Logicality of Conditional Term Rewrite Systems

1999年3月

山田俊行

Logicality of Conditional Term Rewrite Systems

Doctoral Dissertation

Doctoral Program in Engineering
University of Tsukuba

Toshiyuki Yamada

March, 1999

Contents

1 **Introduction** 5
1.1 Background 5
1.2 Objectives 7
1.3 Structure of the Thesis 7

2 **Preliminaries** 9
2.1 Relations 9
2.2 Abstract Rewrite Systems 11
2.3 Terms 16
2.4 Conditional Equational Logic 19
2.5 Conditional Term Rewrite Systems 23

3 **Soundness and Completeness** 27
3.1 Characterizing Semantic Equivalence 27
3.2 Fixpoint Theorem for Relations 31
3.3 Soundness and Completeness of CEL 34
3.4 Soundness and Completeness of Semi-Equational CTRSs 36

4 **Logicality of CTRSs** 41
4.1 Logicality 41
4.2 Logicality of Join CTRSs 43
4.3 Logicality of Oriented CTRSs 44
4.4 Equivalence of Different Types of CTRSs 53
4.5 Systematic Analysis of Different Types of Relations 62

5 **Ensuring Sufficient Conditions for Logicality** 67
5.1 Termination and Decidability of Rewrite Relations 67
5.2 Confluence and Level-Confluence 71
5.3 Stability and Strong Irreducibility 74

6 Concluding Remarks 79
6.1 Research Contributions 79
6.2 Remarks on Applications 80

Bibliography 83

Index 88

List of Notations 91

Chapter 1

Introduction

1.1 Background

Equation is an essential concept in mathematics, logic, and computer science. They are used to give definitions of objects, to specify algorithms for computation, and to express mathematical knowledge. When reasoning about equations, we replace an expression by another one that is equal. Equational reasoning with directional use of equations leads to the concept of term rewriting. A term rewrite system is a set of directed equations, called rewrite rules. For example, consider the following term rewrite system, which specifies addition (+) and the Fibonacci function (fib) defined on natural numbers encoded by zero ($\mathsf{0}$) and successor function (S):

$$\left\{ \begin{array}{rcl} \mathsf{0} + \mathrm{x} & \to & \mathrm{x} \\ \mathsf{S}(\mathrm{x}) + \mathrm{y} & \to & \mathsf{S}(\mathrm{x} + \mathrm{y}) \\ \mathsf{fib}(\mathsf{0}) & \to & \mathsf{0} \\ \mathsf{fib}(\mathsf{S}(\mathsf{0})) & \to & \mathsf{S}(\mathsf{0}) \\ \mathsf{fib}(\mathsf{S}(\mathsf{S}(\mathrm{x}))) & \to & \mathsf{fib}(\mathrm{x}) + \mathsf{fib}(\mathsf{S}(\mathrm{x})) \end{array} \right\}.$$

By repeated applications of oriented equations, the term $\mathsf{fib}(\mathsf{S}(\mathsf{S}(\mathsf{S}(\mathsf{0}))))$ rewrites to $\mathsf{S}(\mathsf{S}(\mathsf{0}))$ as follows:

$$\begin{array}{rclcl} \mathsf{fib}(\mathsf{S}(\mathsf{S}(\mathsf{S}(\mathsf{0})))) & \to & \mathsf{fib}(\mathsf{S}(\mathsf{0})) + \mathsf{fib}(\mathsf{S}(\mathsf{S}(\mathsf{0}))) & \to & \mathsf{S}(\mathsf{0}) + \mathsf{fib}(\mathsf{S}(\mathsf{S}(\mathsf{0}))) \\ & \to & \mathsf{S}(\mathsf{0} + \mathsf{fib}(\mathsf{S}(\mathsf{S}(\mathsf{0})))) & \to & \mathsf{S}(\mathsf{fib}(\mathsf{S}(\mathsf{S}(\mathsf{0})))) \\ & \to & \mathsf{S}(\mathsf{fib}(\mathsf{0}) + \mathsf{fib}(\mathsf{S}(\mathsf{0}))) & \to & \mathsf{S}(\mathsf{0} + \mathsf{fib}(\mathsf{S}(\mathsf{0}))) \\ & \to & \mathsf{S}(\mathsf{0} + \mathsf{S}(\mathsf{0})) & \to & \mathsf{S}(\mathsf{S}(\mathsf{0})). \end{array}$$

The process of rewriting is considered as a proof in equational reasoning. In the above example, the equality of the two terms $\mathsf{fib}(\mathsf{S}(\mathsf{S}(\mathsf{S}(\mathsf{0}))))$ and $\mathsf{S}(\mathsf{S}(\mathsf{0}))$ is derived from the equational axioms expressed by the term rewrite system. We can also consider the rewriting process as computation. In the above example, the value $\mathsf{S}(\mathsf{S}(0))$ of the expression $\mathsf{fib}(\mathsf{S}(\mathsf{S}(\mathsf{S}(0))))$ is computed by rewriting. From this viewpoint, a term rewrite system can be regarded as a program. Theories of term rewriting are applied to a wide range of problems in computation, programming, and logic; for example, analysis and implementation of abstract data type specifications, design of functional and logic programming languages, and automated theorem proving.

In order to enhance the descriptive power of rewrite systems, various extensions of term rewrite systems have been proposed. Conditional term rewriting is one of important extensions. Conditional term rewrite systems (CTRSs) can be used to simulate theories specified by conditional equations. They were first studied in the theory of abstract data types because specifications based on conditional equations arise naturally in algebraic specifications. Many recent proposals for programming languages that integrate functional programming and logic programming paradigms can be modeled by CTRSs. A rewrite rule of a CTRS is a directed equation with conditions. The conditions consist of a possibly empty sequence of equations. For example, consider the following specification of the function merge for merging two sorted lists into one list:

$$\left\{\begin{array}{rcll} 0 > x & \to & \mathsf{F} & \\ \mathsf{S}(x) > 0 & \to & \mathsf{T} & \\ \mathsf{S}(x) > \mathsf{S}(y) & \to & x > y & \\ \mathsf{merge}([],\ ys) & \to & ys & \\ \mathsf{merge}(xs,\ []) & \to & xs & \\ \mathsf{merge}(x : xs,\ y : ys) & \to & y : \mathsf{merge}(x : xs,\ ys) & \Leftarrow\ x > y \approx \mathsf{T} \\ \mathsf{merge}(x : xs,\ y : ys) & \to & x : \mathsf{merge}(xs,\ y : ys) & \Leftarrow\ x > y \approx \mathsf{F} \end{array}\right\}.$$

Here, "[]" denotes the empty list and ":" is the list constructor. We can apply rewrite rules provided the conditions are satisfied. To determine whether a condition is satisfied, three main types of conditional rewriting are considered in the literature: semi-equational, join, and oriented systems. In a semi-equational system the conditions in the conditional rewrite rules are checked by allowing rewriting in both directions. In a join system the satisfiability of conditions is determined by rewriting to a common term. In an oriented system conditions are interpreted as rewriting from left to right. If we consider the above CTRS as an oriented system, then the two sorted lists $0 : \mathsf{S}(\mathsf{S}(0)) : []$ and $\mathsf{S}(0) : []$ are merged as follows:

$$\begin{array}{rcl} \mathsf{merge}(0 : \mathsf{S}(\mathsf{S}(0)) : [],\ \mathsf{S}(0) : []) & \to & 0 : \mathsf{merge}(\mathsf{S}(\mathsf{S}(0)) : [],\ \mathsf{S}(0) : []) \\ & \to & 0 : \mathsf{S}(0) : \mathsf{merge}(\mathsf{S}(\mathsf{S}(0)) : [],\ []) \\ & \to & 0 : \mathsf{S}(0) : \mathsf{S}(\mathsf{S}(0)) : []. \end{array}$$

Note that we apply the last rule in the first rewrite step because the condition is satisfied by rewriting from left to right: $0 > \mathsf{S}(0) \to \mathsf{F}$. Likewise, for the second rewrite step, the satisfiability of the condition is determined by unidirectional rewriting: $\mathsf{S}(\mathsf{S}(0)) > \mathsf{S}(0) \rightarrow \mathsf{S}(0) > 0 \rightarrow \mathsf{T}$. Recently, oriented systems emerged as the most natural type of conditional rewriting when modeling logic and functional programming, especially when allowing so-called extra variables in the conditions and right-hand sides of rewrite rules (e.g. [ALS94],[Han95],[SMI95]).

A conditional rewrite system is called logical if it has the same logical strength as the underlying conditional equational system. Logicality is important because it implies that an equation $s \approx t$ is provable by rewriting ($s \leftrightarrow^* t$) if and only if it is valid in all models of the underlying conditional equational system. Consequently, logicality is a minimum requirement for equational theorem provers and declarative programming languages based on conditional rewriting. Moreover, logicality acts as a bridge between the operational, proof-theoretical, and algebraic semantics of functional-logic programming languages (Hamana [Ham97]). Rewriting in a semi-equational system is very close to equational reasoning in the underlying conditional equational system and hence it is not surprising that semi-equational systems are logical. However, from a rewriting point of view, semi-equational systems are unnatural because the bidirectional use of rewrite rules in the conditions goes against the spirit of rewriting. Kaplan showed that join systems are logical, provided they are confluent (see [Kap84]). In contrast to join systems, confluence is insufficient for ensuring logicality of oriented systems.

1.2 Objectives

The aim of the thesis is to strengthen the proof-theoretical and model-theoretical basis of conditional term rewriting. More precisely, our main goals pursued in this thesis can be summarized as follows.

(1) We characterize soundness and completeness and give a rigorous proof that both conditional equational logic and semi-equational conditional term rewrite systems are sound and complete.
(2) We investigate sufficient conditions for logicality.
(3) We study techniques that ensure sufficient conditions for logicality.

1.3 Structure of the Thesis

The remainder of the thesis is organized as follows.

In Chapter 2 we introduce basic concepts for understanding conditional term rewriting. After a short review of definitions and basic results on relations in Section 2.1, we provide an introduction to the theory of abstract rewrite systems in Section 2.2. The fundamental properties which are common to all rewrite systems are discussed. In Section 2.3 we review concepts concerning terms, substitutions, and contexts. Section 2.4 gives the syntax and the semantics of conditional equational logic, which is the proof-theoretical basis of conditional term rewriting. Section 2.5 contains the definition of CTRS. Three different types of rewrite systems, semi-equational, join, and oriented systems, are introduced.

Chapter 3 is concerned with soundness and completeness. This chapter consists of two parts. The first part provides a technique for proving soundness and completeness. We provide a syntactic characterization of soundness and completeness in Section 3.1 and we review fixpoint theorem, in Section 3.2, for the purpose of analyzing properties of a relation. The second part presents soundness and completeness results based on the technique obtained in the first part. In Section 3.3 we give a rigorous proof of the soundness and completeness of conditional equational logic. In Section 3.4 we prove that every semi-equational CTRS is sound and complete with respect to its underlying conditional equational system.

In Chapter 4 we investigate what class of join and oriented CTRSs are sound and complete. The concept of logicality is introduced in Section 4.1. We also prove that join and oriented CTRSs are always sound in this section. Section 4.2 is a review of known logicality results for join CTRSs. In Section 4.3 new logicality results for oriented CTRSs are provided. In Section 4.4 we give sufficient conditions for logicality of join and oriented systems by imposing restrictions on semi-equational CTRSs. For understanding the principle of logicality, we make a systematic analysis of various combinations of relations which are induced by different types of CTRSs in Section 4.5.

In Chapter 5 we study how to ensure sufficient conditions for logicality and also distinguish the difference between properties which are used for a similar purpose. Section 5.1 deals with properties related to normalization and properties which ensure decidability of the rewrite relation induced by a CTRS. Section 5.2 is concerned with confluence and level-confluence. Section 5.3 deals in detail with properties which are used to restrict the reducibility of a term and discusses sufficient syntactic criteria for two important such properties, stability and strong irreducibility.

Finally, Chapter 6 concludes the thesis with some remarks. Section 6.1 summarizes the research contributions of the thesis. In Section 6.2 we illustrate the usefulness of the logicality results obtained in Chapter 4. We show that our results cover two important classes of conditional rewrite systems considered by Avenhaus and Loría-Sáenz [ALS94] and Suzuki *et al.* [SMI95]. Moreover, we give a solution to an open problem by Toyama [DJK91, Problem 16] by applying one of the new results.

Chapter 2

Preliminaries

The purpose of this chapter is to introduce basic concepts for understanding conditional term rewriting. In Section 2.1 we first review definitions and basic results of relations since properties of relations are frequently used in rewriting. In Section 2.2 the theory of abstract rewrite systems is introduced in order to treat rewrite systems in an abstract way. By taking this approach, fundamental properties that are independent of the term structure become clear. Section 2.3 is concerned with terms, the most fundamental object for term rewriting, and notions that deal with the term structure such as substitutions and contexts. In Section 2.4 we give the syntax and the semantics of conditional equational logic. It provides a formal justification for an inference in equational reasoning and is used as a proof-theoretical basis of conditional term rewriting. Finally, conditional term rewrite systems are defined in Section 2.5. Three different types (semi-equational, join, and oriented) of conditional rewrite systems are introduced.

2.1 Relations

Before dealing with rewrite systems, we first review terminology, notation, and properties concerned with relations. For an algebraic treatment of relations, the reader is referred to the textbook by Schmidt and Ströhlein [SS93].

Definition 2.1.1 (relations)
Let A be an arbitrary set. A *relation* on A is a subset of the Cartesian product $A \times A$. Consider two relations R and S on A.

- We write $(a, b) \in R$ and $a\ R\ b$ interchangeably.
- The *empty relation* is the empty set $\varnothing$.
- The *identity relation* on A is defined by $\mathrm{Id}_A = \{\ (a, a) \mid a \in A\ \}$.
- The *inverse* R^{-1} of R is defined by $R^{-1} = \{\ (b, a) \mid (a, b) \in R\ \}$.

- The *composition* $R \circ S$ of two relations R and S is the relation defined by $R \circ S = \{ (a, c) \mid (a, b) \in R \text{ and } (b, c) \in S \text{ for some } b \in A \}$.
- The relation $R \circ R$ is often written as R^2. This notation is generalized to the powers of a relation: $R^0 = \mathrm{Id}_A$ and $R^{n+1} = R^n \circ R$ for all $n \in \mathbb{N}$.

Since the composition operation of relations is associative, we can write $R \circ S \circ T$ by omitting the parentheses in the expressions $(R \circ S) \circ T$ and $R \circ (S \circ T)$.

Definition 2.1.2 (properties of a relation)
Let R be a relation on a set A.

- R is *reflexive* if $a \; R \; a$ for all $a \in A$.
- R is *symmetric* if $a \; R \; b$ implies $b \; R \; a$, for all $a, b \in A$.
- R is *transitive* if $a \; R \; b$ and $b \; R \; c$ implies $a \; R \; c$, for all $a, b, c \in A$.
- A reflexive, symmetric, and transitive relation is called an *equivalence relation*.

Reflexivity, symmetry, and transitivity of a relation R can be expressed more concisely by means of inclusions $\mathrm{Id}_A \subseteq R$, $R^{-1} \subseteq R$, and $R^2 \subseteq R$ respectively. Consequently, the inclusion $\mathrm{Id}_A \cup R^{-1} \cup R^2 \subseteq R$ is another way of saying that R is an equivalence relation.

Definition 2.1.3 (closure of relations)
Let $\mathcal{P}$ be a property of relations. The smallest relation that contains R and satisfies $\mathcal{P}$ is called the *$\mathcal{P}$-closure* of a relation R, or alternatively the closure of R with respective to $\mathcal{P}$.

Note that there are properties $\mathcal{P}$ for which $\mathcal{P}$-closure does not exist. All properties defined in Definition 2.1.2 admit closures, which are constructively characterized in the following lemma.

Lemma 2.1.4 (constructive characterization of closures)
Let R be a relation on a set A.

- The *reflexive closure* $R^=$ of R is the relation $R^= = R \cup \mathrm{Id}_A$.
- The *symmetric closure* R^{sym} of R is the relation $R^{\mathsf{sym}} = R \cup R^{-1}$.
- The *transitive closure* R^+ of R is the relation $R^+ = \bigcup_{n \in \mathbb{N}} R^{n+1}$.
- The *reflexive-transitive closure* R^* of R is the relation $R^* = \mathrm{Id}_A \cup R^+ = \bigcup_{n \in \mathbb{N}} R^n$.
- The *equivalence closure* R^{eqv} of R is the relation $R^{\mathsf{eqv}} = (R^{\mathsf{sym}})^*$.

Closures can also be viewed as operations. For example, $(\,\cdot\,)^=$, $(\,\cdot\,)^{\mathsf{sym}}$, $(\,\cdot\,)^+$, $(\,\cdot\,)^*$, and $(\,\cdot\,)^{\mathsf{eqv}}$ are considered as operations on relations. The following lemma describes properties that are common to closure operations.

Lemma 2.1.5 (properties of closure operations)
Let $\mathcal{P}$ be a property of relations and $\mathsf{cl}_{\mathcal{P}}(R)$ the $\mathcal{P}$-closure of a relation R. For all relations R and S, the operation $\mathsf{cl}_{\mathcal{P}}$ on relations satisfies the following properties.

(1) R satisfies the property $\mathcal{P}$ if and only if $\mathsf{cl}_{\mathcal{P}}(R) = R$.
(2) $R \subseteq \mathsf{cl}_{\mathcal{P}}(R)$ (*incrementality*).
(3) $\mathsf{cl}_{\mathcal{P}}(\mathsf{cl}_{\mathcal{P}}(R)) = \mathsf{cl}_{\mathcal{P}}(R)$ (*idempotency*).
(4) If $R \subseteq S$ then $\mathsf{cl}_{\mathcal{P}}(R) \subseteq \mathsf{cl}_{\mathcal{P}}(S)$ (*monotonicity*).

2.2 Abstract Rewrite Systems

This section contains a concise introduction to abstract rewriting, which is a theory of relations defined on an arbitrary set that focuses on their computational aspect. For more detailed descriptions on abstract rewriting, see, for example, Klop's survey [Klo92].

Definition 2.2.1 (abstract rewrite system)
An *abstract rewrite system* (ARS) over a set A is a relation on A. An ARS $\to \subseteq A \times A$ is also expressed as the pair $(A, \to)$ to make the underlying set A explicit.

In the theory of ARSs arrow symbols are often used to denote relations because we consider the relation as one step rewriting (from left to right).

Definition 2.2.2 (notation and terminology for ARSs)
Let $(A, \to)$ be an ARS.

- If $a \to b$ we say that there is a *rewrite step* from a to b.
- If $a \to^* b$ we say *a rewrites to b* and we call b a *reduct* of a.
- The symmetric closure $\to^{\mathsf{sym}}$ of $\to$ is written as $\leftrightarrow$.
- The equivalence closure $\to^{\mathsf{eqv}}$ of $\to$ is written as $\leftrightarrow^*$ and called *conversion* or *convertibility*.
- An element $a \in A$ is *irreducible* if there is no element b with $a \to b$. We write $a \to^! b$ if $a \to^* b$ and b is irreducible. An irreducible element is also called a *normal form* and the set of normal forms with respect to $\to$ is denoted by $\mathsf{NF}(\to)$.
- The inverse $\to^{-1}$ of $\to$ is denoted by $\leftarrow$. The relations ${}^{=}\!\leftarrow$, ${}^{+}\!\leftarrow$, ${}^{*}\!\leftarrow$, and ${}^{!}\!\leftarrow$. are similarly defined.

- The *joinability* relation $\downarrow$ is defined by $\downarrow = \to^* \circ {}^*\!\leftarrow$. So $a \downarrow b$ holds if there exists an element $c \in A$ such that $a \to^* c \ {}^*\!\leftarrow b$. Such an element c is called a *common reduct* of a and b.
- A *rewrite sequence* is a sequence of elements in A such that $a \to b$ for every adjacent pair of elements a and b in the sequence.

Basic properties of ARSs are introduced in the following definitions.

Definition 2.2.3 (confluence and related properties)

- An ARS $(A, \to)$ is *confluent* if $b_1 \ {}^*\!\leftarrow a \to^* b_2$ implies $b_1 \downarrow b_2$, for all elements $a, b_1, b_2 \in A$. Confluence is also called *Church-Rosser property* (CR).
- An ARS $(A, \to)$ is *locally confluent* if $b_1 \leftarrow a \to b_2$ implies $b_1 \downarrow b_2$, for all elements $a, b_1, b_2 \in A$. Local confluence is also called *weak Church-Rosser property* (WCR).
- An ARS $(A, \to)$ has *unique normal forms* (UN) if $b_1 \ {}^!\!\leftarrow a \to^! b_2$ implies $b_1 = b_2$, for all elements $a, b_1, b_2 \in A$.

Figure 2.1 illustrates these three properties graphically. In the figure solid arrows are universally quantified and dashed arrows depend on them with existential quantification. Observe that using inclusions of relations, CR, WCR, and UN are concisely expressed by ${}^*\!\leftarrow \circ \to^* \subseteq \downarrow$, $\leftarrow \circ \to \subseteq \downarrow$, and ${}^!\!\leftarrow \circ \to^! \subseteq \mathrm{Id}_A$ respectively. It is clear that every confluent ARS is locally confluent.

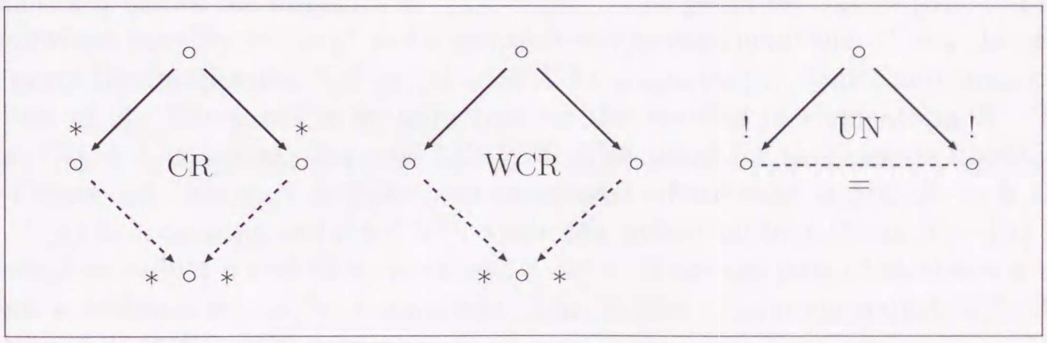


Figure 2.1: Properties of an ARS.

Definition 2.2.4 (properties concerned with normalization)

- An ARS is *strongly normalizing* (SN) or *terminating* if there are no infinite rewrite sequences.
- An ARS $(A, \to)$ is *weakly normalizing* (WN) if every element in A has an irreducible reduct.

Definition 2.2.5 (completeness and semi-completeness)

- A strongly normalizing and confluent ARS is called *complete*.
- A weakly normalizing and confluent ARS is called *semi-complete*.

It is easy to see that strongly normalization implies weak normalization and that completeness implies semi-completeness.

Since confluence is one of the most important properties in the theory of rewrite systems, various equivalent formulations of confluence have been studied. The following definition is required before we state the equivalence results in Lemma 2.2.7.

Definition 2.2.6

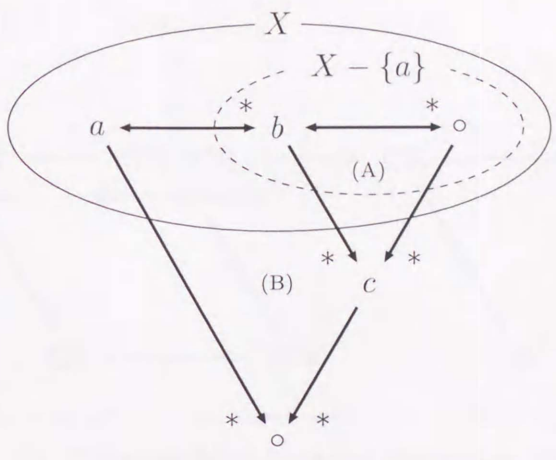
- A subset X of A *has a common reduct* if there exists an $a \in A$ such that $x \to^* a$ for all $x \in X$.
- A subset X of A is *connected* if $x \leftrightarrow^* y$ for all $x, y \in X$.

Lemma 2.2.7 (equivalent formulations of confluence)
Let $(A, \to)$ be an ARS. The following four statements are equivalent.
(1) $\to$ is confluent, i.e., ${}^*\!\leftarrow \circ \to^* \subseteq \downarrow$.
(2) The joinability relation $\downarrow$ is transitive, i.e., $\downarrow \circ \downarrow \subseteq \downarrow$.
(3) Every pair of convertible elements has a common reduct, i.e., $\leftrightarrow^* \subseteq \downarrow$.
(4) Every non-empty, finite, and connected subset of A has a common reduct.

Proof.
One easily shows the implication "(1) $\Rightarrow$ (2)". The proof of "(2) $\Rightarrow$ (3)" employs an obvious equality $\leftrightarrow^* = \downarrow^*$ and a property of closure operations $\downarrow^* = \downarrow$. In order to prove the implication "(3) $\Rightarrow$ (4)", let X be a non-empty, finite, and connected subset of A. The proof is by induction on the number of elements in X. The case $|X| = 1$ is trivial. Suppose $|X| \geq 2$. The proof for this case is illustrated in Figure 2.2. Since X is finite and connected, there exist $a, b \in X$ such that $X - \{a\}$ is connected and $a \leftrightarrow^* b$ [†1]. From the induction hypothesis $X - \{a\}$ has a common reduct c and thus we obtain $a \leftrightarrow^* c$. Thus the pair of elements a and c has a common reduct by assumption. This is also a common reduct of X. The implication "(4) $\Rightarrow$ (1)" is trivial. □

A proof of the equivalence of confluence and property (2), i.e., $\downarrow \circ \downarrow \subseteq \downarrow$, appeared in [Pla93]. The proof of the equivalence of confluence and property (3), i.e., $\leftrightarrow^* \subseteq \downarrow$, was first given by Rosen [Ros73].

[†1] Graph theory tells us the existence of such elements. Consider the graph whose vertices are the elements in X and edges correspond to the convertibility. Construct a spanning-tree and then select one of its leafs.



(A) ⋯ induction hypothesis
(B) ⋯ assumption $\leftrightarrow^* \subseteq \downarrow$

Figure 2.2: Proof of Lemma 2.2.7.

The relationship between various properties of ARSs introduced so far is summarized in Figure 2.3. In Lemma 2.2.8 we only give proofs of some important results.

Lemma 2.2.8 (relationships between properties of ARSs)
(1) Every confluent ARS has unique normal forms.
(2) Every weakly normalizing ARS with unique normal forms is confluent.
(3) Every strongly normalizing and locally confluent ARS is confluent.

Proof.

(1) Let $\to$ be a confluent ARS and suppose $b_1 \;{}^!\!\leftarrow a \to^! b_2$. We have to show the equality $b_1 = b_2$. Since $b_1 \;{}^*\!\leftarrow a \to^* b_2$, we obtain $b_1 \downarrow b_2$ by confluence. The desired equality follows from the fact that b_1 and b_2 are normal forms.
(2) Let $\to$ be a weakly normalizing ARS with unique normal forms and suppose $b_1 \;{}^*\!\leftarrow a \to^* b_2$. We have to prove that $b_1 \downarrow b_2$. By weak normalization, we infer the existence of normal forms c_1 and c_2 such that $b_1 \to^! c_1$ and $b_2 \to^! c_2$. Clearly, we have $c_1 \;{}^!\!\leftarrow a \to^! c_2$. Because the ARS $\to$ has unique normal forms, we obtain $c_1 = c_2$. Hence we have $b_1 \to^! c_1 = c_2 \;{}^!\!\leftarrow b_2$. Therefore $b_1 \downarrow b_2$.
(3) Let $\to$ be a strongly normalizing and local confluent ARS. By well-founded induction with respect to the relation $\to$, we prove that the ARS is confluent.

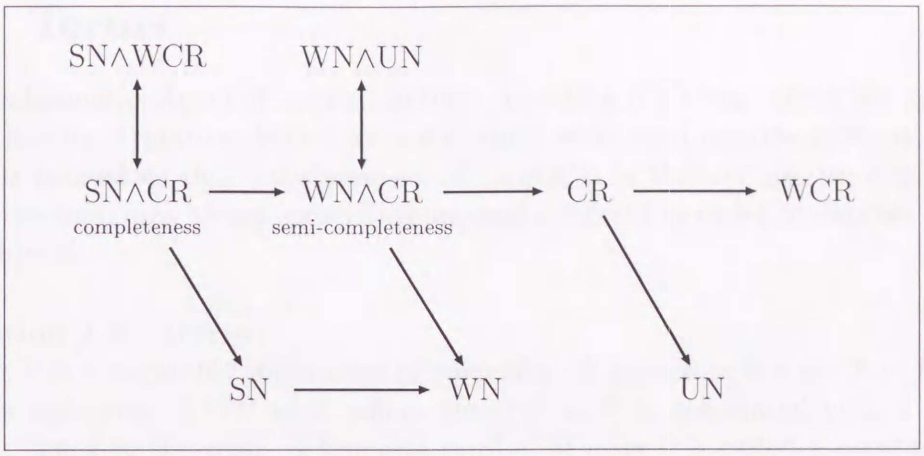


Figure 2.3: Relationships between properties of ARSs.

Suppose $b_1 \ {}^*\!\leftarrow a \to^* b_2$. If $a = b_1$ then $b_1 \to^* b_2$. If $a = b_2$ then $b_1 \ {}^*\!\leftarrow b_2$. Hence in both cases we obtain $b_1 \downarrow b_2$. Suppose $a \neq b_1$ and $a \neq b_2$. In this case we have $b_1 \ {}^*\!\leftarrow c_1 \leftarrow a \to c_2 \to^* b_2$ for some elements c_1 and c_2. From the local confluence assumption, there exists an element d that satisfies $c_1 \to^* d \ {}^*\!\leftarrow c_2$. Since $a \to c_1$ and $b_1 \ {}^*\!\leftarrow c_1 \to^* d$, the induction hypothesis yields the existence of an element e such that $b_1 \to^* e \ {}^*\!\leftarrow d$. Because we have $a \to c_2$ and $e \ {}^*\!\leftarrow c_2 \to^* b_2$, another application of the induction hypothesis yields that $e \downarrow b_2$. Therefore $b_1 \downarrow b_2$.

□

The proof of statement (3) in the above lemma was first given by Newman [New42]. The more simple proof shown above stems from [Hue80]. Another simple proof by Barendregt can be found in [Bar84].

Every element in semi-complete ARS reduces to a unique irreducible element because every semi-complete ARS is weakly normalizing and has unique normal forms. In the following definition, we introduce a notation to denote such an element.

Definition 2.2.9 (unique irreducible reduct)
Let $\to$ be a semi-complete ARS. The unique irreducible reduct of an element a is denoted by $a\!\downarrow$.

2.3 Terms

The fundamental object of interest in term rewriting is a term, which is a syntactic object having structure. When we are dealing with term rewrite systems, the set of terms is used as the underlying set of an ARS. In this section we define basic concepts concerning terms, substitutions, and contexts in order to express and use such objects.

Definition 2.3.1 (term)
The set $\mathcal{V}$ is a countably infinite set of *variables*. A *signature* is a set $\mathcal{F}$ of *function symbols* satisfying $\mathcal{F} \cap \mathcal{V} = \varnothing$ where every $f \in \mathcal{F}$ is associated with a natural number denoting its *arity*. A function symbol of arity 0 is called a *constant*. The set $\mathcal{T}(\mathcal{F}, \mathcal{V})$ of *terms* built from $\mathcal{F}$ and $\mathcal{V}$ is the smallest set such that

- $\mathcal{V} \subseteq \mathcal{T}(\mathcal{F}, \mathcal{V})$,
- If $f \in \mathcal{F}$ has arity n and $t_1, \ldots, t_n \in \mathcal{T}(\mathcal{F}, \mathcal{V})$, then $f(t_1, \ldots, t_n) \in \mathcal{T}(\mathcal{F}, \mathcal{V})$.

We write c instead of $c()$ for every constant c. We abbreviate $\mathcal{T}(\mathcal{F}, \mathcal{V})$ to $\mathcal{T}$ when no confusion can arise.

In order to enhance readability, infix notation is allowed. If $\star$ is a binary function symbol, the term $\star(s, t)$ is also denoted by $s \star t$.

Definition 2.3.2 (variables contained in a term)
Let s and t be terms. The set $\mathsf{Var}(t)$ of variables contained in t is inductively defined as follows:

$$\mathsf{Var}(t) = \begin{cases} \{t\} & \text{if} \quad t \in \mathcal{V}, \\ \bigcup_{i=1}^{n} \mathsf{Var}(t_i) & \text{if} \quad t = f(t_1, \ldots, t_n). \end{cases}$$

We also define $\mathsf{Var}(s, t) = \mathsf{Var}(s) \cup \mathsf{Var}(t)$.

Definition 2.3.3 (linear and ground term)
A term is called *linear* if it does not contain multiple occurrences of a variable. A term is called *ground* if it contains no variables.

Definition 2.3.4 (root symbol)
The *root symbol* of a term t is defined as follows:

$$\mathsf{root}(t) = \begin{cases} t & \text{if} \quad t \in \mathcal{V}, \\ f & \text{if} \quad t = f(t_1, \ldots, t_n). \end{cases}$$

Definition 2.3.5 (subterm)
Let s and t be terms. We say that s is a *subterm* of t and write $s \trianglelefteq t$ if either

$t = f(t_1, \ldots, t_n)$ and s is a subterm of t_i for some $i \in \{1, \ldots, n\}$ or $s = t$. We call s a *proper subterm* of t and write $s \lhd t$ if $s \unlhd t$ and $s \neq t$.

It is easy to verify that the inverse $\unrhd$ of the subterm relation is a well-founded partial order on the set of terms. In order to distinguish multiple occurrences of the same subterm, a subterm in a term is identified by a position.

Definition 2.3.6 (position)
A *position* is a sequence of positive integers. The empty sequence is denoted by ϵ and called the *root position*. Integers in a position are separated by a dot "$\cdot$". The set $\mathsf{Pos}(t)$ of positions in a term t is inductively defined as follows:

$$\mathsf{Pos}(t) = \begin{cases} \{\epsilon\} & \text{if} \quad t \in \mathcal{V}, \\ \{\epsilon\} \cup \{\, i \cdot p \mid 1 \leq i \leq n,\ p \in \mathsf{Pos}(t_i) \,\} & \text{if} \quad t = f(t_1, \ldots, t_n). \end{cases}$$

Positions are partially ordered by $\leq$ defined as follows: $p \leq q$ if there exists a position r such that $p \cdot r = q$. In that case $p \backslash q$ denotes the position r. We write $p < q$ if $p \leq q$ and $p \neq q$.

Let s and t be terms and let $p \in \mathsf{Pos}(t)$. The subterm $t_{|p}$ of t at position p is defined inductively:

$$t_{|p} = \begin{cases} t & \text{if} \quad p = \epsilon, \\ t_{i|q} & \text{if} \quad p = i \cdot q \quad \text{and} \quad t = f(t_1, \ldots, t_n). \end{cases}$$

The term $t[s]_p$ that is obtained from t by replacing the subterm at position p by s is inductively defined as follows:

$$t[s]_p = \begin{cases} u & \text{if} \quad p = \epsilon, \\ f(t_1, \ldots, t_i[s]_q, \ldots, t_n) & \text{if} \quad p = i \cdot q \quad \text{and} \quad t = f(t_1, \ldots, t_n). \end{cases}$$

The set $\mathsf{Pos}(t)$ is partitioned into *variable positions* $\mathsf{Pos}_{\mathcal{V}}(t)$ and *non-variable positions* $\mathsf{Pos}_{\mathcal{F}}(t)$ as follows: $\mathsf{Pos}_{\mathcal{V}}(t) = \{\, p \in \mathsf{Pos}(t) \mid t_{|p} \in \mathcal{V} \,\}$ and $\mathsf{Pos}_{\mathcal{F}}(t) = \mathsf{Pos}(t) \setminus \mathsf{Pos}_{\mathcal{V}}(t)$.

Observe that s is a subterm of t if and only if there exists a position p such that $s = t_{|p}$.

Definition 2.3.7 (substitution)
A *substitution* σ is a mapping from $\mathcal{V}$ to $\mathcal{T}(\mathcal{F}, \mathcal{V})$ such that its *domain*, defined as the set $\{\, x \in \mathcal{V} \mid \sigma(x) \neq x \,\}$, is finite. The set of all substitution is denoted by $\Sigma(\mathcal{F}, \mathcal{V})$ and abbreviated to Σ when the signature $\mathcal{F}$ is clear. A substitution σ is a *variable substitution* if $\sigma(x) \in \mathcal{V}$ for all $x \in \mathcal{V}$. A *variable renaming* is a bijective

variable substitution. A substitution σ is extended to the mapping $\overline{\sigma}$ from $\mathcal{T}(\mathcal{F},\mathcal{V})$ to $\mathcal{T}(\mathcal{F},\mathcal{V})$ as follows:

$$\overline{\sigma}(t) = \begin{cases} \sigma(t) & \text{if} \quad t \in \mathcal{V}, \\ f(\overline{\sigma}(t_1),\ldots,\overline{\sigma}(t_n)) & \text{if} \quad t = f(t_1,\ldots,t_n). \end{cases}$$

We write $t\sigma$ instead of $\overline{\sigma}(t)$ and we call $t\sigma$ an *instance* of t. A substitution σ can be represented by the set $\{\, x \mapsto \sigma(x) \mid \sigma(x) \neq x \,\}$. A substitution σ is *irreducible* if $\sigma(x)$ is irreducible for all $x \in \mathcal{V}$. The *composition* $\sigma\tau$ of two substitutions σ and τ is defined by $\sigma\tau(x) = (x\sigma)\tau$ for all $x \in \mathcal{V}$.

Definition 2.3.8 (context)
Let $\Box$ be a fresh constant called *hole*. A context is a term in $\mathcal{T}(\mathcal{F} \cup \{\Box\},\mathcal{V})$ containing one hole. If C is a context and t is a term, then $C[t]$ is the term which is obtained from C by replacing the holes with t. The set of all contexts is denoted by $\mathcal{C}(\mathcal{F},\mathcal{V})$ and abbreviated to $\mathcal{C}$ when the signature $\mathcal{F}$ is easily inferred.

Note that s is a subterm of t if and only if there exists a context C such that $C[s] = t$.

Definition 2.3.9 (relation closed under contexts and substitutions)
Let R be a relation on terms.

- R is *closed under contexts* if $t\,R\,u$ implies $C[t]\;R\;C[u]$ for all contexts C.
- R is *closed under substitutions* if $t\,R\,u$ implies $t\sigma\;R\;u\sigma$ for all substitutions σ.
- R is called a *rewrite relation* if it is closed under contexts and substitutions.

Closure under contexts and closure under substitutions admit closures, which are characterized in the following lemma.

Lemma 2.3.10 (closures on terms)
Let R be a relation on terms. The context closure $R^{\mathcal{C}}$ and the substitution closure R^{Σ} of R are the following relations:

$$R^{\mathcal{C}} = \{\, (C[s], C[t]) \mid s\,R\,t,\;\; C \in \mathcal{C}(\mathcal{F},\mathcal{V}) \,\},$$
$$R^{\Sigma} = \{\, (s\sigma, t\sigma) \mid s\,R\,t,\;\; \sigma \in \Sigma(\mathcal{F},\mathcal{V}) \,\}.$$

By using the closure operations introduced in this lemma, closure under contexts and closure under substitutions of a relation R are concisely expressed by $R^{\mathcal{C}} \subseteq R$ and $R^{\Sigma} \subseteq R$. So $R^{\mathcal{C}} \cup R^{\Sigma} \subseteq R$ is a shorthand for the fact that R is a rewrite relation.

Definition 2.3.11 (encompassment and unifiability)
Let s and t be terms.

- We say s *encompasses* t if there exist a context C and a substitution σ such that $s = C[t\sigma]$.
- We say s and t are *unifiable* if there exists a substitution σ such that $s\sigma = t\sigma$.

2.4 Conditional Equational Logic

Conditional equational logic (CEL) provides a way to formalize the principles of equational reasoning. In this section, we define the formal system of CEL by giving its syntax, proof-theoretical semantics, and algebraic semantics. For details, see for example [Wec92], [MT92] and [Pla93]. We begin the construction with the syntax. Logical assertions of CEL are expressed by conditional equations.

Definition 2.4.1 (conditional equation)
An *equation* is a pair (s, t) of terms. We write $s \approx t$ instead of (s, t). A *conditional equation* is a pair $(l \approx r, c)$ consisting of an equation $l \approx r$ and a finite sequence $c = s_1 \approx t_1, \ldots, s_n \approx t_n$ of equations. We write $l \approx r \Leftarrow c$ instead of $(l \approx r, c)$. If the *conditional part* c is empty, we simply write $l \approx r$ and call it an *unconditional equation*.

Definition 2.4.2 (conditional equational system)
A *conditional equational system* (CES) over a signature $\mathcal{F}$ is a set $\mathcal{E}$ of conditional equations over terms in $\mathcal{T}(\mathcal{F}, \mathcal{V})$. A CES $\mathcal{E}$ over $\mathcal{F}$ is also denoted by $(\mathcal{F}, \mathcal{E})$ to make the signature explicit. A CES consisting of only unconditional equations is called an *unconditional equational system* or simply an *equational system* (ES).

Example 2.4.3 (conditional equational system)
Let $\mathrm{x}, \mathrm{y} \in \mathcal{V}$ and $\mathcal{F} = \{\mathsf{0}, \mathsf{S}, >, \mathsf{T}, \mathsf{F}, \mathsf{max}\}$. Consider the following set of conditional equations over $\mathcal{T}(\mathcal{F}, \mathcal{V})$, specifying the maximum function on the set of natural numbers:

$$\mathcal{E} = \left\{ \begin{array}{rcll} \mathsf{0} > \mathrm{x} & \approx & \mathsf{F} & \\ \mathsf{S}(\mathrm{x}) > \mathsf{0} & \approx & \mathsf{T} & \\ \mathsf{S}(\mathrm{x}) > \mathsf{S}(\mathrm{y}) & \approx & \mathrm{x} > \mathrm{y} & \\ \mathsf{max}(\mathrm{x}, \mathrm{y}) & \approx & \mathrm{x} & \Leftarrow \ \mathrm{x} > \mathrm{y} \approx \mathsf{T} \\ \mathsf{max}(\mathrm{x}, \mathrm{y}) & \approx & \mathrm{y} & \Leftarrow \ \mathrm{x} > \mathrm{y} \approx \mathsf{F} \end{array} \right\}.$$

The set $\mathcal{E}$ is an example of a CES over the signature $\mathcal{F}$.

In order to give a formal justification for an inference in equational reasoning, we introduce a proof system for CEL. The inference rules are given in Table 2.4.

We can derive an (unconditional) equation from a given CES by applying the inference rules repeatedly.

reflexivity	$\dfrac{}{t \approx t}$ [r]	congruence	$\dfrac{s_1 \approx t_1, \ldots, s_n \approx t_n}{f(s_1, \ldots, s_n) \approx f(t_1, \ldots, t_n)}$ [c] if $f \in \mathcal{F}$ is n-ary
symmetry	$\dfrac{s \approx t}{t \approx s}$ [s]	application	$\dfrac{s_1\sigma \approx t_1\sigma, \ldots, s_n\sigma \approx t_n\sigma}{l\sigma \approx r\sigma}$ [a] if $l \approx r \Leftarrow s_1 \approx t_1, \ldots, s_n \approx t_n \in \mathcal{E}$
transitivity	$\dfrac{s \approx t,\ t \approx u}{s \approx u}$ [t]		

Figure 2.4: Inference rules of conditional equational logic.

These inference rules are suitable for a graphical presentation of a proof. For formal analysis based on the theory of relations, it is more convenient to define the derivability as a relation on the set of terms.

Definition 2.4.4 (provable equivalence)
Let $(\mathcal{F}, \mathcal{E})$ be a CES. We say the equation $s \approx t$ is *provable* from $\mathcal{E}$ if it can be deduced using the inference rules of Table 2.4. We write $s \approx_{\mathcal{E}} t$ in that case. The relation $\approx_{\mathcal{E}}$ is called *provable equivalence* induced by $\mathcal{E}$ and formally defined as follows:

$$
\begin{aligned}
\approx_{\mathcal{E}_0} &= \varnothing, \\
\approx_{\mathcal{E}_{k+1}} &= \{\, (t,t) \mid t \in \mathcal{T}(\mathcal{F}, \mathcal{V}) \,\} \\
&\cup \{\, (t,s) \mid s \approx_{\mathcal{E}_k} t \,\} \\
&\cup \{\, (s,u) \mid s \approx_{\mathcal{E}_k} t,\ t \approx_{\mathcal{E}_k} u \,\} \\
&\cup \{\, (f(s_1, \ldots, s_n), f(t_1, \ldots, t_n)) \mid \\
&\qquad f \in \mathcal{F} \text{ is } n\text{-ary},\ s_i \approx_{\mathcal{E}_k} t_i \text{ for } 1 \le i \le n \,\} \\
&\cup \{\, (l\sigma, r\sigma) \mid l \approx r \Leftarrow c \in \mathcal{E},\ \sigma \in \Sigma,\ c\sigma \subseteq \approx_{\mathcal{E}_k} \,\} \\
&\quad \text{for all } k \in \mathbb{N}, \\
\approx_{\mathcal{E}} &= \textstyle\bigcup_{k \in \mathbb{N}} \approx_{\mathcal{E}_k}.
\end{aligned}
$$

Here $c\sigma$ denotes the set of equations defined by $c\sigma = \{\, s\sigma \approx t\sigma \mid s \approx t \text{ in } c \,\}$. So $c\sigma \subseteq \approx_{\mathcal{E}_k}$ with $c = s_1 \approx t_1, \ldots, s_n \approx t_n$ is a shorthand for $s_1\sigma \approx_{\mathcal{E}_k} t_1\sigma, \ldots, s_n\sigma \approx_{\mathcal{E}_k} t_n\sigma$.

We have $s \approx_{\mathcal{E}} t$ if and only if $s \approx_{\mathcal{E}_k} t$ for some $k \in \mathbb{N}$. The minimum such k is called the *level* of $s \approx_{\mathcal{E}} t$.

Note that with the inference rules of Table 2.4 we can only deduce unconditional equations. The reader is referred to [Sel72] for inference rules for conditional consequences of CESs.

Definition 2.4.5 (equational theory)
The *equational theory* of a CES $\mathcal{E}$ is the set of all equations provable from $\mathcal{E}$.

Example 2.4.6 (proof-theoretical semantics of CES)
Consider the CES $\mathcal{E}$ defined in 2.4.3. The equation $\mathsf{max}(\mathsf{S}(\mathrm{x}), 0) \approx \mathsf{max}(0, \mathsf{S}(\mathrm{x}))$ is in the equational theory of $\mathcal{E}$, since it is provable from $\mathcal{E}$ as shown in the following deduction:

$$\dfrac{\dfrac{\dfrac{}{\mathsf{S}(\mathrm{x}) > 0 \approx \mathsf{T}}\,[\mathsf{a}]}{\mathsf{max}(\mathsf{S}(\mathrm{x}), 0) \approx \mathsf{S}(\mathrm{x})}\,[\mathsf{a}] \qquad \dfrac{\dfrac{\dfrac{}{0 > \mathsf{S}(\mathrm{x}) \approx \mathsf{F}}\,[\mathsf{a}]}{\mathsf{max}(0, \mathsf{S}(\mathrm{x})) \approx \mathsf{S}(\mathrm{x})}\,[\mathsf{a}]}{\mathsf{S}(\mathrm{x}) \approx \mathsf{max}(0, \mathsf{S}(\mathrm{x}))}\,[\mathsf{s}]}{\mathsf{max}(\mathsf{S}(\mathrm{x}), 0) \approx \mathsf{max}(0, \mathsf{S}(\mathrm{x}))}\,[\mathsf{t}]$$

The levels of $\mathsf{S}(\mathrm{x}) > 0 \approx_{\mathcal{E}} \mathsf{T}$, $\mathsf{max}(\mathsf{S}(\mathrm{x}), 0) \approx_{\mathcal{E}} \mathsf{S}(\mathrm{x})$, and $\mathsf{max}(\mathsf{S}(\mathrm{x}), 0) \approx_{\mathcal{E}} \mathsf{max}(0, \mathsf{S}(\mathrm{x}))$ are 1, 2, and 4 respectively. Observe that each of them expresses the smallest depth of a corresponding proof tree.

The semantics of equational logic is given by algebras, namely, we give meaning to syntactic constructs such as terms and (conditional) equations. $\mathcal{F}$-algebras and assignments are used to assign meaning to function symbols and variables respectively. If we fix an $\mathcal{F}$-algebra and an assignment, the interpretation of every term is uniquely determined as an element of the domain of discourse.

Definition 2.4.7 ($\mathcal{F}$-algebra)
Let $\mathcal{F}$ be a signature. An *$\mathcal{F}$-algebra* $\mathbf{A}$ is a set A, called the *carrier* of $\mathbf{A}$, together with operations $f_{\mathbf{A}} : A^n \to A$ for every n-ary function symbol $f \in \mathcal{F}$. The *term algebra* $\mathbf{T}(\mathcal{F}, \mathcal{V})$ is a special $\mathcal{F}$-algebra whose carrier is the set of terms $\mathcal{T}(\mathcal{F}, \mathcal{V})$ and its operations are defined by $f_{\mathbf{T}(\mathcal{F},\mathcal{V})}(t_1, \ldots, t_n) = f(t_1, \ldots, t_n)$ for every n-ary function symbol $f \in \mathcal{F}$. We abbreviate $\mathbf{T}(\mathcal{F}, \mathcal{V})$ to $\mathbf{T}$ and $\mathbf{T}(\mathcal{F}, \varnothing)$ to $\mathbf{T}(\mathcal{F})$ when no confusion can arise.

Definition 2.4.8 (assignment)
Let $\mathbf{A}$ be an $\mathcal{F}$-algebra having the carrier A. An *assignment* α on $\mathbf{A}$ is a mapping

from $\mathcal{V}$ to A. It is extended to the mapping $\overline{\alpha}$ from $\mathcal{T}(\mathcal{F}, \mathcal{V})$ to A as follows:

$$\overline{\alpha}(t) = \begin{cases} \alpha(t) & \text{if} \quad t \in \mathcal{V}, \\ f_{\boldsymbol{A}}(\overline{\alpha}(t_1), \ldots, \overline{\alpha}(t_n)) & \text{if} \quad t = f(t_1, \ldots, t_n). \end{cases}$$

The correctness of a (conditional) equation is determined by using algebra. If an equation is correct, we say it is valid. In that case, the interpretation that validates the equation is called a model. Formal definitions are given below. Notice that variables in conditional equations are (implicitly) universally quantified.

Definition 2.4.9 (valid equation)
An equation $s \approx t$ is *valid* in an $\mathcal{F}$-algebra $\boldsymbol{A}$ if $\overline{\alpha}(s) = \overline{\alpha}(t)$ for all assignments α on $\boldsymbol{A}$. Alternatively we say that $\boldsymbol{A}$ is a *model* of $s \approx t$. We write $s =_{\boldsymbol{A}} t$ if $s \approx t$ is valid in $\boldsymbol{A}$.

Definition 2.4.10 (valid conditional equation)
A conditional equation $l \approx r \Leftarrow c$ is *valid* in an $\mathcal{F}$-algebra $\boldsymbol{A}$ if $\overline{\alpha}(l) = \overline{\alpha}(r)$ for all assignments α on $\boldsymbol{A}$ satisfying $\overline{\alpha}(s) = \overline{\alpha}(t)$ for every equation $s \approx t$ in c. We also say that $\boldsymbol{A}$ is a *model* of $l \approx r \Leftarrow c$.

Definition 2.4.11 (model and semantic equivalence)
An $\mathcal{F}$-algebra $\boldsymbol{A}$ is a *model* of a CES $\mathcal{E}$ if every conditional equation $l \approx r \Leftarrow c \in \mathcal{E}$ is valid in $\boldsymbol{A}$. The *variety* $\mathsf{M}_{\mathcal{E}}$ of a CES $\mathcal{E}$ is the class of all models of $\mathcal{E}$. We write $s =_{\mathsf{M}_{\mathcal{E}}} t$ if the equation $s \approx t$ is valid in every model of $\mathcal{E}$. The relation $=_{\mathsf{M}_{\mathcal{E}}}$ is called *semantic equivalence* induced by $\mathcal{E}$.

Example 2.4.12 (algebraic semantics of CES)
Consider again the CES $\mathcal{E}$ defined in 2.4.3. We define an $\mathcal{F}$-algebra $\boldsymbol{A}$ as follows. The carrier of $\boldsymbol{A}$ is the set $\mathbb{N}$ of natural numbers. Operations are defined by:

$$\begin{array}{lcl} \mathsf{0}_{\boldsymbol{A}} & = & 0, \\ \mathsf{S}_{\boldsymbol{A}}(n) & = & n+1, \\ \mathsf{F}_{\boldsymbol{A}} & = & 0, \\ \mathsf{T}_{\boldsymbol{A}} & = & 1, \\ m >_{\boldsymbol{A}} n & = & \begin{cases} 1 & \text{if} \quad m > n, \\ 0 & \text{if} \quad m \leq n, \end{cases} \\ \mathsf{max}_{\boldsymbol{A}}(m, n) & = & \max(m, n), \end{array}$$

for all $m, n \in \mathbb{N}$. The equation $\mathsf{max}(\mathsf{S}(\mathrm{x}), \mathsf{0}) \approx \mathsf{max}(\mathsf{0}, \mathsf{S}(\mathrm{x}))$ is valid in $\boldsymbol{A}$, since $\max(n+1, 0) = \max(0, n+1)$ for all $n \in \mathbb{N}$. The equation $\mathrm{x} > \mathsf{0} \approx \mathsf{T}$ is not valid in $\boldsymbol{A}$, because $\overline{\alpha}(\mathrm{x} > \mathsf{0}) \neq \mathsf{T}_{\boldsymbol{A}}$ when $\alpha(\mathrm{x}) = 0$.

Birkhoff's soundness and completeness theorem [Bir35] states that provable equivalence coincides with semantic equivalence for all (unconditional) ESs. The following theorem extends it to arbitrary CESs. A proof of the theorem is given in Chapter 3.

Theorem 2.4.13 (soundness and completeness of CEL)
Let $\mathcal{E}$ be a CES. Provable equivalence induced by $\mathcal{E}$ coincides with semantic equivalence induced by $\mathcal{E}$, i.e., $\approx_{\mathcal{E}} \; = \; =_{\mathsf{M}_{\mathcal{E}}}$.

2.5 Conditional Term Rewrite Systems

In the previous section, we introduced CES and defined its syntax, algebraic semantics and proof-theoretical semantics. The purpose of this section is to give the operational semantics of CES by conditional term rewrite systems and provide a basis for studying equational reasoning from the computational view point. For extensive surveys on term rewriting, we refer to [HO80], [DJ90], and [Klo92].

Definition 2.5.1 (conditional term rewrite system)
A *conditional rewrite rule* is a conditional equation which is denoted by $l \to r \Leftarrow c$.

A CES consisting of conditional rewrite rules is called a *conditional term rewrite system* (CTRS). A CTRS consisting of only unconditional equations is called an *unconditional term rewrite system* or simply a *term rewrite system* (TRS).

As opposed to previous works (e.g. [Kap84], [DO90]) in which the underlying CES is restricted, we define rewrite systems as general as possible. This enables a good correspondence between conditional rewriting and conditional equational logic.

Conditional rewrite rules are used to rewrite terms by replacing an instance of the left-hand side l with the corresponding instance of the right-hand side r provided the corresponding instance of the conditional part c is satisfied. To express this directed use of conditional equations we denote conditional rewrite rules by $l \to r \Leftarrow c$.

Definition 2.5.2 (rewrite relation of a CTRS)
The rewrite relation $\to_{\mathcal{R}}$ of a CTRS $\mathcal{R}$ is defined as follows:

$$\begin{array}{lcl} \to_{\mathcal{R}_0} & = & \varnothing, \\ \to_{\mathcal{R}_{k+1}} & = & \{\, (C[l\sigma], C[r\sigma]) \mid l \to r \Leftarrow c \in \mathcal{R},\ C \in \mathcal{C},\ \sigma \in \Sigma,\ \mathcal{R}_k \vdash c\sigma \,\} \\ & & \text{for all } k \in \mathbb{N}, \\ \to_{\mathcal{R}} & = & \bigcup_{k \in \mathbb{N}} \to_{\mathcal{R}_k}. \end{array}$$

So we have $s \to_{\mathcal{R}} t$ if and only if $s \to_{\mathcal{R}_k} t$ for some $k \in \mathbb{N}$. The minimum such k is called the *level* of the rewrite step $s \to_{\mathcal{R}} t$. The subterm $l\sigma$ in the definition of $\to_{\mathcal{R}_{k+1}}$ is called a *redex*. We write $s \xrightarrow{p}_{\mathcal{R}} t$ if we want to make the position p of the redex in s explicit. Note that the meaning of $\mathcal{R}_k \vdash c\sigma$, which determines the satisfiability of conditions, is not fixed in this definition.

Different rewrite relations can be associated with a given CTRS depending on the interpretation of conditions. The most common interpretations are checking conditions by convertibility ($\leftrightarrow^*$), joinability ($\downarrow$), and reduction ($\to^*$). For other interpretations, the reader is referred to [DOS88a].

Definition 2.5.3 (types of CTRSs)
In *semi-equational* CTRSs satisfiability of conditions is defined as convertibility:

$$\mathcal{R}_k \vdash c\sigma \quad \Longleftrightarrow \quad c\sigma \subseteq \leftrightarrow^*_{\mathcal{R}_k}.$$

In *join-equational* CTRSs conditions are interpreted as convertibility:

$$\mathcal{R}_k \vdash c\sigma \quad \Longleftrightarrow \quad c\sigma \subseteq \downarrow_{\mathcal{R}_k}.$$

In *oriented* CTRSs satisfiability is checked by reduction:

$$\mathcal{R}_k \vdash c\sigma \quad \Longleftrightarrow \quad c\sigma \subseteq \to^*_{\mathcal{R}_k}.$$

Similarly to the definition of provable equivalence induced by a CES (Definition 2.4.4), $c\sigma$ denotes the set $\{\, s\sigma \approx t\sigma \mid s \approx t \text{ in } c \,\}$. So, for example, $c\sigma \subseteq \leftrightarrow^*_{\mathcal{R}_k}$ with $c = s_1 \approx t_1, \ldots, s_n \approx t_n$ is a shorthand for $s_1\sigma \leftrightarrow^*_{\mathcal{R}_k} t_1\sigma, \ldots, s_n\sigma \leftrightarrow^*_{\mathcal{R}_k} t_n\sigma$.

The classification of CTRSs by interpretation of the conditional part goes back to Bergstra and Klop [BK86] who use the terminology type I, II, and III for semi-equational, join, and oriented systems respectively. Semi-equational CTRSs are also called natural in the literature and join CTRSs are sometimes called standard. In this thesis we follow the terminology used in [DOS88a].

In the following chapters we frequently compare different types of CTRSs associated with the same CES. Hence it is convenient to make the explicit notational convention of writing $\mathcal{R}^{\mathrm{s}}$ ($\mathcal{R}^{\mathrm{j}}$, $\mathcal{R}^{\mathrm{o}}$) if $\mathcal{R}$ is considered as a semi-equational, (join, oriented) CTRS. Furthermore we abbreviate $\to_{\mathcal{R}^s}$ to $\to_s$ ($\downarrow_{\mathcal{R}^o}$ to $\downarrow_o$, $\leftrightarrow^*_{\mathcal{R}^j}$ to $\leftrightarrow^*_j$, etc.). We write $\mathcal{R}$ and $\to_{\mathcal{R}}$ if something applies to all three kinds of CTRSs (e.g., when defining properties of CTRSs).

Example 2.5.4 (conditional rewriting)
Consider again the CES $\mathcal{E}$ defined in 2.4.3 and let $\mathcal{R} = \mathcal{E}$. We have

$$\begin{array}{rcl} 0 > \mathsf{S}(0) & \to_{\mathcal{R}} & \mathsf{F}, \\ \mathsf{max}(0, \mathsf{S}(0)) & \to_{\mathcal{R}} & \mathsf{S}(0), \\ \mathsf{max}(\mathsf{max}(0, \mathsf{S}(0)), 0) & \to_{\mathcal{R}} & \mathsf{max}(\mathsf{S}(0), 0). \end{array}$$

The level of each rewrite step is 1, 2, and 3 respectively.

We don't put any restrictions on the distribution of variables among the different parts of conditional rewrite rules. In particular, we allow so-called extra variables in the right-hand sides as well as in the conditions of conditional rewrite rules.

Definition 2.5.5 (extra variable)
An *extra variable* in a conditional rewrite rule $l \to r \Leftarrow c$ is a variable in $\mathsf{Var}(r, c)$ which does not occur in l.

CTRSs with extra variables naturally appear in the programming applications.

Example 2.5.6 (CTRS with extra variable [DOS88a])
Consider the following CTRS, which specifies an efficient algorithm for computing the Fibonacci numbers:

$$\mathcal{R} = \left\{ \begin{array}{rcll} \mathsf{fib}(0) & \to & \langle 0,\ \mathsf{S}(0) \rangle & \\ \mathsf{fib}(\mathsf{S}(\mathrm{x})) & \to & \langle \mathrm{z},\ \mathrm{y} + \mathrm{z} \rangle & \Leftarrow \quad \mathsf{fib}(\mathrm{x}) \approx \langle \mathrm{z},\ \mathrm{y} \rangle \end{array} \right\}.$$

In this CTRS z is an extra variable because it does not appear in the left-hand side of the conditional rewrite rule.

Following [MH94], we introduce a classification of CTRSs by the distribution of extra variables in conditional rewrite rules.

Definition 2.5.7 (distribution of extra variables)
Let $\mathcal{R}$ be a CTRS. Every rewrite rule $l \to r \Leftarrow c \in \mathcal{R}$ is classified according to the

distribution of variables among l, r, and c as follows:

type	requirement
1	$\mathsf{Var}(r) \cup \mathsf{Var}(c) \subset \mathsf{Var}(l)$
2	$\mathsf{Var}(r) \subseteq \mathsf{Var}(l)$
3	$\mathsf{Var}(r) \subseteq \mathsf{Var}(l) \cup \mathsf{Var}(c)$
4	(no restrictions)

An *n-CTRS* contains only rules of type n. So a 1-CTRS contains no extra variables, a 2-CTRS may contain extra variables only in the conditions, a 3-CTRS may have extra variables in the right-hand sides provided they appear in the corresponding conditional part. Every CTRS is a 4-CTRS.

The ARS $(\mathcal{T}(\mathcal{F}, \mathcal{V}), \to_{\mathcal{R}})$ can be associated with a CTRS $(\mathcal{F}, \mathcal{R})$. So all the definitions and the properties of ARSs explained in Section 2.2 carry over to CTRSs. In the next chapter, we will see that the relation $\to_{\mathcal{R}}$ defined in Definition 2.5.2 coincides with the minimum solution satisfying the following implication.

$$l \to r \Leftarrow c \in \mathcal{R},\ C \in \mathcal{C},\ \sigma \in \Sigma,\ \mathcal{R} \vdash c\sigma \implies C[l\sigma] \to_{\mathcal{R}} C[r\sigma] \qquad (2.1)$$

Note that the above implication may have no minimum solution if we define the satisfiability of the conditions $\mathcal{R} \vdash c\sigma$ differently, as shown in the following example.

Example 2.5.8 (ill-defined rewrite relation [Klo90])
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{lll} \mathsf{a} \to \mathsf{a} & \Leftarrow & \mathsf{b} \approx \mathsf{b} \\ \mathsf{b} \to \mathsf{b} & \Leftarrow & \mathsf{a} \approx \mathsf{a} \end{array} \right\}.$$

and interpret the satisfiability of conditions as reduction to a normal from:

$$\mathcal{R} \vdash c\sigma \iff c\sigma \subseteq \to^{!}_{\mathcal{R}}.$$

Then implication (2.1) is satisfied by two different minimal solutions $\to_{\mathcal{R}} = \{(\mathsf{a}, \mathsf{a})\}$ and $\to_{\mathcal{R}} = \{(\mathsf{b}, \mathsf{b})\}$. Hence, in general it is not possible to characterize the rewrite relation by the minimum solution to implication (2.1).

Chapter 3

Soundness and Completeness

An ARS over terms can be used to define a provability relation associated with a given CES. One of the most important properties of a provability relation is soundness and completeness, namely, the property that all provable (by means of the ARS) equations are correct (with respect to the CES) and all correct equations are provable. Hence the provability relation can be interchanged with semantic equivalence in sound and complete ARSs.

This chapter consists of two parts. First, we provide tools which make the task of proving soundness and completeness easier. Based on the results obtained in the first part, we prove that both CEL and CTRSs are sound and complete. In Section 3.1 our attention is directed to the semantic aspect of soundness and completeness. We provide a sufficient condition for soundness and completeness which can be applied to any ARS and give a syntactic characterization of semantic equivalence. In Section 3.2 we review the fixpoint theorem and present a method to analyze properties of a relation which depends on a lattice theoretic property. Using the results obtained in these two sections, we give a proof of soundness and completeness of CEL in Section 3.3 and a proof of the soundness and completeness of semi-equational CTRS in Section 3.4.

3.1 Characterizing Semantic Equivalence

Consider an ARS which defines a provability relation on terms associated with a CES. The purpose of this section is to investigate how to ensure the soundness and completeness of the ARS and to give a syntactic characterization of semantic equivalence. We begin the investigation with a formal definition of the soundness and completeness of an ARS.

Definition 3.1.1 (soundness and completeness of an ARS)
Let $(\mathcal{F}, \mathcal{E})$ be a CES and $(\mathcal{T}(\mathcal{F}, \mathcal{V}), \sim)$ an ARS over the set of terms. We say the ARS $\sim$ is *sound* with respect to the CES $\mathcal{E}$ if $\sim \subseteq =_{\mathsf{M}_\mathcal{E}}$ and *complete* with respect to $\mathcal{E}$ if $=_{\mathsf{M}_\mathcal{E}} \subseteq \sim$.

Congruence and applicability are important notions concerning relations on terms in characterizing the semantic equivalence $=_{\mathsf{M}_\mathcal{E}}$.

Definition 3.1.2 (congruence)
(1) A relation $\sim$ on $\mathcal{T}(\mathcal{F}, \mathcal{V})$ is *compatible* if $f(s_1, \ldots, s_n) \sim f(t_1, \ldots, t_n)$ for all function symbols $f \in \mathcal{F}$ having arity n and for all terms $s_1, \ldots, s_n, t_1, \ldots, t_n$ that satisfy $s_1 \sim t_1, \ldots, s_n \sim t_n$.
(2) A compatible equivalence relation on the set of terms is called a *congruence*.

Definition 3.1.3 (applicable relation)
Let $(\mathcal{F}, \mathcal{E})$ be a CES and $\sim$ a relation on $\mathcal{T}(\mathcal{F}, \mathcal{V})$. The relation $\sim$ is *applicable* to $\mathcal{E}$ if $l\sigma \sim r\sigma$ for all conditional equations $l \approx r \Leftarrow c \in \mathcal{E}$ and substitutions σ such that $c\sigma \subseteq \sim$.

The following basic properties concerning semantics will be used in the proof of Theorem 3.1.6, which provides a sufficient condition for the soundness of an ARS.

Lemma 3.1.4 (substitution transfer)
$\overline{\alpha}(t\sigma) = \overline{\overline{\alpha} \circ \sigma}(t)$ for all $\mathcal{F}$-algebras $\boldsymbol{A}$, assignments α on $\boldsymbol{A}$, substitutions σ, and terms t.

Proof.
By structural induction on t. □

Lemma 3.1.5 (basic property of semantic equivalences)
Let $\boldsymbol{A}$ be an $\mathcal{F}$-algebra and $\mathcal{E}$ a CES.
(1) $=_{\boldsymbol{A}}$ and $=_{\mathsf{M}_\mathcal{E}}$ are congruences that are closed under substitutions.
(2) $=_{\mathsf{M}_\mathcal{E}}$ is applicable to $\mathcal{E}$.

Proof.
The proof of property (1) is straightforward. Substitution transfer (Lemma 3.1.4) is useful to show that $=_{\boldsymbol{A}}$ is closed under substitutions. In order to prove property (2), suppose $l \approx r \Leftarrow c \in \mathcal{E}$ and $c\sigma \subseteq =_{\mathsf{M}_\mathcal{E}}$. We have to show that $l\sigma =_{\mathsf{M}_\mathcal{E}} r\sigma$. Let $\boldsymbol{A}$ be an arbitrary model of $\mathcal{E}$ and α an arbitrary assignment on $\boldsymbol{A}$. By applying substitution transfer to $c\sigma \subseteq =_{\mathsf{M}_\mathcal{E}}$ we know that $\overline{\overline{\alpha} \circ \sigma}(s) = \overline{\overline{\alpha} \circ \sigma}(t)$ for all $s \approx t$ in c. Since $\boldsymbol{A} \in \mathsf{M}_\mathcal{E}$, we obtain $\overline{\overline{\alpha} \circ \sigma}(l) = \overline{\overline{\alpha} \circ \sigma}(r)$. Substitution transfer yields

$\overline{\alpha}(l\sigma) = \overline{\alpha}(r\sigma)$. Therefore $l\sigma =_{\mathsf{M}_\mathcal{E}} r\sigma$. □

Theorem 3.1.6 (soundness of an ARS)
Let $\mathcal{E}$ be a CES and $\sim$ an ARS over terms. If $\sim \subseteq \equiv$ for every congruence $\equiv$ that is applicable to $\mathcal{E}$, then $\sim$ is sound with respect to $\mathcal{E}$.

Proof.
According to Lemma 3.1.5, $=_{\mathsf{M}_\mathcal{E}}$ is a congruence which is applicable to $\mathcal{E}$. Hence, we have $\sim \subseteq =_{\mathsf{M}_\mathcal{E}}$ by assumption. □

Next we investigate how to ensure the completeness of an ARS. The proof of the completeness theorem (Theorem 3.1.11) becomes easier by considering validity in the quotient term algebra.

Definition 3.1.7 (quotient term algebra)
Let $\equiv$ be a congruence. The carrier of the *quotient term algebra* $\mathbf{Q}_\equiv$ for $\equiv$ is the quotient set $\mathrm{Q}_\equiv$ given by $\mathrm{Q}_\equiv = \mathcal{T}(\mathcal{F}, \mathcal{V}) \,/ \equiv$. The operations of $\mathbf{Q}_\equiv$ are defined by $f_{\mathbf{Q}_\equiv}([t_1]_\equiv, \ldots, [t_n]_\equiv) = [f(t_1, \ldots, t_n)]_\equiv$ for every n-ary function symbols $f \in \mathcal{F}$. The *natural mapping* $\nu_\equiv$ for $\equiv$ is the assignment on $\mathbf{Q}_\equiv$ defined by $\nu_\equiv(x) = [x]_\equiv$.

Note that it is possible to construct the quotient set $\mathrm{Q}_\equiv$ because a congruence is an equivalence relation. Compatibility of the congruence $\equiv$ guarantees the well-definedness of the operations of $\mathbf{Q}_\equiv$: Suppose $s_1 \equiv t_1, \ldots, s_n \equiv t_n$. The compatibility of $\equiv$ yields $f(s_1, \ldots, s_n) \equiv f(t_1, \ldots, t_n)$. Hence we have $[f(s_1, \ldots, s_n)]_\equiv = [f(t_1, \ldots, t_n)]_\equiv$.

The natural mapping is used in the proof of the completeness theorem to decompose an assignment on the quotient term algebra into two mappings.

Lemma 3.1.8 (property of the natural mapping)
Let $\equiv$ be a congruence and s, t terms.
(1) $\overline{\nu_\equiv}(t) = [t]_\equiv$.
(2) $\overline{\nu_\equiv}(s) = \overline{\nu_\equiv}(t)$, $[s]_\equiv = [t]_\equiv$, and $s \equiv t$ are equivalent.

Proof.
Property (1) is proved by structural induction on t. Property (2) is an easy consequence of (1). □

Lemma 3.1.9 (decomposition of an assignment on $\mathbf{Q}_\equiv$)
Let $\equiv$ be a congruence and t a term. If a substitution σ and an assignment α on $\mathbf{Q}_\equiv$ satisfy $\sigma(x) \in \alpha(x)$ for all $x \in \mathsf{Var}(t)$, then $\overline{\alpha}(t) = \overline{\nu_\equiv}(t\sigma)$.

Proof.

By structural induction on t. Lemma 3.1.8 can be employed for the base case $t \in \mathcal{V}$. □

The next lemma provides the most important result for completeness, stating that the quotient term algebra is a model of a given CES.

Lemma 3.1.10 (property of a quotient term algebra)
Let $\mathcal{E}$ be a CES and $\equiv$ a congruence that is applicable to $\mathcal{E}$. The quotient term algebra for $\equiv$ is a model of $\mathcal{E}$, i.e., $\mathbf{Q}_{\equiv} \in \mathsf{M}_{\mathcal{E}}$.

Proof.
Let $l \approx r \Leftarrow c \in \mathcal{E}$ and α an assignment on $\mathbf{Q}_{\equiv}$. Assume $\overline{\alpha}(s) = \overline{\alpha}(t)$ for all $s \approx t$ in c. We have to show that $\overline{\alpha}(l) = \overline{\alpha}(r)$. In order to decompose the assignment α, let σ be any substitution with the following two properties:

- $\sigma(x) \in \alpha(x)$ if $x \in \mathsf{Var}(l, r) \cup \mathsf{Var}(c)$,
- $\sigma(x) = x$ otherwise.

We decompose α (Lemma 3.1.9) and use the property of the natural mapping (Lemma 3.1.8(1)). Now the statement we have to prove is: $[l\sigma]_{\equiv} = [r\sigma]_{\equiv}$. The assumption becomes: $[s\sigma]_{\equiv} = [t\sigma]_{\equiv}$ for all $s \approx t$ in c. We obtain $l\sigma \equiv r\sigma$ from this assumption using the property of the natural mapping (Lemma 3.1.8(2)) and the applicability of $\equiv$ to $\mathcal{E}$. Therefore $[l\sigma]_{\equiv} = [r\sigma]_{\equiv}$. □

Now we are ready to give a sufficient condition for the completeness of an ARS.

Theorem 3.1.11 (completeness of an ARS)
Let $\mathcal{E}$ be a CES and $\sim$ an ARS over terms. If $\sim$ is a congruence that is applicable to $\mathcal{E}$, then $=_{\mathsf{M}_{\mathcal{E}}} \subseteq =_{\mathbf{Q}_{\sim}} \subseteq \sim$ and hence $\sim$ is complete with respect to $\mathcal{E}$.

Proof.

$$
\begin{array}{rlll}
s =_{\mathsf{M}_{\mathcal{E}}} t & \Leftrightarrow & \forall \mathbf{A} \in \mathsf{M}_{\mathcal{E}}\ s =_{\mathbf{A}} t & (\text{definition of } =_{\mathsf{M}_{\mathcal{E}}}) \\
& \Rightarrow & s =_{\mathbf{Q}_{\sim}} t & (\mathbf{Q}_{\sim} \in \mathsf{M}_{\mathcal{E}} \text{ by Lemma 3.1.10}) \\
& \Leftrightarrow & \forall \alpha : \mathcal{V} \to \mathrm{Q}_{\sim}\ \overline{\alpha}(s) = \overline{\alpha}(t) & (\text{definition of } =_{\mathbf{Q}_{\sim}}) \\
& \Rightarrow & \overline{\nu_{\sim}}(s) = \overline{\nu_{\sim}}(t) & (\nu_{\sim} : \mathcal{V} \to \mathrm{Q}_{\sim}) \\
& \Leftrightarrow & s \sim t & (\text{Lemma 3.1.8(2)})
\end{array}
$$

□

Combining the two theorems, we can give a sufficient condition for ensuring the soundness and completeness of an ARS.

Corollary 3.1.12 (soundness and completeness of an ARS)
Let $\mathcal{E}$ be a CES and $\sim$ an ARS. If $\sim$ is the smallest congruence that is applicable to $\mathcal{E}$, then $\sim$ is sound and complete with respect to $\mathcal{E}$, i.e., $\sim \, = \, =_{\mathsf{M}_{\mathcal{E}}}$.

An easy consequence of this theorem is the following syntactic characterization of the semantic equivalence $=_{\mathsf{M}_{\mathcal{E}}}$.

Corollary 3.1.13 (characterization of semantic equivalence)
Let $\mathcal{E}$ be a CES. The semantic equivalence $=_{\mathsf{M}_{\mathcal{E}}}$ is the smallest congruence that is applicable to $\mathcal{E}$.

3.2 Fixpoint Theorem for Relations

In the previous section we abstracted the semantic part of the soundness and completeness of an ARS. Hence it remains to analyze syntactic properties of relations. In this section we review the fixpoint theorem, which provides a tool to analyze properties of an iteratively constructed object. The results of this section can be applied to characterize both the provable equivalence of a CES and the rewrite relation of a CTRS because they are defined by an iterative construction. For a detailed description on the fixpoint theorem, see, for example, [DP90] and [Wec92].

Since our aim is to analyze properties of relations, we specialize the domain of the discussion to the set of relations.

Definition 3.2.1 (complete set of relations)
We call a set $\mathcal{C}$ of relations *complete* if $\mathcal{C}$ is closed under unions and intersections, i.e., $\bigcup \mathcal{X} \in \mathcal{C}$ and $\bigcap \mathcal{X} \in \mathcal{C}$ for all subsets $\mathcal{X} \subseteq \mathcal{C}$.

Example 3.2.2 (complete set of relations)
Let A be an arbitrary set. The set $\mathsf{Pow}(A \times A)$ of all relations on A is complete. Given a relation $R_0 \subseteq A \times A$, the set $\{\, R \subseteq A \times A \mid R_0 \subseteq R \,\}$ of all relations on A containing R_0 is also complete.

Observe that $(\mathcal{C}, \subseteq)$ is a complete lattice for every complete set $\mathcal{C}$ of relations. Now we construct a sequence of relations by iterative applications of a function and define a relation as the least upper bound of this sequence.

Definition 3.2.3 (iterative construction of a relation)
Let $\mathcal{C}$ be a complete set of relations and $\Phi : \mathcal{C} \to \mathcal{C}$ a function. The relation $\to_\Phi$ is defined as follows:

$$\begin{array}{lcll} \to_{\Phi_0} & = & \bot_{\mathcal{C}}, & \\ \to_{\Phi_{k+1}} & = & \Phi(\to_{\Phi_k}) & \text{for all } k \in \mathbb{N}, \\ \to_{\Phi} & = & \bigcup_{k\in\mathbb{N}} \to_{\Phi_k} \; = \; \bigcup_{k\in\mathbb{N}} \Phi^k(\bot_{\mathcal{C}}). & \end{array}$$

Here $\bot_{\mathcal{C}}$ is the smallest relation in $\mathcal{C}$.

Monotonicity and continuity are two important lattice theoretic properties which the function for iterative construction should satisfy.

Definition 3.2.4 (monotonicity)
Let $\mathcal{C}$ be a complete set of relations. A function $\Phi : \mathcal{C} \to \mathcal{C}$ is *monotone* if $R \subseteq S$ implies $\Phi(R) \subseteq \Phi(S)$ for all relations $R, S \in \mathcal{C}$.

Definition 3.2.5 (chain)
A sequence $(R_n \mid n \in \mathbb{N})$ of relations is called a *chain* if $R_i \subseteq R_j$ for all $i, j \in \mathbb{N}$ such that $i \leq j$, or equivalently, $R_i \subseteq R_{i+1}$ for all $i \in \mathbb{N}$.

Lemma 3.2.6 (chain by monotone function)
Let $\mathcal{C}$ be a complete set of relations. If $\Phi : \mathcal{C} \to \mathcal{C}$ is a monotone function, then the sequence $(\to_{\Phi_k} \mid k \in \mathbb{N})$ of iteratively constructed relations is a chain.

Proof.
To show that $(\to_{\Phi_k} \mid k \in \mathbb{N})$ is a chain, we prove $\to_{\Phi_k} \subseteq \to_{\Phi_{k+1}}$ for all $k \in \mathbb{N}$ by induction on k. If $k = 0$ then we have $\to_{\Phi_k} = \bot_{\mathcal{C}} \subseteq \to_{\Phi_{k+1}}$. Suppose $k \geq 1$. Using the induction hypothesis and the monotonicity of Φ, we obtain $\Phi(\to_{\Phi_{k-1}}) \subseteq \Phi(\to_{\Phi_k})$. Therefore $\to_{\Phi_k} \subseteq \to_{\Phi_{k+1}}$ by definition. □

Definition 3.2.7 (continuity)
Let $\mathcal{C}$ be a complete set of relations. A function $\Phi : \mathcal{C} \to \mathcal{C}$ is *continuous* if $\Phi(\bigcup_{i\in\mathbb{N}} R_i) = \bigcup_{i\in\mathbb{N}} \Phi(R_i)$ for every chain $(R_i \mid i \in \mathbb{N})$ on $\mathcal{C}$.

The next lemma is useful for proving the continuity of a monotone function.

Lemma 3.2.8 (relationship between monotonicity and continuity)
Let $\mathcal{C}$ be a complete set of relations and $\Phi : \mathcal{C} \to \mathcal{C}$ a function.
(1) Φ is monotone if and only if $\bigcup_{i\in\mathbb{N}} \Phi(R_i) \subseteq \Phi(\bigcup_{i\in\mathbb{N}} R_i)$ for every chain $(R_i \mid i \in \mathbb{N})$ on $\mathcal{C}$.
(2) Φ is continuous if and only if Φ is monotone and $\Phi(\bigcup_{i\in\mathbb{N}} R_i) \subseteq \bigcup_{i\in\mathbb{N}} \Phi(R_i)$ for every chain $(R_i \mid i \in \mathbb{N})$ on $\mathcal{C}$.

Proof.

(1) For the proof of the "⇒" direction, suppose Φ is monotone and consider an arbitrary chain $(R_i \mid i \in \mathbb{N})$ on $\mathcal{C}$. Clearly the inclusion $R_j \subseteq \bigcup_{i\in\mathbb{N}} R_i$ holds for all $j \in \mathbb{N}$. By the monotonicity of Φ we have $\Phi(R_j) \subseteq \Phi(\bigcup_{i\in\mathbb{N}} R_i)$ for all $j \in \mathbb{N}$. From this inclusion and the fact that $\bigcup_{i\in\mathbb{N}} \Phi(R_i)$ is the least upper bound of the set $\{\, \Phi(R_i) \mid i \in \mathbb{N} \,\}$, the desired inclusion $\bigcup_{n\in\mathbb{N}} \Phi(R_n) \subseteq \Phi(\bigcup_{n\in\mathbb{N}} R_n)$ is

derived. For the "$\Leftarrow$" direction, suppose $R \subseteq S$ with $R, S \in \mathcal{C}$. We have to show that $\Phi(R) \subseteq \Phi(S)$. Consider the chain $(R_i \mid i \in \mathbb{N})$ such that $R_0 = R$ and $R_i = S$ for all $i \geq 1$. By assumption $\bigcup_{i \in \mathbb{N}} \Phi(R_i) \subseteq \Phi(\bigcup_{i \in \mathbb{N}} R_i)$. From the construction of the sequence $(R_i \mid i \in \mathbb{N})$ we infer that $\bigcup_{i \in \mathbb{N}} \Phi(R_i) = \Phi(R) \cup \Phi(S)$ and $\Phi(\bigcup_{i \in \mathbb{N}} R_i) = \Phi(S)$. Therefore $\Phi(R) \cup \Phi(S) = \Phi(S)$ and hence $\Phi(R) \subseteq \Phi(S)$.

(2) By property (1) and the definition of continuity.

□

The following results are useful when the function for iterative construction is defined by a combination of several functions.

Lemma 3.2.9 (monotonicity of combined functions)
Let $\mathcal{C}$ be a complete set of relations and suppose two functions $\Phi_1, \Phi_2 : \mathcal{C} \to \mathcal{C}$ are monotone.

(1) The function Φ defined by $\Phi(R) = \Phi_1(R) \cup \Phi_2(R)$ is monotone.
(2) The function Φ defined by $\Phi(R) = \Phi_2(\Phi_1(R))$ is monotone.

Proof.
Straightforward. □

Lemma 3.2.10 (continuity of combined functions)
Let $\mathcal{C}$ be a complete set of relations and suppose two functions $\Phi_1, \Phi_2 : \mathcal{C} \to \mathcal{C}$ are continuous.

(1) The function Φ defined by $\Phi(R) = \Phi_1(R) \cup \Phi_2(R)$ is continuous.
(2) The function Φ defined by $\Phi(R) = \Phi_2(\Phi_1(R))$ is continuous.

Proof.
Straightforward. □

The following two lemmata are the key to the fixpoint theorem for relations.

Lemma 3.2.11
Let $\mathcal{C}$ be a complete set of relations and $\Phi : \mathcal{C} \to \mathcal{C}$ a monotone function. If a relation $R \in \mathcal{C}$ satisfies $\Phi(R) \subseteq R$, then $\to_\Phi \subseteq R$.

Proof.
We prove that $\to_{\Phi_k} \subseteq R$ for all $k \in \mathbb{N}$ by induction on k. If $k = 0$ we have $\to_{\Phi_k} = \bot_{\mathcal{C}} \subseteq R$. Suppose $k \geq 1$. We have $\to_{\Phi_k} = \Phi(\to_{\Phi_{k-1}})$ by definition. Combining the induction hypothesis with the monotonicity of Φ yields $\Phi(\to_{\Phi_{k-1}}) \subseteq \Phi(R)$. By assumption $\Phi(R) \subseteq R$. Therefore we obtain $\to_{\Phi_k} \subseteq R$. □

Lemma 3.2.12
Let $\mathcal{C}$ be a complete set of relations and $\Phi : \mathcal{C} \to \mathcal{C}$ a continuous function. The relation $\to_\Phi$ is a fixpoint of Φ, i.e., $\Phi(\to_\Phi) = \to_\Phi$.

Proof.
We have $\to_\Phi = \bigcup_{k\in\mathbb{N}} \to_{\Phi_k}$ by definition and thus $\Phi(\to_\Phi) = \Phi(\bigcup_{k\in\mathbb{N}} \to_{\Phi_k})$. From the continuity assumption, we obtain $\Phi(\bigcup_{k\in\mathbb{N}} \to_{\Phi_k}) = \bigcup_{k\in\mathbb{N}} \Phi(\to_{\Phi_k})$. Using the definition of $\to_\Phi$, we continue as follows: $\bigcup_{k\in\mathbb{N}} \Phi(\to_{\Phi_k}) = \bigcup_{k\in\mathbb{N}} \to_{\Phi_{k+1}} = \bot_\mathcal{C} \cup \bigcup_{k\in\mathbb{N}} \to_{\Phi_{k+1}} = \bigcup_{k\in\mathbb{N}} \to_{\Phi_k} = \to_\Phi$. Therefore we conclude that $\Phi(\to_\Phi) = \to_\Phi$. □

Now we are ready to state the main theorem of this section.

Theorem 3.2.13 (fixpoint theorem for relations)
Let $\mathcal{C}$ be a complete set of relations and $\Phi : \mathcal{C} \to \mathcal{C}$ a continuous function. The relation $\to_\Phi$ is the smallest relation that satisfies $\Phi(\to_\Phi) \subseteq \to_\Phi$.

Proof.
By Lemma 3.2.11 and Lemma 3.2.12. □

We conclude this section with an explanation of how to apply the fixpoint theorem to the analysis of a relation. Consider a complete set $\mathcal{C}$ of relations and a relation $R \in \mathcal{C}$ which is iteratively constructed. Our goal is to characterize the relation R by a property $\mathcal{P}$ on $\mathcal{C}$. We analyze the relation R by the following three steps:
(1) Find a function $\Phi : \mathcal{C} \to \mathcal{C}$ such that $R = \bigcup_{n\in\mathbb{N}} \Phi^n(\bot_\mathcal{C})$.
(2) Prove the continuity of Φ. As a consequence we obtain the fixpoint characterization $R = \bigcap\{\, X \in \mathcal{C} \mid \Phi(X) \subseteq X \,\}$ by Theorem 3.2.13.
(3) Relate Φ with $\mathcal{P}$ and prove the equality $R = \bigcap\{\, X \in \mathcal{C} \mid \mathcal{P}(X) \,\}$.

3.3 Soundness and Completeness of CEL

Based on the technique introduced in the previous two sections, we give a proof of the soundness and completeness of CEL, which is defined in Section 2.4. The goal of this section is to prove that, for every CES $(\mathcal{F}, \mathcal{E})$, the ARS $(\mathcal{T}(\mathcal{F}, \mathcal{V}), \approx_\mathcal{E})$ is sound and complete. In order to characterize provable equivalence by the fixpoint theorem, we introduce a function $\Phi_\mathcal{E}$ for iterative construction.

Definition 3.3.1 (function $\Phi_\mathcal{E}$)
Let $(\mathcal{F}, \mathcal{E})$ be a CES. Define the function $\Phi_\mathcal{E}$ on the set of relations over terms:

$$\Phi_\mathcal{E}(R) = R^{[\mathsf{r}]} \cup R^{[\mathsf{s}]} \cup R^{[\mathsf{t}]} \cup R^{[\mathsf{c}]} \cup R^{[\mathsf{a}]}$$

Here, five operations $(\,\cdot\,)^{[\mathsf{r}]}$, $(\,\cdot\,)^{[\mathsf{s}]}$, $(\,\cdot\,)^{[\mathsf{t}]}$, $(\,\cdot\,)^{[\mathsf{c}]}$, and $(\,\cdot\,)^{[\mathsf{a}]}$ are defined as follows:

$$\begin{array}{rcl} R^{[\mathsf{r}]} & = & \mathrm{Id}_{\mathcal{T}(\mathcal{F},\mathcal{V})}, \\ R^{[\mathsf{s}]} & = & R^{-1}, \\ R^{[\mathsf{t}]} & = & R^2, \\ R^{[\mathsf{c}]} & = & \{\, (f(s_1,\ldots,s_n), f(t_1,\ldots,t_n)) \mid \\ & & \quad f \in \mathcal{F},\ f \text{ is } n\text{-ary},\ s_j \mathrel{R} t_j \text{ for } 1 \le j \le n\}, \\ R^{[\mathsf{a}]} & = & \{\, (l\sigma, r\sigma) \mid l \approx r \Leftarrow c \in \mathcal{E},\ c\sigma \subseteq R\,\}. \end{array}$$

Using the function $\Phi_{\mathcal{E}}$, the definition of provable equivalence (Definition 2.4.4) can be reformulated as follows:

$$\begin{array}{rcll} \approx_{\mathcal{E}_0} & = & \varnothing, & \\ \approx_{\mathcal{E}_{k+1}} & = & {\approx_{\mathcal{E}_k}}^{[\mathsf{r}]} \cup {\approx_{\mathcal{E}_k}}^{[\mathsf{s}]} \cup {\approx_{\mathcal{E}_k}}^{[\mathsf{t}]} \cup {\approx_{\mathcal{E}_k}}^{[\mathsf{c}]} \cup {\approx_{\mathcal{E}_k}}^{[\mathsf{a}]}, & \\ & = & \Phi_{\mathcal{E}}(\approx_{\mathcal{E}_k}) & \text{for all } k \in \mathbb{N}, \\ \approx_{\mathcal{E}} & = & \bigcup_{k\in\mathbb{N}} \approx_{\mathcal{E}_k} \;=\; \bigcup_{k\in\mathbb{N}} {\Phi_{\mathcal{E}}}^k(\varnothing). & \end{array}$$

Next, we prove that $\Phi_{\mathcal{E}}$ is monotone and continuous.

Lemma 3.3.2 (monotonicity and continuity of $\Phi_{\mathcal{E}}$)
Let $\mathcal{E}$ be a CES.
(1) The function $\Phi_{\mathcal{E}}$ is monotone.
(2) The function $\Phi_{\mathcal{E}}$ is continuous.

Proof.
By repeated applications of Lemma 3.2.9(1) we learn that a function constructed from monotone functions by means of unions is also monotone. The same argument is applied to continuity using Lemma 3.2.10(1). Hence we have to show the monotonicity and continuity of the five operations $(\,\cdot\,)^{[\mathsf{r}]}$, $(\,\cdot\,)^{[\mathsf{s}]}$, $(\,\cdot\,)^{[\mathsf{t}]}$, $(\,\cdot\,)^{[\mathsf{c}]}$, and $(\,\cdot\,)^{[\mathsf{a}]}$ introduced in Definition 3.3.1. Monotonicity is easily proved. We only show the continuity of $(\,\cdot\,)^{[\mathsf{a}]}$. The continuity of other operations can be similarly proved. For the proof of continuity we use the monotonicity of $(\,\cdot\,)^{[\mathsf{a}]}$ with Lemma 3.2.8(2). So let $(R_i \mid i \in \mathbb{N})$ be an arbitrary chain on the set of relations on terms and suppose $l\sigma\, (\bigcup_{i\in\mathbb{N}} R_i)^{[\mathsf{a}]}\, r\sigma$ for some conditional equation $l \approx r \Leftarrow c \in \mathcal{E}$ that satisfies $c = s_1 \approx t_1, \ldots s_n \approx t_n$. We have to show that $l\sigma\, (\bigcup_{i\in\mathbb{N}} {R_i}^{[\mathsf{a}]})\, r\sigma$. The inclusion $c\sigma \subseteq \bigcup_{i\in\mathbb{N}} R_i$ holds by the definition of $(\,\cdot\,)^{[\mathsf{a}]}$ and hence we have $s_j\sigma\, (\bigcup_{i\in\mathbb{N}} R_i)\, t_j\sigma$ for all $j \in \{1,\ldots,n\}$. So there is $m_j \in \mathbb{N}$ for every $j \in \{1,\ldots,n\}$ such that $s_j\sigma\, R_{m_j}\, t_j\sigma$. Letting $m = \max\{m_1,\ldots,m_n\}$, we obtain $s_j\sigma\, R_m\, t_j\sigma$ for all $j \in \{1,\ldots,n\}$ and thus $c\sigma \subseteq R_m$. By the definition of $(\,\cdot\,)^{[\mathsf{a}]}$, we have $l\sigma\, {R_m}^{[\mathsf{a}]}\, r\sigma$. Therefore $l\sigma\, (\bigcup_{i\in\mathbb{N}} {R_i}^{[\mathsf{a}]})\, r\sigma$. □

Since the function $\Phi_{\mathcal{E}}$ is continuous, an application of the fixpoint theorem yields the following characterization of provable equivalence.

Lemma 3.3.3 (fixpoint characterization of provable equivalence)
Let $\mathcal{E}$ be a CES. The provable equivalence $\approx_{\mathcal{E}}$ induced by $\mathcal{E}$ is the smallest relation such that $\Phi_{\mathcal{E}}(\approx_{\mathcal{E}}) \subseteq \approx_{\mathcal{E}}$.

Proof.
By the continuity of $\Phi_{\mathcal{E}}$ and the fixpoint theorem for relations (Theorem 3.2.13).
□

For the soundness and completeness of CEL, it remains to relate the inclusion $\Phi_{\mathcal{E}}(\approx_{\mathcal{E}}) \subseteq \approx_{\mathcal{E}}$ with the desired property.

Lemma 3.3.4
Let $(\mathcal{F}, \mathcal{E})$ be a CES and R an arbitrary relation on the set of terms.
(1) R is a congruence if and only if $R^{[\mathsf{r}]} \cup R^{[\mathsf{s}]} \cup R^{[\mathsf{t}]} \cup R^{[\mathsf{c}]} \subseteq R$.
(2) R is applicable to $\mathcal{E}$ if and only if $R^{[\mathsf{a}]} \subseteq R$.
(3) R is a congruence that is applicable to $\mathcal{E}$ if and only if $\Phi_{\mathcal{E}}(R) \subseteq R$.

Proof.
(1) and (2) are straightforward. From these two properties and the definition of $\Phi_{\mathcal{E}}$, property (3) is easily inferred. □

The conjunction of the preceding two lemmata yields the following characterization of provable equivalence, which is equivalent to soundness and completeness.

Corollary 3.3.5 (characterization of provable equivalence)
The provable equivalence $\approx_{\mathcal{E}}$ induced by a CES $\mathcal{E}$ is the smallest congruence that is applicable to $\mathcal{E}$.

Theorem 3.3.6 (soundness and completeness of CEL)
Conditional equational logic is sound and complete, i.e., $\approx_{\mathcal{E}} = =_{\mathsf{M}_{\mathcal{E}}}$ for all CESs $\mathcal{E}$.

Proof.
By Corollary 3.3.5 and Corollary 3.1.12. □

3.4 Soundness and Completeness of Semi-Equational CTRSs

In this section we prove soundness and completeness of semi-equational CTRSs, which is defined in Section 2.5. We regard the convertibility relation induced by a CTRS as the provability relation of the CTRS. So our goal of this section is to prove that, for every semi-equational CTRS $(\mathcal{F}, \mathcal{R})$, the ARS $(\mathcal{T}(\mathcal{F}, \mathcal{V}), \leftrightarrow^*_{\mathcal{R}})$ is sound and complete. Before dealing with the convertibility relation, we first analyze the rewrite relation induced by a semi-equational CTRS. We begin the analysis with the definition of a function for iterative construction.

Definition 3.4.1 (function $\Phi_{\mathcal{R}}$)
Let $\mathcal{R}$ be a CTRS. Define the function $\Phi_{\mathcal{R}}$ on the set of relations over terms, which is used to construct the rewrite relation of the semi-equational CTRS, as follows:

$$\Phi_{\mathcal{R}}(\rightarrow) = (\rightarrow^{\mathcal{R}})^{\mathcal{C}}.$$

Here, operation $(\,\cdot\,)^{\mathcal{R}}$ is defined as follows:

$$\rightarrow^{\mathcal{R}} \quad = \quad \{\, (l\sigma, r\sigma) \mid l \approx r \Leftarrow c \in \mathcal{R},\ c\sigma \subseteq \leftrightarrow^* \}.$$

Using the function $\Phi_{\mathcal{R}}$, the definition of the rewrite relation of a semi-equational CTRS (Definitions 2.5.2 and 2.5.3) can be reformulated as follows:

$$\begin{array}{lllll} \rightarrow_{\mathcal{R}_0} & = & \varnothing, & & \\ \rightarrow_{\mathcal{R}_{k+1}} & = & (\rightarrow_{\mathcal{R}_k}{}^{\mathcal{R}})^{\mathcal{C}} & = & \Phi_{\mathcal{R}}(\rightarrow_{\mathcal{R}_k}) \qquad \text{for all } k \in \mathbb{N}, \\ \rightarrow_{\mathcal{R}} & = & \bigcup_{k\in\mathbb{N}} \rightarrow_{\mathcal{R}_k} & = & \bigcup_{k\in\mathbb{N}} \Phi_{\mathcal{R}}{}^{k}(\varnothing). \end{array}$$

Next, we prove that $\Phi_{\mathcal{R}}$ is monotone and continuous.

Lemma 3.4.2 (monotonicity and continuity of $\Phi_{\mathcal{R}}$)
Let $\mathcal{R}$ be a CTRS.
(1) $\Phi_{\mathcal{R}}$ is monotone.
(2) $\Phi_{\mathcal{R}}$ is continuous.

Proof.
From Lemma 3.2.9(2) and Lemma 3.2.10(2) we learn that composition of functions preserves both monotonicity and continuity. Hence it is sufficient to show the monotonicity and continuity of the two operations introduced in Definition 3.4.1. Since the operation $(\,\cdot\,)^{\mathsf{eqv}}$ is a closure operation, it is monotone. One easily verifies that the operation $(\,\cdot\,)^{\mathcal{R}}$ inherits monotonicity from $(\,\cdot\,)^{\mathsf{eqv}}$. The operation $(\,\cdot\,)^{\mathcal{C}}$ is also monotone because it is a closure operation. The the continuity of $(\,\cdot\,)^{\mathcal{R}}$ is obtained similarly to the $(\,\cdot\,)^{[\mathsf{a}]}$ operation in the proof of Lemma 3.3.2, except that we use the continuity of $(\,\cdot\,)^{\mathsf{eqv}}$, which is easily shown. The continuity of $(\,\cdot\,)^{\mathcal{C}}$ is straightforward. □

Since the function $\Phi_{\mathcal{R}}$ is continuous, an application of the fixpoint theorem yields the following characterization of the rewrite relation.

Lemma 3.4.3 (fixpoint characterization of the rewrite relation)
Let $\mathcal{R}$ be a CTRS.
(1) $\Phi_{\mathcal{R}}(\to_{\mathcal{R}}) = \to_{\mathcal{R}}$.
(2) $\to_{\mathcal{R}}$ is the smallest relation that satisfies $\Phi_{\mathcal{R}}(\to_{\mathcal{R}}) \subseteq \to_{\mathcal{R}}$.

Proof.
Property (1) follows from Lemma 3.2.12 using the monotonicity of $\Phi_{\mathcal{R}}$. Property (2) is derived from the fixpoint theorem (Theorem 3.2.13) using the continuity of $\Phi_{\mathcal{R}}$. □

In order to characterize the rewrite relation of a semi-equational CTRS by properties of relations, it is convenient to introduce a property for CTRSs which is similar to the notion of applicability for CEL.

Definition 3.4.4 (applicable relation)
Let $(\mathcal{F}, \mathcal{R})$ be a CTRS and $\to$ be a relation on $\mathcal{T}(\mathcal{F}, \mathcal{V})$. The relation $\to$ is *applicable* to the semi-equational CTRS $\mathcal{R}^{s}$ if $l\sigma \to r\sigma$ for all conditional equation $l \approx r \Leftarrow c \in \mathcal{R}$ and substitutions σ that satisfy $c\sigma \subseteq \leftrightarrow^*$.

Lemma 3.4.5
Let $\mathcal{R}$ be a CTRS and $\to$ an arbitrary relation on the set of terms.
(1) $\to$ is closed under contexts if and only if $\to^{\mathcal{C}} \subseteq \to$.
(2) $\to$ is applicable to the semi-equational CTRS $\mathcal{R}^{s}$ if and only if $\to^{\mathcal{R}} \subseteq \to$.

Proof.
Straightforward. □

Combining the preceding two lemmata yields a useful characterization of the relation $\to_{\mathcal{R}}$ (associated with the semi-equational CTRS $\mathcal{R}^{s}$).

Lemma 3.4.6 (characterization of $\to_{\mathcal{R}}$)
Let $\mathcal{R}$ be a CTRS. The relation $\to_{\mathcal{R}}$ is the smallest relation that is both closed under contexts and applicable to the semi-equational CTRS $\mathcal{R}^{s}$.

Proof.
Since $\to_{\mathcal{R}}$ is the fixpoint of $\Phi_{\mathcal{R}}$ by Lemma 3.4.3 (1), we have $\Phi_{\mathcal{R}}(\to_{\mathcal{R}}) = \to_{\mathcal{R}}$ and thus $(\to_{\mathcal{R}}{}^{\mathcal{R}})^{\mathcal{C}} = \to_{\mathcal{R}}$ by definition. Hence, using the monotonicity and idempotency of the context closure operation $(\,\cdot\,)^{\mathcal{C}}$, we can compute as follows: $\to_{\mathcal{R}}{}^{\mathcal{C}} = ((\to_{\mathcal{R}}{}^{\mathcal{R}})^{\mathcal{C}})^{\mathcal{C}} = (\to_{\mathcal{R}}{}^{\mathcal{R}})^{\mathcal{C}} = \to_{\mathcal{R}}$. Therefore, $\to_{\mathcal{R}}$ is closed under contexts by Lemma 3.4.5(1). Moreover, by using the incrementality of $(\,\cdot\,)^{\mathcal{C}}$, we obtain $\to_{\mathcal{R}}{}^{\mathcal{R}} \subseteq (\to_{\mathcal{R}}{}^{\mathcal{R}})^{\mathcal{C}} = \to_{\mathcal{R}}$. Hence $\to_{\mathcal{R}}$ is applicable to $\mathcal{R}^{s}$ by Lemma 3.4.5(2). It remains to show that $\to_{\mathcal{R}}$ is the smallest relation satisfying the two properties.

So let $\to$ be a relation which is closed under contexts and applicable to $\mathcal{R}^s$. We have to show that $\to_{\mathcal{R}} \subseteq \to$. From Lemma 3.4.5 and the monotonicity of $(\,\cdot\,)^{\mathcal{C}}$, we infer that $({\to_{\mathcal{R}}}^{\mathcal{R}})^{\mathcal{C}} \subseteq \to^{\mathcal{C}} \subseteq \to$ and hence $\Phi_{\mathcal{R}}(\to) = \to$ by definition. The desired inclusion $\to_{\mathcal{R}} \subseteq \to$ follows from Lemma 3.4.3(2). □

Due to this lemma we can avoid proofs by induction on the level of conditional rewrite steps in the next chapter.

Before proving the soundness and completeness of semi-equational CTRSs we need some preparation.

Lemma 3.4.7 (property of equivalence closure)
Let R be a relation on terms.
(1) If R is closed under contexts, then R^{eqv} is also closed under contexts.
(2) If R is applicable to a semi-equational CTRS $\mathcal{R}^s$, then R^{eqv} is also applicable to $\mathcal{R}^s$,

Proof.
Property (1) follows from the inclusion $(R^{\mathsf{eqv}})^{\mathcal{C}} \subseteq (R^{\mathcal{C}})^{\mathsf{eqv}}$ and property (2) is due to the inclusion $(R^{\mathsf{eqv}})^{\mathcal{R}} \subseteq (R^{\mathcal{R}})^{\mathsf{eqv}}$. Both inclusions are easy to prove. □

Lemma 3.4.8 (relationship between context closure and compatibility)
Let R be a reflexive and transitive relation on terms. R is closed under contexts if and only if R is compatible.

Proof.
The direction "$\Rightarrow$" is proved by induction on the number of arguments for every function symbol. The proof of the direction "$\Leftarrow$" is by structural induction on contexts. □

Now we can give a characterization of the convertibility $\leftrightarrow^*_{\mathcal{R}}$, which is equivalent to soundness and completeness.

Theorem 3.4.9 (characterization of $\leftrightarrow^*_{\mathcal{R}}$)
Let $\mathcal{R}$ be a CTRS. The relation $\leftrightarrow^*_{\mathcal{R}}$ is the smallest congruence that is applicable to the semi-equational CTRS $\mathcal{R}^s$.

Proof.
According to the characterization of the rewrite relation (Lemma 3.4.6), the rewrite relation of a semi-equational CTRS $\mathcal{R}^s$ is closed under contexts and applicable to $\mathcal{R}^s$. So $\leftrightarrow^*_{\mathcal{R}}$ inherits these two properties from the rewrite relation by Lemma 3.4.7. Hence $\leftrightarrow^*_{\mathcal{R}}$ is a congruence by Lemma 3.4.8. It remains to show that $\leftrightarrow^*_{\mathcal{R}}$ is the smallest relation. Let $\sim$ be an arbitrary congruence that is applicable to $\mathcal{R}^s$. We

have to show that $\leftrightarrow^*_{\mathcal{R}} \subseteq \sim$. Since congruence is reflexive and transitive, $\sim$ is closed under contexts by Lemma 3.4.7. An application of the characterization of the rewrite relation yields that $\rightarrow_{\mathcal{R}} \subseteq \sim$ because $\sim^{\mathsf{eqv}} = \sim$. Therefore $\leftrightarrow^*_{\mathcal{R}} \subseteq \sim$ by the monotonicity of $(\,\cdot\,)^{\mathsf{eqv}}$ and the equality $\sim^{\mathsf{eqv}} = \sim$. □

Corollary 3.4.10 (soundness and completeness of $\mathcal{R}^{\mathrm{s}}$ [Kap84])
Every semi-equational CTRS $\mathcal{R}$ is sound and complete, i.e., $\leftrightarrow^*_{\mathcal{R}} = =_{\mathsf{M}_{\mathcal{E}}}$.

Chapter 4

Logicality of CTRSs

From the viewpoint of algebraic semantics, semi-equational CTRSs have the nice properties of soundness and completeness, as we have observed in Section 3.4. Hence they have the nice proof-theoretical property of logicality, meaning that their provability relations corresponds exactly with the provable equivalence of the underlying CES. However, from the computational viewpoint, semi-equational CTRSs are unnatural because satisfiability of the conditional part of conditional equations is checked by bidirectional rewriting, which goes against the spirit of rewriting. The aim of this chapter is to investigate what classes of join and oriented CTRSs are sound and complete.

In Section 4.1 we introduce the notion of logicality, characterize the rewrite relations of CTRSs, and prove that join and oriented CTRSs are always sound. Section 4.2 is a review of known logicality results for join CTRSs. New logicality results for oriented CTRSs are provided in Section 4.3 In Section 4.4 we give sufficient conditions for logicality of join and oriented systems by imposing restrictions on semi-equational CTRSs. In Section 4.5 we systematically compare various pairs of relations induced by different types of CTRSs for understanding the principle of logicality. The results presented in this chapter are based on the joint work with J. Avenhaus, C. Loría-Sáenz, and A. Middeldorp (cf. [YALSM99], [YALSM97]).

4.1 Logicality

In the previous chapter we observed that CEL is sound and complete. Hence, for the purpose of ensuring soundness and completeness of a CTRS, it is sufficient to compare the logical strength of a CTRS with CEL. We call a CTRS logical if it has the same logical strength as CEL.

Definition 4.1.1 (logicality)
A CTRS $\mathcal{R}$ is called *logical* if the relations $\approx_\mathcal{R}$ and $\leftrightarrow^*_\mathcal{R}$ coincide.

The terminology logicality stems from [BG89] although the study of the concept dates back to Kaplan [Kap84]. Logicality is an important property because it entails that (bidirectional) rewriting is sound and complete with respect to the underlying CES. Hence logicality implies that an equation $s \approx t$ is provable by rewriting ($s \leftrightarrow^*_\mathcal{R} t$) if and only if it is valid in all models of the underlying CES. Logicality of semi-equational CTRSs was first proved by Kaplan.

Theorem 4.1.2 (logicality of semi-equational CTRSs [Kap84])
Every semi-equational CTRS is logical.

Proof.
By the characterization of provable equivalence (Corollary 3.3.5) and the characterization of the convertibility induced by a semi-equational CTRS (Theorem 3.4.9).
□

Corollary 4.1.3 (logicality of non-semi-equational CTRSs)
Let $\mathcal{R}$ be a CTRS.
(1) The join CTRS $\mathcal{R}^\mathrm{j}$ is logical if and only if $\leftrightarrow^*_\mathrm{j} = \leftrightarrow^*_\mathrm{s}$.
(2) The oriented CTRS $\mathcal{R}^\mathrm{o}$ is logical if and only if $\leftrightarrow^*_\mathrm{o} = \leftrightarrow^*_\mathrm{s}$.

From Corollary 4.1.3 we learn that it is important for ensuring logicality of non-semi-equational CTRSs to compare the relations $\leftrightarrow^*_\mathrm{o}$, $\leftrightarrow^*_\mathrm{j}$, and $\leftrightarrow^*_\mathrm{s}$. Hence, in this chapter, we compare various relations which are induced by different types of CTRSs and derive logicality results. For that purpose, characterizations of rewrite relations are frequently applied. So we extend the result for semi-equational CTRSs (Theorem 3.4.3) to join and oriented CTRSs.

Definition 4.1.4 (applicable relation)
Let $(\mathcal{F}, \mathcal{R})$ be a CTRS and $\to$ be a relation on $\mathcal{T}(\mathcal{F}, \mathcal{V})$. The relation $\to$ is *applicable* to $\mathcal{R}$ if $l\sigma \to r\sigma$ for all conditional equation $l \to r \Leftarrow c \in \mathcal{R}$ and substitutions σ that satisfy $\mathcal{R} \vdash c\sigma$.

Lemma 4.1.5 (characterization of rewrite relations)
Let $\mathcal{R}$ be a CTRS. The rewrite relation $\to_\mathcal{R}$ induced by $\mathcal{R}$ is the smallest relation that is both closed under contexts and applicable to $\mathcal{R}$.

Proof.
We already proved the result for semi-equational CTRSs in Theorem 3.4.3. The results for join and oriented CTRSs are similarly proved. □

Lemma 4.1.6 (logical strength of CTRSs)
For every CTRS $\mathcal{R}$ we have $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{j}} \subseteq \rightarrow_{\mathrm{s}}$.

Proof.
We prove the inclusion $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{j}}$ by using the characterization of $\rightarrow_{\mathrm{o}}$ (Lemma 4.1.5). Since $\rightarrow_{\mathrm{j}}$ is closed under contexts by the characterization of $\rightarrow_{\mathrm{j}}$, it suffices to show the applicability of $\rightarrow_{\mathrm{j}}$ to $\mathcal{R}^{\mathrm{o}}$. So suppose $l \rightarrow r \Leftarrow c \in \mathcal{R}$ and σ is a substitution with $c\sigma \subseteq \rightarrow_{\mathrm{j}}^*$. Since $\rightarrow_{\mathrm{j}}^* \subseteq \downarrow_{\mathrm{j}}$, $c\sigma \subseteq \rightarrow_{\mathrm{j}}^*$ implies $c\sigma \subseteq \downarrow_{\mathrm{j}}$. Hence we obtain $l\sigma \rightarrow_{\mathrm{j}} r\sigma$ from the characterization of $\rightarrow_{\mathrm{j}}$. The proof of the inclusion $\rightarrow_{\mathrm{j}} \subseteq \rightarrow_{\mathrm{s}}$ is similar: we use Lemma 4.1.5 and the inclusion $\downarrow_{\mathrm{s}} \subseteq \leftrightarrow_{\mathrm{s}}^*$. □

This lemma tells us that every join and oriented CTRS is sound since its corresponding semi-equational CTRS is sound. Nevertheless, they are not always complete as examples in the subsequent sections illustrate. In the next section, we review known logicality results in an uniform way to clarify sufficient conditions that guarantee completeness of join and oriented CTRSs.

4.2 Logicality of Join CTRSs

We start our review of logicality results with join CTRSs. Join CTRSs need not be logical, as shown in the following example.

Example 4.2.1 (join CTRS lacking logicality)
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{lll} \mathsf{a} \rightarrow \mathsf{b} & & \\ \mathsf{a} \rightarrow \mathsf{c} & & \\ \mathsf{d} \rightarrow \mathsf{e} & \Leftarrow & \mathsf{b} \approx \mathsf{c} \end{array} \right\}.$$

We have $\mathsf{d} \rightarrow_{\mathrm{s}} \mathsf{e}$ since $\mathsf{b} \;{}_{\mathrm{s}}\!\leftarrow \mathsf{a} \rightarrow_{\mathrm{s}} \mathsf{c}$. However, $\mathsf{d} \rightarrow_{\mathrm{j}} \mathsf{e}$ does not hold because the condition $\mathsf{b} \downarrow_{\mathrm{j}} \mathsf{c}$ is not satisfied. Hence $\mathsf{d} \leftrightarrow_{\mathrm{j}}^* \mathsf{e}$ does not hold either.

Note that the above $\mathcal{R}^{\mathrm{j}}$ lacks confluence. Kaplan observed that this is essential. From the proof of the theorem below we know that confluence is a natural requirement for the logicality of non-semi-equational CTRSs. The point is that convertibility can be transformed into joinability, using the confluence assumption.

Theorem 4.2.2 (sufficient condition for $\rightarrow_{\mathrm{j}} = \rightarrow_{\mathrm{s}}$)
Let $\mathcal{R}$ be a CTRS. If $\mathcal{R}^{\mathrm{j}}$ is confluent then $\rightarrow_{\mathrm{j}} = \rightarrow_{\mathrm{s}}$.

Proof.
We already know that $\rightarrow_{\mathrm{j}} \subseteq \rightarrow_{\mathrm{s}}$. For the reverse inclusion we use the characterization of rewrite relations (Lemma 4.1.5). Since $\rightarrow_{\mathrm{j}}$ is closed under contexts we

only need to show the applicability of $\rightarrow_{\mathrm{j}}$ to $\mathcal{R}^{\mathrm{s}}$. So let $l \rightarrow r \Leftarrow c \in \mathcal{R}$ and σ a substitution that satisfies $c\sigma \subseteq \leftrightarrow_{\mathrm{j}}^*$. We have to show that $l\sigma \rightarrow_{\mathrm{j}} r\sigma$. By the confluence assumption we obtain $\leftrightarrow_{\mathrm{j}}^* \subseteq \downarrow_{\mathrm{j}}$ and thus $l\sigma \rightarrow_{\mathrm{j}} r\sigma$ follows from the characterization of $\rightarrow_{\mathrm{j}}$. □

Corollary 4.2.3 (logicality of join CTRSs [Kap84])
Every confluent join CTRS is logical.

4.3 Logicality of Oriented CTRSs

Logicality of oriented CTRSs is the main topic of this chapter. For oriented CTRSs confluence is not sufficient for ensuring logicality, as shown by the following example.

Example 4.3.1 (confluent oriented CTRS lacking logicality)
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{lll} \mathsf{a} \rightarrow \mathsf{c} & & \\ \mathsf{b} \rightarrow \mathsf{c} & \Leftarrow & \mathsf{c} \approx \mathsf{a} \end{array} \right\}.$$

We have $\mathsf{b} \rightarrow_{\mathrm{s}} \mathsf{c}$ since $\mathsf{c}\ {}_{\mathrm{s}}\!\leftarrow \mathsf{a}$. However, $\mathsf{b} \rightarrow_{\mathrm{o}} \mathsf{c}$ does not hold because the condition $\mathsf{c} \rightarrow_{\mathrm{o}}^* \mathsf{a}$ is not satisfied. Hence $\mathsf{b} \leftrightarrow_{\mathrm{o}}^* \mathsf{c}$ does not hold either. Note that $\mathcal{R}^{\mathrm{o}}$ is confluent.

In semi-equational systems, the right-hand sides of conditions may be rewriten for checking the satisfiability of the conditions. However, it is not allowed to rewrite the right-hand sides of the conditions in oriented systems. As a consequence, to impose restrictions on reducibility of the right-hand sides of the conditions is crucial for ensuring logicality of oriented systems.

Definition 4.3.2 (normality)
Let $\mathcal{R}$ be a CTRS. A term t is called *normal* if it is ground and does not encompass any left-hand side l of a conditional rewrite rule $l \rightarrow r \Leftarrow c$ in $\mathcal{R}$. The latter requirement means that t is irreducible with respect to the unconditional TRS obtained from $\mathcal{R}$ by dropping all conditions. We say that $\mathcal{R}$ is *normal* if every right-hand side t of an equation $s \approx t$ in the conditional part c of a conditional rewrite rule $l \rightarrow r \Leftarrow c$ in $\mathcal{R}$ is normal.

Normal oriented systems were called to be of type $\mathrm{III}_{\mathrm{n}}$ by Bergstra and Klop [BK86]. Note that every normal term is irreducible and also that normality is a decidable property of finite CTRSs. It is easy to see that the CTRS $\mathcal{R}$ in

Example 4.3.1 is not a normal CTRS. If we impose the syntactic restriction of normality to the CTRS, logicality of oriented CTRSs directly follows from the result in the previous section. This is because the rewrite step of a normal oriented CTRS coincides with that of the associated join CTRS.

Theorem 4.3.3 (equivalence of rewrite relations by normality)
Let $\mathcal{R}$ be a normal CTRS. If $\mathcal{R}^{\mathrm{o}}$ is confluent then $\rightarrow_{\mathrm{o}} = \rightarrow_{\mathrm{j}} = \rightarrow_{\mathrm{s}}$.

Proof.
According to Lemma 4.1.6 we have $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{j}}$. In order to prove the reverse inclusion we employ Lemma 4.1.5. Since $\rightarrow_{\mathrm{o}}$ is closed under contexts we only need to show the applicability of $\rightarrow_{\mathrm{j}}$ to $\mathcal{R}^{\mathrm{s}}$. So let $l \rightarrow r \Leftarrow c \in \mathcal{R}$ and σ a substitution such that $c\sigma \subseteq \leftrightarrow_{\mathrm{j}}^{*}$. We have to show that $l\sigma \rightarrow_{\mathrm{j}} r\sigma$. By the confluence assumption we obtain $\leftrightarrow_{\mathrm{j}}^{*} \subseteq \downarrow_{\mathrm{j}}$ and thus $l\sigma \rightarrow_{\mathrm{j}} r\sigma$ follows from the characterization of $\rightarrow_{\mathrm{j}}$. Hence $\rightarrow_{\mathrm{o}} = \rightarrow_{\mathrm{j}}$ and thus $\mathcal{R}^{\mathrm{j}}$ is also confluent. From Theorem 4.2.2 we obtain $\rightarrow_{\mathrm{j}} = \rightarrow_{\mathrm{s}}$. Hence $\rightarrow_{\mathrm{o}} = \rightarrow_{\mathrm{j}} = \rightarrow_{\mathrm{s}}$. □

Corollary 4.3.4 (logicality of normal CTRSs)
Every confluent normal oriented CTRS is logical.

In the presence of extra variables in the right-hand sides of the conditional rewrite rules, normality is too strong a requirement. Such extra variables appear naturally in applications of conditional rewriting (e.g. [ALS94], [BG89], [Han95], [SMI95]). There are two ways to weaken the normality requirement in Corollary 4.3.4. We weaken normality by imposing the operational requirement of weak normalization on CTRSs, or replacing it with other syntactic requirements on CTRSs. The following key lemma, which gives a more abstract sufficient condition, is useful for ensuring the logicality of oriented CTRSs.

Lemma 4.3.5 (sufficient condition for logicality of oriented CTRSs)
Let $\mathcal{R}^{\mathrm{o}}$ be a confluent oriented CTRS. If for every $l \rightarrow r \Leftarrow c \in \mathcal{R}$ and every substitution σ that satisfies $c\sigma \subseteq \downarrow_{\mathrm{o}}$ there exists a substitution τ such that
(1) $\sigma(x) \rightarrow_{\mathrm{o}}^{*} \tau(x)$ for all $x \in \mathcal{V}$, and
(2) $c\tau \subseteq \rightarrow_{\mathrm{o}}^{*}$
then $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{s}} \subseteq \rightarrow_{\mathrm{o}} \circ \downarrow_{\mathrm{o}}$ hence $\mathcal{R}^{\mathrm{o}}$ is logical.

Proof.
The inclusion $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{s}}$ follows from Lemma 4.1.6. For the inclusion $\rightarrow_{\mathrm{s}} \subseteq \rightarrow_{\mathrm{o}} \circ \downarrow_{\mathrm{o}}$ we use the characterization of $\rightarrow_{\mathrm{s}}$ (Lemma 4.1.5). The proof of this inclusion is sketched in Figure 4.1. Since reflexive-transitive closure and composition preserve closure under contexts, $\rightarrow_{\mathrm{o}} \circ \downarrow_{\mathrm{o}}$ is closed under contexts. It remains to show the

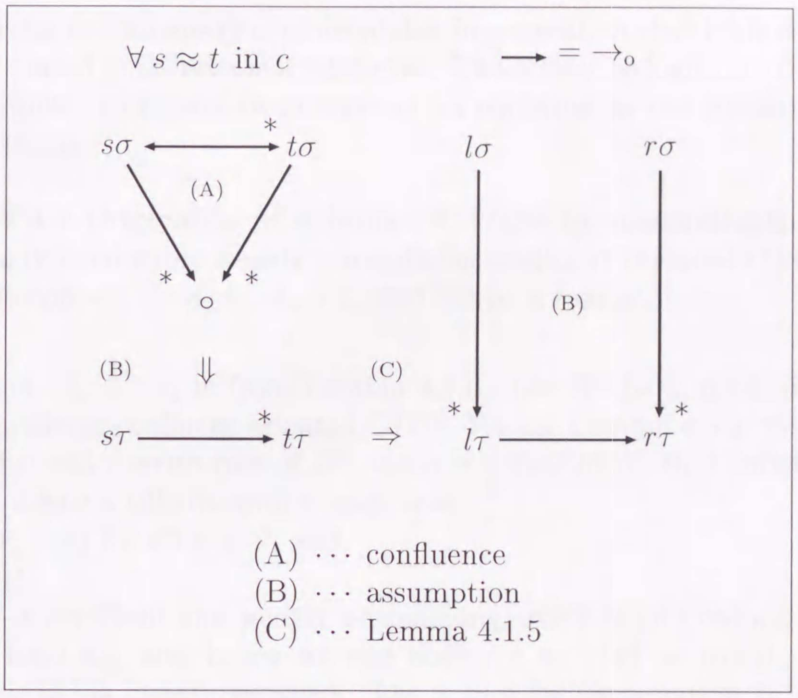


Figure 4.1: Proof of Lemma 4.3.5.

applicability of $\to_{\mathrm{o}} \circ \downarrow_{\mathrm{o}}$ to $\mathcal{R}^{\mathrm{s}}$. So let $l \to r \Leftarrow c \in \mathcal{R}$ and σ a substitution such that $c\sigma \subseteq (\to_{\mathrm{o}} \circ \downarrow_{\mathrm{o}})^{\mathsf{eqv}} = \leftrightarrow^*_{\mathrm{o}}$. We have to show that $l\sigma \to_{\mathrm{o}} \circ \downarrow_{\mathrm{o}} r\sigma$. Confluence of $\mathcal{R}^{\mathrm{o}}$ yields $c\sigma \subseteq \downarrow_{\mathrm{o}}$. By assumption there exists a substitution τ such that $\sigma(x) \to^*_{\mathrm{o}} \tau(x)$ for all $x \in \mathcal{V}$ and $c\tau \subseteq \to^*_{\mathrm{o}}$. The latter statement implies $l\tau \to_{\mathrm{o}} r\tau$. The first statement implies $l\sigma \to^*_{\mathrm{o}} l\tau$ and $r\sigma \to^*_{\mathrm{o}} r\tau$. Therefore $l\sigma \to_{\mathrm{o}} \circ \downarrow_{\mathrm{o}} r\sigma$. □

Now we discuss logicality of oriented CTRSs with weak normalization assumption.

Definition 4.3.6 (strong irreducibility)
Let $\mathcal{R}$ be a CTRS. A term t is called *strongly irreducible* if $t\sigma$ is irreducible for every irreducible substitution σ. We say that $\mathcal{R}$ is strongly irreducible if every right-hand side t of an equation $s \approx t$ in the conditional part c of a conditional rewrite rule $l \to r \Leftarrow c$ in $\mathcal{R}$ is strongly irreducible.

Normal CTRSs are clearly strongly irreducible. Note that irreducibility depends on the rewrite relation associated with $\mathcal{R}$, so it is possible that an oriented CTRS $\mathcal{R}^{\mathrm{o}}$ is strongly irreducible whereas the corresponding join CTRS $\mathcal{R}^{\mathrm{j}}$ is not. Because it is undecidable whether a term is irreducible with respect to a CTRS

[Kap84], strong irreducibility is undecidable in general. A decidable sufficient condition is presented in Definition 4.3.9 below. The following logicality theorem relies on the possibility to reduce both sides of an equation in the conditional part to irreducible terms.

Theorem 4.3.7 (logicality of oriented CTRSs by normalization)
Every strongly irreducible weakly normalizing confluent oriented CTRS $\mathcal{R}^{\mathrm{o}}$ satisfies the inclusion $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{s}} \subseteq \rightarrow_{\mathrm{o}} \circ \downarrow_{\mathrm{o}}$ and hence is logical.[†1]

Proof.
The inclusion $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{s}}$ is from Lemma 4.1.6. Let $\mathcal{R}^{\mathrm{o}}$ be a strongly irreducible weakly normalizing confluent oriented CTRS. We use Lemma 4.3.5. So let $l \rightarrow r \Leftarrow c$ be a conditional rewrite rule of $\mathcal{R}^{\mathrm{o}}$ and σ a substitution that satisfies $c\sigma \subseteq \downarrow_{\mathrm{o}}$. We have to define a substitution τ such that
(1) $\sigma(x) \rightarrow_{\mathrm{o}}^{*} \tau(x)$ for all $x \in \mathcal{V}$, and
(2) $c\tau \subseteq \rightarrow_{\mathrm{o}}^{*}$.
Because $\mathcal{R}^{\mathrm{o}}$ is confluent and weakly normalizing, every term t reduces to a unique irreducible term $t\downarrow_{\mathrm{o}}$ and hence we can define τ as $\tau(x) = \sigma(x)\downarrow_{\mathrm{o}}$ for all $x \in \mathcal{V}$. Property (1) is clearly satisfied. Let $s \approx t$ be an equation in c. We have $s\sigma \downarrow_{\mathrm{o}} t\sigma$. From (1) we infer that $s\sigma \rightarrow_{\mathrm{o}}^{*} s\tau$ and $t\sigma \rightarrow_{\mathrm{o}}^{*} t\tau$ and thus $s\tau \leftrightarrow_{\mathrm{o}}^{*} t\tau$. Since τ is irreducible by construction, $t\tau$ is irreducible by the strong irreducibility assumption. Confluence of $\mathcal{R}^{\mathrm{o}}$ yields $s\tau \rightarrow_{\mathrm{o}}^{*} t\tau$. We conclude that property (2) holds. □

Example 4.3.1 shows that Theorem 4.3.7 cannot be strengthened by dropping the strong irreducibility requirement. The following example shows the necessity of weak normalization.

Example 4.3.8 (necessity of weak normalization)
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{rcll} \mathsf{a} & \rightarrow & \mathsf{a} & \\ \mathsf{f(a)} & \rightarrow & \mathsf{a} & \\ \mathsf{g(x)} & \rightarrow & \mathsf{b} & \Leftarrow \; \mathsf{a} \approx \mathsf{f(x)} \end{array} \right\}.$$

We have $\mathsf{a} \; {}_{\mathrm{s}}\!\leftarrow \mathsf{f(a)}$ and thus $\mathsf{g(a)} \rightarrow_{\mathrm{s}} \mathsf{b}$. However, since there is no term t such that $\mathsf{a} \rightarrow_{\mathrm{o}}^{*} \mathsf{f}(t)$, the relation $\rightarrow_{\mathrm{o}}$ coincides with the rewrite relation induced by the unconditional TRS $\mathcal{S} = \{\mathsf{a} \rightarrow \mathsf{a},\ \mathsf{f(a)} \rightarrow \mathsf{a}\}$. Hence $\mathsf{g(a)} \leftrightarrow_{\mathrm{o}}^{*} \mathsf{b}$ does not hold and hence $\mathcal{R}^{\mathrm{o}}$ is not logical. Clearly the TRS $\mathcal{S}$ and thus $\mathcal{R}^{\mathrm{o}}$ is confluent. Furthermore, $\mathcal{R}^{\mathrm{o}}$ is strongly irreducible because there is no irreducible term t such that $\mathsf{f}(t)$ is reducible.

[†1] This result originates from [ALS93].

Definition 4.3.9 (absolute irreducibility)
Let $\mathcal{R}$ be a CTRS. A term t is called *absolutely irreducible* if no non-variable subterm of t unifies (after variable renaming) with the left-hand side l of a conditional rewrite rule $l \rightarrow r \Leftarrow c \in \mathcal{R}$. We say that $\mathcal{R}$ is absolutely irreducible if every right-hand side t of an equation $s \approx t$ in the conditional part c of a conditional rewrite rule $l \rightarrow r \Leftarrow c$ in $\mathcal{R}$ is absolutely irreducible.

Unlike strong irreducibility, absolute irreducibility does not depend on the rewrite relation associated with $\mathcal{R}$. That is to say, absolute irreducibility is a property of CESs. Note that every normal CTRS is absolutely irreducible but not vice-versa.

The CTRS $\mathcal{R}^{\mathrm{o}}$ of Example 4.3.8 is not absolutely irreducible since the right-hand side $\mathsf{f}(x)$ of the condition $\mathsf{a} \approx \mathsf{f}(x)$ in the rule $\mathsf{g}(x) \rightarrow \mathsf{b} \Leftarrow \mathsf{a} \approx \mathsf{f}(x)$ is unifiable with the left-hand side $\mathsf{f}(\mathsf{a})$ of the rule $\mathsf{f}(\mathsf{a}) \rightarrow \mathsf{a}$. Nevertheless, even if we strengthen strong irreducibility to absolute irreducibility, we cannot dispense with weak normalization in Theorem 4.3.7 as shown by the following example.

Example 4.3.10 (necessity of weak normalization)
Consider the CTRS[†2]

$$\mathcal{R} = \left\{ \begin{array}{rcll} \mathsf{a} & \rightarrow & \mathsf{b} & \\ \mathsf{b} & \rightarrow & \mathsf{a} & \\ \mathsf{f}(\mathsf{a},\mathsf{b}) & \rightarrow & \mathsf{c} & \\ \mathsf{g}(x) & \rightarrow & \mathsf{d} & \Leftarrow \quad \mathsf{c} \approx \mathsf{f}(x,x) \end{array} \right\}.$$

We have $\mathsf{c}\ {}_{\mathrm{s}}\!\!\leftarrow \mathsf{f}(\mathsf{a},\mathsf{b})\ {}_{\mathrm{s}}\!\!\leftarrow \mathsf{f}(\mathsf{a},\mathsf{a})$ and thus $\mathsf{g}(\mathsf{a}) \rightarrow_{\mathrm{s}} \mathsf{d}$. However, since there is no term t such that $\mathsf{c} \rightarrow_{\mathrm{o}}^{*} \mathsf{f}(t,t)$, the relation $\rightarrow_{\mathrm{o}}$ coincides with the rewrite relation induced by the unconditional TRS $\mathcal{S} = \{\mathsf{a} \rightarrow \mathsf{b},\ \mathsf{b} \rightarrow \mathsf{a},\ \mathsf{f}(\mathsf{a},\mathsf{b}) \rightarrow \mathsf{c}\}$. Clearly $\mathsf{g}(\mathsf{a}) \leftrightarrow_{\mathcal{S}}^{*} \mathsf{d}$ does not hold. Hence $\mathcal{R}^{\mathrm{o}}$ is not logical. Note that $\mathcal{S}$ and thus $\mathcal{R}^{\mathrm{o}}$ is confluent. Furthermore, $\mathcal{R}^{\mathrm{o}}$ is absolutely irreducible because the term $\mathsf{f}(x,x)$ does not unify with $\mathsf{f}(\mathsf{a},\mathsf{b})$.

The non-linearity of the term $\mathsf{f}(x,x)$ in the above example is essential, as we will see in Section 5.3.

Since in applications of conditional rewriting weak normalization is often a severe restriction, e.g. CTRSs that model (lazy) functional programs are not weakly normalizing in general, we are especially interested in a sufficient condition for logicality of oriented CTRSs that does not rely on weak normalization. The above examples show that the problem with strong and absolute irreducibility is that the structure of the right-hand sides of equations in the conditional parts are not

[†2] This example refutes [ALS93, Theorem 5.2]

preserved under reduction. For instance, in Example 4.3.8 we have $\mathsf{f}(\mathsf{a}) \rightarrow_{\mathrm{o}} \mathsf{a}$ destroying the structure $\mathsf{f}(\cdot)$. Absolute irreducibility guarantees that the structure of the right-hand sides of equations in the conditional parts is preserved by one-step reduction but not by many-step reduction: in Example 4.3.10 we have $\mathsf{f}(\mathsf{a},\mathsf{a}) \rightarrow_{\mathrm{o}} \mathsf{f}(\mathsf{a},\mathsf{b}) \rightarrow_{\mathrm{o}} \mathsf{c}$ destroying $\mathsf{f}(\cdot,\cdot)$.

The condition defined below guarantees that the structure of the right-hand sides of equations in the conditional parts is preserved by many-step reduction.

Definition 4.3.11 (stability)
Let $\mathcal{R}$ be a CTRS. A term s is called *stable* if $p \notin \mathsf{Pos}_{\mathcal{F}}(s)$ whenever $s\sigma \rightarrow^*_{\mathcal{R}} t \xrightarrow{p}_{\mathcal{R}} u$, for all substitutions σ, terms t and u, and positions p. We say that $\mathcal{R}$ is stable if every right-hand side t of an equation $s \approx t$ in the conditional part c of a conditional rewrite rule $l \rightarrow r \Leftarrow c$ in $\mathcal{R}$ is stable.

The structure preservation of stable terms is formally expressed in the following lemma.

Lemma 4.3.12 (structure preservation of stable terms)
Let $\mathcal{R}$ be a CTRS. If s is a stable term and $s\sigma \rightarrow^*_{\mathcal{R}} t$ then
(1) $\mathsf{root}(s\sigma_{|p}) = \mathsf{root}(t_{|p})$ for all $p \in \mathsf{Pos}_{\mathcal{F}}(s)$, and
(2) $s\sigma_{|p} \rightarrow^*_{\mathcal{R}} t_{|p}$ for all $p \in \mathsf{Pos}_{\mathcal{V}}(s)$.

Proof.
We use induction on the length of the reduction $s\sigma \rightarrow^*_{\mathcal{R}} t$. The case of zero length is trivial. Suppose $s\sigma \rightarrow^*_{\mathcal{R}} t' \xrightarrow{q}_{\mathcal{R}} t$. The proof of this case is illustrated in Figure 4.2. Stability yields $q \notin \mathsf{Pos}_{\mathcal{F}}(s)$ and hence there exists a $q' \in \mathsf{Pos}_{\mathcal{V}}(s)$ such that $q' \leq q$ by property (1) of the induction hypothesis. In order to prove property (1), suppose $p \in \mathsf{Pos}_{\mathcal{F}}(s)$. From $q' \nleq p$ and $q' \leq q$ we infer that $\mathsf{root}(t'_{|p}) = \mathsf{root}(t_{|p})$. The induction hypothesis yields the desired result. Suppose $p \in \mathsf{Pos}_{\mathcal{V}}(s)$ for the proof of property (2). It suffices to show that $t'_{|p} \rightarrow^*_{\mathcal{R}} t_{|p}$ by induction hypothesis. If $p = q'$ then $t'_{|p} \xrightarrow{q\backslash q'}_{\mathcal{R}} t_{|p}$ otherwise $t'_{|p} = t_{|p}$. □

The next lemma expresses the fact that for confluent CTRSs the substitution part of an instance of a stable term can be consistently reduced. This property plays a crucial role in the proof of one of our main results (Theorem 4.3.16 below).

Lemma 4.3.13 (consistent reduction from a stable term)
Let $\mathcal{R}$ be a confluent CTRS. If s is a stable term and $s\sigma \rightarrow^*_{\mathcal{R}} t$ then there exists a substitution τ such that
(1) $\sigma(x) \rightarrow^*_{\mathcal{R}} \tau(x)$ for all $x \in \mathcal{V}$, and

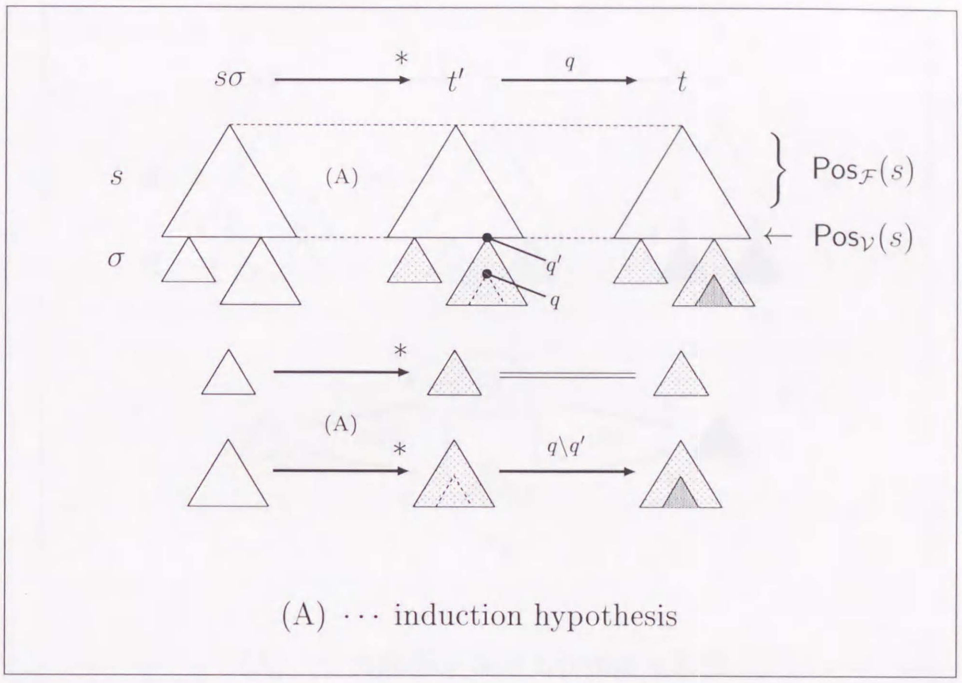


Figure 4.2: Proof of Lemma 4.3.12.

(2) $t \rightarrow^*_{\mathcal{R}} s\tau$.

Proof.

If s is a ground term then it must be irreducible and hence any substitution τ satisfies both requirements. Suppose s is not ground. Let x be an arbitrary variable in s and define $A_x = \{\, t_{|p} \mid s_{|p} = x \,\}$. See Figure 4.3. Since $\sigma(x) \rightarrow^*_{\mathcal{R}} u$ for every $u \in A_x$ by part (2) of Lemma 4.3.12, the set A_x consists of pairwise convertible terms. Since it is finite and non-empty, confluence yields a term u_x such that $u \rightarrow^*_{\mathcal{R}} u_x$ for all $u \in A_x$. Now define τ as follows:

$$\tau(x) = \begin{cases} u_x & \text{if } x \in \mathsf{Var}(s), \\ \sigma(x) & \text{otherwise.} \end{cases}$$

It is easy to see that this τ satisfies both requirements. □

Stability alone is not enough for ensuring the logicality of confluent, not necessarily weakly normalizing, oriented CTRSs. This is shown in the next example.

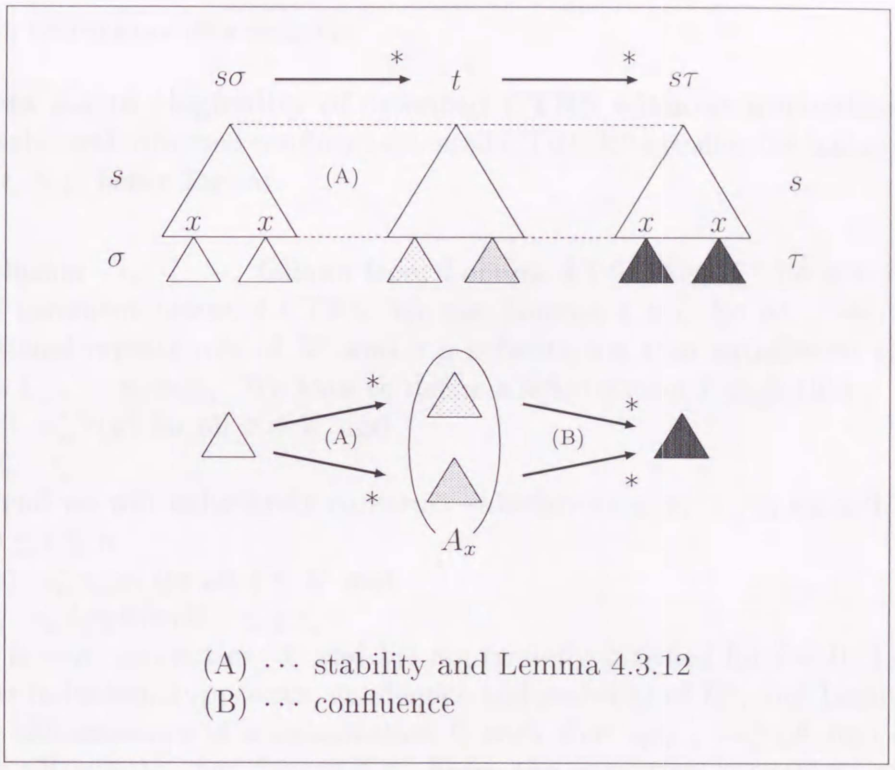


Figure 4.3: Proof of Theorem 4.3.13.

Example 4.3.14
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{rcl} \mathsf{a} \to \mathsf{f(a)} & & \\ \mathsf{g(x)} \to \mathsf{b} & \Leftarrow & \mathsf{f(x)} \approx \mathsf{x} \end{array} \right\}.$$

We have $\mathsf{g(a)} \to_{\mathrm{s}} \mathsf{b}$ since $\mathsf{f(a)}\ {}_{\mathrm{s}}\!\leftarrow \mathsf{a}$. Since there is no term t such that $\mathsf{f}(t) \to_{\mathrm{o}}^{*} t$, the relation $\to_{\mathrm{o}}$ coincides with the rewrite relation induced by the single rewrite rule $\mathsf{a} \to \mathsf{f(a)}$. Hence $\mathcal{R}^{\mathrm{o}}$ is confluent and $\mathsf{g(a)} \leftrightarrow_{\mathrm{o}}^{*} \mathsf{b}$ does not hold. Note that $\mathcal{R}^{\mathrm{o}}$ is stable since variables are trivially stable.

Definition 4.3.15 (well-directedness)
A sequence $s_1 \approx t_1, \ldots, s_n \approx t_n$ of equations is *well-directed* if $\mathsf{Var}(s_j) \cap \mathsf{Var}(t_i) = \varnothing$ for all i, j with $1 \leq j \leq i \leq n$. We say that a CTRS $\mathcal{R}$ is well-directed if the conditional part c is well-directed for every conditional rewrite rule $l \to r \Leftarrow c \in \mathcal{R}$.

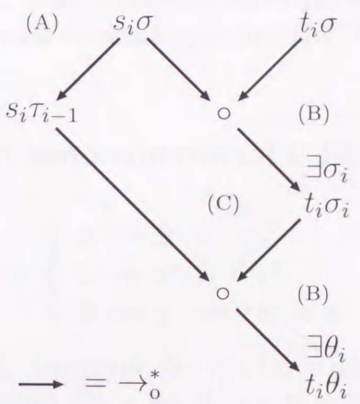
All example CTRSs introduced in this chapter except the one of Example 4.3.14 are well-directed. Normal CTRSs are trivially well-directed. We are now ready for the main theorem of this section.

Theorem 4.3.16 (logicality of oriented CTRS without normalization) Every stable well-directed confluent oriented CTRS $\mathcal{R}^{\mathrm{o}}$ satisfies the inclusion $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{s}} \subseteq \rightarrow_{\mathrm{o}} \circ \downarrow_{\mathrm{o}}$ hence logical.

Proof.
The inclusion $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{s}}$ follows from Lemma 4.1.6. Let $\mathcal{R}^{\mathrm{o}}$ be a stable well-directed confluent oriented CTRS. We use Lemma 4.3.5. So let $l \rightarrow r \Leftarrow c$ be a conditional rewrite rule of $\mathcal{R}^{\mathrm{o}}$ and σ a substitution that satisfies $c\sigma \subseteq \downarrow_{\mathrm{o}}$. Let $c = s_1 \approx t_1, \ldots, s_n \approx t_n$. We have to define a substitution τ such that
(1) $\sigma(x) \rightarrow_{\mathrm{o}}^* \tau(x)$ for all $x \in \mathcal{V}$, and
(2) $c\tau \subseteq \rightarrow_{\mathrm{o}}^*$.
To this end we will inductively construct substitutions $\tau_0, \ldots, \tau_n$ such that for all i with $0 \leq i \leq n$.
(3) $\sigma(x) \rightarrow_{\mathrm{o}}^* \tau_i(x)$ for all $x \in \mathcal{V}$, and
(4) $s_j\tau_i \rightarrow_{\mathrm{o}}^* t_j\tau_i$ for all $1 \leq j \leq i$.
Letting $\tau_0 = \sigma$, properties (3) and (4) are trivially satisfied for $i = 0$. Let $i \geq 1$. From the induction hypothesis, confluence and stability of $\mathcal{R}^{\mathrm{o}}$, and Lemma 4.3.13 we infer the existence of a substitution θ_i such that $s_i\tau_{i-1} \rightarrow_{\mathrm{o}}^* t_i\theta_i$ and $\sigma(x) \rightarrow_{\mathrm{o}}^* \theta_i(x)$ for all $x \in \mathcal{V}$. See Figure 4.4. From the induction hypothesis we obtain $\sigma(x) \rightarrow_{\mathrm{o}}^* \tau_{i-1}(x)$ for all $x \in \mathcal{V}$. Hence confluence yields terms u_x for $x \in \mathcal{V}$ such that $\tau_{i-1}(x) \rightarrow_{\mathrm{o}}^* u_x \;{}_{\mathrm{o}}^*\!\leftarrow \theta_i(x)$. Partition the set of variables $\mathcal{V}$ into $V_1 = \mathsf{Var}(t_i) \cap \bigcup_{1 \leq j < i} \mathsf{Var}(t_j)$, $V_2 = \mathsf{Var}(t_i) \setminus \bigcup_{1 \leq j < i} \mathsf{Var}(t_j)$, and $V_3 = \mathcal{V} \setminus \mathsf{Var}(t_i)$. Now define τ_i as follows:

$$\tau_i(x) = \begin{cases} u_x & \text{if } x \in V_1, \\ \theta_i(x) & \text{if } x \in V_2, \\ \tau_{i-1}(x) & \text{if } x \in V_3. \end{cases}$$

We claim that τ_i has properties (3) and (4). For property (3) we distinguish three cases. If $x \in V_1$ then $\sigma(x) \rightarrow_{\mathrm{o}}^* \tau_{i-1}(x)$ by the induction hypothesis, $\tau_{i-1}(x) \rightarrow_{\mathrm{o}}^* u_x$ by construction of u_x, and $u_x = \tau_i(x)$ by definition of τ_i. If $x \in V_2$ then $\sigma(x) \rightarrow_{\mathrm{o}}^* \theta_i(x)$ by construction of θ_i and $\theta_i(x) = \tau_i(x)$ by definition of τ_i. If $x \in V_3$ then $\sigma(x) \rightarrow_{\mathrm{o}}^* \tau_{i-1}(x)$ by the induction hypothesis and $\tau_{i-1}(x) = \tau_i(x)$ by definition of τ_i. Hence in all cases we obtain the desired $\sigma(x) \rightarrow_{\mathrm{o}}^* \tau_i(x)$. For property (4) we reason as follows. Let $1 \leq j \leq i$. By well-directedness $\mathsf{Var}(s_j) \cap \mathsf{Var}(t_i) = \varnothing$ and thus $\mathsf{Var}(s_j) \subseteq V_3$. Consequently $s_j\tau_i = s_j\tau_{i-1}$ by definition of τ_i. So it remains to show that $s_j\tau_{i-1} \rightarrow_{\mathrm{o}}^* t_j\tau_i$. We distinguish two cases. If $1 \leq j < i$ then $s_j\tau_{i-1} \rightarrow_{\mathrm{o}}^* t_j\tau_{i-1}$ by the induction hypothesis and $t_j\tau_{i-1} \rightarrow_{\mathrm{o}}^* t_j\tau_i$ because $\mathsf{Var}(t_j) \subseteq V_1 \cup V_3$, $\tau_{i-1}(x) \rightarrow_{\mathrm{o}}^* u_x = \tau_i(x)$ for $x \in V_1$, and $\tau_{i-1}(x) = \tau_i(x)$ for $x \in V_3$. If $j = i$ then $s_j\tau_{i-1} \rightarrow_{\mathrm{o}}^* t_j\theta_i$ by construction of θ_i and $t_j\theta_i \rightarrow_{\mathrm{o}}^* t_j\tau_i$ because



(A) ⋯ induction hypothesis
(B) ⋯ Lemma 4.3.13
(C) ⋯ confluence

Figure 4.4: The proof of Theorem 4.3.16.

$\mathsf{Var}(t_j) \subseteq V_1 \cup V_2$, $\theta_i(x) \to_o^* u_x = \tau_i(x)$ for $x \in V_1$, and $\theta_i(x) = \tau_i(x)$ for $x \in V_2$. This concludes the induction step.

Now we define $\tau = \tau_n$. Since properties (3) and (4) for $i = n$ are equivalent to properties (1) and (2), we are done. □

Since normal CTRSs are well-directed Theorem 4.3.4 is a special case of Theorem 4.3.16.

4.4 Equivalence of Different Types of CTRSs

In the previous section we presented restrictions on join and oriented CTRSs which ensure logicality. In this section we present sufficient conditions for logicality of join and oriented CTRSs by imposing conditions on semi-equational CTRSs.

Theorems 4.2.2 and 4.3.3 state that $\to_j = \to_s$ for all confluent join CTRSs and $\to_o = \to_s$ for all confluent normal CTRSs. Hence confluent join and normal CTRSs are not only logical but satisfy all properties of the corresponding semi-equational CTRSs. We develop criteria for semi-equational CTRSs which ensure that $\to_j = \to_s$ and $\to_o = \to_s$.

In the first part of this section we present conditions on semi-equational CTRSs which ensure that $\rightarrow_{\mathrm{j}} = \rightarrow_{\mathrm{s}}$. The following example shows that, unlike the situation for join CTRSs, confluence of semi-equational CTRSs is not sufficient for this equality.

Example 4.4.1 (confluent semi-equational CTRS with $\rightarrow_{\mathrm{j}} \neq \rightarrow_{\mathrm{s}}$)
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{l} \mathsf{a} \rightarrow \mathsf{b} \\ \mathsf{a} \rightarrow \mathsf{c} \\ \mathsf{b} \rightarrow \mathsf{c} \ \Leftarrow \ \mathsf{b} \approx \mathsf{c} \end{array} \right\}.$$

We have $\mathsf{b} \rightarrow_{\mathrm{s}} \mathsf{c}$ since $\mathsf{b} \leftrightarrow^*_{\mathrm{s}} \mathsf{c}$. However, $\mathsf{b} \rightarrow_{\mathrm{j}} \mathsf{c}$ does not hold because the condition $\mathsf{b} \downarrow_{\mathrm{j}} \mathsf{c}$ is not satisfied. Note that $\mathcal{R}^{\mathrm{s}}$ is confluent but $\mathcal{R}^{\mathrm{j}}$ is not.

In the above example we have $\rightarrow_{\mathrm{j}} \neq \rightarrow_{\mathrm{s}}$ but this is not sufficient for the non-logicality of $\mathcal{R}^{\mathrm{j}}$. Actually, the above $\mathcal{R}^{\mathrm{j}}$ is logical because we have $\leftrightarrow^*_{\mathrm{j}} = \leftrightarrow^*_{\mathrm{s}}$. But adding the rewrite rule $\mathsf{d} \rightarrow \mathsf{e} \Leftarrow \mathsf{b} \approx \mathsf{c}$ to $\mathcal{R}$ reveals that confluence of $\mathcal{R}^{\mathrm{s}}$ is not sufficient for logicality of $\mathcal{R}^{\mathrm{j}}$. One way to recover the equality of $\rightarrow_{\mathrm{j}}$ and $\rightarrow_{\mathrm{s}}$, and hence logicality of $\mathcal{R}^{\mathrm{j}}$, is to impose on semi-equational systems, in addition to confluence, the property defined below.

Definition 4.4.2 (decreasingness [DOS88a])
A CTRS $\mathcal{R}$ is *decreasing* if there exists a well-founded order $\succ$ such that
(1) $\rightarrow_{\mathcal{R}} \subseteq \succ$,
(2) $\rhd \subseteq \succ$, and
(3) $l\sigma \succ s\sigma$ and $l\sigma \succ t\sigma$ for all $l \rightarrow r \Leftarrow c \in \mathcal{R}$, $s \approx t$ in c, and $\sigma \in \Sigma(\mathcal{F}, \mathcal{V})$.

Decreasing CTRSs are terminating and, when there are finitely many rewrite rules, have a decidable rewrite relation. Note that the CTRS in the above example is terminating but not decreasing.

Theorem 4.4.3 (sufficient condition for $\rightarrow_{\mathrm{j}} = \rightarrow_{\mathrm{s}}$)
Let $\mathcal{R}$ be a CTRS. If $\mathcal{R}^{\mathrm{s}}$ is confluent and decreasing then $\rightarrow_{\mathrm{j}} = \rightarrow_{\mathrm{s}}$.

Proof.
From Lemma 4.1.6 we know that $\rightarrow_{\mathrm{j}} \subseteq \rightarrow_{\mathrm{s}}$. For the reverse inclusion we show $s \rightarrow_{\mathrm{j}} t$ whenever $s \rightarrow_{\mathrm{s}} t$ by well-founded induction on s with respect to the order $\succ$ which shows that $\mathcal{R}^{\mathrm{s}}$ is decreasing. By definition $s = C[l\sigma]$ and $t = C[r\sigma]$ for some rewrite rule $l \approx r \Leftarrow c \in \mathcal{R}$, context C, and substitution σ such that $c\sigma \subseteq \leftrightarrow^*_{\mathrm{s}}$. Confluence of $\mathcal{R}^{\mathrm{s}}$ yields $c\sigma \subseteq \downarrow_{\mathrm{s}}$. We have $s = C[l\sigma] \succeq l\sigma$ and $l\sigma \succ u\sigma, v\sigma$ for all $u \approx v$ in c by decreasingness. Because $\succ$ contains $\rightarrow_{\mathrm{s}}$ it follows that, for all $u \approx v$ in c, all terms in $u\sigma \downarrow_{\mathrm{s}} v\sigma$ are smaller than s. Hence repeated application of the

induction hypothesis yields $c\sigma \subseteq \downarrow_{\mathrm{j}}$ and therefore $s \to_{\mathrm{j}} t$. □

Dershowitz and Okada [DO90, Theorem 2.3] showed that $\downarrow_{\mathrm{j}} = \downarrow_{\mathrm{s}}$ for every decreasing semi-equational CTRS. A related result by Dershowitz *et al.* [DOS88a, Theorem 3] states that if a semi-equational CTRS $\mathcal{R}^{\mathrm{s}}$ is decreasing then confluence of $\mathcal{R}^{\mathrm{s}}$ implies confluence of the corresponding join CTRS $\mathcal{R}^{\mathrm{j}}$. Theorem 4.4.3 generalizes these two results because if $\mathcal{R}^{\mathrm{s}}$ is confluent and decreasing then $\to_{\mathrm{j}} = \to_{\mathrm{s}}$ and thus $\downarrow_{\mathrm{j}} = \downarrow_{\mathrm{s}}$ and $\mathcal{R}^{\mathrm{j}}$ inherits confluence from $\mathcal{R}^{\mathrm{s}}$.

The proof of Theorem 4.4.3 employs induction on the well-founded order that shows decreasingness. Induction on the level of rewrite steps is also possible, provided we strengthen the confluence requirement.

Definition 4.4.4 (level-confluence [GM86])
A CTRS $\mathcal{R}$ is *level-confluent* if the relation $\to_{\mathcal{R}_k}$ is confluent for every $k \in \mathbb{N}$.

Level-confluence is a fundamental property for ensuring the completeness of conditional narrowing in the presence of extra variables (cf. [GM86] and [MH94]).

Theorem 4.4.5 (sufficient condition for $\to_{\mathrm{j}} = \to_{\mathrm{s}}$)
Let $\mathcal{R}$ be a CTRS. If $\mathcal{R}^{\mathrm{s}}$ is level-confluent then $\to_{\mathrm{j}} = \to_{\mathrm{s}}$.

Proof.
From Lemma 4.1.6 we know that $\to_{\mathrm{j}} \subseteq \to_{\mathrm{s}}$. For the reverse inclusion we show that $\to_{\mathrm{s}_k} \subseteq \to_{\mathrm{j}_k}$ for all $k \in \mathbb{N}$ by induction on k. The base case is trivial. Let $k \geq 1$ and $s \to_{\mathrm{s}_k} t$. By definition there exists a conditional rewrite rule $l \to r \Leftarrow c \in \mathcal{R}$, a substitution σ, and a context C such that $s = C[l\sigma]$, $t = C[r\sigma]$, and $c\sigma \subseteq \leftrightarrow^*_{\mathrm{s}_{k-1}}$. An application of the induction hypothesis yields $c\sigma \subseteq \leftrightarrow^*_{\mathrm{j}_{k-1}}$. Since $\mathcal{R}^{\mathrm{j}}$ is level-confluent we obtain $c\sigma \subseteq \downarrow_{\mathrm{j}_{k-1}}$. Hence $s \to_{\mathrm{j}_k} t$. □

In the second part of this section we present conditions on semi-equational CTRSs which ensure that $\to_{\mathrm{o}} = \to_{\mathrm{s}}$. The following example shows that the conditions in the preceding two theorems are insufficient for ensuring this equality.

Example 4.4.6 (confluent semi-equational CTRS with $\to_{\mathrm{j}} \neq \to_{\mathrm{s}}$)
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{lll} \mathsf{a} \to \mathsf{d} & & \\ \mathsf{b} \to \mathsf{d} & & \\ \mathsf{c} \to \mathsf{d} & \Leftarrow & \mathsf{a} \approx \mathsf{b} \end{array} \right\}.$$

We have $\mathsf{c} \to_{\mathrm{s}} \mathsf{d}$ since $\mathsf{a} \to_{\mathrm{s}} \mathsf{d} \;{}_{\mathrm{s}}\!\leftarrow \mathsf{b}$. However, $\mathsf{c} \to_{\mathrm{o}} \mathsf{d}$ does not hold because the condition $\mathsf{a} \to^*_{\mathrm{o}} \mathsf{b}$ is not satisfied. Note that $\mathcal{R}^{\mathrm{s}}$ is level-confluent and decreasing.

The decreasingness assumption is rather severe because it excludes extra variables in conditional rewrite rules. For the purpose of allowing extra variables, we introduce a much weaker restriction.

Definition 4.4.7 (semi-decreasingness)
A CTRS $\mathcal{R}$ is *semi-decreasing* if there exists a well-founded order $\succ$ satisfying the following three conditions.
(1) $\rightarrow_{\mathcal{R}} \subseteq \succ$.
(2) $\rhd \subseteq \succ$.
(3) If $l \rightarrow r \Leftarrow c \in \mathcal{R}$, $\sigma \in \Sigma(\mathcal{F}, \mathcal{V})$ and $s\sigma \succeq t\sigma$ for all $s \approx t$ in c, then $l\sigma \succ s\sigma$ for all $s \approx t$ in c.

Note that every decreasing CTRS is semi-decreasing, but not vice-versa.
For the equality $\rightarrow_{\mathrm{o}} = \rightarrow_{\mathrm{s}}$ we need to restrict the reducibility of instances of the right-hand sides of conditions in conditional rewrite rules.

Definition 4.4.8 (right-independence)
A CTRS $\mathcal{R}$ is *right-independent* if $\mathsf{Var}(t) \cap \mathsf{Var}(l, r) = \varnothing$ for every $l \approx r \Leftarrow c \in \mathcal{R}$ and every $s \approx t$ in c.

It is easy to see that both normal CTRSs and CTRSs with no extra variables are right-independent.

Theorem 4.4.9 (sufficient condition for $\rightarrow_{\mathrm{o}} = \rightarrow_{\mathrm{s}}$)
Let $\mathcal{R}$ be a CTRS. If $\mathcal{R}^{\mathrm{s}}$ is confluent, semi-decreasing, strongly irreducible, and right-independent then $\rightarrow_{\mathrm{o}} = \rightarrow_{\mathrm{s}}$.

Proof.
From Lemma 4.1.6 we know that $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{s}}$. For the reverse inclusion we show $s \rightarrow_{\mathrm{o}} t$ whenever $s \rightarrow_{\mathrm{s}} t$ by well-founded induction on s with respect to the order $\succ$ which shows that $\mathcal{R}^{\mathrm{s}}$ is semi-decreasing. By definition $s = C[l\sigma]$ and $t = C[r\sigma]$ for some rewrite rule $l \approx r \Leftarrow c \in \mathcal{R}$, context C, and substitution σ that satisfies $c\sigma \subseteq \leftrightarrow_{\mathrm{s}}^*$. Define a substitution τ as follows:

$$\tau(x) = \begin{cases} \sigma(x) & \text{if } x \in \mathsf{Var}(l, r), \\ \sigma(x){\downarrow_{\mathrm{s}}} & \text{otherwise.} \end{cases}$$

Note that $\sigma(x){\downarrow_{\mathrm{s}}}$ is well-defined because $\mathcal{R}^{\mathrm{s}}$ is confluent and weakly normalizing (by semi-decreasingness). We have $c\tau \subseteq \leftrightarrow_{\mathrm{s}}^*$, $l\sigma = l\tau$, and $r\sigma = r\tau$. Because $\mathcal{R}^{\mathrm{s}}$ is confluent, strongly irreducible, and right-independent, we obtain $c\tau \subseteq \rightarrow_{\mathrm{s}}^*$. Since $s = C[l\tau] \succeq l\tau \succ u\tau$ for all $u \approx v$ in c by semi-decreasingness, we obtain $c\tau \subseteq \rightarrow_{\mathrm{o}}^*$ as in the proof of Theorem 4.4.3. Therefore $s \rightarrow_{\mathrm{o}} t$. □

The strong irreducibility requirement is essential because the semi-equational CTRS of Example 4.4.6 is confluent, semi-decreasing, and right-independent. The following example shows that we cannot dispense with right-independence in the preceding theorem.

Example 4.4.10 (necessity of right-independence)
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{rcl} \mathsf{a} \to \mathsf{b} & & \\ \mathsf{f}(\mathsf{x}) \to \mathsf{b} & \Leftarrow & \mathsf{b} \approx \mathsf{x} \end{array} \right\}.$$

We have $\mathsf{f}(\mathsf{a}) \to_{\mathrm{s}} \mathsf{b}$ but not $\mathsf{f}(\mathsf{a}) \to_{\mathrm{o}} \mathsf{b}$. It is not difficult to show that $\mathcal{R}^{\mathrm{s}}$ is (semi-)decreasing, confluent, and strongly irreducible. Right-independence is not satisfied though.

Note that the oriented CTRS $\mathcal{R}^{\mathrm{o}}$ in the above example is logical. It turns out that every oriented CTRS for which the corresponding semi-equational CTRS is confluent, semi-decreasing, and strongly irreducible is logical. In other words, right-independence is not essential for logicality. The proof of this result is an easy consequence of the following theorem, which states that $\mathcal{R}^{\mathrm{o}}$ and $\mathcal{R}^{\mathrm{s}}$ have the same computational power.

Theorem 4.4.11 (sufficient condition for $\to_{\mathrm{o}}^{!} = \to_{\mathrm{s}}^{!}$)
Let $\mathcal{R}$ be a CTRS. If $\mathcal{R}^{\mathrm{s}}$ is confluent, semi-decreasing, and strongly irreducible then $\to_{\mathrm{o}}^{!} = \to_{\mathrm{s}}^{!}$.

Proof.
We show that $t \to_{\mathrm{o}}^{*} t{\downarrow_{\mathrm{s}}}$ for all terms t, from which the equality of $\to_{\mathrm{o}}^{!}$ and $\to_{\mathrm{s}}^{!}$ is easily derived. We use well-founded induction on t with respect to the order $\succ$ which shows that $\mathcal{R}^{\mathrm{s}}$ is semi-decreasing. If t is irreducible we have nothing to prove. Suppose t is reducible. We distinguish two cases.

(1) Suppose a proper subterm s of t is reducible. So $t = C[s]$ for some non-empty context C. We have $t \succ s$ by the subterm property and hence $s \to_{\mathrm{o}}^{+} s{\downarrow_{\mathrm{s}}}$ by the induction hypothesis and the fact that $s \neq s{\downarrow_{\mathrm{s}}}$. Closure under contexts of $\to_{\mathrm{o}}$ yields $t = C[s] \to_{\mathrm{o}}^{+} C[s{\downarrow_{\mathrm{s}}}]$. Because $\succ$ contains $\to_{\mathrm{o}}$, we have $t \succ C[s{\downarrow_{\mathrm{s}}}]$ and thus $C[s{\downarrow_{\mathrm{s}}}] \to_{\mathrm{o}}^{*} C[s{\downarrow_{\mathrm{s}}}]{\downarrow_{\mathrm{s}}}$ by the induction hypothesis. Confluence of $\mathcal{R}^{\mathrm{s}}$ yields $C[s{\downarrow_{\mathrm{s}}}]{\downarrow_{\mathrm{s}}} = t{\downarrow_{\mathrm{s}}}$. Therefore $t \to_{\mathrm{o}}^{+} t{\downarrow_{\mathrm{s}}}$.

(2) If no proper subterm of t is reducible then there must be a rewrite rule $l \to r \Leftarrow c \in \mathcal{R}$ and a substitution σ such that $t = l\sigma$ and $c\sigma \subseteq \leftrightarrow_{\mathrm{s}}^{*}$. Define the substitution τ by $\tau(x) = \sigma(x){\downarrow_{\mathrm{s}}}$ for all variables x. Clearly $c\tau \subseteq \leftrightarrow_{\mathrm{s}}^{*}$. Let $u \approx v$ be an arbitrary equation in c. Confluence and strong irreducibility of $\mathcal{R}^{\mathrm{s}}$ yields $u\tau \to_{\mathrm{s}}^{*} v\tau$ and thus $u\tau \succeq v\tau$. Because $\mathcal{R}^{\mathrm{s}}$ is semi-decreasing we obtain

$l\tau \succ u\tau$. Since $t \rightarrow_{\mathrm{s}}^{*} l\tau$, we have $t \succeq l\tau$ and thus $t \succ u\tau$. Hence we are in a position to apply the induction hypothesis, which yields $u\tau \rightarrow_{\mathrm{o}}^{*} u\tau{\downarrow_{\mathrm{s}}} = v\tau$. We conclude that $c\tau \subseteq \rightarrow_{\mathrm{o}}^{*}$ and therefore $l\tau \rightarrow_{\mathrm{o}} r\tau$. This implies that l cannot be a variable, for otherwise $\tau(l)$ would be reducible and that contradicts the construction of τ. Because proper subterms of t are irreducible, we obtain $t = l\sigma = l\tau$ and consequently $t \succ r\tau$. Another application of the induction hypothesis yields $r\tau \rightarrow_{\mathrm{o}}^{*} r\tau{\downarrow_{\mathrm{s}}}$. Hence $t \rightarrow_{\mathrm{o}}^{+} r\tau{\downarrow_{\mathrm{s}}} = t{\downarrow_{\mathrm{s}}}$ by confluence.

□

Corollary 4.4.12 (sufficient condition for logicality of oriented CTRSs)
Let $\mathcal{R}$ be a CTRS. If $\mathcal{R}^{\mathrm{s}}$ is confluent, semi-decreasing, and strongly irreducible then $\mathcal{R}^{\mathrm{o}}$ is complete and logical.

Employing induction on the level of rewrite steps rather than on the order which comes with semi-decreasingness enables us to weaken the semi-decreasingness requirement in Theorems 4.4.9 and 4.4.11 to level-weak normalization provided we strengthen confluence to level-confluence and strong irreducibility to level-strong irreducibility.

Definition 4.4.13 (level-weak normalization)
A CTRS $\mathcal{R}$ is *level-weakly normalizing* if the relation $\rightarrow_{\mathcal{R}_k}$ is weakly normalizing for every $k \in \mathbb{N}$.

Definition 4.4.14 (level-strong irreducibility)
Let $\mathcal{R}$ be a CTRS. A term t is called *level-strongly irreducible* if, for all $k \in \mathbb{N}$, $t\sigma$ is irreducible with respect to $\rightarrow_{\mathcal{R}_k}$ for every substitution σ that is irreducible with respect to $\rightarrow_{\mathcal{R}_k}$. We say that $\mathcal{R}$ is level-strongly irreducible if every right-hand side t of an equation $s \approx t$ in the conditional part c of a conditional rewrite rule $l \rightarrow r \Leftarrow c$ in $\mathcal{R}$ is level-strongly irreducible.

Note that every level-strongly irreducible term is strongly irreducible but not vice-versa.

Theorem 4.4.15 (sufficient condition for $\rightarrow_{\mathrm{o}} = \rightarrow_{\mathrm{s}}$)
Let $\mathcal{R}$ be a CTRS. If $\mathcal{R}^{\mathrm{s}}$ is level-confluent, level-weakly normalizing, level-strongly irreducible, and right-independent then $\rightarrow_{\mathrm{o}} = \rightarrow_{\mathrm{s}}$.

Proof.
From Lemma 4.1.6 we know that $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{s}}$. For the reverse inclusion we show $\rightarrow_{\mathrm{s}_k} \subseteq \rightarrow_{\mathrm{o}_k}$ by induction on $k \geq 0$. The base case is trivial. Let $k \geq 1$ and $s \rightarrow_{\mathrm{s}_k} t$. By definition there exists a rewrite rule $l \rightarrow r \Leftarrow c \in \mathcal{R}$, a substitution σ, and a

context C such that $s = C[l\sigma]$, $t = C[r\sigma]$, and $c\sigma \subseteq \leftrightarrow^*_{\mathrm{s}_{k-1}}$. Define a substitution τ as follows:

$$\tau(x) = \begin{cases} \sigma(x) & \text{if } x \in \mathsf{Var}(l,r), \\ \sigma(x)\downarrow_{\mathrm{s}_{k-1}} & \text{otherwise.} \end{cases}$$

Note that $\sigma(x)\downarrow_{\mathrm{s}_{k-1}}$ is well-defined because $\mathcal{R}^{\mathrm{s}}$ is level-confluent and level-weakly normalizing. We have $c\tau \subseteq \leftrightarrow^*_{\mathrm{s}_{k-1}}$, $l\sigma = l\tau$, and $r\sigma = r\tau$. Because $\mathcal{R}^{\mathrm{s}}$ is level-confluent, level-strongly irreducible, and right-independent, we obtain $c\tau \subseteq \rightarrow^*_{\mathrm{s}_{k-1}}$. An application of induction hypothesis yields that $c\tau \subseteq \rightarrow^*_{\mathrm{o}_{k-1}}$. Therefore $s \rightarrow_{\mathrm{o}_k} t$.
□

Note the similarity between the proofs of Theorems 4.4.9 and 4.4.15. The next result is the level-version of Theorem 4.4.11.

Theorem 4.4.16 (sufficient condition for $\rightarrow^!_{\mathrm{o}} = \rightarrow^!_{\mathrm{s}}$)
Let $\mathcal{R}$ be a CTRS. If $\mathcal{R}^{\mathrm{s}}$ is level-confluent, level-strongly normalizing, and level-strongly irreducible then $\rightarrow^!_{\mathrm{o}} = \rightarrow^!_{\mathrm{s}}$.

Proof.
We show that $t \rightarrow^*_{\mathrm{o}_k} t\downarrow_{\mathrm{s}_k}$ for all terms t and $k \geq 0$. This implies $\rightarrow^!_{\mathrm{o}_k} = \rightarrow^!_{\mathrm{s}_k}$. From the latter equality the equality of $\rightarrow^!_{\mathrm{o}}$ and $\rightarrow^!_{\mathrm{s}}$ is easily derived. We use induction on k. If $k = 0$ then $t = t\downarrow_{\mathrm{s}_k}$. Suppose $k > 0$. We use the second induction on t with respect to the well-founded order $\succ = (\rightarrow_{\mathrm{s}_k} \cup \rhd)^+$. If t is $\rightarrow_{\mathrm{s}_k}$-irreducible then $t = t\downarrow_{\mathrm{s}_k}$ as before. Suppose t is $\rightarrow_{\mathrm{s}_k}$-reducible. We distinguish two cases.

(1) Suppose a proper subterm s of t is $\rightarrow_{\mathrm{s}_k}$-reducible. So $t = C[s]$ for some non-empty context C. We have $t \succ s$ by definition of $\succ$ and hence $s \rightarrow^+_{\mathrm{o}_k} s\downarrow_{\mathrm{s}_k}$ by the second induction hypothesis and the fact that $s \neq s\downarrow_{\mathrm{s}_k}$. Closure under contexts of $\rightarrow_{\mathrm{s}_k}$ yields $t = C[s] \rightarrow^+_{\mathrm{s}_k} C[s\downarrow_{\mathrm{s}_k}]$. We have $C[s\downarrow_{\mathrm{s}_k}] \rightarrow^*_{\mathrm{o}_k} C[s\downarrow_{\mathrm{s}_k}]\downarrow_{\mathrm{s}_k}$ by the second induction hypothesis. Level-confluence of $\mathcal{R}^{\mathrm{s}}$ yields $C[s\downarrow_{\mathrm{s}_k}]\downarrow_{\mathrm{s}_k} = t\downarrow_{\mathrm{s}_k}$. Therefore $t \rightarrow^+_{\mathrm{o}_k} t\downarrow_{\mathrm{s}_k}$.

(2) If no proper subterm of t is $\rightarrow_{\mathrm{s}_k}$-reducible then there must be a rewrite rule $l \rightarrow r \Leftarrow c \in \mathcal{R}$ and a substitution σ such that $t = l\sigma$ and $c\sigma \subseteq \leftrightarrow^*_{\mathrm{s}_{k-1}}$. Define the substitution τ by $\tau(x) = \sigma(x)\downarrow_{\mathrm{s}_{k-1}}$ for all variables x. Clearly $c\tau \subseteq \leftrightarrow^*_{\mathrm{s}_{k-1}}$. Let $u \approx v$ be an arbitrary equation in c. Level-confluence and level-strong irreducibility of $\mathcal{R}^{\mathrm{s}}$ yields $u\tau \rightarrow^!_{\mathrm{s}_{k-1}} v\tau$. The first induction hypothesis yields $u\tau \rightarrow^*_{\mathrm{o}_{k-1}} (u\tau)\downarrow_{\mathrm{s}_{k-1}} = v\tau$. Hence $l\tau \rightarrow_{\mathrm{o}_k} r\tau$. This implies that l cannot be a variable, for otherwise $\tau(l)$ would be $\rightarrow_{\mathrm{o}_k}$-reducible and that contradicts the construction of τ. Because proper subterms of t are $\rightarrow_{\mathrm{s}_k}$-irreducible, we obtain $t = l\sigma = l\tau$ and consequently $t \succ r\tau$. An application of the second induction hypothesis yields $r\tau \rightarrow^*_{\mathrm{o}_k} r\tau\downarrow_{\mathrm{s}_k}$. Hence $t \rightarrow^*_{\mathrm{o}_k} r\tau\downarrow_{\mathrm{s}_k} = t\downarrow_{\mathrm{s}_k}$.

□

Corollary 4.4.17 (sufficient condition for logicality of oriented CTRSs)
Let $\mathcal{R}$ be a CTRS. If $\mathcal{R}^{\mathrm{s}}$ is level-confluent, level-strongly normalizing, and level-strongly irreducible then $\mathcal{R}^{\mathrm{o}}$ is (level-)complete and logical.

We cannot weaken the level-strong normalization requirement in Theorem 4.4.16 and Corollary 4.4.17 to level-weak normalization. This is shown in the following example.

Example 4.4.18 (necessity of level-strong normalization)
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{lcl} \mathsf{a} \to \mathsf{b} & & \\ \mathsf{a} \to \mathsf{c} & \Leftarrow & \mathsf{a} \approx \mathsf{b} \\ \mathsf{d} \to \mathsf{b} & \Leftarrow & \mathsf{a} \approx \mathsf{b} \\ \mathsf{b} \to \mathrm{x} & \Leftarrow & \mathsf{d} \approx \mathrm{x} \\ \mathsf{e} \to \mathsf{f} & \Leftarrow & \mathsf{b} \approx \mathsf{c} \end{array} \right\}.$$

We have $\mathsf{e} \to_{\mathrm{s}} \mathsf{f}$ but not $\mathsf{e} \leftrightarrow_{\mathrm{o}}^{*} \mathsf{f}$. It is not difficult to show that $\mathcal{R}^{\mathrm{s}}$ is level-confluent, level-strongly irreducible, and (level-)weakly normalizing. Note that $\mathcal{R}^{\mathrm{s}}$ is not level-strongly normalizing as $\mathsf{b} \to_{\mathrm{s}_2} \mathsf{b}$.

The final result of this section, which does not rely on any normalization requirement, is also proved by induction on the level of rewrite steps.

Theorem 4.4.19 (sufficient condition for $\to_{\mathrm{o}} = \to_{\mathrm{s}}$)
Let $\mathcal{R}$ be a CTRS. If $\mathcal{R}^{\mathrm{s}}$ is level-confluent, stable, well-directed, and right-independent then $\to_{\mathrm{o}} = \to_{\mathrm{s}}$.

Proof.
From Lemma 4.1.6 we know that $\to_{\mathrm{o}} \subseteq \to_{\mathrm{s}}$. For the reverse inclusion we show $\to_{\mathrm{s}_k} \subseteq \to_{\mathrm{o}_k}$ by induction on $k \geq 0$. The base case is trivial. Let $k \geq 1$ and $s \to_{\mathrm{s}_k} t$. By definition there exists a rewrite rule $l \to r \Leftarrow c \in \mathcal{R}$, a substitution σ, and a context C such that $s = C[l\sigma]$, $t = C[r\sigma]$, and $c\sigma \subseteq \leftrightarrow_{\mathrm{s}_{k-1}}^{*}$. We are going to define a substitution τ such that
(1) $l\sigma = l\tau$, $r\sigma = r\tau$, and
(2) $c\tau \subseteq \to_{\mathrm{s}_{k-1}}^{*}$.
Let $c = s_1 \approx t_1, \ldots, s_n \approx t_n$. We will inductively define substitutions $\tau_0, \ldots, \tau_n$ such that for all $0 \leq i \leq n$
(3) $\sigma(x) \to_{\mathrm{s}_{k-1}}^{*} \tau_i(x)$ for all $x \in \mathcal{V}$,
(4) $l\sigma = l\tau_i$, $r\sigma = r\tau_i$, and
(5) $s_j\tau_i \to_{\mathrm{s}_{k-1}}^{*} t_j\tau_i$ for all $1 \leq j \leq i$.
Then by defining $\tau = \tau_n$ we obtain properties (1) and (2). From the induction hypothesis and property (2) we know $c\tau \subseteq \to_{\mathrm{o}_{k-1}}^{*}$. Therefore $s = C[l\tau] \to_{\mathrm{o}_k}$

$C[r\tau] = t$ by property (1). Letting $\tau_0 = \sigma$, properties (3)–(5) are trivially satisfied for $i = 0$. Let $i \geq 1$. From the induction hypothesis, level-confluence and stability of $\mathcal{R}^{\mathrm{s}}$, and Lemma 4.3.13 we infer, similar to the proof of Theorem 4.3.16, the existence of a substitution θ_i such that $s_i\tau_{i-1} \to^*_{\mathrm{s}_{k-1}} t_i\theta_i$ and $\sigma(x) \to^*_{\mathrm{s}_{k-1}} \theta_i(x)$ for all $x \in \mathcal{V}$. From the induction hypothesis we obtain $\sigma(x) \to^*_{\mathrm{s}_{k-1}} \tau_{i-1}(x)$ for all $x \in \mathcal{V}$. Hence level-confluence yields for every $x \in \mathcal{V}$ a common $\to^*_{\mathrm{s}_{k-1}}$-reduct u_x of $\tau_{i-1}(x)$ and $\theta_i(x)$. Partition the set of variables $\mathcal{V}$ into $V_1 = \mathsf{Var}(t_i) \cap \bigcup_{1 \leq j < i} \mathsf{Var}(t_j)$, $V_2 = \mathsf{Var}(t_i) \setminus \bigcup_{1 \leq j < i} \mathsf{Var}(t_j)$, and $V_3 = \mathcal{V} \setminus \mathsf{Var}(t_i)$. Now define τ_i as follows:

$$\tau_i(x) = \begin{cases} u_x & \text{if } x \in V_1, \\ \theta_i(x) & \text{if } x \in V_2, \\ \tau_{i-1}(x) & \text{if } x \in V_3. \end{cases}$$

We claim that τ_i has properties (3)–(5). For property (3) we distinguish three cases. If $x \in V_1$ then $\sigma(x) \to^*_{\mathrm{s}_{n-1}} \tau_{i-1}(x)$ by the induction hypothesis, $\tau_{i-1}(x) \to^*_{\mathrm{s}_{k-1}} u_x$ by construction of u_x, and $u_x = \tau_i(x)$ by definition of τ_i. If $x \in V_2$ then $\sigma(x) \to^*_{\mathrm{s}_{k-1}} \theta_i(x)$ by construction of θ_i and $\theta_i(x) = \tau_i(x)$ by definition of τ_i. If $x \in V_3$ then $\sigma(x) \to^*_{\mathrm{s}_{k-1}} \tau_{i-1}(x)$ by the induction hypothesis and $\tau_{i-1}(x) = \tau_i(x)$ by definition of τ_i. Hence in all cases we obtain the desired $\sigma(x) \to^*_{\mathrm{s}_{k-1}} \tau_i(x)$. Next we show property (4). By right-independence neither l nor r contains variables occurring in t_i and hence we have $\mathsf{Var}(l, r) \subseteq V_3$. From the induction hypothesis and definition of τ_i we obtain $l\sigma = l\tau_i$ and $r\sigma = r\tau_i$. For property (5) we reason as follows. Let $1 \leq j \leq i$. By well-directedness $\mathsf{Var}(s_j) \cap \mathsf{Var}(t_i) = \varnothing$ and thus $\mathsf{Var}(s_j) \subseteq V_3$. Consequently $s_j\tau_i = s_j\tau_{i-1}$ by definition of τ_i. So it remains to show that $s_j\tau_{i-1} \to^*_{\mathrm{s}_{k-1}} t_j\tau_i$. We distinguish two cases. If $1 \leq j < i$ then $s_j\tau_{i-1} \to^*_{\mathrm{s}_{k-1}} t_j\tau_{i-1}$ by the induction hypothesis and $t_j\tau_{i-1} \to^*_{\mathrm{s}_{k-1}} t_j\tau_i$ because $\mathsf{Var}(t_j) \subseteq V_1 \cup V_3$, $\tau_{i-1}(x) \to^*_{\mathrm{s}_{k-1}} u_x = \tau_i(x)$ for $x \in V_1$, and $\tau_{i-1}(x) = \tau_i(x)$ for $x \in V_3$. If $j = i$ then $s_j\tau_{i-1} \to^*_{\mathrm{s}_{n-1}} t_j\theta_i$ by construction of θ_i and $t_j\theta_i \to^*_{\mathrm{s}_{k-1}} t_j\tau_i$ because $\mathsf{Var}(t_j) \subseteq V_1 \cup V_2$, $\theta_i(x) \to^*_{\mathrm{s}_{k-1}} u_x = \tau_i(x)$ for $x \in V_1$, and $\theta_i(x) = \tau_i(x)$ for $x \in V_2$. This concludes the induction step. □

Corollary 4.4.20 (sufficient condition for logicality of oriented CTRSs) Let $\mathcal{R}$ be a CTRS. If $\mathcal{R}^{\mathrm{s}}$ is level-confluent, stable, well-directed, and right-independent then $\mathcal{R}^{\mathrm{o}}$ is logical.

Theorem 4.4.19 does not hold if we drop the right-independence requirement. Even stronger, in contrast to the situation in Corollary 4.4.12, without right-independence we lose logicality. This follows from Example 4.4.10 since $\mathcal{R}^{\mathrm{s}}$ is level-confluent, stable, and well-directed.

4.5 Systematic Analysis of Different Types of Relations

In the previous sections we investigated various sufficient conditions for logicality. For a better understanding of the principle which is hidden in the proofs of logicality results, we make a systematic comparison between various pairs of relations induced by different types of CTRSs. We start our analysis with an extension of the soundness result.

Theorem 4.5.1 (soundness results)
Let $\mathcal{R}$ be a CTRS.
(a) $\to_{\mathrm{o}} \subseteq \to_{\mathrm{j}} \subseteq \to_{\mathrm{s}}$.
(b) $\forall k \in \mathbb{N}\ \to_{\mathrm{o}_k} \subseteq \to_{\mathrm{j}_k} \subseteq \to_{\mathrm{s}_k}$.

Proof.
Part (a) is already proved in Lemma 4.1.6. Part (b) is easily proved by induction on k. Similarly to the proof of (a), the inclusion $\to \subseteq \downarrow \subseteq \leftrightarrow^*$ is applied. □

The following theorem gives sufficient conditions for the equality of the two rewrite relations induced by a semi-equational system and the corresponding join system. The following abbreviations are used in the theorem: CR (confluence), D (decreasingness), and LCR (level-confluence).

Theorem 4.5.2 (relationships between $\mathcal{R}^{\mathrm{s}}$ and $\mathcal{R}^{\mathrm{j}}$)
Let $\mathcal{R}$ be a CTRS.
(A) $\mathcal{R}^{\mathrm{s}}$:CR,D $\vee$ $\mathcal{R}^{\mathrm{j}}$:CR $\Rightarrow$ $\to_{\mathrm{s}} = \to_{\mathrm{j}}$.
(B) $\mathcal{R}^{\mathrm{s}}$:LCR $\vee$ $\mathcal{R}^{\mathrm{j}}$:LCR $\Rightarrow$ $\forall k \in \mathbb{N}\ \to_{\mathrm{s}_k} = \to_{\mathrm{j}_k}$.

Proof.
Statement (A) is already proved. We already know (B) for semi-equational systems. The result for join systems is easily proved by induction on k. □

Guided by Table 4.1, we prove various inclusions. Table 4.1 summarizes all the sufficient conditions for ensuring various combinations of inclusions. We have already discussed (a) and (b) in Theorem 4.5.1. Theorem 4.5.2 provides the results (A) and (B). It remains to investigate the relationships between oriented systems and other types of systems, which are indicated by (1) – (10).

As we observed in Sections 4.3 and 4.4, the key idea for proving that a relation in an oriented system can simulate other types of CTRS is to find a substitution such that the conditions are satisfied by unidirectional rewriting. The following lemma abstracts the pattern of finding such a substitution.

$\subseteq$	$\to_s$	$\to_j$	$\to_o$
$\to_s$	—	(A)	(3)
$\to_j$	(a)	—	(1)
$\to_o$	(a)	(a)	—

$\subseteq$	$\to_s^+$	$\to_j^+$	$\to_o^+$
$\to_s$	—	(A)	(7)
$\to_j$	(a)	—	(5)
$\to_o$	(a)	(a)	—

$\subseteq$	$\downarrow_s$	$\downarrow_j$	$\downarrow_o$
$\to_s$	—	(A)	(9)
$\to_j$	(a)	—	(9)
$\to_o$	(a)	(a)	—

$\subseteq$	$\to_{s_k}$	$\to_{j_k}$	$\to_{o_k}$
$\to_{s_k}$	—	(B)	(4)
$\to_{j_k}$	(b)	—	(2)
$\to_{o_k}$	(b)	(b)	—

$\subseteq$	$\to_{s_k}^+$	$\to_{j_k}^+$	$\to_{o_k}^+$
$\to_{s_k}$	—	(B)	(8)
$\to_{j_k}$	(b)	—	(6)
$\to_{o_k}$	(b)	(b)	—

$\subseteq$	$\downarrow_{s_k}$	$\downarrow_{j_k}$	$\downarrow_{o_k}$
$\to_{s_k}$	—	(B)	(10)
$\to_{j_k}$	(b)	—	(10)
$\to_{o_k}$	(b)	(b)	—

Table 4.1: Comparing different types of rewrite relations.

Lemma 4.5.3 (orientation lemma)
Let $\to$ be a confluent ARS over terms, $c = s_1 \approx t_1, \ldots, s_n \approx t_n$ a sequence of equations, V a set of variables satisfying $\mathsf{Var}(t_i) \cap V = \varnothing$ for all i with $1 \leq i \leq n$, and σ a substitution such that $c\sigma \subseteq \leftrightarrow^*$. If either

- $\to$ is weakly normalizing and t_i is strongly irreducible for all i with $1 \leq i \leq n$ or
- t_i is stable for all i with $1 \leq i \leq n$ and c is well-directed.

then there exists a substitution τ with the following three properties.

(1) $c\tau \subseteq \to^*$.
(2) $\sigma(x) \to^* \tau(x)$ for all $x \in \mathcal{V}$.
(3) $\sigma(x) = \tau(x)$ for all $x \in V$.

Proof.
The result based on weak normalization and strong irreducibility is obtained similarly to the proofs of Theorems 4.3.7, 4.4.9, and 4.4.15. The result based on stability and the restriction on variable distribution is shown similarly to the proofs of Theorems 4.3.16 and 4.4.19. □

We introduce sufficient conditions that enable us to apply the orientation lemma.

Definition 4.5.4 (unidirectionality)
A CTRS $\mathcal{R}$ is called *unidirectional* if either

- $\mathcal{R}$ is weakly normalizing and strongly irreducible, or
- $\mathcal{R}$ is stable and well-directed.

A CTRS $\mathcal{R}$ is called *level-unidirectional* if either

- $\mathcal{R}$ is level-weakly normalizing and level-strongly irreducible, or
- $\mathcal{R}$ is stable and well-directed.

Note that we do not use the level version of stability for the definition of level-unidirectionality because stability is equivalent to level-stability as we will observe in Section 5.3.

The orientation lemma is applied to right-independent CTRSs by taking $V = \mathsf{Var}(l, r)$ for every conditional rewrite rule $l \to r \Leftarrow c \in \mathcal{R}$. For CTRSs without right-independence restriction, we take $V = \varnothing$ instead. A weaker restriction on the right-hand side of the conditions is weak right-independence defined below. For weakly right-independent CTRSs we take $V = \mathsf{Var}(l)$ in the orientation lemma.

Definition 4.5.5 (weak right-independence)
A CTRS $\mathcal{R}$ is *weakly right-independent* if $\mathsf{Var}(t) \cap \mathsf{Var}(r) = \varnothing$ for every $l \approx r \Leftarrow c \in \mathcal{R}$ and $s \approx t$ in c.

In the next theorem the following abbreviations are used: CR (confluence), LCR (level-confluence), SD (semi-decreasingness), UD (unidirectionality), LUD (level-unidirectionality), N (Normal), RI (right-independence), WRI (weak right-independence).

Theorem 4.5.6 (comparing different types of rewrite relations)
(1) $\mathcal{R}$:N $\vee$ $\mathcal{R}^{\mathrm{j}}$:CR,SD,UD,RI $\vee$ $\mathcal{R}^{\mathrm{o}}$:CR,UD,RI $\Rightarrow$ $\to_{\mathrm{j}} = \to_{\mathrm{o}}$.
(2) $\mathcal{R}$:N $\vee$ $\mathcal{R}^{\mathrm{j}}$:LCR,LUD,RI $\vee$ $\mathcal{R}^{\mathrm{o}}$:LCR,LUD,RI $\Rightarrow$ $\forall k \in \mathbb{N}\ \to_{\mathrm{j}_k} = \to_{\mathrm{o}_k}$.
(3) $\mathcal{R}^{\mathrm{s}}$:CR,SD,UD,RI $\vee$ $\mathcal{R}^{\mathrm{o}}$:CR,UD,RI $\Rightarrow$ $\to_{\mathrm{s}} = \to_{\mathrm{o}}$.
(4) $\mathcal{R}^{\mathrm{s}}$:LCR,LUD,RI $\vee$ $\mathcal{R}^{\mathrm{o}}$:LCR,LUD,RI $\Rightarrow$ $\forall k \in \mathbb{N}\ \to_{\mathrm{s}_k} = \to_{\mathrm{o}_k}$.
(5) $\mathcal{R}^{\mathrm{j}}$:CR,SD,UD,WRI $\vee$ $\mathcal{R}^{\mathrm{o}}$:CR,UD,WRI $\Rightarrow$ $\to_{\mathrm{o}} \subseteq \to_{\mathrm{j}} \subseteq \to_{\mathrm{o}}^{+}$.
(6) $\mathcal{R}^{\mathrm{j}}$:LCR,LUD,WRI $\vee$ $\mathcal{R}^{\mathrm{o}}$:LCR,LUD,WRI $\Rightarrow$ $\forall k \in \mathbb{N}\ \to_{\mathrm{o}_k} \subseteq \to_{\mathrm{j}_k} \subseteq \to_{\mathrm{o}_k}^{+}$.
(7) $\mathcal{R}^{\mathrm{s}}$:CR,SD,UD,WRI $\vee$ $\mathcal{R}^{\mathrm{o}}$:CR,UD,WRI $\Rightarrow$ $\to_{\mathrm{o}} \subseteq \to_{\mathrm{s}} \subseteq \to_{\mathrm{o}}^{+}$.
(8) $\mathcal{R}^{\mathrm{s}}$:LCR,LUD,WRI $\vee$ $\mathcal{R}^{\mathrm{o}}$:LCR,LUD,WRI $\Rightarrow$ $\forall k \in \mathbb{N}\ \to_{\mathrm{o}_k} \subseteq \to_{\mathrm{s}_k} \subseteq \to_{\mathrm{o}_k}^{+}$.
(9) $\mathcal{R}^{\mathrm{o}}$:CR,UD $\Rightarrow$ $\to_{\mathrm{o}} \subseteq \to_{\mathrm{j}} \subseteq \to_{\mathrm{s}} \subseteq \to_{\mathrm{o}} \circ \downarrow_{\mathrm{o}}$.
(10) $\mathcal{R}^{\mathrm{o}}$:LCR,LUD $\Rightarrow$ $\forall k \in \mathbb{N}\ \to_{\mathrm{o}_k} \subseteq \to_{\mathrm{j}_k} \subseteq \to_{\mathrm{s}_k} \subseteq \to_{\mathrm{o}_k} \circ \downarrow_{\mathrm{o}_k}$.

Sufficient conditions given in Theorems 4.5.1, 4.5.2, and 4.5.6 cover all the results which compare rewrite relations except Theorem 4.4.16 and Theorem 4.4.11. The results which compare a relation induced by a semi-equational system with a relation induced by another type of system imply logicality. So we obtain a new logicality result from statement (10). Moreover, the logicality result obtained from statement (8) generalizes Corollary 4.4.20 since weak right-independence follows from right-independence. Theorem 4.5.6 is not only useful for logicality but also for proving that a certain (level-)property is preserved between different types of CTRSs.

We conclude this section by making a comparison between the logicality results for join and for oriented systems. With every join CTRS $\mathcal{R}$ we can associate a corresponding oriented CTRS using the following transformation.

Definition 4.5.7 (transformation from join CTRSs to oriented CTRSs) Every CTRS $\mathcal{R}$ is transformed into $\mathsf{O}(\mathcal{R})$ as follows:

$$l \to r \Leftarrow s_1 \approx t_1, \ldots, s_n \approx t_n \in \mathcal{R}$$

if and only if

$$l \to r \Leftarrow s_1 \approx x_1, t_1 \approx x_1, \ldots, s_n \approx x_n, t_n \approx x_n \in \mathsf{O}(\mathcal{R})$$

where $x_1, \ldots, x_n$ are fresh and pairwise distinct variables.

Lemma 4.5.8 (transformation from join CTRSs to oriented CTRSs) Let $\mathcal{R}$ be a CTRS.
(1) $\to_{\mathcal{R}^{\mathrm{j}}} = \to_{\mathsf{O}(\mathcal{R})^{\mathrm{o}}}$.
(2) $\forall k \in \mathbb{N} \to_{\mathcal{R}^{\mathrm{j}}_k} = \to_{\mathsf{O}(\mathcal{R})^{\mathrm{o}}_k}$.

Proof.
Property (1) is an easy consequence of property (2), which is easily proved by induction on the level k. □

By using this transformation, every join CTRS is transformed into a stable, well-directed, and right-independent oriented CTRS. Because confluence is preserved under the transformation by the above lemma, Theorem 4.3.16 can be considered as a generalization of Corollary 4.2.3. In the same fashion, Theorems 4.4.9 and 4.4.19 generalize Theorems 4.4.3 and 4.4.5, respectively.

Chapter 5

Ensuring Sufficient Conditions for Logicality

In order to give sufficient conditions for logicality, various properties are used and newly introduced in the previous chapter. The purpose of this chapter is to study how to ensure those properties. We also study the difference between properties which are used for a similar purpose.

Section 5.1 deals with properties related to normalization, such as decreasingness, semi-decreasingness, termination, and weak normalization. We also review how to ensure decidability of the rewrite relation induced by a CTRS. In Section 5.2 we review how to ensure confluence and level-confluence. In Section 5.3 we study properties which are used to restrict the reducibility of a term. Sufficient syntactic criteria for stability and strong irreducibility are discussed.

5.1 Termination and Decidability of Rewrite Relations

Existing techniques for proving termination of TRSs are naturally applied to CTRSs by considering their unconditional versions.

Definition 5.1.1 (unconditional version of a CTRS)
The *unconditional version* of a CTRS $(\mathcal{F}, \mathcal{R})$ is the unconditional TRS $(\mathcal{F}, \mathcal{R}_u)$ where $\mathcal{R}_u$ is the set of unconditional rewrite rules obtained from $\mathcal{R}$ by dropping all the conditions:

$$\mathcal{R}_u = \{\, l \to r \mid l \to r \Leftarrow c \in \mathcal{R} \,\}.$$

The termination of a given CTRS $\mathcal{R}$ follows from the termination of its unconditional version $\mathcal{R}_u$ because the inclusion $\rightarrow_{\mathcal{R}} \subseteq \rightarrow_{\mathcal{R}_u}$ is always satisfied. For survey papers of existing techniques for proving termination of TRSs, we refer to [Der87] and [Ste95].

Conditional rewriting is more complicated than unconditional rewriting because the satisfiability of the conditions is checked recursively. Even if the rewrite relation of a CTRS is terminating, the evaluation procedure may not terminate because of the recursive evaluation of the conditions. Kaplan showed that, even for finite complete CTRSs without extra variables, it is undecidable whether a term is reducible or not (cf. [Kap84]). In order to avoid infinite recursive evaluation, several properties which are stronger than termination have been proposed. Decreasingness (see Definition 4.4.2) is an example of such property for join CTRSs.

Theorem 5.1.2 (decidability by decreasingness [DOS88a])
Let $\mathcal{R}$ be a join CTRS consisting of finitely many conditional rewrite rules. If $\mathcal{R}$ is decreasing, then it has the following properties.
(1) $\mathcal{R}$ is terminating.
(2) The rewrite relation $\rightarrow_{\mathcal{R}}$ is decidable.

Simplifyingness of Kaplan [Kap87] and reductivity of Jouannaud and Waldmann [JW86] are sufficient conditions for decreasingness. Note that the decreasingness of a CTRS relies on the rewrite relation. Hence it is affected by the interpretation of the conditions as shown in the following example.

Example 5.1.3 (decreasingness of different types of CTRSs)
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{lll} \mathsf{a} \rightarrow \mathsf{b} & & \\ \mathsf{c} \rightarrow \mathsf{c} & \Leftarrow & \mathsf{b} \approx \mathsf{a} \end{array} \right\}.$$

The decreasingness of the oriented CTRS $\mathcal{R}^{\mathrm{o}}$ can be shown by means of the well-founded relation that satisfies $\mathsf{a} \succ \mathsf{b}$, $\mathsf{c} \succ \mathsf{b}$, and $\mathsf{c} \succ \mathsf{a}$. However, both the semi-equational CTRS $\mathcal{R}^{\mathrm{s}}$ and the join CTRS $\mathcal{R}^{\mathrm{j}}$ are not decreasing because they are not terminating.

Observe that proving termination by unconditional version is not powerful enough to show the termination of $\mathcal{R}^{\mathrm{o}}$ in the above example.

Lemma 5.1.4 (decreasingness of different types of CTRSs)
Let $\mathcal{R}$ be a CTRS.
(1) If $\mathcal{R}^{\mathrm{s}}$ is decreasing then $\mathcal{R}^{\mathrm{j}}$ is decreasing.
(2) If $\mathcal{R}^{\mathrm{j}}$ is decreasing then $\mathcal{R}^{\mathrm{o}}$ is decreasing.

Proof.
By the soundness result $\rightarrow_{\mathrm{o}} \subseteq \rightarrow_{\mathrm{j}} \subseteq \rightarrow_{\mathrm{s}}$ (Lemma 4.1.6) and the definition of decreasingness (Definition 4.4.2). □

It is easy to see that a CTRS with extra variables in its conditional rewrite rules is non-decreasing. Ganzinger introduced quasi-reductive CTRSs in order to translate order-sorted specifications into oriented CTRSs with extra variables. In quasi-reductive CTRSs we can avoid infinite recursive evaluation.

Definition 5.1.5 (quasi-reductivity [Gan91])
A CTRS $\mathcal{R}$ is called *quasi-reductive* if there exists a well-founded order $\succ$ such that $\succ$ is a rewrite relation with the following properties:
(1) $l\sigma \ (\succ \cup \rhd)^{+} \ s_i\sigma$ for all $1 \leq i \leq n$ such that $s_j\sigma \succ^{=} t_j\sigma$ for all $1 \leq j < i$,
(2) $l\sigma \succ r\sigma$ if $s_i\sigma \succ^{=} t_i\sigma$ for all $1 \leq i \leq n$
for every conditional rewrite rule $l \rightarrow r \Leftarrow s_1 \approx t_1, \ldots, s_n \approx t_n \in \mathcal{R}$ and $\sigma \in \Sigma(\mathcal{F}, \mathcal{V})$.

Lemma 5.1.6 (property of quasi-reductive CTRSs)
If a CTRS $\mathcal{R}$ is quasi-reductive with respect to a well-founded order $\succ$, then the inclusion $\rightarrow_{\mathcal{R}} \subseteq \succ$ holds and hence $\mathcal{R}$ is terminating.

Proof.
We can prove that $s \rightarrow_{\mathcal{R}}^{+} t$ implies $s \succ t$ for all terms s, t by induction on s with respect to $\succ$. □

For the purpose of ensuring the decidability of the rewrite relation induced by a quasi-reductive CTRS, Ganzinger introduced deterministic CTRSs, in which restricted use of extra variables is allowed.

Definition 5.1.7 (deterministic CTRS [Gan91])
A CTRS $\mathcal{R}$ is called *deterministic* if every $l \rightarrow r \Leftarrow s_1 \approx t_1, \ldots, s_n \approx t_n \in \mathcal{R}$ is a conditional rewrite rule of type 3 and

$$\mathsf{Var}(s_i) \subseteq \mathsf{Var}(l) \cup \bigcup_{j=1}^{i-1} \mathsf{Var}(s_j, t_j)$$

for all i with $1 \leq i \leq n$.

In [Gan91] and [ALS94] it is shown that deterministic quasi-reductive CTRSs have a decidable rewrite relation when there are finitely many rewrite rules.

Theorem 5.1.8 (decidability by quasi-reductivity [Gan91])
Let $\mathcal{R}$ be an oriented CTRS consisting of finitely many conditional rewrite rules. If $\mathcal{R}$ is deterministic and quasi-reductive, then the rewrite relation $\rightarrow_{\mathcal{R}}$ is decidable.

The following lemma summarizes the relationships between decreasingness, quasi-reductivity, semi-decreasingness, and termination.

Lemma 5.1.9 (semi-decreasingness and related properties)
(1) Every decreasing CTRS is semi-decreasing.
(2) Every quasi-reductive CTRS is semi-decreasing.
(3) Every semi-decreasing CTRS is terminating.
(4) If a join CTRS $\mathcal{R}^{\mathsf{j}}$ is decreasing then the transformed oriented CTRS $\mathsf{O}(\mathcal{R})^{\mathsf{o}}$ is semi-decreasing.

Proof.
(1) and (2) are obvious by definition. (3) follows from Lemma 5.1.6. (4) is a consequence of Lemma 4.5.8(1). □

The remaining properties of interest in this section are strong normalization (termination), weak normalization, and the corresponding level-properties. In Section 2.2 we already observed that strong normalization implies weak normalization and hence level-strong normalization implies level-weak normalization. The next lemma clarifies the relationship between strong normalization and level-strong normalization.

Lemma 5.1.10 (strong normalization and level-strong normalization)
Every strongly normalizing CTRS is level-strongly normalizing.

Proof.
Let $\mathcal{R}$ be a strongly normalizing CTRS. For a proof by contradiction, suppose there exists a level $k \in \mathbb{N}$ such that the relation $\rightarrow_{\mathcal{R}_k}$ admits an infinite rewrite sequence. Since the inclusion $\rightarrow_{\mathcal{R}_k} \subseteq \rightarrow_{\mathcal{R}}$ holds by definition, $\rightarrow_{\mathcal{R}}$ also admits an infinite rewrite sequence. This contradicts the strong normalization of $\mathcal{R}$. □

Level-strongly normalizing CTRSs need not be strong normalizing as shown in the following example from [MH94].

Example 5.1.11 (strong normalization and level-strong normalization)
The CTRS

$$\mathcal{R} = \{ \ \mathsf{f(x)} \rightarrow \mathsf{f(g(x))} \ \Leftarrow \ \mathsf{f(a)} \approx \mathsf{f(x)} \ \}$$

is not strongly normalizing because it admits the infinite rewrite sequence $\mathsf{f(a)} \rightarrow_{\mathcal{R}_1}$

$\mathsf{f}(\mathsf{g}(\mathsf{a})) \rightarrow_{\mathcal{R}_2} \mathsf{f}(\mathsf{g}(\mathsf{g}(\mathsf{a}))) \rightarrow_{\mathcal{R}_3} \cdots$. However, the relations $\rightarrow_{\mathcal{R}_k}$ are strongly normalizing for $k \in \mathbb{N}$.

The relationships between properties concerned with normalization is summarized in Figure 5.1. Here we use the following abbreviations: D (decreasingness), QR (quasi-reductivity), SD (semi-decreasingness), SN (strong normalization), WN (weak normalization), LSN (level-strong normalization), LWN (level-weak normalization).

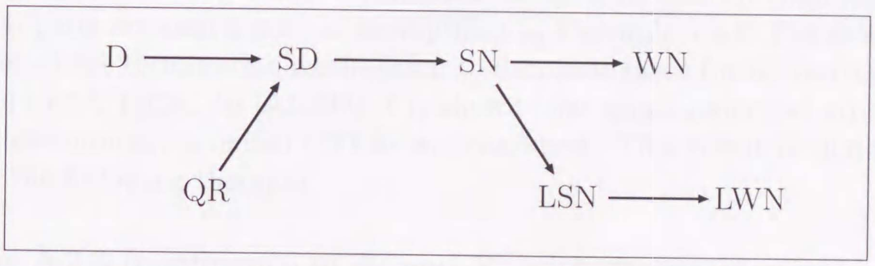


Figure 5.1: Relationships between properties concerned with normalization.

5.2 Confluence and Level-Confluence

In this section we study how to ensure confluence and level-confluence of CTRSs. We first review confluence criteria which rely on termination assumption. The notion of critical pair is important to ensure the confluence of CTRSs.

Definition 5.2.1 (critical pair)
Let $l \rightarrow r \Leftarrow s_1 \approx t_1, \ldots, s_n \approx t_n$ and $l' \rightarrow r' \Leftarrow s'_1 \approx t'_1, \ldots, s'_{n'} \approx t'_{n'}$ be two conditional rewrite rules in a CTRS $\mathcal{R}$ such that they do not share variables (after renaming). Suppose there exists a position $p \in \mathsf{Pos}_{\mathcal{F}}(l)$ satisfying the following properties:

- if the two rules are renamed version of the same rule in $\mathcal{R}$ then $p \neq \epsilon$,
- $l_{|p}$ and l' are unifiable by a most general unifier σ.

We call the conditional equation

$$l\sigma[r'\sigma] \approx r\sigma \Leftarrow s_1\sigma \approx t_1\sigma, \ldots, s_n\sigma \approx t_n\sigma, s'_1\sigma \approx t'_1\sigma, \ldots, s'_{n'}\sigma \approx t'_{n'}\sigma$$

a *conditional critical pair* of $\mathcal{R}$. A conditional critical pair $s \approx t \Leftarrow c$ of $\mathcal{R}$ is *convergent* if $s\sigma \downarrow_{\mathcal{R}} t\sigma$ for all $\sigma \in \Sigma(\mathcal{F}, \mathcal{V})$ such that $\mathcal{R} \vdash c\sigma$.

For terminating unconditional TRSs, confluence is guaranteed by the convergence of all critical pairs [KB70]. Dershowitz and Plaisted generalized this result to CTRSs.

Theorem 5.2.2 (confluence of $\mathcal{R}^{s}$ with termination [DP88])
A terminating semi-equational 2-CTRS is confluent if all its conditional critical pairs are convergent.

In join and oriented systems, termination is not sufficient for confluence even if all critical pairs are convergent as exemplified in Example 4.4.6. Dershowitz *et al.* strengthened the termination assumption to decreasingness for recovering the confluence of join CTRSs. In [ALS94] it is shown that quasi-reductive strongly irreducible deterministic oriented CTRSs are confluent. This result is slightly generalized in the following theorem.

Theorem 5.2.3 (confluence of $\mathcal{R}^{j}$ and $\mathcal{R}^{o}$ with decreasingness)
(1) A decreasing join CTRS is confluent if all its conditional critical pairs are convergent [DOS88a].
(2) A semi-decreasing strongly irreducible deterministic oriented CTRS is confluent if all its conditional critical pairs are convergent.

Note that statement (1) can be considered as a special case of (2) because every decreasing join CTRS $\mathcal{R}$ can be transformed into the equivalent oriented CTRS $\mathsf{O}(\mathcal{R})$, which satisfies all the requirements in (2). In [DOS88b] sufficient conditions for the confluence of terminating join CTRSs are presented.

Next we review confluence criteria which rely on orthogonality instead of termination.

Definition 5.2.4 (orthogonality)
A CTRS $\mathcal{R}$ is *left-linear* if, for every conditional rewrite rule $l \to r \Leftarrow c \in \mathcal{R}$, the left-hand side l is a linear term. A CTRS is called *orthogonal* if it is left-linear and has no conditional critical pairs.

Orthogonality was first applied for ensuring confluence of unconditional TRSs in [Ros73]. The following theorem is its extension by O'Donnell to CTRSs.

Theorem 5.2.5 (confluence of $\mathcal{R}^{s}$ by orthogonality [O'D77])
Every orthogonal semi-equational 2-CTRS is confluent.

Bergstra and Klop showed that orthogonality is not sufficient for the confluence of non-semi-equational CTRSs.

Example 5.2.6 (non-confluent orthogonal CTRSs [BK86])
The CTRS

$$\mathcal{R} = \left\{ \begin{array}{lcl} \mathsf{a} \to \mathsf{f}(\mathsf{a}) & & \\ \mathsf{f}(\mathrm{x}) \to \mathsf{b} & \Leftarrow & \mathrm{x} \approx \mathsf{f}(\mathrm{x}) \end{array} \right\}$$

is left-linear and has no critical pairs, hence orthogonal. We have $\mathsf{b} \;{}_{\mathcal{R}}\!\!\leftarrow \mathsf{f}(\mathsf{f}(\mathsf{a})) \to_{\mathcal{R}} \mathsf{f}(\mathsf{b})$ but neither $\mathsf{b} \downarrow_{\mathsf{j}} \mathsf{f}(\mathsf{b})$ nor $\mathsf{b} \downarrow_{\mathsf{o}} \mathsf{f}(\mathsf{b})$ is satisfied. Therefore both $\mathcal{R}^{\mathsf{j}}$ and $\mathcal{R}^{\mathsf{o}}$ are not confluent.

Suzuki *et al.* [SMI95] introduced the notions of proper orientation and right-stability to provide syntactic criteria for ensuring the level-confluence of oriented CTRSs. Note that the definition of right-stability here is slightly modified by using stability.

Definition 5.2.7 (proper orientation and right-stability)
An oriented CTRS $\mathcal{R}$ is called *properly oriented* if every conditional rewrite rule $l \to r \Leftarrow s_1 \approx t_1, \ldots, s_n \approx t_n \in \mathcal{R}$ with $\mathsf{Var}(r) \not\subseteq \mathsf{Var}(l)$ satisfies

$$\mathsf{Var}(s_i) \subseteq \mathsf{Var}(l) \cup \bigcup_{j=1}^{i-1} \mathsf{Var}(s_j, t_j)$$

for all $1 \leq i \leq n$. An oriented CTRS is called *right-stable* if every conditional rewrite rule $l \to r \Leftarrow s_1 \approx t_1, \ldots, s_n \approx t_n$ satisfies

$$\Big(\mathsf{Var}(l) \cup \bigcup_{j=1}^{i-1} \mathsf{Var}(s_j, t_j) \cup \mathsf{Var}(s_i)\Big) \cap \mathsf{Var}(t_i) = \varnothing$$

and t_i is a linear stable term, for all $1 \leq i \leq n$.

Theorem 5.2.8 (confluence of $\mathcal{R}^{\mathsf{o}}$ by orthogonality)
(1) Every orthogonal normal oriented 2-CTRS is level-confluent [BK86].
(2) Every orthogonal properly oriented right-stable 3-CTRS is level-confluent [SMI95].

Note that (1) is a special case of (2) because every normal oriented 2-CTRS is a properly oriented right-stable 3-CTRS.

Another way to ensure the confluence of CTRSs is modularity. If a property of CTRS is modular we can prove the property by a divide and conquer approach.

Modularity of confluence was first proved by Toyama [Toy87]. Middeldorp extended Toyama's result to CTRSs.

Theorem 5.2.9 (modularity of confluence [Mid90])
Let $(\mathcal{R}_1, \mathcal{F}_1)$ and $(\mathcal{R}_2, \mathcal{F}_2)$ be both semi-equational 2-CTRSs or both join 2-CTRSs and suppose $\mathcal{F}_1 \cap \mathcal{F}_2 = \varnothing$. Then:

$$(\mathcal{R}_1, \mathcal{F}_1) \text{ and } (\mathcal{R}_2, \mathcal{F}_2) \text{ are confluent } \iff (\mathcal{R}_1 \cup \mathcal{R}_2, \mathcal{F}_1 \cup \mathcal{F}_2) \text{ is confluent.}$$

Yamada *et al.* proved that level-confluence is a modular property of CTRSs.

Theorem 5.2.10 (modularity of level-confluence [YMI95])
Let $(\mathcal{R}_1, \mathcal{F}_1)$ and $(\mathcal{R}_2, \mathcal{F}_2)$ be join 2-CTRSs and suppose $\mathcal{F}_1 \cap \mathcal{F}_2 = \varnothing$. Then:

$$(\mathcal{R}_1, \mathcal{F}_1) \text{ and } (\mathcal{R}_2, \mathcal{F}_2) \text{ are level-confluent } \iff (\mathcal{R}_1 \cup \mathcal{R}_2, \mathcal{F}_1 \cup \mathcal{F}_2) \text{ is level-confluent.}$$

5.3 Stability and Strong Irreducibility

In this section we study relationships between various properties which are used to restrict the reducibility of a term.

Because stability depends on the rewrite relation associated with a CTRS, it is an undecidable property. As a sufficient syntactic condition for stability, we introduce strong stability.

Definition 5.3.1 (linearization of a term)
A term t is called a *linearization* of s if t is linear and $t\sigma = s$ for some variable substitution σ.

Definition 5.3.2 (strong stability)
Let $\mathcal{R}$ be a CTRS. A term t is called *strongly stable* if every linearization of t is absolutely irreducible.

Note that it is sufficient to test one (arbitrary) linearization for absolute irreducibility when checking strong stability. One easily verifies that strong stability implies absolute irreducibility. In order to prove that strong stability implies stability, we introduce the following property.

Definition 5.3.3 (t-irreducible term)
Let $\mathcal{R}$ be a CTRS and t a term. A term is called *t-irreducible* if it is an instance of a linearization of t, and no instance of a linearization of t can be contracted at a position in $\mathsf{Pos}_{\mathcal{F}}(t)$.

Lemma 5.3.4 (property of a t-irreducible term)
Let $\mathcal{R}$ be a CTRS and t a term. Every reduct of a t-irreducible term is also t-irreducible.

Proof.
Let s be a t-irreduclible term and suppose $s \to_{\mathcal{R}}^* s'$. We show that s' is t-irreducible by induction on the length of $s \to_{\mathcal{R}}^* s'$. The case of zero length is trivial. Suppose $s \to_{\mathcal{R}}^* u \xrightarrow{p}_{\mathcal{R}} s'$. The t-irreducibility of s and the induction hypothesis yields the existence of a linearization t' of t such that
(1) $u = t'\sigma$ for some substitution σ and
(2) no instance of t' can be contracted at a position in $\mathsf{Pos}_{\mathcal{F}}(t)$.
We have to show the existence of a substitution τ such that $s' = t'\tau$. From property (2) there exists a position $q \in \mathsf{Pos}_{\mathcal{V}}(t)$ such that $q \leq p$. Define the substitution τ as follows: $\tau(x) = s'_{|q}$ if $t'_{|q} = x$, $\tau(x) = \sigma(x)$, otherwise. This is well-defined since t' is linear. Clearly we obtain $s' = t'\tau$. □

Lemma 5.3.5 (syntactic criteria for stability)
Let $\mathcal{R}$ be a CTRS. Every strongly stable term is stable.

Proof.
Let s be a strongly stable term and suppose $s\sigma \to_{\mathcal{R}} t \xrightarrow{p}_{\mathcal{R}} u$. We have to show that $p \notin \mathsf{Pos}_{\mathcal{F}}(s)$. For that purpose we prove the s-irreducibility of t. Since every reduct of s-irreducible term is also s-irreducible by Lemma 5.3.4, it suffices to show that $s\sigma$ is s-irreducible. Let s' be an arbitrary linearization of s. By strong stability of s, no instance of s' can be contracted at a position in $\mathsf{Pos}_{\mathcal{F}}(s)$. We also have to show the existence of a substitution τ such that $s\sigma = s'\tau$. The desired result is obtained by the following definition: $\tau(x) = \sigma(y)$ if $x \in \mathsf{Var}(s)$, $s'_{|p} = x$ and $s_{|p} = y$; $\tau(x) = x$ if $x \in \mathcal{V} \setminus \mathsf{Var}(s)$. □

Stable terms are not always strongly stable as shown in the following example.

Example 5.3.6 (stability and strong stability)
Consider the (unconditional) TRS $\mathcal{R} = \{\mathsf{f}(\mathsf{a}, \mathsf{b}) \to \mathsf{c}\}$. The term $\mathsf{f}(\mathrm{x}, \mathrm{x})$ is stable because $\mathsf{f}(t, t)$ is irreducible for all terms t. But it is not strongly stable because the linearization $\mathsf{f}(\mathrm{x}, \mathrm{y})$ unifies with the left-hand side $\mathsf{f}(\mathsf{a}, \mathsf{b})$.

In the context of programming, the notion of constructor is important to distinguish function symbols which are used to express data.

Definition 5.3.7 (constructor term)
Let $(\mathcal{F}, \mathcal{R})$ be a CTRS. A function symbol $f \in \mathcal{F}$ is called a *constructor* if $\mathsf{root}(l) \neq$

f for every conditional rewrite rule $l \to r \Leftarrow c \in \mathcal{R}$. A term built from constructors and variables is called a *constructor term* if there is no conditional rewrite rule $l \to r \Leftarrow c \in \mathcal{R}$ such that $l \in \mathcal{V}$.

Lemma 5.3.8 (syntactic criteria for stability)
Let $\mathcal{R}$ be a CTRS.
(1) Every linear absolutely irreducible term is stable.
(2) Every constructor term is stable.
(3) Every normal term is stable.

Proof.
Linear absolutely irreducible terms, constructor terms, and normal terms are always strongly stable by definition. Hence they are stable by Lemma 5.3.5. □

Consider again Example 4.3.10. The term $\mathsf{f}(\mathrm{x}, \mathrm{x})$ in the conditional part of the last rule of the CTRS $\mathcal{R}$ is absolutely irreducible but it is not stable. Since $\mathcal{R}$ is well-directed, Lemma 5.3.8(1) shows that the non-linearity of the term $\mathsf{f}(\mathrm{x}, \mathrm{x})$ is essential for the non-logicality of $\mathcal{R}$.

Example 5.3.9 (stability and related properties)
Consider the CTRS

$$\mathcal{R} = \{\ \mathsf{f}(\mathsf{a}, \mathsf{b}) \to \mathsf{g}(\mathsf{a}) \quad \Leftarrow \quad \mathsf{a} \approx \mathsf{b}\ \}.$$

The four terms $\mathsf{g}(\mathsf{a})$, $\mathsf{f}(\mathsf{a}, \mathsf{a})$, $\mathsf{g}(\mathrm{x})$ and $\mathsf{f}(\mathsf{a}, \mathsf{g}(\mathrm{x}))$ are strongly stable hence stable by Lemma 5.3.5. The term $\mathsf{g}(\mathsf{a})$ is an example of a normal constructor term. Since f is not a constructor symbol, $\mathsf{f}(\mathsf{a}, \mathsf{a})$ is not a constructor term but it is normal. The term $\mathsf{g}(\mathrm{x})$ is non-normal because it contains a variable but it is a constructor term.

In Definition 4.5.4 we defined level-unidirectionality using stability. Since stability depends on the rewrite relation, it seems more natural to use level-stability defined below instead of stability. But it turns out that two notions are equivalent.

Definition 5.3.10 (level-stability)
Let $\mathcal{R}$ be a CTRS. A term s is called *level-stable* if $p \notin \mathsf{Pos}_{\mathcal{F}}(s)$ whenever $s\sigma \to^*_{\mathcal{R}_k} t \xrightarrow{p}_{\mathcal{R}_k} u$, for all $k \in \mathbb{N}$, substitutions σ, terms t, u, and positions p. We say that $\mathcal{R}$ is stable if every right-hand side t of an equation $s \approx t$ in the conditional part c of a conditional rewrite rule $l \to r \Leftarrow c$ in $\mathcal{R}$ is stable.

Lemma 5.3.11 (level-stability and stability)
Let $\mathcal{R}$ be a CTRS. A term t is stable if and only if it is level-stable.

Proof.
The "$\Leftarrow$" direction is proved by contradiction. So let s be a level-stable term that is non-stable. By definition there exist terms t, u and a position $p \in \mathsf{Pos}_{\mathcal{F}}(s)$ such that $s\sigma \rightarrow^*_{\mathcal{R}} t \xrightarrow{p}_{\mathcal{R}} u$. Because this rewrite sequence is finite, there is a maximum level $k \in \mathbb{N}$ satisfying $s\sigma \rightarrow^*_{\mathcal{R}_k} t \xrightarrow{p}_{\mathcal{R}_k} u$. This contradicts the level-stability of s. The "$\Rightarrow$" direction is also proved by contradiction using the inclusion $\forall k \in \mathbb{N} \rightarrow_{\mathcal{R}_k} \subseteq \rightarrow_{\mathcal{R}}$. □

Next we investigate relationships between strong irreducibility and other properties.

Lemma 5.3.12 (strong irreducibility and related properties)
Let $\mathcal{R}$ be a CTRS.
(1) Every stable term is level-strongly irreducible.
(2) Every absolute irreducible term is level-strongly irreducible.
(3) Every level-strongly irreducible term is strongly irreducible.

Proof.

(1) We can show that level-stability implies level-strong irreducibility by an easy proof by contradiction. An application of Lemma 5.3.11 yields the desired result.
(2) The proof is by contradiction. We use the inclusion $\forall k \in \mathbb{N} \rightarrow_{\mathcal{R}_k} \subseteq \rightarrow_{\mathcal{R}}$.
(3) Straightforward. The equivalence $t \in \mathsf{NF}(\rightarrow_{\mathcal{R}}) \Leftrightarrow \forall k \in \mathbb{N}\ t \in \mathsf{NF}(\rightarrow_{\mathcal{R}_k})$ is useful.

□

From statements (2) and (3) in this lemma we know that absolute irreducibility is a sufficient syntactic criteria for both strong irreducibility and level-strong irreducibility. The following examples show that all implications in Lemma 5.3.12 are strict.

Example 5.3.13 (level-strong irreducibility and related properties)
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{rcl} \mathsf{a} & \rightarrow & \mathsf{b} \\ \mathsf{f(a,b)} & \rightarrow & \mathsf{c} \\ \mathsf{f(a,a)} & \rightarrow & \mathsf{c} \quad \Leftarrow \quad \mathsf{a} \approx \mathsf{c} \end{array} \right\}.$$

The term $\mathsf{f}(\mathrm{x},\mathrm{x})$ is level-strongly stable because the condition of the last rule is not satisfied. However, it is neither stable nor absolute irreducible: we have

$\mathsf{f(a,a)} \to_{\mathcal{R}} \mathsf{f(a,b)} \to \mathsf{c}$ and $\mathsf{f(x,x)}$ is unifiable with the left-hand side $\mathsf{f(a,a)}$ of the last rule.

Example 5.3.14 (strong irreducibility and related properties)
Consider the CTRS

$$\mathcal{R} = \left\{ \begin{array}{rcl} \mathsf{a} & \to & \mathsf{b} \\ \mathsf{f(b)} & \to & \mathsf{c} \\ \mathsf{b} & \to & \mathsf{c} \quad \Leftarrow \quad \mathsf{a} \approx \mathsf{b} \end{array} \right\}.$$

The term $\mathsf{f(x)}$ is strongly irreducible because $\mathsf{f}(t)$ is irreducible whenever t is an irreducible term. However, it is not level-irreducible because $\mathsf{f(b)} \notin \mathsf{NF}(\to_{\mathcal{R}_1})$, notwithstanding $\mathsf{b} \in \mathsf{NF}(\to_{\mathcal{R}_1})$.

Figure 5.2 summarizes the relationships between various properties investigated in this section. Here we use the following abbreviations: N (normal), SS (strongly stable), S (stable), LS (level-stable), AI (absolutely irreducible), LSI (level-strongly irreducible), and SI (strongly irreducible).

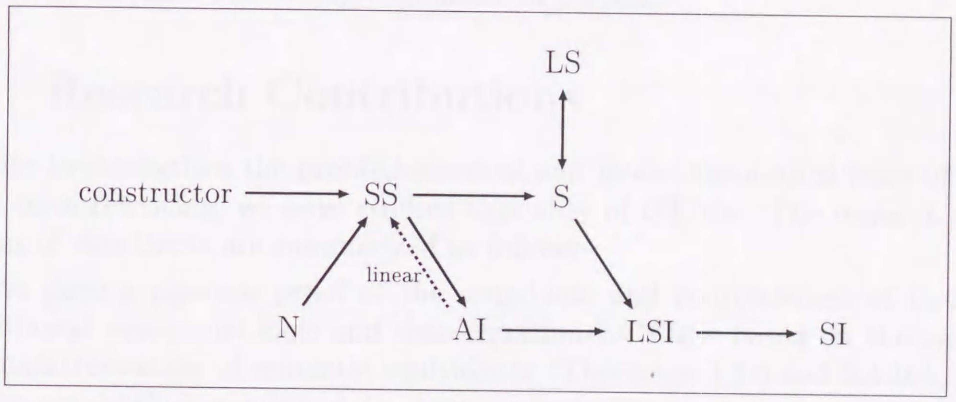


Figure 5.2: Relationships between properties concerned with irreducibility.

Chapter 6

Concluding Remarks

We conclude this thesis with some remarks. In Section 6.1 we summarize the research contributions of the thesis. In Section 6.2 the usefulness of the logicality results is illustrated. We show that our results cover two important classes of CTRSs considered by Avenhaus and Loría-Sáenz [ALS94] and Suzuki *et al.* [SMI95], ensuring their soundness and completeness. Moreover, we give a solution to an open problem by Toyama concerning confluence of CTRSs.

6.1 Research Contributions

In order to strengthen the proof-theoretical and model-theoretical basis of conditional term rewriting, we have studied logicality of CTRSs. The research contributions of this thesis are summarized as follows:

- We gave a rigorous proof of the soundness and completeness of both conditional equational logic and semi-equational CTRSs based on the syntactic characterization of semantic equivalence (Theorems 3.3.6 and 3.4.10).
- We established a method for proving logicality, and hence soundness and completeness, of join and oriented CTRSs (Corollary 4.1.3).
- We provided a uniform proof of known logicality results based on the characterization of rewrite relations of three types of CTRSs (Corollaries 4.2.3 and 4.3.4).
- We developed new sufficient conditions for logicality of oriented systems with weak normalization assumption (Theorem 4.3.7) and without normalization assumption (Theorem 4.3.16).
- We proposed sufficient conditions for the property that two different types of CTRSs have the same rewrite relation (Theorems 4.4.3, 4.4.5, 4.4.9, and 4.4.15) and have the same computational power (Theorems 4.4.11 and 4.4.16) by imposing restrictions on semi-equational CTRSs.

- We systematically analyzed various combinations of relations induced by different types of CTRSs (Theorems 4.5.1, 4.5.2, and 4.5.6).
- We studied techniques to ensure sufficient conditions for logicality.

type	requirements	Theorem/Corollary
semi-equational		4.1.2
join	$\mathcal{R}^{\mathrm{j}}$: CR	4.2.3
	$\mathcal{R}^{\mathrm{s}}$: CR + D	4.4.3
	$\mathcal{R}^{\mathrm{s}}$: LCR	4.4.5
oriented	$\mathcal{R}^{\mathrm{o}}$: CR + WN + SI	4.3.7
	$\mathcal{R}^{\mathrm{o}}$: CR + S + WD	4.3.16
	$\mathcal{R}^{\mathrm{o}}$: LCR + LWN + LSI	4.5.6(10)
	$\mathcal{R}^{\mathrm{s}}$: CR + SD + SI	4.4.12
	$\mathcal{R}^{\mathrm{s}}$: LCR + LSN + LSI	4.4.17
	$\mathcal{R}^{\mathrm{s}}$: LCR + LWN + LSI + WRI	4.5.6(8)
	$\mathcal{R}^{\mathrm{s}}$: LCR + S + WD + WRI	4.5.6(8)

Table 6.1: Summary of logicality results.

Sufficient conditions of logicality discussed in this thesis is summarized in Table 6.1. The following abbreviations are newly introduced in the table: LCR (level-confluence), WD (well-directedness), and WRI (weak right-independence).

6.2 Remarks on Applications

At the end of this thesis we discuss the usefulness of the newly obtained logicality results. First we show that the class of CTRSs proposed by Suzuki *et al.* in [SMI95] falls within the scope of Theorem 4.3.16. This class can be viewed as a computational model for functional logic programming languages with local definitions such as let-expressions and where-constructs.

Theorem 6.2.1 (logicality of level-confluent CTRSs)
Every orthogonal properly oriented right-stable CTRS is logical.

Proof.
According to Lemma 5.2.8, every CTRS in this class is level-confluent and hence confluent. Right-stability implies stability and proper orientation implies well-directedness. Therefore logicality follows from Theorem 4.3.16. □

Suzuki employs the above theorem in his proof of the completeness of narrowing for the class of orthogonal properly oriented right-stable CTRSs (see [Suz98]).

Theorem 4.3.7, the other new sufficient condition for the logicality of oriented TRSs, covers the class of CTRSs studied by Avenhaus and Loría-Sáenz (see [ALS94]). This class is useful for studying the (unique) termination behaviour of well-moded Horn clause programs [GW92].

Theorem 6.2.2 (logicality of confluent CTRSs)
Let $\mathcal{R}$ be a semi-decreasing strongly irreducible deterministic oriented CTRS such that all conditional critical pairs of $\mathcal{R}$ are convergent. Then $\mathcal{R}$ is logical.

Proof.
According to Lemma 5.2.8 every CTRS in this class is level-confluent and hence confluent. Semi-decreasingness implies weak normalization and strong determinism implies strong irreducibility. Hence the conditions of Theorem 4.3.7 are fulfilled. □

The results which compare two different types of relations (Theorem 4.5.1, Theorem 4.5.2, Theorem 4.5.6) are useful for proving the (level-)confluence of a CTRS from the confluence of another type of CTRS. Consider the following open problem by Toyama [DJK91, Problem 16]:

> Under what conditions does confluence of a normal oriented CTRS follow from confluence of the corresponding semi-equational CTRS?

Special cases of Theorem 4.5.6 provide solutions to this problem.

Theorem 6.2.3 (preservation of confluence)
Let $\mathcal{R}$ be a normal CTRS.
(1) If $\mathcal{R}^{s}$ is confluent and semi-decreasing then $\mathcal{R}^{o}$ is confluent.
(2) If $\mathcal{R}^{s}$ is level-confluent then $\mathcal{R}^{o}$ is level-confluent.

Proof.
As discussed in Section 5.3, normality implies both strong irreducibility and stability. Because every right-hand side of the conditional part in a conditional rewrite rule of a normal CTRS is ground $\mathcal{R}$ is both right-independent and well-directed. Therefore statement (1) follows from Theorem 4.5.6(3). From Theorem 4.4.19 we know that $\mathcal{R}^{o}$ is confluent if $\mathcal{R}^{s}$ is level-confluent. The stronger result (2) follows from Theorem 4.5.6(4). □

From Theorem 4.5.2 we know that a semi-equational CTRS is level-confluent if and only if the corresponding CTRS is level-confluent. This result enables us to extend the modularity of level-confluence (Theorem 5.2.10) to the case of semi-equational CTRSs.

Theorem 4.5.6(10) shows that if an oriented CTRS $\mathcal{R}^{\mathrm{o}}$ is level-confluent and level-unidirectional then the equality $\downarrow_{\mathrm{s}_k} = \downarrow_{\mathrm{j}_k} = \downarrow_{\mathrm{o}_k}$ holds for every level $k \in \mathbb{N}$. Hence the syntactic sufficient conditions for level-confluence of oriented CTRSs by Suzuki *et al.* (Theorem 5.2.8(2)) can be also applied to ensuring the level-confluence of join and semi-equational CTRSs.

Bibliography

[ALS93] J. Avenhaus and C. Loría-Sáenz. Canonical conditional rewrite systems containing extra variables. Technical Report SR–93–03, Universität Kaiserslautern, Germany, March 1993.

[ALS94] J. Avenhaus and C. Loría-Sáenz. On conditional rewrite systems with extra variables and deterministic logic programs. In *Proceedings of the 5th International Conference on Logic Programming and Automated Reasoning*, 1994. Lecture Notes in Artificial Intelligence 822, pp. 215–229.

[Bar84] H.P. Barendregt. *The Lambda Calculus: Its Syntax and Semantics.* Studies in Logic and the Foundations of Mathematics. North-Holland, 2nd edition, 1984.

[BK86] J.A. Bergstra and J.W. Klop. Conditional rewrite rules: Confluence and termination. *Journal of Computer and System Science*, 32:323–362, 1986.

[BG89] H. Bertling and H. Ganzinger. Completion-time optimization of rewrite-time goal solving. In *Proceedings of the 3rd International Conference on Rewriting Techniques and Applications*, 1989. Lecture Notes in Computer Science 355, pp. 45–58.

[Bir35] G. Birkhoff. On the structure of abstract algebras. *Proceedings of the Cambridge Philosophical Society*, 31:433–454, 1935.

[DP90] B.A. Davey and H.A. Priestley. *Introduction to Lattices and Order.* Cambridge University Press, 1990.

[Der87] N. Dershowitz. Termination of rewriting. *Journal of Symbolic Computation*, 3:69–116, 1987.

[DJ90] N. Dershowitz and J.-P. Jouannaud. Rewrite systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, chapter 6, pages 243–320. The MIT Press, 1990.

[DJK91] N. Dershowitz, J.-P. Jouannaud, and J.W. Klop. Open problems in rewriting. In *Proceedings of the 4th International Conference on Rewriting Techniques and Applications*, 1991. Lecture Notes in Computer Science 488, pp. 445–456.

[DO90] N. Dershowitz and M. Okada. A rationale for conditional equational programming. *Theoretical Computer Science*, 75:111–138, 1990.

[DOS88a] N. Dershowitz, M. Okada, and G. Sivakumar. Canonical conditional rewrite systems. In *Proceedings of the 9th International Conference on Automated Deduction*, 1988. Lecture Notes in Computer Science 310, pp. 538–549.

[DOS88b] N. Dershowitz, M. Okada, and G. Sivakumar. Confluence of conditional rewrite systems. In *Proceedings of the 1st International Workshop on Conditional Term Rewriting Systems*, Orsay, France, 1988. Lectuer Notes in Computer Science 308, pp. 31–44.

[DP88] N. Dershowitz and D.A. Plaisted. Equational programming. In J.E. Hayes, D. Michie, and J. Richards, editors, *Machine Intelligence*, volume 11, pages 21–56. Oxford Press, 1988.

[Gan91] H. Ganzinger. Order-sorted completion: The many-sorted way. *Theoretical Computer Science*, 89:3–32, 1991.

[GW92] H. Ganzinger and U. Waldmann. Termination proofs of well-moded logic programs via conditional rewrite systems. In *Proceedings of the 3rd International Workshop on Conditional Term Rewriting Systems*, 1992. Lecture Notes in Computer Science 656, pp. 430–437.

[GM86] E. Giovannetti and C. Moiso. A completeness result for E-unification algorithms based on conditional narrowing. In *Proceedings of the Workshop on Foundations of Logic and Functional Programming*, 1986. Lecture Notes in Computer Science 306, pp. 157–167.

[Ham97] M. Hamana. Equivalence of the quotient term model and the least complete Herbrand model for a funcional logic language. *The Journal of Functional and Logic Programming*, 1997(1):1–22, January 1997.

[Han95] M. Hanus. On extra variables in (equational) logic programming. In *Proceedings of the 12th International Conference on Logic Programming*, pages 665–679. MIT Press, Cambridge, 1995.

[Hue80] G. Huet. Confluent reductions: Abstract properties and applications to term rewriting systems. *Journal of the Association for Computing Machinery*, 27(4):797–821, 1980.

[HO80] G. Huet and D.C. Oppen. Equations and rewrite rules: A survey. In R. Book, editor, *Formal Language Theory: Perspectives and Open Problems*, pages 349–405. Academic Press, 1980.

[JW86] J.-P. Jouannaud and B. Waldmann. Reductive conditional term rewriting systems. In *Proceedings of the 3rd IFIP Working Conference on Formal Description of Programming Concepts*, pages 233–244. North-Holland, 1986.

[Kap84] S. Kaplan. Conditional rewrite rules. *Theoretical Computer Science*, 33(2):175–193, 1984.

[Kap87] S. Kaplan. Simplifying conditional term rewriting systems: Unification, termination and confluence. *Journal of Symbolic Computation*, 4:295–334, 1987.

[Klo90] J.W. Klop. Term rewriting systems. Technical Report CS-R9073, Department of Mathematics and Computer Science, Free University, December 1990.

[Klo92] J.W. Klop. Term rewriting systems. In S. Abramsky, D. Gabbay, and T. Maibaum, editors, *Handbook of Logic in Computer Science*, volume 2, chapter 1, pages 1–116. Oxford University Press, 1992.

[KB70] D.E. Knuth and P.B. Bendix. Simple word problems in universal algebras. In J. Leech, editor, *Computational Problems in Abstract Algebra*, pages 263–297. Pergamon Press, 1970.

[MT92] K. Meinke and J.V. Tucker. Universal algebra. In S. Abramsky, Dov M. Gabbay, and T.S.E. Maibaum, editors, *Handbook of Logic in Computer Science*, volume 2, chapter 3, pages 189–411. Oxford University Press, 1992.

[Mid90] A. Middeldorp. *Modular Properties of Term Rewriting Systems*. PhD thesis, Vrije Universiteit, Amsterdam, 1990.

[MH94] A. Middeldorp and E. Hamoen. Completeness results for basic narrowing. *Applicable Algebra in Engineering, Communication and Computing*, 5:231–253, 1994.

[New42] M.H.A. Newman. On theories with a combinatorial definition of equivalence. *Annals of Mathematics*, 43(2):223–243, 1942.

[O'D77] M.J. O'Donnell. *Computing in Systems Described by Equations*, volume 58 of *Lecture Notes in Computer Science*. Springer, 1977.

[Pla93] D.A. Plaisted. Equational reasoning and term rewriting systems. In Dov M. Gabbay, C. J. Hogger, and J. A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 1 (Logical Foundations), chapter 5, pages 273–364. Oxford University Press, New York, 1993.

[Ros73] B.K. Rosen. Tree-manipulating systems and Church-Rosser theorems. *Journal of the ACM*, 20(1):160–187, 1973.

[SS93] G. Schmidt and T. Ströhlein. *Relations and graphs: Discrete Mathematics for Computer Scientists*. EATCS Monographs on Theoretical Computer Science. Springer-Verlag, 1993.

[Sel72] A. Selman. Completeness of calculi for axiomatically defined classes of algebras. *Algebra Universalis*, 2:20–32, 1972.

[Ste95] J. Steinbach. Simplification orderings: History of results. *Fundamenta Informaticae*, 24:24–87, 1995.

[Suz98] T. Suzuki. *Studies on Conditional Narrowing for Rewrite Systems with Extra Variables*. PhD thesis, University of Tokyo, 1998.

[SMI95] T. Suzuki, A. Middeldorp, and T. Ida. Level-confluence of conditional rewrite systems with extra variables in right-hand sides. In *Proceedings of the 6th International Conference on Rewriting Techniques and Applications*, 1995. Lecture Notes in Computer Science 914, pp. 179–193.

[Toy87] Y. Toyama. On the Church-Rosser property for the direct sum of term rewriting systems. *Journal of the Association for Computing Machinery*, 34(1):128–143, 1987.

[Wec92] W. Wechler. *Universal Algebra for Computer Scientists*. EATCS Monographs on Theoretical Computer Science. Springer-Verlag, 1992.

[YALSM97] T. Yamada, J. Avenhaus, C. Loría-Sáenz, and A. Middeldorp. Logicality of conditional rewrite systems. In *Proceedings of the 22nd International Colloquium on Trees in Algebra and Programming*, 1997. Lecture Notes in Computer Science 1214, pp. 141–152.

[YALSM99] T. Yamada, J. Avenhaus, C. Loría-Sáenz, and A. Middeldorp. Logicality of conditional rewrite systems. *Theoretical Computer Science*, 1999. To appear.

[YMI95] T. Yamada, A. Middeldorp, and T. Ida. Modularity of level-confluence for conditional term rewriting systems. *Computer Software*, 12(5):72–84, 1995. (In Japanese).

Index

A

absolutely irreducible 48
abstract rewrite system 11
AI 78
applicable 28, 38, 42
arity 16
ARS 11
assignment 21

C

carrier 21
CEL 19
CES 19
chain 32
Church-Rosser property 12
closed under contexts 18
closed under substitutions 18
common reduct 12, 13
compatible 28
complete 13, 28, 31
composition 10, 18
conditional critical pair 71
conditional equation 19
conditional equational system ... 19
conditional part 19
conditional rewrite rule 23
confluent 12
congruence 28
connected 13
constant 16
constructor 75
constructor term 76
continuous 32
convergent 71
conversion 11
convertibility 11
CR 12
CTRS 23

D

D 71
decreasing 54
deterministic 69
domain 17

E

empty relation 9
equation 19
equational system 19
equational theory 21
equivalence closure 10
equivalence relation 10
ES 19
extra variable 25

F

$\mathcal{F}$-algebra 21
function symbols 16

G

ground 16

H

hole 18

I

idempotency 11
identity relation 9
incrementality 11
instance 18
inverse 9
irreducible 11, 18

J

join-equational 24
joinability 12

L

LCR 80
left-linear 72
level-confluent 55
level-stable 76
level-strongly irreducible 58
level-unidirectional 63
level-weakly normalizing 58
linear 16
linearization 74
locally confluent 12
logical 42
LS 78
LSI 78
LSN 71
LWN 71

M

model 22
monotone 32
monotonicity 11

N

N 78
natural mapping 29
n-CTRS 26
non-variable position 17
normal 44
normal form 11

O

oriented 24
orthogonal 72

P

$\mathcal{P}$-closure 10
position 17
proper subterm 17
properly oriented 73
provable 20
provable equivalence 20

Q

QR 71
quasi-reductive 69
quotient term algebra 29

R

reduct 11
reflexive 10
reflexive closure 10
reflexive-transitive closure 10
relation 9
rewrite relation 18
rewrite sequence 12
rewrite step 11
right-independent 56
right-stable 73
root position 17
root symbol 16

S

S 78
SD 71
semantic equivalence 22
semi-complete 13
semi-decreasing 56
semi-equational 24
SI 78
signature 16
SN 12
sound 28
SS 78
stable 49
strongly irreducible 46
strongly normalizing 12
strongly stable 74
substitution 17
subterm 16
symmetric 10
symmetric closure 10

T

term 16
term algebra 21
term rewrite system 23
terminating 12
t-irreducible 74
transitive 10
transitive closure 10
TRS 23

U

UN 12
unconditional equation 19
unconditional version 67
unidirectional 63
unifiable 19
unique normal forms 12

V

valid 22
variable position 17
variable renaming 17
variable substitution 17
variables 16
variety 22

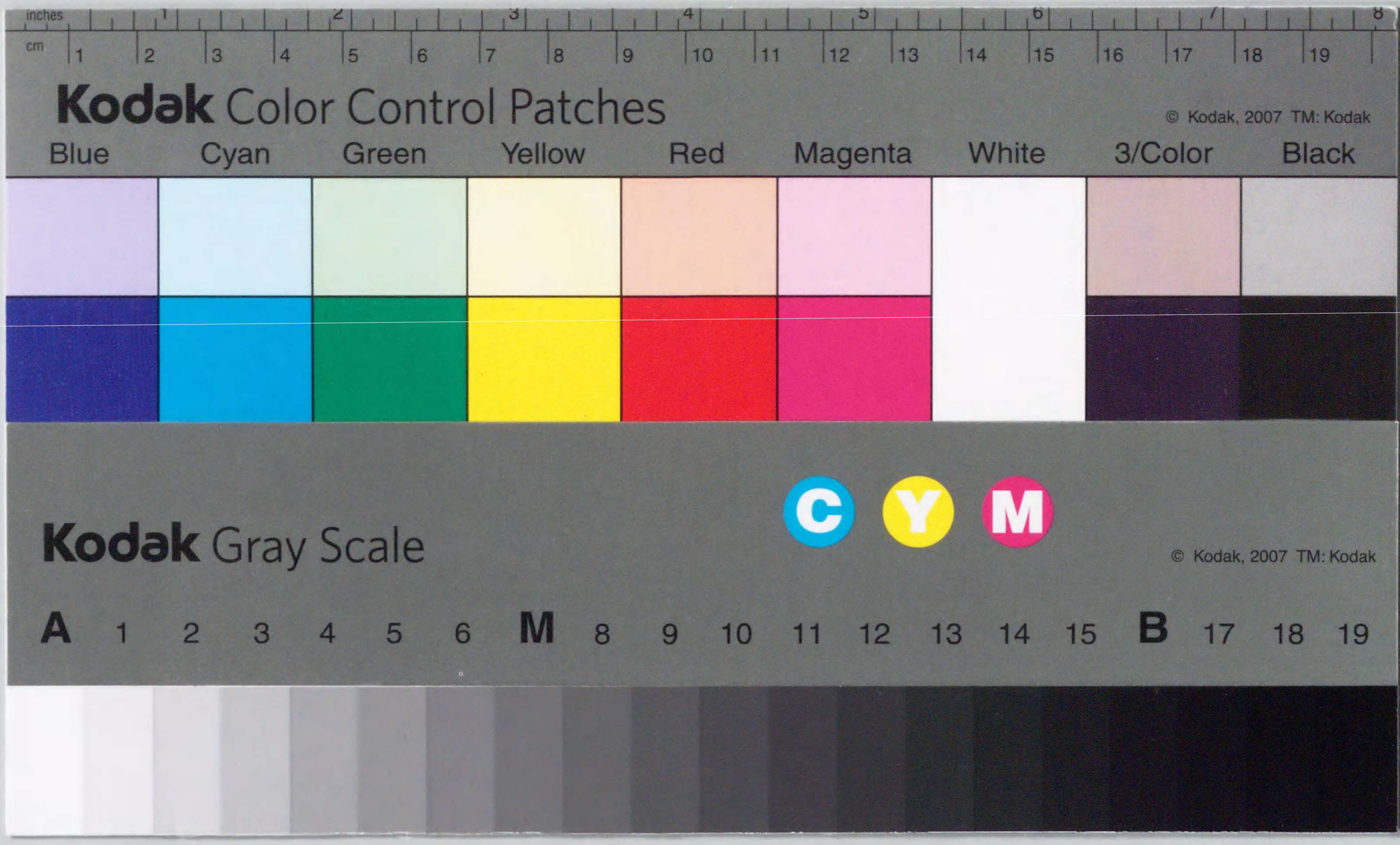
W

WCR 12
WD 80
weak Church-Rosser property ... 12
weakly normalizing 12
weakly right-independent 64
well-directed 51
WN 12
WRI 80

List of Notations

$a\,R\,b$ 9
$\varnothing$ 9
Id_A 9
R^{-1} 9
$R \circ S$ 10
R^2 10
R^n 10
$R^=$ 10
R^{sym} 10
R^+ 10
R^* 10
R^{eqv} 10
$(A, \rightarrow)$ 11
$\rightarrow^*$ 11
$\leftrightarrow$ 11
$\leftrightarrow^*$ 11
$\mathsf{NF}(\rightarrow)$ 11
$\leftarrow$ 11
${}^=\!\!\leftarrow$ 11
${}^+\!\!\leftarrow$ 11
${}^*\!\!\leftarrow$ 11
$\downarrow$ 12
$a\downarrow$ 15
$\mathcal{V}$ 16
$\mathcal{F}$ 16
$\mathcal{T}(\mathcal{F}, \mathcal{V})$ 16
$\mathsf{Var}(t)$ 16
$\mathsf{root}(t)$ 16
$\trianglelefteq$ 16
$\triangleleft$ 17
ϵ 17
$i \cdot p$ 17
$\mathsf{Pos}(t)$ 17
$t_{|p}$ 17
$t[s]_p$ 17
$\mathsf{Pos}_{\mathcal{V}}(t)$ 17
$\mathsf{Pos}_{\mathcal{F}}(t)$ 17
$\Sigma(\mathcal{F}, \mathcal{V})$ 17
$\overline{\sigma}$ 18
$t\sigma$ 18
$\sigma\tau$ 18
$\Box$ 18
$C[t]$ 18
$\mathcal{C}(\mathcal{F}, \mathcal{V})$ 18
$R^{\mathcal{C}}$ 18
R^{Σ} 18
$s \approx t$ 19
$l \approx r \Leftarrow c$ 19
$(\mathcal{F}, \mathcal{E})$ 19
$\approx_{\mathcal{E}}$ 20
$\approx_{\mathcal{E}_k}$ 20
$f_{\mathbf{A}}$ 21
$\mathbf{T}(\mathcal{F}, \mathcal{V})$ 21
$\overline{\alpha}$ 22
$\mathsf{M}_{\mathcal{E}}$ 22
$=_{\mathsf{M}_{\mathcal{E}}}$ 22
$l \rightarrow r \Leftarrow c$ 23
$\rightarrow_{\mathcal{R}}$ 23
$\rightarrow_{\mathcal{R}_k}$ 24
$\vdash$ 24
$\xrightarrow{p}_{\mathcal{R}}$ 24
$\mathcal{R}^{\mathsf{s}}$ 24
$\mathcal{R}^{\mathsf{j}}$ 24
$\mathcal{R}^{\mathsf{o}}$ 24
$\rightarrow_{\mathsf{s}}$ 24
$\rightarrow_{\mathsf{j}}$ 24

$\rightarrow_{\mathrm{o}}$ 24
$\mathbf{Q}_{\equiv}$ 29
$\nu_{\equiv}$ 29
$\rightarrow_{\Phi}$ 31
$\rightarrow_{\Phi_k}$ 31
$\perp_{\mathcal{C}}$ 31
$\Phi_{\mathcal{E}}$ 34
$R^{[\mathsf{r}]}$ 35
$R^{[\mathsf{s}]}$ 35
$R^{[\mathsf{t}]}$ 35
$R^{[\mathsf{c}]}$ 35
$R^{[\mathsf{a}]}$ 35
$\Phi_{\mathcal{R}}$ 37
$\rightarrow^{\mathcal{R}}$ 37
$\mathsf{O}(\mathcal{R})$ 65
$\mathcal{R}_{\mathrm{u}}$ 67


Kodak Color Control Patches
© Kodak, 2007 TM: Kodak
Blue Cyan Green Yellow Red Magenta White 3/Color Black
Kodak Gray Scale
C Y M
© Kodak, 2007 TM: Kodak
A 1 2 3 4 5 6 M 8 9 10 11 12 13 14 15 B 17 18 19