$, $x \cdot x^- \to \mathsf{e}$, $(x \cdot y) \cdot z \to x \cdot (y \cdot z)$, and $(x \cdot y)^- \to y^- \cdot x^-$. The last one requires the precedence $^- > \cdot$: since both y^- and x^- are properly embedded in $(x \cdot y)^-$, we obtain $(x \cdot y)^- >_{\mathsf{lpo}} y^- \cdot x^-\ \langle 2\rangle$. The third rule does not need any assumptions on the precedence: as x is properly embedded in $x \cdot y$ and $y \cdot z$ is properly embedded in $(x \cdot y) \cdot z$, we have $(x \cdot y) \cdot z >_{\mathsf{lpo}} x \cdot (y \cdot z)\ \langle 1, 1\rangle$. The first two rules require either $\cdot > \mathsf{e}$ or $^- > \mathsf{e}$.

Corollary 4.3.10. *If $>$ is a precedence on a finite signature $\mathcal{F}$ then $>_{\mathsf{lpo}}$ is a simplification order on $\mathcal{T}(\mathcal{F}, \mathcal{V})$.*

So every TRS over a finite signature that is compatible with $>_{\mathsf{lpo}}$ for some precedence $>$ is simply terminating.

The following lemma states that by extending the precedence a stronger lexicographic path order is obtained. This useful property is known as the *incrementality* of LPO.

Lemma 4.3.11. *If $>$ and $\sqsupset$ are precedences on the same signature with $> \subseteq \sqsupset$ then $>_{\mathsf{lpo}} \subseteq \sqsupset_{\mathsf{lpo}}$.*

Proof Let $s = f(s_1, \ldots, s_n) >_{\mathsf{lpo}} t$. By induction on $\|s\| + \|t\|$ we show $s \sqsupset_{\mathsf{lpo}} t$. The base case is $\|s\| = 1$ and $\|t\| = 0$. So $s = f(s_1, \ldots, s_n)$ with $s_1, \ldots, s_n \in \mathcal{V}$ and necessarily $t \in \{s_1, \ldots, s_n\}$. Clearly $s \sqsupset_{\mathsf{lpo}} t$ $\langle 3 \rangle$. Assume the lemma holds for all terms s', t' with $\|s'\| + \|t'\| < k$ $(k > 1)$. Let $\|s\| + \|t\| = k$. As usual, we distinguish three cases corresponding to the three clauses of Definition 4.3.2.

1. If $s >_{\mathsf{lpo}} t$ $\langle 1, i \rangle$ then $t = f(t_1, \ldots, t_n)$ such that $s_j = t_j$ for all $1 \leqslant j < i$, $s_i >_{\mathsf{lpo}} t_i$, and $s >_{\mathsf{lpo}} t_j$ for all $i < j \leqslant n$. The induction hypothesis yields $s_i \sqsupset_{\mathsf{lpo}} t_i$ and $s \sqsupset_{\mathsf{lpo}} t_j$ for all $i < j \leqslant n$. Hence $s \sqsupset_{\mathsf{lpo}} t$ $\langle 1, i \rangle$.
2. If $s >_{\mathsf{lpo}} t$ $\langle 2 \rangle$ then $t = g(t_1, \ldots, t_m)$ with $f > g$ and $s >_{\mathsf{lpo}} t_i$ for all $1 \leqslant i \leqslant m$. The induction hypothesis yields $s \sqsupset_{\mathsf{lpo}} t_i$ for all $1 \leqslant i \leqslant m$. Therefore we obtain $s \sqsupset_{\mathsf{lpo}} t$ $\langle 2 \rangle$ from $f \sqsupset g$.
3. If $s >_{\mathsf{lpo}} t$ $\langle 3, i \rangle$ then $s_i >_{\mathsf{lpo}}^{=} t$. If $s_i = t$ then clearly $s \sqsupset_{\mathsf{lpo}} t$ $\langle 3 \rangle$. Otherwise $s_i >_{\mathsf{lpo}} t$ and we need the induction hypothesis to infer $s_i \sqsupset_{\mathsf{lpo}} t$ and consequently $s \sqsupset_{\mathsf{lpo}} t$ $\langle 3, i \rangle$. □

Since every proper order on a finite set is well-founded, for finite signatures the following result is an immediate consequence of Corollaries 4.3.10 and 4.2.9. We study the general case in Chapter 9.

Theorem 4.3.12. *Every* TRS $\mathcal{R}$ *that is compatible with* $>_{\mathsf{lpo}}$ *for some well-founded precedence* $>$ *is terminating.*

The following result explains why LPO is a much more friendly method for obtaining termination than the polynomial interpretations of Section 4.1.

Theorem 4.3.13. *The following problems are decidable:*

1. *instance:* *a finite* TRS $\mathcal{R}$ *and a precedence* $>$
   question: *is* $\mathcal{R}$ *compatible with* $>_{\mathsf{lpo}}$?
2. *instance:* *a finite* TRS $\mathcal{R}$
   question: *does there exist a precedence* $>$ *such that* $\mathcal{R}$ *is compatible with* $>_{\mathsf{lpo}}$?

Proof

1. Let s and t be arbitrary terms. We use induction on $\|s\| + \|t\|$ to show that $s >_{\mathsf{lpo}} t$ is decidable. If $\|s\| + \|t\| = 0$ then $s \in \mathcal{V}$ and thus $s >_{\mathsf{lpo}} t$ cannot hold. Suppose $s' >_{\mathsf{lpo}} t'$ is decidable for all terms s' and t' with $\|s'\| + \|t'\| < k$ $(k > 0)$ and let $\|s\| + \|t\| = k$. If $s \in \mathcal{V}$ then $s >_{\mathsf{lpo}} t$ does not hold, so let $s = f(s_1, \ldots, s_n)$. The following case analysis makes clear that it is decidable whether $s >_{\mathsf{lpo}} t$.
   (a) Suppose $t = f(t_1, \ldots, t_n)$. According to the induction hypothesis for every $i \in \{1, \ldots, n\}$ it is decidable whether $s_j = t_j$ for all $1 \leqslant j < i$, $s_i >_{\mathsf{lpo}} t_i$, and $s >_{\mathsf{lpo}} t_j$ for all $i < j \leqslant n$. Hence $s >_{\mathsf{lpo}} t$ $\langle 1 \rangle$ is decidable.
   (b) Suppose $t = g(t_1, \ldots, t_m)$ with $f > g$. According to the induction hypothesis it is decidable whether $s >_{\mathsf{lpo}} t_i$ for all $1 \leqslant i \leqslant m$. As a consequence $s >_{\mathsf{lpo}} t$ $\langle 2 \rangle$ is decidable.
   (c) Finally, the decidability of $s >_{\mathsf{lpo}} t$ $\langle 3 \rangle$ follows from the decidability of the existence of an $i \in \{1, \ldots, n\}$ such that $s_i >_{\mathsf{lpo}}^{=} t$.

Now, since $\mathcal{R}$ is finite, we can apply the above decision procedure to determine whether $\ell >_{\mathsf{lpo}} r$ for every rewrite rule $\ell \to r$ of $\mathcal{R}$. If this holds for all rewrite rules of $\mathcal{R}$, $\mathcal{R} \subseteq >_{\mathsf{lpo}}$. Otherwise $\mathcal{R} \subseteq >_{\mathsf{lpo}}$ is false.

2 Let F be the set of function symbols appearing in the rules of $\mathcal{R}$. It is not difficult to show that we may restrict the search to precedences $>$ that are defined on F. Since F is finite, there are finitely many such precedences. Hence the result follows from part **1**. □

The decision procedure in the proof of Theorem 4.3.13 is very inefficient. Automatic termination tools employ SAT and SMT solvers to search for suitable precedences. We refer to Appendix C for details.

Lemma 4.3.14. *If $>$ is a total precedence then $>_{\mathsf{lpo}}$ is a total order on ground terms.*

Proof By induction on $\|s\| + \|t\|$ we show $s >_{\mathsf{lpo}} t$ or $t >_{\mathsf{lpo}} s$ for any two different ground terms s and t. The base case is $\|s\| = 1$ and $\|t\| = 1$, i.e., s and t are different constants. Because the precedence $>$ is total, we have either $s > t$ or $t > s$. In the former case we have $s >_{\mathsf{lpo}} t\ \langle 2\rangle$ and in the latter case $t >_{\mathsf{lpo}} s\ \langle 2\rangle$. Assume $s' >_{\mathsf{lpo}} t'$ or $t' >_{\mathsf{lpo}} s'$ for any two different ground terms s' and t' with $\|s'\| + \|t'\| < k$ $(k > 2)$, and let $\|s\| + \|t\| = k$. Suppose $s >_{\mathsf{lpo}} t$ does not hold. We have to show $t >_{\mathsf{lpo}} s$. Write $s = f(s_1, \ldots, s_n)$ and $t = g(t_1, \ldots, t_m)$. Since $s >_{\mathsf{lpo}} t\ \langle 3\rangle$ does not hold, we know that $s_i >^{=}_{\mathsf{lpo}} t$ for no $i \in \{1, \ldots, n\}$. Hence, using the induction hypothesis, we have $t >_{\mathsf{lpo}} s_i$ for all $i \in \{1, \ldots, n\}$. We distinguish the following three cases.

$f = g$ Take the smallest $i \in \{1, \ldots, n\}$ such that $s_i \neq t_i$. The existence of i is guaranteed by $s \neq t$. According to the induction hypothesis we either have $s_i >_{\mathsf{lpo}} t_i$ or $t_i >_{\mathsf{lpo}} s_i$. In the former case there must be a $j \in \{i+1, \ldots, n\}$ such that $s >_{\mathsf{lpo}} t_j$ does not hold, for otherwise $s >_{\mathsf{lpo}} t\ \langle 1\rangle$ would hold. The induction hypothesis yields $t_j >^{=}_{\mathsf{lpo}} s$ and thus $t >_{\mathsf{lpo}} s\ \langle 3, j\rangle$. In the latter case we obtain $t >_{\mathsf{lpo}} s\ \langle 1, i\rangle$ using the fact that $t >_{\mathsf{lpo}} s_j$ for all $j \in \{i+1, \ldots, n\}$.

$f > g$ Because $s >_{\mathsf{lpo}} t\ \langle 2\rangle$ does not hold, we cannot have $s >_{\mathsf{lpo}} t_i$ for all $i \in \{1, \ldots, m\}$. Hence, using the induction hypothesis, there exists an $i \in \{1, \ldots, m\}$ such that $t_i >^{=}_{\mathsf{lpo}} s$. Therefore $t >_{\mathsf{lpo}} s\ \langle 3, i\rangle$.

$g > f$ We already observed $t >_{\mathsf{lpo}} s_i$ for all $i \in \{1, \ldots, n\}$. This immediately yields $t >_{\mathsf{lpo}} s\ \langle 2\rangle$.

Since $>$ is a total precedence, there are no other cases to consider. □

Exercises

4.27 ***a*** Prove the termination of the TRS $\mathcal{R}_2$ of Table 3.1 by means of LPO.

b Prove the termination of the TRS of Table 3.3 by means of LPO.

c Show that LPO applies to the TRSs of Exercises 4.7 and 4.9.

4.28 Let $>$ be a precedence.

a Show $\mathcal{V}\mathsf{ar}(t) \subseteq \mathcal{V}\mathsf{ar}(s)$ whenever $s >_{\mathsf{lpo}} t$.

b Let $t \in \mathcal{V}$. Show $s >_{\mathsf{lpo}} t$ if and only if $s \neq t$ and $t \in \mathcal{V}\mathsf{ar}(s)$.

4.29 ***a*** Suppose we additionally require "$t \in \mathcal{V}$ or $f \not\trianglerighteq \mathsf{root}(t)$" in the third clause of Definition 4.3.2. Does this affect $>_{\mathsf{lpo}}$?

b Does $>_{\mathsf{lpo}}$ coincide with $\rhd_{\mathsf{emb}}$ whenever the precedence $>$ is the empty relation?

4.30 Construct a polynomially terminating TRS that cannot be proved terminating by LPO.

4.31 Show that the TRS consisting of the rewrite rules

$$\begin{array}{lll} 0 + x \to x & 0 \times x \to 0 & -0 \to 0 \\ \mathsf{s}(x) + y \to \mathsf{s}(x + y) & \mathsf{s}(x) \times y \to (x \times y) + y & -\mathsf{s}(x) \to \mathsf{p}(-x) \\ \mathsf{p}(x) + y \to \mathsf{p}(x + y) & \mathsf{p}(x) \times y \to (x \times y) + (-y) & -\mathsf{p}(x) \to \mathsf{s}(-x) \end{array}$$

is simply terminating.

4.32 The *multiset path order* (MPO) $>_{\mathsf{mpo}}$ is defined like $>_{\mathsf{lpo}}$ except that in case $\boxed{1}$ the arguments are compared as multisets: $\boxed{1}\ t = f(t_1, \ldots, t_n)$ and $\{s_1, \ldots, s_n\} >_{\mathsf{mpo}}^{\mathsf{mul}} \{t_1, \ldots, t_n\}$.

a Prove that $>_{\mathsf{mpo}}$ is a well-defined simplification order for any precedence $>$.

b Show that LPO and MPO are incomparable.

4.33 In this exercise we unify and strengthen LPO and MPO by incorporating a *status function* which is a mapping σ that assigns to every n-ary function symbol f either the symbol mul, indicating that the arguments of terms rooted by f are compared as multisets, or a permutation $\pi_f \colon \{1, \ldots, n\} \to \{1, \ldots, n\}$, indicating the order in which the arguments of terms rooted by f are compared in a lexicographic comparison.

a Give a formal definition of the *recursive path order* (RPO) $>_{\mathsf{rpo}}^{\sigma}$ which is parameterized by a precedence $>$ and a status function σ.

b Construct a TRS that can be proved terminating by RPO but not by LPO or MPO.

4.4 Knuth–Bendix Order

In this section we introduce another method for establishing (simple) termination. Unlike LPO, the method is not only parameterized by a precedence; also the semantic component defined below is an important ingredient.

Definition 4.4.1. A *weight function* for a signature $\mathcal{F}$ is a pair (w, w_0) consisting of a mapping $w \colon \mathcal{F} \to \mathbb{N}$ and a constant $w_0 > 0$ such that $w(c) \geqslant w_0$ for every constant $c \in \mathcal{F}$.

Definition 4.4.2. Let $\mathcal{F}$ be a signature and (w, w_0) a weight function for $\mathcal{F}$. The *weight* of a term $t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ is defined as follows:

$$w(t) = \begin{cases} w_0 & \text{if } t \text{ is a variable} \\ w(f) + \sum_{i=1}^{n} w(t_i) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

Example 4.4.3. Consider the TRS $\mathcal{R}$ of Table 3.2 and the weight function (w, w_0) with $w(\mathsf{e}) = w(\cdot) = w_0 = 1$ and $w(^-) = 0$. One readily checks $w(\ell) \geqslant w(r)$ for all rewrite rules $\ell \to r$ of $\mathcal{R}$. For instance, $w(\mathsf{e} \cdot x) = 3 > 1 = w(x)$ and $w((x \cdot y)^-) = 3 = w(y^- \cdot x^-)$.

Lemma 4.4.4. *Let (w, w_0) be a weight function. We have $w(t) \geqslant w_0$ for all terms t.*

Proof Easy induction on the structure of t. □

Definition 4.4.5. Let $>$ be a precedence. A weight function (w, w_0) is *admissible* for $>$ if $f > g$ for all function symbols g different from f, whenever f is a unary function symbol with $w(f) = 0$.

Example 4.4.6. The weight function of Example 4.4.3 is admissible for the precedence $>$ with $^- > \cdot > \mathsf{e}$ because $^-$, the unary function symbol of weight zero, is maximal in the precedence.

The Knuth–Bendix order first checks whether the *variable condition* "$|s|_x \geqslant |t|_x$ for all variables x" is satisfied. This entails that *duplicating* rewrite rules like $\mathsf{f}(x) \to \mathsf{g}(x, x)$ in which a variable occurs more often on the right-hand side than on the left-hand side cannot be oriented (from left to right). Next the weights of the two terms are compared. In case of equal weights, a case analysis like in the lexicographic path order takes place. However, only one case is recursive and in that case there is a single recursive call.

Definition 4.4.7. Let $>$ be a precedence and (w, w_0) a weight function. We define the *Knuth–Bendix order* (KBO for short) $>_{\mathsf{kbo}}$ on terms inductively as follows: $s >_{\mathsf{kbo}} t$ if $|s|_x \geqslant |t|_x$ for all $x \in \mathcal{V}$ and either

(a) $w(s) > w(t)$, or

(b) $w(s) = w(t)$ and one of the following alternatives holds:

- **1** $t \in \mathcal{V}$ and $s = f^n(t)$ for some unary function symbol f and $n > 0$,
- **2** $s = f(s_1, \ldots, s_n)$, $t = f(t_1, \ldots, t_n)$, and there exists an $i \in \{1, \ldots, n\}$ such that $s_j = t_j$ for all $1 \leqslant j < i$ and $s_i >_{\mathsf{kbo}} t_i$, or
- **3** $s = f(s_1, \ldots, s_n)$, $t = g(t_1, \ldots, t_m)$, and $f > g$.

We find it convenient to write $s >_{\mathsf{kbo}} t \ \langle i \rangle$ if $w(s) = w(t)$ and $s >_{\mathsf{kbo}} t$ by clause (b) **i** in the above definition. Moreover, if $s >_{\mathsf{kbo}} t \ \langle 2 \rangle$ then we add the index i such that $s_i >_{\mathsf{kbo}} t_i$ and $s_j = t_j$ for all $1 \leqslant j < i$.

Example 4.4.8. Consider the TRS $\mathcal{R}$ of Table 3.2, the precedence $>$ of Example 4.4.6, and the weight function (w, w_0) of Example 4.4.3. We claim that the induced $>_{\mathsf{kbo}}$ orients the rewrite rules from $\mathcal{R}$ from left to right. Clearly every rewrite rule $\ell \to r$ satisfies the variable condition. If $\ell \to r$ is one of the rules $\mathsf{e} \cdot x \to x$, $x \cdot \mathsf{e} \to x$, $x^- \cdot x \to \mathsf{e}$, $x \cdot x^- \to \mathsf{e}$, $x^- \cdot (x \cdot y) \to y$, or $x \cdot (x^- \cdot y) \to y$, then $w(\ell) > w(r)$ and thus $\ell >_{\mathsf{kbo}} r$ by clause (a). For the four remaining rules both sides have equal weight. If $\ell \to r$ is one of the rules $\mathsf{e}^- \to \mathsf{e}$ or $(x \cdot y)^- \to y^- \cdot x^-$ then $\ell >_{\mathsf{kbo}} r \ \langle 3 \rangle$. We have $x^{--} >_{\mathsf{kbo}} x \ \langle 1 \rangle$ and $(x \cdot y) \cdot z >_{\mathsf{kbo}} x \cdot (y \cdot z) \ \langle 2, 1 \rangle$ because $x \cdot y >_{\mathsf{kbo}} x$ by clause (a). Hence $\mathcal{R}$ and $>_{\mathsf{kbo}}$ are compatible. According to Theorem 4.4.13 below, this is sufficient to conclude that $\mathcal{R}$ is terminating.

The next lemma states some easy properties of KBO.

Lemma 4.4.9. *Let $>$ be a precedence and (w, w_0) an admissible weight function.*
1 *If $s >_{\mathsf{kbo}} t$ then $s \notin \mathcal{V}$.*
2 *If $t \in \mathcal{V}\mathsf{ar}(s)$ and $s \neq t$ then $s >_{\mathsf{kbo}} t$.*

Proof

1 Suppose $s \in \mathcal{V}$ and $s >_{\mathsf{kbo}} t$. We have $w(s) = w_0$ and $w(t) \geqslant w_0$ according to Lemma 4.4.4. Hence $w(s) = w(t)$. Clearly, none of the three cases in the definition of the Knuth–Bendix order applies, contradicting $s >_{\mathsf{kbo}} t$.

2 Because $t \in \mathcal{V}\mathsf{ar}(s)$, the variable condition is clearly satisfied. Furthermore, $w(s) \geqslant w(t)$. If $w(s) = w(t)$ then there must be a unary function symbol f with $w(f) = 0$ such that $s = f^n(t)$ for some $n > 0$. Hence $s >_{\mathsf{kbo}} t\ \langle 1\rangle$. If $w(s) > w(t)$ then $s >_{\mathsf{kbo}} t$ by clause (a). □

Lemma 4.4.10. *Let $>$ be a precedence and (w, w_0) an admissible weight function. The Knuth–Bendix order $>_{\mathsf{kbo}}$ is a rewrite order.*

Proof First we show that $>_{\mathsf{kbo}}$ is irreflexive. If $>_{\mathsf{kbo}}$ is not irreflexive then there exists a minimal term t such that $t >_{\mathsf{kbo}} t$. This is only possible if $t >_{\mathsf{kbo}} t\ \langle 2, i\rangle$. Since t_i is a proper subterm of t with $t_i >_{\mathsf{kbo}} t_i$, this contradicts the minimality of t.

Next we show that $>_{\mathsf{kbo}}$ is transitive. Suppose $s >_{\mathsf{kbo}} t$ and $t >_{\mathsf{kbo}} u$. We use induction on $|s| + |t| + |u|$ to show $s >_{\mathsf{kbo}} u$. We have $|s|_x \geqslant |t|_x$ and $|t|_x \geqslant |u|_x$, and thus $|s|_x \geqslant |u|_x$ for all variables $x \in \mathcal{V}$. Furthermore $w(s) \geqslant w(t)$ and $w(t) \geqslant w(u)$ imply $w(s) \geqslant w(u)$. If $w(s) > w(u)$ then we are done. So suppose $w(s) = w(t) = w(u)$. By Lemma 4.4.9 **1**, $s, t \notin \mathcal{V}$ and hence we have either **1** $s >_{\mathsf{kbo}} t\ \langle 2, i\rangle$ or **2** $s >_{\mathsf{kbo}} t\ \langle 3\rangle$. Let $s = f(s_1, \ldots, s_n)$.

1 We have $t = f(t_1, \ldots, t_n)$. If $t >_{\mathsf{kbo}} u\ \langle 1\rangle$ then $u \in \mathcal{V}$ and thus $s >_{\mathsf{kbo}} u$ by Lemma 4.4.9 **2**. Next suppose $t >_{\mathsf{kbo}} u\ \langle 2, j\rangle$. Write $u = f(u_1, \ldots, u_n)$ and let $k = \min\{i, j\}$. We have $s_k >^{=}_{\mathsf{kbo}} t_k$ and $t_k >^{=}_{\mathsf{kbo}} u_k$, where at least one of these inequalities is strict. If both are strict then we obtain $s_k >_{\mathsf{kbo}} u_k$ from the induction hypothesis, otherwise $s_k >_{\mathsf{kbo}} u_k$ holds by assumption. For all $1 \leqslant l < k$ we have $s_l = t_l = u_l$. Hence $s >_{\mathsf{kbo}} u\ \langle 2, k\rangle$. Finally, if $t >_{\mathsf{kbo}} u\ \langle 3\rangle$ then $u = g(u_1, \ldots, u_m)$ and with $f > g$ and thus also $s >_{\mathsf{kbo}} u\ \langle 3\rangle$.

2 We have $f > \mathsf{root}(t)$. If $t >_{\mathsf{kbo}} u\ \langle 1\rangle$ then $u \in \mathcal{V}$ and thus $s >_{\mathsf{kbo}} u$ by Lemma 4.4.9 **2**. (Actually, it is not difficult to show that this case cannot happen.) If $t >_{\mathsf{kbo}} u\ \langle 2, i\rangle$ then $\mathsf{root}(t) = \mathsf{root}(u)$ and thus $s >_{\mathsf{kbo}} u\ \langle 3\rangle$. If $t >_{\mathsf{kbo}} u\ \langle 3\rangle$ then $\mathsf{root}(t) > \mathsf{root}(u)$. Hence $f > \mathsf{root}(u)$ and thus $s >_{\mathsf{kbo}} u\ \langle 3\rangle$.

It remains to show that $>_{\mathsf{kbo}}$ is closed under contexts and substitutions. Assume $s >_{\mathsf{kbo}} t$. For closure under contexts it suffices to show $C[s] >_{\mathsf{kbo}} C[t]$ for every context C of the form $f(t_1, \ldots, \Box, \ldots, t_n)$. Let i be the position of $\Box$ in C. We clearly have $|C[s]|_x \geqslant |C[t]|_x$ for all variables $x \in \mathcal{V}$ and $w(C[s]) - w(C[t]) = w(s) - w(t) \geqslant 0$. If $w(C[s]) - w(C[t]) = 0$ then $C[s] >_{\mathsf{kbo}} C[t]\ \langle 2, i\rangle$. Let σ be an arbitrary substitution. We show $s\sigma >_{\mathsf{kbo}} t\sigma$ by induction on $|s|$. Let $V = \mathcal{V}\mathsf{ar}(s) \cup \mathcal{V}\mathsf{ar}(t)$ and $F = \mathcal{F}\mathsf{un}(s) \cup \mathcal{F}\mathsf{un}(t)$. Because $|s|_x \geqslant |t|_x$

and $w(\sigma(x)) \geqslant w_0$ for all variables x, we obtain $|s\sigma|_x \geqslant |t\sigma|_x$ for all variables x and

$$\begin{aligned} w(s\sigma) - w(t\sigma) &= \sum_{x \in V} w(\sigma(x)) \cdot (|s|_x - |t|_x) + \sum_{f \in F} w(f) \cdot (|s|_f - |t|_f) \\ &\geqslant \sum_{x \in V} w_0 \cdot (|s|_x - |t|_x) + \sum_{f \in F} w(f) \cdot (|s|_f - |t|_f) \\ &= w(s) - w(t) \geqslant 0 \end{aligned}$$

Suppose $w(s\sigma) = w(t\sigma)$. If $s >_{\mathsf{kbo}} t$ $\langle 2 \rangle$ then $s\sigma >_{\mathsf{kbo}} t\sigma$ $\langle 2 \rangle$ follows with help of the induction hypothesis. If $s >_{\mathsf{kbo}} t$ $\langle 3 \rangle$ then $s\sigma >_{\mathsf{kbo}} t\sigma$ $\langle 3 \rangle$ because $\mathsf{root}(s\sigma) = \mathsf{root}(s)$ and $\mathsf{root}(t\sigma) = \mathsf{root}(t)$. The interesting case is $s >_{\mathsf{kbo}} t$ $\langle 1 \rangle$. So $s = f^n(t)$ and $t \in \mathcal{V}$ for some $n > 0$ and unary function symbol f with $w(f) = 0$. By a second induction on $\sigma(t)$ we show $s\sigma >_{\mathsf{kbo}} t\sigma$. If $\sigma(t)$ is a variable then $s\sigma >_{\mathsf{kbo}} t\sigma$ $\langle 1 \rangle$. If $\sigma(t) = f(u)$ then $f^n(u) >_{\mathsf{kbo}} u$ by the (second) induction hypothesis and thus $s\sigma = f^{n+1}(u) >_{\mathsf{kbo}} f(u) = t\sigma$ $\langle 2 \rangle$. Finally let $\sigma(t) = g(u_1, \ldots, u_m)$. Because f is a unary function symbol of weight 0, we must have $f > g$ and thus $s\sigma >_{\mathsf{kbo}} t\sigma$ $\langle 3 \rangle$. □

Lemma 4.4.11. *Let $>$ be a precedence and (w, w_0) an admissible weight function. The Knuth–Bendix order $>_{\mathsf{kbo}}$ has the subterm property.*

Proof Let t be a term and $p \neq \epsilon$ a position in t. We show $t >_{\mathsf{kbo}} t|_p$. Let x be a variable that does not appear in t. We have $t[x]_p >_{\mathsf{kbo}} x$ by Lemma 4.4.9 **2**. Let $\sigma = \{x \mapsto t|_p\}$. Closure under substitutions yields $t = t[x\sigma]_p = t[x]_p\sigma >_{\mathsf{kbo}} x\sigma = t|_p$. □

Corollary 4.4.12. *If $>$ is a precedence on a finite signature $\mathcal{F}$ and (w, w_0) an admissible weight function then $>_{\mathsf{kbo}}$ is a simplification order.*

So every TRS over a finite signature that is compatible with $>_{\mathsf{kbo}}$ for some precedence $>$ is simply terminating. In Chapter 9 we lift the restriction that the signature must be finite.

Theorem 4.4.13. *Every* TRS *that is compatible with $>_{\mathsf{kbo}}$ for some well-founded precedence $>$ and admissible weight function (w, w_0) is terminating.*

We consider two more examples. The first one involves an SRS. For SRSs the value of w_0 is irrelevant because there are no constants and both sides of a rewrite rule contain exactly one occurrence of a variable.

Example 4.4.14. Consider the SRS $\mathcal{R}$ consisting of the two rewrite rules

$$\mathsf{aa} \to \mathsf{bbb} \qquad\qquad \mathsf{bbbbb} \to \mathsf{aaa}$$

If we take $w(\mathsf{a}) = 3$ and $w(\mathsf{b}) = 2$ then the second rewrite rule is oriented by weight. For the first rewrite rule we need $\mathsf{a} > \mathsf{b}$. This is just one way to prove the termination of $\mathcal{R}$ with KBO. Another possibility is $w(\mathsf{a}) = 5$ and $w(\mathsf{b}) = 3$ with $\mathsf{b} > \mathsf{a}$. If we take $w(\mathsf{a}) = 8$ and $w(\mathsf{b}) = 5$ instead, there is no need to compare a and b in the precedence.

Example 4.4.15. Consider the TRS $\mathcal{R}$ of Table 3.4. By taking the weight function

$$w(\mathsf{0}) = w(\mathsf{1}) = w(\mathsf{2}) = w(\mathsf{3}) = w(\mathsf{4}) = w(+) = w_0 = 1 \qquad w(:) = 2$$
$$w(\mathsf{5}) = w(\mathsf{6}) = w(\mathsf{7}) = w(\mathsf{8}) = w(\mathsf{9}) = 3$$

together with the precedence $+ > f$ for all $f \in \{\mathsf{5}, \mathsf{6}, \mathsf{7}, \mathsf{8}, :\}$, all 104 rewrite rules of $\mathcal{R}$ are oriented by $>_{\mathsf{kbo}}$ from left to right.

The second part of the following result is important for automatically proving termination of TRSs by KBO. Because of the infinite search space—recall that the weight of a function symbol can be an arbitrary natural number—it is much harder to prove than the corresponding result for LPO (Theorem 4.3.13).

Theorem 4.4.16. *The following problems are decidable:*

1. *instance:* *a finite* TRS $\mathcal{R}$*, a precedence* $>$*, and a weight function* (w, w_0)
   question: *is* $\mathcal{R}$ *compatible with* $>_{\mathsf{kbo}}$?
2. *instance:* *a finite* TRS $\mathcal{R}$
   question: *does there exist a precedence* $>$ *and a weight function* (w, w_0) *such that* $\mathcal{R}$ *is compatible with* $>_{\mathsf{kbo}}$?

Lemma 4.4.17. *Let* (w, w_0) *be an arbitrary weight function. If* $>$ *is a total precedence then* $>_{\mathsf{kbo}}$ *is a total order on ground terms.*

Proof Let s and t be different ground terms. We prove $s >_{\mathsf{kbo}} t$ or $t >_{\mathsf{kbo}} s$ by induction on $|s|$. The variable condition is vacuously satisfied. If $w(s) > w(t)$ then $s >_{\mathsf{kbo}} t$. If $w(t) > w(s)$ then $t >_{\mathsf{kbo}} s$. So suppose $w(s) = w(t)$. Let $f = \mathsf{root}(s)$ and $g = \mathsf{root}(t)$. If $f \neq g$ then, because $>$ is total, either $f > g$ and thus $s >_{\mathsf{kbo}} t\ \langle 3 \rangle$ or $g > f$ and $t >_{\mathsf{kbo}} s\ \langle 3 \rangle$. In the remaining case we have $f = g$. Write $s = f(s_1, \ldots, s_n)$ and $t = f(t_1, \ldots, t_n)$. Since $s \neq t$, there must be an $i \in \{1, \ldots, n\}$ such that $s_j = t_j$ for all $1 \leqslant j < i$ and $s_i \neq t_i$. According to the induction hypothesis $s_i >_{\mathsf{kbo}} t_i$ or $t_i >_{\mathsf{kbo}} s_i$ and thus $s >_{\mathsf{kbo}} t\ \langle 2, i \rangle$ or $t >_{\mathsf{kbo}} s\ \langle 2, i \rangle$. □

Figure 4.1 compares the termination methods introduced so far. The dashed areas indicate that we cannot decide membership.

We conclude this section with presenting an order that combines LPO and KBO. Instead of a weight function, a simple monotone algebra (cf. Definition 4.2.12) is employed for the initial term comparison.

Definition 4.4.18. Let $\sqsupset$ be a precedence and $(\mathcal{A}, >)$ a simple monotone algebra. We define the *weighted path order* (WPO for short) $>_{\mathsf{wpo}}$ on terms inductively as follows: $s >_{\mathsf{wpo}} t$ if either $s >_{\mathcal{A}} t$ or $s \geqslant_{\mathcal{A}} t$, $s = f(s_1, \ldots, s_n)$ and one of the following alternatives holds:

1. $t = f(t_1, \ldots, t_n)$ and there exists an $i \in \{1, \ldots, n\}$ such that
   (a) $s_j = t_j$ for all $1 \leqslant j < i$,
   (b) $s_i >_{\mathsf{wpo}} t_i$, and
   (c) $s >_{\mathsf{wpo}} t_j$ for all $i < j \leqslant n$,

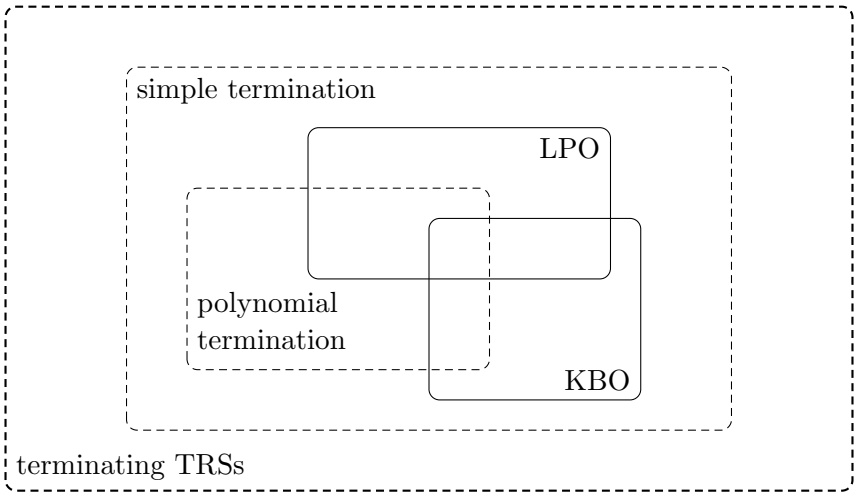


Figure 4.1: Comparison of termination methods.

2 $t = g(t_1, \ldots, t_m)$, $f \sqsupset g$, and $s >_{\mathsf{wpo}} t_j$ for all $j \in \{1, \ldots, m\}$, or
3 $s_i \geq^{=}_{\mathsf{wpo}} t$ for some $i \in \{1, \ldots, n\}$.

Example 4.4.19. Consider the SRS $\mathcal{R}$ consisting of the two rewrite rules

$$\mathsf{ab} \to \mathsf{baa} \qquad\qquad \mathsf{ac} \to \mathsf{cca}$$

Using the simple monotone algebra $(\mathcal{A}, >_{\mathbb{N}})$ with $\mathsf{a}_{\mathcal{A}}(x) = \mathsf{c}_{\mathcal{A}}(x) = x$ and $\mathsf{b}_{\mathcal{A}}(x) = x + 1$ together with the precedence $\mathsf{a} \sqsupset \mathsf{b}$ and $\mathsf{a} \sqsupset \mathsf{c}$ enables WPO to orient both rules:

$$\dfrac{\mathsf{ab} \geqslant_{\mathcal{A}} \mathsf{baa} \quad \mathsf{a} \sqsupset \mathsf{b} \quad \dfrac{\mathsf{ab} >_{\mathcal{A}} \mathsf{aa}}{\mathsf{ab} >_{\mathsf{wpo}} \mathsf{aa}}}{\mathsf{ab} >_{\mathsf{wpo}} \mathsf{baa}}\,\boxed{2}$$

$$\dfrac{\mathsf{ac} \geqslant_{\mathcal{A}} \mathsf{cca} \quad \mathsf{a} \sqsupset \mathsf{c} \quad \dfrac{\mathsf{ac} \geqslant_{\mathcal{A}} \mathsf{ca} \quad \mathsf{a} \sqsupset \mathsf{c} \quad \dfrac{\mathsf{ac} \geqslant_{\mathcal{A}} \mathsf{a} \quad \dfrac{\mathsf{c} \geqslant_{\mathcal{A}} \epsilon \quad \epsilon = \epsilon}{\mathsf{c} >_{\mathsf{wpo}} \epsilon}\,\boxed{3}}{\mathsf{ac} >_{\mathsf{wpo}} \mathsf{a}}\,\boxed{1}}{\mathsf{ac} >_{\mathsf{wpo}} \mathsf{ca}}\,\boxed{2}}{\mathsf{ac} >_{\mathsf{wpo}} \mathsf{cca}}\,\boxed{2}$$

Theorem 4.4.20. *If $\sqsupset$ is precedence on a finite signature and $(\mathcal{A}, >)$ a simple monotone algebra then $>_{\mathsf{wpo}}$ is simplification order.*

In connection with Theorem 4.2.13 we conclude that WPO characterizes the class of simply terminating TRSs.

Corollary 4.4.21. *A finite* TRS *is simply terminating if and only if it is compatible with $>_{\mathsf{wpo}}$ for some precedence $\sqsupset$ and simple monotone algebra $(\mathcal{A}, >)$.*

The challenge when using WPO is how to find a suitable simple monotone algebra. According to the proof of Corollary 4.2.14, any weakly monotone polynomial interpretation can be used, but we are not restricted to polynomial interpretations. The final example in this section shows the use of the weakly monotone maximum function.

Example 4.4.22. Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\mathsf{f}(x,y) \to \mathsf{g}(x) \qquad \mathsf{f}(\mathsf{g}(x),y) \to \mathsf{f}(x,\mathsf{g}(x)) \qquad \mathsf{f}(x,\mathsf{g}(y)) \to \mathsf{f}(y,y)$$

Consider the simple monotone algebra $(\mathcal{A},>_{\mathbb{N}})$ with $\mathsf{f}_{\mathcal{A}}(x,y) = \mathsf{max}(x,y)+1$ and $\mathsf{g}_{\mathcal{A}}(x) = x+1$. We have

$$\mathsf{f}(x,y) \geqslant_{\mathcal{A}} \mathsf{g}(x) \qquad \mathsf{f}(\mathsf{g}(x),y) \geqslant_{\mathcal{A}} \mathsf{f}(x,\mathsf{g}(x)) \qquad \mathsf{f}(x,\mathsf{g}(y)) >_{\mathcal{A}} \mathsf{f}(y,y)$$

So the rule $\mathsf{f}(x,\mathsf{g}(y)) \to \mathsf{f}(y,y)$ is compatible with $>_{\mathsf{wpo}}$. For $\mathsf{f}(x,y) \to \mathsf{g}(x)$ we use the precedence $\mathsf{f} \sqsupset \mathsf{g}$ and the remaining rule $\mathsf{f}(\mathsf{g}(x),y) \to \mathsf{f}(x,\mathsf{g}(x))$ is handled by case $\boxed{1}$ in the definition of WPO.

Exercises

4.34 Prove the termination of the TRS consisting of the rewrite rules

$$\begin{aligned} \mathsf{average}(0,0) &\to 0 & \mathsf{average}(\mathsf{s}(x),y) &\to \mathsf{average}(x,\mathsf{s}(y)) \\ \mathsf{average}(0,\mathsf{s}(0)) &\to 0 & \mathsf{average}(x,\mathsf{s}(\mathsf{s}(\mathsf{s}(y)))) &\to \mathsf{s}(\mathsf{average}(\mathsf{s}(x),y)) \\ \mathsf{average}(0,\mathsf{s}(\mathsf{s}(0))) &\to \mathsf{s}(0) \end{aligned}$$

using KBO.

4.35 Prove the termination of the TRS consisting of the rewrite rules

$$\begin{aligned} \ominus(0) &\to 0 & x-0 &\to x & \mathsf{s}(\mathsf{p}(x)) &\to x \\ \ominus(\mathsf{s}(x)) &\to \mathsf{p}(\ominus(x)) & x-\mathsf{s}(y) &\to \mathsf{p}(x-y) & \mathsf{p}(\mathsf{s}(x)) &\to x \\ \ominus(\mathsf{p}(x)) &\to \mathsf{s}(\ominus(x)) & x-\mathsf{p}(y) &\to \mathsf{s}(x-y) & x+y &\to x-\ominus(y) \end{aligned}$$

using KBO.

4.36 Show that the Knuth–Bendix order $>_{\mathsf{kbo}}$ induced by a weight function (w,w_0) and a precedence $>$ need not be well-founded if (w,w_0) is not admissible for $>$.

4.37 Prove the termination of the TRS consisting of the rewrite rules

$$\begin{aligned} \mathsf{p1}+\mathsf{p1} &\to \mathsf{p2} & \mathsf{p1}+(\mathsf{p1}+x) &\to \mathsf{p2}+x \\ \mathsf{p2}+\mathsf{p1} &\to \mathsf{p1}+\mathsf{p2} & \mathsf{p2}+(\mathsf{p1}+x) &\to \mathsf{p1}+(\mathsf{p2}+x) \\ \mathsf{p5}+\mathsf{p1} &\to \mathsf{p1}+\mathsf{p5} & \mathsf{p5}+(\mathsf{p1}+x) &\to \mathsf{p1}+(\mathsf{p5}+x) \\ \mathsf{p5}+\mathsf{p2} &\to \mathsf{p2}+\mathsf{p5} & \mathsf{p5}+(\mathsf{p2}+x) &\to \mathsf{p2}+(\mathsf{p5}+x) \\ \mathsf{p5}+\mathsf{p5} &\to \mathsf{p10} & \mathsf{p5}+(\mathsf{p5}+x) &\to \mathsf{p10}+x \\ \mathsf{p10}+\mathsf{p1} &\to \mathsf{p1}+\mathsf{p10} & \mathsf{p10}+(\mathsf{p1}+x) &\to \mathsf{p1}+(\mathsf{p10}+x) \\ \mathsf{p10}+\mathsf{p2} &\to \mathsf{p2}+\mathsf{p10} & \mathsf{p10}+(\mathsf{p2}+x) &\to \mathsf{p2}+(\mathsf{p10}+x) \\ \mathsf{p10}+\mathsf{p5} &\to \mathsf{p5}+\mathsf{p10} & \mathsf{p10}+(\mathsf{p5}+x) &\to \mathsf{p5}+(\mathsf{p10}+x) \\ \mathsf{p1}+(\mathsf{p2}+\mathsf{p2}) &\to \mathsf{p5} & \mathsf{p1}+(\mathsf{p2}+(\mathsf{p2}+x)) &\to \mathsf{p5}+x \\ \mathsf{p2}+(\mathsf{p2}+\mathsf{p2}) &\to \mathsf{p1}+\mathsf{p5} & \mathsf{p2}+(\mathsf{p2}+(\mathsf{p2}+x)) &\to \mathsf{p1}+(\mathsf{p5}+x) \end{aligned}$$

$$(x + y) + z \to x + (y + z)$$

using KBO.

4.38 Show that KBO is incremental: If (w, w_0) is a weight function and $>$ and $\sqsupset$ are precedences with $> \subseteq \sqsupset$ then $>_{\mathsf{kbo}} \subseteq \sqsupset_{\mathsf{kbo}}$.

4.39 ***a*** Prove Theorem 4.4.16 **1**.

b Construct for every $k \geqslant 0$ a TRS $\mathcal{R}_k$ that can be proved terminating with KBO but only if weights larger than k are used.

c Prove Theorem 4.4.16 **2** by providing for every finite TRS $\mathcal{R}$ a computable number $b_{\mathcal{R}} \in \mathbb{N}$ such that if the termination of $\mathcal{R}$ can be proved using KBO, weights in $\{0, \ldots, b_R\}$ suffice.

4.40 Prove the termination of the SRS consisting of the rewrite rules

$11 \to 43$	$33 \to 56$	$55 \to 62$
$12 \to 21$	$34 \to 11$	$56 \to 12$
$22 \to 111$	$44 \to 3$	$66 \to 21$

using KBO.

4.41 ***a*** Prove that w_0 can be assumed to be 1 in Theorem 4.4.13.

b For any $k > 0$, prove that w_0 can be assumed to be k in Theorem 4.4.13.

c What goes wrong when $w_0 = 0$?

4.42 Consider Figure 4.1.

a Show that all areas are inhabited by TRSs.

b Incorporate MPO (Exercise 4.32) and RPO (Exercise 4.33).

4.43 Prove the termination of the TRS consisting of the rewrite rules

$$\mathsf{f}(\mathsf{a}, \mathsf{b}) \to \mathsf{f}(\mathsf{b}, \mathsf{f}(\mathsf{b}, \mathsf{a})) \qquad \mathsf{f}(\mathsf{a}, \mathsf{f}(\mathsf{b}, x)) \to \mathsf{f}(x, \mathsf{f}(\mathsf{b}, \mathsf{b}))$$

using WPO.

4.44 In this exercise we extend KBO to handle duplicating rules like $\mathsf{f}(x) \to \mathsf{g}(x, x)$. To this end we recalculate the weight of terms by incorporating a *subterm coefficient function*, which is a partial mapping $s\colon \mathcal{F} \times \mathbb{N} \to \mathbb{N}$ such that for a function symbol f of arity n we have $s(f, i) > 0$ for all $1 \leqslant i \leqslant n$. Given a weight function (w, w_0) and a subterm coefficient function s, the weight of a term is inductively defined as follows:

$$w_s(t) = \begin{cases} w_0 & \text{if } t \in \mathcal{V} \\ w(f) + \displaystyle\sum_{1 \leqslant i \leqslant n} s(f, i) \cdot w_s(t_i) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

a Consider the weight function of Example 4.4.3 together with the subterm coefficients $s(\cdot, 1) = 1$, $s(\cdot, 2) = 3$, and $s(^-, 1) = 2$. Calculate the weights of all left- and right-hand sides of the TRS of Table 3.2.

b We want to use w_s instead of w in Definition 4.4.7. Reformulate the variable condition "$|s|_x \geqslant |t|_x$ for all variables x" in Definition 4.4.7 such that we obtain a simplification order $>^s_{\mathsf{kbo}}$ which properly extends $>_{\mathsf{kbo}}$.

c Construct a precedence $>$, a weight function (w, w_0), and a subterm coefficient function s such that the TRS consisting of the rules

$$\mathsf{f}(x) \to \mathsf{g}(x, x) \qquad \mathsf{f}(\mathsf{h}(x)) \to \mathsf{h}(\mathsf{h}(\mathsf{f}(x)))$$

is compatible with $>^s_{\mathsf{kbo}}$.

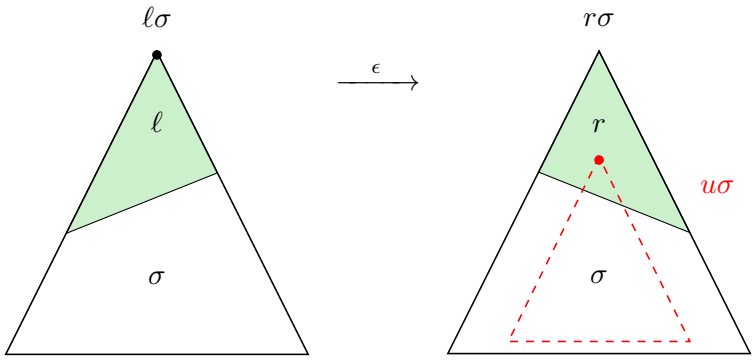


Figure 4.2: An illustration of Lemma 4.5.1.

4.45 Prove the termination of the TRS consisting of the rewrite rules

$$\mathsf{f}(\mathsf{f}(x,y),z) \to \mathsf{f}(x,\mathsf{f}(y,z)) \qquad \mathsf{g}(\mathsf{f}(\mathsf{a},x),\mathsf{b}) \to \mathsf{g}(\mathsf{f}(x,\mathsf{b}),x)$$

4.5 Dependency Pairs

In this section we introduce a termination method which is easy to automate and readily applies to TRSs that are not simply terminating. The new method does not attempt to prove termination directly. Rather, the TRS $\mathcal{R}$ under consideration is transformed into a set of ordering constraints. If a suitable order can be found which satisfies the constraints then $\mathcal{R}$ is terminating.

Let us start with some easy observations. If $\mathcal{R}$ is not terminating then there must be a *minimal* non-terminating term, minimal in the sense that all its proper subterms are terminating. Let us denote the set of all minimal non-terminating terms by $\mathcal{T}_\infty$.

Lemma 4.5.1. *For every term $t \in \mathcal{T}_\infty$ there exist a rewrite rule $\ell \to r$, a substitution σ, and a non-variable subterm u of r such that $t \xrightarrow{>\epsilon}{}^* \ell\sigma \xrightarrow{\epsilon} r\sigma \trianglerighteq u\sigma$ and $u\sigma \in \mathcal{T}_\infty$.*

Proof Let A be an infinite rewrite sequence starting at t. Since all proper subterms of t are terminating, A must contain a root rewrite step. By considering the first root rewrite step in A it follows that there exist a rewrite rule $\ell \to r$ and a substitution σ such that A starts with $t \xrightarrow{>\epsilon}{}^* \ell\sigma \xrightarrow{\epsilon} r\sigma$. Write $\ell = f(\ell_1,\ldots,\ell_n)$. Since the rewrite steps in $t \to^* \ell\sigma$ take place below the root, $t = f(t_1,\ldots,t_n)$ and $t_i \to^* \ell_i\sigma$ for all $1 \leqslant i \leqslant n$. By assumption the arguments $t_1,\ldots,t_n$ of t are terminating. Hence so are the terms $\ell_1\sigma,\ldots,\ell_n\sigma$. It follows that $\sigma(x)$ is terminating for every $x \in \mathcal{V}\mathsf{ar}(r) \subseteq \mathcal{V}\mathsf{ar}(\ell)$. As $r\sigma$ is non-terminating it has a subterm $t' \in \mathcal{T}_\infty$. Because t' cannot occur in the substitution part, there must be a non-variable subterm u of r such that $t' = u\sigma$. □

The proof is illustrated in Figure 4.2. Observe that the term $\ell\sigma$ in Lemma 4.5.1 belongs to $\mathcal{T}_\infty$ as well. Further note that $u\sigma$ cannot be a proper subterm of $\ell\sigma$ (since all arguments of $\ell\sigma$ are terminating).

Corollary 4.5.2. *Every term in $\mathcal{T}_\infty$ has a defined root symbol.*

If we were to define a new TRS $\mathcal{S}$ consisting of all rewrite rules $\ell \to u$ for which there exist a rewrite rule $\ell \to r \in \mathcal{R}$ and a subterm u of r with defined function symbol, then the sequence in the conclusion of Lemma 4.5.1 is of the form $\xrightarrow{>\epsilon}{}^*_{\mathcal{R}} \cdot \xrightarrow{\epsilon}_{\mathcal{S}}$. The idea is now to get rid of the position constraints by marking the root symbols of the terms in the rewrite rules of $\mathcal{S}$.

Definition 4.5.3. Let $\mathcal{R}$ be a TRS over a signature $\mathcal{F}$. Let $\mathcal{F}^\sharp$ denote the union of $\mathcal{F}$ and $\{ f^\sharp \mid f \in \mathcal{F}_\mathcal{D}\}$ where $f^\sharp$ is a fresh function symbol with the same arity as f. We call these new symbols *dependency pair symbols*. Given a term $t = f(t_1, \ldots, t_n) \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ with $f \in \mathcal{F}_\mathcal{D}$, we write $t^\sharp$ for the term $f^\sharp(t_1, \ldots, t_n)$. If $\ell \to r \in \mathcal{R}$ and u is a subterm of r with defined root symbol such that u is not a proper subterm of ℓ then the rewrite rule $\ell^\sharp \to u^\sharp$ is called a *dependency pair* of $\mathcal{R}$. The set of all dependency pairs of $\mathcal{R}$ is denoted by $\mathsf{DP}(\mathcal{R})$.

Although dependency pair symbols are defined symbols of $\mathsf{DP}(\mathcal{R})$, they are not defined symbols of $\mathcal{R}$. In the following, defined symbols always refer to the original TRS $\mathcal{R}$.

Example 4.5.4. The TRS of Table 4.1 has three dependency pairs:

$$\mathsf{s}(x) -^\sharp \mathsf{s}(y) \to x -^\sharp y \qquad \mathsf{s}(x) \div^\sharp \mathsf{s}(y) \to (x - y) \div^\sharp \mathsf{s}(y) \qquad \mathsf{s}(x) \div^\sharp \mathsf{s}(y) \to x -^\sharp y$$

Observe that a finite TRS admits finitely many dependency pairs.

Definition 4.5.5. For any subset $T \subseteq \mathcal{T}(\mathcal{F}, \mathcal{V})$ the set $\{t^\sharp \mid t \in T \text{ and } \mathsf{root}(t) \text{ is defined}\}$ is denoted by $T^\sharp$. We write $\mathcal{T}^\sharp$ for $\mathcal{T}(\mathcal{F}, \mathcal{V})^\sharp$. Given a term $t \in \mathcal{T}^\sharp$, we write $t^\flat$ for the term in $\mathcal{T}(\mathcal{F}, \mathcal{V})$ obtained from t by replacing its root symbol $f^\sharp$ by f.

Lemma 4.5.6. *If* $t_0 \to^*_\mathcal{R} t_1 \to_{\mathsf{DP}(\mathcal{R})} t_2 \to^*_\mathcal{R} t_3 \to_{\mathsf{DP}(\mathcal{R})} \cdots$ *with* $t_0 \in \mathcal{T}^\sharp$ *then* $t_i^\flat$ *is non-terminating for all* $i \geqslant 0$.

Proof We clearly have $t_i \in \mathcal{T}^\sharp$ for all $i \geqslant 0$. Let $j \in \{1, 3, 5, \ldots\}$ and consider the step $t_j \to_{\mathsf{DP}(\mathcal{R})} t_{j+1}$. There exist a dependency pair $\ell_j \to r_j$ and a substitution σ_j such that $t_j = \ell_j\sigma_j$ and $t_{j+1} = r_j\sigma_j$. By definition of dependency pair there exists a context C_j such that $\ell_j^\flat \to C_j[r_j^\flat]$ is a rewrite rule in $\mathcal{R}$. Define $D_j = C_j\sigma_j$. We have $t_j^\flat = \ell_j^\flat\sigma_j \to_\mathcal{R} C_j[r_j^\flat]\sigma_j = D_j[t_{j+1}^\flat]$. Hence the given sequence is transformed into the infinite rewrite sequence

$$t_0^\flat \to^*_\mathcal{R} t_1^\flat \to_\mathcal{R} D_1[t_2^\flat] \to^*_\mathcal{R} D_1[t_3^\flat] \to_\mathcal{R} D_1[D_3[t_4^\flat]] \to^*_\mathcal{R} \cdots$$

showing that $t_i^\flat$ is non-terminating for all $i \geqslant 0$. □

$0 - y \to 0$	$0 \div \mathsf{s}(y) \to 0$
$x - 0 \to x$	$\mathsf{s}(x) \div \mathsf{s}(y) \to \mathsf{s}((x - y) \div \mathsf{s}(y))$
$\mathsf{s}(x) - \mathsf{s}(y) \to x - y$	

Table 4.1: A TRS for division and subtraction of natural numbers.

Theorem 4.5.7. *A* TRS $\mathcal{R}$ *is terminating if and only if there are no infinite rewrite sequences of the form* $t_0 \to_{\mathcal{R}}^* t_1 \to_{\mathsf{DP}(\mathcal{R})} t_2 \to_{\mathcal{R}}^* t_3 \to_{\mathsf{DP}(\mathcal{R})} \cdots$ *with* $t_i \in \mathcal{T}_\infty^\sharp$ *for all* $i \geqslant 0$.

Proof The if direction is an immediate consequence of Lemma 4.5.6. For the only-if direction we use Lemma 4.5.1, Corollary 4.5.2, and the preceding definitions to conclude that for every term $s \in \mathcal{T}_\infty^\sharp$ there exist terms $t, u \in \mathcal{T}_\infty^\sharp$ such that $s \to_{\mathcal{R}}^* t \to_{\mathsf{DP}(\mathcal{R})} u$. So if $\mathcal{R}$ is non-terminating then $\mathcal{T}_\infty^\sharp$ is non-empty and we obtain an infinite rewrite sequence of the desired form. □

So to prove termination of a TRS $\mathcal{R}$ it is sufficient (and necessary) to show that $\mathcal{R} \cup \mathsf{DP}(\mathcal{R})$ does not admit infinite sequences of the form

$$t_0 \to_{\mathcal{R}}^* t_1 \to_{\mathsf{DP}(\mathcal{R})} t_2 \to_{\mathcal{R}}^* t_3 \to_{\mathsf{DP}(\mathcal{R})} \cdots$$

with $t_i \in \mathcal{T}_\infty^\sharp$ for all $i \geqslant 0$. This can be guaranteed by the notion defined below.

Definition 4.5.8. A *rewrite preorder* is a preorder on terms that is a rewrite relation. A *reduction pair* $(>, \gtrsim)$ consists of a well-founded order $>$ that is closed under substitutions and a rewrite preorder $\gtrsim$ such that $> \cdot \gtrsim \subseteq >$ or $\gtrsim \cdot > \subseteq >$.

Most reduction pairs $(>, \gtrsim)$ that we will encounter satisfy the (slightly) stronger inclusion $\gtrsim \cdot > \cdot \gtrsim \subseteq >$.

Theorem 4.5.9. *Let* $\mathcal{R}$ *be a* TRS. *If there exists a reduction pair* $(>, \gtrsim)$ *such that* $\ell \gtrsim r$ *for all* $\ell \to r \in \mathcal{R}$ *and* $\ell > r$ *for all* $\ell \to r \in \mathsf{DP}(\mathcal{R})$ *then* $\mathcal{R}$ *is terminating.*

Proof Because $\gtrsim$ is a rewrite relation we have $s \gtrsim t$ for every rewrite step $s \to_{\mathcal{R}} t$. If $s \to_{\mathsf{DP}(\mathcal{R})} t$ with $s, t \in \mathcal{T}^\sharp$ then $s = \ell\sigma$ and $t = r\sigma$ for some dependency pair $\ell \to r \in \mathsf{DP}(\mathcal{R})$ and substitution σ. Hence $s > t$ because $>$ is closed under substitutions. Now suppose $\mathcal{R}$ is non-terminating. According to Theorem 4.5.7 there exists an infinite rewrite sequence $t_0 \to_{\mathcal{R}}^* t_1 \to_{\mathsf{DP}(\mathcal{R})} t_2 \to_{\mathcal{R}}^* t_3 \to_{\mathsf{DP}(\mathcal{R})} \cdots$ with $t_i \in \mathcal{T}_\infty^\sharp$ for all $i \geqslant 0$. By the preceding observations and the fact that $\gtrsim$ is a preorder, we obtain $t_0 \gtrsim t_1 > t_2 \gtrsim t_3 > \cdots$ and, using the fact that $> \cdot \gtrsim \subseteq >$ or $\gtrsim \cdot > \subseteq >$, an infinite descending sequence starting from t_2. This contradicts the well-foundedness of $>$. □

Example 4.5.10. Consider again the TRS of Table 4.1. According to the preceding theorem, termination follows if we can find a reduction pair $(>, \gtrsim)$ such that

$$\begin{array}{lll}
0 - y \gtrsim 0 & 0 \div \mathsf{s}(y) \gtrsim 0 & \mathsf{s}(x) -^\sharp \mathsf{s}(y) > x -^\sharp y \\
x - 0 \gtrsim x & \mathsf{s}(x) \div \mathsf{s}(y) \gtrsim \mathsf{s}((x - y) \div \mathsf{s}(y)) & \mathsf{s}(x) \div^\sharp \mathsf{s}(y) > (x - y) \div^\sharp \mathsf{s}(y) \\
\mathsf{s}(x) - \mathsf{s}(y) \gtrsim x - y & & \mathsf{s}(x) \div^\sharp \mathsf{s}(y) > x -^\sharp y
\end{array}$$

Reduction pairs are generated by well-founded *weakly* monotone algebras.

Definition 4.5.11. A *weakly monotone* $\mathcal{F}$-algebra $(\mathcal{A}, >, \gtrsim)$ consists of a non-empty $\mathcal{F}$-algebra $\mathcal{A}$ together with a proper order $>$ and a preorder $\gtrsim$ on the carrier A of $\mathcal{A}$ such that $> \cdot \gtrsim \subseteq >$ or $\gtrsim \cdot > \subseteq >$, and every algebra operation is monotone with respect to $\gtrsim$

in all coordinates, i.e., if $f \in \mathcal{F}$ has arity $n \geqslant 1$ then

$$f_{\mathcal{A}}(a_1, \ldots, a_i, \ldots, a_n) \gtrsim f_{\mathcal{A}}(a_1, \ldots, b, \ldots, a_n)$$

for all $a_1, \ldots, a_n, b \in A$ and $i \in \{1, \ldots, n\}$ with $a_i \gtrsim b$. We call a weakly monotone $\mathcal{F}$-algebra $(\mathcal{A}, >, \gtrsim)$ *well-founded* if $>$ is well-founded.

Definition 4.5.12. Let $(\mathcal{A}, >, \gtrsim)$ be a weakly monotone algebra. We define relations $>_{\mathcal{A}}$ and $\gtrsim_{\mathcal{A}}$ on terms as follows:

1. $s >_{\mathcal{A}} t$ if $[\alpha]_{\mathcal{A}}(s) > [\alpha]_{\mathcal{A}}(t)$ for all assignments α,
2. $s \gtrsim_{\mathcal{A}} t$ if $[\alpha]_{\mathcal{A}}(s) \gtrsim [\alpha]_{\mathcal{A}}(t)$ for all assignments α.

The proof of the following lemma is very similar to the proof of Lemma 4.1.8.

Lemma 4.5.13. *If $(\mathcal{A}, >, \gtrsim)$ is a weakly monotone algebra then $\gtrsim_{\mathcal{A}}$ is a rewrite relation.*

Lemma 4.5.14. *If $(\mathcal{A}, >, \gtrsim)$ is a well-founded weakly monotone algebra then $(>_{\mathcal{A}}, \gtrsim_{\mathcal{A}})$ is a reduction pair.*

Proof The relation $\gtrsim_{\mathcal{A}}$ is easily shown to be a preorder. According to Lemma 4.5.13 it is a rewrite relation and hence a rewrite preorder. Lemma 4.1.8 states that $>_{\mathcal{A}}$ is a reduction order for well-founded monotonic algebras $(\mathcal{A}, >, \gtrsim)$. An inspection of the proof reveals that monotonicity is only used to infer the closure under contexts of $>_{\mathcal{A}}$. Hence, for well-founded weakly monotone algebras $(\mathcal{A}, >, \gtrsim)$, $>_{\mathcal{A}}$ is a well-founded order closed under substitutions. It remains to show $>_{\mathcal{A}} \cdot \gtrsim_{\mathcal{A}} \subseteq >_{\mathcal{A}}$ or $\gtrsim_{\mathcal{A}} \cdot >_{\mathcal{A}} \subseteq >_{\mathcal{A}}$. This is an easy consequence of the definitions. □

Example 4.5.15. We can easily solve the constraints of Example 4.5.10 by a weakly monotone polynomial interpretation. Formally, consider the algebra $\mathcal{A}$ with carrier $\mathbb{N}$, standard order $>_{\mathbb{N}}$, preorder $\geqslant_{\mathbb{N}}$, and interpretations $-_{\mathcal{A}}(x, y) = \div_{\mathcal{A}}(x, y) = -^{\sharp}_{\mathcal{A}}(x, y) = \div^{\sharp}_{\mathcal{A}}(x, y) = x$, $\mathsf{s}_{\mathcal{A}}(x) = x + 1$, and $\mathsf{0}_{\mathcal{A}} = 0$, for all $x, y \in \mathbb{N}$. Note that the interpretations of $-$, $\div$, $-^{\sharp}$ and $\div^{\sharp}$ are monotone with respect to $\geqslant_{\mathbb{N}}$ but not with respect to $>_{\mathbb{N}}$. Hence they cannot be used in a direct termination proof. According to the preceding lemma $(>_{\mathcal{A}}, \geqslant_{\mathcal{A}})$ is a reduction pair and one easily verifies that the constraints of Example 4.5.10 are satisfied. For instance, the constraint $\mathsf{s}(x) \div^{\sharp} \mathsf{s}(y) >_{\mathcal{A}} x -^{\sharp} y$ reduces to $x + 1 >_{\mathbb{N}} x$, which is obviously true for all $x \in \mathbb{N}$. Theorem 4.5.9 yields the termination of the TRS of Table 4.1.

Unlike (weakly monotone) polynomial interpretations, LPO is unable to solve the constraints of Example 4.5.10, even though $(>_{\mathsf{lpo}}, >^{=}_{\mathsf{lpo}})$ is a reduction pair for any well-founded precedence $>$. Because $\mathsf{s}(x) \div^{\sharp} \mathsf{s}(\mathsf{s}(x))$ is embedded in $(x - \mathsf{s}(x)) \div^{\sharp} \mathsf{s}(\mathsf{s}(x))$, no simplification order orients the dependency pair $\mathsf{s}(x) \div^{\sharp} \mathsf{s}(y) \to (x - y) \div^{\sharp} \mathsf{s}(y)$ from left to right.

Below we introduce a simple but very useful transformation for simplifying the constraints generated by the dependency pair method. This transformation replaces function symbols by one of their arguments or eliminates certain arguments of function symbols. The resulting constraints are often satisfiable by LPO or KBO.

Definition 4.5.16. An *argument filter* (*AF* for short) for a signature $\mathcal{F}$ is a mapping π that associates with every n-ary function symbol an argument position $i \in \{1,\ldots,n\}$ or a (possibly empty) list $[i_1,\ldots,i_m]$ of argument positions with $1 \leqslant i_1 < \cdots < i_m \leqslant n$. The signature $\mathcal{F}_\pi$ consists of all function symbols f such that $\pi(f)$ is some list $[i_1,\ldots,i_m]$, where in $\mathcal{F}_\pi$ the arity of f is m. Every argument filter π induces a mapping from $\mathcal{T}(\mathcal{F},\mathcal{V})$ to $\mathcal{T}(\mathcal{F}_\pi,\mathcal{V})$, also denoted by π:

$$\pi(t) = \begin{cases} t & \text{if } t \text{ is a variable} \\ \pi(t_i) & \text{if } t = f(t_1,\ldots,t_n) \text{ and } \pi(f) = i \\ f(\pi(t_{i_1}),\ldots,\pi(t_{i_m})) & \text{if } t = f(t_1,\ldots,t_n) \text{ and } \pi(f) = [i_1,\ldots,i_m] \end{cases}$$

Given a binary relation R on $\mathcal{T}(\mathcal{F}_\pi,\mathcal{V})$, we write $s\ R^\pi\ t$ if and only if $\pi(s)\ R\ \pi(t)$.

Lemma 4.5.17. *If $(>,\gtrsim)$ is a reduction pair and π an AF then $(>^\pi,\gtrsim^\pi)$ is a reduction pair.*

Proof The relation $>^\pi$ inherits irreflexivity, transitivity, and well-foundedness from $>$. The relation $\gtrsim^\pi$ inherits reflexivity and transitivity from $\gtrsim$. It is easy to prove that for all terms t and substitutions σ we have $\pi(t\sigma) = \pi(t)\sigma_\pi$ where σ_π denotes the substitution $\{x \mapsto \pi(\sigma(x)) \mid x \in \mathcal{D}\mathsf{om}(\sigma)\}$. It follows that both $>^\pi$ and $\gtrsim^\pi$ are closed under substitutions. Next we show that $\gtrsim^\pi$ is closed under contexts. Suppose $s \gtrsim^\pi t$ and let C be a context. If $\pi(C)$ does not contain a hole then $\pi(C[s]) = \pi(C) = \pi(C[t])$. Otherwise $\pi(C)$ contains exactly one hole, and we obtain $\pi(C[s]) = \pi(C)[\pi(s)] \gtrsim \pi(C)[\pi(t)] = \pi(C[t])$ from $\pi(s) \gtrsim \pi(t)$ and the closure under contexts of $\gtrsim$. It remains to show $>^\pi \cdot \gtrsim^\pi \subseteq >^\pi$ or $\gtrsim^\pi \cdot >^\pi \subseteq >^\pi$. This follows from the definitions of $>^\pi$ and $\gtrsim^\pi$ and the assumption that $> \cdot \gtrsim \subseteq >$ or $\gtrsim \cdot > \subseteq >$. □

Example 4.5.18. If we take the AF π with $\pi(-) = \pi(\div) = \pi(-^\sharp) = \pi(\div^\sharp) = 1$, $\pi(\mathsf{0}) = [\,]$, and $\pi(\mathsf{s}) = [1]$, the constraints of Example 4.5.10 simplify to

$$\begin{array}{rcl} \mathsf{0} \gtrsim \mathsf{0} & \mathsf{0} \gtrsim \mathsf{0} & \mathsf{s}(x) > x \\ x \gtrsim x & \mathsf{s}(x) \gtrsim \mathsf{s}(x) & \mathsf{s}(x) > x \\ \mathsf{s}(x) \gtrsim x & & \mathsf{s}(x) > x \end{array}$$

and are trivially satisfied by LPO (with empty precedence), i.e., we take $> = \sqsupset_{\mathsf{lpo}}$ and $\gtrsim = \sqsupset^{=}_{\mathsf{lpo}}$ with $\sqsupset = \varnothing$.

Exercises

4.46 Prove that the TRS $\mathcal{R}$ of Table 4.1 is not simply terminating.

4.47 Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\mathsf{f}(\mathsf{0}) \to \mathsf{s}(\mathsf{0}) \qquad \mathsf{f}(\mathsf{s}(\mathsf{0})) \to \mathsf{s}(\mathsf{0}) \qquad \mathsf{f}(\mathsf{s}(\mathsf{s}(x))) \to \mathsf{f}(\mathsf{f}(\mathsf{s}(x)))$$

a Compute the dependency pairs of $\mathcal{R}$.

b Prove the termination of $\mathcal{R}$ by constructing a suitable reduction pair based on a weakly monotone interpretation in $\mathbb{N}$.

c Is $\mathcal{R}$ simply terminating?

4.48 Let $(\mathcal{A}, >)$ be a well-founded monotone algebra. Show that $(\mathcal{A}, >, \geqslant)$ is a well-founded weakly monotone algebra.

4.49 Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{array}{rcr} 0 \leq y \to \mathsf{true} & x - 0 \to x & \mathsf{gcd}(0, y) \to y \\ \mathsf{s}(x) \leq 0 \to \mathsf{false} & x - \mathsf{s}(y) \to \mathsf{p}(x - y) & \mathsf{gcd}(\mathsf{s}(x), 0) \to \mathsf{s}(x) \\ \mathsf{s}(x) \leq \mathsf{s}(y) \to x \leq y & \mathsf{p}(\mathsf{s}(x)) \to x & \mathsf{gcd}(\mathsf{s}(x), \mathsf{s}(y)) \to \mathsf{if}(y \leq x, \mathsf{s}(x), \mathsf{s}(y)) \\ \mathsf{if}(\mathsf{true}, \mathsf{s}(x), \mathsf{s}(y)) \to \mathsf{gcd}(x - y, \mathsf{s}(y)) & & \mathsf{if}(\mathsf{false}, \mathsf{s}(x), \mathsf{s}(y)) \to \mathsf{gcd}(y - x, \mathsf{s}(x)) \end{array}$$

a Compute the dependency pairs of $\mathcal{R}$.

b How many different AFs does $\mathcal{R} \cup \mathsf{DP}(\mathcal{R})$ admit?

c Prove the termination of $\mathcal{R}$.

d Is $\mathcal{R}$ simply terminating?

4.50 Construct a reduction pair $(>, \gtrsim)$ which does not satisfy the inclusion $\gtrsim \cdot > \cdot \gtrsim \subseteq >$.

4.51 Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{array}{rcl} \mathsf{append}(\mathsf{nil}, z) \to z & & \mathsf{append}(x : y, z) \to x : \mathsf{append}(y, z) \\ \mathsf{reverse}(\mathsf{nil}) \to \mathsf{nil} & & \mathsf{reverse}(x : y) \to \mathsf{append}(\mathsf{reverse}(y), x : \mathsf{nil}) \\ \mathsf{shuffle}(\mathsf{nil}) \to \mathsf{nil} & & \mathsf{shuffle}(x : y) \to x : \mathsf{shuffle}(\mathsf{reverse}(y)) \end{array}$$

a Prove the termination of $\mathcal{R}$.

b Is $\mathcal{R}$ simply terminating?

4.52 Discuss the usefulness of AFs for reduction pairs obtained from well-founded weakly monotone algebras.

4.53 Prove the termination of the TRS obtained by omitting the rule $\mathsf{from}(x) \to x : \mathsf{from}(\mathsf{s}(x))$ from the TRS of Table 3.5.

Bibliographic Notes

Termination is one of the most studied topics in term rewriting. An influential early survey is [28]. A more recent survey is [142]. Polynomial interpretations were first used by Lankford [83]. Zantema [141] popularized the monotone algebra approach for proving termination. Exercise 4.15 is based on [96]. Simplification orders were introduced by Dershowitz [25]. Middeldorp and Gramlich [93] proved that simple termination is undecidable (Exercise 4.26), even for TRSs consisting of a single rule. The condition in Theorem 4.2.13 goes back to Touzet [129]. The lexicographic path order is due to Kamin and Lévy [65]. It is preceeded by the multiset path order (Exercise 4.32) of Dershowitz [27]. Several similar orders were developed around the same time. We refer to Rusinowitch [116] for a comparison. The Knuth–Bendix order was invented by Knuth and Bendix [77]. Theorem 4.4.16 [2], the decidability of KBO orientability, is due to Dick, Kalmus and Martin [35]. Korovin and Voronkov [81] proved that KBO orientability can be decided in polynomial time. The decision problem of Theorem 4.4.16 [1] has linear complexity [84]. Exercise 4.39(c) is from [140]. Exercise 4.41 is based on [138]. The extension of KBO with subterm coefficients (Exercise 4.44) stems from Ludwig and Waldmann [88]. Many more simplification orders are put in a historical perspective by Steinbach [122]. Yamada, Kusakari and Sakabe [139] introduced the weighted path order. Dependency pairs were introduced by Arts and Giesl [3]. The condition $u \not\trianglelefteq \ell$ in Definition 4.5.3 is from Dershowitz [30]. Further references on termination can be found in the bibliographic notes of Chapter 9 and Chapter 10.

Chapter 5

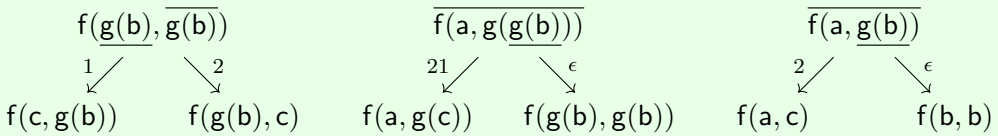
Completion

One of the most important applications of term rewriting is deciding the validity problem in equational theories. We discuss completion, a general method to transform equational systems into equivalent complete rewrite systems. In Section 5.1 we introduce critical pairs, which provide a characterization of local confluence. In combination with Newman's Lemma, we obtain a completion method for terminating rewrite systems which is explained in Section 5.2. In Section 5.3 the efficiency of the completion method is improved by incorporating simplification. In Section 5.4 we present an abstract inference system which captures the essence of completion. Finally, in Section 5.5 we discuss the limitations of completion for deciding the validity problem in equational theories.

5.1 Critical Pairs

How do we prove that a given TRS $\mathcal{R}$ is confluent? If $\mathcal{R}$ is known to be terminating then, according to Newman's Lemma, it suffices to prove local confluence. For showing local confluence, we have to consider all possible peaks $t_1 \leftarrow s \rightarrow t_2$. In general there are infinitely many peaks, even if $\mathcal{R}$ is finite. However, we will see that it suffices to restrict our attention to a kind of 'critical' peaks, of which there are only finitely many in case $\mathcal{R}$ is finite. Basically there are three types of peaks.

Example 5.1.1. Let $\mathcal{R}$ be a TRS containing the rewrite rules $\mathsf{f}(\mathsf{a}, \mathsf{g}(x)) \rightarrow \mathsf{f}(x, x)$ and $\mathsf{g}(\mathsf{b}) \rightarrow \mathsf{c}$ and consider the peaks



The first peak in Example 5.1.1 consists of redex contractions at parallel positions, which is harmless:

$$\mathsf{f}(\mathsf{c}, \underline{\mathsf{g}(\mathsf{b})}) \rightarrow \mathsf{f}(\mathsf{c}, \mathsf{c}) \leftarrow \mathsf{f}(\underline{\mathsf{g}(\mathsf{b})}, \mathsf{c})$$

In both the second and third peak one of the contracted redexes contains the other contracted redex. There is however an important difference between the two peaks. In the second peak contraction of the smaller redex $\mathsf{g}(\mathsf{b})$ does not affect the bigger redex

$\mathsf{f}(\mathsf{a},\mathsf{g}(\mathsf{g}(\mathsf{b})))$ in an essential way: the reduct $\mathsf{f}(\mathsf{a},\mathsf{g}(\mathsf{c}))$ is still a redex with respect to the rewrite rule $\mathsf{f}(\mathsf{a},\mathsf{g}(x)) \to \mathsf{f}(x,x)$. This is not so for the third peak: contraction of the smaller redex $\mathsf{g}(\mathsf{b})$ destroys the bigger redex $\mathsf{f}(\mathsf{a},\mathsf{g}(\mathsf{b}))$. Observe that joinability of the two reducts $\mathsf{f}(\mathsf{a},\mathsf{c})$ and $\mathsf{f}(\mathsf{b},\mathsf{b})$ in the third peak depends on the presence of other rewrite rules, whereas the two reducts $\mathsf{f}(\mathsf{a},\mathsf{g}(\mathsf{c}))$ and $\mathsf{f}(\mathsf{g}(\mathsf{b}),\mathsf{g}(\mathsf{b}))$ in the second peak can be joined with respect to the given rewrite rules:

$$\underline{\mathsf{f}(\mathsf{a},\mathsf{g}(\mathsf{c}))} \to \mathsf{f}(\mathsf{c},\mathsf{c}) \;{}^*\!\leftarrow \mathsf{f}(\underline{\mathsf{g}(\mathsf{b})},\underline{\mathsf{g}(\mathsf{b})})$$

We say that the two redexes in the term $\mathsf{f}(\mathsf{a},\mathsf{g}(\mathsf{b}))$ are *overlapping*. We now give a formal description of this intuitively clear concept.

Definition 5.1.2. An *overlap* of a TRS $(\mathcal{F},\mathcal{R})$ is a triple $\langle \ell_1 \to r_1, p, \ell_2 \to r_2\rangle$ satisfying the following properties:

1. $\ell_1 \to r_1$ and $\ell_2 \to r_2$ are variants of rewrite rules of $\mathcal{R}$ without common variables,
2. $p \in \mathcal{P}\mathsf{os}_{\mathcal{F}}(\ell_2)$,
3. ℓ_1 and $\ell_2|_p$ are unifiable,
4. if $p = \epsilon$ then $\ell_1 \to r_1$ and $\ell_2 \to r_2$ are not variants.

Example 5.1.3. In Example 5.1.1 we have the overlap $\langle \mathsf{g}(\mathsf{b}) \to \mathsf{c}, 2, \mathsf{f}(\mathsf{a},\mathsf{g}(x)) \to \mathsf{f}(x,x)\rangle$. The non-left-linear TRS consisting of the two rewrite rules

$$\mathsf{f}(x,x) \to \mathsf{g}(x,x) \qquad\qquad \mathsf{h}(\mathsf{a}) \to \mathsf{b}$$

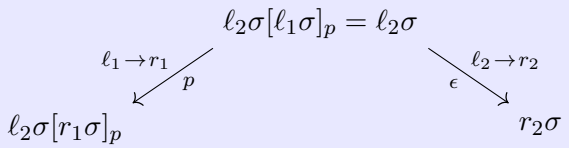
does not have overlaps. Consider the peak

$$\mathsf{f}(\mathsf{b},\mathsf{h}(\mathsf{a})) \leftarrow \overline{\mathsf{f}(\underline{\mathsf{h}(\mathsf{a})},\mathsf{h}(\mathsf{a}))} \to \mathsf{g}(\mathsf{h}(\mathsf{a}),\mathsf{h}(\mathsf{a}))$$

Contraction of the smaller redex $\mathsf{h}(\mathsf{a})$ at position 1 destroys the bigger redex $\mathsf{f}(\mathsf{h}(\mathsf{a}),\mathsf{h}(\mathsf{a}))$ as $\mathsf{f}(\mathsf{b},\mathsf{h}(\mathsf{a}))$ is no longer a redex. This is not really a problem since we can turn the reduct $\mathsf{f}(\mathsf{b},\mathsf{h}(\mathsf{a}))$ into a redex by contracting the redex $\mathsf{h}(\mathsf{a})$ at position 2:

$$\mathsf{f}(\mathsf{b},\underline{\mathsf{h}(\mathsf{a})}) \to \underline{\mathsf{f}(\mathsf{b},\mathsf{b})} \to \mathsf{g}(\mathsf{b},\mathsf{b}) \;{}^*\!\leftarrow \mathsf{g}(\underline{\mathsf{h}(\mathsf{a})},\underline{\mathsf{h}(\mathsf{a})})$$

Definition 5.1.4. Suppose $\langle \ell_1 \to r_1, p, \ell_2 \to r_2\rangle$ is an overlap of a TRS $\mathcal{R}$. Let σ be a most general unifier of ℓ_1 and $\ell_2|_p$. The term $\ell_2\sigma[\ell_1\sigma]_p = \ell_2\sigma$ can be rewritten in two different ways:



We call the quadruple $(\ell_2\sigma[r_1\sigma]_p, p, \ell_2\sigma, r_2\sigma)$ a *critical peak* and the equation $\ell_2\sigma[r_1\sigma]_p \approx r_2\sigma$ a *critical pair* of $\mathcal{R}$, obtained from the overlap $\langle \ell_1 \to r_1, p, \ell_2 \to r_2\rangle$.

A critical peak (t, p, s, u) is usually denoted by $t \overset{p}{\leftarrow} s \overset{\epsilon}{\rightarrow} u$.

Example 5.1.5. The overlap $\langle \mathsf{g(b)} \to \mathsf{c}, 2, \mathsf{f(a, g(}x\mathsf{))} \to \mathsf{f}(x, x)\rangle$ gives rise to the critical peak $\mathsf{f(a, c)} \overset{2}{\leftarrow} \mathsf{f(a, g(b))} \overset{\epsilon}{\rightarrow} \mathsf{f(b, b)}$ and the critical pair $\mathsf{f(a, c)} \approx \mathsf{f(b, b)}$.

Definition 5.1.6. The set of all critical pairs of $\mathcal{R}$ is denoted by $\mathsf{CP}(\mathcal{R})$. Furthermore, if $\mathcal{R}_1, \mathcal{R}_2 \subseteq \mathcal{R}$ then $\mathsf{CP}(\mathcal{R}_1, \mathcal{R}_2)$ denotes the set of all critical pairs of $\mathcal{R}$ obtained from overlaps $\langle \rho_1, p, \rho_2\rangle$ with $\rho_1 \in \mathcal{R}_1$ and $\rho_2 \in \mathcal{R}_2$.

For the results presented in this chapter the order of the terms in a critical pair is irrelevant. This is not true for the confluence results presented in Chapter 6. If we want to stress the order, we write $\ell_2\sigma[r_1\sigma]_p \leftarrow\!\!\rightarrow r_2\sigma$ instead of $\ell_2\sigma[r_1\sigma]_p \approx r_2\sigma$.

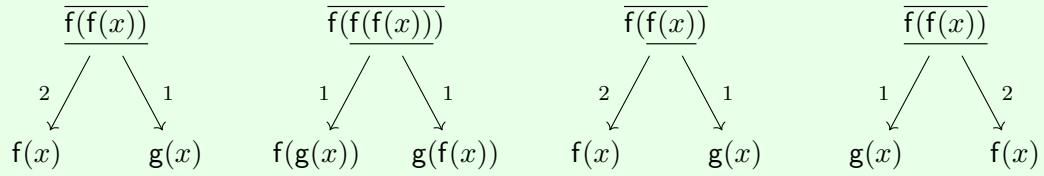
Definition 5.1.7. An overlap $\langle \rho_1, p, \rho_2\rangle$ is a *variant* of an overlap $\langle \rho_1', p, \rho_2'\rangle$ if ρ_1 is a variant of ρ_1' and ρ_2 is a variant of ρ_2'. A critical pair $s \approx t$ is a *variant* of a critical pair $s' \approx t'$ if there exists a renaming σ such that $s = s'\sigma$ and $t = t'\sigma$.

Overlaps that are variants of each other will be identified. Critical pairs obtained from the same overlap are variants (Exercise 5.8) and are also identified. Hence every overlap gives rise to exactly one critical pair and consequently a finite TRS has only finitely many critical pairs (Exercise 5.4).

Example 5.1.8. Consider the TRS consisting of the following two rewrite rules:

$$\mathsf{f(f(}x\mathsf{))} \overset{1}{\longrightarrow} \mathsf{g}(x) \qquad\qquad \mathsf{f}(x) \overset{2}{\longrightarrow} x$$

There are four different overlaps: $\langle 2, \epsilon, 1\rangle$, $\langle 1, 1, 1\rangle$, $\langle 2, 1, 1\rangle$, and $\langle 1, \epsilon, 2\rangle$, and thus four critical pairs:



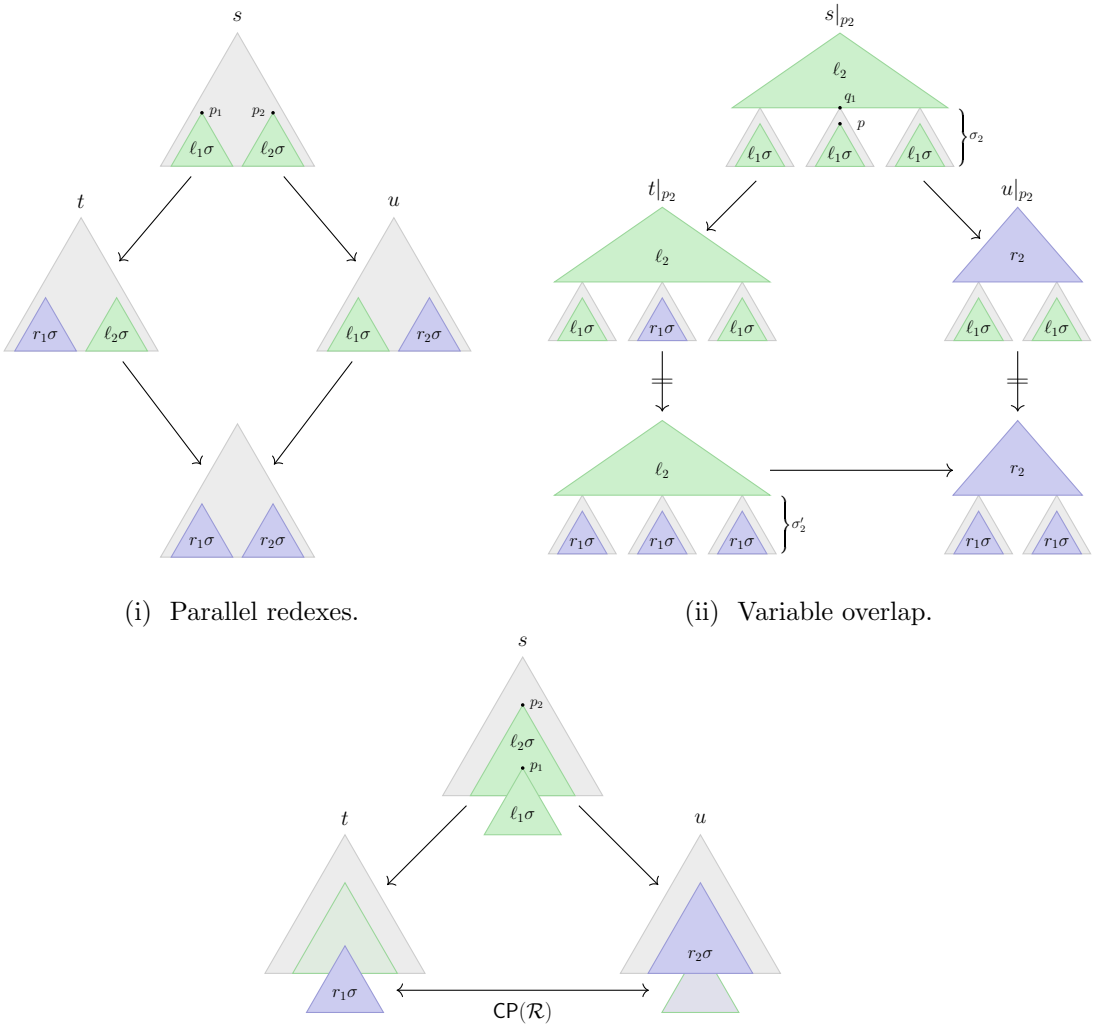
The critical pairs obtained from the overlaps $\langle 2, \epsilon, 1\rangle$ and $\langle 2, 1, 1\rangle$ are identical.

Definition 5.1.9. A critical pair $s \approx t$ of a TRS is called *joinable* if $s \downarrow t$.

We are now ready for the main result of this section. The different cases in the proof are illustrated in Figure 5.1.

Lemma 5.1.10. *Let $\mathcal{R}$ be a* TRS. *If $t \,{}_{\mathcal{R}}\!\leftarrow \cdot \rightarrow_{\mathcal{R}} u$ then $t \downarrow_{\mathcal{R}} u$ or $t \leftrightarrow_{\mathsf{CP}(\mathcal{R})} u$.*

Proof Consider an arbitrary peak $t \leftarrow_{p_1|\ell_1\to r_1|\sigma_1} s \rightarrow_{p_2|\ell_2\to r_2|\sigma_2} u$. We may assume that the rewrite rules $\ell_1 \to r_1$ and $\ell_2 \to r_2$ have no variables in common, and consequently



(i) Parallel redexes.

(ii) Variable overlap.

(iii) Critical overlap.

Figure 5.1: Different cases in the proof of the Lemma 5.1.10.

$\mathcal{D}\mathsf{om}(\sigma_1)\cap\mathcal{D}\mathsf{om}(\sigma_2)=\varnothing$. If $p_1 \parallel p_2$ then $t \to_{p_2\mid \ell_2\to r_2\mid\sigma_2} t[r_2\sigma_2]_{p_2} = u[r_1\sigma_1]_{p_1} \leftarrow_{p_1\mid\ell_1\to r_1\mid\sigma_1} u$. This easy case is illustrated in Figure 5.1(i). If the positions of the contracted redexes are not parallel then one of them is above the other. Without loss of generality we assume $p_1 \geqslant p_2$. Let $p = p_1 \setminus p_2$. We have $t = s[r_1\sigma_1]_{p_1} = s[\ell_2\sigma_2[r_1\sigma_1]_p]_{p_2}$ and $u = s[r_2\sigma_2]_{p_2}$. We consider two cases depending on whether the triple $\langle \ell_1 \to r_1, p, \ell_2 \to r_2\rangle$ is an overlap.

1 Suppose $\langle \ell_1 \to r_1, p, \ell_2 \to r_2\rangle$ is an overlap. Let σ be a most general unifier of $\ell_2|_p$ and ℓ_1. We have $\ell_2\sigma[r_1\sigma]_p \leftrightarrow\!\!\!\!\rightarrow r_2\sigma \in \mathsf{CP}(\mathcal{R})$. Let $\sigma' = \sigma_1 \cup \sigma_2$. The substitution σ' is a unifier of $\ell_2|_p$ and ℓ_1: $(\ell_2|_p)\sigma' = (\ell_2\sigma_2)|_p = \ell_1\sigma_1 = \ell_1\sigma'$. Hence there exists a substitution τ such that $\sigma' = \sigma\tau$. Therefore

$$\ell_2\sigma_2[r_1\sigma_1]_p \;=\; (\ell_2\sigma[r_1\sigma]_p)\tau \;\leftrightarrow_{\mathsf{CP}(\mathcal{R})}\; (r_2\sigma)\tau \;=\; r_2\sigma_2$$

and thus $t \leftrightarrow_{\mathsf{CP}(\mathcal{R})} u$. (This case is illustrated in Figure 5.1(iii).)

2 If $\langle \ell_1 \to r_1, p, \ell_2 \to r_2\rangle$ is not an overlap then either $p = \epsilon$ and $\ell_1 \to r_1$, $\ell_2 \to r_2$ are

variants, or $p \notin \mathcal{P}\mathsf{os}_{\mathcal{F}}(\ell_2)$. In the former case it is easy to show $r_1\sigma_1 = r_2\sigma_2$ and hence $t = u$. In the latter case, which is illustrated in Figure 5.1(ii), there exist positions q_1, q_2 such that $p = q_1q_2$ and $q_1 \in \mathcal{P}\mathsf{os}_{\mathcal{V}}(\ell_2)$. Let $\ell_2|_{q_1}$ be the variable x. We have $\sigma_2(x)|_{q_2} = \ell_1\sigma_1$. Define the substitution σ_2' as follows:

$$\sigma_2'(y) = \begin{cases} \sigma_2(y)[r_1\sigma_1]_{q_2} & \text{if } y = x \\ \sigma_2(y) & \text{if } y \neq x \end{cases}$$

Clearly $\sigma_2(x) \to_{q_2|\ell_1 \to r_1|\sigma_1} \sigma_2'(x)$. Hence $r_2\sigma_2 \to^* r_2\sigma_2'$. We also have

$$\ell_2\sigma_2[r_1\sigma_1]_p = \ell_2\sigma_2[\sigma_2'(x)]_{q_1} \to^* \ell_2\sigma_2' \to_{\epsilon|\ell_2 \to r_2|\sigma_2'} r_2\sigma_2'$$

Note that $\ell_2\sigma_2[r_1\sigma_1]_p = \ell_2\sigma_2'$ when the rule $\ell_2 \to r_2$ is left-linear. Consequently, $t \to^* s[r_2\sigma_2']_{p_2}$ ${}^*\!\leftarrow u$. □

Critical Pair Lemma. *A* TRS *is locally confluent if and only if all its critical pairs are joinable.*

Proof Let $\mathcal{R}$ be a TRS. If all critical pairs of $\mathcal{R}$ are joinable then $\leftrightarrow_{\mathsf{CP}(\mathcal{R})} \subseteq \downarrow_{\mathcal{R}}$ and thus ${}_{\mathcal{R}}\!\leftarrow \cdot \to_{\mathcal{R}} \subseteq \downarrow_{\mathcal{R}}$ by Lemma 5.1.10. Hence $\mathcal{R}$ is locally confluent. Conversely, if $\mathcal{R}$ is locally confluent then clearly all its critical pairs are joinable since every critical pair $s \approx t$ originates from a peak $s \leftarrow \cdot \to t$. □

Combining the Critical Pair Lemma with Newman's Lemma yields the following result.

Corollary 5.1.11. *A terminating* TRS *is confluent if and only if all its critical pairs are joinable.*

This result forms the theoretical basis for the completion procedure of Knuth and Bendix, to be presented in the next section.

Corollary 5.1.12. *Confluence is a decidable property of finite terminating* TRS*s.*

Proof Let $\mathcal{R}$ be a finite terminating TRS. Because $\mathcal{R}$ has finitely many rewrite rules, there are only a finite number of critical pairs (Exercise 5.4). Moreover, for finite terminating TRSs it is decidable whether $s \downarrow t$, for all terms s and t (cf. Exercise 3.28(b)). Hence we can decide whether every critical pair of $\mathcal{R}$ is joinable. According to Corollary 5.1.11 this decides confluence. □

We conclude this section with the observation that not all critical pairs have to be computed in order to infer confluence.

Definition 5.1.13. A critical peak $t \overset{p}{\leftarrow} s \overset{\epsilon}{\to} u$ is *prime* if all proper subterms of $s|_p$ are in normal form. A critical pair is called *prime* if it is derived from a prime critical peak. We write $\mathsf{PCP}(\mathcal{R})$ to denote the set of all prime critical pairs of a TRS $\mathcal{R}$.

Example 5.1.14. Consider the TRS of Table 3.2. The critical pair $y\cdot\mathsf{e} \approx y^{--}$ originating from the overlap $\langle x \cdot x^- \to \mathsf{e}, 2, y \cdot (y^- \cdot z) \to z\rangle$ with critical peak

$$y \cdot \mathsf{e} \overset{2}{\leftarrow} y \cdot (y^- \cdot y^{--}) \overset{\epsilon}{\to} y^{--}$$

is non-prime since the term $y \cdot (y^- \cdot y^{--})$ is reducible at position $22 > 2$. Also the critical pair $\mathsf{e}^- \approx \mathsf{e}$ originating from $\langle x \cdot \mathsf{e} \to x, \epsilon, y^- \cdot y \to \mathsf{e}\rangle$ with critical peak $\mathsf{e}^- \overset{\epsilon}{\leftarrow} \mathsf{e}^- \cdot \mathsf{e} \overset{\epsilon}{\to} \mathsf{e}$ is not prime. The critical pair $\mathsf{e} \cdot \mathsf{e} \approx \mathsf{e}$ originating from the overlap $\langle \mathsf{e}^- \to \mathsf{e}, 1, x^- \cdot x \to \mathsf{e}\rangle$ is prime.

Definition 5.1.15. Given a TRS $\mathcal{R}$ and terms s, t, and u, we write $t \bigtriangledown_s u$ if $s \to_{\mathcal{R}}^+ t$, $s \to_{\mathcal{R}}^+ u$, and $t \downarrow_{\mathcal{R}} u$ or $t \leftrightarrow_{\mathsf{PCP}(\mathcal{R})} u$.

Lemma 5.1.16. *Let $\mathcal{R}$ be a* TRS*. If $t \overset{p}{\leftarrow} s \overset{\epsilon}{\to} u$ is a critical peak then $t \bigtriangledown_s^2 u$.*

Proof First suppose that all proper subterms of $s|_p$ are in normal form. Then $t \approx u \in \mathsf{PCP}(\mathcal{R})$ and thus $t \bigtriangledown_s u$. Since also $u \bigtriangledown_s u$, we obtain the desired $t \bigtriangledown_s^2 u$. This leaves us with the case that there is a proper subterm of $s|_p$ not in normal form. By considering an innermost redex in $s|_p$ we obtain a position $q > p$ and a term v such that $s \overset{q}{\to} v$ and all proper subterms of $s|_q$ are in normal form. Now, if $v \overset{q}{\leftarrow} s \overset{\epsilon}{\to} u$ is an instance of a critical peak then $v \to_{\mathsf{PCP}(\mathcal{R})} u$. Otherwise, $v \downarrow_{\mathcal{R}} u$ by Lemma 5.1.10. In both cases we obtain $v \bigtriangledown_s u$. Finally, we analyze the peak $t \overset{p}{\leftarrow} s \overset{q}{\to} v$ by another application of Lemma 5.1.10.

1. If $t \downarrow_{\mathcal{R}} v$, we obtain $t \bigtriangledown_s v$ and thus $t \bigtriangledown_s^2 u$, since also $v \bigtriangledown_s u$.
2. Since $p < q$, in the remaining case $v|_p \overset{q\backslash p}{\longleftarrow} s|_p \overset{\epsilon}{\to} t|_p$ is an instance of a critical peak. All proper subterms of $s|_q$ are in normal form and thus we have an instance of a prime critical peak. Hence $t \leftarrow_{\mathsf{PCP}(\mathcal{R})} v$ and together with $v \bigtriangledown_s u$ we conclude $t \bigtriangledown_s^2 u$. □

Corollary 5.1.17. *A terminating* TRS *is confluent if and only if all its prime critical pairs are joinable.*

Proof Let $\mathcal{R}$ be a terminating TRS such that $\mathsf{PCP}(\mathcal{R}) \subseteq \downarrow_{\mathcal{R}}$. We label rewrite steps by their starting term and we claim that $\mathcal{R}$ is peak decreasing. As well-founded order we take $> = \to_{\mathcal{R}}^+$. Consider an arbitrary peak $t \,{}_{\mathcal{R}}\!\leftarrow s \to_{\mathcal{R}} u$. Lemma 5.1.16 yields a term v such that $t \bigtriangledown_s v \bigtriangledown_s u$. From the assumption $\mathsf{PCP}(\mathcal{R}) \subseteq \downarrow_{\mathcal{R}}$ we obtain $t \downarrow_{\mathcal{R}} v \downarrow_{\mathcal{R}} u$. Since $s \to_{\mathcal{R}}^+ v$, all steps in the conversion $t \downarrow_{\mathcal{R}} v \downarrow_{\mathcal{R}} u$ are labeled with a term that is smaller than s. Since the two steps in the peak receive the same label s, peak decreasingness is established and hence we obtain the confluence of $\mathcal{R}$ from Lemma 1.4.9. The reverse direction is trivial. □

Exercises

5.1 Consider the TRS $\mathcal{R}$ of Exercise 3.5.

a Compute all overlaps of $\mathcal{R}$.

b Compute all critical peaks and critical pairs of $\mathcal{R}$.

5.2 Compute the critical pairs of the SRS consisting of the rewrite rules

$$\mathsf{TCAT} \to \mathsf{T} \qquad \mathsf{GAG} \to \mathsf{AG} \qquad \mathsf{CTC} \to \mathsf{TC} \qquad \mathsf{AGTA} \to \mathsf{A} \qquad \mathsf{TAT} \to \mathsf{CT}$$

5.3 Consider the TRS $\mathcal{R}$ consisting of the rewrite rules $\mathsf{f}(x, \mathsf{g}(y)) \to x$ and $\mathsf{g}(\mathsf{g}(x)) \to \mathsf{f}(x, x)$.

a Compute all critical pairs of $\mathcal{R}$.

b Is $\mathcal{R}$ locally confluent?

5.4 Prove that a TRS with finitely many rewrite rules has a finite number of critical pairs.

5.5 Consider the TRS $\mathcal{R}$ of Table 3.2.

a Compute all critical pairs of $\mathcal{R}$.

b Which critical pairs of $\mathcal{R}$ are prime?

5.6 Does the Critical Pair Lemma remain true if we only consider prime critical pairs?

5.7 Is the TRS consisting of the rewrite rules

$$\begin{array}{lll} 0 + x \to x & \mathsf{gcd}(x, 0) \to x & \mathsf{gcd}(x + y, x) \to \mathsf{gcd}(x, y) \\ \mathsf{s}(x) + y \to \mathsf{s}(x + y) & \mathsf{gcd}(0, x) \to x & \mathsf{gcd}(x, x + y) \to \mathsf{gcd}(x, y) \end{array}$$

confluent?

5.8 ***a*** Suppose the overlaps $\langle \ell_1 \to r_1, p, \ell_2 \to r_2 \rangle$ and $\langle \ell_1' \to r_1', p, \ell_2' \to r_2' \rangle$ are variants. Let σ be a most general unifier of ℓ_1 and $\ell_2|_p$, and σ' a most general unifier of ℓ_1' and $\ell_2'|_p$. Show that the corresponding critical pairs $\ell_2\sigma[r_1\sigma]_p \approx r_2\sigma$ and $\ell_2'\sigma'[r_1'\sigma']_p \approx r_2'\sigma'$ are variants.

b Show that joinability of critical pairs is variant independent, i.e., if the critical pairs $s \approx t$ and $s' \approx t'$ are variants then $s \approx t$ is joinable if and only if $s' \approx t'$ is joinable.

5.9 Prove that the TRS consisting of the rewrite rules

$$\begin{array}{lll} 0 + x \to x & \mathsf{bin}(x, 0) \to \mathsf{s}(0) & \mathsf{bin}(\mathsf{s}(x), \mathsf{s}(y)) \to \mathsf{bin}(x, \mathsf{s}(y)) + \mathsf{bin}(x, y) \\ \mathsf{s}(x) + y \to \mathsf{s}(x + y) & \mathsf{bin}(0, \mathsf{s}(x)) \to 0 & \end{array}$$

for computing binomial numbers is complete.

5.10 A critical pair originating from an overlap $\langle \ell_1 \to r_1, p, \ell_2 \to r_2 \rangle$ with mgu σ is *blocked* if $x\sigma$ is a normal form for every $x \in \mathcal{V}\mathsf{ar}(\ell_1) \cup \mathcal{V}\mathsf{ar}(\ell_2)$.

a Prove that every prime critical pair is blocked.

b Determine which critical pairs of the TRS of Table 3.2 are blocked.

5.11 Let $\mathcal{R}$ be the TRS of Exercise 4.31 and let $\mathcal{S}$ consist of the rewrite rules

$$\mathsf{p}(\mathsf{s}(x)) \to x \qquad\qquad \mathsf{s}(\mathsf{p}(x)) \to x$$

Compute $\mathsf{CP}(\mathcal{R}, \mathcal{S})$ and $\mathsf{CP}(\mathcal{S}, \mathcal{R})$.

5.12 ***a*** Prove that every left-reduced ground TRS has random descent.

b Is every reduced ground TRS complete?

c Is every complete ground TRS reduced?

5.2 Elementary Completion

The Knuth–Bendix completion procedure attempts to transform a given ES into a complete TRS defining the same conversion. After orienting the equations of the ES into a

terminating TRS, according to Corollary 5.1.11 it suffices to make all critical pairs joinable. The basic idea underlying the Knuth–Bendix completion procedure is to add a new rewrite rule whenever a non-joinable critical pair is encountered, in order to make it joinable. This has to be repeated until all critical pairs are joinable. Let us illustrate the completion process by means of a simple example. Consider the ES consisting of the following six equations:

$$\begin{array}{lll} x + 0 \approx x & x - 0 \approx x & \mathsf{s}(\mathsf{p}(x)) \approx x \\ x + \mathsf{s}(y) \approx \mathsf{s}(x + y) & x - \mathsf{s}(y) \approx \mathsf{p}(x - y) & \mathsf{p}(\mathsf{s}(x)) \approx x \end{array}$$

We orient the equations, as follows:

$$\begin{array}{lll} x + 0 \xrightarrow{1} x & x - 0 \xrightarrow{3} x & \mathsf{s}(\mathsf{p}(x)) \xrightarrow{5} x \\ x + \mathsf{s}(y) \xrightarrow{2} \mathsf{s}(x + y) & x - \mathsf{s}(y) \xrightarrow{4} \mathsf{p}(x - y) & \mathsf{p}(\mathsf{s}(x)) \xrightarrow{6} x \end{array}$$

There are no critical pairs among the rules 1, 2, 3, 4, and 6. There is also no critical pair between rules 1, 3, and rule 5. Between rules 2, 4, 6, and rule 5 there are four critical pairs:

$$\begin{array}{cccc} \overline{x + \underline{\mathsf{s}(\mathsf{p}(y))}} & \overline{\mathsf{s}(\underline{\mathsf{p}(\mathsf{s}(x))})} & \overline{x - \underline{\mathsf{s}(\mathsf{p}(y))}} & \overline{\mathsf{p}(\underline{\mathsf{s}(\mathsf{p}(x))})} \\ {\scriptstyle 5}\swarrow \quad \searrow{\scriptstyle 2} & {\scriptstyle 6}\swarrow \quad \searrow{\scriptstyle 5} & {\scriptstyle 5}\swarrow \quad \searrow{\scriptstyle 4} & {\scriptstyle 5}\swarrow \quad \searrow{\scriptstyle 6} \\ x + y \quad \mathsf{s}(x + \mathsf{p}(y)) & \mathsf{s}(x) \quad \mathsf{s}(x) & x - y \quad \mathsf{p}(x - \mathsf{p}(y)) & \mathsf{p}(x) \quad \mathsf{p}(x) \end{array}$$

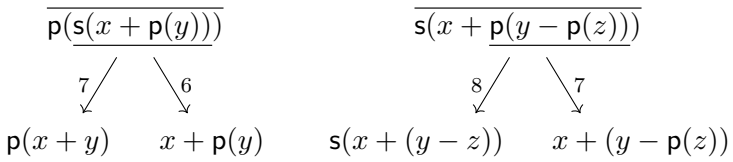
The pairs $\mathsf{s}(x) \approx \mathsf{s}(x)$ and $\mathsf{p}(x) \approx \mathsf{p}(x)$ are trivially joinable. The pairs $x + y \approx \mathsf{s}(x + \mathsf{p}(y))$ and $x - y \approx \mathsf{p}(x - \mathsf{p}(y))$ are not joinable since they consist of different normal forms. The first pair can be made joinable by adding the new rewrite rule $\mathsf{s}(x + \mathsf{p}(y)) \to x + y$. Since $\mathsf{s}(x + \mathsf{p}(y))$ and $x + y$ are convertible with respect to the old rewrite rules, the addition of $\mathsf{s}(x + \mathsf{p}(y)) \to x + y$ does not change conversion. Observe that adding $x + y \to \mathsf{s}(x + \mathsf{p}(y))$ instead of $\mathsf{s}(x + \mathsf{p}(y)) \to x + y$ also makes the critical pair $x + y \approx \mathsf{s}(x + \mathsf{p}(y))$ joinable, but without preserving termination. The pair $x - y \approx \mathsf{p}(x - \mathsf{p}(y))$ is made joinable by adding the rule $\mathsf{p}(x - \mathsf{p}(y)) \to x - y$. So we add the following two rewrite rules:

$$\mathsf{s}(x + \mathsf{p}(y)) \xrightarrow{7} x + y \qquad\qquad \mathsf{p}(x - \mathsf{p}(y)) \xrightarrow{8} x - y$$

At this stage all critical pairs among the first six rules are joinable, but the addition of rules 7 and 8 creates eight new critical pairs:

$$\begin{array}{ccc} \overline{x + \underline{\mathsf{s}(y + \mathsf{p}(z))}} & \overline{\mathsf{s}(\underline{\mathsf{p}(x - \mathsf{p}(y))})} & \overline{\mathsf{p}(x - \underline{\mathsf{p}(\mathsf{s}(y))})} \\ {\scriptstyle 7}\swarrow \quad \searrow{\scriptstyle 2} & {\scriptstyle 8}\swarrow \quad \searrow{\scriptstyle 5} & {\scriptstyle 6}\swarrow \quad \searrow{\scriptstyle 8} \\ x + (y + z) \quad \mathsf{s}(x + (y + \mathsf{p}(z))) & \mathsf{s}(x - y) \quad x - \mathsf{p}(y) & \mathsf{p}(x - y) \quad x - \mathsf{s}(y) \end{array}$$

$$\begin{array}{ccc} \overline{x - \underline{\mathsf{s}(y + \mathsf{p}(z))}} & \overline{\mathsf{s}(x + \underline{\mathsf{p}(\mathsf{s}(y))})} & \overline{\mathsf{p}(x - \underline{\mathsf{p}(y - \mathsf{p}(z))})} \\ {\scriptstyle 7}\swarrow \quad \searrow{\scriptstyle 4} & {\scriptstyle 6}\swarrow \quad \searrow{\scriptstyle 7} & {\scriptstyle 8}\swarrow \quad \searrow{\scriptstyle 8} \\ x - (y + z) \quad \mathsf{p}(x - (y + \mathsf{p}(z))) & \mathsf{s}(x + y) \quad x + \mathsf{s}(y) & \mathsf{p}(x - (y - z)) \quad x - (y - \mathsf{p}(z)) \end{array}$$



The pairs $\mathsf{s}(x+y) \approx x + \mathsf{s}(y)$ and $\mathsf{p}(x-y) \approx x - \mathsf{s}(y)$ are joinable by rules 2 and 4, respectively. The critical pairs $\mathsf{p}(x+y) \approx x + \mathsf{p}(y)$ and $\mathsf{s}(x-y) \approx x - \mathsf{p}(y)$ are made joinable by adding the following rewrite rules:

$$x + \mathsf{p}(y) \xrightarrow{9} \mathsf{p}(x+y) \qquad\qquad x - \mathsf{p}(y) \xrightarrow{10} \mathsf{s}(x-y)$$

The four remaining pairs are now also joinable, e.g.

$$\mathsf{s}(x + (y + \mathsf{p}(z))) \to \mathsf{s}(x + \mathsf{p}(y+z)) \to x + (y+z)$$

by applications of rules 9, 7 and

$$x - (y - \mathsf{p}(z)) \to x - \mathsf{s}(y-z) \to \mathsf{p}(x - (y-z))$$

by applications of rules 10, 4. The rules 9 and 10 give rise to six new critical pairs. These pairs are easily shown to be joinable (Exercise 5.13). Hence we end up with the complete TRS $\mathcal{R}$ of Table 5.1. Termination of $\mathcal{R}$ can be shown by the lexicographic path order with precedence $+ > \mathsf{s}$, $+ > \mathsf{p}$, $- > \mathsf{s}$ and $- > \mathsf{p}$.

Figure 5.2 shows a version of the Knuth–Bendix completion procedure. The procedure takes as input a given ES $\mathcal{E}$. The procedure presupposes a reduction order in order to solve the orientation problem of new rewrite rules in a uniform way.

The procedure of Figure 5.2 has three possibilities: 1 it may terminate successfully, 2 it may fail because a pair of terms cannot be oriented into a rewrite rule, or 3 it may loop infinitely. The proof that upon successful termination $\mathcal{R}$ is a complete presentation of $\mathcal{E}$ is not difficult.

Theorem 5.2.1. *If the procedure of Figure 5.2 terminates successfully then $\mathcal{R}$ is a complete* TRS *that represents $\mathcal{E}$.*

Proof Let $\mathcal{R}_i$ and C_i denote the respective values of $\mathcal{R}$ and C after the i-th iteration of the while loop. Suppose the procedure terminates after n iterations of the while loop. A straightforward induction argument (Exercise 5.15) reveals that for all $0 \leqslant i \leqslant n$

1 $\mathcal{R}_i$ is compatible with $>$

2 $(\leftrightarrow_{\mathcal{R}_i} \cup \leftrightarrow_{C_i})^* = \leftrightarrow_{\mathcal{E}}^*$

$x + 0 \to x$	$x - 0 \to x$
$x + \mathsf{s}(y) \to \mathsf{s}(x+y)$	$x - \mathsf{s}(y) \to \mathsf{p}(x-y)$
$x + \mathsf{p}(y) \to \mathsf{p}(x+y)$	$x - \mathsf{p}(y) \to \mathsf{s}(x-y)$
$\mathsf{s}(x + \mathsf{p}(y)) \to x + y$	$\mathsf{p}(x - \mathsf{p}(y)) \to x - y$
$\mathsf{s}(\mathsf{p}(x)) \to x$	$\mathsf{p}(\mathsf{s}(x)) \to x$

Table 5.1: A complete TRS for addition and subtraction over the integers.

Knuth–Bendix Completion Procedure

Input: ▷ an ES $\mathcal{E}$
▷ a reduction order $>$
Output: ▷ a complete TRS $\mathcal{R}$ that represents $\mathcal{E}$

$\mathcal{R} := \varnothing$;
$C := \mathcal{E}$;
`while` $C \neq \varnothing$ `do`
  select an equation $s \approx t \in C$;
  $C := C \setminus \{s \approx t\}$;
  rewrite s and t to normal forms s' and t' with respect to $\mathcal{R}$;
  `if` $s' \neq t'$ `then`
    `if` $s' > t'$ `then` $\mathcal{S} := \{s' \to t'\}$
    `else if` $t' > s'$ `then` $\mathcal{S} := \{t' \to s'\}$
    `else` *failure*;
    $C := C \cup \mathsf{CP}(\mathcal{R},\mathcal{S}) \cup \mathsf{CP}(\mathcal{S},\mathcal{R}) \cup \mathsf{CP}(\mathcal{S})$;
    $\mathcal{R} := \mathcal{R} \cup \mathcal{S}$

Figure 5.2: A completion procedure.

3 every equation in $\mathsf{CP}(\mathcal{R}_i) \setminus C_i$ is joinable with respect to $\mathcal{R}_i$

Because $C_n = \varnothing$ we have $\leftrightarrow^*_{\mathcal{R}_n} = \leftrightarrow^*_{\mathcal{E}}$ and thus $\mathcal{R} = \mathcal{R}_n$ represents $\mathcal{E}$. Furthermore, all critical pairs of $\mathcal{R}$ are joinable. Termination of $\mathcal{R}$ follows from its compatibility with the reduction order $>$. We conclude that $\mathcal{R}$ is a complete TRS representing $\mathcal{E}$. □

So if the procedure of Figure 5.2 terminates successfully then the validity problem for the ES given as input is solvable.

Example 5.2.2. Consider the ES $\mathcal{E}$ consisting of the two equations

$$\mathsf{f}(x,y) \approx \mathsf{g}(x) \qquad\qquad \mathsf{f}(x,y) \approx \mathsf{h}(y)$$

Any reduction order will orient these equations from left to right and hence we obtain the critical pairs $\mathsf{g}(x) \approx \mathsf{h}(y)$ and $\mathsf{h}(y) \approx \mathsf{g}(x)$. None of these pairs can be oriented into a (terminating) rewrite rule and hence completion fails. Nevertheless, the validity problem for $\mathcal{E}$ is easily decidable.

If the third possibility occurs, we say that the completion procedure *diverges*.

Example 5.2.3. Consider the ES $\mathcal{E}$ consisting of the rewrite rules

$$\mathsf{f}(\mathsf{g}(x)) \approx \mathsf{g}(\mathsf{h}(x)) \qquad\qquad \mathsf{g}(\mathsf{a}) \approx \mathsf{b}$$

As reduction order we use the lexicographic path order with precedence $\mathsf{a} > \mathsf{f} > \mathsf{g} > \mathsf{h} > \mathsf{b}$, which orients the equations from left to right. The single critical pair $\mathsf{f}(\mathsf{b}) \approx \mathsf{g}(\mathsf{h}(\mathsf{a}))$ is

oriented from right to left. At this point our TRS consists of the rewrite rules

$$\mathsf{f}(\mathsf{g}(x)) \to \mathsf{g}(\mathsf{h}(x)) \qquad \mathsf{g}(\mathsf{a}) \to \mathsf{b} \qquad \mathsf{g}(\mathsf{h}(\mathsf{a})) \to \mathsf{f}(\mathsf{b})$$

There is one new critical pair: $\mathsf{f}(\mathsf{f}(\mathsf{b})) \approx \mathsf{g}(\mathsf{h}(\mathsf{h}(\mathsf{a})))$, which is again oriented from right to left. To make a long story short, this process will not terminate. In the limit, we obtain the infinite TRS $\mathcal{R}$ consisting of the rewrite rules $\mathsf{f}(\mathsf{g}(x)) \to \mathsf{g}(\mathsf{h}(x))$ and $\mathsf{g}(\mathsf{h}^i(\mathsf{a})) \to \mathsf{f}^i(\mathsf{b})$ for all $i \geqslant 0$.

Exercises

5.13 Show that the critical pairs which arise after adding rules 9 and 10 in the completion example are joinable.

5.14 Use the procedure of Figure 5.2 to complete the ES consisting of the five equations

$$\mathsf{d}(0, y) \approx -y \qquad \mathsf{d}(\mathsf{s}(x), \mathsf{s}(y)) \approx \mathsf{d}(x, y) \qquad \mathsf{s}(\mathsf{p}(x)) \approx x$$
$$\mathsf{d}(\mathsf{p}(x), \mathsf{p}(y)) \approx \mathsf{d}(x, y) \qquad \mathsf{p}(\mathsf{s}(x)) \approx x$$

using a suitable LPO as reduction order.

5.15 Complete the proof of Theorem 5.2.1.

5.16 ***a*** Show that the infinite TRS $\mathcal{R}$ obtained in Example 5.2.3 is complete.

b Construct a finite complete TRS that represents the ES $\mathcal{E}$ of Example 5.2.3.

5.17 Complete the ES consisting of the equation $(x \cdot y) \cdot (y \cdot z) \approx y$.

5.18 Consider the ES $\mathcal{E}$ consisting of the four equations

$$\mathsf{f}(x, \mathsf{g}(x, y)) \approx y \qquad \mathsf{g}(\mathsf{a}, x) \approx x$$
$$\mathsf{h}(\mathsf{g}(x, y), y) \approx x \qquad \mathsf{g}(x, \mathsf{a}) \approx x$$

a Construct a model for $\mathcal{E}$ in which the equation $\mathsf{f}(x, y) \approx \mathsf{f}(y, x)$ is not valid.

b Construct a complete TRS that represents $\mathcal{E}$.

5.19 Consider the TRS $\mathcal{R}'$

$$\begin{array}{lll} x + 0 \to x & \mathsf{s}(x + \mathsf{p}(y)) \to x + y & \mathsf{p}(\mathsf{s}(x)) \to x \\ x - 0 \to x & \mathsf{p}(x - \mathsf{p}(y)) \to x - y & \mathsf{s}(\mathsf{p}(x)) \to x \\ x + \mathsf{s}(y) \to \mathsf{s}(x + y) & \mathsf{p}(x + y) \to x + \mathsf{p}(y) & \\ x - \mathsf{s}(y) \to \mathsf{p}(x - y) & \mathsf{s}(x - y) \to x - \mathsf{p}(y) & \end{array}$$

obtained from the TRS $\mathcal{R}$ of Table 5.1 by reversing the two rewrite rules $x + \mathsf{p}(y) \to \mathsf{p}(x + y)$ and $x - \mathsf{p}(y) \to \mathsf{s}(x - y)$.

a Prove that $\mathcal{R}'$ is terminating.

b Show that $\mathcal{R}'$ is not confluent.

c Try to complete $\mathcal{R}'$.

5.3 Normalization Equivalence

The completion process described in the previous section can be made more efficient by simplifying rewrite rules and removing redundant rules. For instance, after adding the rule $x + \mathsf{p}(y) \to \mathsf{p}(x + y)$ in the completion example of the previous section, the rule

$\mathsf{s}(x+\mathsf{p}(y)) \to x+y$ becomes superfluous since it can be simulated by an application of the new rule $x+\mathsf{p}(y) \to \mathsf{p}(x+y)$ followed by an application of the rule $\mathsf{s}(\mathsf{p}(x)) \to x$. Hence there is no particular reason to keep the rule $\mathsf{s}(x+\mathsf{p}(y)) \to x+y$. On the contrary, the elimination of redundant rules can greatly reduce the number of critical pairs.

The notion of *reducibility* (Definition 3.1.11) captures the class of TRSs without redundancy: right-hand sides of rewrite rules are in normal form and left-hand sides of rewrite rules cannot be rewritten by other rules.

Theorem 5.3.3 below states we can always eliminate redundancy in a complete TRS. This is achieved by the two-stage transformation defined below.

Definition 5.3.1. Given a complete TRS $\mathcal{R}$, the TRSs $\dot{\mathcal{R}}$ and $\ddot{\mathcal{R}}$ are defined as follows:

$$\dot{\mathcal{R}} = \{\ell \to r{\downarrow_{\mathcal{R}}} \mid \ell \to r \in \mathcal{R}\}$$
$$\ddot{\mathcal{R}} = \{\ell \to r \in \dot{\mathcal{R}} \mid \ell \in \mathsf{NF}(\dot{\mathcal{R}} \setminus \{\ell \to r\})\}$$

Recall from Section 1.2 (page 17) that $r{\downarrow_{\mathcal{R}}}$ denotes the (unique) normal form of r with respect to $\mathcal{R}$.

Example 5.3.2. Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\mathsf{f}(\mathsf{a}) \to \mathsf{a} \quad \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{a}))) \to \mathsf{f}(\mathsf{f}(\mathsf{a})) \quad \mathsf{g}(x,\mathsf{b}) \to \mathsf{f}(\mathsf{a}) \quad \mathsf{g}(x,\mathsf{h}(\mathsf{a})) \to \mathsf{g}(\mathsf{b},\mathsf{b}) \quad \mathsf{h}(\mathsf{a}) \to \mathsf{b}$$

Completeness is easily established. The TRS $\dot{\mathcal{R}}$ is obtained from $\mathcal{R}$ by normalizing the right-hand sides:

$$\mathsf{f}(\mathsf{a}) \to \mathsf{a} \quad \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{a}))) \to \mathsf{a} \quad \mathsf{g}(x,\mathsf{b}) \to \mathsf{a} \quad \mathsf{g}(x,\mathsf{h}(\mathsf{a})) \to \mathsf{a} \quad \mathsf{h}(\mathsf{a}) \to \mathsf{b}$$

Note that $\dot{\mathcal{R}}$ is complete and right-reduced. To obtain $\ddot{\mathcal{R}}$ we remove the rules of $\dot{\mathcal{R}}$ whose left-hand sides are reducible with another rule of $\dot{\mathcal{R}}$:

$$\mathsf{f}(\mathsf{a}) \to \mathsf{a} \qquad \mathsf{g}(x,\mathsf{b}) \to \mathsf{a} \qquad \mathsf{h}(\mathsf{a}) \to \mathsf{b}$$

The TRS $\ddot{\mathcal{R}}$ is canonical.

According to the following result, the two-stage transformation of Definition 5.3.1 transforms any complete TRS into a (normalization) equivalent canonical TRS.

Theorem 5.3.3. *If $\mathcal{R}$ is a complete* TRS *then $\ddot{\mathcal{R}}$ is a (normalization) equivalent canonical* TRS.

Proof Let $\mathcal{R}$ be a complete TRS. The inclusions $\ddot{\mathcal{R}} \subseteq \dot{\mathcal{R}} \subseteq \to_{\mathcal{R}}^{+}$ are obvious from the definitions. Since $\mathcal{R}$ and $\dot{\mathcal{R}}$ have the same left-hand sides, their normal forms coincide. We show $\mathsf{NF}(\ddot{\mathcal{R}}) \subseteq \mathsf{NF}(\dot{\mathcal{R}})$. To this end we show $\ell \notin \mathsf{NF}(\ddot{\mathcal{R}})$ whenever $\ell \to r \in \dot{\mathcal{R}}$, by induction on ℓ with respect to the well-founded (Theorem 2.4.23) order $\rhd$. If $\ell \to r \in \ddot{\mathcal{R}}$ then $\ell \notin \mathsf{NF}(\ddot{\mathcal{R}})$ trivially holds. So suppose $\ell \to r \notin \ddot{\mathcal{R}}$. By definition of $\ddot{\mathcal{R}}$, $\ell \notin \mathsf{NF}(\dot{\mathcal{R}} \setminus \{\ell \to r\})$. So there exists a rewrite rule $\ell' \to r' \in \dot{\mathcal{R}}$ different from $\ell \to r$ such that $\ell \trianglerighteq \ell'$. We distinguish two cases.

1 If $\ell \rhd \ell'$ then we obtain $\ell' \notin \mathsf{NF}(\ddot{\mathcal{R}})$ from the induction hypothesis and hence $\ell \notin \mathsf{NF}(\ddot{\mathcal{R}})$ as desired.

2 If $\ell \doteq \ell'$ then there exists a renaming such that $\ell = \ell'\sigma$. Since $\dot{\mathcal{R}}$ is right-reduced by construction, r and r' are normal forms of $\dot{\mathcal{R}}$. The same holds for $r'\sigma$ because normal forms are closed under renaming. We have $r \mathrel{\dot{\mathcal{R}}\!\leftarrow} \ell = \ell'\sigma \mathrel{\rightarrow\!\dot{\mathcal{R}}} r'\sigma$. Since $\dot{\mathcal{R}}$ is confluent as a consequence of Lemma 1.2.21, $r = r'\sigma$. Hence $\ell' \to r'$ is a variant of $\ell \to r$, contradicting the assumption that TRSs are variant-free (cf. page 75).

From Lemma 1.2.21 we infer that the TRSs $\dot{\mathcal{R}}$ and $\ddot{\mathcal{R}}$ are complete and normalization equivalent to $\mathcal{R}$. The TRS $\ddot{\mathcal{R}}$ is right-reduced because $\ddot{\mathcal{R}} \subseteq \dot{\mathcal{R}}$ and $\dot{\mathcal{R}}$ is right-reduced. From $\mathsf{NF}(\ddot{\mathcal{R}}) = \mathsf{NF}(\dot{\mathcal{R}})$ we easily infer that $\ddot{\mathcal{R}}$ is left-reduced. It follows that $\ddot{\mathcal{R}}$ is canonical. It remains to show that $\ddot{\mathcal{R}}$ is not only normalization equivalent but also (conversion) equivalent to $\mathcal{R}$. This is an immediate consequence of Lemma 1.2.20. □

Every TRS that is the result of the completion procedure of Figure 5.2 can be turned into a (normalization) equivalent canonical TRS.

Example 5.3.4. Consider the complete TRS $\mathcal{R}$ of Table 5.1. Since $\mathcal{R}$ is right-reduced, $\dot{\mathcal{R}} = \mathcal{R}$. The left-hand sides of the rewrite rules $\mathsf{s}(x+\mathsf{p}(y)) \to x+y$ and $\mathsf{p}(x-\mathsf{p}(y)) \to x-y$ are reducible by other rules in $\dot{\mathcal{R}}$. Hence they are eliminated in the second stage to obtain the TRS $\ddot{\mathcal{R}}$ consisting of the rewrite rules

$$\begin{array}{llll} x+\mathsf{0} \to x & x+\mathsf{s}(y) \to \mathsf{s}(x+y) & x+\mathsf{p}(y) \to \mathsf{p}(x+y) & \mathsf{s}(\mathsf{p}(x)) \to x \\ x-\mathsf{0} \to x & x-\mathsf{s}(y) \to \mathsf{p}(x-y) & x-\mathsf{p}(y) \to \mathsf{s}(x-y) & \mathsf{p}(\mathsf{s}(x)) \to x \end{array}$$

which is a canonical presentation of $\mathcal{R}$.

For efficiency reasons, performing such simplifications during the completion process is preferable.

Example 5.3.5. Consider the SRS consisting of the two rewrite rules

$$\mathsf{aaa} \to \epsilon \qquad\qquad \mathsf{aaaaa} \to \epsilon$$

Before computing critical pairs, we simplify the SRS. The left-hand side of the second rule can be rewritten using the first rule, resulting in the equation $\mathsf{aa} \approx \epsilon$. This equation can only be oriented from left to right:

$$\mathsf{aaa} \to \epsilon \qquad\qquad \mathsf{aa} \to \epsilon$$

Now the left-hand side of the first rule can be rewritten using the second rule, resulting in the equation $\mathsf{a} \approx \epsilon$. Again, only the orientation from left to right is possible:

$$\mathsf{a} \to \epsilon \qquad\qquad \mathsf{aa} \to \epsilon$$

At this point the second rule is superfluous since it simplifies to the trivial equation $\epsilon \approx \epsilon$ by applying the first rule twice. So we obtain the canonical SRS consisting of the single rule $\mathsf{a} \to \epsilon$, without computing a single critical pair!

We refrain from incorporating simplification into the completion procedure of Fig-

ure 5.2. Rather, in the next section we present inference rules that capture the essence of completion with simplification and we provide sufficient conditions on strategies in this inference system which guarantee that an obtained result is a canonical TRS that represents the initial ES.

We conclude this section with two uniqueness results for reduced TRSs.

Lemma 5.3.6. *Let $\mathcal{R}$ be a right-reduced* TRS *and let s be a reducible term which is minimal with respect to $\rhd$. If $s \to_{\mathcal{R}}^+ t$ then $s \to t$ is a variant of a rule in $\mathcal{R}$*

Proof Let $\ell \to r$ be the rewrite rule that is used in the first step from s to t. So $s \trianglerighteq \ell$. By assumption, $s \rhd \ell$ does not hold and thus $s \doteq \ell$. According to Lemma 2.4.15 there exists a renaming σ such that $s = \ell\sigma$. We have $s \to_{\mathcal{R}} r\sigma \to_{\mathcal{R}}^* t$. Because $\mathcal{R}$ is right-reduced, $r \in \mathsf{NF}(\mathcal{R})$. Since normal forms are closed under renaming, also $r\sigma \in \mathsf{NF}(\mathcal{R})$ and thus $r\sigma = t$. It follows that $s \to t$ is a variant of $\ell \to r$. □

Theorem 5.3.7. *Normalization equivalent reduced* TRS*s are unique.*

Proof Let $\mathcal{R}$ and $\mathcal{S}$ be normalization equivalent reduced TRSs. Suppose $\ell \to r \in \mathcal{R}$. Because $\mathcal{R}$ is right-reduced, $r \in \mathsf{NF}(\mathcal{R})$ and thus $\ell \neq r$. Hence $\ell \to_{\mathcal{S}}^+ r$ by normalization equivalence. Because $\mathcal{R}$ is left-reduced, ℓ is a minimal (with respect to $\rhd$) $\mathcal{R}$-reducible term. Another application of normalization equivalence yields that ℓ is minimal $\mathcal{S}$-reducible. Hence $\ell \to r$ is a variant of a rule in $\mathcal{S}$ by Lemma 5.3.6. □

Theorem 5.3.8. *Let $\mathcal{R}$ and $\mathcal{S}$ be equivalent canonical* TRS*s. If $\mathcal{R}$ and $\mathcal{S}$ are compatible with the same reduction order then $\mathcal{R} \doteq \mathcal{S}$.*

Proof Suppose $\mathcal{R}$ and $\mathcal{S}$ are compatible with the reduction order $>$. We show $\to_{\mathcal{R}}^! \subseteq \to_{\mathcal{S}}^!$. Let $s \to_{\mathcal{R}}^! t$. We show $t \in \mathsf{NF}(\mathcal{S})$. Let u be the unique $\mathcal{S}$-normal form of t. We have $t \to_{\mathcal{S}}^! u$ and thus $t \leftrightarrow_{\mathcal{R}}^* u$ because $\mathcal{R}$ and $\mathcal{S}$ are equivalent. Since $t \in \mathsf{NF}(\mathcal{R})$, we have $u \to_{\mathcal{R}}^! t$. If $t \neq u$ then both $t > u$ (as $t \to_{\mathcal{S}}^! u$) and $u > t$ (as $u \to_{\mathcal{R}}^! t$), which is impossible. Hence $t = u$ and thus $t \in \mathsf{NF}(\mathcal{S})$. Together with $s \leftrightarrow_{\mathcal{S}}^* t$, which follows from the equivalence of $\mathcal{R}$ and $\mathcal{S}$, we conclude $s \to_{\mathcal{S}}^! t$. We obtain $\to_{\mathcal{S}}^! \subseteq \to_{\mathcal{R}}^!$ by symmetry. Hence $\mathcal{R}$ and $\mathcal{S}$ are normalization equivalent and the result follows from Theorem 5.3.7. □

Example 5.3.9. Consider the ES $\mathcal{E}$ consisting of the single string equation

$$\mathsf{aa} \approx \mathsf{bcb}$$

As reduction order we take the lexicographic path order with precedence $\mathsf{a} > \mathsf{b}$ and $\mathsf{a} > \mathsf{c}$, which orients the equation from left to right. The single critical pair $\mathsf{abcb} \approx \mathsf{bcba}$ is oriented from left to right. There is one new critical pair: $\mathsf{abcba} \approx \mathsf{bcbbcb}$. Its left-hand side rewrites in two steps to its right-hand side. It follows that the SRS consisting of the rewrite rules

$$\mathsf{aa} \to \mathsf{bcb} \qquad\qquad \mathsf{abcb} \to \mathsf{bcba}$$

is canonical and represents $\mathcal{E}$. In order to obtain a different canonical SRS representing $\mathcal{E}$, we must change the employed reduction order. Consider the polynomial interpretation with $\mathsf{a}_{\mathbb{N}}(x) = 2x + 1$, $\mathsf{b}_{\mathbb{N}}(x) = x$, and $\mathsf{c}_{\mathbb{N}}(x) = 2x$. Again, the given equation is oriented from left to right, but this time the critical pair $\mathsf{abcb} \approx \mathsf{bcba}$ is oriented from right to left

as $4x + 1 < 4x + 2$ for all $x \in \mathbb{N}$. The single new critical pair $\mathsf{bcbbcb} \approx \mathsf{abcba}$ is joinable and hence we obtain the canonical SRS consisting of the rewrite rules

$$\mathsf{aa} \to \mathsf{bcb} \qquad\qquad \mathsf{bcba} \to \mathsf{abcb}$$

The above example also shows that equivalent canonical TRSs are not unique.

Exercises

5.20 Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{array}{lll} \mathsf{f}(\mathsf{f}(\mathsf{f}(x))) \to \mathsf{g}(\mathsf{f}(\mathsf{g}(\mathsf{f}(\mathsf{a})))) & \mathsf{f}(\mathsf{g}(\mathsf{f}(x))) \to \mathsf{g}(\mathsf{a}) & \mathsf{g}(\mathsf{g}(x)) \to \mathsf{f}(\mathsf{f}(\mathsf{a})) \\ \mathsf{g}(\mathsf{f}(\mathsf{f}(\mathsf{a}))) \to \mathsf{f}(\mathsf{f}(\mathsf{a})) & \mathsf{f}(\mathsf{f}(\mathsf{a})) \to \mathsf{g}(\mathsf{f}(\mathsf{a})) & \mathsf{g}(\mathsf{a}) \to \mathsf{g}(\mathsf{f}(\mathsf{a})) \end{array}$$

a Show that $\mathcal{R}$ is confluent. You may assume that $\mathcal{R}$ is terminating, which can readily be established with techniques presented in Chapter 9.

b Transform $\mathcal{R}$ into an equivalent canonical TRS.

5.21 In this exercise we do not assume that TRSs are variant-free. Construct a complete and variant-free TRS $\mathcal{R}$ such that $\dot{\mathcal{R}}$ is not variant-free and $\ddot{\mathcal{R}}$ is not equivalent to $\mathcal{R}$.

5.22 Consider the following six TRSs:

$$\begin{array}{rcl} \mathcal{R}_1 & \mathcal{R}_2 & \mathcal{R}_3 \\ \mathsf{f}(x,x) \to \mathsf{a} & \mathsf{f}(x,x) \to \mathsf{a} & \mathsf{f}(x,x) \to \mathsf{g}(\mathsf{b}) \\ \mathsf{f}(x,\mathsf{g}(\mathsf{a})) \to \mathsf{b} & \mathsf{f}(x,\mathsf{a}) \to \mathsf{a} & \mathsf{f}(x,\mathsf{b}) \to \mathsf{b} \\ \mathsf{g}(\mathsf{b}) \to \mathsf{a} & \mathsf{g}(\mathsf{a}) \to \mathsf{a} & \mathsf{g}(\mathsf{b}) \to \mathsf{b} \\ \mathsf{g}(\mathsf{g}(x)) \to \mathsf{g}(\mathsf{a}) & \mathsf{g}(\mathsf{g}(x)) \to \mathsf{g}(x) & \mathsf{g}(\mathsf{g}(x)) \to \mathsf{b} \\ \mathsf{a} \to \mathsf{g}(\mathsf{b}) & \mathsf{b} \to \mathsf{a} & \mathsf{a} \to \mathsf{g}(\mathsf{b}) \end{array}$$

$$\begin{array}{rcl} \mathcal{R}_4 & \mathcal{R}_5 & \mathcal{R}_6 \\ \mathsf{f}(x,x) \to \mathsf{a} & \mathsf{f}(x,x) \to \mathsf{b} & \mathsf{f}(x,x) \to \mathsf{g}(\mathsf{b}) \\ \mathsf{f}(x,\mathsf{g}(\mathsf{a})) \to \mathsf{b} & \mathsf{f}(x,\mathsf{b}) \to \mathsf{b} & \mathsf{f}(x,\mathsf{g}(\mathsf{b})) \to \mathsf{b} \\ \mathsf{g}(\mathsf{b}) \to \mathsf{a} & \mathsf{g}(\mathsf{b}) \to \mathsf{b} & \mathsf{a} \to \mathsf{g}(\mathsf{b}) \\ \mathsf{g}(\mathsf{g}(x)) \to \mathsf{g}(x) & \mathsf{g}(\mathsf{g}(x)) \to \mathsf{b} & \mathsf{g}(\mathsf{g}(x)) \to \mathsf{g}(x) \\ \mathsf{a} \to \mathsf{b} & \mathsf{a} \to \mathsf{b} & \mathsf{b} \to \mathsf{g}(\mathsf{b}) \end{array}$$

a Which of these TRSs are canonical?

b Complete the following tables:

	conversion equivalent				
$\mathcal{R}_1$					
$\mathcal{R}_2$					
$\mathcal{R}_3$					
$\mathcal{R}_4$					
$\mathcal{R}_5$	✓				
$\mathcal{R}_6$					

	normalization equivalent				
$\mathcal{R}_1$					
$\mathcal{R}_2$					
$\mathcal{R}_3$					
$\mathcal{R}_4$					
$\mathcal{R}_5$	×				
$\mathcal{R}_6$					

5.23 Suppose we interchange the two steps in Definition 5.3.1. Does the resulting procedure transform any complete TRS into an equivalent canonical TRS?

5.24 Does the ES of Example 5.3.9 admit canonical presentations different from the given ones?

5.25 ***a*** Show that the construction in Definition 5.3.1 does not transform every terminating TRS into an equivalent terminating reduced TRS.

b Can every terminating TRS be transformed into an equivalent terminating reduced TRS?

c Show that the construction in Definition 5.3.1 does not transform every semi-complete TRS into an equivalent semi-complete reduced TRS.

d Can every semi-complete TRS be transformed into an equivalent semi-complete reduced TRS?

5.4 Abstract Completion

At any moment during completion we have a set of rewrite rules and a set of equations, which are transformed in some controlled way. The following definition captures the operations that are used in completion.

Definition 5.4.1. The inference system KB operates on pairs consisting of an ES $\mathcal{E}$ and a TRS $\mathcal{R}$ over a common signature $\mathcal{F}$. It consists of the following eight inference rules:

deduce	$\dfrac{\mathcal{E},\mathcal{R}}{\mathcal{E}\cup\{s\approx t\},\mathcal{R}}$	if $s\ {}_{\mathcal{R}}\!\leftarrow\cdot\rightarrow_{\mathcal{R}} t$	compose	$\dfrac{\mathcal{E},\mathcal{R}\uplus\{s\to t\}}{\mathcal{E},\mathcal{R}\cup\{s\to u\}}$	if $t\to_{\mathcal{R}} u$
orient	$\dfrac{\mathcal{E}\uplus\{s\approx t\},\mathcal{R}}{\mathcal{E},\mathcal{R}\cup\{s\to t\}}$	if $s>t$	simplify	$\dfrac{\mathcal{E}\uplus\{s\approx t\},\mathcal{R}}{\mathcal{E}\cup\{u\approx t\},\mathcal{R}}$	if $s\to_{\mathcal{R}} u$
	$\dfrac{\mathcal{E}\uplus\{s\approx t\},\mathcal{R}}{\mathcal{E},\mathcal{R}\cup\{t\to s\}}$	if $t>s$		$\dfrac{\mathcal{E}\uplus\{s\approx t\},\mathcal{R}}{\mathcal{E}\cup\{s\approx u\},\mathcal{R}}$	if $t\to_{\mathcal{R}} u$
delete	$\dfrac{\mathcal{E}\uplus\{s\approx s\},\mathcal{R}}{\mathcal{E},\mathcal{R}}$		collapse	$\dfrac{\mathcal{E},\mathcal{R}\uplus\{t\to s\}}{\mathcal{E}\cup\{u\approx s\},\mathcal{R}}$	if $t\to_{\mathcal{R}} u$

Here $>$ is a fixed reduction order on $\mathcal{T}(\mathcal{F},\mathcal{V})$.

The side condition in deduce is very general. It allows to generate equations that are known in advance to be joinable due to the Critical Pair Lemma. In practice the use of deduce is restricted to (a subset of) critical pairs, cf. Definition 5.4.6 below.

In the body of the while loop in the procedure of Figure 5.2, simplify is used repeatedly to compute the normal forms s' and t' of the selected equation $s\approx t$. If the normal forms are equal, delete is used to finish the current iteration. Otherwise, orient is attempted to orient $s'\approx t'$ into $s'\to t'$ or $t'\to s'$ and then deduce is used to compute (new) critical pairs. So the inference system KB can simulate the completion procedure of Figure 5.2. However, the order in which to apply the various inference rules is not fixed. In this section we will show that, under some mild restrictions, any execution of the inference rules produces a complete TRS. As a result, we gain a lot of flexibility when implementing completion.

The inference rules compose and collapse are not used in the simple completion procedure of Figure 5.2. They are needed to perform the operations in the preceding section to obtain canonical TRSs.

Notation. We write $(\mathcal{E},\mathcal{R})$ for the pair $\mathcal{E},\mathcal{R}$ when it increases readability, typically in running text. We write $(\mathcal{E},\mathcal{R})\vdash_{\mathsf{KB}}(\mathcal{E}',\mathcal{R}')$ if $(\mathcal{E}',\mathcal{R}')$ can be obtained from $(\mathcal{E},\mathcal{R})$ by applying one of the inference rules of Definition 5.4.1.

Example 5.4.2. Consider the SRS of Example 5.3.5:

$$\mathsf{aaa} \xrightarrow{1} \epsilon \qquad\qquad \mathsf{aaaaa} \xrightarrow{2} \epsilon$$

We start from the pair $(\varnothing, \{1,2\})$. An application of collapse produces $(\{\mathsf{aa} \approx \epsilon\}, \{1\})$. An application of orient results in the pair $(\varnothing, \{1,3\})$ with

$$\mathsf{aa} \xrightarrow{3} \epsilon$$

Another application of collapse produces $(\{\mathsf{a} \approx \epsilon\}, \{3\})$. A second application of orient results in the pair $(\varnothing, \{3,4\})$ with

$$\mathsf{a} \xrightarrow{4} \epsilon$$

A third application of collapse produces $(\{\mathsf{a} \approx \epsilon\}, \{4\})$. Using simplify we obtain the pair $(\{\epsilon \approx \epsilon\}, \{4\})$. At this point delete is applied to obtain the final pair $(\varnothing, \{4\})$. Note that none of the inference rules of KB is applicable to this pair.

According to the following lemma the equational theory induced by $\mathcal{E} \cup \mathcal{R}$ is not affected by application of the inference rules of KB.

Lemma 5.4.3. *Suppose* $(\mathcal{E}, \mathcal{R}) \vdash_{\mathsf{KB}} (\mathcal{E}', \mathcal{R}')$.
1 *If* $s \to_{\mathcal{E} \cup \mathcal{R}} t$ *then* $s \to_{\mathcal{R}'}^{=} \cdot \to_{\mathcal{E}' \cup \mathcal{R}'}^{=} \cdot {}_{\mathcal{R}'}^{=}\!\leftarrow t$.
2 *If* $s \to_{\mathcal{E}' \cup \mathcal{R}'} t$ *then* $s \leftrightarrow_{\mathcal{E} \cup \mathcal{R}}^{*} t$.

Proof By inspecting the inference rules of KB we easily obtain the following inclusions:

deduce

$$\mathcal{E} \cup \mathcal{R} \subseteq \mathcal{E}' \cup \mathcal{R}' \qquad\qquad \mathcal{E}' \cup \mathcal{R}' \subseteq \mathcal{E} \cup \mathcal{R} \cup \underset{\mathcal{R}}{\longleftarrow} \cdot \underset{\mathcal{R}}{\longrightarrow}$$

orient

$$\mathcal{E} \cup \mathcal{R} \subseteq \mathcal{E}' \cup \mathcal{R}' \cup \mathcal{R}'^{-} \qquad\qquad \mathcal{E}' \cup \mathcal{R}' \subseteq \mathcal{E} \cup \mathcal{R} \cup \mathcal{E}^{-}$$

delete

$$\mathcal{E} \cup \mathcal{R} \subseteq \mathcal{E}' \cup \mathcal{R}' \cup = \qquad\qquad \mathcal{E}' \cup \mathcal{R}' \subseteq \mathcal{E} \cup \mathcal{R}$$

compose

$$\mathcal{E} \cup \mathcal{R} \subseteq \mathcal{E}' \cup \mathcal{R}' \cup \underset{\mathcal{R}'}{\longrightarrow} \cdot \underset{\mathcal{R}'}{\longleftarrow} \qquad\qquad \mathcal{E}' \cup \mathcal{R}' \subseteq \mathcal{E} \cup \mathcal{R} \cup \underset{\mathcal{R}}{\longrightarrow} \cdot \underset{\mathcal{R}}{\longrightarrow}$$

simplify

$$\mathcal{E} \cup \mathcal{R} \subseteq \mathcal{E}' \cup \mathcal{R}' \cup \underset{\mathcal{R}'}{\longrightarrow} \cdot \underset{\mathcal{E}'}{\longrightarrow} \cup \underset{\mathcal{E}'}{\longrightarrow} \cdot \underset{\mathcal{R}'}{\longleftarrow} \qquad \mathcal{E}' \cup \mathcal{R}' \subseteq \mathcal{E} \cup \mathcal{R} \cup \underset{\mathcal{R}}{\longleftarrow} \cdot \underset{\mathcal{E}}{\longrightarrow} \cup \underset{\mathcal{E}}{\longrightarrow} \cdot \underset{\mathcal{R}}{\longrightarrow}$$

collapse

$$\mathcal{E} \cup \mathcal{R} \subseteq \mathcal{E}' \cup \mathcal{R}' \cup \underset{\mathcal{R}'}{\longrightarrow} \cdot \underset{\mathcal{E}'}{\longrightarrow} \qquad\qquad \mathcal{E}' \cup \mathcal{R}' \subseteq \mathcal{E} \cup \mathcal{R} \cup \underset{\mathcal{R}}{\longleftarrow} \cdot \underset{\mathcal{R}}{\longrightarrow}$$

Consider for instance the collapse rule and suppose $s \approx t \in \mathcal{E} \cup \mathcal{R}$. If $s \approx t \in \mathcal{E}$ then $s \approx t \in \mathcal{E}'$ because $\mathcal{E} \subseteq \mathcal{E}'$. If $s \approx t \in \mathcal{R}$ then either $s \approx t \in \mathcal{R}'$ or $s \to_{\mathcal{R}} u$ with $u \approx t \in \mathcal{E}'$ and thus $s \to_{\mathcal{R}'} \cdot \to_{\mathcal{E}'} t$. This proves the inclusion on the left. For the inclusion on the right the reasoning is similar. Suppose $s \approx t \in \mathcal{E}' \cup \mathcal{R}'$. If $s \approx t \in \mathcal{R}'$ then $s \approx t \in \mathcal{R}$ because $\mathcal{R}' \subseteq \mathcal{R}$. If $s \approx t \in \mathcal{E}'$ then either $s \approx t \in \mathcal{E}$ or there exists a rule $u \to t \in \mathcal{R}$ with $u \to_{\mathcal{R}} s$ and thus $s \,{}_{\mathcal{R}}\!\leftarrow \cdot \to_{\mathcal{R}} t$.

Since rewrite relations are closed under contexts and substitutions, the inclusions in the right column prove statement **2**. Because each inclusion in the left column is a special case of

$$\mathcal{E} \cup \mathcal{R} \subseteq \xrightarrow[\mathcal{R}']{=} \cdot \xrightarrow[\mathcal{E}' \cup \mathcal{R}']{=} \cdot \xleftarrow[\mathcal{R}']{=}$$

also statement **1** follows from the closure under contexts and substitutions of rewrite relations. □

Corollary 5.4.4. *If* $(\mathcal{E}, \mathcal{R}) \vdash_{\mathsf{KB}}^{*} (\mathcal{E}', \mathcal{R}')$ *then* $\xleftrightarrow[\mathcal{E} \cup \mathcal{R}]{*} = \xleftrightarrow[\mathcal{E}' \cup \mathcal{R}']{*}$.

The next lemma states that termination of $\mathcal{R}$ is preserved by applications of the inference rules of KB. It is the final result in this section whose proof refers to the inference rules.

Lemma 5.4.5. *If* $(\mathcal{E}, \mathcal{R}) \vdash_{\mathsf{KB}}^{*} (\mathcal{E}', \mathcal{R}')$ *and* $\mathcal{R} \subseteq >$ *then* $\mathcal{R}' \subseteq >$.

Proof We consider a single step $(\mathcal{E}, \mathcal{R}) \vdash_{\mathsf{KB}} (\mathcal{E}', \mathcal{R}')$. The statement of the lemma follows by a straightforward induction proof. Observe that deduce, delete, and simplify do not change the set of rewrite rules and hence $\mathcal{R}' = \mathcal{R} \subseteq >$. For collapse we have $\mathcal{R}' \subsetneq \mathcal{R} \subseteq >$. In the case of orient we have $\mathcal{R}' = \mathcal{R} \cup \{s \to t\}$ with $s > t$ and hence $\mathcal{R}' \subseteq >$ follows from the assumption $\mathcal{R} \subseteq >$. Finally, consider an application of compose. So $\mathcal{R} = \mathcal{R}'' \uplus \{s \to t\}$ and $\mathcal{R}' = \mathcal{R}'' \cup \{s \to u\}$ with $t \to_{\mathcal{R}} u$. We obtain $s > t$ from the assumption $\mathcal{R} \subseteq >$. Since $>$ is a reduction order, $t > u$ follows from $t \to_{\mathcal{R}} u$. Transitivity of $>$ yields $s > u$ and hence $\mathcal{R}' \subseteq >$ as desired. □

To guarantee that the result of a finite KB derivation is a complete TRS equivalent to the initial $\mathcal{E}$, KB derivations must satisfy the fairness condition defined below. Fairness requires that critical pairs of the final TRS $\mathcal{R}_n$ which were not considered during the run are joinable in $\mathcal{R}_n$.

Definition 5.4.6. A *run* for a given ES $\mathcal{E}$ is a finite sequence

$$\mathcal{E}_0, \mathcal{R}_0 \vdash_{\mathsf{KB}} \mathcal{E}_1, \mathcal{R}_1 \vdash_{\mathsf{KB}} \cdots \vdash_{\mathsf{KB}} \mathcal{E}_n, \mathcal{R}_n$$

such that $\mathcal{E}_0 = \mathcal{E}$ and $\mathcal{R}_0 = \varnothing$. The run *fails* if $\mathcal{E}_n \neq \varnothing$. The run is *fair* if

$$\mathsf{PCP}(\mathcal{R}_n) \subseteq \downarrow_{\mathcal{R}_n} \cup \bigcup_{i=0}^{n} \leftrightarrow_{\mathcal{E}_i}$$

The reason for writing $\leftrightarrow_{\mathcal{E}_i}$ instead of $\mathcal{E}_i$ in the definition of fairness is that critical pairs are ordered, so in a fair run a (prime) critical pair $s \leftarrow\!\!\rightarrow t$ of $\mathcal{R}_n$ may be ignored

by deduce if $t \approx s$ was generated, or more generally, if $s \leftrightarrow_{\mathcal{E}_i} t$ holds at some point in the run. Non-prime critical pairs can always be ignored.

The following example illustrates the concepts defined above.

Example 5.4.7. Consider the ES $\mathcal{E}$ consisting of the two equations

$$\mathsf{f}(\mathsf{f}(x)) \approx \mathsf{g}(x) \qquad\qquad \mathsf{g}(\mathsf{a}) \approx \mathsf{b}$$

As reduction order we take LPO with precedence $\mathsf{g} > \mathsf{f} > \mathsf{b}$. Consider the run

$$\gamma\colon\ \mathcal{E}_0, \mathcal{R}_0 \vdash_{\mathsf{KB}} \mathcal{E}_1, \mathcal{R}_1 \vdash_{\mathsf{KB}} \mathcal{E}_2, \mathcal{R}_2$$

where $\mathcal{E}_0 = \mathcal{E}$, $\mathcal{R}_0 = \mathcal{E}_2 = \varnothing$, $\mathcal{E}_1 = \{\mathsf{g}(\mathsf{a}) \approx \mathsf{b}\}$, $\mathcal{R}_1 = \{1\}$, and $\mathcal{R}_2 = \{1, 2\}$ with

$$\mathsf{g}(x) \xrightarrow{1} \mathsf{f}(\mathsf{f}(x)) \qquad\qquad \mathsf{g}(\mathsf{a}) \xrightarrow{2} \mathsf{b}$$

The run is obviously non-failing. It is not fair since the (prime) critical pair $\mathsf{f}(\mathsf{f}(\mathsf{a})) \approx \mathsf{b}$ of $\mathcal{R}_2$ is neither joinable in $\mathcal{R}_2$ nor contained in $\leftrightarrow_{\mathcal{E}_0}$ or $\leftrightarrow_{\mathcal{E}_1}$. Note that $\mathcal{R}_2$ is not confluent. Suppose we extend γ with a collapse and an orient step:

$$\mathcal{E}_2, \mathcal{R}_2 \vdash_{\mathsf{KB}} \mathcal{E}_3, \mathcal{R}_3 \vdash_{\mathsf{KB}} \mathcal{E}_4, \mathcal{R}_4$$

where $\mathcal{E}_3 = \{\mathsf{f}(\mathsf{f}(\mathsf{a})) \approx \mathsf{b}\}$, $\mathcal{R}_3 = \mathcal{R}_1$, $\mathcal{E}_4 = \varnothing$, and $\mathcal{R}_4 = \{1, 3\}$ with

$$\mathsf{f}(\mathsf{f}(\mathsf{a})) \xrightarrow{3} \mathsf{b}$$

The extended run is non-failing and fair, and $\mathcal{R}_4$ is a complete presentation of $\mathcal{E}$.

According to the main result of this section (Theorem 5.4.10), a completion procedure that produces fair runs is correct. The challenge is the confluence proof of $\mathcal{R}_n$. We show that $\mathcal{R}_n$ is peak decreasing by labeling rewrite steps (not only in $\mathcal{R}_n$) with multisets of terms. As well-founded order on these multisets we take the multiset extension of $>$.

Definition 5.4.8. Let $\to$ be a rewrite relation and M a finite multiset of terms. We write $s \xrightarrow{M} t$ if $s \to t$ and there exist terms $s', t' \in M$ such that $s' \geqslant s$ and $t' \geqslant t$. Here $\geqslant$ denotes the reflexive closure of the given reduction order $>$.

Lemma 5.4.9. *Suppose* $(\mathcal{E}, \mathcal{R}) \vdash_{\mathsf{KB}} (\mathcal{E}', \mathcal{R}')$. *If* $s \xleftrightarrow[\mathcal{E}\cup\mathcal{R}]{M}{}^{*} t$ *and* $\mathcal{R}' \subseteq >$ *then* $s \xleftrightarrow[\mathcal{E}'\cup\mathcal{R}']{M}{}^{*} t$.

Proof We consider a single $(\mathcal{E} \cup \mathcal{R})$-step from s to t. The statement of the lemma follows then by induction on the length of the conversion between s and t. According to Lemma 5.4.3 1 there exist terms u and v such that

$$s \xrightarrow[\mathcal{R}']{=} u \xrightarrow[\mathcal{E}'\cup\mathcal{R}']{=} v \xleftarrow[\mathcal{R}']{=} t$$

We claim that the (non-empty) steps can be labeled by M. There exist terms $s', t' \in M$ with $s' \geqslant s$ and $t' \geqslant t$. Since $\mathcal{R}' \subseteq >$, $s \geqslant u$ and $t \geqslant v$ and thus also $s' \geqslant u$ and $t' \geqslant v$.

Hence

$$s \xrightarrow[\mathcal{R}']{M}{}^{=} u \xrightarrow[\mathcal{E}'\cup\mathcal{R}']{M}{}^{=} v {}^{=}\xleftarrow[\mathcal{R}']{M} t$$

and thus also $s \xleftrightarrow[\mathcal{E}'\cup\mathcal{R}']{M}{}^{*} t$. □

After these preliminaries we are ready for the main result of this section.

Theorem 5.4.10. *For every fair non-failing run* γ

$$\mathcal{E}_0, \mathcal{R}_0 \vdash_{\mathsf{KB}} \mathcal{E}_1, \mathcal{R}_1 \vdash_{\mathsf{KB}} \cdots \vdash_{\mathsf{KB}} \mathcal{E}_n, \mathcal{R}_n$$

the TRS $\mathcal{R}_n$ *is a complete presentation of* $\mathcal{E}$.

Proof We have $\mathcal{E}_n = \varnothing$. From Corollary 5.4.4 we know $\leftrightarrow^*_{\mathcal{E}} = \leftrightarrow^*_{\mathcal{R}_n}$. Lemma 5.4.5 yields $\mathcal{R}_n \subseteq >$ and hence $\mathcal{R}_n$ is terminating. It remains to prove that $\mathcal{R}_n$ is confluent. Let

$$t \xleftarrow[\mathcal{R}_n]{M_1} s \xrightarrow[\mathcal{R}_n]{M_2} u$$

From Lemma 5.1.16 we obtain $t \mathrel{\nabla_s^2} u$. Let $v \mathrel{\nabla_s} w$ appear in this sequence (so $t = v$ or $w = u$). We obtain

$$(v, w) \in \downarrow_{\mathcal{R}_n} \cup \bigcup_{i=0}^{n} \leftrightarrow_{\mathcal{E}_i}$$

from the definition of ∇_s and fairness of γ. We label all steps beween v and w with the multiset $\{v, w\}$. Because $s > v$ and $s > w$ we have $M_1 >_{\mathsf{mul}} \{v, w\}$ and $M_2 >_{\mathsf{mul}} \{v, w\}$. Hence by repeated applications of Lemma 5.4.9 we obtain a conversion in $\mathcal{R}_n$ between v and w in which each step is labeled with a multiset that is smaller than both M_1 and M_2. It follows that $\mathcal{R}_n$ is peak decreasing. □

A completion procedure is a program that generates KB runs. In order to ensure that the final outcome $\mathcal{R}_n$ is a complete presentation of the initial ES, fair runs should be produced. Fairness requires that prime critical pairs of $\mathcal{R}_n$ are considered during the run. Of course, $\mathcal{R}_n$ is not known during the run, so to be on the safe side, prime critical pairs of any $\mathcal{R}$ that appears during the run should be generated by deduce. (If a critical pair is generated from a rewrite rule that disappears at a later stage, it can be safely deleted from the run.) In particular, there is no need to deduce equations that are not prime critical pairs. So we may strengthen the condition $s \mathrel{\mathcal{R}}\!\leftarrow \cdot \rightarrow_{\mathcal{R}} t$ of deduce to $s \approx t \in \mathsf{PCP}(\mathcal{R})$ without affecting Theorem 5.4.10.

Exercises

5.26 Consider the ES $\mathcal{E}$ of Example 5.4.7. Use LPO with precedence $\mathsf{b} > \mathsf{g} > \mathsf{f} > \mathsf{a}$ to construct a fair non-failing run starting from $(\mathcal{E}, \varnothing)$.

5.27 ***a*** Complete the ES consisting of the equations $\mathsf{aa} \approx \mathsf{a}$ and $\mathsf{bba} \approx \mathsf{b}$.

b Obtain the canonical TRS of Table 3.2 by completing the ES of Exercise 2.39.

$\mathsf{e} \cdot x \approx x$
$x^{-} \cdot x \approx \mathsf{e}$
$(x \cdot y) \cdot z \approx x \cdot (y \cdot z)$
$x \cdot y \approx y \cdot x$

Table 5.2: Abelian groups.

5.28 Consider the ES $\mathcal{E}$ consisting of the following two equations:

$$\mathsf{f}(\mathsf{a},\mathsf{a}) \approx \mathsf{b} \qquad\qquad \mathsf{f}(\mathsf{f}(x,x),\mathsf{a}) \approx \mathsf{f}(x,\mathsf{f}(\mathsf{a},\mathsf{f}(x,x)))$$

a Construct three different canonical TRSs that represent $\mathcal{E}$.

b Construct a model for $\mathcal{E}$ in which the equation $\mathsf{f}(\mathsf{a},\mathsf{b}) \approx \mathsf{f}(\mathsf{b},\mathsf{a})$ is not valid.

5.29 Find all canonical TRSs that represent the ES consisting of the single equation $\mathsf{aa} \approx \mathsf{b}$.

5.30 Consider the ES $\mathcal{E}$ consisting of the two equations $\mathsf{f}(x,\mathsf{g}(y)) \approx \mathsf{g}(\mathsf{g}(x))$ and $\mathsf{g}(\mathsf{f}(x,x)) \approx x$. Does $\mathcal{E} \vDash \mathsf{g}(\mathsf{g}(x)) \approx x$ hold?

5.31 Consider the SRS $\mathcal{R}$ of Exercise 5.2.

a Transform $\mathcal{R}$ into an equivalent canonical TRS.

b Does $\mathcal{R} \vDash \mathsf{TAGCTAGCTAGCT} \approx \mathsf{CTGACTGACT}$ hold?

c Does $\mathcal{R} \vDash \mathsf{TAGCTAGCTAGCT} \approx \mathsf{CTGCTACTGACT}$ hold?

5.5 Limitations of Completion

There are ESs with a decidable validity problem but which lack a complete presentation. Consider for instance the axiomatization of *abelian groups* in Table 5.2. The commutativity axiom $x \cdot y \approx y \cdot x$ cannot be oriented into a terminating rewrite rule and it can be shown that there is no equivalent complete TRS. However, in Chapter 12 we present specialized rewrite techniques in the presence of associativity and commutativity axioms, which readily apply to abelian groups.

Commutativity axioms are not the only reason why finite complete presentations need not exist.

Theorem 5.5.1. *The validity problem for the* ES *consisting of the single string equation* $\mathsf{aba} \approx \mathsf{bab}$ *is decidable but there is no equivalent finite complete* TRS.

Proof Let $\mathcal{E}$ be the given ES. Decidability of the validity problem for $\mathcal{E}$ follows from the obvious observation that equivalence classes are finite. We have $\mathsf{abbab} \underset{\mathcal{E}}{\leftarrow} \mathsf{ababa} \underset{\mathcal{E}}{\rightarrow} \mathsf{babba}$ and by induction we obtain

$$\mathsf{a}^{i+1}\mathsf{b}^{j+2}\mathsf{ab} \leftrightarrow_{\mathcal{E}}^{*} \mathsf{bab}^{i+2}\mathsf{a}^{j+1} \tag{5.1}$$

for all $i, j \geqslant 0$. Given a string s, we write $[s]_{\mathcal{E}}$ for the set $\{t \mid s \overset{*}{\underset{\mathcal{E}}{\leftrightarrow}} t\}$ of equivalent strings. Another easy induction proof yields

$$[\mathsf{b}^n\mathsf{ab}]_{\mathcal{E}} = \{\mathsf{b}^{n-i}\mathsf{aba}^i \mid 0 \leqslant i \leqslant n\} \quad \text{and} \quad [\mathsf{bab}^n]_{\mathcal{E}} = \{\mathsf{a}^i\mathsf{bab}^{n-i} \mid 0 \leqslant i \leqslant n\} \tag{5.2}$$

for all $n > 0$. Now suppose to the contrary that there exists a finite complete presentation $\mathcal{R}$ of $\mathcal{E}$. According to Theorem 5.3.3 we may assume that $\mathcal{R}$ is canonical. Since proper

subterms of aba and bab are not equivalent to any other term, $\mathcal{R}$ must contain $\mathsf{aba} \to \mathsf{bab}$ or $\mathsf{bab} \to \mathsf{aba}$. Assume without loss of generality that the former is the case. Let m be the length of the longest left-hand side of $\mathcal{R}$ (where we do not count the implicit variable). We have

$$s_1 = \mathsf{a}^{m+1}\mathsf{b}^{m+2}\mathsf{ab} \leftrightarrow_{\mathcal{E}}^* \mathsf{bab}^{m+2}\mathsf{a}^{m+1} = s_2$$

by taking $i = j = m$ in (5.1). These two strings must have a common reduct in $\mathcal{R}$. Strings of the form $\mathsf{a}^i\mathsf{b}^j$ and $\mathsf{b}^i\mathsf{a}^j$ are in normal form since they are not equivalent to any other string. Together with the choice of m, this implies that the only rewrite rules applicable to s_1 must have $\mathsf{b}^k\mathsf{ab}$ for some $1 < k < m-1$ as left-hand side. According to (5.2) we must have $r = \mathsf{b}^{k-i}\mathsf{aba}^i$ for the right-hand side r of a rule $\mathsf{b}^k\mathsf{ab} \to r$. Since $i > 0$, r can be rewritten using the rule $\mathsf{aba} \to \mathsf{bab}$, contradicting the fact that $\mathcal{R}$ is right-reduced. Hence s_1 is a normal form of $\mathcal{R}$. A similar argument shows that s_2 is a normal form of $\mathcal{R}$. It follows that s_1 and s_2 are not joinable, yielding the desired contradiction. □

The following example shows that even if an ES $\mathcal{E}$ admits a finite complete presentation $\mathcal{R}$, completion may not be able to complete $\mathcal{E}$, irrespective of the used reduction order.

Example 5.5.2. Consider the ES $\mathcal{E}$ consisting of the two equations

$$\mathsf{f}(x) \approx \mathsf{f}(\mathsf{a}) \qquad\qquad \mathsf{f}(\mathsf{b}) \approx \mathsf{b}$$

Let $>$ be a reduction order. If $\mathsf{f}(\mathsf{b}) > \mathsf{b}$ does not hold, no inference rule of KB is applicable to $(\mathcal{E}, \varnothing)$. If $\mathsf{f}(\mathsf{b}) > \mathsf{b}$ then the second equation can be oriented

$$(\mathcal{E}, \varnothing) \vdash (\{\mathsf{f}(x) \approx \mathsf{f}(\mathsf{a})\}, \{\mathsf{f}(\mathsf{b}) \to \mathsf{b}\})$$

At this point trivial equations of the shape $\mathsf{f}^n(\mathsf{b}) \approx \mathsf{f}^n(\mathsf{b})$ with $n \geqslant 0$ can be deduced and subsequently deleted. No other possibilities exist and hence completion will fail on $\mathcal{E}$. Nevertheless, the TRS $\mathcal{R}$ consisting of the rewrite rule $\mathsf{f}(x) \to \mathsf{b}$ constitutes a complete presentation of $\mathcal{E}$.

We proceed to show that *ground* ESs can always be completed.

Definition 5.5.3. The inference system KB^- consists of the inference rules of KB except deduce.

The following result does not hold if we replace KB^- by KB. The proof is left to the reader (Exercise 5.36).

Lemma 5.5.4. *There are no infinite sequences* $\mathcal{E}_0, \varnothing \vdash_{\mathsf{KB}^-} \mathcal{E}_1, \mathcal{R}_1 \vdash_{\mathsf{KB}^-} \cdots$.

Theorem 5.5.5. *If* $>$ *is total on* $\mathcal{E}$*-equivalent ground terms then every maximal* KB^- *run produces an equivalent canonical presentation for every ground* ES $\mathcal{E}$.

Proof Consider a maximal KB^- run

$$\mathcal{E}_0, \varnothing \vdash_{\mathsf{KB}^-} \mathcal{E}_1, \mathcal{R}_1 \vdash_{\mathsf{KB}^-} \cdots \vdash_{\mathsf{KB}^-} \mathcal{E}_n, \mathcal{R}_n$$

where $\mathcal{E}_0 = \mathcal{E}$ is a ground ES. Because the run is maximal, no inference rule of KB^- is applicable to the final pair $(\mathcal{E}_n, \mathcal{R}_n)$. In particular, compose and collapse are not applicable and hence the final TRS $\mathcal{R}_n$ is reduced. Since $\mathcal{R}_n$ is also ground, it is canonical. From Corollary 5.4.4 we infer that $\mathcal{E}$ and $\mathcal{E}_n \cup \mathcal{R}_n$ are equivalent. It follows that $>$ is total on $\mathcal{E}_n$-equivalent ground terms and thus $\mathcal{E}_n = \varnothing$, for otherwise the run could be extended with an application of delete, orient or simplify. Hence $\mathcal{R}_n$ and $\mathcal{E}$ are equivalent. □

The restriction on the reduction order $>$ in the above correctness theorem is easy to satisfy. In particular, it holds for any LPO or KBO based on a total precedence (cf. Lemmata 4.3.14 and 4.4.17).

The final result of this section states the completeness of ground completion.

Theorem 5.5.6. *For every ground* ES $\mathcal{E}$ *and every equivalent reduced ground* TRS $\mathcal{R}$ *there exist a reduction order* $>$ *and a derivation* $\mathcal{E}, \varnothing \vdash_{\mathsf{KB}^-} \cdots \vdash_{\mathsf{KB}^-} \varnothing, \mathcal{R}$.

Proof Let $>$ be a reduction order that contains $\mathcal{R}$ and is total on $\mathcal{E}$-equivalent ground terms. Consider a maximal KB^- run starting from $\mathcal{E}$ and using $>$. According to Theorem 5.5.5, the run produces an equivalent reduced TRS $\mathcal{R}'$. Since $\mathcal{R} \subseteq >$ and $\mathcal{R}' \subseteq >'$, we obtain $\mathcal{R} = \mathcal{R}'$ from Theorem 5.3.8. It remains to show that $>$ exists. Let $\sqsupset$ be a total precedence and define $s > t$ if and only if $s \leftrightarrow^*_{\mathcal{E}} t$ and either $d_{\mathcal{R}}(s) > d_{\mathcal{R}}(t)$ or both $d_{\mathcal{R}}(s) = d_{\mathcal{R}}(t)$ and $s \sqsupset_{\mathsf{lpo}} t$. Here $d_{\mathcal{R}}(u)$ is the number of rewrite steps in $\mathcal{R}$ to normalize the term u, which is well-defined since all normalizing sequences in a left-reduced ground TRS have the same length (which follows from Theorem 1.5.15 in connection with Exercise 5.12(a)). It is easy to show that $>$ has the required properties (Exercise 5.38). □

Exercises

5.32 Prove that the ES $\mathcal{E}$ of Table 5.2 admits no equivalent complete TRS.

5.33 Consider the SRS $\mathcal{R}$ consisting of the single rewrite rule $\mathsf{abba} \to \epsilon$.

a Prove that the validity problem for $\mathcal{R}$ is decidable.

b Prove that there is no finite equivalent complete SRS.

5.34 Consider Example 5.5.2. Prove that the TRS $\mathcal{R}$ is a complete presentation of the ES $\mathcal{E}$.

5.35 Consider the ES $\mathcal{E}$ consisting of the following three equations:

$$\mathsf{f}(\mathsf{a}, \mathsf{g}(\mathsf{h}(x), x)) \approx \mathsf{b} \qquad \mathsf{f}(\mathsf{a}, \mathsf{g}(x, \mathsf{h}(x))) \approx \mathsf{g}(x, \mathsf{h}(x)) \qquad \mathsf{g}(\mathsf{h}(x), x) \approx \mathsf{g}(y, \mathsf{h}(y))$$

a Construct an equivalent complete TRS.

b Prove that there is no KB run resulting in a complete TRS.

5.36 Prove Lemma 5.5.4 for arbitrary ESs $\mathcal{E}_0$.

5.37 Consider the ES $\mathcal{E}$ of Theorem 5.5.1 and let $\mathcal{E}'$ be the extension with the equation $\mathsf{ab} \approx \mathsf{c}$.

a Prove that the relations $\leftrightarrow^*_{\mathcal{E}}$ and $\leftrightarrow^*_{\mathcal{E}'}$ coincide on strings without occurrences of c.

b Complete $\mathcal{E}'$.

5.38 Prove that the relation $>$ defined in the proof of Theorem 5.5.6 is a reduction order that contains $\mathcal{R}$ and is total on $\mathcal{E}$-equivalent ground terms.

5.39 ***a*** Let $\mathcal{R} \uplus \{\ell \to r\}$ be a reduced ground TRS such that r is not a proper subterm of ℓ. Let $\mathcal{R}'$ be the ground TRS obtained from $\mathcal{R}$ by replacing every occurrence of r in the rewrite rules by ℓ. Prove that $\mathcal{R}' \cup \{r \to \ell\}$ is reduced.

b Use the result of part (a) to compute six different canonical presentations of the ground ES consisting of the following equations:

$$\begin{array}{rcr} \mathsf{f}(\mathsf{a}) \approx \mathsf{g}(\mathsf{b},\mathsf{b}) & \mathsf{f}(\mathsf{f}(\mathsf{a})) \approx \mathsf{a} & \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{a}))) \approx \mathsf{a} \\ \mathsf{g}(\mathsf{b},\mathsf{h}(\mathsf{a})) \approx \mathsf{g}(\mathsf{b},\mathsf{b}) & \mathsf{h}(\mathsf{a}) \approx \mathsf{b} & \mathsf{i}(\mathsf{f}(\mathsf{a})) \approx \mathsf{c} \end{array}$$

5.40 Prove that every ground ES consisting of n equations admits at most 2^n different equivalent reduced TRSs.

Bibliographic Notes

Completion was introduced in the landmark paper of Knuth and Bendix [77]. Huet [58] was the first to give a correctness proof of a completion procedure incorporating simplification. An abstract inference system for completion was first proposed in Bachmair, Dershowitz and Hsiang [10] and further developed in [6, 9]. The version of Definition 5.4.1 differs in that the usual encompassment restriction in collapse is missing. This is possible because only finite runs are considered, an observation due to Sternagel and Thiemann [123]. In these and many other papers on completion, *proof orders* are used for showing correctness. The approach based on peak decreasingness developed in Section 5.4 is due to Hirokawa *et al.* [55]. The Critical Pair Lemma is from Huet [57]. Corollaries 5.1.11 and 5.1.12 are from [77]. Corollary 5.1.17 is from Kapur *et al.* [67]. Further critical pair criteria for completion are discussed in Bachmair and Dershowitz [8, 9]. Theorems 5.3.3 and 5.3.8 are due to Métivier [91]. Theorem 5.3.7 is from [55]. Exercise 5.25 is based on Gramlich [48]. Theorem 5.5.1 is from Kapur and Narendran [66], the result of Exercise 5.33 is from Jantzen [62]. Example 5.5.2 is due to Dominik Klein (personal communication). The more complicated counterexample in Exercise 5.35 is from [9]. The results on ground completion in Section 5.5 are from Snyder [120].

Chapter 6

Confluence

This chapter is devoted to the study of confluence. In Section 6.1 we introduce the important class of orthogonal rewrite systems, for which a sizable amount of theory has been developed. In Section 6.2 we present proofs terms, which provide a convenient way to represent rewrite sequences as terms. Orthogonal rewrite systems lack critical pairs. Sufficient conditions for confluence based on joinability of critical pairs in are presented in Section 6.3. The proof of the main result in that section—the confluence of left-linear development closed rewrite systems—employs proof terms. In Section 6.4 we present decreasing diagrams, a very powerful confluence technique for abstract rewrite systems. In Section 6.5 we employ decreasing diagrams to obtain a concrete confluence criterion for term rewrite systems based on labelings of rewrite steps. We also present transformation techniques that ease the task of (dis)proving confluence.

6.1 Orthogonality

In the preceding chapter we have seen that for terminating TRSs joinability of all critical pairs is a sufficient and necessary condition for confluence (Corollary 5.1.11). If a TRS is not terminating then joinability of all its critical pairs is in general not sufficient for confluence. There are even TRSs without critical pairs that are not confluent.

Example 6.1.1. Because the terms $\mathsf{f}(x,x)$ and $\mathsf{f}(y,\mathsf{g}(y))$ are not unifiable, the TRS $\mathcal{R}_1$ of Table 6.1 has no critical pairs. However, $\mathcal{R}_1$ is not confluent as the term $\mathsf{f}(\mathsf{c},\mathsf{c})$ has different normal forms:

$$\mathsf{a} \leftarrow \mathsf{f}(\mathsf{c},\mathsf{c}) \to \mathsf{f}(\mathsf{c},\mathsf{g}(\mathsf{c})) \to \mathsf{b}$$

If we change the rewrite rule $\mathsf{f}(x,\mathsf{g}(x)) \to \mathsf{b}$ of $\mathcal{R}_1$ into $\mathsf{g}(x) \to \mathsf{f}(x,\mathsf{g}(x))$ then we obtain the TRS $\mathcal{R}_2$ of Table 6.1. This TRS also lacks critical pairs. We have

$$\mathsf{c} \to \mathsf{g}(\mathsf{c}) \to \mathsf{f}(\mathsf{c},\mathsf{g}(\mathsf{c})) \to \mathsf{f}(\mathsf{g}(\mathsf{c}),\mathsf{g}(\mathsf{c})) \to \mathsf{a}$$

and hence also $\mathsf{c} \to \mathsf{g}(\mathsf{c}) \to^+ \mathsf{g}(\mathsf{a})$. One easily shows that the terms a and $\mathsf{g}(\mathsf{a})$ do not have a common reduct. Hence $\mathcal{R}_2$ lacks confluence. Unlike $\mathcal{R}_1$, the TRS $\mathcal{R}_2$ does have unique normal forms. This will easily follow from a result presented in Chapter 11.

Observe that the TRSs of Table 6.1 are not left-linear.

$\mathcal{R}_1$	$\mathcal{R}_2$
$\mathsf{f}(x,x) \to \mathsf{a}$	$\mathsf{f}(x,x) \to \mathsf{a}$
$\mathsf{f}(x,\mathsf{g}(x)) \to \mathsf{b}$	$\mathsf{g}(x) \to \mathsf{f}(x,\mathsf{g}(x))$
$\mathsf{c} \to \mathsf{g}(\mathsf{c})$	$\mathsf{c} \to \mathsf{g}(\mathsf{c})$

Table 6.1: Two non-confluent TRSs without critical pairs.

Definition 6.1.2. A TRS without critical pairs is called *non-ambiguous*. An *orthogonal* TRS is both left-linear and non-ambiguous.

Observe that it is decidable whether a given finite TRS is orthogonal. The main result of this section is the fact that every orthogonal TRS is confluent. We present two proofs of this result. The first one uses parallel rewriting, which was defined in Definition 3.2.12. The key observation is that for orthogonal rewrite systems parallel rewriting has the diamond property.

Example 6.1.3. Consider the TRS $\mathcal{R}_1$ of Table 3.1. We have

$$\mathsf{s}(0 \times 0) + (\mathsf{s}(0) \times (0 + \mathsf{s}(0))) \mathrel{+\!\!\!\rightarrow} \mathsf{s}(0) + ((0 \times (0 + \mathsf{s}(0))) + (0 + \mathsf{s}(0)))$$

but

$$\mathsf{s}(0 \times 0) + (\mathsf{s}(0) \times (0 + \mathsf{s}(0))) \mathrel{+\!\!\!\rightarrow} \mathsf{s}(0 \times 0) + ((0 \times \mathsf{s}(0)) + (0 + \mathsf{s}(0)))$$

does not hold.

Definition 6.1.4. Let R be a binary relation on terms. We write $\sigma\ R\ \tau\ [V]$ for substitutions σ, τ and a set V of variables if $\sigma(x)\ R\ \tau(x)$ for all variables $x \in V$. If V is the set of all variables $\mathcal{V}$, we simply write $\sigma\ R\ \tau$.

Lemma 6.1.5. *If $\sigma \mathrel{+\!\!\!\rightarrow} \tau\ [V]$ then $t\sigma \mathrel{+\!\!\!\rightarrow} t\tau$ for all terms t such that $\mathcal{V}\mathsf{ar}(t) \subseteq V$.*

Proof Easy induction on the structure of t. □

The following auxiliary lemma essentially states that in orthogonal TRSs proper subterms of left-hand sides cannot be changed by (parallel) rewriting, no matter how the variables are instantiated.

We write $s \mathrel{+\!\!\!\rightarrow}^{\langle i \rangle} t$ if we want to indicate the clause $\boxed{i}$ that was used to derive $s \mathrel{+\!\!\!\rightarrow} t$ in Definition 3.2.12.

Lemma 6.1.6. *Let s be a proper subterm of a left-hand side of a rule in an orthogonal* TRS. *For every substitution σ and term t such that $s\sigma \mathrel{+\!\!\!\rightarrow} t$ there exists a substitution τ such that $t = s\tau$ and $\sigma \mathrel{+\!\!\!\rightarrow} \tau\ [\mathcal{V}\mathsf{ar}(s)]$.*

Proof Let $\ell \to r$ be the rewrite rule such that $s \lhd \ell$. Since orthogonal TRSs are left-linear, ℓ and hence also s is a linear term. We use induction on s. If s is a variable then we can take $\tau = \{s \mapsto t\}$. We clearly have $t = s\tau$ and $\sigma(s) \mathrel{+\!\!\!\rightarrow} \tau(s)$. Suppose $s = f(s_1, \ldots, s_n)$. We distinguish two cases.

1 If $s\sigma \mathrel{+\!\!\!\rightarrow}^{\langle 2\rangle} t$ then $t = f(t_1, \ldots, t_n)$ and $s_i\sigma \mathrel{+\!\!\!\rightarrow} t_i$ for all $1 \leqslant i \leqslant n$. According to the induction hypothesis there exist substitutions $\tau_1, \ldots, \tau_n$ such that $t_i = s_i\tau_i$ and $\sigma \mathrel{+\!\!\!\rightarrow} \tau_i$ $[\mathcal{V}\mathsf{ar}(s_i)]$. We assume without loss of generality that $\mathcal{D}\mathsf{om}(\tau_i) \subseteq \mathcal{V}\mathsf{ar}(s_i)$. Since s is linear, the substitution $\tau = \tau_1 \cup \cdots \cup \tau_n$ is well-defined and satisfies $\sigma \mathrel{+\!\!\!\rightarrow} \tau$ $[\mathcal{V}\mathsf{ar}(s)]$. We have $s\tau = f(s_1\tau, \ldots, s_n\tau) = f(s_1\tau_1, \ldots, s_n\tau_n) = f(t_1, \ldots, t_n)$.

2 Suppose $s\sigma \mathrel{+\!\!\!\rightarrow}^{\langle 3\rangle} t$. So there exist a rewrite rule $\ell' \to r'$ and a substitution τ such that $s\sigma = \ell'\tau$. (We of course also have $t = r'\tau$, but this fact is not needed.) Without loss of generality we assume that $\ell' \to r'$ has no variables in common with $\ell \to r$. Since $\mathcal{V}\mathsf{ar}(s) \cap \mathcal{V}\mathsf{ar}(\ell') = \varnothing$, s and ℓ' are unifiable. Let $p \in \mathcal{P}\mathsf{os}_{\mathcal{F}}(\ell)$ such that $\ell|_p = s$. It follows that $\langle \ell' \to r', p, \ell \to r\rangle$ is an overlap, contradicting the fact that orthogonal TRSs are non-ambiguous. □

We are now ready to prove that parallel rewriting in orthogonal TRSs has the diamond property.

Parallel Moves Lemma. *Parallel rewriting has the diamond property for every orthogonal* TRS.

Proof Let $s \mathrel{+\!\!\!\rightarrow} t$ and $s \mathrel{+\!\!\!\rightarrow} u$. We show the existence of a term v such that $t \mathrel{+\!\!\!\rightarrow} v$ and $u \mathrel{+\!\!\!\rightarrow} v$ by induction on the derivation of $s \mathrel{+\!\!\!\rightarrow} t$.

1 If $s \mathrel{+\!\!\!\rightarrow}^{\langle 1\rangle} t$ then s and t are the same variable, and also u is equal to this variable, so we can take $v = s$.

2 Suppose $s \mathrel{+\!\!\!\rightarrow}^{\langle 2\rangle} t$. So $s = f(s_1, \ldots, s_n)$, $t = f(t_1, \ldots, t_n)$, and $s_i \mathrel{+\!\!\!\rightarrow} t_i$ for all $1 \leqslant i \leqslant n$. We distinguish two further cases, depending on whether $s \mathrel{+\!\!\!\rightarrow} u$ was obtained by case 2 or case 3.

- ▷ If $s \mathrel{+\!\!\!\rightarrow}^{\langle 2\rangle} u$ then $u = f(u_1, \ldots, u_n)$ and $s_i \mathrel{+\!\!\!\rightarrow} u_i$ for all $1 \leqslant i \leqslant n$. According to the induction hypothesis, there exist terms $v_1, \ldots, v_n$ such that $t_i \mathrel{+\!\!\!\rightarrow} v_i$ and $u_i \mathrel{+\!\!\!\rightarrow} v_i$ for all $1 \leqslant i \leqslant n$. Let $v = f(v_1, \ldots, v_n)$. We have $t \mathrel{+\!\!\!\rightarrow}^{\langle 2\rangle} v$ and $u \mathrel{+\!\!\!\rightarrow}^{\langle 2\rangle} v$.
- ▷ Suppose $s \mathrel{+\!\!\!\rightarrow}^{\langle 3\rangle} u$. So there exist a rewrite rule $\ell \to r$ and a substitution σ such that $s = \ell\sigma$ and $u = r\sigma$. Since the root symbol of s is f, we may write $\ell = f(\ell_1, \ldots, \ell_n)$. Fix $i \in \{1, \ldots, n\}$. We have $s_i = \ell_i\sigma \mathrel{+\!\!\!\rightarrow} t_i$. According to Lemma 6.1.6 there exists a substitution τ_i such that $t_i = \ell_i\tau_i$ and $\sigma \mathrel{+\!\!\!\rightarrow} \tau_i$ $[\mathcal{V}\mathsf{ar}(\ell_i)]$. We assume without loss of generality that $\mathcal{D}\mathsf{om}(\tau_i) \subseteq \mathcal{V}\mathsf{ar}(\ell_i)$. Since $\ell = f(\ell_1, \ldots, \ell_n)$ is a linear term, the substitution $\tau = \tau_1 \cup \cdots \cup \tau_n$ is well-defined and satisfies $\ell\tau = t$ and $\sigma \mathrel{+\!\!\!\rightarrow} \tau$ $[\mathcal{V}\mathsf{ar}(\ell)]$. Let $v = r\tau$. Lemma 6.1.5 yields $u = r\sigma \mathrel{+\!\!\!\rightarrow} r\tau$ and we have $t \mathrel{+\!\!\!\rightarrow}^{\langle 3\rangle} v$.

3 Suppose $s \mathrel{+\!\!\!\rightarrow}^{\langle 3\rangle} t$. So there exist a rewrite rule $\ell \to r$ and a substitution σ such that $s = \ell\sigma$ and $t = r\sigma$. If $s \mathrel{+\!\!\!\rightarrow}^{\langle 2\rangle} u$ then we obtain $t \mathrel{+\!\!\!\rightarrow} \cdot \mathrel{\leftarrow\!\!\!+} u$ as in the preceding case (by simply swapping s and t). So suppose $s \mathrel{+\!\!\!\rightarrow}^{\langle 3\rangle} u$. There exist a rewrite rule $\ell' \to r'$ and a substitution τ such that $s = \ell'\tau$ and $u = r'\tau$. Without loss of generality we assume that $\ell \to r$ and $\ell' \to r'$ have no common variables. We have $\ell\sigma = \ell'\tau$ and because ℓ and ℓ' are linear terms, they are unifiable. Since orthogonal TRSs lack critical pairs, $\ell \to r$ and $\ell' \to r'$ must be variants. So there exists a renaming ρ such that $\ell' = \ell\rho$ and $r' = r\rho$. We have $\ell\sigma = \ell'\tau = \ell\rho\tau$ and thus $\sigma = \rho\tau$ $[\mathcal{V}\mathsf{ar}(\ell)]$. Hence also $t = r\sigma = r\rho\tau = r'\tau = u$. In particular, $t \mathrel{+\!\!\!\rightarrow} \cdot \mathrel{\leftarrow\!\!\!+} u$. □

In connection with Lemmata 1.2.13 and 3.2.14 we arrive at the main result of this section.

Corollary 6.1.7. *Orthogonal* TRS*s are confluent.*

Our second confluence proof uses multi-step rewriting.

Definition 6.1.8. Let $\mathcal{R}$ be a TRS. The *multi-step* relation $\multimap_{\mathcal{R}}$ is inductively defined on terms as follows:

1. $x \multimap_{\mathcal{R}} x$ for all variables x,
2. $f(s_1,\ldots,s_n) \multimap_{\mathcal{R}} f(t_1,\ldots,t_n)$ if $s_i \multimap_{\mathcal{R}} t_i$ for all $1 \leqslant i \leqslant n$, and
3. $\ell\sigma \multimap_{\mathcal{R}} r\tau$ if $\ell \to r \in \mathcal{R}$ and $\sigma \multimap_{\mathcal{R}} \tau\ [\mathcal{V}\mathsf{ar}(\ell)]$.

Here $\sigma \multimap_{\mathcal{R}} \tau\ [\mathcal{V}\mathsf{ar}(\ell)]$ denotes $\sigma(x) \multimap_{\mathcal{R}} \tau(x)$ for all $x \in \mathcal{V}\mathsf{ar}(\ell)$.

Like for parallel rewriting, we drop the subscript $\mathcal{R}$ when the TRS can be inferred from the context or is irrelevant and we write $s \multimap^{\langle i \rangle} t$ to indicate the clause $\boxed{\mathbf{i}}$ that was used to derive $s \multimap t$ in Definition 6.1.8.

Example 6.1.9. Consider again the TRS $\mathcal{R}_1$ of Table 3.1. We have

$$\mathsf{s}(0 \times 0) + (\mathsf{s}(0) \times (0 + \mathsf{s}(0))) \multimap \mathsf{s}(0) + ((0 \times \mathsf{s}(0)) + \mathsf{s}(0))$$

but

$$\mathsf{s}(0 \times 0) + (\mathsf{s}(0) \times (0 + \mathsf{s}(0))) \multimap \mathsf{s}(0) + ((0 \times (0 + \mathsf{s}(0))) + \mathsf{s}(0))$$

does not hold.

The following two results state that multi-step rewriting enjoys the properties expressed in Lemmata 6.1.5 and 6.1.6 for parallel rewriting. The (omitted) proofs are almost identical.

Lemma 6.1.10. *If $\sigma \multimap \tau\ [V]$ then $t\sigma \multimap t\tau$ for all terms t such that $\mathcal{V}\mathsf{ar}(t) \subseteq V$.*

Lemma 6.1.11. *Let s be a proper subterm of a left-hand side of a rule in an orthogonal* TRS*. For every substitution σ and term t such that $s\sigma \multimap t$ there exists a substitution τ such that $t = s\tau$ and $\sigma \multimap \tau\ [\mathcal{V}\mathsf{ar}(s)]$.*

Parallel rewriting is a special case of multi-step rewriting.

Lemma 6.1.12. *For every* TRS *the inclusions $\to \subseteq \twoheadrightarrow \subseteq \multimap \subseteq \to^*$ hold.*

We prove below that multi-step rewriting enjoys the diamond property. Actually, it satisfies a stronger *triangle* property. In order to state this result, we restrict multi-steps.

Definition 6.1.13. Let $\mathcal{R}$ be a TRS. The *maximal* multi-step relation $\multimap\!\!\bullet_{\mathcal{R}}$ is inductively defined on terms as follows:

1. $x \multimap\!\!\bullet_{\mathcal{R}} x$ for all variables x,
2. $f(s_1,\ldots,s_n) \multimap\!\!\bullet_{\mathcal{R}} f(t_1,\ldots,t_n)$ if $s_i \multimap\!\!\bullet_{\mathcal{R}} t_i$ for all $1 \leqslant i \leqslant n$ and $f(s_1,\ldots,s_n)$ is not a redex, and

3 $\ell\sigma \mathrel{-\!\!\bullet\!\!\to_{\mathcal{R}}} r\tau$ if $\ell \to r \in \mathcal{R}$ and $\sigma \mathrel{-\!\!\bullet\!\!\to_{\mathcal{R}}} \tau$ $[\mathcal{V}\mathsf{ar}(\ell)]$.

We usually drop the subscript $\mathcal{R}$. We obviously have $\mathrel{-\!\!\bullet\!\!\to} \subseteq \mathrel{-\!\!\circ\!\!\to}$ for every TRS. Note that the relation $\mathrel{-\!\!\bullet\!\!\to}$ is deterministic for orthogonal TRSs because the condition that $f(s_1, \ldots, s_n)$ is not a redex in case 2 excludes the possibility that case 3 also applies.

Example 6.1.14. Consider again the TRS $\mathcal{R}_1$ of Table 3.1. We do not have

$$\mathsf{s}(0 \times 0) + (\mathsf{s}(0) \times (0 + \mathsf{s}(0))) \mathrel{-\!\!\bullet\!\!\to} \mathsf{s}(0) + ((0 \times \mathsf{s}(0)) + \mathsf{s}(0))$$

but

$$\mathsf{s}(0 \times 0) + (\mathsf{s}(0) \times (0 + \mathsf{s}(0))) \mathrel{-\!\!\bullet\!\!\to} \mathsf{s}(0 + ((0 \times \mathsf{s}(0)) + \mathsf{s}(0)))$$

holds.

Lemma 6.1.15. *For every orthogonal* TRS*, if $s \mathrel{-\!\!\circ\!\!\to} t$ then $t \mathrel{-\!\!\circ\!\!\to} u$ for the term u that satisfies $s \mathrel{-\!\!\bullet\!\!\to} u$.*

Proof We use induction on the derivation of $s \mathrel{-\!\!\circ\!\!\to} t$. If $s \mathrel{-\!\!\circ\!\!\to}^{\langle 1 \rangle} t$ then s, t and u are the same variable, so $t \mathrel{-\!\!\circ\!\!\to}^{\langle 1 \rangle} u$. Suppose $s \mathrel{-\!\!\circ\!\!\to}^{\langle 2 \rangle} t$. So $s = f(s_1, \ldots, s_n)$, $t = f(t_1, \ldots, t_n)$, and $s_i \mathrel{-\!\!\circ\!\!\to} t_i$ for all $1 \leqslant i \leqslant n$. We distinguish two further cases, depending on whether s is a redex.

1 If s is no redex then $u = f(u_1, \ldots, u_n)$ and $s_i \mathrel{-\!\!\bullet\!\!\to} u_i$ for all $1 \leqslant i \leqslant n$. According to the induction hypothesis, $t_i \mathrel{-\!\!\circ\!\!\to} u_i$ for all $1 \leqslant i \leqslant n$. Hence $t \mathrel{-\!\!\circ\!\!\to}^{\langle 2 \rangle} u$.

2 If s is a redex then there exist a rewrite rule $\ell \to r$ and a substitution σ such that $s = \ell\sigma$. Let $\ell = f(\ell_1, \ldots, \ell_n)$. Fix $i \in \{1, \ldots, n\}$. We have $s_i = \ell_i\sigma \mathrel{-\!\!\circ\!\!\to} t_i$. According to Lemma 6.1.11 there exists a substitution τ_i such that $t_i = \ell_i\tau_i$ and $\sigma \mathrel{-\!\!\circ\!\!\to} \tau_i$ $[\mathcal{V}\mathsf{ar}(\ell_i)]$. We assume without loss of generality that $\mathcal{D}\mathsf{om}(\tau_i) \subseteq \mathcal{V}\mathsf{ar}(\ell_i)$. Since $\ell = f(\ell_1, \ldots, \ell_n)$ is a linear term, the substitution $\tau = \tau_1 \cup \cdots \cup \tau_n$ is well-defined and satisfies $\ell\tau = t$ and $\sigma \mathrel{-\!\!\circ\!\!\to} \tau$ $[\mathcal{V}\mathsf{ar}(\ell)]$. According to the definition of $\mathrel{-\!\!\bullet\!\!\to}$ we have $u = r\nu$ where the substitution ν satisfies $\sigma \mathrel{-\!\!\bullet\!\!\to} \nu$ $[\mathcal{V}\mathsf{ar}(\ell)]$. For every variable $x \in \mathcal{V}\mathsf{ar}(\ell)$ we have $\sigma(x) \mathrel{-\!\!\circ\!\!\to} \tau(x)$ and $\sigma(x) \mathrel{-\!\!\bullet\!\!\to} \nu(x)$ and thus $\tau(x) \mathrel{-\!\!\circ\!\!\to} \nu(x)$ by the induction hypothesis. Consequently $t = \ell\tau \mathrel{-\!\!\circ\!\!\to}^{\langle 3 \rangle} r\nu = u$.

The final case is $s \mathrel{-\!\!\circ\!\!\to}^{\langle 3 \rangle} t$. So there exist a rewrite rule $\ell \to r$ and substitutions σ, τ such that $s = \ell\sigma$, $t = r\tau$ and $\sigma \mathrel{-\!\!\circ\!\!\to} \tau$ $[\mathcal{V}\mathsf{ar}(\ell)]$. We obtain $t \mathrel{-\!\!\circ\!\!\to} u$ exactly as in the final part of the preceding case. □

The property expressed in Lemma 6.1.15 is known as the *triangle property*.

Corollary 6.1.16. *Multi-step rewriting has the diamond property for orthogonal* TRS*s.*

Proof Let $s \mathrel{-\!\!\circ\!\!\to} t$ and $s \mathrel{-\!\!\circ\!\!\to} u$. According to Lemma 6.1.15 we have $t \mathrel{-\!\!\circ\!\!\to} v$ and $u \mathrel{-\!\!\circ\!\!\to} v$ for the term v that satisfies $s \mathrel{-\!\!\bullet\!\!\to} v$. □

We conclude this section in Figure 6.1 with two diagrams that nicely summarize the key relations and properties of orthogonal TRSs.

Figure 6.1: Diamonds and triangles.

Exercises

6.1 ***a*** Which of the TRSs in Tables 3.1–3.6 are non-ambiguous?

b Which of these TRSs are orthogonal?

c Repeat these questions for the TRSs $\mathcal{R}_M$ and $\mathcal{R}_P$ of Definitions 3.3.7 and 3.3.17.

6.2 Define *multi-hole* contexts and prove $s \twoheadrightarrow\!\!\!\!+\; t$ if and only if there exist a context C with $n \geqslant 0$ holes and terms $s_1, \ldots, s_n, t_1, \ldots, t_n$ such that $s = C[s_1, \ldots, s_n]$, $t = C[t_1, \ldots, t_n]$, and $s_i \to t_i$ for all $1 \leqslant i \leqslant n$.

6.3 Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{array}{lll} 0 + y \to y & 0 \times y \to 0 & \mathsf{s}(\mathsf{s}(\mathsf{s}(x))) \to x \\ \mathsf{s}(x) + y \to \mathsf{s}(x + y) & \mathsf{s}(x) \times y \to (x \times y) + y & \end{array}$$

Determine whether the following pairs of terms are connected by a parallel step, multi-step, or maximal multi-step:

1. $\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(x))))))$ and $\mathsf{s}(\mathsf{s}(\mathsf{s}(x)))$
2. $(0 + 0) + (0 + (0 + 0))$ and $0 + (0 + 0)$
3. $(0 + 0) + (0 + (0 + 0))$ and $0 + 0$
4. $\mathsf{s}(0 \times 0) \times (0 + 0)$ and $(0 \times (0 + 0)) + 0$

6.4 Consider the TRS CL of Table 3.6.

a Compute all multi-steps starting from the term $\mathsf{S}(\mathsf{KS})(\mathsf{II})(\mathsf{IK})(\mathsf{KK})$.

b Which of these multi-steps are maximal?

6.5 Prove that every orthogonal TRS has the Z-property.

6.6 Does the relation $\twoheadrightarrow\!\!\!\!\bullet$ have the diamond property for every orthogonal TRS?

6.2 Proof Terms

Proof terms represent rewrite sequences as terms, which allows one to analyze confluence by term manipulations. Proof terms built from function symbols, variables, and *rule symbols* represent multi-steps. In a later chapter we will add a composition operator to model arbitrary rewrite sequences. Rule symbols represent rewrite rules and have a fixed arity which is the number of different variables in the represented rule. Before formally defining proof terms, we give a motivating example that demonstrates their use. Throughout this section we deal with left-linear TRSs.

Example 6.2.1. Consider the left-linear TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{array}{rcl@{\qquad\qquad}rcl} \mathsf{h}(\mathsf{f}(x,\mathsf{g}(y))) & \xrightarrow{\alpha} & \mathsf{h}(\mathsf{f}(x,\mathsf{g}(x))) & \mathsf{g}(\mathsf{a}) & \xrightarrow{\gamma} & \mathsf{g}(\mathsf{b}) \\ \mathsf{f}(\mathsf{g}(x),y) & \xrightarrow{\beta} & \mathsf{f}(\mathsf{g}(x),\mathsf{g}(x)) & \mathsf{b} & \xrightarrow{\delta} & \mathsf{a} \end{array}$$

The rewrite step $\mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{g}(\mathsf{a}))) \to \mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{g}(\mathsf{a})))$ uses rule β at position 1 and is represented by the proof term $\mathsf{h}(\beta(\mathsf{a},\mathsf{g}(\mathsf{a})))$. Because the rule β contains two variables, the rule symbol β has two arguments. The first (second) argument corresponds to the variable x (y). The multi-step $\mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{g}(\mathsf{a}))) \twoheadrightarrow \mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{b}),\mathsf{g}(\mathsf{g}(\mathsf{b}))))$ is represented by the proof term $\alpha(\gamma,\mathsf{a})$.

We use Greek letters ($\alpha,\beta,\gamma,\dots$) for rule symbols and uppercase letters ($A,B,C,\dots$) for proof terms. The following definition fixes some notation.

Definition 6.2.2. Given a linear term t, we write $\mathsf{var}(t)$ for the list $(x_1,\dots,x_n)$ of variables appearing in t in some fixed order. Moreover, $\mathsf{vpos}(t)$ denotes the corresponding list $(p_1,\dots,p_n)$ of positions in t where these variables occur. If α is a rule symbol then $\mathsf{lhs}(\alpha)$ ($\mathsf{rhs}(\alpha)$) denotes the left-hand (right-hand) side of the rewrite rule denoted by α. Furthermore $\mathsf{var}(\alpha) = \mathsf{var}(\mathsf{lhs}(\alpha))$ and similarly $\mathsf{vpos}(\alpha) = \mathsf{vpos}(\mathsf{lhs}(\alpha))$. The length of this list is the *arity* of α. Given a rule symbol α with $\mathsf{var}(\alpha) = (x_1,\dots,x_n)$ and terms $t_1,\dots,t_n$, we write $\langle t_1,\dots,t_n\rangle_\alpha$ for the substitution $\{x_i \mapsto t_i \mid 1 \leqslant i \leqslant n\}$.

The next definition introduces two basic operations on proof terms.

Definition 6.2.3. Given a proof term A, its *source* $\mathsf{src}(A)$ and *target* $\mathsf{tgt}(A)$ are computed by the following clauses (for all variables x, n-ary function symbols f, and n-ary rule symbols α):

$$\begin{array}{rcl} \mathsf{src}(x) = \mathsf{tgt}(x) & = & x \\ \mathsf{src}(f(A_1,\dots,A_n)) & = & f(\mathsf{src}(A_1),\dots,\mathsf{src}(A_n)) \\ \mathsf{src}(\alpha(A_1,\dots,A_n)) & = & \mathsf{lhs}(\alpha)\langle \mathsf{src}(A_1),\dots,\mathsf{src}(A_n)\rangle_\alpha \\ \mathsf{tgt}(f(A_1,\dots,A_n)) & = & f(\mathsf{tgt}(A_1),\dots,\mathsf{tgt}(A_n)) \\ \mathsf{tgt}(\alpha(A_1,\dots,A_n)) & = & \mathsf{rhs}(\alpha)\langle \mathsf{tgt}(A_1),\dots,\mathsf{tgt}(A_n)\rangle_\alpha \end{array}$$

Proof terms A and B are *co-initial* if they have the same source.

Example 6.2.4. Consider the TRS $\mathcal{R}$ of Example 6.2.1 and the proof terms $A = \alpha(\gamma,\mathsf{a})$ and $B = \mathsf{h}(\beta(\mathsf{a},\gamma))$. We have

$$\begin{array}{rcl} \mathsf{src}(A) & = & \mathsf{lhs}(\alpha)\langle \mathsf{src}(\gamma),\mathsf{src}(\mathsf{a})\rangle_\alpha = \mathsf{h}(\mathsf{f}(x,\mathsf{g}(y)))\{x \mapsto \mathsf{g}(\mathsf{a}), y \mapsto \mathsf{a}\} \\ & = & \mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{g}(\mathsf{a}))) \end{array}$$

and

$$\begin{array}{rcl} \mathsf{tgt}(A) & = & \mathsf{rhs}(\alpha)\langle \mathsf{tgt}(\gamma),\mathsf{tgt}(\mathsf{a})\rangle_\alpha = \mathsf{h}(\mathsf{f}(x,\mathsf{g}(x)))\{x \mapsto \mathsf{g}(\mathsf{b}), y \mapsto \mathsf{a}\} \\ & = & \mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{b}),\mathsf{g}(\mathsf{g}(\mathsf{b})))) \end{array}$$

Similar calculations yield $\mathsf{src}(B) = \mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{g}(\mathsf{a})))$ and $\mathsf{tgt}(B) = \mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{g}(\mathsf{a}))$. Since

they have the same source, A and B are co-initial.

Proof terms witness multi-steps.

Lemma 6.2.5.

1 *If A is a proof term then* $\mathsf{src}(A) \twoheadrightarrow \mathsf{tgt}(A)$.

2 *If $s \twoheadrightarrow t$ then there exists a proof term A such that* $\mathsf{src}(A) = s$ *and* $\mathsf{tgt}(A) = t$.

Proof The first statement we prove by induction on the proof term A. If A is a variable x then $\mathsf{src}(A) = \mathsf{tgt}(A) = x$ and $x \twoheadrightarrow^{\langle 1 \rangle} x$. If $A = f(A_1, \ldots, A_n)$ then $\mathsf{src}(A) = f(\mathsf{src}(A_1), \ldots, \mathsf{src}(A_n))$ and $\mathsf{tgt}(A) = f(\mathsf{tgt}(A_1), \ldots, \mathsf{tgt}(A_n))$. The induction hyopthesis yields $\mathsf{src}(A_i) \twoheadrightarrow \mathsf{tgt}(A_i)$ for $1 \leqslant i \leqslant n$. Hence $\mathsf{src}(A) \twoheadrightarrow^{\langle 2 \rangle} \mathsf{tgt}(A)$. If $A = \alpha(A_1, \ldots, A_n)$ then $\mathsf{src}(A) = \mathsf{lhs}(\alpha)\langle \mathsf{src}(A_1), \ldots, \mathsf{src}(A_n)\rangle_\alpha$ and $\mathsf{tgt}(A) = \mathsf{rhs}(\alpha)\langle \mathsf{tgt}(A_1), \ldots, \mathsf{tgt}(A_n)\rangle_\alpha$. The induction hypothesis yields $\mathsf{src}(A_i) \twoheadrightarrow \mathsf{tgt}(A_i)$ for $1 \leqslant i \leqslant n$ and thus

$$\langle \mathsf{src}(A_1), \ldots, \mathsf{src}(A_n)\rangle_\alpha \twoheadrightarrow \langle \mathsf{tgt}(A_1), \ldots, \mathsf{tgt}(A_n)\rangle_\alpha$$

Hence $\mathsf{src}(A) \twoheadrightarrow^{\langle 3 \rangle} \mathsf{tgt}(A)$.

The second statement is proved by induction on the definition of $\twoheadrightarrow$. If $s \twoheadrightarrow^{\langle 1 \rangle} t$ then s and t are identical variables and we can take $A = s$. If $s \twoheadrightarrow^{\langle 2 \rangle} t$ then $s = f(s_1, \ldots, s_n)$ and $t = f(t_1, \ldots, t_n)$ with $s_i \twoheadrightarrow t_i$ for all $1 \leqslant i \leqslant n$. The induction hypothesis yields proof terms $A_1, \ldots, A_n$ such that $\mathsf{src}(A_i) = s_i$ and $\mathsf{tgt}(A_i) = t_i$ for $1 \leqslant i \leqslant n$. Let $A = f(A_1, \ldots, A_n)$. We have $\mathsf{src}(A) = s$ and $\mathsf{tgt}(A) = t$ as desired. If $s \twoheadrightarrow^{\langle 3 \rangle} t$ then $s = \ell\sigma$ and $t = r\tau$ for some rewrite rule $\ell \to r$ and substitutions σ and τ such that $\sigma(x) \twoheadrightarrow \tau(x)$ for all $x \in \mathcal{V}\mathsf{ar}(\ell)$. Let $\mathsf{var}(\ell) = (x_1, \ldots, x_n)$. The induction hypothesis yields proof terms $A_1, \ldots, A_n$ that witness $\sigma(x_i) \twoheadrightarrow \tau(x_i)$ for $1 \leqslant i \leqslant n$. Let α be the rule symbol of $\ell \to r$ and define $A = \alpha(A_1, \ldots, A_n)$. We have $\mathsf{src}(A) = \ell\langle \mathsf{src}(A_1), \ldots, \mathsf{src}(A_n)\rangle_\alpha = \ell\sigma = s$ and similarly $\mathsf{tgt}(A) = r\tau = t$. □

The special case of a proof term with only one rule symbol corresponds to a single step and a proof term without any rule symbols denotes an empty step. Parallel rewrite steps are witnessed by proof terms without nested rule symbols.

The binary orthogonality predicate defined below determines whether the redexes of two proof terms interfere with each other.

Definition 6.2.6. We define the *orthogonality* predicate $A \circ B$ by the following clauses (for all variables x, n-ary function symbols f and n-ary rule symbols α):

$$\begin{aligned}
x &\circ x \\
f(A_1, \ldots, A_n) &\circ f(B_1, \ldots, B_n) && \iff A_i \circ B_i \quad \text{for all } 1 \leqslant i \leqslant n \\
\alpha(A_1, \ldots, A_n) &\circ \alpha(B_1, \ldots, B_n) && \iff A_i \circ B_i \quad \text{for all } 1 \leqslant i \leqslant n \\
\alpha(A_1, \ldots, A_n) &\circ \mathsf{lhs}(\alpha)\langle B_1, \ldots, B_n\rangle_\alpha && \iff A_i \circ B_i \quad \text{for all } 1 \leqslant i \leqslant n \\
\mathsf{lhs}(\alpha)\langle A_1, \ldots, A_n\rangle_\alpha &\circ \alpha(B_1, \ldots, B_n) && \iff A_i \circ B_i \quad \text{for all } 1 \leqslant i \leqslant n
\end{aligned}$$

In all other cases $A \circ B$ is false. If $A \circ B$ then A and B are said to be *orthogonal*. Orthogonal proof terms A and B are *disjoint*, denoted by $A \perp B$, if the third clause in the above definition is not used to show orthogonality.

Example 6.2.7. Consider the TRS $\mathcal{R}$ of Example 6.2.1. The proof terms $\alpha(\mathsf{g}(\mathsf{a}),\mathsf{b})$ and $\mathsf{h}(\mathsf{f}(\gamma,\mathsf{g}(\delta)))$ are disjoint. The proof terms $\beta(\mathsf{g}(\mathsf{b}),\delta)$ and $\mathsf{f}(\mathsf{g}(\delta),\delta)$ are orthogonal but not disjoint. Furthermore, $\alpha(\mathsf{g}(\mathsf{a}),\mathsf{a})$ and $\mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{a}),\gamma))$ are co-initial but not orthogonal.

Lemma 6.2.8. *Co-initial proof terms in orthogonal* TRS*s are orthogonal.*

The join operation $A \sqcup B$ is used to obtain a single proof term containing all redexes of A and B.

Definition 6.2.9. The partial *join* $(A \sqcup B)$ operation is defined as follows:

$$\begin{aligned}
x \sqcup x &= x\\
f(A_1,\ldots,A_n) \sqcup f(B_1,\ldots,B_n) &= f(A_1 \sqcup B_1,\ldots,A_n \sqcup B_n)\\
\alpha(A_1,\ldots,A_n) \sqcup \alpha(B_1,\ldots,B_n) &= \alpha(A_1 \sqcup B_1,\ldots,A_n \sqcup B_n)\\
\alpha(A_1,\ldots,A_n) \sqcup \mathsf{lhs}(\alpha)\langle B_1,\ldots,B_n\rangle_\alpha &= \alpha(A_1 \sqcup B_1,\ldots,A_n \sqcup B_n)\\
\mathsf{lhs}(\alpha)\langle A_1,\ldots,A_n\rangle_\alpha \sqcup \alpha(B_1,\ldots,B_n) &= \alpha(A_1 \sqcup B_1,\ldots,A_n \sqcup B_n)
\end{aligned}$$

Example 6.2.10. Consider the TRS $\mathcal{R}$ of Example 6.2.1. We have

$$\alpha(\mathsf{g}(\mathsf{a}),\mathsf{b}) \sqcup \mathsf{h}(\mathsf{f}(\gamma,\mathsf{g}(\delta))) = \alpha(\gamma,\delta)$$

whereas $\alpha(\mathsf{g}(\mathsf{a}),\mathsf{a}) \sqcup \mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{a}),\gamma))$ is undefined.

The important residual operation $A \,/\, B$ is used to compute which redexes in A remain after contracting the redexes of B.

Definition 6.2.11. The partial *residual* $(A \,/\, B)$ operation is defined as follows:

$$\begin{aligned}
x \,/\, x &= x\\
f(A_1,\ldots,A_n) \,/\, f(B_1,\ldots,B_n) &= f(A_1 \,/\, B_1,\ldots,A_n \,/\, B_n)\\
\alpha(A_1,\ldots,A_n) \,/\, \alpha(B_1,\ldots,B_n) &= \mathsf{rhs}(\alpha)\langle A_1 \,/\, B_1,\ldots,A_n \,/\, B_n\rangle_\alpha\\
\alpha(A_1,\ldots,A_n) \,/\, \mathsf{lhs}(\alpha)\langle B_1,\ldots,B_n\rangle_\alpha &= \alpha(A_1 \,/\, B_1,\ldots,A_n \,/\, B_n)\\
\mathsf{lhs}(\alpha)\langle A_1,\ldots,A_n\rangle_\alpha \,/\, \alpha(B_1,\ldots,B_n) &= \mathsf{rhs}(\alpha)\langle A_1 \,/\, B_1,\ldots,A_n \,/\, B_n\rangle_\alpha
\end{aligned}$$

Example 6.2.12. Consider the TRS $\mathcal{R}$ of Example 6.2.1 and the proof terms $C = \alpha(\mathsf{g}(\mathsf{a}),\delta)$ and $D = \mathsf{h}(\mathsf{f}(\gamma,\mathsf{g}(\delta)))$. We have

$$C \,/\, D = \alpha(\mathsf{g}(\mathsf{a}) \,/\, \gamma, \delta \,/\, \delta) = \alpha(\mathsf{rhs}(\gamma),\mathsf{rhs}(\delta)) = \alpha(\mathsf{g}(\mathsf{b}),\mathsf{a})$$

and

$$D \,/\, C = \mathsf{rhs}(\alpha)\langle \gamma \,/\, \mathsf{g}(\mathsf{a}), \delta \,/\, \delta\rangle_\alpha = \mathsf{h}(\mathsf{f}(x,\mathsf{g}(x)))\{x \mapsto \gamma, y \mapsto \mathsf{rhs}(\delta)\} = \mathsf{h}(\mathsf{f}(\gamma,\mathsf{g}(\gamma)))$$

Note that $\mathsf{src}(C \,/\, D) = \mathsf{tgt}(D)$, $\mathsf{src}(D \,/\, C) = \mathsf{tgt}(C)$ and $\mathsf{tgt}(C \,/\, D) = \mathsf{tgt}(D \,/\, C)$. This is not a coincidence (cf. Lemma 6.2.16 [4] below). The expression $\alpha(\mathsf{g}(\mathsf{a}),\mathsf{a}) \,/\, \mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{a}),\gamma))$ is

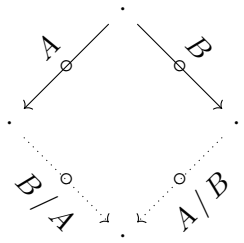


Figure 6.2: Diamond property of $\multimap$ with proof terms.

undefined.

Another useful operation on proof terms is deletion $A - B$, which is used to delete the redexes of B from A.

Definition 6.2.13. The partial *deletion* $(A - B)$ operation is defined as follows:

$$\begin{array}{rcl} x - x & = & x \\ f(A_1,\ldots,A_n) - f(B_1,\ldots,B_n) & = & f(A_1 - B_1,\ldots,A_n - B_n) \\ \alpha(A_1,\ldots,A_n) - \alpha(B_1,\ldots,B_n) & = & \mathsf{lhs}(\alpha)\langle A_1 - B_1,\ldots,A_n - B_n\rangle_\alpha \\ \alpha(A_1,\ldots,A_n) - \mathsf{lhs}(\alpha)\langle B_1,\ldots,B_n\rangle_\alpha & = & \alpha(A_1 - B_1,\ldots,A_n - B_n) \end{array}$$

Example 6.2.14. Consider the TRS $\mathcal{R}$ of Example 6.2.1. We have

$$\alpha(\gamma,\delta) - \mathsf{h}(\mathsf{f}(\gamma,\mathsf{g}(\delta))) \;=\; \alpha(\mathsf{g}(\mathsf{a}),\mathsf{b})$$

but $\mathsf{f}(\mathsf{g}(\delta),\delta) - \beta(\mathsf{b},\delta)$ is undefined.

We assume that $-$ and $/$ bind stronger than $\sqcup$.

Lemma 6.2.15. *If A and B are orthogonal proof terms then $A \sqcup B$ and $A \,/\, B$ are defined.*

The converse also holds. The following lemma states some useful properties. The fourth property can be seen as a reformulation of Corollary 6.1.16 using proof terms, cf. Figure 6.2.

Lemma 6.2.16. *If A and B are orthogonal proof terms then*

1 $A \sqcup B = B \sqcup A$,

2 $\mathsf{src}(A \sqcup B) = \mathsf{src}(A)$ *and* $\mathsf{tgt}(A \sqcup B) = \mathsf{tgt}(A) \sqcup \mathsf{tgt}(B)$,

3 *if $A - B$ is defined then* $\mathsf{src}(A - B) = \mathsf{src}(A)$,

4 $\mathsf{src}(A \,/\, B) = \mathsf{tgt}(B)$ *and* $\mathsf{tgt}(A \,/\, B) = \mathsf{tgt}(B \,/\, A)$.

Proof We proof the first part of **4** by induction. If A and B are identical variables then $A \,/\, B = A = B$ and the result holds trivially. If $A = f(A_1,\ldots,A_n)$ and $B = f(B_1,\ldots,B_n)$ then $\mathsf{src}(A \,/\, B) = f(\mathsf{src}(A_1 \,/\, B_1),\ldots,\mathsf{src}(A_n \,/\, B_n))$ and $\mathsf{tgt}(B) = f(\mathsf{tgt}(B_1),\ldots,\mathsf{tgt}(B_n))$.

Hence $\mathsf{src}(A \,/\, B) = \mathsf{tgt}(B)$ by the induction hypothesis. If $A = \alpha(A_1, \ldots, A_n)$ and $B = \alpha(B_1, \ldots, B_n)$ then

$$\begin{aligned} \mathsf{src}(A \,/\, B) &= \mathsf{rhs}(\alpha)\langle \mathsf{src}(A_1 \,/\, B_1), \ldots, \mathsf{src}(A_n \,/\, B_n)\rangle_\alpha \\ &= \mathsf{rhs}(\alpha)\langle \mathsf{tgt}(B_1), \ldots, \mathsf{tgt}(B_n)\rangle_\alpha && \text{(induction hypothesis)} \\ &= \mathsf{tgt}(B) \end{aligned}$$

If $A = \mathsf{lhs}(\alpha)\langle A_1, \ldots, A_n\rangle_\alpha$ and $B = \alpha(B_1, \ldots, B_n)$ then we obtain $\mathsf{src}(A \,/\, B) = \mathsf{tgt}(B)$ in exactly the same way. If $A = \alpha(A_1, \ldots, A_n)$ and $B = \mathsf{lhs}(\alpha)\langle B_1, \ldots, B_n\rangle_\alpha$ then $A \,/\, B = \alpha(A_1 \,/\, B_1, \ldots, A_n \,/\, B_n)$. Hence $\mathsf{src}(A \,/\, B) = \mathsf{rhs}(\alpha)\langle \mathsf{src}(A_1 \,/\, B_1), \ldots, \mathsf{src}(A_n \,/\, B_n)\rangle_\alpha$ and $\mathsf{tgt}(B) = \mathsf{rhs}(\alpha)\langle \mathsf{tgt}(B_1), \ldots, \mathsf{tgt}(B_n)\rangle_\alpha$ and we obtain $\mathsf{src}(A \,/\, B) = \mathsf{tgt}(B)$ as before.

In Exercise 6.11 the reader is asked to prove the remaining properties. □

Exercises

6.7 Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$0 + y \xrightarrow{\alpha} y \qquad 0 \times y \xrightarrow{\gamma} 0 \qquad \mathsf{s}(\mathsf{s}(x)) \xrightarrow{\epsilon} x$$
$$\mathsf{s}(x) + y \xrightarrow{\beta} \mathsf{s}(x + y) \qquad \mathsf{s}(x) \times y \xrightarrow{\delta} (x \times y) + y$$

Compute all proof terms that have the following term pairs as source and target:

1. $\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0)))))$ and $\mathsf{s}(0)$
2. $(0 + 0) + (0 + (0 + 0))$ and $0 + (0 + 0)$
3. $(0 + 0) + (0 + (0 + 0))$ and $0 + 0$
4. $\mathsf{s}(0 \times (0 + 0)) \times (0 + 0)$ and $(0 \times 0) + 0$

6.8 Consider CL

$$\mathsf{S}xyz \xrightarrow{\sigma} xz(yz) \qquad \mathsf{K}xy \xrightarrow{\kappa} x \qquad \mathsf{I}x \xrightarrow{\iota} x$$

with the following proof terms:

$$A = \sigma(\mathsf{KSI}, \iota(\mathsf{I}), \mathsf{IK})(\mathsf{KKI}) \quad B = \mathsf{S}(\mathsf{KSI})(\mathsf{II})(\iota(\mathsf{K}))(\kappa(\mathsf{K}, \mathsf{I})) \quad C = \sigma(\kappa(\mathsf{S}, \mathsf{I}), \iota(\mathsf{I}), \iota(\mathsf{K}))(\mathsf{KKI})$$

a Compute $\mathsf{src}(A)$, $\mathsf{src}(B)$ and $\mathsf{src}(C)$. Are A, B and C co-initial?

b Compute $\mathsf{tgt}(A)$, $\mathsf{tgt}(B)$ and $\mathsf{tgt}(C)$.

c Compute $(A \sqcup B) \sqcup C$ and $C - A$. Is $A - C$ defined?

d Compute $P \,/\, Q$ for all $P, Q \in \{A, B, C\}$.

6.9 Prove the identities $A \sqcup A = A$, $A \,/\, A = \mathsf{tgt}(A)$ and $A - A = \mathsf{src}(A)$ for an arbitrary proof term A.

6.10 ***a*** The join operation $\sqcup$ is known to be associative: $A \sqcup (B \sqcup C) = (A \sqcup B) \sqcup C$, even for proof terms A, B and C that are not co-initial. Prove this identity for the case $A = f(A_1, \ldots, A_n)$, $B = \beta(B_1, \ldots, B_m)$ and $C = g(C_1, \ldots, C_p)$.

b Are $/$ and $-$ associative?

6.11 Prove the remaining properties of Lemma 6.2.16.

6.3 Critical Pair Criteria

Consider again the TRSs $\mathcal{R}_1$ and $\mathcal{R}_2$ of Table 6.1, which revealed that the lack of critical pairs is no guarantee for confluence. Both TRSs are not left-linear. Moreover, $\mathcal{R}_2$ is not right-linear. Even for linear TRSs joinability of all critical pairs is not sufficient for confluence, since in the absence of termination local confluence does not guarantee confluence as we have seen in Section 1.2 (cf. Figure 1.4(i)). However, by imposing restrictions on how critical pairs join it is sometimes possible to conclude confluence for (left-)linear TRSs that are not terminating. We present three such results. The first one provides a sufficient confluence condition for linear TRSs.

Definition 6.3.1. A TRS $\mathcal{R}$ is *strongly closed* if $\mathsf{CP}(\mathcal{R}) \subseteq (\to^* \cdot {}^{=}\!\leftarrow) \cap (\to^{=} \cdot {}^{*}\!\leftarrow)$.

Theorem 6.3.2. *Linear strongly closed* TRS*s are confluent.*

Proof We follow the proof of the Critical Pair Lemma, but instead of $t_1 \downarrow t_2$, we show both $t_1 \to^* \cdot {}^{=}\!\leftarrow t_2$ and $t_1 \to^{=} \cdot {}^{*}\!\leftarrow t_2$ whenever $t_1 \leftarrow s \to t_2$. The case of parallel redexes (yielding $t_1 \to \cdot \leftarrow t_2$) and the case of identical redexes (yielding $t_1 = t_2$) require no change. In the variable overlap case we obtain $t_1 \to \cdot {}^{=}\!\leftarrow t_2$; since $\ell_2 \to r_2$ is linear the parallel step on the left in Figure 5.1(ii) is empty and the one of the right reduces to a single or empty step. In the critical overlap case we obtain $t_1 \to^* \cdot {}^{=}\!\leftarrow t_2$ and $t_1 \to^{=} \cdot {}^{*}\!\leftarrow t_2$ from the assumption on critical pairs and the closure of $\to$ under contexts and substitutions. It follows that $\leftarrow \cdot \to \subseteq \to^* \cdot {}^{=}\!\leftarrow$ and hence the relation $\to$ is strongly confluent and thus confluent (cf. Exercise 1.17). □

Example 6.3.3. Consider the linear TRS $\mathcal{R}$ consisting of the following rewrite rules:

$$(x + y) + z \to x + (y + z) \qquad x + (y + z) \to (x + y) + z \qquad x + y \to y + x$$

There are twelve critical pairs:

$$\begin{array}{ll}
(x + (y + z)) + w \approx (x + y) + (z + w) & (y + x) + z \approx x + (y + z) \\
((x + y) + z) + w \approx x + ((y + z) + w) & z + (y + x) \approx x + (y + z) \\
((x + y) + z) + w \approx x + (y + (z + w)) & x + (y + z) \approx z + (y + x) \\
x + (y + (z + w)) \approx ((x + y) + z) + w & x + (z + y) \approx (x + y) + z \\
x + (y + (z + w)) \approx (x + (y + z)) + w & (y + z) + x \approx (x + y) + z \\
x + ((y + z) + w) \approx (x + y) + (z + w) & (x + y) + z \approx (y + z) + x
\end{array}$$

We have (with $\leftrightarrows$ denoting $\to \cap \leftarrow$)

$$(x + (y + z)) + w \leftrightarrows ((x + y) + z) + w \leftrightarrows (x + y) + (z + w)$$

and hence $(x + (y + z)) + w\ (\to^* \cdot {}^{=}\!\leftarrow) \cap (\to^{=} \cdot {}^{*}\!\leftarrow)\ (x + y) + (z + w)$. The same holds for the other critical pairs (Exercise 6.13(a)). Hence $\mathcal{R}$ is strongly closed. Theorem 6.3.2 yields the confluence of $\mathcal{R}$.

The following example shows that linearity in Theorem 6.3.2 cannot be weakened to left-linearity.

Example 6.3.4. Consider the left-linear TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{array}{lll} \mathsf{h}(\mathsf{f},\mathsf{a},\mathsf{a}) \to \mathsf{h}(\mathsf{g},\mathsf{a},\mathsf{a}) & \mathsf{a} \to \mathsf{b} & \mathsf{h}(x,\mathsf{b},y) \to \mathsf{h}(x,y,y) \\ \mathsf{h}(\mathsf{g},\mathsf{a},\mathsf{a}) \to \mathsf{h}(\mathsf{f},\mathsf{a},\mathsf{a}) & & \mathsf{h}(x,y,\mathsf{b}) \to \mathsf{h}(x,y,y) \end{array}$$

One easily verifies that $\mathcal{R}$ is strongly closed (Exercise 6.13(b)). However, $\mathcal{R}$ is not confluent since $\mathsf{h}(\mathsf{f},\mathsf{b},\mathsf{b}) \mathrel{{}^*\!\leftarrow} \mathsf{h}(\mathsf{f},\mathsf{a},\mathsf{a}) \to \mathsf{h}(\mathsf{g},\mathsf{a},\mathsf{a}) \to^* \mathsf{h}(\mathsf{g},\mathsf{b},\mathsf{b})$ but $\mathsf{h}(\mathsf{f},\mathsf{b},\mathsf{b}) \downarrow \mathsf{h}(\mathsf{g},\mathsf{b},\mathsf{b})$ does not hold.

The next result does not require right-linearity but imposes a stronger restriction on the joinability of critical pairs.

Definition 6.3.5. A TRS $\mathcal{R}$ is *development closed* if $\mathsf{CP}(\mathcal{R}) \subseteq \mathrel{-\!\!\circ\!\!\to}$.

Theorem 6.3.6. *Left-linear development closed* TRS*s are confluent.*

Before presenting a proof sketch, we give an example.

Example 6.3.7. The left-linear TRS $\mathcal{R}$ of Example 6.2.1 has three critical pairs:

$$\begin{array}{rll} \mathsf{h}(\mathsf{f}(\mathsf{g}(x),\mathsf{g}(x))) \approx \mathsf{h}(\mathsf{f}(\mathsf{g}(x),\mathsf{g}(\mathsf{g}(x)))) & & \alpha(\mathsf{g}(x),x) \\ \mathsf{h}(\mathsf{f}(x,\mathsf{g}(\mathsf{b}))) \approx \mathsf{h}(\mathsf{f}(x,\mathsf{g}(x))) & & \alpha(x,\mathsf{b}) \\ \mathsf{f}(\mathsf{g}(\mathsf{b}),y) \approx \mathsf{f}(\mathsf{g}(\mathsf{a}),\mathsf{g}(\mathsf{a})) & & \beta(\delta,y) \end{array}$$

The proof terms witness the multi-steps that close the critical pairs from left to right. Hence $\mathcal{R}$ is development closed and thus confluent by Theorem 6.3.6.

The proof of Theorem 6.3.6 that we sketch below employs proof terms to establish the diamond property of $\mathrel{-\!\!\circ\!\!\to}$. The latter guarantees confluence by Lemmata 1.2.13 and 6.1.12.

Assume $t \mathrel{\leftarrow\!\!\circ} s \mathrel{-\!\!\circ\!\!\to} u$ and let A be a proof term representing $s \mathrel{-\!\!\circ\!\!\to} t$ and let B be a proof term representing $s \mathrel{-\!\!\circ\!\!\to} u$. We show $t \mathrel{-\!\!\circ\!\!\to} v \mathrel{\leftarrow\!\!\circ} u$ for some term v by well-founded induction on the amount of overlap between A and B. This measure counts the function symbol positions in s that contribute to the rule symbols in both A and B. The formal definition is given below.

Definition 6.3.8. Given a rule symbol α, we write $\mathsf{lhs}^\sharp(\alpha)$ for the result of labeling every function symbol in $\mathsf{lhs}(\alpha)$ with α as well as the distance to the root of α:

$$\mathsf{lhs}^\sharp(\alpha) = \varphi(\mathsf{lhs}(\alpha),\alpha,0)$$

with

$$\varphi(t,\alpha,i) = \begin{cases} t & \text{if } t \in \mathcal{V} \\ f_{\alpha^i}(\varphi(t_1,\alpha,i+1),\ldots,\varphi(t_n,\alpha,i+1)) & \text{if } t = f(t_1,\ldots,t_n) \end{cases}$$

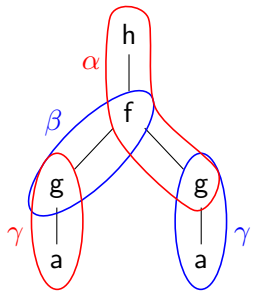


Figure 6.3: The redexes of $A = \alpha(\gamma, \mathsf{a})$ (red) and $B = \mathsf{h}(\beta(\mathsf{a}, \gamma))$ (blue).

The mapping $\mathsf{src}^\sharp$ computes the labeled source of a proof term:

$$\mathsf{src}^\sharp(A) \;=\; \begin{cases} A & \text{if } A \in \mathcal{V} \\ f(\mathsf{src}^\sharp(A_1), \ldots, \mathsf{src}^\sharp(A_n)) & \text{if } A = f(A_1, \ldots, A_n) \\ \mathsf{lhs}^\sharp(\alpha)\langle \mathsf{src}^\sharp(A_1), \ldots, \mathsf{src}^\sharp(A_n)\rangle_\alpha & \text{if } A = \alpha(A_1, \ldots, A_n) \end{cases}$$

The function ℓ extracts labels from function symbols: $\ell(f_{\alpha^n}) = \alpha^n$. The set of labeled positions for a proof term A is defined as

$$\mathcal{P}\mathsf{os}_L(A) \;=\; \{p \in \mathcal{P}\mathsf{os}(\mathsf{src}^\sharp(A)) \mid \ell(\mathsf{src}^\sharp(A)(p)) \text{ is defined}\}$$

For co-initial proof terms A and B we use the number of positions that are labeled in both $\mathsf{src}^\sharp(A)$ and $\mathsf{src}^\sharp(B)$ as a measure for the amount of overlap between A and B:

$$\blacktriangle(A, B) \;=\; |\mathcal{P}\mathsf{os}_L(A) \cap \mathcal{P}\mathsf{os}_L(B)|$$

By definition every label α^n occurring in a labeled source must be nested below n function symbols which are labeled $\alpha^{n-1}, \ldots, \alpha^0$ from bottom to top.

Example 6.3.9. Consider the TRS $\mathcal{R}$ and the proof terms $A = \alpha(\gamma, \mathsf{a})$ and $B = \mathsf{h}(\beta(\mathsf{a}, \gamma))$ of Example 6.2.4. The redexes of A and B are depicted in Figure 6.3. We have

$$\begin{aligned} \mathsf{src}^\sharp(A) &= \mathsf{h}_{\alpha^0}(\mathsf{f}_{\alpha^1}(\mathsf{g}_{\gamma^0}(\mathsf{a}_{\gamma^1}), \mathsf{g}_{\alpha^2}(\mathsf{a}))) & \mathcal{P}\mathsf{os}_L(A) &= \{\epsilon, 1, 11, 12, 111\} \\ \mathsf{src}^\sharp(B) &= \mathsf{h}(\mathsf{f}_{\beta^0}(\mathsf{g}_{\beta^1}(\mathsf{a}), \mathsf{g}_{\gamma^0}(\mathsf{a}_{\gamma^1}))) & \mathcal{P}\mathsf{os}_L(B) &= \{1, 11, 12, 121\} \end{aligned}$$

and thus $\blacktriangle(A, B) = |\{1, 11, 12\}| = 3$.

Proof (of Theorem 6.3.6) Let $\mathcal{R}$ be a left-linear development closed TRS and assume $t \mathrel{\multimap\!\!\!\!\circ\!\!\!\!\rightarrow} s \mathrel{\multimap\!\!\!\!\circ\!\!\!\!\rightarrow} u$. Let A be a proof term representing $s \mathrel{\multimap\!\!\!\!\circ\!\!\!\!\rightarrow} t$ and let B be a proof term representing $s \mathrel{\multimap\!\!\!\!\circ\!\!\!\!\rightarrow} u$. We use induction on $\blacktriangle(A, B)$ to show $t \mathrel{\multimap\!\!\!\!\circ\!\!\!\!\rightarrow} v \mathrel{\multimap\!\!\!\!\circ\!\!\!\!\leftarrow} u$ for some term v. If $\blacktriangle(A, B) = 0$ then $A \perp B$ (Exercise 6.16) and thus $A \,/\, B$ and $B \,/\, A$ are well-defined and witness the multi-steps $t \mathrel{\multimap\!\!\!\!\circ\!\!\!\!\rightarrow} \mathsf{tgt}(A \,/\, B)$ and $u \mathrel{\multimap\!\!\!\!\circ\!\!\!\!\rightarrow} \mathsf{tgt}(B \,/\, A)$. Since $\mathsf{tgt}(A \,/\, B) = \mathsf{tgt}(B \,/\, A)$ by Lemma 6.2.16 this concludes the base case of the induction.

In the induction step we have $\blacktriangle(A, B) > 0$. The picture in Figure 6.4 illustrates this case. The proof will be continued after formally defining overlaps of proof terms. □

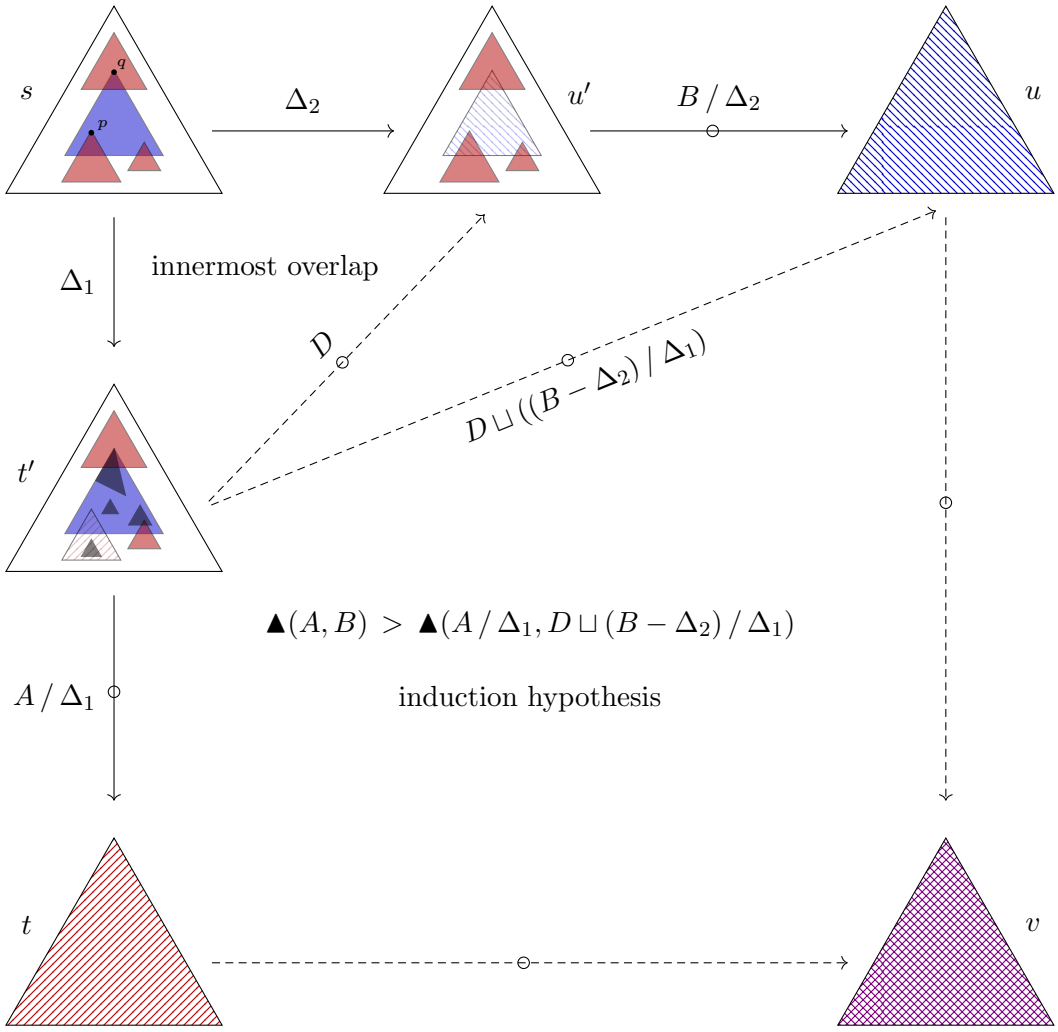


Figure 6.4: Illustration of the induction step in the proof of Theorem 6.3.6.

Definition 6.3.10. An *overlap* of co-initial proof terms A and B is a pair of positions (p, q) such that $\ell(\mathsf{src}^\sharp(A)(p)) = \alpha^0$ and $\ell(\mathsf{src}^\sharp(B)(q)) = \beta^0$ for some rule symbols α and β, and either $p \leqslant q$ and $\ell(\mathsf{src}^\sharp(A)(q)) = \alpha^d$, or $q < p$ and $\ell(\mathsf{src}^\sharp(B)(p)) = \beta^d$. Here d is the difference in length between positions p and q. We write $\mathsf{overlaps}(A, B)$ for the set of overlaps of A and B. An *innermost overlap* of A and B is a maximal element in $\mathsf{overlaps}(A, B)$ with respect to the following order on overlaps:

$$(p_1, q_1) \leqslant (p_2, q_2) \quad \Longleftrightarrow \quad p_1 \leqslant p_2 \text{ and } q_1 \leqslant q_2$$

The condition $\ell(\mathsf{src}^\sharp(A)(q)) = \alpha^d$ in the above definition ensures that $q \setminus p$ is a position in $\mathsf{lhs}(\alpha)$, and similarly for $\ell(\mathsf{src}^\sharp(B)(p)) = \beta^d$.

Example 6.3.11. In Example 6.3.9 we have $\mathsf{overlaps}(A, B) = \{(\epsilon, 1), (\epsilon, 12), (11, 1)\}$. Both $(11, 1)$ and $(\epsilon, 12)$ are innermost overlaps.

Proof (of Theorem 6.3.6, cont'd) Because $\blacktriangle(A, B) > 0$, there must be overlaps between

A and B. We select an innermost overlap $o = (p, q) \in \mathsf{overlaps}(A, B)$ and assume $q \leqslant p$ without loss of generality. Next we extract the corresponding critical pair in the underlying TRS. Let $q' = p \setminus q$ and let α and β be the rule symbols at positions p and q in $\mathsf{src}(A)$ and $\mathsf{src}(B)$ such that $\ell(\mathsf{src}^\sharp(A)(p)) = \alpha^0$ and $\ell(\mathsf{src}^\sharp(B)(q)) = \beta^0$. Furthermore let $\mathsf{vpos}(\alpha) = (p_1, \ldots, p_n)$, $\mathsf{var}(\alpha) = (x_1, \ldots, x_n)$, $\mathsf{vpos}(\beta) = (q_1, \ldots, q_m)$, and $\mathsf{var}(\beta) = (y_1, \ldots, y_m)$ where we assume $\{x_1, \ldots, x_n\} \cap \{y_1, \ldots, y_m\} = \varnothing$ without loss of generality. We associate proof terms Δ_1 and Δ_2 corresponding to the selected overlapping redexes in the common source s of A and B:

$$\Delta_1 \;=\; s[\alpha(s|_{pp_1}, \ldots, s|_{pp_n})]_p \qquad\qquad \Delta_2 \;=\; s[\beta(s|_{qq_1}, \ldots, s|_{qq_m})]_q$$

We have $\mathsf{src}(\Delta_1) = \mathsf{src}(\Delta_2) = s$. Let $t' = \mathsf{tgt}(\Delta_1)$ and $u' = \mathsf{tgt}(\Delta_2)$. The proof terms $A \,/\, \Delta_1$ and $B \,/\, \Delta_2$ witness $t' \mathrel{-\!\!\circ\!\!\to} t$ and $u' \mathrel{-\!\!\circ\!\!\to} u$. Because $\mathsf{lhs}(\alpha)$ and $\mathsf{lhs}(\beta)|_{q'}$ are linear and variable disjoint, the substitution

$$\begin{aligned}\tau_o \;=\; &\{x_i \mapsto \mathsf{lhs}(\beta)|_{q'p_i} \mid 1 \leqslant i \leqslant n \text{ and } p_i \in \mathcal{P}\mathsf{os}(\mathsf{lhs}(\beta)|_{q'})\} \\ &\cup \{y_j \mapsto \mathsf{lhs}(\alpha)|_{q_j \setminus q} \mid 1 \leqslant j \leqslant m \text{ and } q_j \setminus q \in \mathcal{P}\mathsf{os}_{\mathcal{F}}(\mathsf{lhs}(\alpha))\}\end{aligned}$$

is an idempotent mgu of these terms (Exercise 2.50). Hence we obtain the critical peak (cf. Exercise 6.18)

$$\mathsf{lhs}(\beta)[\mathsf{rhs}(\alpha)\tau_o]_{q'} \;\leftarrow\; \mathsf{lhs}(\beta)[\mathsf{lhs}(\alpha)\tau_o]_{q'} \;=\; \mathsf{lhs}(\beta)\tau_o \;\to\; \mathsf{rhs}(\beta)\tau_o \qquad (*)$$

The development closedness assumption yields a multi-step $\mathsf{lhs}(\beta)[\mathsf{rhs}(\alpha)\tau_o]_{q'} \mathrel{-\!\!\circ\!\!\to} \mathsf{rhs}(\beta)\tau_o$. Let D' be a proof term witnessing this multi-step. In Figure 6.4 this is illustrated with the black redexes in the term t'. The substitution

$$\sigma_o \;=\; \{x_i \mapsto s|_{pp_i} \mid 1 \leqslant i \leqslant n\} \cup \{y_j \mapsto s|_{qq_j} \mid 1 \leqslant j \leqslant m\}$$

maps the variables of $\mathsf{lhs}(\alpha)$ and $\mathsf{lhs}(\beta)$ to subterms of s such that $\mathsf{lhs}(\alpha)\sigma_o = s|_p$ and $\mathsf{lhs}(\beta)\sigma_o = s|_q$. We have $s = s[\mathsf{lhs}(\beta)[\mathsf{lhs}(\alpha)\tau_o]_{q'}\sigma_o]_q$ and hence the proof term $D = s[D'\sigma_o]_q$ witnesses the multi-step $t' \mathrel{-\!\!\circ\!\!\to} u'$. Composing D and $B \,/\, \Delta_2$ yields a proof term $D \sqcup (B - \Delta_2) \,/\, \Delta_1$ witnessing $t' \mathrel{-\!\!\circ\!\!\to} u$. Since position p does not contribute to the amount of overlap between $A \,/\, \Delta_1$ and $D \sqcup (B - \Delta_2) \,/\, \Delta_1$, the measure strictly decreases:

$$\blacktriangle(A, B) \;>\; \blacktriangle(A \,/\, \Delta_1, D \sqcup (B - \Delta_2) \,/\, \Delta_1)$$

A formal proof of this statement is developed in Exercise 6.19. As a consequence, the induction hypothesis is applicable to $A \,/\, \Delta_1$ and $D \sqcup (B - \Delta_2) \,/\, \Delta_1$, yielding $t \mathrel{-\!\!\circ\!\!\to} v \mathrel{\leftarrow\!\!\circ\!\!-} u$ for some term v. Hence $\mathrel{-\!\!\circ\!\!\to}$ has the diamond property and we obtain the confluence of $\mathcal{R}$ from Lemmata 1.2.13 and 6.1.12. □

Example 6.3.12. Selecting $o = (11, 1) \in \mathsf{overlaps}(A, B)$ in Example 6.3.9 results in

$$\Delta_1 \;=\; \mathsf{h}(\mathsf{f}(\gamma, \mathsf{g}(\mathsf{a}))) \qquad\qquad \Delta_2 \;=\; \mathsf{h}(\beta(\mathsf{a}, \mathsf{g}(\mathsf{a})))$$

Note that 11 is the position of γ in Δ_1 and 1 is the position of β in Δ_2. We have $q' = 1$

and $\tau_o = \{x \mapsto \mathsf{a}\}$, resulting in the critical pair

$$\mathsf{f}(\mathsf{g}(\mathsf{b}), y) \overset{\gamma}{\longleftarrow} \mathsf{f}(\mathsf{g}(\mathsf{a}), y) \overset{\beta}{\longrightarrow} \mathsf{f}(\mathsf{g}(\mathsf{a}), \mathsf{g}(\mathsf{a}))$$

which can be closed by a multi-step witnessed by the proof term $D' = \beta(\delta, y)$. Moreover, $\sigma_o = \{x \mapsto \mathsf{a}, y \mapsto \mathsf{g}(\mathsf{a})\}$ and thus $D = s[D'\sigma_o]_q = \mathsf{h}(\beta(\delta, \mathsf{g}(\mathsf{a})))$. The proof term

$$\begin{aligned} D \sqcup ((B - \Delta_2) \,/\, \Delta_1) &= \mathsf{h}(\beta(\delta, \mathsf{g}(\mathsf{a}))) \sqcup ((\mathsf{h}(\beta(\mathsf{a}, \gamma)) - \mathsf{h}(\beta(\mathsf{a}, \mathsf{g}(\mathsf{a})))) \,/\, \mathsf{h}(\mathsf{f}(\gamma, \mathsf{g}(\mathsf{a})))) \\ &= \mathsf{h}(\beta(\delta, \mathsf{g}(\mathsf{a}))) \sqcup (\mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{a}), \gamma)) \,/\, \mathsf{h}(\mathsf{f}(\gamma, \mathsf{g}(\mathsf{a})))) \\ &= \mathsf{h}(\beta(\delta, \mathsf{g}(\mathsf{a}))) \sqcup \mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{b}), \gamma)) \\ &= \mathsf{h}(\beta(\delta, \gamma)) \end{aligned}$$

witnesses the multi-step

$$\mathsf{tgt}(\Delta_1) = \mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{b}), \mathsf{g}(\mathsf{a}))) \mathrel{-\!\!\circ\!\!\to} \mathsf{h}(\mathsf{f}(\mathsf{g}(\mathsf{a}), \mathsf{g}(\mathsf{a}))) = \mathsf{tgt}(B)$$

We have $A \,/\, \Delta_1 = \alpha(\mathsf{g}(\mathsf{b}), \mathsf{a})$ and obtain $\blacktriangle(A \,/\, \Delta_1, D \sqcup (B - \Delta_2) \,/\, \Delta_1) = 2 < 3 = \blacktriangle(A, B)$.

Theorem 6.3.6 is a proper generalization of orthogonality. Note that the joinability requirement on critical pairs is not symmetric. At the time of writing it is unknown whether $\longleftarrow\!\!\rtimes\!\!\longrightarrow \subseteq \mathrel{-\!\!\circ\!\!\to}$ is a sufficient condition for confluence of left-linear TRSs. In the bibliographic notes several other open problems concerning restricted joinability conditions are listed.

Definition 6.3.13. A TRS $\mathcal{R}$ is *parallel closed* if $\mathsf{CP}(\mathcal{R}) \subseteq \twoheadrightarrow$.

Since parallel steps are multi-steps, the following is a special case of Theorem 6.3.6.

Corollary 6.3.14. *Left-linear parallel closed* TRS*s are confluent.*

It is possible to strengthen Theorem 6.3.6 somewhat by weakening the development closed requirement to $s \mathrel{-\!\!\circ\!\!\to} \cdot\, {}^*\!\longleftarrow t$ whenever $s \longleftarrow\!\!\rtimes\!\!\longrightarrow t$ is a so-called *overlay*.

Definition 6.3.15. A *root* overlap has the form $\langle \ell_1 \to r_1, \epsilon, \ell_2 \to r_2 \rangle$. A critical pair obtained from a root overlap is called an *overlay*. We write $s \longleftarrow\!\!\bowtie\!\!\longrightarrow t$ to indicate that the critical pair $s \longleftarrow\!\!\rtimes\!\!\longrightarrow t$ is an overlay and $s \overset{>\epsilon}{\longleftarrow\!\!\rtimes\!\!\longrightarrow} t$ if we want to stress that $s \longleftarrow\!\!\rtimes\!\!\longrightarrow t$ is not an overlay. A TRS is *almost development closed* if $\overset{>\epsilon}{\longleftarrow\!\!\rtimes\!\!\longrightarrow} \subseteq \mathrel{-\!\!\circ\!\!\to}$ and $\longleftarrow\!\!\bowtie\!\!\longrightarrow \subseteq \mathrel{-\!\!\circ\!\!\to} \cdot\, {}^*\!\longleftarrow$.

Example 6.3.16. The TRS of Example 6.2.1 has no overlays. Consider the TRS $\{(1), (2)\}$ of Example 5.1.8. Two of the four critical pairs are overlays. The critical pair $\mathsf{f}(x) \longleftarrow\!\!\rtimes\!\!\longrightarrow \mathsf{g}(x)$ obtained from the root overlap $\langle (1), \epsilon, (2) \rangle$ is an overlay, but the identical critical pair obtained from the overlap $\langle (1), 1, (2) \rangle$ is not an overlay.

Theorem 6.3.17. *Left-linear almost development closed* TRS*s are confluent.*

Proof sketch The proof of Theorem 6.3.6 is modified as follows. Instead of the diamond property $\mathrel{-\!\!\circ\!\!\to}$, we show that $\mathrel{-\!\!\circ\!\!\to}$ is strongly confluent ($\mathrel{\circ\!\!\leftarrow} \cdot \mathrel{-\!\!\circ\!\!\to} \subseteq \mathrel{-\!\!\circ\!\!\to} \cdot\, {}^*\!\longleftarrow$), using the same

induction measure as before. So let $t \leftarrow\!\!\circ\ s \mathrel{\circ\!\!\rightarrow} u$. Let A be a proof term representing $s \mathrel{\circ\!\!\rightarrow} t$ and let B be a proof term representing $s \mathrel{\circ\!\!\rightarrow} u$. By induction on $\blacktriangle(A, B)$ it is shown that $t \mathrel{\circ\!\!\rightarrow} v \mathrel{}^*\!\leftarrow u$ for some term v. We follow the proof of Theorem 6.3.6 and consider an innermost overlap $o = (p, q) \in \mathsf{overlaps}(A, B)$. Since strong confluence is an asymmetric condition, we cannot simply assume $q \leqslant p$ without loss of generality. However, the two cases $q < p$ and $p < q$ still work as in the proof of Theorem 6.3.6 by constructing a proof term for $\mathsf{tgt}(\Delta_1) \mathrel{\circ\!\!\rightarrow} \mathsf{tgt}(B \,/\, \Delta_2)$ and $\mathsf{tgt}(\Delta_2) \mathrel{\circ\!\!\rightarrow} \mathsf{tgt}(A \,/\, \Delta_1)$ respectively, and showing that the measure decreases. In both cases this allows us to apply the induction hypothesis and obtain $t \mathrel{\circ\!\!\rightarrow} v \mathrel{}^*\!\leftarrow u$. If $p = q$ then we have a root overlap. This special case is covered in Exercise 6.21. □

Example 6.3.18. The TRS consisting of the rewrite rules

$$\mathsf{f}(\mathsf{f}(x)) \to x \qquad \mathsf{f}(x) \to \mathsf{f}(\mathsf{f}(x)) \qquad \mathsf{f}(x) \to x$$

admits six non-trivial critical pairs:

$$\mathsf{f}(x) \approx x \quad \mathsf{f}(\mathsf{f}(x)) \approx x \quad \mathsf{f}(\mathsf{f}(\mathsf{f}(x))) \approx x \quad x \approx \mathsf{f}(x) \quad x \approx \mathsf{f}(\mathsf{f}(x)) \quad x \approx \mathsf{f}(\mathsf{f}(\mathsf{f}(x)))$$

Note that Theorem 6.3.6 is not applicable. Further note that the ones with variable left-hand side are overlays. We have $\mathsf{f}^n(x) \mathrel{\circ\!\!\rightarrow} x$ for all $n \geqslant 0$ and thus $\mathcal{R}$ is confluent by Theorem 6.3.17.

So critical pairs $s \leftarrow\!\!\!\rightarrow t$ that do not satisfy $s \mathrel{\circ\!\!\rightarrow} t$ must be overlays and satisfy $s \mathrel{\circ\!\!\rightarrow} \cdot \mathrel{}^*\!\leftarrow t$ in order to conclude confluence. Note that $s \to^* \cdot \leftarrow\!\circ\ t$ must hold as well since if $s \leftarrow\!\!\!\rightarrow t$ is an overlay then also $t \leftarrow\!\!\!\rightarrow s$ is an overlay (cf. Exercise 6.15(b)).

We conclude this section with a simple definition.

Definition 6.3.19. A left-linear TRS all of whose critical pairs are trivial overlays is called *almost orthogonal*. A left-linear TRS with only trivial critical pairs is called *weakly orthogonal*.

Clearly every orthogonal TRS is almost orthogonal and every almost orthogonal TRS is weakly orthogonal. An appeal to Theorem 6.3.6 yields the confluence of weakly orthogonal TRSs.

Example 6.3.20. The TRS consisting of the rewrite rules $\top \vee x \to \top$ and $x \vee \top \to \top$ is almost orthogonal but not orthogonal. The TRS consisting of the rewrite rules $\mathsf{s}(\mathsf{p}(x)) \to x$ and $\mathsf{p}(\mathsf{s}(x)) \to x$ is weakly orthogonal but not almost orthogonal.

Exercises

6.12 ***a*** Prove the confluence of the SRS consisting of the rewrite rules $\mathsf{aba} \to \mathsf{aa}$ and $\mathsf{caa} \to \mathsf{caba}$.

b Is the TRS consisting of the rewrite rules

$$\mathsf{max}(x, 0) \to x \quad \mathsf{max}(0, y) \to y \quad \mathsf{max}(\mathsf{s}(x), \mathsf{s}(y)) \to \mathsf{s}(\mathsf{max}(x, y)) \quad \mathsf{max}(x, y) \to \mathsf{max}(y, x)$$

confluent?

6.13 ***a*** Show that the TRS of Example 6.3.3 is strongly closed.

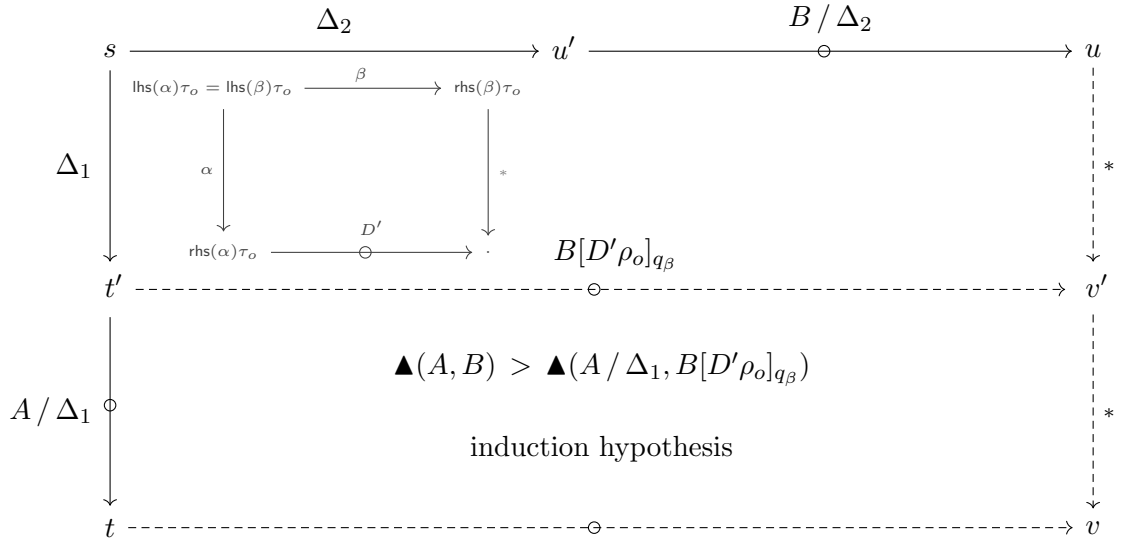


Figure 6.5: Illustration of the overlay case in the proof of Theorem 6.3.17.

b Show that the TRS of Example 6.3.4 is strongly closed.

6.14 ***a*** Show that the TRSs of Table 6.1 are not normalizing.

b Construct a normalizing and non-confluent TRS that lacks critical pairs.

6.15 ***a*** Does the TRS $\mathcal{R}$ of Exercise 3.5 have overlays?

b Show that $s \leftarrow\!\!\rtimes\!\!\rightarrow t$ is an overlay if and only if $t \leftarrow\!\!\rtimes\!\!\rightarrow s$ is an overlay.

c Show that all critical pairs of a CS are overlays.

6.16 Let A and B be co-initial proof terms in a left-linear TRS. Show that $\blacktriangle(A, B) = 0$ if and only if $A \perp B$.

6.17 Consider the TRS $\mathcal{R}$ of Example 6.2.1 with $A = \mathsf{f}(\gamma, \mathsf{h}(\beta(\mathsf{b}, \gamma)))$ and $B = \beta(\mathsf{a}, \alpha(\mathsf{g}(\delta), \mathsf{a}))$.

a Compute $\blacktriangle(A, B)$.

b Compute the (proof) terms that decorate Figure 6.4.

6.18 If $p = q$ and $\alpha = \beta$ in the proof of Theorem 6.3.6 then $(*)$ is not a critical peak. How to resolve this situation?

6.19 Consider the proof of Theorem 6.3.6. Let q_β be the unique position in $\mathcal{P}\mathsf{os}(B)$ such that $B = B[\beta(B_1, \ldots, B_m)]_{q_\beta}$ and $\mathsf{src}(B)[\,]_q = \mathsf{src}(B[\,]_{q_\beta})$. Define the substitution

$$\rho_o \;=\; \{x_i \mapsto \mathsf{lhs}(\beta)\langle B_1, \ldots, B_m\rangle_\beta|_{q'p_i} \mid 1 \leqslant i \leqslant n\} \cup \{y_j \mapsto B_j \mid 1 \leqslant j \leqslant m\}$$

Prove the following properties:

1. $\tau_o(y_j)\rho_o = B_j$ for $1 \leqslant j \leqslant m$
2. $D \sqcup (B - \Delta_2) \,/\, \Delta_1 = B[D'\rho_o]_{q_\beta}$
3. $\mathcal{P}\mathsf{os}_L(A \,/\, \Delta_1) \cap \mathcal{P}\mathsf{os}_L(B[D'\rho_o]_{q_\beta})$ contains no positions below p
4. $\blacktriangle(A, B) > \blacktriangle(A \,/\, \Delta_1, D \sqcup (B - \Delta_2) \,/\, \Delta_1)$

6.20 What goes wrong in the proof of Theorem 6.3.6 if the selected overlap (p, q) is not innermost?

6.21 This exercise is about the overlay case in the proof of Theorem 6.3.17, which is illustrated in Figure 6.5. So let $p = q$. By the almost development closedness assumption there exist a term v',

a proof term D' witnessing the multi-step $\mathsf{rhs}(\alpha)\tau_o \mathrel{-\!\!\circ\!\!\to} v'$, and a rewrite sequence $\mathsf{rhs}(\beta)\tau_o \to^* v'$. Define q_β and ρ_o as in Exercise 6.19.

a Prove that the proof term $B[D'\rho]_{q_\beta}$ witnesses a multi-step $t' \mathrel{-\!\!\circ\!\!\to} v'$ for some term v'.

We have $\blacktriangle(A,B) > \blacktriangle(A \,/\, \Delta_1, B[D'\rho]_{q_\beta})$ by Exercise 6.19 **4**. The induction hypothesis yields a term v, a multi-step $t \mathrel{-\!\!\circ\!\!\to} v$, and a rewrite sequence $v' \to^* v$.

b Prove that $u \to^* v'$.

6.22 A rewrite step $a \to b$ in an ARS $\mathcal{A} = \langle A, \to \rangle$ is said to be *reversible* if $b \to^* a$.

a Suppose all rewrite steps of $\mathcal{A}$ are reversible. Show that $\mathcal{A}$ is confluent.

b Use part (a) to show that the TRS $\mathcal{R}$ consisting of the rules

$$\mathsf{f}(x,y) \to \mathsf{f}(y,x) \qquad\qquad \mathsf{f}(\mathsf{f}(x,y),z) \to \mathsf{f}(x,\mathsf{f}(y,z))$$

is confluent.

6.23 A TRS $\mathcal{R}$ is *upside parallel closed* if ${}_p{\overset{>\epsilon}{\leftarrow\!\!\!\times\!\!\!\rightarrow}} \subseteq \to_\epsilon \cup {}_Q{\twoheadleftarrow}$ with $|p| \geqslant |q|$ for all $q \in Q$, and

$$\leftarrow\!\!\!\bowtie\!\!\!\rightarrow \;\subseteq\; (\twoheadrightarrow \cup\, {\overset{=}{\underset{\epsilon}{\leftarrow}}}) \cdot (\twoheadrightarrow_Q \cup\, {\overset{=}{\underset{\epsilon}{\leftarrow}}})$$

with $Q \neq \{\epsilon\}$. If we change the first requirement to ${}_p{\overset{>\epsilon}{\leftarrow\!\!\!\times\!\!\!\rightarrow}} \subseteq \to_\epsilon \cup {\overset{=}{\underset{q}{\leftarrow}}}$ with $q \not> p$ then the TRS is called *outside closed*.

a Consider the TRS $\mathcal{R}$ consisting of the rules

$$\mathsf{f}(\mathsf{g}(x),\mathsf{h}(\mathsf{a})) \to \mathsf{f}(\mathsf{i}(x,x),\mathsf{h}(\mathsf{b})) \qquad \mathsf{b} \to \mathsf{a} \qquad \mathsf{g}(x) \to \mathsf{i}(x,x)$$

Is $\mathcal{R}$ upside parallel closed? Is $\mathcal{R}$ outside closed?

b Construct a left-linear upside parallel closed TRS that is not almost development closed.

c Prove that left-linear upside parallel closed and left-linear outside closed TRSs are confluent.

6.24 ***a*** Is the extension of the TRS of Example 6.3.4 with the single rule $\mathsf{f} \to \mathsf{g}$ confluent?

b Prove the confluence of the TRS consisting of the rewrite rules

$$\begin{array}{lll} \mathsf{f}(\mathsf{g}(x),\mathsf{a}) \to \mathsf{f}(x,\mathsf{a}) & \mathsf{g}(\mathsf{a}) \to \mathsf{b} & \mathsf{f}(\mathsf{b},\mathsf{a}) \to \mathsf{f}(\mathsf{a},\mathsf{a}) \\ \mathsf{f}(\mathsf{g}(\mathsf{g}(x)),y) \to \mathsf{f}(\mathsf{g}(x),y) & \mathsf{g}(\mathsf{b}) \to \mathsf{g}(\mathsf{a}) & \end{array}$$

6.25 A critical pair $s \approx t$ is *deeply* joinable if $u \downarrow v$ for any two reducts u and v of s and t. Construct a linear non-confluent TRS with the property that all its critical pairs are deeply joinable.

6.26 Is every innermost terminating TRS with the property that all critical pairs are innermost joinable confluent?

6.4 Decreasing Diagrams

In this section we present the very useful *decreasing diagrams* technique. In the abstract ARS setting the decreasing diagrams technique subsumes virtually all confluence results presented in Chapter 1. In the next section we use the technique to derive concrete confluence criteria for TRSs.

Definition 6.4.1. An ARS $\mathcal{A}$ is said to be *locally decreasing* if there exist a presentation $\langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ of $\mathcal{A}$ and a well-founded order $>$ on I such that

$$\underset{\alpha}{\leftarrow} \cdot \underset{\beta}{\rightarrow} \;\subseteq\; \underset{\vee\alpha}{\overset{*}{\longleftrightarrow}} \cdot \underset{\beta}{\overset{=}{\rightarrow}} \cdot \underset{\vee\alpha\beta}{\overset{*}{\longleftrightarrow}} \cdot \underset{\alpha}{\overset{=}{\leftarrow}} \cdot \underset{\vee\beta}{\overset{*}{\longleftrightarrow}} \qquad (\mathrm{LD}_{\alpha\beta})$$

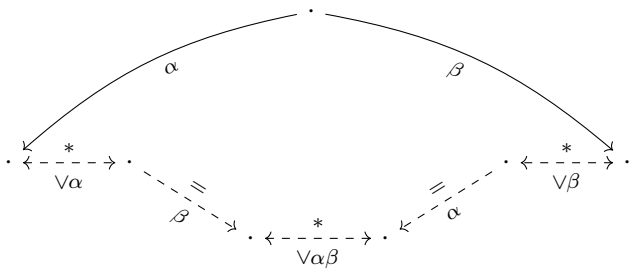


Figure 6.6: Local decreasingness.

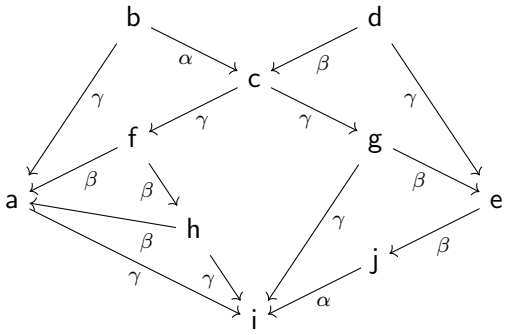


Figure 6.7: A labeled ARS.

for all $\alpha, \beta \in I$, see Figure 6.6. Here $\vee\alpha$ denotes the set $\{\gamma \in I \mid \alpha > \gamma\}$, $\vee\alpha\beta$ denotes $\{\gamma \in I \mid \alpha > \gamma \text{ or } \beta > \gamma\}$, and $\vee\beta$ denotes $\{\gamma \in I \mid \beta > \gamma\}$, cf. Definition 1.4.8.

Example 6.4.2. Consider the ARS $\mathcal{A} = \langle A, \{\alpha, \beta, \gamma\}\rangle$ of Figure 6.7 with well-founded order $\gamma > \alpha$ and $\gamma > \beta$. Consider the peak $\mathsf{a} \,{}_\gamma\!\leftarrow \mathsf{b} \rightarrow_\alpha \mathsf{c}$. We have

$$\xleftarrow[\vee\gamma]{*} \cdot \xrightarrow[\alpha]{=} \cdot \xleftarrow[\vee\gamma\alpha]{*} \cdot \xleftarrow[\gamma]{=} \cdot \xleftarrow[\vee\alpha]{*} \;=\; \xleftarrow[\{\alpha,\beta\}]{*} \cdot \xrightarrow[\alpha]{=} \cdot \xleftarrow[\{\alpha,\beta\}]{*} \cdot \xleftarrow[\gamma]{=} \;=\; \xleftarrow[\{\alpha,\beta\}]{*} \cdot \xleftarrow[\gamma]{=}$$

and the valley $\mathsf{a} \,{}_\beta\!\leftarrow \mathsf{f} \,{}_\gamma\!\leftarrow \mathsf{c}$ fits this pattern. Also the valley $\mathsf{a} \,{}_\beta\!\leftarrow \mathsf{h} \,{}_\beta\!\leftarrow \mathsf{f} \,{}_\gamma\!\leftarrow \mathsf{c}$ fits but $\mathsf{a} \rightarrow_\gamma \mathsf{i} \,{}_\alpha\!\leftarrow \mathsf{j} \,{}_\beta\!\leftarrow \mathsf{e} \,{}_\beta\!\leftarrow \mathsf{g} \,{}_\gamma\!\leftarrow \mathsf{c}$ does not. One readily checks that for every peak there exists a conversion that satisfies the inclusion of Definition 6.4.1 and hence $\mathcal{A}$ is locally decreasing.

We prove that every locally decreasing ARS is confluent. To this end we define a reduction order $>$ on conversions such that if a conversion contains a peak, a smaller conversion is obtained by replacing the peak with the conversion corresponding to the right-hand side of the inclusion of Definition 6.4.1. Since the only relevant parts of conversions are the labels and the directions of the arrows, we associate with each conversion a special *proof string* that captures this information.

Definition 6.4.3. Let $\mathcal{A} = \langle A, \{\rightarrow_\alpha\}_{\alpha\in I}\rangle$ be an ARS. *Proof strings* are strings over the alphabet $I_\pm = I \uplus I_-$, where $I_- = \{\alpha^- \mid \alpha \in I\}$. With every proof string $w \in I_\pm^*$ we

associate a finite multiset consisting of pairs in $I_\pm \times I_\pm^*$ as follows:

$$M(w) = \{(\alpha, u) \mid \alpha \in I \text{ and } w = u\alpha v\} \uplus \{(\alpha, v) \mid \alpha \in I_- \text{ and } w = u\alpha v\}$$

In the construction of $M(w)$ each label α occurring in w is paired with either its prefix (if $\alpha \in I$) or its suffix (if $\alpha \in I_-$). Note that for every pair (u, v) in $M(w)$, v is a proper substring of w. Labels in I_- are used to model left arrows.

Example 6.4.4. Consider again the ARS $\mathcal{A} = \langle A, \{\alpha, \beta, \gamma\}\rangle$ and well-founded order $>$ of Example 6.4.2. The conversion $\mathsf{a} \xleftarrow[\gamma]{} \mathsf{b} \xrightarrow[\alpha]{} \mathsf{c} \xleftarrow[\beta]{} \mathsf{d} \xrightarrow[\gamma]{} \mathsf{e}$ corresponds to the proof string $w_1 = \gamma^-\alpha\beta^-\gamma$. We have

$$M(w_1) = \{(\gamma^-, \alpha\beta^-\gamma), (\alpha, \gamma^-), (\beta^-, \gamma), (\gamma, \gamma^-\alpha\beta^-)\}$$

Replacing the left peak by $\mathsf{a} \xleftarrow[\beta]{} \mathsf{f} \xleftarrow[\gamma]{} \mathsf{c}$ results in the proof string $w_2 = \beta^-\gamma^-\beta^-\gamma$ with

$$M(w_2) = \{(\beta^-, \gamma^-\beta^-\gamma), (\gamma^-, \beta^-\gamma), (\beta^-, \gamma), (\gamma, \beta^-\gamma^-\beta^-)\}$$

Definition 6.4.5. Let $\mathcal{A} = \langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ be an ARS and let $>$ be a proper order on I. We extend $>$ to $I_\pm$ by adding all pairs (α^-, β), (α, β^-), and (α^-, β^-) for $\alpha, \beta \in I$ with $\alpha > \beta$. For every $i \geqslant 0$ we define a relation $\gg^i$ on proof strings inductively as follows: $\gg^0 = \varnothing$ and $u \gg^{i+1} v$ if $M(u)\ ((>, \gg^i)_{\mathsf{lex}})_{\mathsf{mul}}\ M(v)$. The infinite union $\bigcup\{\gg^i \mid i \geqslant 0\}$ is denoted by $\gg$.

Lemma 6.4.6. *Let $\mathcal{A} = \langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ be an* ARS *and $>$ a proper order on I. The relation $\gg$ is a proper order on proof strings.*

Proof Since $\varnothing$ is a proper order and lexicographic product (Theorem A.1.24) and multiset extension (Lemma A.3.7) preserve proper orders, it follows by induction that every relation $\gg^i$ is a proper order. Hence $\gg$ is irreflexive and transitivity follows if we can show $\gg^i \subseteq \gg^j$ whenever $i \leqslant j$. The latter follows by a straightforward induction on $j - i$, using the monotonicity of lexicographic product (Exercise A.14) and multiset extension (Exercise A.27). □

Example 6.4.7. Continuing Example 6.4.4, we claim $w_1 \gg w_2$. We have

$$(\gamma^-, \alpha\beta^-\gamma)\ (>, \gg^0)_{\mathsf{lex}}\ (\beta^-, \gamma^-\beta^-\gamma)$$

because $\gamma^- > \beta^-$. Furthermore, $\alpha\beta^- \gg^1 \beta^-$ because

$$\{(\alpha, \epsilon), (\beta^-, \epsilon)\}\ ((>, \gg^0)_{\mathsf{lex}})_{\mathsf{mul}}\ \{(\beta^-, \epsilon)\}$$

and thus also $(\gamma, \alpha\beta^-)\ (>, \gg^1)_{\mathsf{lex}}\ (\gamma, \beta^-)$. The latter implies

$$\{(\alpha, \epsilon), (\beta^-, \gamma), (\gamma, \alpha\beta^-)\}\ ((>, \gg^1)_{\mathsf{lex}})_{\mathsf{mul}}\ \{(\beta^-, \gamma), (\gamma, \beta^-)\}$$

and hence $\alpha\beta^-\gamma \gg^2 \beta^-\gamma$ and $(\gamma^-, \alpha\beta^-\gamma)\ (>, \gg^2)_{\mathsf{lex}}\ (\gamma^-, \beta^-\gamma)$. In the same fashion one obtains $(\gamma, \gamma^-\alpha\beta^-)\ (>, \gg^2)_{\mathsf{lex}}\ (\gamma, \beta^-\gamma^-\beta^-)$. Hence $M(w_1)\ ((>, \gg^2)_{\mathsf{lex}})_{\mathsf{mul}}\ M(w_2)$ and

therefore $w_1 \gg^3 w_2$.

The next lemma states that $\gg$ is a rewrite order on proof strings. So it is closed under concatenation, which can be used to simplify the calculations in the preceding example.

Lemma 6.4.8. *Let $\mathcal{A} = \langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ be an* ARS *and $>$ a proper order on I. The relation $\gg$ is a rewrite order on proof strings.*

Proof From Lemma 6.4.6 we know that $\gg$ is a proper order. We show that $\gg$ is closed under contexts (left-concatenation). Let $u \gg v$ and consider an arbitrary string $w \in I_\pm^*$. We show $wu \gg wv$ by induction on the length of u. If $u = \epsilon$ then there is nothing to show (since $u \gg v$ does not hold). So suppose $u \neq \epsilon$. We use a second induction on the length of w. If $w = \epsilon$, the result is obvious. Let $w = w'\alpha$. We show $\alpha u \gg \alpha v$ by distinguishing two cases.

1 If $\alpha \in I$ then

$$\begin{aligned} M(\alpha u) &= \{(\alpha,\epsilon)\} \uplus \{(\beta,\alpha x) \mid (\beta,x) \in M(u) \text{ and } \beta \in I\} \\ &\quad \uplus \{(\beta,x) \mid (\beta,x) \in M(u) \text{ and } \beta \in I_-\} \\ M(\alpha v) &= \{(\alpha,\epsilon)\} \uplus \{(\beta,\alpha x) \mid (\beta,x) \in M(v) \text{ and } \beta \in I\} \\ &\quad \uplus \{(\beta,x) \mid (\beta,x) \in M(v) \text{ and } \beta \in I_-\} \end{aligned}$$

We claim that the comparisons between the elements of $M(u)$ and $M(v)$ carry over to the respective elements of $M(\alpha u)$ and $M(\alpha v)$. Let $(\beta,x) \in M(u)$ and $(\gamma,y) \in M(v)$ with $(\beta,x)\ (>,\gg)_{\mathsf{lex}}\ (\gamma,y)$. We consider four cases. If $\beta \in I_-$ and $\gamma \in I$ then $(\beta,x) \in M(\alpha u)$ and $(\gamma,\alpha y) \in M(\alpha v)$. Since $\beta \neq \gamma$, we must have $\beta > \gamma$ and thus also $(\beta,x)\ (>,\gg)_{\mathsf{lex}}\ (\gamma,\alpha y)$. The case $\beta \in I$ and $\gamma \in I_-$ is dealt with in exactly the same way. If $\beta,\gamma \in I_-$ then $(\beta,x) \in M(\alpha u)$ and $(\gamma,y) \in M(\alpha v)$ and so there is nothing to prove. In the remaining case we have $\beta,\gamma \in I$ and thus $(\beta,\alpha x) \in M(\alpha u)$ and $(\gamma,\alpha y) \in M(\alpha v)$. If $\beta = \gamma$ then we use the first induction hypothesis to obtain $\alpha x \gg \alpha y$.

2 If $\alpha \in I_-$ then $M(\alpha u)$ contains (α,u) instead of (α,ϵ) and $M(\alpha v)$ contains (α,v) instead of (α,ϵ), so we need the additional observation $(\alpha,u)\ (>,\gg)_{\mathsf{lex}}\ (\alpha,v)$, which follows from the assumption $u \gg v$.

So $\alpha u \gg \alpha v$ and we obtain the desired $wu \gg wv$ from the induction hypothesis. Closure under substitutions (right-concatenation) of $\gg$ is obtained in the same way. It follows that $\gg$ is a rewrite order on proof strings. □

We now show that $\gg$ is a reduction order whenever $>$ is well-founded. In the proof we make use of (a special version of) Higman's Lemma.

Definition 6.4.9. Let $>$ be a relation on a set A. The SRS $\mathcal{E}\mathsf{mb}(>)$ consists of all rules $a \to \epsilon$ for all $a \in A$ and $a \to b$ for all $a,b \in A$ with $a > b$. We write $>_{\mathsf{emb}}$ for the relation $\to^+_{\mathcal{E}\mathsf{mb}(>)}$ on A^*.

Higman's Lemma (Special Version). *If $>$ is a well-order on a set A then every proper order on A^* that contains $>_{\mathsf{emb}}$ is well-founded.*

The proof of Higman's Lemma can be found in Appendix B.4.

Lemma 6.4.10. *Let $\mathcal{A} = \langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ be an* ARS *and $>$ a well-founded order on I. The relation $\gg$ is a reduction order on proof strings.*

Proof We extend the well-founded order $>$ on $I_\pm$ to a well-order $>'$ (which is possible using Zorn's Lemma). If we show $>'_{\mathsf{emb}} \subseteq \gg'$ then $\gg'$ is well-founded by Higman's Lemma. Since $\gg'$ is a rewrite order by Lemma 6.4.8, it suffices to show $\mathcal{E}\mathsf{mb}(>') \subseteq \gg'$. Let $\alpha \in I_\pm$. We have $M(\alpha) = \{(\alpha, \epsilon)\}$. Since $M(\epsilon)$ is the empty multiset, we obtain $\alpha \gg' \epsilon$. Next suppose $\alpha >' \beta$. We have $(\alpha, \epsilon)\ (>', \gg')_{\mathsf{lex}}\ (\beta, \epsilon)$ and thus $\alpha \gg' \beta$. It remains to show $\gg \subseteq \gg'$ in order to conclude that $\gg$ is well-founded. This follows by a straightforward induction proof (Exercise 6.31). □

Theorem 6.4.11. *Every locally decreasing* ARS *is confluent.*

Proof Let $\mathcal{A} = \langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ be an ARS that is locally decreasing with respect to a well-founded order $>$ on I. Consider an arbitrary labeled conversion in $\mathcal{A}$ between elements a and b. Let z be the proof string corresponding to this conversion. We prove $a \downarrow b$ by induction on z with respect to $\gg$. If the conversion is already a valley, there is nothing to show. Otherwise the conversion contains a (local) peak and z must thus have the form $z_1\alpha^-\beta z_2$. Using the local decreasingness assumption, the given conversion is transformed into one with associated proof string $z_1 uvwxyz_2$ where $u \in (\vee_\pm\alpha)^*$, $v \in \{\epsilon, \beta\}$, $w \in (\vee_\pm\alpha\beta)^*$, $x \in \{\epsilon, \alpha^-\}$, and $y \in (\vee_\pm\beta)^*$. Here $\vee_\pm\alpha$ denotes the set $\{\gamma \in I_\pm \mid \alpha > \gamma\}$ and $\vee_\pm\alpha\beta$ stands for $\vee_\pm\alpha \cup \vee_\pm\beta$. We show $\alpha^-\beta \gg uvwxy$. We have $M(\alpha^-\beta) = \{(\alpha^-, \beta), (\beta, \alpha^-)\}$. With at most two exceptions, every pair in $M(uvwxy)$ is of the form (γ, z') with $\alpha > \gamma$ or $\beta > \gamma$. In the former case we have $(\alpha^-, \beta)\ (>, \gg)_{\mathsf{lex}}\ (\gamma, z')$ and in the latter case we have $(\beta, \alpha^-)\ (>, \gg)_{\mathsf{lex}}\ (\gamma, z')$. Note that also $\alpha^- \gg u$ and $\beta \gg y$. The two exceptions are (β, u) and (α^-, y). We obtain $(\beta, \alpha^-)\ (>, \gg)_{\mathsf{lex}}\ (\beta, u)$ from $\alpha^- \gg u$ and $(\alpha^-, \beta)\ (>, \gg)_{\mathsf{lex}}\ (\alpha^-, y)$ from $\beta \gg y$. It follows that $M(\alpha^-\beta)\ ((>, \gg)_{\mathsf{lex}})_{\mathsf{mul}}\ M(uvwxy)$ and thus $\alpha^-\beta \gg uvwxy$. Because $\gg$ is a rewrite relation, we obtain $z = z_1\alpha^-\beta z_2 \gg z_1 uvwxyz_2$. According to the induction hypothesis, the conversion between a and b associated with $z_1 uvwxyz_2$ is transformed into a valley. □

The applicability of Theorem 6.4.11 for proving confluence of a given ARS $\mathcal{A} = \langle A, \to\rangle$ depends on the ability to find a suitable presentation $\langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ of $\mathcal{A}$, a well-founded order $>$ on the set I of its labels such that $\mathrm{LD}_{\alpha\beta}$ holds for all $\alpha, \beta \in I$. Here we show how this can be done for Newman's Lemma. In the exercises several other consequences of Theorem 6.4.11 are presented.

Third Proof of Newman's Lemma Let $\mathcal{A} = \langle A, \to\rangle$ be a terminating and locally confluent ARS. Define for every element $a \in A$ the relation $\to_a$ as the restriction of $\to$ to $\{a\} \times A$. Since $\to = \cup_{a\in A} \to_a$, $\langle A, \{\to_a\}_{a\in A}\rangle$ is a presentation of $\mathcal{A}$. Note that every rewrite step in $\mathcal{A}$ is labeled with its starting element. We define $a > b$ if and only if $a \to^+ b$. Because $\mathcal{A}$ is terminating, $>$ is a well-founded order on A. The important observation is that every rewrite sequence of the form $a \to b \to^* c$ translates to $a \to_a b \to^*_{\vee a} c$ since $a \to^+ d$ for every element in the rewrite sequence from b to c, and hence local confluence of $a \in A$ amounts to LD_{aa}, see Figure 6.8. Since LD_{ab} holds vacuously for $a \neq b$, the ARS $\mathcal{A}$ is locally decreasing and hence confluent by Theorem 6.4.11. □

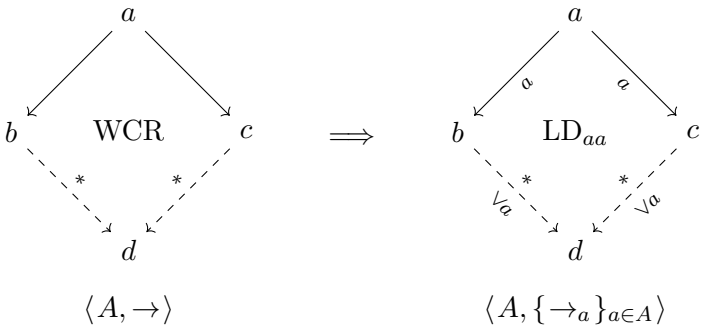


Figure 6.8: Third proof of Newman's Lemma.

We conclude this section by presenting a variant of Theorem 6.4.11 which is useful for the results in the next section. Here $>$ and $\geqslant$ are said to be compatible if the inclusion $\geqslant \cdot > \cdot \geqslant \subseteq >$ holds.

Theorem 6.4.12. *An* ARS $\mathcal{A}$ *is confluent if there exist a presentation* $\langle A, \{\to_\alpha\}_{\alpha\in I}\rangle$ *of* $\mathcal{A}$ *and a well-founded order* $>$ *and a compatible preorder* $\geqslant$ *on* I *such that*

$$\xleftarrow[\alpha]{} \cdot \xrightarrow[\beta]{} \subseteq \xleftarrow[\curlyvee\alpha]{*} \cdot \xrightarrow[\curlyvee/\beta]{=} \cdot \xleftarrow[\curlyvee\alpha\beta]{*} \cdot \xleftarrow[\curlyvee/\alpha]{=} \cdot \xleftarrow[\curlyvee\beta]{*}$$

for all $\alpha, \beta \in I$.

Exercises

6.27 Consider the ARS $\mathcal{A}$ of Figure 6.7. Which of the following well-founded orders on $\{\alpha, \beta, \gamma\}$ show that $\mathcal{A}$ is locally decreasing?

a $\alpha > \beta > \gamma$

b $\beta > \gamma > \alpha$

6.28 Show that every peak decreasing ARS is locally decreasing.

6.29 ***a*** Show that Lemma 1.4.4 is a corollary of Theorem 6.4.11.

b Show that Lemma 1.4.7 is a corollary of Theorem 6.4.11.

6.30 Let $\langle A, \{\alpha, \beta\}\rangle$ be an ARS such that $\alpha = \{\to_i \mid i \in I\}$ and $\beta = \{\to_j \mid j \in J\}$ for some index sets I and J. Suppose $>$ is a well-founded order $>$ on $I \cup J$ such that

$$\xleftarrow[\alpha]{} \cdot \xrightarrow[\beta]{} \subseteq \xleftarrow[\curlyvee\alpha]{*} \cdot \xrightarrow[\beta]{=} \cdot \xleftarrow[\curlyvee\alpha\beta]{*} \cdot \xleftarrow[\alpha]{=} \cdot \xleftarrow[\curlyvee\beta]{*}$$

for all $\alpha \in I$ and $\beta \in J$. Here $\leftrightharpoons_K$ denotes the union of all ${}_k\leftarrow$ with $k \in I \cap K$ and all $\to_k$ with $k \in J \cap K$.

a Prove that α and β commute.

b Show that the implication SCR $\implies$ CR (Exercise 1.17) follows from the result of part (a).

6.31 Show $\gg \subseteq \gg'$ whenever $> \subseteq >'$.

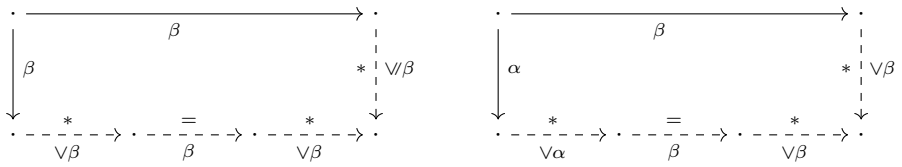
6.32 Consider the reduction order $\gg$ on proof strings of Definition 6.4.5.

a Consider the ARS $\mathcal{A}$ and well-founded order $>$ of Example 6.4.2. Compare the proof strings $\alpha\beta$, $\gamma^-\alpha\beta^-\gamma$, $\gamma\alpha^-\beta^-\gamma^-$, and $\beta^-\beta\gamma\gamma^-\gamma^-\beta^-$ with $\gg$.

b Prove $u^{-1} \gg v^{-1}$ whenever $u \gg v$. Here the inverse operation $(\cdot)^{-1}$ is inductively defined by $\epsilon^{-1} = \epsilon$ and

$$(\alpha x)^{-1} = \begin{cases} x^{-1}\alpha^- & \text{if } \alpha \in I \\ x^{-1}\alpha & \text{if } \alpha \in I_- \end{cases}$$

6.33 Let $\mathcal{A} = \langle A, \{\to_\alpha\}_{\alpha \in I} \rangle$ be an ARS and suppose $>$ is a well-order on I such that



for all $\alpha, \beta \in I$ with $\alpha < \beta$.

a Show that $\mathcal{A}$ is confluent.

b Why do we require $>$ to be a *total* well-founded order?

6.34 Let $\mathcal{A} = \langle A, \{\alpha, \beta\} \rangle$ be an ARS such that $\mathsf{NF}(\alpha) \subseteq \mathsf{NF}(\beta)$, ${}_\alpha{\leftarrow} \cdot \to_\beta \subseteq (\to_\beta \cup \to_\alpha^*) \cdot {}_\alpha^*{\leftarrow}$, and α is complete. Show that $\mathcal{A}$ is confluent.

6.35 Prove that every semi-complete ARS is locally decreasing.

6.36 Let $\mathcal{A} = \langle A, \to \rangle$ be a confluent ARS such that A is countable. Prove that $\mathcal{A}$ is locally decreasing using only two labels.

6.37 Prove Theorem 6.4.12.

6.5 Transformation Techniques

We start this section with a few transformation techniques to ease the task of establishing confluence. The first one is based on a simple criterion that allows one to add rewrite rules. By choosing suitable rules, (dis)proving confluence may become easier.

Lemma 6.5.1. *If $\ell \to_\mathcal{R}^* r$ for every rule $\ell \to r$ from $\mathcal{S}$ then $\to_\mathcal{R}^* = \to_{\mathcal{R} \cup \mathcal{S}}^*$.*

Proof The inclusion $\to_\mathcal{R}^* \subseteq \to_{\mathcal{R} \cup \mathcal{S}}^*$ is obvious. For the reverse inclusion it suffices to show $s \to_\mathcal{R}^* t$ whenever $s \to_{\mathcal{R} \cup \mathcal{S}} t$. The latter ensures the existence of a position p in s, a rewrite rule $\ell \to r$ in $\mathcal{R} \cup \mathcal{S}$, and a substitution σ such that $s|_p = \ell\sigma$ and $t = s[r\sigma]_p$. We obtain $\ell \to_\mathcal{R}^* r$ from the assumption of the lemma. Closure (of $\to_\mathcal{R}^*$) under contexts and substitutions yields the desired $s \to_\mathcal{R}^* t$. □

Corollary 6.5.2. *If $\ell \to_\mathcal{R}^* r$ for every rule $\ell \to r$ from $\mathcal{S}$ then $\mathcal{R}$ is confluent if and only if $\mathcal{R} \cup \mathcal{S}$ is confluent.*

Proof We obtain $\to_\mathcal{R}^* = \to_{\mathcal{R} \cup \mathcal{S}}^*$ from the preceding lemma. Hence also $\downarrow_\mathcal{R} = \downarrow_{\mathcal{R} \cup \mathcal{S}}$ and $\uparrow_\mathcal{R} = \uparrow_{\mathcal{R} \cup \mathcal{S}}$. Therefore $\uparrow_\mathcal{R} \subseteq \downarrow_\mathcal{R}$ if and only if $\uparrow_{\mathcal{R} \cup \mathcal{S}} \subseteq \downarrow_{\mathcal{R} \cup \mathcal{S}}$. □

Definition 6.5.3. A rule $\ell \to r \in \mathcal{R}$ is *redundant* if $\ell \to_{\mathcal{R} \setminus \{\ell \to r\}}^* r$.

According to Corollary 6.5.2 adding redundant rules does not affect confluence.

Example 6.5.4. The TRS $\mathcal{R}$ consisting of the rules $\mathsf{f}(\mathsf{f}(x)) \to x$ and $\mathsf{f}(x) \to \mathsf{f}(\mathsf{f}(x))$ admits two non-trivial critical pairs:

$$\mathsf{f}(\mathsf{f}(\mathsf{f}(x))) \approx x \qquad\qquad x \approx \mathsf{f}(\mathsf{f}(\mathsf{f}(x)))$$

Because $\mathsf{f}(\mathsf{f}(\mathsf{f}(x))) \to \mathsf{f}(x) \to \mathsf{f}(\mathsf{f}(x)) \to x$, the critical pairs are joinable but not by a multi-step. Hence Theorem 6.3.17 is not applicable. Adding the rewrite rule $\mathsf{f}(x) \to x$ makes Theorem 6.3.17 applicable, as shown in Example 6.3.18. Because $\mathsf{f}(x) \to_{\mathcal{R}} \mathsf{f}(\mathsf{f}(x)) \to_{\mathcal{R}} x$, the added rule is redundant and hence $\mathcal{R}$ is confluent by Corollary 6.5.2.

Next we present a simple criterion that allows one to remove rewrite rules.

Lemma 6.5.5. *If $\ell \leftrightarrow^*_{\mathcal{R}} r$ for every rule $\ell \to r$ from $\mathcal{S}$ then $\leftrightarrow^*_{\mathcal{R}\cup\mathcal{S}} = \leftrightarrow^*_{\mathcal{R}}$.*

Proof The inclusion $\leftrightarrow^*_{\mathcal{R}} \subseteq \leftrightarrow^*_{\mathcal{R}\cup\mathcal{S}}$ is obvious. For the reverse direction it suffices to show $s \leftrightarrow^*_{\mathcal{R}} t$ whenever $s \to_{\mathcal{R}\cup\mathcal{S}} t$. The latter ensures the existence of a position p in s, a rewrite rule $\ell \to r$ in $\mathcal{R}\cup\mathcal{S}$, and a substitution σ such that $s|_p = \ell\sigma$ and $t = s[r\sigma]_p$. We obtain $\ell \leftrightarrow^*_{\mathcal{R}} r$ from the assumption of the lemma. Closure (of $\leftrightarrow^*_{\mathcal{R}}$) under contexts and substitutions yields the desired $s \leftrightarrow^*_{\mathcal{R}} t$. □

Corollary 6.5.6. *If $\mathcal{R}$ is confluent and $\ell \leftrightarrow^*_{\mathcal{R}} r$ for every rule $\ell \to r$ from $\mathcal{S}$ then $\mathcal{R}\cup\mathcal{S}$ is confluent.*

Proof From the preceding lemma and the confluence of $\mathcal{R}$ we obtain

$$\leftrightarrow^*_{\mathcal{R}\cup\mathcal{S}} \;=\; \leftrightarrow^*_{\mathcal{R}} \;\subseteq\; {\downarrow_{\mathcal{R}}} \;\subseteq\; {\downarrow_{\mathcal{R}\cup\mathcal{S}}}$$

Hence $\mathcal{R}\cup\mathcal{S}$ is confluent. □

Example 6.5.7. Consider the TRS consisting of the four rewrite rules

$$\mathsf{f}(x,x) \to \mathsf{f}(\mathsf{g}(x),\mathsf{g}(x)) \qquad \mathsf{f}(x,y) \to \mathsf{f}(\mathsf{h}(x),\mathsf{h}(y)) \qquad \mathsf{g}(x) \to \mathsf{i}(x) \qquad \mathsf{h}(x) \to \mathsf{i}(x)$$

Because of the conversion

$$\mathsf{f}(x,x) \to \mathsf{f}(\mathsf{h}(x),\mathsf{h}(x)) \to \mathsf{f}(\mathsf{i}(x),\mathsf{h}(x)) \to \mathsf{f}(\mathsf{i}(x),\mathsf{i}(x)) \leftarrow \mathsf{f}(\mathsf{g}(x),\mathsf{i}(x)) \leftarrow \mathsf{f}(\mathsf{g}(x),\mathsf{g}(x))$$

we can remove the first rule. Since the resulting TRS is orthogonal, the original TRS is confluent by Corollary 6.5.6.

It can also be beneficial to both add and remove rules. In particular adding a redundant rule can help with removing other, problematic rules, as shown in the following example.

Example 6.5.8. Consider the TRS consisting of the three rewrite rules

$$\mathsf{f}(x,y) \to \mathsf{f}(\mathsf{g}(x),\mathsf{g}(x)) \qquad\qquad \mathsf{f}(x,x) \to \mathsf{a} \qquad\qquad \mathsf{g}(x) \to x$$

After adding the rule $\mathsf{f}(x,y) \to \mathsf{a}$, which is justified since $\mathsf{f}(x,y) \to \mathsf{f}(\mathsf{g}(x),\mathsf{g}(x)) \to \mathsf{a}$, we can remove the first two original rules, due to the following conversions:

$$\mathsf{f}(x,y) \to \mathsf{a} \leftarrow \mathsf{f}(\mathsf{g}(x),\mathsf{g}(x)) \qquad\qquad \mathsf{f}(x,x) \to \mathsf{a}$$

The resulting TRS is orthogonal and hence the original TRS is confluent by Corollaries 6.5.2 and 6.5.6.

In the remainder of this section we use decreasing diagrams to derive a concrete confluence technique for linear TRSs that is based on labeling of steps. The following example conveys the idea.

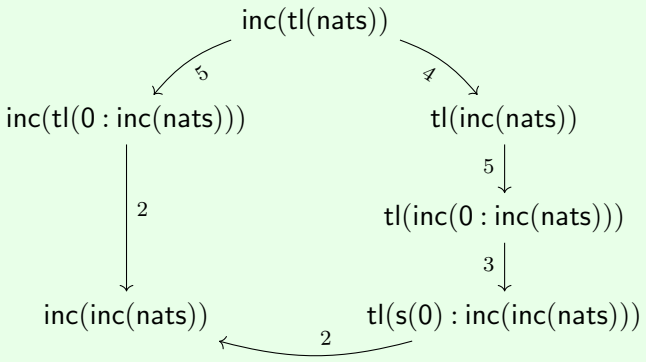
Example 6.5.9. Consider the TRS $\mathcal{R}$ consisting of the five rewrite rules

$$\begin{array}{lll} \mathsf{hd}(x:y) \xrightarrow[1]{} x & \mathsf{inc}(x:y) \xrightarrow[3]{} \mathsf{s}(x):\mathsf{inc}(y) & \mathsf{nats} \xrightarrow[5]{} 0:\mathsf{inc}(\mathsf{nats}) \\ \mathsf{tl}(x:y) \xrightarrow[2]{} y & \mathsf{inc}(\mathsf{tl}(\mathsf{nats})) \xrightarrow[4]{} \mathsf{tl}(\mathsf{inc}(\mathsf{nats})) & \end{array}$$

where the number of each rewrite rule is written below the arrow. The single critical peak $\mathsf{inc}(\mathsf{tl}(0:\mathsf{inc}(\mathsf{nats}))) \leftarrow \mathsf{inc}(\mathsf{tl}(\mathsf{nats})) \to \mathsf{tl}(\mathsf{inc}(\mathsf{nats}))$ is joinable:

$$\begin{array}{rl} & \mathsf{inc}(\mathsf{tl}(0:\mathsf{inc}(\mathsf{nats}))) \to \mathsf{inc}(\mathsf{inc}(\mathsf{nats})) \\ \mathsf{tl}(\mathsf{inc}(\mathsf{nats})) \to \mathsf{tl}(\mathsf{inc}(0:\mathsf{inc}(\mathsf{nats}))) & \to \mathsf{tl}(\mathsf{s}(0):\mathsf{inc}(\mathsf{inc}(\mathsf{nats}))) \to \mathsf{inc}(\mathsf{inc}(\mathsf{nats})) \end{array}$$

If we label rewrite steps with the number of the applied rewrite rule, we obtain a decreasing diagram by letting $4 > 3, 2$:



Because the TRS in Example 6.5.9 is linear, confluence is obtained from Corollary 6.5.15 below.

Notation. We write $\Lambda_{\mathcal{R}}$ for the set of rewrite steps of a TRS $\mathcal{R}$.

The different kinds of labelings that we consider in this section adhere to the following definition.

Definition 6.5.10. A *labeling* $L = (l, \geqslant, >)$ for a TRS $\mathcal{R}$ consists of a labeling function $l\colon \Lambda_{\mathcal{R}} \to W$ for some set W of labels equipped with a preorder $\geqslant$ and a compatible well-founded order $>$ such that

$$\begin{aligned} l(s \to t) \geqslant l(u \to v) &\implies l(C[s\sigma] \to C[t\sigma]) \geqslant l(C[u\sigma] \to C[v\sigma]) \\ l(s \to t) > l(u \to v) &\implies l(C[s\sigma] \to C[t\sigma]) > l(C[u\sigma] \to C[v\sigma]) \end{aligned}$$

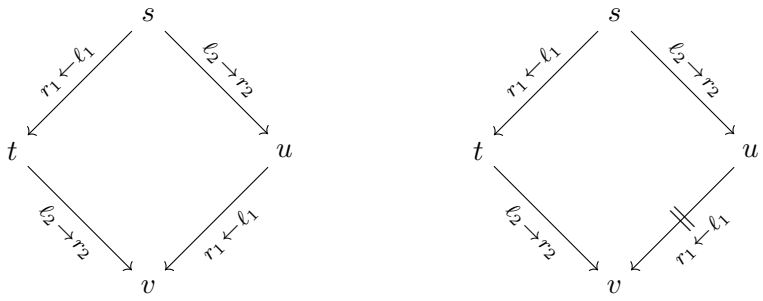


Figure 6.9: Non-critical diagrams.

for all rewrite steps $s \to t, u \to v \in \Lambda_{\mathcal{R}}$, contexts C, and substitutions σ.

The labeling used in Example 6.5.9 is a special case.

Definition 6.5.11. The *rule labeling* $\mathsf{RL}_i = (l^i_{\mathsf{rl}}, \geqslant_{\mathbb{N}}, >_{\mathbb{N}})$ is parameterized by an index function $i\colon \mathcal{R} \to \mathbb{N}$ and uses the labeling function

$$l^i_{\mathsf{rl}}(s \xrightarrow[\ell \to r]{} t) = i(\ell \to r)$$

that labels every rewrite step by the index of the employed rewrite rule. We write RL if the index function can be inferred from the context or is irrelevant.

It is easy to show that the rule labeling fulfills the conditions in Definition 6.5.10. To obtain an effective technique for proving confluence, we require that critical peaks can be joined decreasingly (according to the diagram of Figure 6.6 or Theorem 6.4.12; Definition 6.5.12 contains a formal definition) with respect to some labeling and demand that non-critical peaks can be ignored. As shown in Figure 5.1, there are two kinds of non-critical peaks, corresponding to parallel redexes and variable overlaps. These can can always be joined. Since we deal with left-linear TRSs, the variable overlap case in Figure 5.1(ii) can be simplified, cf. the right diagram in Figure 6.9.

Definition 6.5.12. Let L be a labeling for a TRS $\mathcal{R}$. The labeling $l(C)$ of an arbitrary conversion C in $\mathcal{R}$ is defined as the union of the labels of its individual steps. Here we ignore the direction of the steps, so $l(t \leftarrow s) = l(s \to t)$. A local peak $t \leftarrow s \to u$ in $\mathcal{R}$ is *L-decreasing* if there exists a labeled conversion

$$t \xleftarrow[\vee\alpha]{*} \cdot \xrightarrow[\vee\beta]{=} \cdot \xleftarrow[\vee\alpha\beta]{*} \cdot \xleftarrow[\vee\alpha]{=} \cdot \xleftarrow[\vee\beta]{*} u$$

with $\alpha = l(t \leftarrow s)$ and $\beta = l(s \to u)$. We call L *compatible* with $\mathcal{R}$ if all parallel and variable diagrams of $\mathcal{R}$ are L-decreasing.

Note that we speak of *diagrams* rather than *peaks* in the definition of compatibility since we demand that the required conversions must be of the shape given in the diagrams in Figure 6.9. So the only choice is the order of the steps in a serialization of the parallel step $u \mathrel{+\!\!\!\to} v$, which is of no concern for the rule labeling.

Lemma 6.5.13. *The rule labeling is a compatible labeling for linear* TRS*s.*

Proof Parallel diagrams are clearly RL-decreasing. Variable diagrams in linear TRSs have the shape $t \underset{a}{\leftarrow} s \underset{b}{\rightarrow} u$ with $t \underset{b}{\rightarrow} v \underset{a}{\leftarrow} u$ or $t \underset{b}{\rightarrow} u$, and therefore are RL-decreasing, too. □

Theorem 6.5.14. *A* TRS *is confluent if every critical peak is L-decreasing for a compatible labeling L.*

Proof Let $\mathcal{R}$ be a TRS and $L = (\ell, \geqslant, >)$ a compatible labeling with $\ell\colon \Lambda_{\mathcal{R}} \to W$ such that every critical peak is L-decreasing. The closure conditions in Definition 6.5.10 ensure that every local peak consisting of overlapping steps is L-decreasing. The other local peaks are taken care of by the definition of compatible labeling. It follows that the ARS $\langle \mathcal{T}, \{\rightarrow_\alpha |\ \alpha \in W\}\rangle$ with $\rightarrow_\alpha = \{(s,t) \mid \ell(s \to t) = \alpha\}$ is a locally decreasing presentation of the ARS $\mathcal{A} = \langle \mathcal{T}, \rightarrow_{\mathcal{R}}\rangle$. We obtain the confluence of $\mathcal{A}$ and thus of $\mathcal{R}$ from Theorem 6.4.12. □

Combining Lemma 6.5.13 with Theorem 6.5.14 yields the following concrete result.

Corollary 6.5.15. *A linear* TRS *is confluent if every critical peak is* RL*-decreasing.*

The following example shows that linearity in Corollary 6.5.15 cannot be weakened to left-linearity.

Example 6.5.16. The left-linear TRS $\mathcal{R}$ consisting of the four rewrite rules

$$\mathsf{f}(\mathsf{a},\mathsf{a}) \underset{1}{\rightarrow} \mathsf{c} \qquad \mathsf{f}(\mathsf{b},x) \underset{2}{\rightarrow} \mathsf{f}(x,x) \qquad \mathsf{f}(x,\mathsf{b}) \underset{3}{\rightarrow} \mathsf{f}(x,x) \qquad \mathsf{a} \underset{4}{\rightarrow} \mathsf{b}$$

admits three (modulo symmetry) critical pairs:

$$\mathsf{f}(\mathsf{a},\mathsf{b}) \underset{4}{\leftarrow} \mathsf{f}(\mathsf{a},\mathsf{a}) \underset{1}{\rightarrow} \mathsf{c} \qquad \mathsf{f}(\mathsf{b},\mathsf{a}) \underset{4}{\leftarrow} \mathsf{f}(\mathsf{a},\mathsf{a}) \underset{1}{\rightarrow} \mathsf{c} \qquad \mathsf{f}(\mathsf{b},\mathsf{b}) \underset{2}{\leftarrow} \mathsf{f}(\mathsf{b},\mathsf{b}) \underset{3}{\rightarrow} \mathsf{f}(\mathsf{b},\mathsf{b})$$

Since

$$\mathsf{f}(\mathsf{a},\mathsf{b}) \underset{3}{\rightarrow} \mathsf{f}(\mathsf{a},\mathsf{a}) \underset{1}{\rightarrow} \mathsf{c} \qquad\qquad \mathsf{f}(\mathsf{b},\mathsf{a}) \underset{2}{\rightarrow} \mathsf{f}(\mathsf{a},\mathsf{a}) \underset{1}{\rightarrow} \mathsf{c}$$

it follows that the critical peaks are RL-decreasing by taking the order $4 > 2, 3$. Nevertheless, the conversion

$$\mathsf{f}(\mathsf{b},\mathsf{b}) \leftarrow \mathsf{f}(\mathsf{b},\mathsf{a}) \leftarrow \mathsf{f}(\mathsf{a},\mathsf{a}) \rightarrow \mathsf{c}$$

reveals that $\mathcal{R}$ is not confluent. The problem is that the variable diagram

$$\mathsf{f}(\mathsf{b},\mathsf{b}) \leftarrow \mathsf{f}(\mathsf{b},\mathsf{a}) \rightarrow \mathsf{f}(\mathsf{a},\mathsf{a}) \quad \text{with} \quad \mathsf{f}(\mathsf{b},\mathsf{b}) \nleftrightarrow \mathsf{f}(\mathsf{a},\mathsf{a})$$

is not RL-decreasing.

In a later chapter we present labeling results that do not demand right-linearty.

Exercises

6.38 Use the redundant rules technique to show the confluence of the TRS of Example 6.5.9.

6.39 Prove that the TRS consisting of the rewrite rules

$$\mathsf{f}(x) \to \mathsf{g}(x, \mathsf{f}(x)) \qquad\qquad \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(x)))) \to \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{g}(x, \mathsf{f}(x)))))$$

is confluent.

6.40 Prove that the TRS consisting of the rewrite rules

$$\mathsf{f}(x, \mathsf{b}, y) \to \mathsf{f}(x, y, y) \qquad \mathsf{f}(\mathsf{c}, \mathsf{a}, \mathsf{a}) \to \mathsf{f}(\mathsf{d}, \mathsf{a}, \mathsf{a}) \qquad \mathsf{f}(\mathsf{d}, \mathsf{a}, \mathsf{a}) \to \mathsf{f}(\mathsf{c}, \mathsf{a}, \mathsf{a})$$
$$\mathsf{a} \to \mathsf{b} \qquad\qquad \mathsf{c} \to \mathsf{d} \qquad\qquad \mathsf{d} \to \mathsf{c}$$

is confluent.

6.41 Prove that the TRS consisting of the rewrite rules

$$\mathsf{a} \to \mathsf{f}(\mathsf{b}) \qquad \mathsf{a} \to \mathsf{f}(\mathsf{c}) \qquad \mathsf{b} \to \mathsf{g}(\mathsf{b}) \qquad \mathsf{c} \to \mathsf{g}(\mathsf{g}(\mathsf{b})) \qquad \mathsf{f}(x) \to \mathsf{h}(x, x)$$

is confluent.

6.42 Does the converse of Corollary 6.5.6 hold?

6.43 Prove that is undecidable whether the confluence of a locally confluent linear TRS can be shown by rule labeling.

Bibliographic Notes

The TRS $\mathcal{R}_1$ in Table 6.1 is from Huet [57] whereas $\mathcal{R}_2$ is attributed to Barendregt in [57]. Theorem 6.3.2 is from [57]. Proof terms were introduced by van Oostrom and de Vrijer to study equivalence of reductions in [133] and [126, Chapter 8]. Example 6.3.4 is from [106], simplifying an earlier counterexample attributed to Lévy in [57]. Theorem 6.3.6, due to van Oostrom [101], extends the earlier result of Huet [57] stated in Corollary 6.3.14. There are several long-standing open problems related to these results. It is unknown whether any of the following conditions is sufficient for the confluence of left-linear TRSs: $\leftarrow\!\!\rtimes\!\!\rightarrow \subseteq \leftarrow\!\!\circ\!\!\rightarrow$, $\leftarrow\!\!\rtimes\!\!\rightarrow \subseteq \leftarrow\!\!\shortparallel\!\!\rightarrow$, $\leftarrow\!\!\rtimes\!\!\rightarrow \subseteq {}^{=}\!\leftarrow$, and $\leftarrow\!\!\rtimes\!\!\rightarrow \subseteq \leftrightarrow^{=}$. We refer to Dershowitz [31] for a historical overview of these problems and the attempts to solve them. The proof of Theorem 6.3.6 presented in Section 6.3 is based on [79]. Theorem 6.3.17 is also from [101]. It extends Theorem 6.3.6 by adopting an idea originally due to Toyama [130]. The proof (in Exercise 6.21) of Theorem 6.3.17 is from [80]. The result of Exercise 6.23 is from Oyamaguchi and Ohta [107]. Exercise 6.25 is from [95] and Exercise 6.26 is from [61]. The decreasing diagrams technique is due to van Oostrom [100, 103]. It generalizes an earlier result of de Bruijn [16], which is covered in Exercise 6.33, and revisited in [39]. The proof of Theorem 6.4.11 is from [41]. Alternative correctness proofs can be found in [100, 12, 76]. Exercise 6.34 is from Stump *et al.* [124]. It is an open problem whether any confluent ARS has a locally decreasing presentation. The sufficient condition in Exercise 6.35 is from [100]. Exercise 6.36 is from [40], strengthening an earlier result of van Oostrom [100].

Chapter 7

Strategies

In this chapter we address the question how to compute normal forms in rewrite systems that need not be terminating. In Section 7.1 we introduce various strategies for selecting redexes in terms. Their properties are studied in Section 7.2. In Section 7.3 we investigate how to avoid selecting redexes that do not contribute to the computation of normal forms. Strategy annotations provide a fine-grained approach for controlling the evaluation of terms and are discussed in Section 7.4. In Section 7.5 we introduce context-sensitive rewriting, a less precise but useful mechanism to restrict computations.

7.1 Rewrite Strategies

In Section 3.2 we have seen examples of finite TRSs that are not terminating. If a term t in such a TRS has a normal form we can always compute a normal form of t by computing its reducts in a breadth-first manner until we encounter a normal form. However, in general this is a highly inefficient way to compute normal forms. In this section we present more efficient ways to compute normal forms. Rather than exploring all possible rewrite sequences from a given term, a rewrite strategy dictates which sequences must be computed. We refer to Section 1.5 for the basic definitions concerning rewrite strategies. The strategies considered there can be called *one-step* strategies. In this chapter we also consider *many-step* strategies in which several redexes can be contracted in a single strategy step. The abstract definitions and results of Section 1.5 apply to many-step strategies via the associated ARS $\langle \mathcal{T}(\mathcal{F},\mathcal{V}), \to_{\mathcal{R}}^{+} \rangle$.

In this chapter we are mainly concerned with *computable* strategies.

Example 7.1.1. Consider the strategy $\mathcal{S}$ that always contracts a largest redex in a reducible term. For most TRSs this strategy is non-deterministic, since (1) a term may have different redexes of the same size and (2) a given redex may be contracted using different rewrite rules. For instance, with respect to the TRS $\mathcal{R}$ of Table 3.4 we have

$$(1+2)+(3+4) \xrightarrow{\mathcal{S}} 3+(3+4) \qquad 0:(1:2) \xrightarrow{\mathcal{S}} 1:2$$
$$(1+2)+(3+4) \xrightarrow{\mathcal{S}} (1+2)+7 \qquad 0:(1:2) \xrightarrow{\mathcal{S}} (0+1):2$$

The strategy $\mathcal{S}$ is trivially (hyper-)normalizing for $\mathcal{R}$ because $\mathcal{R}$ is a terminating TRS.

Several rewrite strategies are defined by selecting the redexes which are to be con-

tracted in each step based on their position, independent of the TRS under consideration. Below we define four such *positional* strategies. Recall from Section 3.2 the definitions of innermost and outermost redexes.

Definition 7.1.2. Let $\mathcal{R}$ be a TRS. We write $s \xrightarrow{\mathsf{i}}_{\mathcal{R}} t$ if $s \to t$ by contracting an innermost redex in s. The relation $\xrightarrow{\mathsf{i}}_{\mathcal{R}}$ is called *innermost rewriting*. We write $s \xrightarrow{\mathsf{o}}_{\mathcal{R}} t$ if $s \to t$ by contracting an outermost redex in s. The relation $\xrightarrow{\mathsf{o}}_{\mathcal{R}}$ is called *outermost rewriting*.

We drop the subscript $\mathcal{R}$ when it can be inferred from the context or is irrelevant. Note that $\xrightarrow{\mathsf{i}}$ and $\xrightarrow{\mathsf{o}}$ are in general *not* rewrite relations in the sense of Definition 2.2.31.

Example 7.1.3. Consider the TRS consisting of the single rule $\mathsf{f}(x) \to \mathsf{a}$. We have $\mathsf{f}(x) \xrightarrow{\mathsf{i}} \mathsf{a}$ and $\mathsf{f}(x) \xrightarrow{\mathsf{o}} \mathsf{a}$ but $\mathsf{f}(\mathsf{f}(\mathsf{a})) \xrightarrow{\mathsf{i}} \mathsf{a}$ and $\mathsf{f}(\mathsf{f}(x)) \xrightarrow{\mathsf{o}} \mathsf{f}(\mathsf{a})$ do not hold. So innermost steps are not closed under substitutions and outermost steps are not closed under contexts.

Definition 7.1.4. The *leftmost outermost* rewrite strategy $\mathcal{S}_{\mathsf{lo}}$ always contracts the leftmost of the outermost redexes. The *maximal outermost* rewrite strategy $\mathcal{S}_{\mathsf{mo}}$ contracts all outermost redexes in parallel. Likewise, the *leftmost innermost* rewrite strategy $\mathcal{S}_{\mathsf{li}}$ contracts the leftmost of the innermost redexes and the *maximal innermost* rewrite strategy $\mathcal{S}_{\mathsf{mi}}$ contracts all innermost redexes in parallel.

We write $\xrightarrow{\mathsf{lo}}$ for $\xrightarrow{\mathcal{S}_{\mathsf{lo}}}$ and similarly for the other strategies defined in Definition 7.1.4. The strategies are defined for arbitrary TRSs, but for non-orthogonal TRSs they need not be deterministic.

Example 7.1.5. Consider again the TRS $\mathcal{R}$ of Table 3.4. Because $\mathcal{R}$ is terminating, all strategies will normalize the term $t = (0{:}(1{+}2){+}(3{+}4)){+}(5{+}6)$. The leftmost outermost strategy needs 12 steps to reach the normal form $2:1$:

$$\begin{aligned}
&\underline{(0:(1+2)+(3+4))}+(5+6) \xrightarrow{\mathsf{lo}} \underline{0:((1+2)+(3+4))+(5+6)}\\
&\quad\xrightarrow{\mathsf{lo}} \underline{0:(((1+2)+(3+4))+(5+6))} \xrightarrow{\mathsf{lo}} ((\underline{1+2})+(3+4))+(5+6)\\
&\quad\xrightarrow{\mathsf{lo}} (3+(\underline{3+4}))+(5+6) \xrightarrow{\mathsf{lo}} (\underline{3+7})+(5+6) \xrightarrow{\mathsf{lo}} \underline{1:0+(5+6)}\\
&\quad\xrightarrow{\mathsf{lo}} 1:(0+(\underline{5+6})) \xrightarrow{\mathsf{lo}} 1:(\underline{0+1:1}) \xrightarrow{\mathsf{lo}} \underline{1:(1:(0+1))}\\
&\quad\xrightarrow{\mathsf{lo}} (\underline{1+1}):(0+1) \xrightarrow{\mathsf{lo}} 2:(\underline{0+1}) \xrightarrow{\mathsf{lo}} 2:1
\end{aligned}$$

The maximal innermost strategy contracts 10 redexes in 8 steps:

$$\begin{aligned}
&(0:(\underline{1+2})+(\underline{3+4}))+(\underline{5+6}) \xrightarrow{\mathsf{mi}} (\underline{0:3}+7)+1:1 \xrightarrow{\mathsf{mi}} (\underline{3+7})+1:1\\
&\quad\xrightarrow{\mathsf{mi}} \underline{1:0+1:1} \xrightarrow{\mathsf{mi}} 1:(\underline{0+1:1}) \xrightarrow{\mathsf{mi}} 1:(1:(\underline{0+1})) \xrightarrow{\mathsf{mi}} \underline{1:(1:1)}\\
&\quad\xrightarrow{\mathsf{mi}} (\underline{1+1}):1 \xrightarrow{\mathsf{mi}} 2:1
\end{aligned}$$

This is not the only sequence computed by the maximal innermost strategy:

$$\begin{aligned}
&(0:(\underline{1+2})+(\underline{3+4}))+(\underline{5+6}) \xrightarrow{\mathsf{mi}}{}^{*} \underline{1:0+1:1} \xrightarrow{\mathsf{mi}} 1:(\underline{1:0+1})\\
&\quad\xrightarrow{\mathsf{mi}} 1:(1:(\underline{0+1})) \xrightarrow{\mathsf{mi}} \underline{1:(1:1)} \xrightarrow{\mathsf{mi}} (\underline{1+1}):1 \xrightarrow{\mathsf{mi}} 2:1
\end{aligned}$$

Whereas the maximal innermost strategy produces innermost rewrite sequences, the following example shows that the rewrite sequences produced by the maximal outermost strategy need not be outermost.

Example 7.1.6. Consider the TRS consisting of the rewrite rules

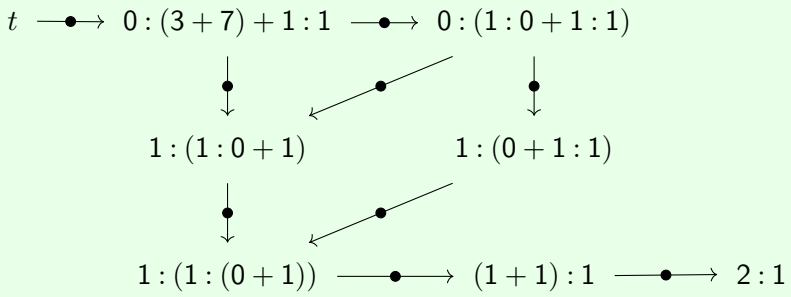
$$\mathsf{a} \to \mathsf{b} \qquad \mathsf{f}(\mathsf{b}, x) \to x \qquad \mathsf{f}(x, \mathsf{b}) \to x$$

and the term $t = \mathsf{f}(\mathsf{a}, \mathsf{a})$. We have $t \xrightarrow{\mathsf{mo}} \mathsf{f}(\mathsf{b}, \mathsf{b})$ but $t \xrightarrow{\mathsf{o}}{}^* \mathsf{f}(\mathsf{b}, \mathsf{b})$ does not hold.

The final strategy that we consider in this section contracts as many redexes as possible, in a top-down fashion (cf. Definition 6.1.13).

Definition 7.1.7. The *maximal* rewrite strategy $\mathcal{S}_{\mathsf{m}}$ performs a maximal multi-step for every reducible term.

Example 7.1.8. Consider again the TRS $\mathcal{R}$ of Table 3.4. There are several ways to normalize the term $t = (0 : (1 + 2)) + (3 + 4)) + (5 + 6)$ using the maximal strategy:



Exercises

7.1 Compare the five rewrite strategies defined in this section on the term $\mathsf{s}(0 + 0) \times (0 + \mathsf{s}(0))$ with respect to the TRS $\mathcal{R}_2$ of Table 3.1.

7.2 Suppose we extend CL with constants P and $\underline{n}$ for every natural number n and rewrite rules $\mathsf{P}\,\underline{m}\,\underline{n} \to \underline{m + n}$ for all natural numbers m and n. Normalize the term $\mathsf{S}(\mathsf{S}(\mathsf{KP})(\mathsf{SPI}))(\mathsf{SPI})(\mathsf{SP}(\mathsf{SPI})\underline{2})$ using the five rewrite strategies defined in this section.

7.3 Write $s \xrightarrow{\neg\mathsf{i}} t$ if t is obtained from s by contracting a non-innermost redex in s. Show that $\xrightarrow{\neg\mathsf{i}}$ is not a strategy in general.

7.4 Example 7.1.6 shows that the inclusion $\xrightarrow{\mathsf{mo}} \subseteq \xrightarrow{\mathsf{o}}{}^*$ does not hold in general. What about orthogonal TRSs?

7.5 ***a*** Show that $\mathcal{S}_{\mathsf{m}}$ is a strategy.

b Show that $\mathcal{S}_{\mathsf{m}}$ is deterministic for orthogonal TRSs.

7.2 Normalization

In this section we investigate the normalization behaviour of the five strategies defined in the preceding section. First we show that innermost strategies only work well for terminating systems.

Definition 7.2.1. A term is called *weakly innermost normalizing* (WIN) or simply *innermost normalizing* if it admits an innermost rewrite sequence to normal form. A TRS is innermost normalizing if all its terms are innermost normalizing.

Note that a TRS $\mathcal{R}$ is innermost normalizing if and only if the relation $\xrightarrow{\mathsf{i}}_{\mathcal{R}}$ is normalizing. We will prove below that innermost normalization coincides with termination for (terms in) *non-ambiguous* TRSs, from which the main result concerning innermost strategies (Corollary 7.2.13 below) easily follows. As intermediate result, we relate innermost normalization to *innermost termination*.

Definition 7.2.2. A term is called *innermost terminating* or *strongly innermost normalizing* (SIN) if it admits no infinite innermost rewrite sequence. A TRS is innermost terminating if all its terms have this property.

Example 7.2.3. Consider the TRS consisting of the rewrite rules

$$\mathsf{a} \to \mathsf{b} \qquad \mathsf{a} \to \mathsf{c} \qquad \mathsf{c} \to \mathsf{f}(\mathsf{a}) \qquad \mathsf{f}(x) \to \mathsf{f}(\mathsf{a})$$

The term a is innermost normalizing but not innermost terminating. The term c is not innermost normalizing. The TRS consisting of the rewrite rules

$$\mathsf{g}(x, \mathsf{h}(x)) \to \mathsf{g}(\mathsf{d}, \mathsf{h}(x)) \qquad \mathsf{h}(\mathsf{d}) \to \mathsf{h}(\mathsf{e})$$

is innermost terminating but not terminating.

Theorem 7.2.4. *Innermost rewriting has random descent for non-ambiguous* TRS*s.*

Proof We show that innermost rewriting has the property BWCR, which is a sufficient condition for random descent (Exercise 1.45(a)). Let $s \xrightarrow{\mathsf{i}} t_1$ by contracting an innermost redex Δ_1 at position p_1 and $s \xrightarrow{\mathsf{i}} t_2$ by contracting an innermost redex Δ_2 at position p_2. If p_1 and p_2 are parallel then we obtain $t_1 \xrightarrow{\mathsf{i}} \cdot \xleftarrow{\mathsf{i}} t_2$. Note that $p_1 < p_2$ and $p_2 < p_1$ are impossible because p_1 and p_2 are positions of innermost redexes. The remaining case is therefore $p_1 = p_2$ and, because $\mathcal{R}$ is non-ambiguous, $\Delta_1 = \Delta_2$ and there is a unique rewrite rule matching this redex. Consequently $t_1 = t_2$. □

Corollary 7.2.5. *Let $\mathcal{R}$ be a non-ambiguous* TRS*.*

1. *A term t is innermost normalizing if and only if it is innermost terminating.*
2. *The* TRS *$\mathcal{R}$ is innermost normalizing if and only if it is innermost terminating.*

Proof The second statement is an immediate consequence of the first statement and the if direction of the first statement is trivial. For the only-if direction we observe that innermost rewriting has random descent by Theorem 7.2.4. Hence innermost normalizing terms are innermost terminating as a consequence of Theorem 1.5.15. □

Next we show the equivalence of innermost termination and termination for non-ambiguous TRSs. In the proof we use some auxiliary results.

Definition 7.2.6. Let $s \to t$. We write $s \to_{\mathsf{c}} t$ if $s \to t$ by contracting a redex that is a complete subterm of s and $s \to_{\mathsf{nc}} t$ if $s \to t$ by contracting a redex that is not a complete subterm of s.

Lemma 7.2.7. *The relation $\to_{\mathsf{c}}$ is terminating for every* TRS.

Proof For a term t let $P(t)$ be the set of positions $p \in \mathcal{P}\mathsf{os}(t)$ such that $t|_p$ is not complete. If $t \to_{\mathsf{c}} u$ then either $P(t) = P(u)$ or $P(t) \supsetneq P(u)$. Now suppose for a proof by contradiction that the relation $\to_{\mathsf{c}}$ is not terminating. So there exists an infinite rewrite sequence $t = t_0 \to_{\mathsf{c}} t_1 \to_{\mathsf{c}} t_2 \to_{\mathsf{c}} \cdots$. Since $P(t_i) \supsetneq P(t_{i+1})$ can hold only finitely many times, there exists an index $N \geqslant 0$ such that $P(t_N) = P(t_i)$ for all $i > N$. Using the pigeonhole principle it follows that a complete subterm of t_N admits an infinite rewrite sequence. This is clearly impossible. □

Hence any infinite rewrite sequence contains infinitely many $\to_{\mathsf{nc}}$ steps.

Definition 7.2.8. Given a term t, we write $t{\downarrow}^{\mathsf{m}}_{\mathsf{c}}$ for the term obtained from t by replacing its maximal complete subterms by their unique normal forms.

Lemma 7.2.9. *If $s \to_{\mathsf{c}} t$ then $s{\downarrow}^{\mathsf{m}}_{\mathsf{c}} \to^* t{\downarrow}^{\mathsf{m}}_{\mathsf{c}}$, for every* TRS.

Proof We clearly have $t \to^* s{\downarrow}^{\mathsf{m}}_{\mathsf{c}}$ by contracting redexes in complete subterms of t. Hence $s{\downarrow}^{\mathsf{m}}_{\mathsf{c}} \to^* t{\downarrow}^{\mathsf{m}}_{\mathsf{c}}$. □

Example 7.2.10. Consider the TRS consisting of the rewrite rules

$$\mathsf{a} \to \mathsf{b} \qquad \mathsf{f}(\mathsf{a}) \to \mathsf{f}(\mathsf{a}) \qquad \mathsf{f}(\mathsf{b}) \to \mathsf{c}$$

and the rewrite step $s = \mathsf{f}(\mathsf{a}) \to_{\mathsf{c}} \mathsf{f}(\mathsf{b}) = t$. We have $s{\downarrow}^{\mathsf{m}}_{\mathsf{c}} = \mathsf{f}(\mathsf{b}) \to \mathsf{c} = t{\downarrow}^{\mathsf{m}}_{\mathsf{c}}$. Next consider the TRS consisting of the rewrite rules

$$\mathsf{a} \to \mathsf{b} \qquad \mathsf{f}(\mathsf{a}) \to \mathsf{g}(\mathsf{a}) \qquad \mathsf{g}(x) \to \mathsf{f}(x)$$

and the rewrite step $s = \mathsf{f}(\mathsf{a}) \to_{\mathsf{nc}} \mathsf{g}(\mathsf{a}) = t$. We have $s{\downarrow}^{\mathsf{m}}_{\mathsf{c}} = \mathsf{f}(\mathsf{b})$ and $t{\downarrow}^{\mathsf{m}}_{\mathsf{c}} = \mathsf{g}(\mathsf{b})$. Since $\mathsf{f}(\mathsf{b}) \to^* \mathsf{g}(\mathsf{b})$ does not hold, the analogous statement of Lemma 7.2.9 for $\to_{\mathsf{nc}}$ fails.

We can recover and even strengthen Lemma 7.2.9 for $\to_{\mathsf{nc}}$ by imposing non-ambiguity.

Lemma 7.2.11. *If $s \to_{\mathsf{nc}} t$ then $s{\downarrow}^{\mathsf{m}}_{\mathsf{c}} \to^! t{\downarrow}^{\mathsf{m}}_{\mathsf{c}}$, for every non-ambiguous* TRS.

Proof Suppose $s \to_{\mathsf{nc}} t$ by applying the rewrite rule $\ell \to r$ at position p with substitution σ. We have $p \in \mathcal{P}\mathsf{os}_{\mathcal{F}}(s{\downarrow}^{\mathsf{m}}_{\mathsf{c}})$ because the subterm $s|_p$ is not complete. Since the TRS lacks critical pairs, $s{\downarrow}^{\mathsf{m}}_{\mathsf{c}}|_p \geq \ell$. We have $s{\downarrow}^{\mathsf{m}}_{\mathsf{c}}|_p = \ell\tau$ for the substitution $\tau = {\downarrow}^{\mathsf{m}}_{\mathsf{c}} \circ \sigma$. Hence $s{\downarrow}^{\mathsf{m}}_{\mathsf{c}} \to s{\downarrow}^{\mathsf{m}}_{\mathsf{c}}[r\tau]_p$. Clearly $t \to^* s{\downarrow}^{\mathsf{m}}_{\mathsf{c}}[r\tau]_p$ by contracting redexes in complete subterms of t. Consequently $s{\downarrow}^{\mathsf{m}}_{\mathsf{c}}[r\tau]_p \to^* t{\downarrow}^{\mathsf{m}}_{\mathsf{c}}$ and thus $s{\downarrow}^{\mathsf{m}}_{\mathsf{c}} \to^! t{\downarrow}^{\mathsf{m}}_{\mathsf{c}}$ as desired. □

Hence by applying the mapping $u \mapsto u{\downarrow_{\mathsf{c}}^{\mathsf{m}}}$ to all terms in any infinite rewrite sequence starting from a non-terminating term t in a non-ambiguous TRS, we obtain an infinite rewrite sequence starting from $t{\downarrow_{\mathsf{c}}^{\mathsf{m}}}$.

Theorem 7.2.12. *Let $\mathcal{R}$ be a non-ambiguous* TRS.
1. *A term t is innermost terminating if and only if it is terminating.*
2. *The* TRS $\mathcal{R}$ *is innermost terminating if and only if it is terminating.*

Proof The second statement is an immediate consequence of the first statement and the if direction of the first statement is trivial. So let t be an innermost terminating term. We show that t is terminating. Suppose to the contrary that t is non-terminating. We consider an infinite rewrite sequence A starting at t with the property that the first non-innermost step is essential; contracting any innermost redex at that point would result in a terminating term. Write

$$A\colon\ t = t_0 \xrightarrow{\mathsf{i}} t_1 \xrightarrow{\mathsf{i}} \cdots \xrightarrow{\mathsf{i}} t_i \to t_{i+1} \to \cdots$$

such that $t_i \to t_{i+1}$ is the first non-innermost step. By assumption, contracting an innermost redex in t_i yields a terminating term. In particular, innermost redexes in t_i are terminating. Since $\mathcal{R}$ is locally confluent by the non-ambiguity assumption, every innermost redex in t_i is complete by Lemma 1.3.1, and thus contained in a maximal complete subterm of t_i. It follows that $t_i{\downarrow_{\mathsf{c}}^{\mathsf{m}}}$ is terminating. However, since A contains infinitely many $\to_{\mathsf{nc}}$ steps, from Lemmata 7.2.9 and 7.2.11 we obtain an infinite rewrite sequence

$$t_i{\downarrow_{\mathsf{c}}^{\mathsf{m}}} \to^* t_{i+1}{\downarrow_{\mathsf{c}}^{\mathsf{m}}} \to^* \cdots$$

This is a contradiction. □

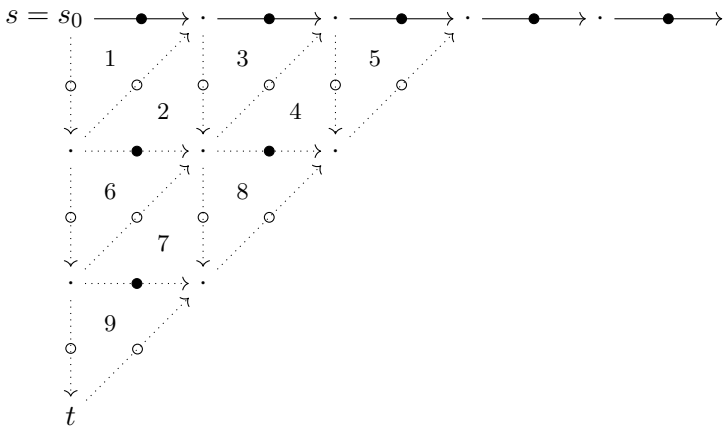
Corollary 7.2.13. *Every innermost strategy is perpetual for non-ambiguous* TRS*s.*

Proof Let t be a non-terminating term in a non-ambiguous TRS. According to Corollary 7.2.5 and Theorem 7.2.12, t is not innermost normalizing. Since innermost strategies produce innermost rewrite sequences, the result follows. □

After this negative result we present two positive results; the normalization of the maximal and maximal outermost strategies for orthogonal TRSs. The former is considerably easier to prove. For orthogonal TRSs the maximal strategy is not only normalizing, it actually satisfies the stronger cofinal property (Definition 1.5.6).

Theorem 7.2.14. *The maximal strategy is cofinal for orthogonal* TRS*s.*

Proof Let $\mathcal{R}$ be an orthogonal TRS. Suppose $s = s_0 \mathrel{-\!\!\bullet\!\!\to} s_1 \mathrel{-\!\!\bullet\!\!\to} s_2 \mathrel{-\!\!\bullet\!\!\to} \cdots$ and $s \to^* t$. Since $\to \subseteq \mathrel{-\!\!\bullet\!\!\to}$ by Lemmata 3.2.14 and 6.1.12, $s \mathrel{-\!\!\bullet\!\!\to}^n t$ for some $n \geqslant 0$. By using Lemma 6.1.15 repeatedly, the following diagram is obtained (where we assume $n = 3$):



The numbers in the triangles indicate the order in which the diagram is filled. Since $\multimap \subseteq \to^*$ by Lemma 6.1.12, the result follows. □

Theorem 7.2.15. *The maximal strategy is hyper-normalizing for orthogonal* TRS*s.*

Proof We obtain $\multimap \cdot \bullet\!\!\to \subseteq \bullet\!\!\to \cdot \multimap$ by two applications of the triangle property. Hence $\to \cdot \bullet\!\!\to \subseteq \bullet\!\!\to \cdot \to^*$ and thus $\bullet\!\!\to$ quasi-commutates over $\to$. Since $\bullet\!\!\to$ is normalizing by Theorem 7.2.14, hyper-normalization follows from Lemma 1.5.5. □

The proof of our next normalization result is considerably harder. This should not come as a surprise since the possibility of missing the normal form increases by performing less redex contractions.

Theorem 7.2.16. *The maximal outermost strategy is hyper-normalizing for orthogonal* TRS*s.*

Next we turn our attention to the leftmost outermost strategy. The following example shows that this strategy is not normalizing for orthogonal TRSs.

Example 7.2.17. Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\mathsf{a} \to \mathsf{b} \qquad \mathsf{c} \to \mathsf{c} \qquad \mathsf{f}(x, \mathsf{b}) \to \mathsf{b}$$

and the term $\mathsf{f}(\mathsf{c}, \mathsf{a})$. The leftmost outermost strategy selects redex c and hence will produce the infinite rewrite sequence

$$\mathsf{f}(\mathsf{c}, \mathsf{a}) \xrightarrow{\mathsf{lo}} \mathsf{f}(\mathsf{c}, \mathsf{a}) \xrightarrow{\mathsf{lo}} \mathsf{f}(\mathsf{c}, \mathsf{a}) \xrightarrow{\mathsf{lo}} \cdots$$

whereas the maximal outermost strategy normalizes $\mathsf{f}(\mathsf{c}, \mathsf{a})$ in two steps:

$$\mathsf{f}(\mathsf{c}, \mathsf{a}) \xrightarrow{\mathsf{mo}} \mathsf{f}(\mathsf{c}, \mathsf{b}) \xrightarrow{\mathsf{mo}} \mathsf{b}$$

Nevertheless, there is an important subclass of the orthogonal TRSs for which $\mathcal{S}_{\mathsf{lo}}$ is normalizing.

Definition 7.2.18. A term t is called *left-normal* if function symbols precede variables when t is written in prefix notation. Formally, if $p \in \mathcal{P}\mathsf{os}_{\mathcal{V}}(t)$ and $q \in \mathcal{P}\mathsf{os}(t)$ with $p <_{\mathsf{left}} q$ (cf. Exercise 2.12) then $q \in \mathcal{P}\mathsf{os}_{\mathcal{V}}(t)$. A TRS $\mathcal{R}$ is *left-normal* if the left-hand side ℓ of every rule $\ell \to r \in \mathcal{R}$ is a left-normal term.

Example 7.2.19. The TRS $\mathcal{R}$ of Example 7.2.17 is not left-normal since the variable x in the left-hand side $\mathsf{f}(x, \mathsf{b})$ of the third rewrite rule precedes the constant b. The prime example of a left-normal orthogonal TRS is CL.

Below we prove that $\mathcal{S}_{\mathsf{lo}}$ is normalizing for all left-normal orthogonal TRSs.

Definition 7.2.20. We write $s \overset{n}{\mathrel{+\!\!\!\to}} t$ if $s \mathrel{+\!\!\!\to} t$ by contracting n redexes in s.

Lemma 7.2.21. *The inclusion* $\overset{\mathsf{lo}}{\leftarrow} \cdot \overset{n}{\mathrel{+\!\!\!\to}} \subseteq \overset{n-1}{\mathrel{+\!\!\!\to}} \cup \mathrel{+\!\!\!\to} \cdot \overset{\mathsf{lo}}{\leftarrow}$ *holds for all left-normal orthogonal* TRS*s.*

In the proof below we write $\mathsf{SN}(\mathsf{lo})$ for the set of $\mathcal{S}_{\mathsf{lo}}$-terminating terms.

Theorem 7.2.22. *The leftmost outermost strategy is normalizing for all left-normal orthogonal* TRS*s.*

Proof First we prove the statement

$$\text{if } s \overset{n}{\mathrel{+\!\!\!\to}} t \text{ and } t \in \mathsf{SN}(\mathsf{lo}) \text{ then } s \overset{\mathsf{lo}}{\leftrightarrow}{}^* t \qquad (*)$$

by induction on (t, n) with respect to the well-founded order $(\succ, >)_{\mathsf{lex}}$ where $\succ$ is the restriction of $\overset{\mathsf{lo}}{\longrightarrow}$ to terms in $\mathsf{SN}(\mathsf{lo})$. If s is a normal form or $n = 0$ then $s = t$ and thus trivially $s \overset{\mathsf{lo}}{\leftrightarrow}{}^* t$. Otherwise, $s \overset{\mathsf{lo}}{\longrightarrow} s'$ and $n > 0$. We distinguish two cases according to Lemma 7.2.21.

1. If $s' \overset{n-1}{\mathrel{+\!\!\!\to}} t$ then we obtain $s' \overset{\mathsf{lo}}{\leftrightarrow}{}^* t$ from the induction hypothesis.
2. If $s' \mathrel{+\!\!\!\to} t' \overset{\mathsf{lo}}{\leftarrow} t$ then $t \succ t' \in \mathsf{SN}(\mathsf{lo})$. Hence we can apply the induction hypothesis to $s' \mathrel{+\!\!\!\to} t'$ to obtain $s' \overset{\mathsf{lo}}{\leftrightarrow}{}^* t'$.

In both cases we obtain $s \overset{\mathsf{lo}}{\leftrightarrow}{}^* t$. Now let $s \to^! t$. Clearly $s \to^n t$ for some $n \geqslant 0$ and $t \in \mathsf{SN}(\mathsf{lo})$. We show $s \in \mathsf{SN}(\mathsf{lo})$ by induction on n. The base case is trivial. Suppose $s \to s' \to^{n-1} t$. We obtain $s' \in \mathsf{SN}(\mathsf{lo})$ from the induction hypothesis. In particular, $s' \overset{\mathsf{lo}}{\longrightarrow}{}^* t$. We have $s \mathrel{+\!\!\!\to} s'$ and thus we obtain $s \overset{\mathsf{lo}}{\leftrightarrow}{}^* s'$ from $(*)$. Hence $s \overset{\mathsf{lo}}{\leftrightarrow}{}^* t$. Since $\mathcal{S}_{\mathsf{lo}}$ is a deterministic strategy for orthogonal TRSs, it has random descent and thus $s \in \mathsf{SN}(\mathsf{lo})$ follows from Theorem 1.5.15. □

Exercises

7.6 Corollary 7.2.5 and Theorem 7.2.12 are stated for non-ambiguous TRSs. One of these two theorems holds for the larger class of locally confluent *overlay* systems. Which one? Give a proof and a counterexample. (An overlay system is a TRS with the property that all its critical pairs are overlays.)

7.7 A TRS $\mathcal{R}$ is called *non-erasing* if $\mathcal{V}\mathsf{ar}(\ell) = \mathcal{V}\mathsf{ar}(r)$ for every rewrite rule $\ell \to r$ of $\mathcal{R}$. Prove that any strategy is normalizing for non-erasing non-ambiguous TRSs.

7.8 Is the maximal outermost strategy cofinal for every orthogonal TRS?

7.9 Which of the TRSs in Tables 3.1 – 3.6 are left-normal?

7.10 Give an inductive definition of $\overset{n}{\twoheadrightarrow}$ and prove that $\twoheadrightarrow$ equals the infinite union of $\overset{n}{\twoheadrightarrow}$ for all $n \geqslant 0$.

7.11 ***a*** State a result similar to Theorem 7.2.22 for the rightmost outermost strategy.

b Prove that leftmost outermost is hyper-normalizing for left-normal orthogonal TRSs.

7.3 Call by Need Strategies

In the previous section we identified two normalizing rewrite strategies for orthogonal TRSs: maximal outermost and the maximal strategy. Both strategies perform useless contractions when normalizing terms.

Example 7.3.1. Consider the TRS $\mathcal{R}_1$ of Table 3.1. The maximal outermost strategy normalizes the term $(0 \times \mathsf{s}(0)) \times (0 + \mathsf{s}(0))$ by contracting the three underlined redexes:

$$\underline{(0 \times \mathsf{s}(0))} \times \underline{(0 + \mathsf{s}(0))} \xrightarrow{\mathsf{mo}} \underline{0 \times \mathsf{s}(0)} \xrightarrow{\mathsf{mo}} 0$$

The same sequence is computed by the maximal strategy. The following sequence shows that the normal form can be computed by contracting only two redexes:

$$\underline{(0 \times \mathsf{s}(0))} \times (0 + \mathsf{s}(0)) \to \underline{0 \times (0 + \mathsf{s}(0))} \to 0$$

So for reaching the normal form of $(0 \times \mathsf{s}(0)) \times (0 + \mathsf{s}(0))$ it is unnecessary to contract the redex $0 + \mathsf{s}(0)$. In the following definition we make this precise.

Definition 7.3.2. Let $\mathcal{R}$ be a TRS. Let $\bullet$ be a fresh constant and consider the extension $\mathcal{R}_\bullet = \mathcal{R} \cup \{\bullet \to \bullet\}$. A redex Δ in a term $t = C[\Delta]$ is called *needed* if $C[\bullet]$ has no normal form in $\mathcal{R}_\bullet$. We write $C[\ell\sigma] \xrightarrow{\mathcal{S}_\mathsf{n}} C[r\sigma]$ or simply $C[\ell\sigma] \xrightarrow{\mathsf{n}} C[r\sigma]$ if $\ell \to r \in \mathcal{R}$ and $\ell\sigma$ is needed in $C[\ell\sigma]$. The relation $\xrightarrow{\mathsf{n}}$ is called *needed rewriting*.

Note that normal forms in $\mathcal{R}_\bullet$ cannot contain the symbol $\bullet$.

Example 7.3.3. Consider again the TRS $\mathcal{R}_1$ of Table 3.1. The redex $0 \times \mathsf{s}(0)$ in the term $(0 \times \mathsf{s}(0)) \times (0 + \mathsf{s}(0))$ is needed since $\bullet \times (0 + \mathsf{s}(0))$ rewrites only to itself and $\bullet \times \mathsf{s}(0)$. The redex $0 + \mathsf{s}(0)$ is not needed since $(0 \times \mathsf{s}(0)) \times \bullet \to 0 \times \bullet \to 0$.

We have the following important theorem.

Theorem 7.3.4. *Needed rewriting is a hyper-normalizing rewrite strategy for orthogonal* TRS*s.*

The above result provides an optimal normalizing strategy: always contract an innermost needed redex. By definition, needed redexes have to be contracted in order to reach a normal form. Moreover, by selecting an innermost needed redex we ensure that

other needed redexes are not duplicated. Alternatively, if we adopt a graph representation to allow sharing of identical subterms, we can drop the restriction to innermost needed redexes; any needed rewrite strategy is optimal. Unfortunately, in general needed redexes are not computable.

Theorem 7.3.5. *The following problem is undecidable:*

instance: *an orthogonal* TRS $\mathcal{R}$*, a term* t*, and a redex* Δ *in* t

question: *is* Δ *needed in* t*?*

Proof We use a reduction from the halting problem for Turing machines. Let $M = (Q, \Sigma, \Gamma, \sqcup, \delta, s, F)$ be a Turing machine and let α be a configuration of M. Consider the TRS $\mathcal{R}_M$ of Definition 3.3.7. The TRS $\mathcal{R}$ is obtained from $\mathcal{R}_M$ by adding a rewrite rule $p(x, a(y)) \to \mathsf{stop}$ for all $p \in Q$ and $a \in \Gamma$ such that $\delta(p, a)$ is undefined, a rewrite rule $p(x, \infty) \to \mathsf{stop}$ for all $p \in Q$ such that $\delta(p, \sqcup)$ is undefined, and the rewrite rule $\mathsf{f}(\mathsf{stop}, x) \to \mathsf{stop}$. Note that $\mathcal{R}$ is orthogonal. Consider the term $t = \mathsf{f}(\phi(\alpha), \Delta)$ with an arbitrary redex Δ. If configuration α is terminating then $\phi(\alpha)$ rewrites to stop and thus $\mathsf{f}(\phi(\alpha), \bullet) \to^* \mathsf{f}(\mathsf{stop}, \bullet) \to \mathsf{stop}$ and so Δ is not needed. If configuration α is not terminating then $\phi(\alpha)$ does not rewrite to stop, and so any reduct of $\mathsf{f}(\phi(\alpha), \bullet)$ will have the shape $\mathsf{f}(u, \bullet)$ and hence Δ is needed. □

Exercises

7.12 For each of the following rewrite sequences in the TRS combinatory logic, determine which redexes are needed.

a $\mathsf{KI(KKI)I} \to \mathsf{KIKI} \to \mathsf{II}$

b $\mathsf{SII(SII)} \to \mathsf{I(SII)(I(SII))} \to \mathsf{SII(I(SII))} \to \mathsf{SII(SII)}$

c $\mathsf{SSSSSS} \to \mathsf{SS(SS)SS} \to \mathsf{SS(SSS)S} \to \mathsf{SS(SSSS)} \to \mathsf{SS(SS(SS))}$

7.13 Show that every reducible term in an orthogonal TRS has an outermost redex that is needed.

7.14 Show that for every TRS all redexes in a term without normal form are needed.

7.15 Show that in a left-normal orthogonal TRS the leftmost outermost redex in every reducible term is needed.

7.4 Strategy Annotations

In this section we equip TRSs with annotations to control the evaluation strategy.

Definition 7.4.1. A *strategy annotation* for a function symbol f is a finite list $A(f)$ consisting of argument positions of f and (labels of) rewrite rules for f. We say that $A(f)$ is *full* if $A(f)$ contains all argument positions of f and all rewrite rules for f.

The following example illustrates how a strategy annotation guides the search for a redex to contract.

Example 7.4.2. Consider the TRS consisting of the rewrite rules

$$x \wedge \mathsf{T} \xrightarrow{\alpha} x \qquad \mathsf{T} \vee x \xrightarrow{\gamma} \mathsf{T} \qquad \infty \xrightarrow{\epsilon} \infty$$
$$x \wedge \mathsf{F} \xrightarrow{\beta} \mathsf{F} \qquad \mathsf{F} \vee x \xrightarrow{\delta} x$$

and the strategy annotation A with

$$A(\wedge) = [2, \alpha, \beta, 1] \qquad A(\vee) = [1, \gamma, \delta, 2] \qquad A(\infty) = [\epsilon] \qquad A(\mathsf{T}) = A(\mathsf{F}) = [\,]$$

Suppose we want to evaluate the term $t = (\infty \wedge \mathsf{F}) \vee (\mathsf{T} \vee \infty)$.

- ▷ The strategy annotation $[1, \gamma, \delta, 2]$ of its root symbol $\vee$ tells us that we first look for a redex in the first argument $\infty \wedge \mathsf{F}$ of t.
- ▷ The strategy annotation $[2, \alpha, \beta, 1]$ for $\wedge$ indicates to look for a redex in the second argument F of $\infty \wedge \mathsf{F}$. Since $A(\mathsf{F}) = [\,]$, this will fail. So we discard the first element of $[2, \alpha, \beta, 1]$ and try whether rule α applies. This also fails. Next up is rule β. Since β is applicable, we have found our redex and hence $\infty \wedge \mathsf{F}$ rewrites to F.

So t rewrites to $\mathsf{F} \vee (\mathsf{T} \vee \infty)$.

A formal definition of the evaluation induced by a strategy annotation is not hard to give. Here $[H \mid T]$ denotes the list with head H and tail T.

Definition 7.4.3. Given a strategy annotation A and a term t, we define $\mathrm{redex}_A(t)$ as $\mathrm{redex}'_A(t, A(\mathsf{root}(t)))$ where the latter is defined by the following clauses:

$$\begin{aligned}
\mathrm{redex}'_A(t, [\,]) &= \bot \\
\mathrm{redex}'_A(t, [\ell \to r \mid L]) &= \begin{cases} (\epsilon, \ell \to r) & \text{if } t \trianglerighteq \ell \\ \mathrm{redex}'_A(t, L) & \text{otherwise} \end{cases} \\
\mathrm{redex}'_A(t, [i \mid L]) &= \begin{cases} (ip, \ell \to r) & \text{if } \mathrm{redex}_A(t|_i) = (p, \ell \to r) \\ \mathrm{redex}'_A(t, L) & \text{otherwise} \end{cases}
\end{aligned}$$

The call $\mathrm{redex}_A(t)$ returns a pair $(p, \ell \to r)$ consisting of a position p in t and a rule $\ell \to r$ such that $t|_p \trianglerighteq \ell$, or $\bot$ when no redex was found.

Definition 7.4.4. For an annotation A we write $s \xrightarrow{\mathcal{S}_A} t$ if $\mathrm{redex}_A(t) = (p, \ell \to r)$ and $s \to_{p|\ell\to r} t$.

Fullness is a sufficient condition for $\mathcal{S}_A$ to be a rewrite strategy.

Lemma 7.4.5. *If the strategy annotation A is full then $\mathit{redex}_A(t) = \bot$ if and only if t is a normal form, for all terms t.*

Proof The if direction is obvious. For the only-if direction we use induction on t. If t is not a normal form then it contains a redex. We distinguish two cases.

1. Suppose $t = \ell\sigma$ for some rewrite rule $\alpha\colon \ell \to r$. Since the strategy annotation A is full, $\alpha \in A(\mathsf{root}(\ell))$. One easily proves $\mathrm{redex}'_A(t, A(\mathsf{root}(\ell))) \neq \bot$ by induction on $A(\mathsf{root}(\ell))$.
2. Suppose the i-th argument $t|_i$ of t contains a redex. We obtain $\mathrm{redex}_A(t) \neq \bot$ from the induction hypothesis. Together with $i \in A(\mathsf{root}(\ell))$, a straightforward induction proof on $A(\mathsf{root}(\ell))$ yields $\mathrm{redex}'_A(t, A(\mathsf{root}(\ell))) \neq \bot$.

Hence in both cases we have $\mathrm{redex}_A(t) \neq \bot$. □

Corollary 7.4.6. *If the strategy annotation A is full then $\mathcal{S}_A$ is a deterministic rewrite strategy.*

Suppose we want to normalize a term t with $\mathcal{S}_A$. After contracting the redex selected by $\mathrm{redex}_A(t)$ we call redex_A on the reduct of t. So the earlier computation is ignored. A more efficient way to normalize t is to continue the computation at the position of the redex contracted in the first step. This is formalized in the next definition.

Definition 7.4.7. Given a strategy annotation A and a term t, we define $\mathrm{normalize}_A(t)$ as $\mathrm{normalize}'_A(t, A(\mathsf{root}(t)))$ where the latter is defined by the following clauses:

$$\begin{aligned}
\mathrm{normalize}'_A(t, [\,]) &= t \\
\mathrm{normalize}'_A(t, [\ell \to r \mid L]) &= \begin{cases} \mathrm{normalize}_A(r\sigma) & \text{if } t = \ell\sigma \\ \mathrm{normalize}'_A(t, L) & \text{otherwise} \end{cases} \\
\mathrm{normalize}'_A(t, [i \mid L]) &= \mathrm{normalize}'_A(t[\mathrm{normalize}_A(t|_i)]_i, L)
\end{aligned}$$

We assume $\mathrm{normalize}_A$ and $\mathrm{normalize}'_A$ are evaluated in a call-by-value manner.

The evaluation of $\mathrm{normalize}_A(t)$ need not terminate when the term t is not terminating.

Example 7.4.8. Consider again the TRS and strategy annotation of Example 7.4.2. We compute $\mathrm{normalize}_A((\infty \wedge \mathsf{F}) \vee (\mathsf{T} \vee \infty))$:

$$\begin{aligned}
&\mathrm{normalize}_A((\infty \wedge \mathsf{F}) \vee (\mathsf{T} \vee \infty)) = \mathrm{normalize}'_A((\infty \wedge \mathsf{F}) \vee (\mathsf{T} \vee \infty), [1, \gamma, \delta, 2]) \\
&\quad = \mathrm{normalize}'_A(\mathrm{normalize}_A(\infty \wedge \mathsf{F}) \vee (\mathsf{T} \vee \infty), [\gamma, \delta, 2]) \\
&\quad = \mathrm{normalize}'_A(\mathsf{F} \vee (\mathsf{T} \vee \infty), [\gamma, \delta, 2]) = \mathrm{normalize}'_A(\mathsf{F} \vee (\mathsf{T} \vee \infty), [\delta, 2]) \\
&\quad = \mathrm{normalize}_A(\mathsf{T} \vee \infty) = \mathrm{normalize}'_A(\mathsf{T} \vee \infty, [1, \gamma, \delta, 2]) \\
&\quad = \mathrm{normalize}'_A(\mathrm{normalize}_A(\mathsf{T}) \vee \infty, [\gamma, \delta, 2]) = \mathrm{normalize}'_A(\mathsf{T} \vee \infty, [\gamma, \delta, 2]) \\
&\quad = \mathrm{normalize}_A(\mathsf{T}) = \mathsf{T}
\end{aligned}$$

where the third equality follows from

$$\begin{aligned}
&\mathrm{normalize}_A(\infty \wedge \mathsf{F}) = \mathrm{normalize}'_A(\infty \wedge \mathsf{F}, [2, \alpha, \beta, 1]) \\
&\quad = \mathrm{normalize}'_A(\infty \wedge \mathrm{normalize}_A(\mathsf{F}), [\alpha, \beta, 1]) = \mathrm{normalize}'_A(\infty \wedge \mathsf{F}, [\alpha, \beta, 1]) \\
&\quad = \mathrm{normalize}'_A(\infty \wedge \mathsf{F}, [\beta, 1]) = \mathrm{normalize}_A(\mathsf{F}) = \mathsf{F}
\end{aligned}$$

It is easy to see that $t \to_{\mathcal{R}}^* \mathrm{normalize}_A(t)$ holds whenever $\mathrm{normalize}_A(t)$ is defined. However, $t \xrightarrow{\mathcal{S}_A}{}^{!} \mathrm{normalize}_A(t)$ does not hold in general as $\mathrm{normalize}_A(t)$ may not be in normal form. Requiring strategy annotations to be *in-time* solves this problem, as stated in Theorem 7.4.12 below.

Definition 7.4.9. A rewrite rule $f(s_1, \ldots, s_n) \to t$ *needs* an argument position i if s_i is non-variable or s_i is variable that occurs more than once in $f(s_1, \ldots, s_n)$. A strategy annotation $A(f)$ for a function symbol f is *in-time* if the argument positions are listed in $A(f)$ before the rewrite rules that *need* them.

Example 7.4.10. Consider the TRS consisting of the rewrite rules

$$\mathsf{if}(\mathsf{T},x,y) \xrightarrow{\alpha} x \qquad \mathsf{if}(\mathsf{F},x,y) \xrightarrow{\beta} y \qquad \mathsf{if}(z,x,x) \xrightarrow{\gamma} x$$

The strategy annotation B given by $B(\mathsf{if}) = [1,\alpha,\beta,2,\gamma,3]$ is full but not in-time since γ needs argument position 3. The strategy annotation A of Example 7.4.2 is in-time.

Lemma 7.4.11. *Let $\ell = f(\ell_1,\ldots,\ell_n)$ such that argument position i is not needed for ℓ. If $t \trianglerighteq \ell$ then $t[u]_i \trianglerighteq \ell$ for any term u.*

Proof Let σ be a substitution such that $t = f(\ell_1\sigma,\ldots,\ell_n\sigma)$. Because i is not needed for ℓ, ℓ_i is a variable. Define the substitution τ as follows:

$$\tau(x) = \begin{cases} u & \text{if } x = \ell_i \\ \sigma(x) & \text{otherwise} \end{cases}$$

We have $\ell_j\tau = \ell_j\sigma$ for all $j \neq i$ because $\ell_i \notin \mathcal{V}\mathsf{ar}(\ell_j)$, due to the fact that i is not needed for ℓ. We have $\ell_j\tau = u$ by definition. Hence $t[u]_i = \ell\tau$. □

Theorem 7.4.12. *Let A be a full and in-time strategy annotation.*

1 *A term t is $\mathcal{S}_A$-normalizing if and only if $\mathit{normalize}_A(t)$ is defined.*

2 *For all normalizing terms t, $t{\downarrow_{\mathcal{S}_A}} = \mathit{normalize}_A(t)$.*

The final result of this section states that strategies induced by full and in-time strategies are not worse than the leftmost innermost strategy, from a normalization perspective.

Theorem 7.4.13. *If the strategy annotation A is full and in-time then $\mathcal{S}_A$ normalizes all terms that $\mathcal{S}_{\mathsf{li}}$ normalizes.*

Exercises

7.16 Is fullness of A a necessary condition for $\mathcal{S}_A$ to be a rewrite strategy?

7.17 Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{aligned}
0+x &\xrightarrow{\pi_1} x & \qquad x<0 &\xrightarrow{\lambda_1} \mathsf{F}\\
\mathsf{s}(y)+x &\xrightarrow{\pi_2} \mathsf{s}(y+x) & \qquad 0<\mathsf{s}(x) &\xrightarrow{\lambda_2} \mathsf{T}\\
x-0 &\xrightarrow{\mu_1} x & \qquad \mathsf{s}(x)<\mathsf{s}(y) &\xrightarrow{\lambda_3} x<y\\
0-x &\xrightarrow{\mu_2} 0 & \qquad \mathsf{if}(\mathsf{T},x,y) &\xrightarrow{\iota_1} x\\
\mathsf{s}(x)-\mathsf{s}(y) &\xrightarrow{\mu_3} x-y & \qquad \mathsf{if}(\mathsf{F},x,y) &\xrightarrow{\iota_2} y\\
x \div y &\xrightarrow{\delta} \mathsf{if}(x<y,0,\mathsf{s}((x-y)\div y)) & \qquad \mathsf{if}(x,y,y) &\xrightarrow{\iota_3} y
\end{aligned}$$

and the strategy annotation A given by

$$A(+) = [1,\pi_1,2,\pi_2] \qquad A(-) = [2,\mu_1,\mu_3,1] \qquad A(\div) = [\delta] \qquad A(<) = [1,2,\lambda_1,\lambda_2,\lambda_3]$$
$$A(\mathsf{if}) = [\iota_3,2,\iota_2,1,3,\iota_1,\iota_3]$$

a Determine whether A is full and in-time.

b Construct a strategy annotation for $\mathcal{R}$ that is both in-time and full.

c Compute $\mathrm{redex}_A(\mathsf{if}(\mathsf{T}, 0 - \mathsf{s}(0), \mathsf{s}(0) - (0 \div 0)))$.

7.18 Prove that we may assume without loss of generality that in-time strategy annotations do not contain duplicates.

7.19 Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{array}{lll} 0 \oplus x \xrightarrow{\alpha_1} x & x \wedge 0 \xrightarrow{\beta_1} 0 & \neg x \xrightarrow{\gamma} x \oplus 1 \\ 1 \oplus 0 \xrightarrow{\alpha_2} \neg 1 & x \wedge 1 \xrightarrow{\beta_2} x & x \vee y \xrightarrow{\delta} \neg(\neg x \wedge \neg y) \\ x \oplus x \xrightarrow{\alpha_3} 0 & x \wedge x \xrightarrow{\beta_3} x \oplus 0 & \end{array}$$

and the strategy annotation A given by

$$\begin{array}{lll} A(\oplus) = [1, \alpha_1, 2, \alpha_2, \alpha_3] & A(\neg) = [\gamma, 1] & A(0) = A(1) = [\,] \\ A(\wedge) = [2, \beta_3, \beta_1, \beta_2, 1] & A(\vee) = [1, \delta, 2] & \end{array}$$

a Compute $\mathrm{normalize}_A((0 \wedge 0) \wedge \neg 0)$.

b Give a full strategy annotation B such that $\mathrm{normalize}_B((0 \oplus 0) \oplus (0 \oplus 0)) \neq 0$.

7.20 Suppose we modify strategy annotations such that rules are no longer allowed but instead an entry 0 in $A(f)$ signals that all rewrite rules that define f are tried. So $\mathrm{redex}_A(t)$ returns a non-empty set of pairs $(p, \ell \to r)$, or $\bot$.

a Give a formal definition of $\mathrm{redex}'_A(t, [0 \mid L])$.

b Show that Theorem 7.4.13 does not hold for these modified strategy annotations.

7.5 Context-Sensitive Rewriting

In this section we consider a simple mechanism to control where rewrite steps might take place. Like the strategy annotations of the preceding section, for each function symbol we specify which argument positions are eligible, but without imposing an order among them. Also, rewrite steps at the root of terms are always possible, without restrictions on the rewrite rules.

Definition 7.5.1. Let $\mathcal{F}$ be a signature. A *replacement map* μ for $\mathcal{F}$ associates with every n-ary function symbol in $\mathcal{F}$ a subset $\mu(f)$ of $\{1, \ldots, n\}$. The set $\mathcal{P}\mathsf{os}^\mu(t)$ of *active* positions in $t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ is inductively defined as follows:

$$\mathcal{P}\mathsf{os}^\mu(t) = \begin{cases} \{\epsilon\} & \text{if } t \text{ is a variable} \\ \{\epsilon\} \cup \{ip \mid i \in \mu(f),\ 1 \leqslant i \leqslant n, \text{ and } p \in \mathcal{P}\mathsf{os}^\mu(t_i)\} & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

We write $\mathcal{P}\mathsf{os}^{\neg\mu}(t)$ for $\mathcal{P}\mathsf{os}(t) \setminus \mathcal{P}\mathsf{os}^\mu(t)$ and $\mathcal{P}\mathsf{os}^\mu_\mathcal{V}(t)$ for $\mathcal{P}\mathsf{os}_\mathcal{V}(t) \cap \mathcal{P}\mathsf{os}^\mu(t)$.

In examples we do not specify $\mu(c)$ for constants c since there is no choice: $\mu(c) = \varnothing$. Replacement maps partition terms in an active part, where rewrite steps are allowed, and an inactive part where rewrite steps are forbidden.

Definition 7.5.2. A *context-sensitive* rewrite system (CSRS for short) $(\mathcal{R}, \mu)$ consists of a TRS $\mathcal{R}$ over a signature $\mathcal{F}$ and a replacement μ for $\mathcal{F}$. The relation $\to_\mu$ is defined as follows: $s \to_\mu t$ if $s \to_p t$ with $p \in \mathcal{P}\mathsf{os}^\mu(s)$.

Despite the name, only the function symbols in the context along the path to the root of the term determine whether a redex in a term may be contracted in a context-sensitive rewrite step.

When we speak of confluence or termination or any other property of a CSRS $(\mathcal{R},\mu)$, we always refer to the relation $\to_\mu$.

Example 7.5.3. Consider the TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{array}{lll} \mathsf{nats} \to \mathsf{adi}(\mathsf{zeros}) & \mathsf{adi}(x:y) \to \mathsf{inc}(x:\mathsf{adi}(y)) & \mathsf{hd}(x:y) \to x \\ \mathsf{zeros} \to \mathsf{0}:\mathsf{zeros} & \mathsf{inc}(x:y) \to \mathsf{s}(x):\mathsf{inc}(y) & \mathsf{tl}(x:y) \to x \end{array}$$

and the following replacement maps μ and ν:

$$\begin{array}{ll} \mu(:) = \mu(\mathsf{s}) = \varnothing & \mu(\mathsf{adi}) = \mu(\mathsf{inc}) = \mu(\mathsf{hd}) = \mu(\mathsf{tl}) = \{1\} \\ \nu(:) = \nu(\mathsf{adi}) = \nu(\mathsf{tl}) = \varnothing & \nu(\mathsf{s}) = \nu(\mathsf{hd}) = \nu(\mathsf{inc}) = \{1\} \end{array}$$

We have

$$\mathsf{nats} \to_\mu \mathsf{adi}(\mathsf{zeros}) \to_\mu \mathsf{adi}(\mathsf{0}:\mathsf{zeros}) \to_\mu \mathsf{inc}(\mathsf{0}:\mathsf{adi}(\mathsf{zeros})) \to_\mu \mathsf{s}(\mathsf{0}):\mathsf{inc}(\mathsf{adi}(\mathsf{zeros}))$$

and this is the only (maximal) rewrite sequence starting from nats in $(\mathcal{R},\mu)$. The final term $t = \mathsf{s}(\mathsf{0}):\mathsf{inc}(\mathsf{adi}(\mathsf{zeros}))$ is a normal form of $(\mathcal{R},\mu)$ since $\mathcal{P}\mathsf{os}^\mu(t) = \{\epsilon\}$ and t is not a redex. In $(\mathcal{R},\nu)$ we have $\mathsf{nats} \to_\nu \mathsf{adi}(\mathsf{zeros})$ with normal form $\mathsf{adi}(\mathsf{zeros})$.

Since $\to_\mu$ is a restriction of $\to$, a natural question is whether computations with $\to$ can be simulated with $\to_\mu$. Below we present a sufficient condition for left-linear TRSs and replacement maps that make all function symbols in left-hand sides active.

Definition 7.5.4. A replacement map μ is *compatible* with a term t if $\mathcal{P}\mathsf{os}_\mathcal{F}(t) \subseteq \mathcal{P}\mathsf{os}^\mu(t)$. We say that μ is *compatible* with a TRS $\mathcal{R}$ if μ is compatible with the left-hand side of every rewrite rule in $\mathcal{R}$.

Example 7.5.5. Consider the term $\mathsf{f}(\mathsf{f}(\mathsf{a},\mathsf{g}(x)),y)$. The replacement map μ defined by $\mu(\mathsf{f}) = \{1,2\}$ and $\mu(\mathsf{g}) = \varnothing$ is compatible with t because $\mathcal{P}\mathsf{os}^\mu(t) = \{\epsilon,1,11,12,2\} \supseteq \{\epsilon,1,11,12\} = \mathcal{P}\mathsf{os}_\mathcal{F}(t)$. The replacement map ν defined by $\nu(\mathsf{f}) = \nu(\mathsf{g}) = \{1\}$ is not compatible with t as $12 \notin \mathcal{P}\mathsf{os}^\nu(t)$.

We find it convenient to view a replacement map μ as a set consisting of the pairs (f,i) for which $i \in \mu(f)$. So in the preceding example we have $\mu = \{(\mathsf{f},1),(\mathsf{f},2)\}$ and $\nu = \{(\mathsf{f},1),(\mathsf{g},1)\}$.

Lemma 7.5.6. *Every* TRS $\mathcal{R}$ *admits a unique minimal compatible replacement map.*

Proof Define

$$\mu_\mathsf{c} = \{(f,i) \mid t|_i \notin \mathcal{V} \text{ for some subterm } t \text{ of a left-hand side of } \mathcal{R} \text{ with } f = \mathsf{root}(t)\}$$

By construction μ_c is a compatible replacement map for $\mathcal{R}$. Let ν be any compatible replacement map for $\mathcal{R}$. We show $\mu_\mathsf{c} \subseteq \nu$. So let $(f,i) \in \mu_\mathsf{c}$. There exists a subterm t of

a left-hand side ℓ of $\mathcal{R}$ such that $\mathsf{root}(t) = f$ and $t|_i \notin \mathcal{V}$. Let p be the position of t in ℓ, so $pi \in \mathcal{P}\mathsf{os}_{\mathcal{F}}(\ell)$. Because ν is compatible with $\mathcal{R}$ we have $\mathcal{P}\mathsf{os}_{\mathcal{F}}(\ell) \subseteq \mathcal{P}\mathsf{os}^{\nu}(t)$ and thus $pi \in \mathcal{P}\mathsf{os}^{\nu}(t)$. Hence $(f,i) \in \nu$. □

The replacement map μ_{c} defined in the previous proof is called the *canonical* replacement map. So a replacement ν is compatible with a TRS $\mathcal{R}$ if and only if $\mu_{\mathsf{c}} \subseteq \nu$.

Notation. We write $\to_{\neg\mu}$ for the difference of $\to$ and $\to_{\mu}$.

Lemma 7.5.8 below is a useful result concerning compatible replacement maps for left-linear TRSs. In the proof we make use of the following result for parallel rewriting.

Lemma 7.5.7. *Let $\mathcal{R}$ be a* TRS*. If $s \mathrel{+\!\!\!\!\rightarrow}^{P} \ell\tau$ for a linear term ℓ with $\mathcal{P}\mathsf{os}_{\mathcal{F}}(\ell) \cap P = \varnothing$ then there exists a substitution σ such that $s = \ell\sigma$ and $\sigma \mathrel{+\!\!\!\!\rightarrow} \tau$.*

Proof Exercise 7.22. □

Lemma 7.5.8. *Let $\mathcal{R}$ be a left-linear* TRS*. If μ is a compatible replacement map for $\mathcal{R}$ then $\mathrel{+\!\!\!\!\rightarrow}_{\neg\mu} \cdot \to_{\mu} \subseteq \to_{\mu}^{+} \cdot \mathrel{+\!\!\!\!\rightarrow}_{\neg\mu}$.*

Proof Let $s \mathrel{+\!\!\!\!\rightarrow}_{\neg\mu} t \to_{\mu}^{q} u$ with $P \subseteq \mathcal{P}\mathsf{os}^{\neg\mu}(s)$ the set of pairwise parallel positions of the redexes contracted in the parallel step $s \mathrel{+\!\!\!\!\rightarrow}_{\neg\mu} t$. We show $s \to_{\mu} \cdot \mathrel{+\!\!\!\!\rightarrow} u$ by induction on q. Since $\mathrel{+\!\!\!\!\rightarrow} \subseteq \mathrel{+\!\!\!\!\rightarrow}_{\mu} \cdot \mathrel{+\!\!\!\!\rightarrow}_{\neg\mu}$, the desired conclusion $s \to_{\mu}^{+} \cdot \mathrel{+\!\!\!\!\rightarrow}_{\neg\mu} u$ follows.

▷ If $q = \epsilon$ then $t = \ell\tau$ and $u = r\tau$ for some rule $\ell \to r \in \mathcal{R}$ and substitution τ. Since μ is compatible, $\mathcal{P}\mathsf{os}_{\mathcal{F}}(\ell) \subseteq \mathcal{P}\mathsf{os}^{\mu}(t)$. Because ℓ is linear and $P \subseteq \mathcal{P}\mathsf{os}^{\neg\mu}(t)$, we can use Lemma 7.5.7 to obtain a substitution σ such that $s = \ell\sigma$ and $\sigma \mathrel{+\!\!\!\!\rightarrow} \tau$. Hence $s \to_{\mu} r\sigma \mathrel{+\!\!\!\!\rightarrow} r\tau = u$ by Lemma 6.1.5.

▷ Suppose $q = iq'$. Since $\epsilon \in \mathcal{P}\mathsf{os}^{\mu}(s)$, $\epsilon \notin P$ and thus we may write

$$s = f(s_1, \ldots, s_i, \ldots, s_n) \qquad t = f(t_1, \ldots, t_i, \ldots, t_n) \qquad u = f(t_1, \ldots, u', \ldots, t_n)$$

with $s_j \mathrel{+\!\!\!\!\rightarrow}_{\neg\mu} t_j$ for all $1 \leqslant j \leqslant n$ and $t_i \to_{\mu}^{q'} u'$. The induction hypothesis yields $s_i \to_{\mu} v \mathrel{+\!\!\!\!\rightarrow} u'$ for some term v. Hence $s \to_{\mu} f(s_1, \ldots, v, \ldots, s_n) \mathrel{+\!\!\!\!\rightarrow} u$ as desired. □

The following consequence is obtained by a straightforward induction argument.

Corollary 7.5.9. *Let $\mathcal{R}$ be a left-linear* TRS*. If μ is a compatible replacement map for $\mathcal{R}$ then $\to_{\neg\mu}^{*} \cdot \to_{\mu}^{+} \subseteq \to_{\mu}^{+} \cdot \to_{\neg\mu}^{*}$.*

Moreover, under the same conditions, any rewrite sequence can be factorized into active steps and forbidden steps.

Corollary 7.5.10. *Let $\mathcal{R}$ be a left-linear* TRS*. If μ is a compatible replacement map for $\mathcal{R}$ then $\to^{*} \subseteq \to_{\mu}^{*} \cdot \to_{\neg\mu}^{*}$.*

Lemma 7.5.11. *If μ is a compatible replacement map for a left-linear* TRS *$\mathcal{R}$ then every μ-normal form is root-stable.*

Proof Let t be a term that is not root-stable. We show that t is not a μ-normal form. We have $t \to^* \ell\sigma \to_\epsilon r\sigma$ for some rewrite rule $\ell \to r \in \mathcal{R}$ and substitution σ by the definition of root-stability. Clearly $\ell\sigma \to_\mu r\sigma$. We obtain $t \to_\mu \cdot \to^* r\sigma$ with help of the preceding corollaries. Hence t is not a μ-normal form. □

The next example shows that we cannot dispense with left-linearity in Lemma 7.5.11.

Example 7.5.12. Consider the CSRS $(\mathcal{R}, \mu_{\mathsf{c}})$ with $\mathcal{R} = \{\mathsf{f}(x,x) \to \mathsf{a}\}$. The term $t = \mathsf{f}(\mathsf{a},\mathsf{f}(\mathsf{a},\mathsf{a}))$ is a μ_{c}-normal form but not root-stable as $t \to_{\mathcal{R}} \mathsf{f}(\mathsf{a},\mathsf{a}) \to_{\mathcal{R}} \mathsf{a}$.

Theorem 7.5.13. *Let $(\mathcal{R}, \mu)$ be a* CSRS *such that $\mathcal{R}$ is left-linear and μ is compatible. If $s \to_{\mathcal{R}}^* t$ and $\mathcal{P}\mathsf{os}(t) = \mathcal{P}\mathsf{os}^\mu(t)$ then $s \to_\mu^* t$.*

Proof We obtain $s \to_\mu^* u \to_{\neg\mu}^* t$ from Corollary 7.5.10. We use induction on t. First suppose $t \in \mathcal{V}$. Since forbidden steps take place below the root, the sequence from u to t is empty and thus $s \to_\mu^* u = t$. Next consider $t = f(t_1, \ldots, t_n)$. We have $u = f(u_1, \ldots, u_n)$ with $u_i \to^* t_i$ for all $1 \leqslant i \leqslant n$ and thus $u_i \to_\mu^* t_i$ by the induction hypothesis. From the assumption $\mathcal{P}\mathsf{os}(t) \subseteq \mathcal{P}\mathsf{os}^\mu(t)$ we infer $\mu(f) = \{1, \ldots, n\}$. Hence $u \to_\mu^* t$ and therefore $s \to_\mu^* t$. □

The condition $\mathcal{P}\mathsf{os}(t) = \mathcal{P}\mathsf{os}^\mu(t)$ in the above theorem is problematic since normal forms of the term s are not known in advance. For computing ground constructor terms, which represent values, a sufficient condition is to demand $\mu(c) = \{1, \ldots, n\}$ for every n-ary constructor symbol.

In Section 7.2 we have seen that the leftmost outermost strategy is normalizing for left-normal orthogonal TRSs. In the remainder of this section we use *usable* replacement maps to present a larger class of orthogonal TRSs for which the leftmost outermost strategy is normalizing for a restricted set of terms. We use the following variant of the TRS of Table 3.5 to illustrate the subsequent definitions.

Example 7.5.14. Consider the orthogonal TRS $\mathcal{R}$ consisting of the rewrite rules

$$\begin{aligned}
\mathsf{primes}(x) &\to \mathsf{take}(x, \mathsf{sieve}(\mathsf{from}(\mathsf{s}(\mathsf{s}(0))))) & \mathsf{take}(0, y) &\to \mathsf{nil}\\
\mathsf{sieve}(0 : y) &\to \mathsf{sieve}(y) & \mathsf{take}(\mathsf{s}(x), y : z) &\to y : \mathsf{take}(x, z)\\
\mathsf{sieve}(\mathsf{s}(x) : y) &\to \mathsf{s}(x) : \mathsf{sieve}(\mathsf{filter}(x, y, x)) & \mathsf{filter}(0, y : z, w) &\to 0 : \mathsf{filter}(w, z, w)\\
\mathsf{from}(x) &\to x : \mathsf{from}(\mathsf{s}(x)) & \mathsf{filter}(\mathsf{s}(x), y : z, w) &\to y : \mathsf{filter}(x, z, w)
\end{aligned}$$

Since $\mathcal{R}$ is not left-normal, Theorem 7.2.22 is not applicable and hence it is not clear whether $\mathcal{S}_{\mathsf{lo}}$ is normalizing for $\mathcal{R}$.

Definition 7.5.15. We write $\mathcal{T}(\mu)$ for the set of all terms t such that $(f, i) \in \mu$ whenever $f(t_1, \ldots, t_n) \trianglelefteq t$ and t_i is reducible.

So $\mathcal{T}(\mu)$ consists of the terms t such that all redex positions in t are active.

Definition 7.5.16. A term t is called *basic* if $\mathcal{P}\mathsf{os}_{\mathcal{D}}(t) = \{\epsilon\}$ for $\mathcal{D} = \{\mathsf{root}(\ell) \mid \ell \to r \in \mathcal{R}\}$. A replacement map μ is *usable* if $t \in \mathcal{T}(\mu)$ for all basic terms s and $s \to^* t$.

Definition 7.5.17. The operator $\Phi_{\mathcal{R}}$ on replacement maps for TRSs $\mathcal{R}$ is defined as follows:

$$\Phi_{\mathcal{R}}(\mu) = \{(f,i) \mid \ell \to C[f(r_1,\ldots,r_n)] \in \mathcal{R} \text{ and } \mathsf{CAP}^{\ell}_{\mu}(r_i) \neq r_i\}$$

with

$$\mathsf{CAP}^{\ell}_{\mu}(t) = \begin{cases} t & \text{if } t = \ell|_p \text{ for some } p \notin \mathcal{P}\mathsf{os}^{\mu}(\ell) \\ u & \text{if } t = g(t_1,\ldots,t_m) \text{ and } u \text{ does not unify with a left-hand side of } \mathcal{R} \\ x & \text{otherwise} \end{cases}$$

where $u = g(\mathsf{CAP}^{\ell}_{\mu}(t_1),\ldots,\mathsf{CAP}^{\ell}_{\mu}(t_m))$ and x is a fresh variable.

The choice of the variable x in the otherwise case is irrelevant, but different calls to $\mathsf{CAP}^{\ell}_{\mu}$ must use different fresh variables.

Example 7.5.18. Consider the rule $\ell = \mathsf{sieve}(\mathsf{s}(x) : y) \to \mathsf{s}(x) : \mathsf{sieve}(\mathsf{filter}(x,y,x)) = r$ and let $\mu = \varnothing$. We compute $\mathsf{CAP}^{\ell}_{\mu}(r)$ in a bottom-up manner, according to the table

position p	11	1	212	21	2	ϵ
$t = r\|_p$	x	$\mathsf{s}(x)$	y	$\mathsf{filter}(x,y,x)$	$\mathsf{sieve}(\mathsf{filter}(x,y,x))$	r
$\mathsf{CAP}^{\ell}_{\mu}(t)$	x	$\mathsf{s}(x)$	y	a	b	$\mathsf{s}(x) : b$
u				$\mathsf{filter}(x,y,x)$	$\mathsf{sieve}(a)$	$\mathsf{s}(x) : b$

Here a and b are variables. Hence $(\mathsf{sieve},1),(:,2) \in \Phi_{\mathcal{R}}(\mu)$.

The following lemma connects Definitions 7.5.17 and 7.5.15.

Lemma 7.5.19. *If $s\sigma \in \mathcal{T}(\mu)$ and $\mathsf{CAP}^{s}_{\mu}(t) = t$ then $t\sigma$ is a normal form.*

Proof Let $s\sigma \in \mathcal{T}(\mu)$ and $\mathsf{CAP}^{s}_{\mu}(t) = t$. We prove that $t\sigma$ is a normal form by induction on t and distinguish two cases. If $t = s|_p$ with $p \in \mathcal{P}\mathsf{os}(s) \setminus \mathcal{P}\mathsf{os}^{\mu}(s)$ then $t\sigma$ is a subterm of $s\sigma$ at an inactive position. Hence $t\sigma$ is a normal form. Suppose $t = f(t_1,\ldots,t_n)$. Because $\mathsf{CAP}^{s}_{\mu}(t) = t$, we must have $u = f(\mathsf{CAP}^{s}_{\mu}(t_1),\ldots,\mathsf{CAP}^{s}_{\mu}(t_n)) = t$ and u does not unify with a left-hand side of $\mathcal{R}$. According to the induction hypothesis, $t_i\sigma$ is a normal form for $1 \leqslant i \leqslant n$. Since t does not unify with a left-hand side of $\mathcal{R}$, $t\sigma$ does not match the left-hand side of a rule of $\mathcal{R}$. We conclude that $t\sigma$ is a normal form. □

Lemma 7.5.20. *If $\ell \to r \in \mathcal{R}$ and $\ell\sigma \in \mathcal{T}(\mu)$ then $r\sigma \in \mathcal{T}(\mu \cup \Phi_{\mathcal{R}}(\mu))$.*

Proof Let $f(t_1,\ldots,t_n) \trianglelefteq r\sigma$ with reducible argument t_i. We show $(f,i) \in \mu \cup \Phi_{\mathcal{R}}(\mu)$. Let $p \in \mathcal{P}\mathsf{os}(r\sigma)$ such that $(r\sigma)|_p = f(t_1,\ldots,t_n)$. We distinguish two cases. If p is below a variable position in r then $f(t_1,\ldots,t_n) \trianglelefteq \ell\sigma$ and thus $(f,i) \in \mu$ follows from $\ell\sigma \in \mathcal{T}(\mu)$. Next suppose $p \in \mathcal{P}\mathsf{os}_{\mathcal{F}}(r\sigma)$. We may write $r|_p = f(r_1,\ldots,r_n)$ with $r_i\sigma = t_i$. From Lemma 7.5.19 (with $s = \ell$ and $t = r_i$) we obtain $\mathsf{CAP}^{\ell}_{\mu}(r_i) \neq r_i$. Hence $(f,i) \in \Phi_{\mathcal{R}}(\mu)$ according to the definition of $\Phi_{\mathcal{R}}$. □

The operator $\Phi_{\mathcal{R}}$ is monotone, according to the following lemma.

Lemma 7.5.21. *If $\mu \subseteq \nu$ then $\Phi_{\mathcal{R}}(\mu) \subseteq \Phi_{\mathcal{R}}(\nu)$.*

Proof Consider $(f,i) \in \Phi_{\mathcal{R}}(\mu)$. So $\ell \to C[f(r_1,\ldots,r_n)] \in \mathcal{R}$ and $\mathsf{CAP}^{\ell}_{\mu}(r_i) \neq r_i$. We need to show $\mathsf{CAP}^{\ell}_{\nu}(r_i) \neq r_i$ in order to conclude $(f,i) \in \Phi_{\mathcal{R}}(\nu)$. This follows from the more general statement

$$\mathsf{CAP}^{\ell}_{\nu}(t) = t \implies \mathsf{CAP}^{\ell}_{\mu}(t) = t$$

which we prove by induction on the term t.

- ▷ If $t \in \mathcal{V}$ then the first case of the definition of $\mathsf{CAP}^{\ell}_{\nu}$ must apply and thus $t = \ell|_p$ for some $p \notin \mathcal{P}\mathsf{os}^{\nu}(\ell)$. Since $\mu \subseteq \nu$, $\mathcal{P}\mathsf{os}^{\mu}(\ell) \subseteq \mathcal{P}\mathsf{os}^{\nu}(\ell)$ and thus $p \in \mathcal{P}\mathsf{os}^{\mu}(\ell)$. Hence $\mathsf{CAP}^{\ell}_{\mu}(t) = t$.
- ▷ Suppose $t = g(t_1,\ldots,t_m)$. Since $\mathsf{CAP}^{\ell}_{\nu}(t) = t$ the first or second case of the definition of $\mathsf{CAP}^{\ell}_{\nu}$ is used. For the former possibility we reason exactly as in the base case. We consider the second case, so let $u = g(\mathsf{CAP}^{\ell}_{\nu}(t_1),\ldots,\mathsf{CAP}^{\ell}_{\nu}(t_m))$. By assumption, u does not unify with the left-hand side of a rule and thus $\mathsf{CAP}^{\ell}_{\nu}(t) = u = t$. Hence $\mathsf{CAP}^{\ell}_{\nu}(t_j) = t_j$ for all $1 \leqslant j \leqslant m$. The induction hypothesis yields $\mathsf{CAP}^{\ell}_{\mu}(t_j) = t_j$ for all $1 \leqslant j \leqslant m$. Consequently, $u' = g(\mathsf{CAP}^{\ell}_{\mu}(t_1),\ldots,\mathsf{CAP}^{\ell}_{\mu}(t_m)) = t$ and we obtain $\mathsf{CAP}^{\ell}_{\mu}(t) = t$ by the second case of the definition. □

Definition 7.5.22. We define the replacement map μ_{f} as the least fixed point of $\Phi_{\mathcal{R}}$.

The well-definedness of μ_{f} follows from the monotonicity of $\Phi_{\mathcal{R}}$.

Lemma 7.5.23. *If $s \in \mathcal{T}(\mu_{\mathsf{f}})$ and $s \to t$ then $t \in \mathcal{T}(\mu_{\mathsf{f}})$.*

Proof Let $s \to_p t$. Since $\mathcal{T}(\mu_{\mathsf{f}})$ is closed under subterms, $s|_p \in \mathcal{T}(\mu_{\mathsf{f}})$. Lemma 7.5.20 yields $t|_p \in \mathcal{T}(\mu_{\mathsf{f}} \cup \Phi_{\mathcal{R}}(\mu_{\mathsf{f}}))$. Since μ_{f} is a fixed point of $\Phi_{\mathcal{R}}$, $\mu_{\mathsf{f}} \cup \Phi_{\mathcal{R}}(\mu_{\mathsf{f}}) = \mu_{\mathsf{f}}$ and hence $t|_p \in \mathcal{T}(\mu_{\mathsf{f}})$. From $s \in \mathcal{T}(\mu_{\mathsf{f}})$ we obtain $p \in \mathcal{P}\mathsf{os}^{\mu_{\mathsf{f}}}(s)$. Hence $t = s[t|_p]_p \in \mathcal{T}(\mu_{\mathsf{f}})$. □

Corollary 7.5.24. *The replacement map μ_{f} is usable.*

Definition 7.5.25. A term t is called left-normal with respect to a replacement map μ, or simply *μ-left-normal*, if $p \in \mathcal{P}\mathsf{os}^{\mu}_{\mathcal{V}}(t)$ and $q \in \mathcal{P}\mathsf{os}^{\mu}(t)$ with $p <_{\mathsf{left}} q$ imply $q \in \mathcal{P}\mathsf{os}_{\mathcal{V}}(t)$. Let $\mathcal{R}$ be a TRS with a usable replacement map μ. The TRS $\mathcal{R}$ is *basically left-normal* if the left-hand side ℓ of every rule $\ell \to r \in \mathcal{R}$ is μ-left-normal.

A left-normal TRS is μ-left-normal for any replacement map μ. As a consequence, left-normal TRSs are basically left-normal. The reverse is not true, as shown in the following example.

Example 7.5.26. The replacement map μ given by

$$\mu(\mathsf{primes}) = \mu(\mathsf{from}) = \mu(\mathsf{head}) = \mu(\mathsf{tail}) = \mu(\mathsf{s}) = \mu(\mathsf{0}) = \varnothing \qquad \mu(\mathsf{sieve}) = \{1\}$$
$$\mu(\mathsf{filter}) = \mu(\mathsf{:}) = \{2\}$$

is a usable replacement map for the TRS $\mathcal{R}$ of Table 3.5. For example, consider the left-hand side $\ell = \mathsf{filter}(\mathsf{s}(x), y : z, w)$. There are no active positions $p \in \mathcal{P}\mathsf{os}^{\mu}_{\mathcal{V}}(\ell) = \{22\}$ and $q \in \mathcal{P}\mathsf{os}^{\mu}(\ell) = \{\epsilon, 2, 22\}$ such that $p <_{\mathsf{left}} q$. So ℓ is μ-left-normal. In a similar way one can verify basic left-normality of all other left-hand sides of $\mathcal{R}$. Hence, $\mathcal{R}$ is basically left-normal with respect to μ.

Theorem 7.5.27. *The leftmost outermost strategy is basically hyper-normalizing for all basically left-normal orthogonal* TRS*s.*

Exercises

7.21 Let $(\mathcal{R}, \mu)$ be a CSRS.

a Prove that $\to^{\mu}_{\mathcal{R}}$ is closed under substitutions.

b Is $\to^{\mu}_{\mathcal{R}}$ closed under contexts?

7.22 Prove Lemma 7.5.7.

7.23 Compute μ_{c} and μ_{f} for the TRS of Example 7.5.14.

7.24 Consider Lemma 7.5.11.

a Prove that compatibility of μ is not a necessary condition.

b Suppose we generalize the definition of compatible replacement map such that non-linear variable occurrences in left-hand sides become active: For every rewrite rule $\ell \to r \in \mathcal{R}$ and position $p \in \mathcal{P}\mathsf{os}(\ell)$, if $\ell|_p \in \mathcal{F}$ or both $\ell|_p \in \mathcal{V}$ and $\ell|_p = \ell|_q$ for some position $q \neq p$ then $p \in \mathcal{P}\mathsf{os}^{\mu}(\ell)$. Is this sufficient to remove the left-linearity requirement?

7.25 Can Theorem 7.5.13 be strengthened by replacing the condition $\mathcal{P}\mathsf{os}(t) = \mathcal{P}\mathsf{os}^{\mu}(t)$ with $\mathcal{P}\mathsf{os}_{\mathcal{F}}(t) \subseteq \mathcal{P}\mathsf{os}^{\mu}(t)$?

7.26 Let $\mathcal{F}$ be a signature and let μ be a replacement map for $\mathcal{F}$. A *μ-monotone* $\mathcal{F}$-algebra $(\mathcal{A}, >)$ consists of a non-empty $\mathcal{F}$-algebra $\mathcal{A}$ and a proper order $>$ on the carrier A of $\mathcal{A}$ such that every algebra operation is strictly monotone in all active coordinates, i.e., if $f \in \mathcal{F}$ has arity $n \geqslant 1$ then $f_{\mathcal{A}}(a_1, \ldots, a_i, \ldots, a_n) > f_{\mathcal{A}}(a_1, \ldots, b, \ldots, a_n)$ for all $a_1, \ldots, a_n, b \in A$ and $i \in \mu(f)$ with $a_i > b$.

a Prove that a CSRS $(\mathcal{R}, \mu)$ is terminating if and only if $\mathcal{R}$ is compatible with $>_{\mathcal{A}}$ for a well-founded μ-monotone $\mathcal{F}$-algebra $(\mathcal{A}, >)$.

b Prove the termination of the CSRS $(\mathcal{R}, \mu_{\mathsf{c}})$ with $\mathcal{R}$ consisting of the rewrite rules

$$\begin{array}{lll}
\mathsf{inc}(x : y) \to \mathsf{s}(x) : \mathsf{inc}(y) & \mathsf{zip}(\mathsf{nil}, x) \to \mathsf{nil} & \mathsf{zip}(x : u, y : v) \to \mathsf{pair}(x, y) : \mathsf{zip}(u, v) \\
\mathsf{from}(x) \to x : \mathsf{from}(\mathsf{s}(x)) & \mathsf{zip}(x, \mathsf{nil}) \to \mathsf{nil} & \mathsf{tail}(x : y) \to y \\
\mathsf{even} \to 0 : \mathsf{inc}(\mathsf{odd}) & \mathsf{rep}(\mathsf{nil}) \to \mathsf{nil} & \mathsf{rep}(x : y) \to x : (x : \mathsf{rep}(y)) \\
\mathsf{odd} \to \mathsf{inc}(\mathsf{even}) & \mathsf{take}(0, x) \to \mathsf{nil} & \mathsf{take}(\mathsf{s}(n), x : y) \to x : \mathsf{take}(n, y)
\end{array}$$

7.27 The replacement map μ_{i} is defined as $\Phi_{\mathcal{R}}(\varnothing)$.

a Let $s \in \mathcal{T}(\mu_{\mathsf{i}})$ and $s \xrightarrow{\mathsf{i}} t$. Show that $t \in \mathcal{T}(\mu_{\mathsf{i}})$.

b Prove that a TRS $\mathcal{R}$ is innermost terminating if the CSRS $(\mathcal{R}, \mu_{\mathsf{i}})$ is terminating.

c Does the converse of part (b) hold?

7.28 Is it decidable whether a replacement map is usable?

Bibliographic Notes

In the literature the maximal outermost (innermost) strategy is often referred to as *parallel* outermost (innermost). Restricting the maximal strategy to orthogonal TRSs gives a strategy that is known as *full-substitution* or the Gross–Knuth strategy. Corollary 7.2.5 and Theorem 7.2.12 are from Gramlich [47] and generalize earlier results for orthogonal TRSs by O'Donnell [99]. Theorems 7.2.16 and 7.2.22 are from [99]. Example 7.2.17 and Theorem 7.3.4 are from Huet and Lévy [60]. Section 7.4 is based on van de Pol [109, 110]. Theorem 7.4.13 is from [111]. Context-sensitive rewriting was introduced by Lucas [85]. In Section 7.5 we only touch upon the topic. Further results can be found in the survey papers [86, 87]. Theorem 7.5.27 is from [54].

Part III

Appendices

Appendix A

Mathematical Background

In this chapter we recall basic definitions and results concerning relations and orders. In Section A.1 binary relations and orders are defined. Section A.2 covers well-founded and transfinite induction and Section A.3 is devoted to multiset orders.

A.1 Relations

In this section we recall basic facts about relations and orders. The presented material is standard and can be found in numerous textbooks.

Definition A.1.1. A (binary) *relation* R on a Cartesian product $A \times B$ is a subset of $A \times B$. A relation on a set A is a subset of $A \times A$. Instead of $(a, b) \in R$ we usually write $a\ R\ b$. The *empty* relation $\varnothing$ on $A \times B$ is the empty set. The *identity* relation id_A on A is the set $\{(a, a) \mid a \in A\}$. The *domain* $\mathcal{D}\mathsf{om}(R)$ of a relation R on $A \times B$ is the set $\{a \in A \mid a\ R\ b \text{ for some } b \in B\}$. Its *range* $\mathsf{ran}(R)$ is defined as $\{b \in B \mid a\ R\ b \text{ for some } a \in A\}$.

We introduce two important operations on (binary) relations.

Definition A.1.2. Let R be a relation on $A \times B$. Its *converse* or *inverse* R^{-1} is the relation on $B \times A$ defined by $b\ R^{-1}\ a$ if and only if $a\ R\ b$. Let S be a relation on $B \times C$. The *composition* $R \cdot S$ of R and S is the relation on $A \times C$ defined by $a\ R \cdot S\ c$ if and only if there exists a $b \in B$ such that $a\ R\ b$ and $b\ S\ c$.

When possible, we mirror the denotation of the relation to denote its inverse. For instance, we write $\leqslant$ instead of $\geqslant^{-1}$ and $\leftarrow$ instead of $\rightarrow^{-1}$. The exercises contain many easy results on the interplay between identity, inverse, composition, and other (set) operations on relations.

Let R be a relation on $A \times B$, S a relation on $B \times C$, and b an arbitrary element of B. We find it convenient to denote the subset $\{(a, c) \mid a\ R\ b \text{ and } b\ S\ c\}$ of $R \cdot S$ by $R\ b\ S$.

Lemma A.1.3. *Composition of relations is associative, i.e.,* $(R \cdot S) \cdot T = R \cdot (S \cdot T)$ *for all relations* R *on* $A \times B$, S *on* $B \times C$ *and* T *on* $C \times D$.

So there is no need to write parentheses in expressions like $R \cdot S \cdot T$. The proof of Lemma A.1.3 is left to the reader (Exercise A.3(b)).

Definition A.1.4. A relation R on a set A is called *reflexive* if $a\ R\ a$ for all $a \in A$, *irreflexive* if there is no $a \in A$ such that $a\ R\ a$, *transitive* if $a\ R\ c$ whenever $a\ R\ b$ and $b\ R\ c$, for all $a, b, c \in A$, *symmetric* if $a\ R\ b$ whenever $b\ R\ a$, for all $a, b \in A$, *asymmetric* if $a\ R\ b$ and $b\ R\ a$ for no $a, b \in A$, and *antisymmetric* if $a = b$ whenever $a\ R\ b$ and $b\ R\ a$, for all $a, b \in A$.

There is precisely one relation that is both reflexive and irreflexive: the (necessarily) empty relation on the empty set. The following lemma, whose proof is left to the reader (Exercise A.7), provides more concise definitions of the properties defined above.

Lemma A.1.5. *A relation R on a set A is*

1. *reflexive if and only if* $\mathrm{id}_A \subseteq R$,
2. *irreflexive if and only if* $\mathrm{id}_A \cap R = \varnothing$,
3. *transitive if and only if* $R \cdot R \subseteq R$,
4. *symmetric if and only if* $R = R^{-1}$,
5. *asymmetric if and only if* $R \cap R^{-1} = \varnothing$,
6. *antisymmetric if and only if* $R \cap R^{-1} \subseteq \mathrm{id}_A$.

Definition A.1.6. Let R be a relation on a set A. We inductively define relations R^n on A for $n \in \mathbb{N}$ as follows: $R^0 = \mathrm{id}_A$ and $R^{n+1} = R^n \cdot R$ for all $n \in \mathbb{N}$. The relations

$$\bigcup_{n \geqslant 0} R^n \qquad \text{and} \qquad \bigcup_{n \geqslant 1} R^n$$

will be denoted by R^* and R^+, and we denote $R \cup \mathrm{id}_A$ by $R^=$.

It is easy to see that $R^* = R^+ \cup \mathrm{id}_A$ for any relation on A.

Example A.1.7. Consider the *successor* relation $>_1$ on $\mathbb{N}$ defined as follows: $x >_1 y$ if and only if $x = y + 1$. A straightforward induction argument shows $x >_1^n y$ if and only if $x = y + n$, for all $n \in \mathbb{N}$. Hence $>_1^+ = >$ and $>_1^* = \geqslant$, where $>$ denotes the standard order on $\mathbb{N}$.

Lemma A.1.8. *A relation R on a set A is*

1. *transitive if and only if* $R = R^+$,
2. *transitive and reflexive if and only if* $R = R^*$,
3. *transitive, reflexive, and symmetric if and only if* $R^{-1} = R^*$.

Proof

1. According to Lemma A.1.5 it suffices to show $R^2 \subseteq R$ if and only if $R = R^+$. The if direction is obvious as $R^2 \subseteq R^+$ by definition. For the only-if direction we first show $R^n \subseteq R$ for all $n \geqslant 1$ by induction on n. If $n = 1$ then $R^n = R \subseteq R$. If $n > 1$ then $R^n = R^{n-1} \cdot R$. The induction hypothesis yields $R^{n-1} \subseteq R$ and thus $R^n \subseteq R \cdot R = R^2 \subseteq R$. It follows that $R^+ \subseteq R$. We obviously have $R = R^1 \subseteq R^+$ and hence $R = R^+$.

2 First suppose R is transitive and reflexive. So $R = R^+$ by part **1** and $\mathrm{id}_A \subseteq R$ by Lemma A.1.5. Hence $R \subseteq R^* = R^+ \cup \mathrm{id}_A = R \cup \mathrm{id}_A \subseteq R$ and thus $R = R^*$. Conversely, if $R = R^*$ then $\mathrm{id}_A \subseteq R$ and $R^+ \subseteq R$. From the former we obtain the reflexivity of R by Lemma A.1.5. Since $R = \mathrm{id}_A \cdot R = R^0 \cdot R = R^1 \subseteq R^+$, from the latter and part **1** we obtain the transitivity of R.

3 First suppose R is transitive, reflexive, and symmetric. So $R = R^*$ by part **2** and $R = R^{-1}$ by Lemma A.1.5. Hence $R^{-1} = R^*$. Conversely, suppose $R^{-1} = R^*$. If we show $R = R^*$ then R is transitive and reflexive by part **2** and symmetric by Lemma A.1.5. We have $R = (R^{-1})^{-1} = (R^*)^{-1} = (R^{-1})^* = (R^*)^* = R^*$ where the first equality is Exercise A.3(a) and the third and fifth equality follow by straightforward induction proofs (Exercises A.8(d) and A.9(c)). □

Theorem A.1.9. *For every relation R on a set A,*

1 *$R^=$ is the smallest reflexive relation that contains R,*

2 *R^+ is the smallest transitive relation that contains R,*

3 *$R \cup R^{-1}$ is the smallest symmetric relation that contains R.*

Proof

1 Since $\mathrm{id}_A \subseteq R^=$, $R^=$ is reflexive. Clearly $R \subseteq R^=$. Let S be any reflexive relation on A that extends R. We have to show $R^= \subseteq S$ in order to conclude that $R^=$ is the *smallest* reflexive relation on A that contains R. Reflexivity of S yields $\mathrm{id}_A \subseteq S$. Combined with $R \subseteq S$, we obtain $R^= = R \cup \mathrm{id}_A \subseteq S$.

2 Lemma A.1.8 **1** shows that R^+ is transitive. Clearly $R \subseteq R^+$. Let S be any transitive relation on A that extends R. We show by induction on n that $R^n \subseteq S$ for every $n \geqslant 1$. The case $n = 1$ is trivial. Suppose $R^n \subseteq S$ and consider R^{n+1}. We have $R^{n+1} = R^n \cdot R \subseteq S \cdot S \subseteq S$. The first inclusion follows from Exercise A.4 and the second is a consequence of the transitivity of S. We conclude $R^+ \subseteq S$. Hence R^+ is the smallest transitive relation on A that extends R.

3 Clearly $R \cup R^{-1}$ extends R. We have

$$(R \cup R^{-1})^{-1} = R^{-1} \cup (R^{-1})^{-1} = R^{-1} \cup R = R \cup R^{-1}$$

Here the first two equalities follow from Exercise A.4(b) and Exercise A.3(a). So $R \cup R^{-1}$ is a symmetric relation. Let S be any symmetric relation on A that extends R. We infer $R^{-1} \subseteq S^{-1} = S$ from Exercise A.4(a). Hence $R \cup R^{-1} \subseteq S$. This concludes the proof. □

Theorem A.1.10. *For every relation R on a set A,*

1 *R^* is the smallest transitive and reflexive relation extending R,*

2 *$(R \cup R^{-1})^*$ is the smallest transitive, reflexive, and symmetric relation extending R.*

Proof

1 According to parts **1** and **2** of Theorem A.1.9, any transitive and reflexive extension of R must contain $R^=$ and R^+, and hence also $R^= \cup R^+ = R^*$. It remains to show that R^* is transitive and reflexive. This follows from Lemma A.1.8 **2** in connection with Exercise A.9(c).

2 According to Theorem A.1.9 **3** and part **1**, any transitive, reflexive, and symmetric extension of R must contain R^* and $R \cup R^{-1}$. Since $R^* \cup R \cup R^{-1} \subseteq (R \cup R^{-1})^*$, it remains to show that any transitive, reflexive, and symmetric extension S of R contains $(R \cup R^{-1})^*$. By induction on $n \geqslant 0$ we show $(R \cup R^{-1})^n \subseteq S$. The case $n = 0$ is trivial. Consider $(R \cup R^{-1})^{n+1}$. We have

$$(R \cup R^{-1})^{n+1} = (R \cup R^{-1})^n \cdot (R \cup R^{-1}) \subseteq S \cdot (S \cup S^{-1}) = S \cdot S \subseteq S$$

The first inclusion follows from the induction hypothesis in connection with Exercise A.4, the second equality and second inclusion follow from Lemma A.1.5. □

Definition A.1.11. Let $\mathcal{P}$ be a property of relations. The *$\mathcal{P}$-closure* of a relation R is the smallest relation that contains R and satisfies the property $\mathcal{P}$.

Using the closure terminology, the preceding two theorems read as follows: The reflexive closure of a relation R on a set A is $R^=$, its transitive closure is R^+, its symmetric closure is $R \cup R^{-1}$, its transitive and reflexive closure is R^*, and its transitive, reflexive, and symmetric closure is $(R \cup R^{-1})^*$. Observe that there are properties $\mathcal{P}$ for which the notion of $\mathcal{P}$-closure does not make sense. For instance, the irreflexive closure of a nonempty reflexive relation R does not exist because every extension of R contains a pair (a, a). Exercise A.10 provides an alternative formulation of Definition A.1.11 in terms of *closure operators*.

Definition A.1.12. An *equivalence* relation is a reflexive, transitive, and symmetric relation. Let R be an equivalence relation on a set A. We call two elements $a, b \in A$ *equivalent* whenever $a \; R \; b$. The *equivalence class* $[a]_R$ of an element $a \in A$ consists of all elements $b \in A$ that are equivalent to a. The element a is called a *representative* of the equivalence class $[a]_R$. The set of all equivalence classes, denoted by A/R, is called the *quotient set* of A *modulo* R.

Equivalence relations are usually denoted by symbols like $\sim$ and $\approx$.

Definition A.1.13. Consider a set A equipped with an equivalence relation $\sim$ and an arbitrary relation R on A. The *quotient relation* $R/\!\sim$ on $A/\!\sim$ is defined as follows: $[a]_\sim \; R/\!\sim \; [b]_\sim$ if and only if $a \sim \cdot \, R \, \cdot \sim b$.

The above definition does not depend on the representatives a and b, because if $[a]_\sim = [a']_\sim$ and $[b]_\sim = [b']_\sim$ then $a \sim \cdot \, R \, \cdot \sim b$ if and only if $a' \sim \cdot \, R \, \cdot \sim b'$.

Example A.1.14. Consider the equivalence relation $\sim$ on $\mathbb{N}$ defined by $x \sim y$ if and only if x and y are congruent modulo 7 (i.e., x and y have the same rest after division by 7). There are seven equivalence classes: $C_i = \{x \mid x \bmod 7 = i\}$ for $i = 0, \ldots, 6$. The relation $>/\!\sim$ contains all pairs (C_i, C_j) for $0 \leqslant i, j \leqslant 6$. For instance, $C_2 >/\!\sim C_5$ because $2 \sim 9 > 5 \sim 5$.

We conclude this section with defining different kinds of *orders*.

Definition A.1.15. A *partial order* is a reflexive, transitive, and antisymmetric relation. A *proper order* is an irreflexive and transitive relation.

Proper orders are also known as *strict orders*.

Example A.1.16. The subset relation $\subseteq$ is a partial order on $\mathcal{P}(\mathbb{N})$, the set of all subsets of natural numbers. Its strict part $\subsetneq$ is a proper order on $\mathcal{P}(\mathbb{N})$.

Partial orders are usually denoted by symbols like $\geqslant$ and $\sqsupseteq$. We use $>$ and $\sqsupset$ to denote proper orders.

Definition A.1.17. With every proper order $>$ on a set A we associate a partial order $\geqslant$ on A defined as follows: $a \geqslant b$ if and only if $a > b$ or $a = b$. So $\geqslant$ is simply the reflexive closure of $>$. Conversely, if $\geqslant$ is a partial order on A then its *strict part* is the proper order $>$ on A defined as follows: $a > b$ if and only if $a \geqslant b$ and $a \neq b$.

Definition A.1.18. A *total* order is a proper order $>$ on a set A with the additional property that for any two distinct elements $a, b \in A$, either $a > b$ or $b > a$.

Theorem A.1.19. *Every proper order can be extended to a total order on the same set.*

Definition A.1.20. A *preorder* is a reflexive and transitive relation.

For preorders we normally employ the symbol $\gtrsim$. Observe that every partial order is a preorder. The converse does not hold. The easiest example of a preorder that is not a partial order is the relation $R = \{a, b\} \times \{a, b\}$.

Lemma A.1.21. *If $\gtrsim$ is a preorder on a set A then $> = \gtrsim \setminus \lesssim$ is a proper order on A and $\sim = \gtrsim \cap \lesssim$ is an equivalence relation on A.*

Proof First note that $\lesssim$ is a preorder. Hence $>$ is transitive and $\sim$ is reflexive and transitive. Since both $\gtrsim$ and $\lesssim$ are reflexive, the difference $\gtrsim \setminus \lesssim$ is irreflexive. It remains to show that $\sim$ is symmetric. Because $\sim^{-1} = \gtrsim^{-1} \cap \lesssim^{-1} = \lesssim \cap \gtrsim = \sim$, this follows from Lemma A.1.5. □

Definition A.1.22. Let A_1 and A_2 be sets equipped with proper orders $>_1$ and $>_2$. We define a relation $(>_1, >_2)_{\mathsf{lex}}$ on $A_1 \times A_2$ as follows: $(a_1, a_2)\ (>_1, >_2)_{\mathsf{lex}}\ (b_1, b_2)$ if and only if $a_1 >_1 b_1$ or both $a_1 = b_1$ and $a_2 >_2 b_2$. The relation $(>_1, >_2)_{\mathsf{lex}}$ is called the *lexicographic product* of $>_1$ and $>_2$.

Example A.1.23. On $\mathbb{N} \times \mathbb{N}$ we have $(3, 2)\ (>, >)_{\mathsf{lex}}\ (2, 5)$ and $(3, 2)\ (>, >)_{\mathsf{lex}}\ (3, 1)$ but not $(3, 2)\ (>, >)_{\mathsf{lex}}\ (4, 0)$.

The proof of the final result of this section is left to the reader (Exercise A.13).

Theorem A.1.24.

1 *The lexicographic product of proper orders is a proper order.*

2 *The lexicographic product of total orders is total.*

The above results easily generalize to lexicographic products of more than two components.

Exercises

A.1 Compute $\mathcal{D}\mathsf{om}(>_1^n)$ and $\mathsf{ran}(>_1^n)$ for every $n \in \mathbb{N}$. Here $>_1$ denotes the successor relation on $\mathbb{N}$.

A.2 Let R be a relation on a set A.

a Show $R \cdot \mathrm{id}_A = \mathrm{id}_A \cdot R = R$.

b Does the equality $R \cdot R^{-1} = \mathrm{id}_A$ hold?

A.3 ***a*** Show $(R^{-1})^{-1} = R$ for all relations R.

b Show that composition of relations is associative.

A.4 Let R_1 and R_2 be relations on $A \times B$ and let S_1 and S_2 be relations on $B \times C$.

a Show $R_1 \subseteq R_2$ if and only if $R_1^{-1} \subseteq R_2^{-1}$.

b Show $(R_1 \cup R_2)^{-1} = R_1^{-1} \cup R_2^{-1}$.

c Show $R_1 \cdot S_1 \subseteq R_2 \cdot S_2$ whenever $R_1 \subseteq R_2$ and $S_1 \subseteq S_2$.

d Does the converse of part (c) hold?

A.5 Let R be a relation on $A \times B$ and S and T be relations on $B \times C$.

a Show $R \cdot (S \cup T) = (R \cdot S) \cup (R \cdot T)$.

b Show $R \cdot (S \cap T) \subseteq (R \cdot S) \cap (R \cdot T)$.

c Does $R \cdot (S \cap T) \supseteq (R \cdot S) \cap (R \cdot T)$ hold?

A.6 Complete the following table:

	$>$ on $\mathbb{N}$	$\geqslant$ on $\mathbb{Z}$	$\varnothing$ on $\mathbb{N}$	$\mathrm{id}_{\mathbb{N}}$	$\varnothing$ on $\varnothing$
reflexive			$\times$		
irreflexive					
transitive	✓				
symmetric					
asymmetric					
antisymmetric					

A.7 Prove Lemma A.1.5.

A.8 Let R be a relation on a set A.

a Show $R^m \cdot R^n = R^{m+n}$ and $(R^m)^n = R^{m \cdot n}$ for all $m, n \in \mathbb{N}$.

b Show $(R^n)^{-1} = (R^{-1})^n$ for all $n \in \mathbb{N}$.

c Does the equality $R^{m+n} \cdot (R^{-1})^n = R^m$ hold for all $m, n \in \mathbb{N}$?

d Use part (b) to show $(R^*)^{-1} = (R^{-1})^*$ and $(R^+)^{-1} = (R^{-1})^+$.

e Prove the following inclusions:

- ▷ $R^* \cdot R \subseteq R^+$
- ▷ $R^* \cdot R^* \subseteq R^*$
- ▷ $R^* \cdot R^+ \subseteq R^+$

▷ $R^+ \cdot R^+ \subseteq R^+$

f Do the reverse inclusions hold?

A.9 Let R and S be relations on a set A.

a Show $R^+ = R^*$ whenever R is reflexive.

b Does the converse of part (a) hold?

c Show $(R^*)^* = (R^+)^* = (R^*)^+ = R^*$.

d Show $(R \cup S)^* = R^* \cdot (S \cdot R^*)^*$.

e Show that the following statements are equivalent:

▷ $S \cdot R^* \subseteq R^* \cdot S^*$

▷ $S^* \cdot R^* \subseteq R^* \cdot S^*$

▷ $(R \cup S)^* \subseteq R^* \cdot S^*$

A.10 Let U be the set of all relations on a set A. Let ϕ be a mapping from U to U. A relation $R \in U$ has the property $\mathcal{P}_\phi$ if $R = \phi(S)$ for some $S \in U$. We call ϕ a *closure operator* if

1. $R \subseteq \phi(R)$,
2. $\phi(\phi(R)) = \phi(R)$ (*idempotence*), and
3. $\phi(R) \subseteq \phi(S)$ whenever $R \subseteq S$ (*monotonicity*),

for all $R, S \in U$.

a Show that $(\cdot)^+$ is a closure operator.

b Show the equivalence of the following two statements:

▷ ϕ is a closure operator,

▷ $\phi(R)$ is the $\mathcal{P}_\phi$-closure of R, for every $R \in U$.

c Let $\mathcal{P}$ be a property such that the $\mathcal{P}$-closure of every $R \in U$ exists. Show that the mapping ϕ that assigns to every $R \in U$ its $\mathcal{P}$-closure is a closure operator with $\mathcal{P}_\phi = \mathcal{P}$.

A.11 Let R be a relation on a finite set A.

a Show $R^+ = R \cup R^2 \cup \cdots \cup R^n$ for some $n \in \mathbb{N}$.

b Give an upper bound of the smallest such n in terms of the size of A.

A.12 Let $\sim$ be an equivalence relation on a set A. Show that the following statements are equivalent, for all $a, b \in A$:

▷ $a \sim b$

▷ $[a]_\sim \cap [b]_\sim \neq \varnothing$

▷ $[a]_\sim = [b]_\sim$

A.13 Prove Theorem A.1.24.

A.14 Show that lexicographic product is monotone, i.e., if $>_i$ and $\sqsupset_i$ are proper orders on A_i such that $>_i \subseteq \sqsupset_i$ for $i = 1, 2$ then $(>_1, >_2)_{\mathsf{lex}} \subseteq (\sqsupset_1, \sqsupset_2)_{\mathsf{lex}}$.

A.2 Well-founded Induction

Induction is one of the most common proof techniques used in this book. In this section we put it on a solid footing.

Definition A.2.1. Let R be a relation on a set A. We say that R *admits induction* if a property $\mathcal{P}$ of elements of A holds for all elements of A whenever the following condition is satisfied:

An element $a \in A$ has the property $\mathcal{P}$ if all elements $b \in A$ with $a \; R \; b$ have the property $\mathcal{P}$.

Specializing this definition to the successor relation $>_1$ on $\mathbb{N}$, the induction principle reads as follows: A property $\mathcal{P}$ holds for all natural numbers if, for all $a \in \mathbb{N}$, $\mathcal{P}(a)$ is a consequence of $\mathcal{P}(b)$ for all b with $a >_1 b$. Since for $a = 0$ there is no b with $a >_1 b$, and for every $a > 0$ there is exactly one b with $a >_1 b$, viz. $a - 1$, this is equivalent to the usual formulation of *mathematical induction over the natural numbers*: if

[1] $\mathcal{P}(0)$, and

[2] for all $n > 0$, $\mathcal{P}(n-1)$ implies $\mathcal{P}(n)$,

then $\mathcal{P}$ holds for all natural numbers. Since mathematical induction over the natural numbers is a valid principle, the relation $>_1$ admits induction. Specializing Definition A.2.1 to the transitive closure $>$ of $>_1$, the corresponding induction principle is easily seen to be equivalent to the following statement: If $\mathcal{P}(0)$ and, for all $n > 0$, $\mathcal{P}(0), \ldots, \mathcal{P}(n-1)$ imply $\mathcal{P}(n)$, then $\mathcal{P}$ holds for all natural numbers. This is also a valid statement, hence $>$ admits induction. Sometimes induction with respect to $>_1$ is called *weak* and induction with respect to $>$ *strong* induction.

Definition A.2.2. Let R be a relation on a set A and let B be a subset of A. An element $a \in B$ is called a *minimal* element of B if there is no $b \in B$ such that $a \; R \; b$.

The following result characterizes the relations that admit induction.

Theorem A.2.3. *Let R be a relation on a set A. The following statements are equivalent.*

[1] *The relation R admits induction.*

[2] *There are no infinite descending sequences $a_1 \; R \; a_2 \; R \; a_3 \; R \cdots$ of elements of A.*

[3] *Every non-empty subset B of A contains a minimal element.*

Proof

[1] $\Longrightarrow$ [2] Define a property $\mathcal{P}$ as follows: $\mathcal{P}(a)$ holds if and only if there are no infinite descending sequences starting from element a. We have to show that $\mathcal{P}(a)$ holds for all elements a. Because R admits induction, it is sufficient to show $\mathcal{P}(a)$ whenever $\mathcal{P}(b)$ for all $b \in A$ with $a \; R \; b$. So take an arbitrary element $a \in A$ such that $\mathcal{P}(b)$ for all $b \in A$ with $a \; R \; b$. By assumption, there are no infinite descending sequences starting from any $b \in A$ with $a \; R \; b$. This implies that there are no infinite descending sequences starting from a, since the second element in such a sequence is an element $b \in A$ with $a \; R \; b$.

[2] $\Longrightarrow$ [3] Suppose there is a non-empty subset B of A without a minimal element. So for every $a \in B$ there is a $b \in B$ such that $a \; R \; b$. This implies that there is an infinite descending sequence of elements of B. (Here we use the Axiom of Choice, see Exercise A.16.)

[3] $\Longrightarrow$ [1] Suppose R does not admit induction. This means that there is a property $\mathcal{P}$ such that $\mathcal{P}(a)$ whenever $\mathcal{P}(b)$ for all $b \in A$ with $a \; R \; b$, but $\mathcal{P}$ does not hold for all elements of A. So the set $B = \{a \in A \mid \mathcal{P}(a) \text{ does not hold}\}$ is non-empty. Hence it contains a minimal element, say a. Because $\mathcal{P}(a)$ does not hold, there must be an

element $b \in A$ with $a \; R \; b$ such that $\mathcal{P}(b)$ does not hold. Because $b \in B$, this contradicts the minimality of a. □

Definition A.2.4. A relation is called *well-founded* if it satisfies one of the equivalent assertions of Theorem A.2.3.

It is easy to see that every well-founded relation is irreflexive.

Definition A.2.5. A well-founded proper order will simply be called *well-founded order*. A partial order is well-founded if its strict part is well-founded. A *well-order* is a total well-founded order.

Example A.2.6. The standard order $>$ on $\mathbb{N}$ is a well-order. The subset relation $\subseteq$ on $\mathcal{P}(\mathbb{N})$ is not well-founded: $\varnothing \subsetneq \{0\} \subsetneq \{0,1\} \subsetneq \cdots$. The proper superset relation $\supsetneq$ on the set of finite subsets of $\mathbb{N}$is well-founded but not a well-order as neither $\{1\} \supsetneq \{2\}$ nor $\{2\} \supsetneq \{1\}$ holds.

Lemma A.2.7. *Every proper order on a finite set is well-founded.*

Proof Let $>$ be a proper order on a finite set A. For a proof by contradiction, suppose there exists an infinite descending sequence $a_1 > a_2 > a_3 > \cdots$ of elements of A. Because A is finite, we must have $a_i = a_j$ for some $1 \leqslant i < j$. Transitivity of $>$ yields $a_i > a_j$. This contradicts the irreflexivity of $>$. □

Theorem A.2.8. *Every well-founded order can be extended to a well-order on the same set.*

An immediate consequence of Theorem A.2.8 is the *principle of well-ordering*: every set can be well-ordered.

Theorem A.2.9. *The lexicographic product of well-founded orders is well-founded.*

Proof Let A_1 and A_2 be sets equipped with well-founded orders $>_1$ and $>_2$. We prove that the relation $> = (>_1, >_2)_{\mathsf{lex}}$ on $A_1 \times A_2$ is well-founded. Suppose to the contrary that there exists an infinite descending sequence $(a_1, b_1) > (a_2, b_2) > (a_3, b_3) > \cdots$ of elements of $A_1 \times A_2$. For all $i \geqslant 1$ we either have $a_i >_1 a_{i+1}$ or both $a_i = a_{i+1}$ and $b_i >_2 b_{i+1}$. Since $>_1$ is well-founded, the first alternative cannot occur infinitely often. Hence there exists an index $j \geqslant 1$ such that $b_i >_2 b_{i+1}$ for all $i \geqslant j$, contradicting the well-foundedness of $>_2$. □

The remainder of this section is devoted to ordinals. The presented material is used in Section 9.5.

Definition A.2.10. A *transitive* set is a set whose members are also subsets. *Ordinals* are transitive sets that are well-ordered by set membership. Hence $\alpha < \beta$ if and only if $\alpha \in \beta$. The smallest ordinal is the empty set, denoted by 0. If α is an ordinal then the ordinal $\alpha \cup \{\alpha\}$ is its successor, denoted by $\alpha + 1$. An ordinal β is a *successor ordinal* if there is some ordinal α such that $\beta = \alpha + 1$. In this case the ordinal α is the (unique)

maximal element of β. Non-zero ordinals without maximal element are *limit ordinals*.

So we distinguish three types of ordinals: $0 = \varnothing$, successor ordinals $\alpha + 1 = \alpha \cup \{\alpha\}$, and limit ordinals $\alpha = \sup\{\beta \mid \beta < \alpha\}$. In the latter case, the set $\{\beta \mid \beta < \alpha\}$ lacks a maximal element.

Example A.2.11. By identifying the ordinals

$$\varnothing \qquad \{\varnothing\} \qquad \{\varnothing, \{\varnothing\}\} \qquad \{\varnothing, \{\varnothing\}, \{\varnothing, \{\varnothing\}\}\} \qquad \ldots$$

with 0, 1, 2, 3, ..., the natural numbers are embedded in the ordinals. Apart from 0, these are successor ordinals. The smallest limit order is denoted by ω and can be viewed as the set of natural numbers. The next successor ordinal is $\omega + 1$ which denotes $\omega \cup \{\omega\}$.

The following ordinal arithmetic operations constitute extensions of the respective operations on natural numbers. They are defined by *transfinite recursion*, which operates on ordinals. Proving properties of these arithmetic operations typically requires *transfinite induction*.

Definition A.2.12. For ordinals α and β their *sum* $\alpha + \beta$ is defined as follows:

$$\alpha + \beta = \begin{cases} \alpha & \text{if } \beta = 0 \\ (\alpha + \gamma) + 1 & \text{if } \beta = \gamma + 1 \\ \sup\{\alpha + \xi \mid \xi < \beta\} & \text{if } \beta \text{ is a limit ordinal} \end{cases}$$

Here sup stands for *supremum*, which is the least upper bound of its argument. In the case of ordinals, sup can be replaced by union, without affecting the definition.

Example A.2.13. We have $1 + \omega = \sup\{1 + \xi \mid \xi < \omega\} = \omega \neq \omega + 1$, so ordinal addition is not commutative.

Lemma A.2.14. *Ordinal addition is*

1. *associative,*
2. *weakly monotone in its first argument,*
3. *strictly monotone in its second argument.*

Proof We show the third property. So let α, β and γ be ordinals with $\beta < \gamma$. We prove $\alpha + \beta < \alpha + \gamma$ by transfinite induction on γ.

- ▷ If $\gamma = 0$ then $\beta < \gamma$ does not exist.
- ▷ If $\gamma = \delta + 1$ then $\beta \leqslant \delta$. If $\beta = \delta$ then $\alpha + \beta = \alpha + \delta < (\alpha + \delta) + 1 = \alpha + \gamma$. If $\beta < \delta$ then $\alpha + \beta < \alpha + \delta$ follows from the induction hypothesis and hence also $\alpha + \beta < (\alpha + \delta) + 1 = \alpha + \gamma$.
- ▷ In the remaining case γ is a limit ordinal. Hence $\alpha + \gamma = \sup\{\alpha + \xi \mid \xi < \gamma\}$ and $\beta + 1 < \gamma$. Moreover, $\alpha + \beta < (\alpha + \beta) + 1 = \alpha + (\beta + 1) \in \{\alpha + \xi \mid \xi < \gamma\}$ and therefore $\alpha + \beta \in \sup\{\alpha + \xi \mid \xi < \gamma\} = \alpha + \gamma$. □

Example A.2.15. Ordinal addition is not strictly monotone in its first argument: $0 < 1$ and $0 + \omega \not< 1 + \omega$ as $0 + \omega = \omega = 1 + \omega$.

Next we define ordinal multiplication.

Definition A.2.16. For ordinals α and β their *product* $\alpha \cdot \beta$ is defined as follows:

$$\alpha \cdot \beta = \begin{cases} 0 & \text{if } \beta = 0 \\ (\alpha \cdot \gamma) + \alpha & \text{if } \beta = \gamma + 1 \\ \sup\{\alpha \cdot \xi \mid \xi < \beta\} & \text{if } \beta \text{ is a limit ordinal} \end{cases}$$

Example A.2.17. Since $2 \cdot \omega = \sup\{2 \cdot \xi \mid \xi < \omega\} = \omega \neq \omega \cdot 2$ multiplication is not commutative. As $(\omega + 1) \cdot 2 = (\omega + 1) + (\omega + 1) = \omega + \omega + 1 = \omega \cdot 2 + 1$ also not right-distributive.

We often write αa for $\alpha \cdot a$ when α is an ordinal and a a finite ordinal, i.e., $a < \omega$. Ordinal multiplication is left-distributive over ordinal addition.

Lemma A.2.18. *We have $\alpha \cdot (\beta + \gamma) = \alpha \cdot \beta + \alpha \cdot \gamma$ for all ordinals α, β, γ.*

Left-distributivity is needed to prove the associatity of ordinal multiplication.

Lemma A.2.19. *Ordinal multiplication is*

1. *associative,*
2. *weakly monotone in its first argument,*
3. *strictly monotone in its second argument, provided the first argument is non-zero.*

The next operation is ordinal exponentiation.

Definition A.2.20. For ordinals α and β we define α^β as follows:

$$\alpha^\beta = \begin{cases} 1 & \text{if } \beta = 0 \\ \alpha^\gamma \cdot \alpha & \text{if } \beta = \gamma + 1 \\ \sup\{\alpha^\xi \mid \xi < \beta\} & \text{if } \beta \text{ is a limit ordinal} \end{cases}$$

Example A.2.21. We have $1^\omega = 1$, $2^\omega = \omega$ and $2^{\omega+1} = 2^\omega \cdot 2 = \omega \cdot 2$. We also have $(\omega^5 + \omega) + (\omega^3 + 2) = \omega^5 + \omega^3 + 2$ and $(\omega^5 + \omega) \cdot (\omega^3 + 2) = \omega^8 + \omega^5 \cdot 2 + \omega$.

Lemma A.2.22. *Ordinal exponentiation is*

1. *weakly monotone in its base,*
2. *strictly monotone in its exponent for bases exceeding 1.*

Proof We show the first property. So let α, β and γ be ordinals with $\alpha < \beta$. We prove $\alpha^\gamma \leqslant \beta^\gamma$ by transfinite induction on γ.

- ▷ If $\gamma = 0$ then $\alpha^\gamma = 1 = \beta^\gamma$.
- ▷ If $\gamma = \delta + 1$ then $\alpha^\delta \leqslant \beta^\delta$ by the induction hypothesis and thus $\alpha^\gamma = \alpha^\delta \cdot \alpha \leqslant \beta^\delta \cdot \alpha$ by Lemma A.2.19 **2** and $\beta^\delta \cdot \alpha \leqslant \beta^\delta \cdot \beta = \beta^\gamma$ by Lemma A.2.19 **3**. Hence $\alpha^\gamma \leqslant \beta^\gamma$.
- ▷ In the remaining case γ is a limit ordinal. The induction hypothesis yields $\alpha^\xi \leqslant \beta^\xi$ for all $\xi < \gamma$. Hence $\alpha^\gamma = \sup\{\alpha^\xi \mid \xi < \gamma\} \leqslant \sup\{\beta^\xi \mid \xi < \gamma\} = \beta^\gamma$. □

Definition A.2.23. The limit ordinal ϵ_0 is defined as $\sup\{\alpha_n \mid n < \omega\}$ where $\alpha_0 = \omega$ and $\alpha_n = \omega^{\alpha_{n-1}}$ for $n > 0$.

It can be shown that ϵ_0 is the smallest ordinal α such that $\omega^\alpha = \alpha$.

Notation. We write $\mathbb{O}$ for the set of ordinals smaller than ϵ_0.

We have $\mathbb{O} = \epsilon_0$.

Theorem A.2.24. *Every ordinal $\alpha < \epsilon_0$ can be uniquely written as*

$$\alpha = \omega^{\alpha_1} \cdot k_1 + \cdots + \omega^{\alpha_n} \cdot k_n$$

such that $\alpha > \alpha_1 > \cdots > \alpha_n$ and $k_1, \ldots, k_n$ are positive natural numbers.

Recursively writing the exponents $\alpha_1, \ldots, \alpha_n$ in the same form, which is possible as $\alpha > \alpha_1, \ldots, \alpha_n$, results in the *Cantor normal form* of ordinals. (Theorem A.2.24 without the conclusion $\alpha > \alpha_1$ also holds for ordinals $\alpha \geqslant \epsilon_0$.) A convenient variation allows for $\alpha_1 \geqslant \cdots \geqslant \alpha_n$ to avoid the multiplication factors $k_1, \ldots, k_n$.

The final operation on ordinals that we consider here is a commutative version of addition.

Definition A.2.25. The *natural sum* $\alpha \oplus \beta$ of two ordinals $\alpha = \omega^{\alpha_1} + \cdots + \omega^{\alpha_n}$ and $\beta = \omega^{\alpha_{n+1}} + \cdots + \omega^{\alpha_{n+m}}$ in $\mathbb{O}$ is defined as the ordinal

$$\omega^{\alpha_{\pi(1)}} + \cdots + \omega^{\alpha_{\pi(n+m)}}$$

where π is permutation on $\{1, \ldots, n+m\}$ such that $\alpha_{\pi(1)} \geqslant \cdots \geqslant \alpha_{\pi(n+m)}$.

Example A.2.26. We have $2 \oplus \omega = \omega \oplus 2 = \omega + 2$. Furthermore, $(\omega^5 + \omega) \oplus (\omega^3 + 2) = \omega^5 + \omega^3 + \omega + 2$.

Lemma A.2.27. *Natural addition on ordinals is*

1 *commutative,*

2 *associative,*

3 *strictly monotone in both arguments.*

Exercises

A.15 Show that the transitive closure of a well-founded relation is well-founded.

A.16 Let R be a relation on a set A.

a Prove the equivalence of the following two statements:

- ▷ there exists an infinite descending sequence of elements of A,
- ▷ there exist a non-empty subset B of A and a mapping $\phi\colon B \to B$ such that $b\ R\ \phi(b)$ for all $b \in B$.

b Let B be a non-empty subset of A without minimal element. Show how the existence of an infinite descending sequence of elements of B is a consequence of the Axiom of Choice, which states that for any non-empty family $\{A_i\}_{i\in I}$ of non-empty sets there exists a mapping ϕ from I to $\bigcup\{A_i \mid i \in I\}$ such that $\phi(i) \in A_i$ for all $i \in I$.

A.17 Show that a proper order on a set A is a well-order if and only if every subset of A has a unique minimal element.

A.18 ***a*** Prove parts 1 and 2 of Lemma A.2.14.

b Prove Lemma A.2.18.

c Prove Lemma A.2.19.

A.19 ***a*** Prove that $\alpha^\omega = \omega$ for every ordinal $1 < \alpha \in \omega$.

b Prove that ordinal exponentiation is not strictly monotone in its base for positive exponents.

c Prove Lemma A.2.22 2.

d Exactly one of the following two statements holds for all ordinals α, β and γ:

$$\alpha^{\beta+\gamma} = \alpha^\beta \cdot \alpha^\gamma \qquad\qquad (\alpha \cdot \beta)^\gamma = \alpha^\gamma \cdot \beta^\gamma$$

Which one? Give a proof and counterexample.

A.20 Prove that ϵ_0 is the smallest ordinal α such that $\omega^\alpha = \alpha$.

A.21 ***a*** Prove Lemma A.2.27.

b The natural product $\alpha \otimes \beta$ of two ordinals $\alpha = \omega^{\alpha_1} + \cdots + \omega^{\alpha_n}$ and $\beta = \omega^{\beta_1} + \cdots + \omega^{\beta_m}$ in $\mathbb{O}$ is defined as

$$\bigoplus_{\substack{1\leqslant i\leqslant n\\ 1\leqslant j\leqslant m}} \omega^{\alpha_i \oplus \beta_j}$$

Prove that natural multiplication is associative and commutative.

A.3 Multiset Orders

In this final section we introduce multisets and study multiset extensions of proper orders.

Definition A.3.1. A *multiset* is a collection in which elements are allowed to occur more than once. In this book we are only concerned with finite multisets. Formally, a multiset M over a set A is a function from A to $\mathbb{N}$, the set of natural numbers, such that $M(a) = 0$ for all but finitely many elements $a \in A$. For any $a \in A$, $M(a)$ denotes how often a occurs in M—it is called the *multiplicity* of a in M. If $M(a) = 0$ then M does not contain occurrences of a. The set of all multisets over A is denoted by $\mathcal{M}(A)$.

Example A.3.2. An example of a multiset belonging to $\mathcal{M}(\mathbb{N})$ is $\{1,1,3,4,4,4\}$. Formally, $\{1,1,3,4,4,4\}$ is the function M from $\mathbb{N}$ to $\mathbb{N}$ defined by

$$M(a) = \begin{cases} 2 & \text{if } a = 1 \\ 1 & \text{if } a = 3 \\ 3 & \text{if } a = 4 \\ 0 & \text{otherwise} \end{cases}$$

Observe that we adopt the usual set building notation for describing multisets. In particular, the empty multiset will be denoted by $\varnothing$. Operations on sets are naturally extended to multisets.

Definition A.3.3. Given two multisets $M_1, M_2 \in \mathcal{M}(A)$, their sum $M_1 \uplus M_2$ is defined by $(M_1 \uplus M_2)(a) = M_1(a) + M_2(a)$ for all $a \in A$, their difference $M_1 - M_2$ by

$$(M_1 - M_2)(a) = \begin{cases} M_1(a) - M_2(a) & \text{if } M_1(a) \geqslant M_2(a) \\ 0 & \text{otherwise} \end{cases}$$

and M_1 is contained in M_2, denoted by $M_1 \subseteq M_2$, if $M_1(a) \leqslant M_2(a)$ for all $a \in A$.

Example A.3.4. We have for example $\{1,2,3\} \uplus \{2,2\} = \{1,2,2,2,3\}$ and $\{2,0\} - \{1,2,3\} = \{0\}$. The multiset $\{2,2\}$ is not contained in $\{1,2,3\}$. Note that $\{1\} \uplus \{1\} = \{1,1\} \neq \{1\}$, so the sum of two sets (i.e., multisets whose members have multiplicity one) is not necessarily a set.

Definition A.3.5. Let $>$ be a proper order on a set A. The *multiset extension* of $>$ is a binary relation $>_{\mathsf{mul}}$ on $\mathcal{M}(A)$ defined as follows: $M_1 >_{\mathsf{mul}} M_2$ if $M_2 = (M_1 - X) \uplus Y$ for multisets $X, Y \in \mathcal{M}(A)$ that satisfy

1 $\varnothing \neq X \subseteq M_1$,

2 for all $y \in Y$ there exists an $x \in X$ such that $x > y$.

Informally, $M_1 >_{\mathsf{mul}} M_2$ if M_2 can be obtained from M_1 by replacing at least one element (those in X) with a finite (possibly zero) number of elements (those in Y) each of which is smaller (with respect to $>$) than one of the elements that is to be removed.

Example A.3.6. For example, in $\mathcal{M}(\mathbb{N})$ we have $\{1,2,3\} >_{\mathsf{mul}} \{2,2,2,2,2\}$ by letting $X = \{1,3\}$ and $Y = \{2,2,2,2\}$ in the above definition. Here $>$ denotes the usual order on the natural numbers. Note that we can also take $X = \{1,2,3\}$ and $Y = \{2,2,2,2,2\}$.

Lemma A.3.7. *The multiset extension of a proper order is a proper order.*

Proof Let $>$ be a proper order on a set A. We have to show that $>_{\mathsf{mul}}$ is a transitive and irreflexive relation on $\mathcal{M}(A)$.

Suppose $M_1 >_{\mathsf{mul}} M_2$ and $M_2 >_{\mathsf{mul}} M_3$. By definition there exist multisets $X_1, Y_1, X_2, Y_2 \in \mathcal{M}(A)$ such that $M_2 = (M_1 - X_1) \uplus Y_1$, $\varnothing \neq X_1 \subseteq M_1$, for all $y \in Y_1$ there exists an $x \in X_1$ such that $x > y$, $M_3 = (M_2 - X_2) \uplus Y_2$, $\varnothing \neq X_2 \subseteq M_2$, and for all $y \in Y_2$ there exists an $x \in X_2$ such that $x > y$. Define $X = X_1 \uplus (X_2 - Y_1)$ and $Y = (Y_1 - X_2) \uplus Y_2$. One easily verifies that X is a non-empty submultiset of M_1. The following sequence of equalities proves $(M_1 - X) \uplus Y = M_3$:

$$\begin{aligned}
(M_1 - X) \uplus Y &= (M_1 - (X_1 \uplus (X_2 - Y_1))) \uplus (Y_1 - X_2) \uplus Y_2 \\
&= ((M_1 - X_1) - (X_2 - Y_1)) \uplus (Y_1 - X_2) \uplus Y_2 \\
&= ((M_2 - Y_1) - (X_2 - Y_1)) \uplus (Y_1 - X_2) \uplus Y_2 \\
&= (((M_2 - Y_1) \uplus (Y_1 - X_2)) - (X_2 - Y_1)) \uplus Y_2 \\
&= (((M_2 - X_2) \uplus (X_2 - Y_1)) - (X_2 - Y_1)) \uplus Y_2 \\
&= (M_2 - X_2) \uplus Y_2 \\
&= M_3
\end{aligned}$$

The fourth equality follows because $X_2 - Y_1 \subseteq M_2 - Y_1$. The fifth equality is a consequence of the inclusion $Y_1 \subseteq M_2$. Before we can conclude $M_1 >_{\mathsf{mul}} M_3$ we have to show that for all $y \in Y$ there exists an $x \in X$ such that $x > y$. We distinguish two cases: $y \in Y_1 - X_2$ and $y \in Y_2$. If $y \in Y_1 - X_2$ then $x > y$ for some $x \in X_1 \subseteq X$. If $y \in Y_2$ then $x' > y$ for some $x' \in X_2$. If $x' \notin Y_1$ then $x' \in X_2 - Y_1 \subseteq X$ and hence we can take $x = x'$. Otherwise $x > x'$ for some $x \in X_1 \subseteq X$. Transitivity of $>$ yields $x > y$. This concludes the proof of transitivity of $>_{\mathsf{mul}}$.

If $>_{\mathsf{mul}}$ is not irreflexive then $M >_{\mathsf{mul}} M$ for some multiset M. By definition there exist multisets $X, Y \in \mathcal{M}(A)$ such that $M = (M - X) \uplus Y$, $\varnothing \neq X \subseteq M$, and for all $y \in Y$ there exists an $x \in X$ such that $x > y$. Clearly $X = Y$. Because X is non-empty, we infer the existence of an infinite descending sequence $x_1 > x_2 > x_3 > \cdots$ of elements of X. However, as X is finite we have $x_i = x_j$ for some $i < j$, contradicting the fact that $>$ is a proper order on A. □

Theorem A.3.8. *The multiset extension of a well-founded order is well-founded.*

Proof Let $>$ be a well-founded order on a set A. First we extend A with a new element $\bot$, so let $A' = A \cup \{\bot\}$. Now suppose $>_{\mathsf{mul}}$ is not well-founded. Then there exists an infinite descending sequence $M_0 >_{\mathsf{mul}} M_1 >_{\mathsf{mul}} M_2 >_{\mathsf{mul}} \cdots$ of multisets in $\mathcal{M}(A)$. From this sequence we will construct an infinite tree whose nodes (except for the root) are labeled with an element of A'. After each step in the construction the leaves of the tree form a multiset in $\mathcal{M}(A')$. More precisely, after the i-th construction step, the labels of the leaves of the tree form a multiset which after removal of the $\bot$s is identical to M_i.

We start the construction with a root node with children labeled with the elements of M_0. From $M_0 >_{\mathsf{mul}} M_1$ we infer the existence of multisets $X, Y \in \mathcal{M}(A)$ such that $M_1 = (M_0 - X) \uplus Y$, $\varnothing \neq X \subseteq M_0$, and for all $y \in Y$ there exists an $x \in X$ such that $x > y$. In the second step of the construction we add a child labeled y to the corresponding x, for every $y \in Y$. In addition, we add a child labeled $\bot$ for every $x \in X$. This process is repeated for $M_1 >_{\mathsf{mul}} M_2$, $M_2 >_{\mathsf{mul}} M_3$, and so forth. (Figure A.1 illustrates the tree construction for the sequence $\{3\} >_{\mathsf{mul}} \{1,2\} >_{\mathsf{mul}} \{2\} >_{\mathsf{mul}} \{1,1\}$ of multisets in $\mathcal{M}(\mathbb{N})$.)

Observe that during every construction step at least one node is added (since $X \neq \varnothing$). Hence the resulting tree is infinite. Moreover, the tree is easily seen to be finitely branching.

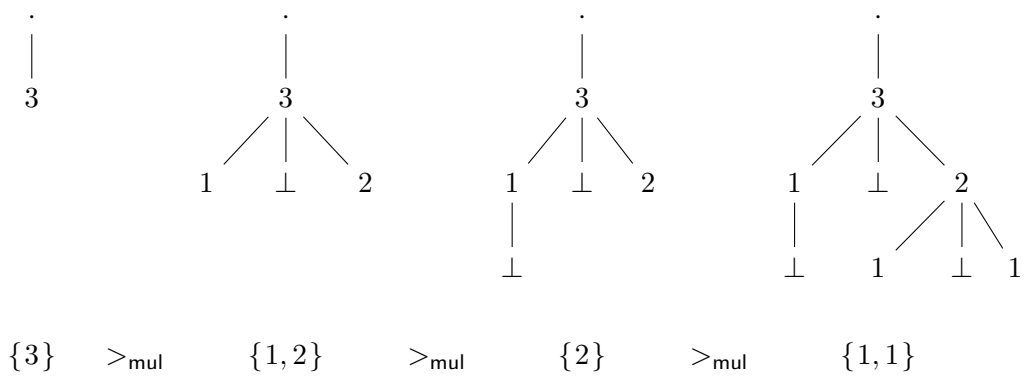


Figure A.1: Tree construction in the proof of Theorem A.3.8.

According to König's Lemma, an infinite tree that is finitely branching has an infinite path. By construction, every infinite path in the tree corresponds to a descending sequence (with respect to $>$) of elements of A. This contradicts the well-foundedness of $>$. □

Lemma A.3.9. *The multiset extension of a total order is total.*

Exercises

A.22 Order the multisets $\{2\}$, $\{1,3\}$, $\{1,1,1,2\}$, $\{2,2,2,2\}$, and $\{1,2\}$ with respect to the multiset extension of the usual order on natural numbers.

A.23 Give formal definitions of the empty multiset ($\varnothing$), multiset membership ($a \in M$), multiset equality ($M_1 = M_2$), and multiset intersection ($M_1 \cap M_2$).

A.24 Show that we can always choose $X \cap Y = \varnothing$ in Definition A.3.5.

A.25 Let $>$ be a proper order on a set A. Show that $>$ is total if and only if $>_{\mathsf{mul}}$ is a total order on $\mathcal{M}(A)$.

A.26 Let $>$ be a proper order on a set A. We define a relation $>_1$ on $\mathcal{M}(A)$ as follows: $M_1 >_1 M_2$ if M_2 can be obtained from M_1 by replacing a single element of M_1 by a finite number of smaller (with respect to $>$) elements.

a Is $>_1$ a proper order on $\mathcal{M}(A)$?

b Show that $>_{\mathsf{mul}}$ is the transitive closure of $>_1$.

A.27 Show that the multiset extension is monotone, i.e., if $>$ and $\sqsupset$ are proper orders on a set A such that $> \subseteq \sqsupset$ then $>_{\mathsf{mul}} \subseteq \sqsupset_{\mathsf{mul}}$.

A.28 Let $>$ be a proper order on a set A. Suppose $M_1, M_2 \in \mathcal{M}(A)$.

a Show the equivalence of the following two statements:

- ▷ $M_1 >_{\mathsf{mul}} M_2$
- ▷ $M_1 \neq M_2$ and for all $x \in A$ with $M_1(x) < M_2(x)$ there exists a $y \in A$ such that $y > x$ and $M_1(y) > M_2(y)$

The second statement can be rephrased as follows: $M_1 \neq M_2$ and for all $x \in M_2 - M_1$ there exists a $y \in M_1 - M_2$ such that $y > x$.

b Suppose $>$ is a total order on A. Show that the statements in part (a) are equivalent to the following statement:

▷ there exists an element $a \in A$ such that $M_1(a) > M_2(a)$ and $M_1(x) = M_2(x)$ for all $x > a$

A.29 Let $>$ be a proper order on a set A. Show that $M_1 >_{\mathsf{mul}} M_2$ if and only if $M_1 \uplus N >_{\mathsf{mul}} M_2 \uplus N$, for all $M_1, M_2, N \in \mathcal{M}(A)$.

A.30 Show that the converse of Theorem A.3.8 and Lemma A.3.9 also hold.

A.31 A *nested multiset* over a set A is either an element of A or a multiset of nested multisets over A. The set of all nested multisets over A is denoted by $\mathcal{M}^*(A)$. The *nested multiset extension* of a proper order $>$ on A is a binary relation $>^*_{\mathsf{mul}}$ on $\mathcal{M}^*(A)$ defined as follows: $M_1 >^*_{\mathsf{mul}} M_2$ if either

1. $M_1, M_2 \in A$ and $M_1 > M_2$, or
2. $M_1 \notin A$ and $M_2 \in A$, or
3. $M_1, M_2 \notin A$ and $M_2 = (M_1 - X) \uplus Y$ for certain $X, Y \in \mathcal{M}^*(A)$ that satisfy
   - ▷ $\varnothing \neq X \subseteq M_1$
   - ▷ for all $y \in Y$ there exists an $x \in X$ such that $x >^*_{\mathsf{mul}} y$

a Show that the nested multiset extension is a proper order.

b Show that the nested multiset extension of a total order is again total.

c Order $\{\{1,0,0\},5,\{\{0\},1,1,1\}\}$, $\{\{\varnothing,1,2\},\{5,2,5\},5\}$, and $\{\{1,1\},\{\{0\},1,2\},0\}$ with respect to the nested multiset extension of the usual order on $\mathbb{N}$.

d Show that $>^*_{\mathsf{mul}}$ is a well-founded order on $\mathcal{M}^*(A)$ if and only if $>$ is a well-founded order on A.

A.32 Let $>$ be a proper order on a set A. We define a relation $>_{\mathsf{m}}$ on $\mathcal{M}(A)$ as follows: $M_1 >_{\mathsf{m}} M_2$ if $M_1 \neq \varnothing$ and either

▷ $\hat{M}_1 >_{\mathsf{mul}} \hat{M}_2$, or

▷ $\hat{M}_1 = \hat{M}_2$ and $\check{M}_1 >_{\mathsf{m}} \check{M}_2$.

Here $\hat{M}$ denotes the submultiset of M consisting of all maximal elements with respect to $>$ and $\check{M}$ stands for $M - \hat{M}$, so if $M = \{1,3,2,3\}$ then $\hat{M} = \{3,3\}$ and $\check{M} = \{1,2\}$.

a Show $\hat{M} >_{\mathsf{mul}} \check{M}$ for any non-empty multiset M.

b Show that $>_{\mathsf{m}}$ is a proper order on $\mathcal{M}(A)$.

c Show that $>_{\mathsf{m}}$ properly contains the multiset extension $>_{\mathsf{mul}}$ of Definition A.3.5.

d Show that $>_{\mathsf{m}}$ coincides with $>_{\mathsf{mul}}$ whenever $>$ is a total order.

e Show that $>_{\mathsf{m}}$ inherits well-foundedness from $>$. (Hint: use well-founded induction with respect to $>_{\mathsf{mul}}$.)

f Is $>_{\mathsf{m}}$ monotone?

Bibliographic Notes

Theorem A.1.19 is from Szpilrajn [125]. Well-founded relations are also called Noetherian in the literature. Ordinals were invented by Cantor. The Cantor normal form originates from Cantor [17]. Natural addition and multiplication on ordinals were introduced by Hessenberg [50]. Multiset orders were introduced by Dershowitz and Manna [34]. Variants appear in Jouannaud and Lescanne [63], from which we extracted Exercises A.28 and A.32.

Appendix B

Kruskal's Tree Theorem

Throughout this appendix we deal with infinite sequences of some kind. We find it convenient to abbreviate an infinite sequence $(a_i)_{i\geqslant 0} = a_0, a_1, a_2, \ldots$ to $\mathbf{a}$. Moreover, we denote $(f(a_i))_{i\geqslant 0}$ by $f(\mathbf{a})$, $(a_{\psi(i)})_{i\geqslant 0}$ by $\mathbf{a}_\psi$, and $(a_i)_{i\geqslant n}$ by $\mathbf{a}_{\geqslant n}$.

B.1 Dickson's Lemma

In this section we introduce one of the essential ideas underlying the proof of the finite version of Kruskal's Tree Theorem by means of a simpler result, known as *Dickson's Lemma*. This lemma is of independent interest also as it plays an important role in the theory of Gröbner bases, cf. Chapter 13.

Definition B.1.1. An infinite sequence $\mathbf{n}$ of natural numbers is said to be *increasing* if $n_i \leqslant n_{i+1}$ for all $i \geqslant 0$.

Lemma B.1.2. *Every infinite sequence of natural numbers contains an increasing subsequence.*

Proof Let $\mathbf{n}$ be an infinite sequence of natural numbers. If some natural number occurs infinitely often in this sequence then we clearly have an increasing subsequence. So suppose every natural number occurs a finite number of times in the sequence $\mathbf{n}$. We construct an increasing subsequence as follows. Its first element will be n_0. Since there are only finitely many numbers in the sequence $\mathbf{n}$ less than n_0, there exists an index N such that all numbers in the subsequence $\mathbf{n}_{\geqslant N}$ are greater than or equal to n_0. The number n_N will be the second element of our subsequence. By repeating this process with the sequence $\mathbf{n}_{\geqslant N}$ we eventually arrive at an increasing subsequence of the original sequence $\mathbf{n}$. □

Dickson's Lemma. *If* $\mathbf{e}$ *is an infinite sequence of* n*-tuples of natural numbers then there exist indices* i*,* j *with* $i < j$ *such that* $e_i = (a_1, \ldots, a_n)$*,* $e_j = (b_1, \ldots, b_n)$*, and* $a_k \leqslant b_k$ *for every* $k \in \{1, \ldots, n\}$*.*

Proof Let us write $(a_1, \ldots, a_n) \leqslant (b_1, \ldots, b_n)$ if $a_k \leqslant b_k$ for every $k \in \{1, \ldots, n\}$. By induction on n we will show the stronger property of the existence of an infinite subsequence

$$e_{\varphi(0)} \leqslant e_{\varphi(1)} \leqslant e_{\varphi(2)} \leqslant \cdots$$

If $n = 0$ then all elements e_i equal the empty tuple (), in which case the property trivially holds. Suppose $\mathbf{e}$ is an infinite sequence of $n+1$-tuples. Write $e_i = (a_i^1, \ldots, a_i^{n+1})$ and define $e_i^* = (a_i^2, \ldots, a_i^{n+1})$. According to Lemma B.1.2 the infinite sequence $\mathbf{a^1}$ of first coordinates contains an increasing subsequence $\mathbf{a}^{\mathbf{1}}_{\psi}$. So the sequence $\mathbf{e}^*_{\psi}$ of n-tuples is infinite and hence we obtain an infinite subsequence

$$e^*_{\varphi(0)} \leqslant e^*_{\varphi(1)} \leqslant e^*_{\varphi(2)} \leqslant \cdots$$

from the induction hypothesis. By construction we have also

$$e_{\varphi(0)} \leqslant e_{\varphi(1)} \leqslant e_{\varphi(2)} \leqslant \cdots$$ □

Exercises

B.1 ***a*** Let $>$ be a well-order on a set A. Show that any infinite sequence of elements of A contains an increasing subsequence. (An infinite sequence $\mathbf{a}$ of elements of A is called increasing if $a_i \leqslant a_{i+1}$ for all $i \geqslant 0$.)

b Does part (a) hold for well-founded orders?

B.2 Let $e = (a_1, \ldots, a_n)$ and $e' = (b_1, \ldots, b_n)$ be n-tuples of natural numbers. We write $e >_n e'$ if $e \neq e'$ and $a_i \geqslant b_i$ for all $i \in \{1, \ldots, n\}$.

a Show that $>_n$ is a proper order on n-tuples of natural numbers.

b Is $>_n$ well-founded?

c Is $>_n$ total?

B.2 Kruskal's Tree Theorem — Finite Version

In this section we assume that $\mathcal{F}$ is a finite signature. The following definition introduces some concepts that play an important role in the proof of (the finite version of) Kruskal's Tree Theorem.

Definition B.2.1. Let $\mathbf{t}$ be an infinite sequence of terms. The sequence $\mathbf{t}$ is called *self-embedding* if $t_i \trianglelefteq_{\mathsf{emb}} t_j$ for some $i < j$. We say that $\mathbf{t}$ is *bad* if it is not self-embedding. The sequence $\mathbf{t}$ is called a *chain* if $t_i \trianglelefteq_{\mathsf{emb}} t_{i+1}$ for all $i \geqslant 0$. We say that $\mathbf{t}$ contains a chain if it has a subsequence that is a chain. An element t_i of $\mathbf{t}$ is called *terminal* if there is no $j > i$ such that $t_i \trianglelefteq_{\mathsf{emb}} t_j$. (So $\mathbf{t}$ is bad if and only if all its elements are terminal.) Let $\mathbf{s}$ be an infinite sequence of terms. We write $\mathbf{s} \subset \mathbf{t}$ if there exists an infinite subsequence $\mathbf{u}$ of $\mathbf{t}$ such that s_i is a proper subterm of u_i for all $i \geqslant 0$; see Figure B.1.

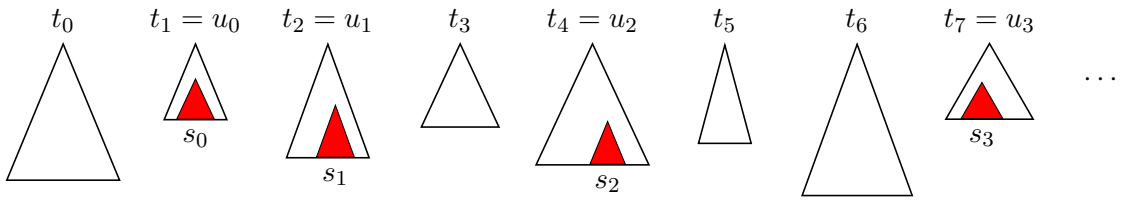


Figure B.1: $\mathbf{s} \subset \mathbf{t}$.

Kruskal's Tree Theorem (Finite Version). *Every infinite sequence of ground terms is self-embedding.*

Proof We have to show that there are no bad sequences of ground terms. Suppose to the contrary that there exists a bad sequence. We construct a special bad sequence $\mathbf{t}$ as follows:

- ▷ We choose t_0 to be a smallest (with respect to size) first element of any bad sequence.
- ▷ Suppose we already chose the first n elements $t_0, \ldots, t_{n-1}$. Consider all bad sequences that begin with these n elements. Define t_n to be a smallest n-th element in these sequences.

The sequence $\mathbf{t}$ has the following minimality property: if $\mathbf{s} \subset \mathbf{t}$ then $\mathbf{s}$ is self-embedding. This can be proved as follows. Suppose s_0 is a proper subterm of t_k and consider the infinite sequence $t_0, \ldots, t_{k-1}, \mathbf{s}$. Since the size of s_0 is less than the size of t_k, this sequence is self-embedding. The situation $t_i \trianglelefteq_{\mathsf{emb}} t_j$ $(0 \leqslant i < j \leqslant k-1)$ is impossible since $\mathbf{t}$ is bad. Likewise, $t_i \trianglelefteq_{\mathsf{emb}} s_j$ $(0 \leqslant i \leqslant k-1,\ j \geqslant 0)$ is impossible since s_j is a subterm of t_l for some $l > i$ and we know $t_i \trianglelefteq_{\mathsf{emb}} t_l$ does not hold. Hence we must have $s_i \trianglelefteq_{\mathsf{emb}} s_j$ for some $i < j$.

Even stronger, if $\mathbf{s} \subset \mathbf{t}$ then $\mathbf{s}$ contains a chain. First observe that $\mathbf{s}$ contains only finitely many terminal elements, otherwise $\mathbf{s}$ would contain a bad sequence of terminal elements, contradicting the minimality of $\mathbf{t}$. So there exists an index $N \geqslant 0$ such that for all s_i with $i \geqslant N$ there exists a $j > i$ with $s_i \trianglelefteq_{\mathsf{emb}} s_j$. From this we easily obtain a chain starting at s_N.

After these preliminaries, we will show that $\mathbf{t}$ contains a chain, which yields a contradiction with the fact that $\mathbf{t}$ is bad. Because the signature is finite, the infinite sequence $\mathsf{root}(\mathbf{t})$ of root symbols of $\mathbf{t}$ must contain a function symbol f that occurs infinitely many often. Consider the subsequence $\mathbf{t}^*$ of $\mathbf{t}$ containing all terms whose root symbol is f. Suppose f is n-ary. If $n = 0$ then $\mathbf{t}^*$ is trivially a chain. So assume $n \geqslant 1$ and write $t_i^* = f(t_i^1, \ldots, t_i^n)$ for $i \geqslant 1$. Let e_i be the n-tuple $(t_i^1, \ldots, t_i^n)$ of arguments of t_i^*, for all $i \geqslant 0$. An induction argument, similar to the one in the proof of Dickson's Lemma, yields an infinite subsequence

$$e_{\varphi(0)} \trianglelefteq_{\mathsf{emb}} e_{\varphi(1)} \trianglelefteq_{\mathsf{emb}} e_{\varphi(2)} \trianglelefteq_{\mathsf{emb}} \cdots$$

Here $(u_1, \ldots, u_n) \trianglelefteq_{\mathsf{emb}} (v_1, \ldots, v_n)$ stands for the conjunction of $u_1 \trianglelefteq_{\mathsf{emb}} v_1$, …, $u_n \trianglelefteq_{\mathsf{emb}} v_n$. From this we immediately obtain

$$t^*_{\varphi(0)} \trianglelefteq_{\mathsf{emb}} t^*_{\varphi(1)} \trianglelefteq_{\mathsf{emb}} t^*_{\varphi(2)} \trianglelefteq_{\mathsf{emb}} \cdots$$

which shows that $\mathbf{t}$ contains a chain. Hence $\mathbf{t}$ does not exist. We conclude that there are no bad sequences. □

Exercises

B.3 Is the relation $\subset$ defined in Definition B.2.1 a proper order on infinite sequences of terms?

B.4 Show that Dickson's Lemma is a special case of Kruskal's Tree Theorem.

B.5 Why is Kruskal's Tree Theorem restricted to *ground* terms?

B.3 Partial Well-Orders

Definition B.3.1. Let $>$ be a proper order on a set A and suppose $\mathbf{a}$ is an infinite sequence of elements of A. The sequence $\mathbf{a}$ is called *good* if there exist indices $0 \leqslant i < j$ with $a_i \leqslant a_j$, otherwise it is called *bad*. We say that $\mathbf{a}$ is a *chain* if $a_i \leqslant a_{i+1}$ for all $i \geqslant 0$. We say that $\mathbf{a}$ contains a chain if it has a subsequence that is a chain. The sequence $\mathbf{a}$ is called an *antichain* if neither $a_i \leqslant a_j$ nor $a_j \leqslant a_i$, for all $0 \leqslant i < j$.

Observe that the notions bad and chain reduce to the ones defined in Definition B.2.1 when A is $\mathcal{T}(\mathcal{F},\mathcal{V})$ and $>$ equals $\rhd_{\mathsf{emb}}$.

Theorem B.3.2. *Let $>$ be a proper order on a set A. The following statements are equivalent.*

1. *Every proper order that extends $>$ (including $>$ itself) is well-founded.*
2. *Every infinite sequence over A is good.*
3. *Every infinite sequence over A contains a chain.*
4. *The proper order $>$ is well-founded and does not admit antichains.*

Proof

1 $\Longrightarrow$ 2 Suppose $\mathbf{a}$ is a bad sequence. Let $\sqsupset_1 = \{(a_i, a_{i+1}) \mid i \geqslant 0\}$ and define $\sqsupset = (> \cup \sqsupset_1)^+$. Assume $a \sqsupset a$ for some $a \in A$. Since $a > a$ does not hold, we must have $a \geqslant \cdot\ (\sqsupset_1 \cdot \geqslant)^+\ a$ and thus $a \geqslant a_k\ (\sqsupset_1 \cdot \geqslant)^+\ a$ for some $k \geqslant 0$. Hence also $a_k\ (\sqsupset_1 \cdot \geqslant)^+ \cdot \geqslant a_k$ and thus $a_k\ (\sqsupset_1 \cdot \geqslant)^+\ a_k$ as

$$(\sqsupset_1 \cdot \geqslant)^+ \cdot \geqslant\ =\ (\sqsupset_1 \cdot \geqslant)^* \cdot \sqsupset_1 \cdot \geqslant \cdot \geqslant\ =\ (\sqsupset_1 \cdot \geqslant)^* \cdot \sqsupset_1 \cdot \geqslant\ =\ (\sqsupset_1 \cdot \geqslant)^+$$

An easy induction proof reveals $i < j$ whenever $a_i\ (\sqsupset_1 \cdot \geqslant)^+\ a_j$. Hence $a_k\ (\sqsupset_1 \cdot \geqslant)^+\ a_k$ implies the impossible $k < k$. It follows that $\sqsupset$ is irreflexive. By definition it is transitive, hence it is a proper order extending $>$. However, since $a_0 \sqsupset a_1 \sqsupset a_2 \sqsupset \cdots$, it is not well-founded.

2 $\Longrightarrow$ 3 Let $\mathbf{a}$ be any infinite sequence over A. Consider the subsequence consisting of all elements a_i with the property that $a_i \leqslant a_j$ holds for no $j > i$. If this subsequence is infinite then it is a bad sequence, contradicting 2. Hence it is finite, and thus there exists an index $N \geqslant 0$ such that for every $i \geqslant N$ there exists a $j > i$ with $a_i \leqslant a_j$. Define inductively

$$\phi(i)\ =\ \begin{cases} N & \text{if } i = 0 \\ \min\{j \mid j > \phi(i-1) \text{ and } a_{\phi(i-1)} \leqslant a_j\} & \text{if } i > 0 \end{cases}$$

Now $\mathbf{a}_\phi$ is a chain.

3 $\Longrightarrow$ 4 If $>$ is not well-founded then there exists an infinite sequence $a_0 > a_1 > \cdots$. Clearly $a_i \leqslant a_j$ does not hold for any $0 \leqslant i < j$. Hence this sequence does not contain a chain. If $>$ admits an antichain then this antichain is an infinite sequence not containing a chain.

4 $\Longrightarrow$ 1 For a proof by contradiction, let $>$ be a well-founded order that does not satisfy statement 1. So there is an extension $\sqsupset$ of $>$ that is not well-founded. Hence

there exists an infinite sequence $a_0 \sqsupset a_1 \sqsupset \cdots$. Since $>$ is well-founded, the sequence $\mathbf{a}$ contains an element a_i with the property that for no $j > i$, $a_i > a_j$ holds. Actually, $\mathbf{a}$ contains infinitely many such elements. We claim that the infinite subsequence $\mathbf{a}_\phi$ consisting of those elements is an antichain (with respect to $>$). Let $0 \leqslant i < j$. By construction $a_{\phi(i)} > a_{\phi(j)}$ is impossible. If $a_{\phi(i)} \leqslant a_{\phi(j)}$ then also $a_{\phi(i)} \sqsubseteq a_{\phi(j)}$, contradicting $a_{\phi(i)} \sqsupset a_{\phi(j)}$. Hence $>$ admits an antichain. □

Definition B.3.3. A proper order $>$ on a set A is called a *partial well-order* if it satisfies one of the four equivalent assertions of Theorem B.3.2.

By definition every partial well-order is a well-founded order, but the converse does not hold. For instance, the empty relation on an infinite set is a well-founded order but not a partial well-order. Clearly every well-order is a partial well-order.

Using the terminology of partial well-orders, the results of the preceding two sections read as follows: the standard order on natural numbers is a partial well-order (Lemma B.1.2), the relation $>_n$ defined in Exercise B.2 is a partial well-order on n-tuples of natural numbers (Dickson's Lemma), and $\rhd_{\mathsf{emb}}$ is a partial well-order on ground terms over a finite signature (Kruskal's Tree Theorem).

Exercises

B.6 Show that any proper order on a finite set is a partial well-order.

B.7 In Exercise B.2 we defined a proper order $>_n$ on n-tuples of natural numbers, for every $n \in \mathbb{N}$. Suppose we extend these proper orders to a proper order $>$ on arbitrary (finite) tuples of natural numbers by putting $e = (a_1, \ldots, a_m) > (b_1, \ldots, b_n) = e'$ if and only if either $m > n$ or both $m = n$ and $e >_m e'$. Is $>$ a partial well-order?

B.8 Let $>$ be a proper order on a set A. Show that $>$ is a partial well-order if and only if for every non-empty subset B of A, the subset of minimal elements of B is non-empty but finite.

B.9 Let $>$ be a partial well-order on a set A and let $\sqsupset$ be a partial well-order on a set B. Let $\varphi\colon A \to B$ be any function. Show that the relation $>'$ on A defined by $a >' b$ if and only if $a > b$ and $\varphi(a) \sqsupseteq \varphi(b)$ is a partial well-order.

B.10 Show that the intersection of two partial well-orders on a set A is a partial well-order on A. (Hint: use the previous exercise.)

B.11 Show that the product of two partial well-orders is a partial well-order.

B.12 Let $>$ be a partial well-order on a set A.

a Show that every proper order on A that contains $>$ is a partial well-order.

b Show that the multiset extension $>_{\mathsf{mul}}$ of $>$ is a partial well-order on $\mathcal{M}(A)$.

B.4 Kruskal's Tree Theorem — General Version

Before generalizing Kruskal's Tree Theorem to arbitrary signatures, we present a version for strings, known as *Higman's Lemma*.

We recall the following definition from Section 6.4.

Definition B.4.1. Let $>$ be a relation on a set A. The SRS $\mathcal{E}\mathsf{mb}(>)$ consists of all rules $a \to \epsilon$ for all $a \in A$ and $a \to b$ for all $a, b \in A$ with $a > b$. We write $>_{\mathsf{emb}}$ for the relation

$\to^+_{\mathcal{E}\mathsf{mb}(>)}$ on A^*.

Example B.4.2. Let $A = \{a, b, c, d\}$ with $a > d > c$. The SRS $\mathcal{E}\mathsf{mb}(>)$ consists of the following seven rules:

$$a \to d \qquad a \to c \qquad d \to c \qquad a \to \epsilon \qquad b \to \epsilon \qquad c \to \epsilon \qquad d \to \epsilon$$

We have $abcd >_{\mathsf{emb}} ccd$ since $\underline{a}bcd \to c\underline{b}cd \to ccd$ by applying the rules $a \to c$ and $b \to \epsilon$.

Lemma B.4.3. *If $>$ is a proper order on A then $>_{\mathsf{emb}}$ is a proper order on A^*.*

Proof Transitivity is immediate from the definition of $>_{\mathsf{emb}}$. We prove irreflexivity by contradiction. Consider a shortest string w such that $w >_{\mathsf{emb}} w$. Clearly $w \neq \epsilon$. Since rules in $\mathcal{E}\mathsf{mb}(>)$ do not increase the length of strings, collapsing rules $a \to \epsilon$ cannot be used in the derivation $w \to^+_{\mathcal{E}\mathsf{mb}(>)} w$. Write $w = aw'$. We obtain $a \to^*_{\mathcal{E}\mathsf{mb}(>)} a$ and $w' \to^*_{\mathcal{E}\mathsf{mb}(>)} w'$ by decomposition. Due to the minimality of w, the latter sequence must be empty and hence $a \to^+_{\mathcal{E}\mathsf{mb}(>)} a$. Therefore $a >^+ a$, contradicting the irreflexivity of $>$. □

Higman's Lemma. *If $>$ is a partial well-order on a set A then $>_{\mathsf{emb}}$ is a partial well-order on A^*.*

Proof We have to show that there are no bad sequences over A^*. Suppose to the contrary that there exist bad sequences over A^*. We construct a *minimal bad sequence* $\mathbf{w}$ as follows:

> Suppose we already chose the first n strings $w_0, \ldots, w_{n-1}$. Define w_n to be a shortest string such that there are bad sequences that start with $w_0, \ldots, w_n$.

Because $\epsilon \leqslant_{\mathsf{emb}} w$ for all $w \in A^*$, we have $w_i \neq \epsilon$ for all $i \geqslant 0$. Hence we may write $w_i = a_i v_i$ $(i \geqslant 0)$. Since $>$ is a partial well-order on A, the infinite sequence $\mathbf{a}$ contains a chain, say $\mathbf{a}_\phi$. Because $v_{\phi(0)}$ is shorter than $w_{\phi(0)}$, the sequence $w_0, \ldots, w_{\phi(0)-1}, \mathbf{v}_\phi$ must be good. Clearly $w_i \leqslant_{\mathsf{emb}} w_j$ $(0 \leqslant i < j \leqslant \phi(0) - 1)$ is impossible as $\mathbf{w}$ is bad. Likewise, $w_i \leqslant_{\mathsf{emb}} v_{\phi(j)}$ $(0 \leqslant i \leqslant \phi(0) - 1$ and $0 \leqslant j)$ contradicts the badness of $\mathbf{w}$ since $v_{\phi(j)} \leqslant_{\mathsf{emb}} w_{\phi(j)}$ and therefore $w_i \leqslant_{\mathsf{emb}} w_{\phi(j)}$. Hence we must have $v_{\phi(i)} \leqslant_{\mathsf{emb}} v_{\phi(j)}$ for some $0 \leqslant i < j$. Combining this with $a_{\phi(i)} \leqslant a_{\phi(j)}$ easily yields $w_{\phi(i)} = a_{\phi(i)}v_{\phi(i)} \leqslant_{\mathsf{emb}} a_{\phi(j)}v_{\phi(j)} = w_{\phi(j)}$, contradicting the badness of $\mathbf{w}$. We conclude that there are no bad sequences over A^*. □

The following special case is used in the proof of Theorem 6.4.11.

Higman's Lemma (Special Version). *If $>$ is a well-order on a set A then every proper order on A^* that contains $>_{\mathsf{emb}}$ is well-founded.*

Proof Since every well-order is a partial well-order, Higman's Lemma yields that $>_{\mathsf{emb}}$ is a partial well-order. By definition, every proper order extending $>_{\mathsf{emb}}$ is well-founded. □

After these preliminaries we prove the general version of Kruskal's Tree Theorem.

Kruskal's Tree Theorem (General Version). *If $>$ is a partial well-order on a signature $\mathcal{F}$ then $>_{\mathsf{emb}}$ is a partial well-order on $\mathcal{T}(\mathcal{F})$.*

Proof The proof has the same structure as the proof of Higman's Lemma. We have to show that there are no bad sequences of terms in $\mathcal{T}(\mathcal{F})$. Suppose to the contrary that there exist bad sequences of ground terms. We construct a *minimal bad sequence* $\mathbf{t}$ as follows:

> Suppose we already chose the first n terms $t_0, \ldots, t_{n-1}$. Define t_n to be a smallest (with respect to size) term such that there are bad sequences that start with $t_0, \ldots, t_n$.

For every $i \geqslant 0$, let f_i be the root symbol of t_i and let A_i be the set of arguments of t_i (if t_i is a constant then $A_i = \varnothing$). Moreover, let w_i be the string of arguments (from left to right) of t_i. Finally, let $A = \bigcup_{i \geqslant 0} A_i$.

We claim that $>_{\mathsf{emb}}$ is a partial well-order on the subset A of $\mathcal{T}(\mathcal{F})$. For a proof by contradiction, suppose $\mathbf{a}$ is a bad sequence over A. Let $a_0 \in A_k$. Since $A' = \bigcup_{i=0}^{k} A_i$ is a finite set, only finitely many elements of $\mathbf{a}$ belong to A'. Thus there exists an index $l > 0$ such that $a_i \in A \setminus A'$ for all $i \geqslant l$. Because a_0 is a proper subterm of t_k, the sequence $t_0, \ldots, t_{k-1}, a_0, \mathbf{a}_{\geqslant l}$ must be good. Clearly $t_i \leqslant_{\mathsf{emb}} t_j$ $(0 \leqslant i < j \leqslant k-1)$ is impossible as $\mathbf{t}$ is bad. Likewise, $t_i \leqslant_{\mathsf{emb}} a_j$ $(0 \leqslant i \leqslant k-1$ and $j = 0$ or $l \leqslant j)$ contradicts the badness of $\mathbf{t}$ since $a_j \leqslant_{\mathsf{emb}} t_m$ for some $m \geqslant k$—recall that a_0 is a proper subterm of t_k and if $j \geqslant l$ then $a_j \in A \setminus A'$—and thus $t_i \leqslant_{\mathsf{emb}} t_j$. Hence we must have $a_i \leqslant_{\mathsf{emb}} a_j$ for some $0 \leqslant i < j$ (and $i, j \notin \{1, \ldots, l-1\}$), contradicting the badness of $\mathbf{a}$. Hence $>_{\mathsf{emb}}$ is a partial well-order on A. From Higman's Lemma we infer that $(>_{\mathsf{emb}})_{\mathsf{emb}}$ is a partial well-order on A^*.

Since $>$ is a partial well-order on $\mathcal{F}$, the infinite sequence $\mathbf{f}$ contains a chain, say $\mathbf{f}_\phi$. Consider the infinite sequence $\mathbf{w}_\phi$ over A^*. Since $(>_{\mathsf{emb}})_{\mathsf{emb}}$ is a partial well-order on A^*, we have $w_{\phi(i)}$ $(\leqslant_{\mathsf{emb}})_{\mathsf{emb}}$ $w_{\phi(j)}$ for some $0 \leqslant i < j$. A straightforward case analysis reveals that $f_{\phi(i)} \leqslant f_{\phi(j)}$ and $w_{\phi(i)}$ $(\leqslant_{\mathsf{emb}})_{\mathsf{emb}}$ $w_{\phi(j)}$ imply $t_{\phi(i)} \leqslant_{\mathsf{emb}} t_{\phi(j)}$. Hence we obtained a contradiction with the badness of $\mathbf{t}$. We conclude there are no bad sequences over $\mathcal{T}(\mathcal{F})$. □

Exercises

B.13 Let $\mathcal{F}$ be a signature and suppose $\sqsupset$ is a proper order on $\mathcal{T}(\mathcal{F})$. Define $>$ as the least proper order on $\mathcal{T}(\mathcal{F})$ satisfying the following two properties:

1. $>$ has the subterm property
2. $f(s_1, \ldots, s_n) > g(t_1, \ldots, t_m)$ whenever $f(s_1, \ldots, s_n) \sqsupset g(t_1, \ldots, t_m)$ and there exist $i_1, \ldots, i_m$ with $1 \leqslant i_1 < \cdots < i_m \leqslant n$ such that $s_{i_j} \geqslant t_j$ for all $j \in \{1, \ldots, m\}$

a Suppose $\sqsupset$ is a partial well-order on $\mathcal{T}(\mathcal{F})$. Prove that $>$ is a partial well-order on $\mathcal{T}(\mathcal{F})$.

b Show that part (a) generalizes (the general version of) Kruskal's Tree Theorem.

Bibliographic Notes

Dickson's Lemma is from Dickson [36]. Higman's Lemma is from Higman [51]. Kruskal's Tree Theorem is from Kruskal [82].

Bibliography

[1] Het cola-gen. *Natuurwetenschap & Techniek*, 73(1), 2005. In Dutch.

[2] Beniamino Accattoli. An abstract factorization theorem for explicit substitutions. In Tiwari [128], pages 6–21. doi: 10.4230/LIPIcs.RTA.2012.6.

[3] Thomas Arts and Jürgen Giesl. Termination of term rewriting using dependency pairs. *Theoretical Computer Science*, 236:133–178, 2000. doi: 10.1016/S0304-3975(99)00207-8.

[4] Franz Baader, editor. *Proc. 18th International Conference on Rewriting Techniques and Applications*, volume 4533 of *Lecture Notes in Computer Science*. Springer, 2007. doi: 10.1007/978-3-540-73449-9.

[5] Franz Baader and Tobias Nipkow. *Term Rewriting and All That*. Cambridge University Press, 1998. doi: 10.1017/CBO9781139172752.

[6] Leo Bachmair. *Canonical Equational Proofs*. Birkhäuser, 1991. doi: 10.1007/978-1-4684-7118-2.

[7] Leo Bachmair and Nachum Dershowitz. Commutation, transformation, and termination. In *Proc. 8th International Conference on Automated Deduction*, volume 230 of *Lecture Notes in Computer Science*, pages 5–20, 1986. doi: 10.1007/3-540-16780-3_76.

[8] Leo Bachmair and Nachum Dershowitz. Critical pair criteria for completion. *Journal of Symbolic Computation*, 6(1):1–18, 1988. doi: 10.1016/S0747-7171(88)80018-X.

[9] Leo Bachmair and Nachum Dershowitz. Equational inference, canonical proofs, and proof orderings. *Journal of the ACM*, 41(2):236–276, 1994. doi: 10.1145/174652.174655.

[10] Leo Bachmair, Nachum Dershowitz, and Jieh Hsiang. Orderings for equational proofs. In *Proc. 1st IEEE Symposium on Logic in Computer Science*, pages 346–357, 1986.

[11] Hendrik Pieter Barendregt. *The Lambda Calculus, its Syntax and Semantics*. North-Holland, 2nd edition, 1984.

[12] Marc Bezem, Jan Willem Klop, and Vincent van Oostrom. Diagram techniques for confluence. *Information and Computation*, 141(2):172–204, 1998. doi: 10.1006/inco.1997.2683.

[13] Garrett Birkhoff. On the structure of abstract algebras. *Mathematical Proceedings of the Cambridge Philosophical Society*, 31(4):433–454, 1935. doi: 10.1017/S0305004100013463.

[14] Nikolai Bjørner and Andrei Voronkov, editors. *Proc. 18th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning*, volume 7180 of *Lecture Notes in Computer Science (Advanced Research in Computing and Software Science)*. Springer, 2012. doi: 10.1007/978-3-642-28717-6.

[15] Ronald V. Book, editor. *Proc. 4th International Conference on Rewriting Techniques and Applications*, volume 488 of *Lecture Notes in Computer Science*. Springer, 1991. doi: 10.1007/3-540-53904-2.

[16] Nicolaas Govert de Bruijn. A note on weak diamond properties. Memorandum 78-08, Eindhoven University of Technology, 1978.

[17] Georg Cantor. Beiträge zur Begründung der transfiniten Mengenlehre II. *Mathematische Annalen*, 49:207–246, 1897.

[18] Hubert Comon, editor. *Proc. 8th International Conference on Rewriting Techniques and Applications*, volume 1232 of *Lecture Notes in Computer Science*. Springer, 1997. doi: 10.1007/3-540-62950-5.

[19] Pierre-Louis Curien and Giorgio Ghelli. On confluence for weakly normalizing systems. In Book [15], pages 215–225. doi: 10.1007/3-540-53904-2_98.

[20] Haskell B. Curry. Grundlagen der kombinatorischen Logik. *American Journal of Mathematics*, 52: 500–536 and 789–834, 1930.

[21] Haskell B. Curry and Robert Feys. *Combinatory Logic, Vol. I*. North-Holland, 1958.

[22] Haskell B. Curry, J. Roger Hindley, and Jonathan P. Seldin. *Combinatory Logic, Vol. II*, volume 65 of *Studies in Logic*. North-Holland, 1972.

[23] Max Dauchet. Simulation of Turing machines by a regular rewrite rule. *Theoretical Computer Science*, 103(2):409–420, 1992. doi: 10.1016/0304-3975(92)90022-8.

[24] Max Dauchet, Thierry Heuillard, Pierre Lescanne, and Sophie Tison. Decidability of the confluence of finite ground term rewriting systems and of other related term rewriting systems. *Information and Computation*, 88(2):187–201, 1990. doi: 10.1016/0890-5401(90)90015-A.

[25] Nachum Dershowitz. A note on simplification orderings. *Information Processing Letters*, 9(5):212–215, 1979. doi: 10.1016/0020-0190(79)90071-1.

[26] Nachum Dershowitz. Termination of linear rewriting systems (preliminary version). In *Proc. 8th International Colloquium on Automata, Languages and Programming*, volume 115, pages 448–458, 1981. doi: 10.1007/3-540-10843-2_36.

[27] Nachum Dershowitz. Orderings for term-rewriting systems. *Theoretical Computer Science*, 17:279–301, 1982. doi: 10.1016/0304-3975(82)90026-3.

[28] Nachum Dershowitz. Termination of rewriting. *Journal of Symbolic Computation*, 3(1-2):69–116, 1987. doi: 10.1016/S0747-7171(87)80022-6.

[29] Nachum Dershowitz, editor. *Proc. 3rd International Conference on Rewriting Techniques and Applications*, volume 355 of *Lecture Notes in Computer Science*. Springer, 1989. doi: 10.1007/3-540-51081-8.

[30] Nachum Dershowitz. Termination by abstraction. In *Proc. 20th International Conference on Logic Programming*, volume 3132 of *Lecture Notes in Computer Science*, pages 1–18, 2004. doi: 10.1007/978-3-540-27775-0_1.

[31] Nachum Dershowitz. Open. Closed. Open. In Giesl [45], pages 276–393. doi: 10.1007/978-3-540-32033-3_28.

[32] Nachum Dershowitz. Jumping and escaping: Modular termination and the abstract path ordering. *Theoretical Computer Science*, 464:35–47, 2012. doi: 10.1016/j.tcs.2012.09.013.

[33] Nachum Dershowitz and Jean-Pierre Jouannaud. Rewrite systems. In *Handbook of Theoretical Computer Science*, volume B: Formal Models and Semantics, chapter 6, pages 243–320. Elsevier, 1990. doi: 10.1016/B978-0-444-88074-1.50011-1.

[34] Nachum Dershowitz and Zohar Manna. Proving termination with multiset orderings. *Communications of the ACM*, 22(8):465–476, 1979. doi: 10.1145/359138.359142.

[35] Jeremy Dick, John Kalmus, and Ursula Martin. Automating the Knuth Bendix ordering. *Acta Informatica*, 28:95–119, 1990. doi: 10.1007/BF01237233.

[36] Leonard Eugene Dickson. Finiteness of the odd perfect and primitive abundant numbers with n distinct prime factors. *American Journal of Mathematics*, 35(4):413–426, 1913. doi: 10.2307/2370405.

[37] Henk Doornbos, Roland Carl Backhouse, and Jaap van der Woude. A calculational approach to mathematical induction. *Theoretical Computer Science*, 179(1-2):103–135, 1997. doi: 10.1016/S0304-3975(96)00154-5.

[38] Elmar Eder. Properties of substitutions and unifications. *Journal of Symbolic Computation*, 1(1): 31–46, 1985. doi: 10.1016/S0747-7171(85)80027-4.

[39] Jörg Endrullis and Jan Willem Klop. De Bruijn's weak diamond property revisited. *Indagationes Mathematicae*, 24(4):1050–1072, 2013. doi: 10.1016/j.indag.2013.08.005.

[40] Jörg Endrullis, Jan Willem Klop, and Roy Overbeek. Decreasing diagrams with two labels are complete for confluence of countable systems. In Kirchner [71], pages 14:1–14:15. doi: 10.4230/LIPIcs.FSCD.2018.14.

[41] Bertram Felgenhauer and Vincent van Oostrom. Proof orders for decreasing diagrams. In Raamsdonk [113], pages 174–189. doi: 10.4230/LIPIcs.RTA.2013.174.

[42] Maribel Fernández, editor. *Proc. 26th International Conference on Rewriting Techniques and Applications*, volume 36 of *Leibniz International Proceedings in Informatics*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2015.

[43] Harald Ganzinger, editor. *Proc. 7th International Conference on Rewriting Techniques and Applications*, volume 1103 of *Lecture Notes in Computer Science*. Springer, 1996. doi: 10.1007/3-540-61464-8.

[44] Alfons Geser. *Relative Termination*. PhD thesis, Universität Passau, 1990. Available as technical report 91-03.

[45] Jürgen Giesl, editor. *Proc. 16th International Conference on Rewriting Techniques and Applications*, volume 3467 of *Lecture Notes in Computer Science*. Springer, 2005. doi: 10.1007/b135673.

[46] Guillem Godoy, Ashish Tiwari, and Rakesh M. Verma. Characterizing confluence by rewrite closure and right ground term rewrite systems. *Applicable Algebra in Engineering, Communication and Computing*, 15(1):13–36, 2004. doi: 10.1007/s00200-004-0148-6.

[47] Bernhard Gramlich. Abstract relations between restricted termination and confluence properties of rewrite systems. *Fundamenta Informaticae*, 24(1-2):2–23, 1995. doi: 10.3233/FI-1995-24121.

[48] Bernhard Gramlich. On interreduction of semi-complete term rewriting systems. *Theoretical Computer Science*, 258(1-2):435–451, 2001. doi: 10.1016/S0304-3975(00)00030-X.

[49] David Gries. *The Science of Programming*. Texts and Monographs in Computer Science. Springer, 1981. doi: 10.1007/978-1-4612-5983-1.

[50] Gerhard Hessenberg. *Grundbegriffe der Mengenlehre*. Göttingen, 1906.

[51] Graham Higman. Ordering by divisibility in abstract algebras. *Proceedings of the London Mathematical Society*, s3-2(1):326–336, 1952. doi: 10.1112/plms/s3-2.1.326.

[52] J. Roger Hindley. *The Church–Rosser Property and a Result in Combinatory Logic*. PhD thesis, University of Newcastle-upon-Tyne, 1964.

[53] J. Roger Hindley and Jonathan P. Seldin. *Lambda-Calculus and Combinators, an Introduction*. Cambridge University Press, 2008.

[54] Nao Hirokawa, Aart Middeldorp, and Georg Moser. Leftmost outermost revisited. In Fernández [42], pages 209–222. doi: 10.4230/LIPIcs.RTA.2015.209.

[55] Nao Hirokawa, Aart Middeldorp, Christian Sternagel, and Sarah Winkler. Abstract completion, formalized. *Logical Methods in Computer Science*, 15(3):19:1–19:42, 2019. doi: 10.23638/LMCS-15(3:19)2019.

[56] Jieh Hsiang, editor. *Proc. 6th International Conference on Rewriting Techniques and Applications*, volume 914 of *Lecture Notes in Computer Science*. Springer, 1995. doi: 10.1007/3-540-59200-8.

[57] Gérard Huet. Confluent reductions: Abstract properties and applications to term rewriting systems. *Journal of the ACM*, 27(4):797–821, 1980. doi: 10.1145/322217.322230.

[58] Gérard Huet. A complete proof of correctness of the Knuth–Bendix completion algorithm. *Journal of Computer and System Sciences*, 23(1):11–21, 1981. doi: 10.1016/0022-0000(81)90002-7.

[59] Gérard Huet and Dallas Lankford. On the uniform halting problem for term rewriting systems. Technical Report 283, INRIA, 1978.

[60] Gérard Huet and Jean-Jacques Lévy. Computations in orthogonal rewriting systems, I. In *Computational Logic – Essays in Honor of Alan Robinson*, pages 395–414. The MIT Press, 1991.

[61] Sayaka Ishizuki, Michio Oyamaguchi, and Masahiko Sakai. Conditions for confluence of innermost terminating term rewriting systems. *Applicable Algebra in Engineering, Communication and Computing*, 30(4):349–360, 2019. doi: 10.1007/s00200-018-0377-8.

[62] Matthias Jantzen. A note on a special one-rule semi-Thue system. *Information Processing Letters*, 21(3):135–140, 1985. doi: 10.1016/0020-0190(85)90018-3.

[63] Jean-Pierre Jouannaud and Pierre Lescanne. On multiset orderings. *Information Processing Letters*, 15(2):57–63, 1982. doi: 10.1016/0020-0190(82)90107-7.

[64] Lukasz Kaiser. Confluence of right ground term rewriting systems is decidable. In *Proc. 8th International Conference on Foundations of Software Science and Computational Structures*, volume 3441 of *Lecture Notes in Computer Science*, pages 470–489, 2005. doi: 10.1007/978-3-540-31982-5_30.

[65] Sam Kamin and Jean-Jacques Lévy. Two generalizations of the recursive path ordering. Unpublished manuscript, University of Illinois, 1980.

[66] Deepak Kapur and Paliath Narendran. A finite Thue system with decidable word problem and without equivalent finite canonical system. *Theoretical Computer Science*, 35:337–344, 1985. doi: 10.1016/0304-3975(85)90023-4.

[67] Deepak Kapur, David R. Musser, and Paliath Narendran. Only prime superpositions need be considered in the Knuth–Bendix completion procedure. *Journal of Symbolic Computation*, 6(1):19–36, 1988. doi: 10.1016/S0747-7171(88)80019-1.

[68] Deepak Kapur, Paliath Narendran, and Friedrich Otto. On ground-confluence of term rewriting systems. *Information and Computation*, 86(1):14–31, 1990. doi: 10.1016/0890-5401(90)90023-B.

[69] Delia Kesner and Brigitte Pientka, editors. *Proc. 1st International Conference on Formal Structures for Computation and Deduction*, volume 52 of *Leibniz International Proceedings in Informatics*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2016.

[70] Laurence Kirby and Jeff Paris. Accessible independency results for Peano arithmetic. *Bulletin of the London Mathematical Society*, 14:285–325, 1982. doi: 10.1112/blms/14.4.285.

[71] Hélène Kirchner, editor. *Proc. 3rd International Conference on Formal Structures for Computation and Deduction*, volume 108 of *Leibniz International Proceedings in Informatics*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018.

[72] Jan Willem Klop. *Combinatory Reduction Systems*. PhD thesis, Utrecht University, 1980.

[73] Jan Willem Klop. Term rewriting systems: A tutorial. *Bulletin of the EATCS*, 32:143–182, 1987.

[74] Jan Willem Klop. Term rewriting systems. In Samsom Abramsky, Dov M. Gabbay, and Thomas S. E. Maibaum, editors, *Handbook of Logic in Computer Science*, volume 2, pages 1–116. Oxford University Press, 1992.

[75] Jan Willem Klop. Term rewriting systems. In *Handbook of Logic in Computer Science*, volume 2, pages 1–116. Oxford University Press, 1992.

[76] Jan Willem Klop, Vincent van Oostrom, and Roel de Vrijer. A geometric proof of confluence by decreasing diagrams. *Journal of Logic and Computation*, 10(3):437–460, 2000. doi: 10.1093/logcom/10.3.437.

[77] Donald E. Knuth and Peter B. Bendix. Simple word problems in universal algebras. In John Leech, editor, *Computational Problems in Abstract Algebra*, pages 263–297. Pergamon Press, 1970.

[78] Naoki Kobayashi, editor. *Proc. 6th International Conference on Formal Structures for Computation and Deduction*, volume 195 of *Leibniz International Proceedings in Informatics*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

[79] Christina Kohl and Aart Middeldorp. A formalization of the development closedness criterion for left-linear term rewrite systems. In *Proc. 12th ACM SIGPLAN International Conference on Certified Programs and Proofs*, pages 197–210, 2023. doi: 10.1145/3573105.3575667.

[80] Christina Kohl and Aart Middeldorp. Formalizing almost development closed critical pairs. In *Proc. 14th International Conference on Interactive Theorem Proving*, volume 268 of *Leibniz International Proceedings in Informatics*, pages 38:1–38:8, 2023. doi: 10.4230/LIPIcs.ITP.2023.38.

[81] Konstantin Korovin and Andrei Voronkov. Orienting rewrite rules with the Knuth–Bendix order. *Information and Computation*, 183(2):165–186, 2003. doi: 10.1016/S0890-5401(03)00021-X.

[82] Joseph Bernard Kruskal. Well-quasi-ordering, the tree theorem, and Vazsonyi's conjecture. *Transactions of the American Mathematical Society*, 95(2):210–225, 1960. doi: 10.2307/1993287.

[83] Dallas Lankford. Canonical algebraic simplification in computational logic. Technical Report ATP-25, University of Texas, Austin, TX, USA, 1975.

[84] Bernd Löchner. Things to know when implementing KBO. *Journal of Automated Reasoning*, 36(4): 289–310, 2006. doi: 10.1007/s10817-006-9031-4.

[85] Salvador Lucas. Context-sensitive computations in functional and functional logic programs. *Journal of Functional and Logic Programming*, 1998(1), 1998.

[86] Salvador Lucas. Context-sensitive rewriting strategies. *Information and Computation*, 178(1):294–343, 2002. doi: 10.1006/inco.2002.3176.

[87] Salvador Lucas. Context-sensitive rewriting. *ACM Computing Surveys*, 53(4), 2020. doi: 10.1145/3397677.

[88] Michel Ludwig and Uwe Waldmann. An extension of the Knuth–Bendix ordering with LPO-like properties. In *Proc. 14th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning*, volume 4790 of *Lecture Notes in Artificial Intelligence*, pages 348–362, 2007. doi: 10.1007/978-3-540-75560-9_26.

[89] Christopher Lynch, editor. *Proc. 21st International Conference on Rewriting Techniques and Applications*, volume 6 of *Leibniz International Proceedings in Informatics*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2010.

[90] Alberto Martelli and Ugo Montanari. An efficient unification algorithm. *ACM Transactions on Programming Languages and Systems*, 4(2):258–282, 1982. doi: 10.1145/357162.357169.

[91] Yves Métivier. About the rewriting systems produced by the Knuth–Bendix completion algorithm. *Information Processing Letters*, 16(1):31–34, 1983. doi: 10.1016/0020-0190(83)90009-1.

[92] Aart Middeldorp, editor. *Proc. 12th International Conference on Rewriting Techniques and Applications*, volume 2051 of *Lecture Notes in Computer Science*. Springer, 2001. doi: 10.1007/3-540-45127-7.

[93] Aart Middeldorp and Bernhard Gramlich. Simple termination is difficult. *Applicable Algebra in Engineering, Communication and Computing*, 6:115–128, 1995. doi: 10.1007/BF01225647.

[94] Dale Miller, editor. *Proc. 2nd International Conference on Formal Structures for Computation and Deduction*, volume 84 of *Leibniz International Proceedings in Informatics*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017.

[95] Julian Nagele, Bertram Felgenhauer, and Aart Middeldorp. Improving automatic confluence analysis of rewrite systems by redundant rules. In Fernández [42], pages 257–268. doi: 10.4230/LIPIcs.RTA.2015.257.

[96] Friedrich Neurauter and Aart Middeldorp. Polynomial interpretations over the reals do not subsume polynomial interpretations over the integers. In Lynch [89], pages 243–258. doi: 10.4230/LIPIcs.RTA.2010.243.

[97] Max H. A. Newman. On theories with a combinatorial definition of equivalence. *Annals of Mathematics*, 43(2):223–243, 1942.

[98] Tobias Nipkow, editor. *Proc. 9th International Conference on Rewriting Techniques and Applications*, volume 1379 of *Lecture Notes in Computer Science*. Springer, 1998. doi: 10.1007/BFb0052355.

[99] Michael J. O'Donnell. *Computing in Systems Described by Equations*, volume 58 of *Lecture Notes in Computer Science*. Springer, 1977. doi: 10.1007/3-540-08531-9.

[100] Vincent van Oostrom. Confluence by decreasing diagrams. *Theoretical Computer Science*, 126(2): 259–280, 1994. doi: 10.1016/0304-3975(92)00023-K.

[101] Vincent van Oostrom. Developing developments. *Theoretical Computer Science*, 175(1):159–181, 1997. doi: 10.1016/S0304-3975(96)00173-9.

[102] Vincent van Oostrom. Random descent. In Baader [4], pages 314–328. doi: 10.1007/978-3-540-73449-9_24.

[103] Vincent van Oostrom. Confluence by decreasing diagrams – Converted. In Voronkov [134], pages 306–320. doi: 10.1007/978-3-540-70590-1_21.

[104] Vincent van Oostrom. Z; syntax-free developments. In Kobayashi [78], pages 24:1–24:22. doi: 10.4230/LIPIcs.FSCD.2021.24.

[105] Vincent van Oostrom and Yoshihito Toyama. Normalisation by random descent. In Kesner and Pientka [69], pages 32:1–32:18. doi: 10.4230/LIPIcs.FSCD.2016.32.

[106] Michio Oyamaguchi and Yoshikatsu Ohta. A new parallel closed condition for Church–Rosser of left-linear term rewriting systems. In Comon [18], pages 187–201. doi: 10.1007/3-540-62950-5_70.

[107] Michio Oyamaguchi and Yoshikatsu Ohta. On the open problems concerning Church–Rosser of left-linear term rewriting systems. *IEICE Transactions on Information and Systems*, E87-D(1):290–298, 2004.

[108] Andreas Podelski and Andrey Rybalchenko. Transition invariants. In *Proc. 19th IEEE Symposium on Logic in Computer Science*, pages 32–41, 2004. doi: 10.1109/LICS.2004.1319598.

[109] Jaco van de Pol. Just-in-time: On strategy annotations. In *Proc. 1st International Workshop on Reduction Strategies in Rewriting and Programming*, volume 57 of *Electronic Notes in Theoretical Computer Science*, pages 41–63, 2001. doi: 10.1016/S1571-0661(04)00267-1.

[110] Jaco van de Pol. JITty: A rewriter with strategy annotations. In Tison [127], pages 367–370. doi: 10.1007/3-540-45610-4_26.

[111] Jaco van de Pol and Hans Zantema. Generalized innermost rewriting. In Giesl [45], pages 2–16. doi: 10.1007/978-3-540-32033-3_2.

[112] Christian Prehofer. On modularity in term rewriting and narrowing. In *Proc. 1st International Conference on Constraints in Computational Logics*, volume 845 of *Lecture Notes in Computer Science*, pages 253–268, 1994. doi: 10.1007/BFb0016858.

[113] Femke van Raamsdonk, editor. *Proc. 24th International Conference on Rewriting Techniques and Applications*, volume 21 of *Leibniz International Proceedings in Informatics*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2013.

[114] J. Alan Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM*, 12(1):23–41, 1965. doi: 10.1145/321250.321253.

[115] Barry K. Rosen. Tree-manipulating systems and Church–Rosser theorems. *Journal of the ACM*, 20 (1):160–187, 1973. doi: 10.1145/321738.321750.

[116] Michaël Rusinowitch. Path of subterms ordering and recursive decomposition ordering revisited. *Journal of Symbolic Computation*, 3(1-2):117–131, 1987. doi: 10.1016/S0747-7171(87)80023-8.

[117] Manfred Schmidt-Schauß, editor. *Proc. 22nd International Conference on Rewriting Techniques and Applications*, volume 10 of *Leibniz International Proceedings in Informatics*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2011.

[118] Moses Schönfinkel. Über die Bausteine der mathematischen Logik. *Mathematische Annalen*, 92(3-4): 305–316, 1924. doi: 10.1007/BF01448013.

[119] Dana S. Scott. A system of functional abstraction. Lecture Notes, Stanford University, 1963.

[120] Wayne Snyder. A fast algorithm for generating reduced ground rewriting systems from a set of ground equations. *Journal of Symbolic Computation*, 15(4):415–450, 1993. doi: 10.1006/jsco.1993.1029.

[121] John Staples. Church–Rosser theorems for replacement systems. In John Crosley, editor, *Algebra and Logic*, volume 450 of *Lecture Notes in Mathematics*, pages 291–307. Springer, 1975.

[122] Joachim Steinbach. Simplification orderings: History of results. *Fundamenta Informaticae*, 24(1-2): 47–87, 1995. doi: 10.3233/FI-1995-24123.

[123] Christian Sternagel and René Thiemann. Formalizing Knuth–Bendix orders and Knuth–Bendix completion. In Raamsdonk [113], pages 286–301. doi: 10.4230/LIPIcs.RTA.2013.286.

[124] Aaron Stump, Hans Zantema, Garrin Kimmell, and Roba El Haj Omar. A rewriting view of simple typing. *Logical Methods in Computer Science*, 9(1):1–29, 2012. doi: 10.2168/LMCS-9(1:4)2013.

[125] Edward Szpilrajn. Sur l'extension de l'ordre partiel. *Fundamenta Mathematicae*, 16:386–389, 1930.

[126] Terese, editor. *Term Rewriting Systems*, volume 55 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 2003.

[127] Sophie Tison, editor. *Proc. 13th International Conference on Rewriting Techniques and Applications*, volume 2378 of *Lecture Notes in Computer Science*. Springer, 2002. doi: 10.1007/3-540-45610-4.

[128] Ashish Tiwari, editor. *Proc. 23rd International Conference on Rewriting Techniques and Applications*, volume 15 of *Leibniz International Proceedings in Informatics*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2012.

[129] Hélène Touzet. Encoding the Hydra battle as a rewrite system. In *Proc. 23rd International Symposium on Mathematical Foundations of Computer Science*, volume 1450 of *Lecture Notes in Computer Science*, pages 267–276, 1998. doi: 10.1007/BFb0055776.

[130] Yoshihito Toyama. Commutativity of term rewriting systems. In Kazuhiro Fuchi and Laurent Kott, editors, *Programming of Future Generation Computers II*, pages 393–407. North-Holland, 1988.

[131] Yoshihito Toyama. Strong sequentiality of left-linear overlapping term rewriting systems. In *Proc. 7th IEEE Symposium on Logic in Computer Science*, pages 253–268, 1992. doi: 10.1109/LICS.1992.185540.

[132] David A. Turner. A new implementation technique for applicative languages. *Software Practice and Experience*, 9:31–49, 1979. doi: 10.1002/spe.4380090105.

[133] Vincent van Oostrom and Roel de Vrijer. Four equivalent equivalences of reductions. In *Proc. 2nd International Workshop on Reduction Strategies in Rewriting and Programming*, volume 70(6) of *Electronic Notes in Theoretical Computer Science*, pages 21–61, 2002. doi: 10.1016/S1571-0661(04)80599-1.

[134] Andrei Voronkov, editor. *Proc. 19th International Conference on Rewriting Techniques and Applications*, volume 5117 of *Lecture Notes in Computer Science*. Springer, 2008. doi: 10.1007/978-3-540-70590-1.

[135] Pum Walters and Hans Zantema. Rewrite systems for integer arithmetic. In Hsiang [56], pages 324–338. doi: 10.1007/3-540-59200-8_67.

[136] Alex J. Wilkie. On exponentiation – A solution to Tarski's high school algebra problem. *Quaderni di Matematica*, 6:107–129, 2000.

[137] Franz Winkler and Bruno Buchberger. A criterion for eliminating unnecessary reductions in the Knuth–Bendix algorithm. In *Proceedings of the Colloquium on Algebra, Combinatorics and Logic in Computer Science, Vol. II*, volume 42 of *Colloquia Mathematica Societatis J. Bolyai*, pages 849–869, 1986.

[138] Sarah Winkler, Harald Zankl, and Aart Middeldorp. Ordinals and Knuth–Bendix orders. In Bjørner and Voronkov [14], pages 420–434. doi: 10.1007/978-3-642-28717-6_33.

[139] Akihisa Yamada, Keiichirou Kusakari, and Toshiki Sakabe. A unified ordering for termination proving. *Science of Computer Programming*, 111:110–134, 2015. doi: 10.1016/j.scico.2014.07.009.

[140] Harald Zankl, Nao Hirokawa, and Aart Middeldorp. KBO orientability. *Journal of Automated Reasoning*, 43(2):173–201, 2009. doi: 10.1007/s10817-009-9131-z.

[141] Hans Zantema. Termination of term rewriting: Interpretation and type elimination. *Journal of Symbolic Computation*, 17(1):23–50, 1994. doi: 10.1006/jsco.1994.1003.

[142] Hans Zantema. Termination. In Terese [126], chapter 6, pages 181–259.

Index