Strict Coherence of Conditional Rewriting Modulo Axioms

José Meseguer

University of Illinois at Urbana-Champaign, IL 61801, USA

Abstract

Conditional rewriting modulo axioms with rich types makes specifications and declarative programs very expressive and succinct and is used in all well-known rule-based languages. However, the current foundations of rewriting modulo axioms have focused for the most part on the unconditional and untyped case. The main purpose of this work is to generalize the foundations of rewriting modulo axioms to the conditional order-sorted case. A related goal is to simplify such foundations. In particular, even in the unconditional case, the notion of *strict coherence* proposed here makes rewriting modulo axioms simpler and easier to understand. Properties of strictly coherent conditional theories, like operational equi-termination of the $\rightarrow_{R/B}$ and $\rightarrow_{R,B}$ relations and general conditions for the conditional Church-Rosser property modulo B are also studied.

Keywords: Conditional rewriting modulo equations, coherence, order-sorted specifications, operational termination, Church-Rosser property.

1. Introduction

Techniques for rewriting modulo axioms B are very useful. In practical, declarative programming terms, what they afford —particularly in combination with an expressive type structure such as order-sorted or membership equational logic [31, 54]— is making available to the programmer a very rich variety of user-definable *data types* such as multisets, sets, lists with associative matching, combinations of all these with the usual tree-like data structures, and so on. Such expressive data types and matching *modulo* their algebraic properties allow the formulation of very expressive and remarkably succinct solutions to many computational problems as declarative programs. Substantiating this claim would require another paper; but I can refer the reader to [11, 55] and, regarding efficient implementation of rewriting modulo commonly used axioms to [22], for what I consider by now unquestionable evidence.

In spite of the great usefulness of techniques for rewriting modulo axioms, this is a quite specialized topic, a kind of *esoterica* among rewriting techniques,

Email address: meseguer@illinois.edu (José Meseguer)

with many puzzling phenomena. The difficulties in accessing this area were recognized and addressed by Narendran, Subramanian and Guo in their expository note [59], studded with many intriguing examples, where they say:

> *Over the years we (and several others) have found the concepts very hard to internalize. One of the problems is that only the case where $E = AC$ has been investigated in detail. . . . . Experience with other equational theories has been lacking. As a result, many misleading and erroneous statements have crept into even some excellent papers.*

The above words sound a warning, since any study in this area runs the risk of making the entire subject even more impenetrable or, what would be worse, of adding some erroneous statements to the literature. Yet, there is a need to press on and make further progress, because the foundations of rewriting modulo axioms are still relatively undeveloped, and many practical applications require new foundations that are both simpler and more general.

I am referring particularly to the area of *conditional rewriting modulo axioms*. Conditional rewrite rules make declarative programs more expressive and are, for this reason, supported, in combination with rewriting modulo axioms, by many declarative rule-based languages such as OBJ [32], ASF+SDF [69], ELAN [7], CafeOBJ [25] and Maude [11]. However, there are at present two related foundational problems. The first problem is that, except for the limited foundations of conditional rewriting modulo equations provided by papers such as, e.g., [27, 44, 53, 6, 9, 18, 21], the overwhelming majority of studies in this area, e.g., [36, 45, 47, 46, 61, 37, 40, 38, 3, 39, 59, 43, 26] treat only the *unconditional* rewriting case. The second problem is that all the papers treating the unconditional case do so for *unsorted* signatures, which are unusable in practice, since all the declarative languages just mentioned support, for obvious reasons of expressiveness, many-sorted, order-sorted, or membership equational logic type structures.

To address the practical needs just explained, the main goal of this paper is to develop new foundations for *conditional* rewriting modulo axioms for rewrite theories with an *order-sorted* [31, 54] type structure. Order-sorted algebra provides an algebraic semantics for a typing discipline in which types are structured in *inheritance hierarchies*, and function symbols can be *subtype polymorphic* (e.g., $+$ and $\times$ for $Nat < Int < Rat < Real < Complex$). This makes specifications and programs very natural and expressive, and solves many partiality issues impossible to solve in a simply-typed setting, such as, for example, division by zero, top of an empty stack, or first element of an empty list. For these reasons, order-sorted typing is widely used: it has been adopted by many declarative programming and specification languages, e.g., [32, 66, 28, 30, 11, 25, 5, 58], and is used also in resolution theorem proving, e.g., [71, 12, 64, 10], artificial intelligence, e.g., [66, 13, 24], and unification theory, e.g., [57, 67, 42, 34, 23]. Furthermore, it has a smooth, conservative extension to a higher-order typing discipline supporting subtypes [51, 33]. In all these areas, having well-developed foundations for conditional rewriting modulo equational axioms can be very useful. But of course, whoever solves a more general problem solves in one blow all

its instances. This is important, because order-sorted typing contains the widely used many-sorted typing as a special case which, in turn, contains the untyped approach (the unsorted case) as the *least general possible* instance. Fortunately, the added technical complexities involved in treating the much more general order-sorted case are relatively minor ones.

However, since conditional rewriting is unavoidably more complex and subtle than unconditional rewriting, how can one even hope to make the somewhat esoteric subject of rewriting modulo axioms more accessible in a conditional setting? My answer to this real and challenging question is based on distinguishing two issues:

1. Since conditional rewriting includes unconditional rewriting as a special case, the overall subject of rewriting modulo axioms *will* be simplified and made more accessible if the *specialization* of the new, more general treatment to the unconditional case is indeed simpler and more easily understandable than earlier unconditional treatments. This is achieved in Section 2, where several equivalent notions characterizing what I call *strict coherence* of rewriting modulo axioms are presented. Strict coherence is considerably simpler and has substantially better properties than the well-known notion of *coherence* [37, 38], which is at present the standard notion in the subject. In Section 4 I revisit this matter and show in detail that if the axioms B are regular and linear, specifications —both unconditional and conditional ones— can be completed (at least in the limit) to become strictly coherent, and can be easily checked for this property.[1]

2. Conditional rewriting modulo axioms *is* intrinsically more complex and subtle than unconditional rewriting modulo: this is the price to be paid for more expressive specifications and programs. The key issue, however, is whether conditional rewriting modulo axioms can be made *as simple as possible with the best possible properties*. The desired simplification and good properties are achieved: (i) in Section 4 by showing that if the axioms B are linear and regular the closure under B-extensions of a conditional specification enjoys the remarkably good and simple property of strict coherence; (ii) in Section 5 by showing that if a conditional specification is closed under B-extensions, the R/B- and R, B-rewrite relations are *operationally equi-terminating*, i.e., they have the same conditional termination properties; and (iii) in Section 6 by stating several theorems characterizing classes of conditional rewrite theories that enjoy the Church-Rosser property modulo B, and by identifying general conditions under which

[1] Admittedly, the regularity and linearity of B make the applications of strict coherence less general than those of coherence, where no such limitations are imposed. However, I argue in Section 2.1 that such greater generality comes at a considerably high price of losing many useful properties and gaining a number of anomalies; and I further argue in Section 7 that *strong coherence* [70, 21] is an attractive, alternative general framework for conditional rewriting modulo very general equational axioms, which can even be conditional.

provable equality of a conditional equational theory becomes decidable by rewriting modulo B.

The paper is organized as follows. Strict coherence is defined in Section 2. Conditional order-sorted rewriting modulo axioms is defined in Section 3. Strict coherence of conditional order-sorted rewrite theories modulo regular and linear axioms B is studied in Section 4. Equi-termination of conditional R/B- and R, B-rewriting for theories closed under B-extensions is proved in Section 5. The conditional Church-Rosser property modulo B is studied in Section 6. A discussion of related work and concluding remarks are given in Section 7.

2. Abstract Characterization of Strict Coherence

In this section strict coherence is characterized by means of four equivalent properties, namely, **Completeness**, **Bisimulation**, **Strict Coherence**, and **Strict Local Coherence**. The last two notions are *stronger versions* of the notions of *coherence* (resp. *E-commuting*) and *local coherence* (resp. *locally E-commuting*) in [38] (resp. [40]); and they are related to, yet different from, the notions of *strong coherence* and *strong local coherence* in [70, 21]. The qualification "strict" has been chosen to avoid terminological confusion with all these related but different notions of coherence in [38, 40, 70, 21].

The equivalences with **Completeness** and with **Bisimulation** help bringing out some important semantic properties implicit in strict coherence, as already emphasized in [20, 21]. A discussion of some semantic consequences of these four equivalent properties, why they are important, and why in general they do not hold when the axioms B fail to be regular and linear is also given. The treatment below is in terms of *abstract relations* in the spirit of, e.g., [36, 38].

Let T be a set, whose elements are denoted $t, t', u, u', v, v', w, w'$ and so on. We will assume several binary relations on T, including the following. First, a symmetric relation on T, denoted $\leftrightarrow_B$. Second, the smallest equivalence relation generated by $\leftrightarrow_B$, i.e., its reflexive-transitive closure, denoted $=_B$. Third, a binary relation on T, denoted $\rightarrow_{\{R\}}$. Fourth, the relation $\rightarrow_{R/\!\!/B}$, defined as the composition[2] $=_B; \rightarrow_{\{R\}}; =_B$. We also assume a fifth relation $\rightarrow_{R:B}$ between $\rightarrow_{\{R\}}$ and $\rightarrow_{R/\!\!/B}$, that is, such that $\rightarrow_{\{R\}} \subseteq \rightarrow_{R:B} \subseteq \rightarrow_{R/\!\!/B}$.

Example 1. *Although a much more general and detailed description of rewriting modulo axioms B at the conditional order-sorted level will be given in Section 3, the treatment of the special case of unsorted rewriting modulo B with unconditional rewrite rules R is so simple and well-known that a short summary can already be given here. The reader is referred to Section 3 for any remaining notational issues that he/she might be unfamiliar with.*

We assume an unsorted signature Σ of function symbols, say $f \in \Sigma$, each with an arity $ar(f) \in \mathbb{N}$. The set $\mathcal{T}_\Sigma(\mathcal{X})$ of Σ-terms *with variables $\mathcal{X}$ is defined*

[2] A relation composition $G; H$ is written in diagrammatic order, i.e., $G; H = \{(x, z) \mid (\exists y)\ xGy \wedge yHz\}$.

in the obvious, inductive way, so that variables and constants are Σ-terms, and if $ar(f) = n$ and $t_1, \ldots, t_n$ are Σ-terms, then $f(t_1, \ldots, t_n)$ is also a Σ-term. a set B of equations is a set of formulas of the form $u = v$, with u, v Σ-terms. Likewise, a set R of rewrite rules is a set of sequents of the form $l \to r$ with l, r Σ-terms. Call a triple $\mathcal{R} = (\Sigma, B, R)$ an unsorted and unconditional rewrite theory*.*

We can then instantiate the above framework of abstract relations for an unsorted and unconditional rewrite theory (Σ, B, R) as follows:

1. *The set T on which all relations are based is $\mathcal{T}_\Sigma(\mathcal{X})$.*

2. *The abstract relation $\to_{\{R\}}$ is interpreted as the R-*rewriting relation *$\to_R$ defined as follows: $t \to_R t'$ iff there exists $l \to r \in R$, a position p in the term t viewed as a tree, and a substitution σ such that $t|_p = l\sigma$, and $t' = t[r\sigma]_p$, where $t[w]_p$ denotes the result of replacing in term t at position p the subterm $t|_p$ by the term w (see Section 3 for further details).*

3. *The abstract relation $\leftrightarrow_B$ is interpreted as the $\overset{\leftrightarrow}{B}$-rewrite relation $\to_{\overset{\leftrightarrow}{B}}$ for the set of rewrite rules $\overset{\leftrightarrow}{B} = \{u \to v \mid u = v \in B \vee v = u \in B\}$, is denoted $\leftrightarrow_B$, and is called the* one-step B-equality *relation.*

4. *The abstract relation $=_B$ is interpreted as the reflexive-transitive closure $(\leftrightarrow_B)^*$ of $\leftrightarrow_B$, is denoted $=_B$, and is called the* B-equality *relation.*

5. *The abstract relation $\to_{R/\!\!/B}$ is interpreted as the relation $\to_{R/B}$ defined as the composition $\to_{R/B} = (=_B; \to_R; =_B)$, and is called* R/B-rewriting*.*

6. *The abstract relation $\to_{R:B}$ is interpreted as a relation $\to_{R,B}$ that goes back to [61] and is defined as follows: $t \to_{R,B} t'$ iff there exist a position p in t, a rule $l \to r$ in R and a substitution σ such that $t|_p =_B l\sigma$, and $t' = t[r\sigma]_p$. It is called* R, B-rewriting*.*

Note that the relation $\to_{R/B}$ describes rewriting modulo B*. That is, it describes the action of $\to_R$ on $=_B$-equivalence classes. Indeed, $\to_{R/B}$ induces a binary relation —also denoted $\to_{R/B}$ by abuse of language— on the quotient algebra $\mathcal{T}_\Sigma(\mathcal{X})/=_B$ defined by the equivalence:*

$$[t]_B \to_{R/B} [t']_B \quad \Leftrightarrow \quad t \to_{R/B} t',$$

where $[t]_B$ abbreviates the equivalence class $[t]_{=_B}$. The point of introducing also the relation $\to_{R,B}$ is that it is much simpler to implement than $\to_{R/B}$, yet, as explained below at the level of abstract relations, it can be made bisimilar *to $\to_{R/B}$ under an appropriate strict coherence condition. The concrete consequences of this to* implement *the relation $\to_{R/B}$ by means of the relation $\to_{R,B}$ thanks to strict coherence are explained right after Corollary 1 below.*

Remark 1. *The use of a different notation:* $\to_{\{R\}}$ *vs.* $\to_R$, $\to_{R/\!\!/B}$ *vs.* $\to_{R/B}$, *and* $\to_{R:B}$ *vs.* $\to_{R,B}$, *for the abstract and concrete relations might seem pedantic. In fact, if the only applications envisioned were to unconditional rewriting modulo* B, *it* would be *pedantic: no such distinctions are needed.*

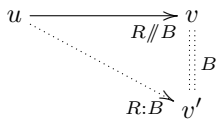
However, as explained in Section 4.3, one of the new insights provided in this paper is that, in the natural generalization of the concrete relations $\to_R$, $\to_{R/B}$, *and* $\to_{R,B}$ *to the* conditional *case, the above, perfect alignment between the abstract relations* $\to_{\{R\}}$, $\to_{R/\!\!/B}$, *and* $\to_{R:B}$, *and the concrete relations* $\to_R$, $\to_{R/B}$, *and* $\to_{R,B}$, breaks down*, so that the notational distinction between abstract and concrete relations becomes essential.*

The intuition about the abstract relation $\to_{R:B}$ is that it is easier to compute than $\to_{R/\!\!/B}$, but under suitable conditions it should still allow us to perform "essentially the same rewrites" as $\to_{R/\!\!/B}$. This is captured by the **Completeness** property below. I follow the usual diagrammatic convention, where dashed lines indicate existential quantification. I spell this out in a first instance (the **Completeness** property), and freely use the convention from then on.

1. **Completeness**. This property relates $\to_{R/\!\!/B}$ and $\to_{R:B}$ by the following property:

$$(\forall u, v \in T)\ u \to_{R/\!\!/B} v \Rightarrow ((\exists v' \in T)\ u \to_{R:B} v' \wedge v =_B v').$$

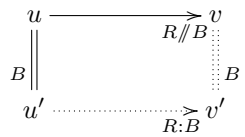
Diagrammatically, the existential quantification is precisely described by the dashed lines in the diagram below.



where here and in what follows dotted lines indicate existential quantification.

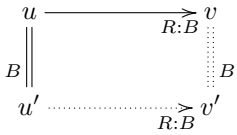
A related intuition about $\to_{R:B}$ is that it should in an appropriate sense *simulate* $\to_{R/\!\!/B}$. This is captured by the **Bisimulation** property below.

2. **Bisimulation**. This property also relates $\to_{R/\!\!/B}$ and $\to_{R:B}$ as follows:

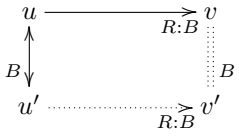


As its name indicates, this property is equivalent to the relation $=_B$ being a *bisimulation* between the transition systems $(T, \to_{R/\!\!/B})$ and $(T, \to_{R:B})$. Strictly speaking, the property just states that $=_B$ is a *simulation relation* of $(T, \to_{R/\!\!/B})$ by $(T, \to_{R:B})$. However, $=_B$ is *always* a simulation of $(T, \to_{R:B})$ by $(T, \to_{R/\!\!/B})$ (and thus a bisimulation), since we have $(=_B; \to_{R:B}) \subseteq (=_B; \to_{R/\!\!/B}) = \to_{R/\!\!/B}$. Therefore, $(u \to_{R:B} v \wedge u =_B u')$ implies $u' \to_{R/\!\!/B} v$.

3. **Strict Coherence**. This property relates $\rightarrow_{R:B}$ to itself. It can be briefly summarized by saying that $=_B$ is a *bisimulation* of $(T, \rightarrow_{R:B})$. Instead of writing the corresponding formula, I state it diagrammatically:

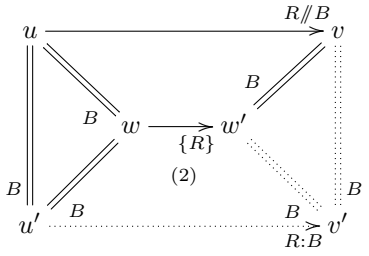


4. **Strict Local Coherence**. This property also relates $\rightarrow_{R:B}$ to itself, but is easier to check because it uses $\leftrightarrow_B$ instead of $=_B$ in its universal part (the existential part still uses $=_B$).



Theorem 1. *The above four properties are equivalent.*

Proof. Since $\rightarrow_{R:B} \subseteq \rightarrow_{R/\!\!/B}$, $= \subseteq =_B$, and $\leftrightarrow_B \subseteq =_B$, we obviously have (2) $\Rightarrow$ (1), (2) $\Rightarrow$ (3) and (3) $\Rightarrow$ (4). The implication (1) $\Rightarrow$ (2) follows directly from the relational identity $\rightarrow_{R/\!\!/B} = (=_B; \rightarrow_{R/\!\!/B})$. Since $=_B$ is the reflexive-transitive closure of $\leftrightarrow_B$, the implication (4) $\Rightarrow$ (3) follows easily by induction on the lenght of the chain of $\leftrightarrow_B$ steps. The proof that (3) $\Rightarrow$ (2) is summarized in the diagram below.



□

The above theorem has as an immediate consequence the *equi-termination*[3] of $\rightarrow_{R/\!\!/B}$ and $\rightarrow_{R:B}$.

Corollary 1. *Under any of* (1)–(4), $\rightarrow_{R/\!\!/B}$ *terminates iff* $\rightarrow_{R:B}$ *terminates.*

[3]For the concrete instances of $\rightarrow_{R/\!\!/B}$ and $\rightarrow_{R:B}$ where $\rightarrow_{\{R\}}$ is interpreted as the R-rewrite relation $\rightarrow_R$ for R a set of unsorted and unconditional rewrite rules, $=_B$ is interpreted as the B-equality relation for B a set of regular and linear equations (see Section 2.1), $\rightarrow_{R/\!\!/B}$ is interpreted as $\rightarrow_{R/B}$, and $\rightarrow_{R:B}$ is interpreted as $\rightarrow_{R,B}$, equi-termination of of $\rightarrow_{R/\!\!/B}$ and $\rightarrow_{R:B}$ under the **Completeness** assumption has been proved in [26].

PROOF. Since $\rightarrow_{R:B} \subseteq \rightarrow_{R/\!\!/B}$, the ($\Rightarrow$) part is obvious. To prove the ($\Leftarrow$) part, assume that $\rightarrow_{R:B}$ terminates but $\rightarrow_{R/\!\!/B}$ does not, so that we have an infinite sequence

$$t_0 \rightarrow_{R/\!\!/B} t_1 \rightarrow_{R/\!\!/B} t_2 \ldots t_n \rightarrow_{R/\!\!/B} t_{n+1} \ldots$$

because of the **Bisimulation** property we then obtain an infinite sequence

$$t_0 \rightarrow_{R:B} t'_1 \rightarrow_{R:B} t'_2 \ldots t'_n \rightarrow_{R:B} t'_{n+1} \ldots$$

with $t_i =_B t'_i$, contradicting the termination of $\rightarrow_{R:B}$. □

Another important consequence of the above theorem is that the relation $\rightarrow_{R:B}$ can be *directly used to rewrite in equivalence classes*, since **Completeness** gives us the equivalence:

$$[t]_B \rightarrow_{R/\!\!/B} [t']_B \quad \Leftrightarrow \quad (\exists u)\, t \rightarrow_{R:B} u \ \wedge \ u =_B t'.$$

When the abstract relations $\rightarrow_{R/\!\!/B}$ and $\rightarrow_{R:B}$ are interpreted as the rewrite relations $\rightarrow_{R/B}$ and $\rightarrow_{R,B}$ with R finite and unconditional and there is a finitary B-matching algorithm, the set of $u's$ such that $t \rightarrow_{R,B} u$ is *finite* and can be effectively computed. Therefore, one can effectively describe the finite set of one-step rewrites $[t]_B \rightarrow_{R/B} [t']_B$ from $[t]_B$.

2.1. Discussion

The motivation behind the above four equivalent notions of *strict coherence* when we instantiate the abstract relations $\rightarrow_{\{R\}}$, $\rightarrow_{R/\!\!/B}$, and $\rightarrow_{R:B}$, as the concrete relations $\rightarrow_R$, $\rightarrow_{R/B}$, and $\rightarrow_{R,B}$, is twofold: (i) they are in practice the right notions for rewriting with rules R modulo a set B of *regular and linear* equations; and (ii) if one drops from B either the linearity or the regularity requirements, then various serious anomalies make rewriting modulo B highly problematic. In what follows I assume some basic familiarity with unsorted Σ-terms as sketched in Example 1. The reader may find any missing notational details in Section 3.

An equation $u = v$ is *regular* iff $\mathcal{V}ar(u) = \mathcal{V}ar(v)$, that is, both sides have the exact same variables. A term t is *linear* iff each of its variables occurs only once (at a single position) in t. An equation $u = v$ is *linear* iff both u and v are linear. The nilpotency equation $x * x = 0$ is neither regular nor linear.

Suppose that B has a non-regular equation $u = v$ with, say, $x \in \mathcal{V}ar(v) - \mathcal{V}ar(u)$ and with $n \geq 1$ occurrences of x in v. Let $l \rightarrow r$ be *any* rewrite rule in R. Then the relation $\rightarrow_{R/B}$ is non-terminating, since we have the looping rewrite $u \rightarrow^n_{R/B} u$ given by:

$$u =_B v(x \mapsto l) \rightarrow^n_R v(x \mapsto r) =_B u.$$

where $(x \mapsto l)$ denotes the obvious substitution of x by l. This makes rewriting modulo non-regular equations B hopeless in practice.

Even assuming regularity, non-linearity has also adverse consequences. The first immediate consequence is that $\rightarrow_{R,B}$ can no longer *bisimulate* $\rightarrow_{R/B}$, so that we can no longer use $\rightarrow_{R,B}$ to efficiently achieve the effect of rewriting in B-equivalence classes. Consider, for example, an equation $u = v$ where x occurs once in u but $n > 1$ times in v. Let $l \rightarrow r$ be a rewrite rule in R, and consider the term $v(x \mapsto l)$. We obviously have $v(x \mapsto l) \rightarrow_{R/B} v(x \mapsto r)$, since we have

$$v(x \mapsto l) =_B u(x \mapsto l) \rightarrow_R u(x \mapsto r) =_B v(x \mapsto r)$$

but with $\rightarrow_{R,B}$ in general we may need to take $n > 1$ steps to get $v(x \mapsto l) \rightarrow^n_{R,B} v(x \mapsto r)$, so that **Bisimulation** is lost, and with it the ability to use $\rightarrow_{R,B}$ to get the effect of rewriting on B-equivalence classes. For example, for $B = \{f(x) \cdot f(x) = f(x)\}$ and $R = \{a \rightarrow b\}$, we have $f(a) \cdot f(a) \rightarrow_{R/B} f(b) \cdot f(b)$, but we only have $f(a) \cdot f(a) \rightarrow^2_{R,B} f(b) \cdot f(b)$.

What is worse, if linearity is dropped, *equi-termination* no longer holds. Consider, for example, B consisting of a single idempotency equation $x \cdot x = x$, and R having the single rule $a \rightarrow b$. Since the equation and the rule have no symbols in common, this rule is *coherent* in the sense of [38]. Note that the relation $\rightarrow_{R,B}$ is *terminating*, since the number of occurrences of a in a term decreases at least by 1 each time it is used, and $\rightarrow_{R,B}$ cannot be applied to a term unless some a occurs in it. However, $\rightarrow_{R/B}$ is non-terminating, as witnessed by the sequence:

$$a =_B a \cdot a \rightarrow_R b \cdot a =_B b \cdot (a \cdot a) \rightarrow_R b \cdot (b \cdot a) =_B b \cdot (b \cdot (a \cdot a)) \rightarrow_R \ldots$$

All the above-mentioned anomalies are well-known. In fact, many things can go wrong in the non-regular and/or non-linear cases. For good sources of "counterexamples" that show how unreliable one's intuition can become in such treacherous waters see, e.g., [59, 43].

Rewriting modulo B for general sets of equations B, and the relations $\rightarrow_{R/B}$ and $\rightarrow_{R,B}$, have been thoroughly studied (see, e.g., [61, 38, 3, 39]), and to deal with general sets of equations B more general notions of coherence (resp. E-commuting) and local coherence (resp. locally E-commuting) have been proposed, e.g., in [37, 38] (resp. [40]). All the just-mentioned treatments are unconditional. In Section 3 I generalize rewriting modulo B (with no restrictions on B) to *conditional order-sorted* rewrite theories. However, for all the reasons mentioned above about the difficulties with non-regular and/or non-linear axioms B, in Sections 4, 5, and parts of Section 6, I focus on the simpler and much better behaved case of equations B that are both regular and linear. This seems to be the most practical case. Yet, except for [61] and a few papers afterwards, e.g., [26, 20, 21], this subcase seems comparatively less studied than rewriting modulo an arbitrary set of equations B.

Example 2. *(Exclusive Or). The equational theory of* exclusive or *has a signature* $\Sigma_\oplus$ *with a constant* 0 *(understood as "false"), and a binary operator* $\oplus$*, and has equations* $B \uplus G$*, where* B *are the (regular and linear) equational axioms of associativity* $x \oplus (y \oplus z) = (x \oplus y) \oplus z$ *and commutativity* $x \oplus y = y \oplus x$ *(AC)*

for $\oplus$, and G are the identity equation $x \oplus 0 = x$ and the nilpotency equation $x \oplus x = 0$. By orienting the equations G from left to right as rewrite rules $\vec{G} = \{x \oplus 0 \to x, x \oplus x \to 0\}$, we can associate to the exclusive or equational theory the rewrite theory $\mathcal{R}_{\oplus} = (\Sigma_{\oplus}, B, \vec{G})$.

However, this theory is not *strictly coherent. This can be easily shown by means of the term $x \oplus (y \oplus x)$, since we obviously have $x \oplus (y \oplus x) =_B (x \oplus x) \oplus y \to_{\vec{G}} 0 \oplus y$, so that $x \oplus (y \oplus x) \to_{\vec{G}/B} 0 \oplus y$, but the term $x \oplus (y \oplus x)$* cannot *be rewritten by the relation $\to_{\vec{G},B}$, so that the* **Completeness** *property, and therefore the* **Strict Coherence** *property break down.*

In Section 4 I will show in detail that extending $\mathcal{R}_{\oplus}$ by just adding to $\vec{G}$ the extra rule $y \oplus (x \oplus x) \to y \oplus 0$ strict coherence is attained. Since it is easy to check that this extended rewrite theory is confluent and terminating modulo B, it will then follow as a very special case of the more general Church-Rosser Theorem 6 that this extension of $\mathcal{R}_{\oplus}$ provides a decision procedure *for the equational theory of exclusive or (see Example 14 for details).*

3. Conditional Order-Sorted Rewriting Modulo Axioms

In this section unsorted and unconditional rewriting modulo B as defined in Example 1 is generalized, in *two* orthogonal dimensions, to rewriting modulo B in conditional order-sorted rewrite theories. The first dimension of generality has to do with typing. Most treatments of rewriting deal with the unsorted case. This is a *special case* of many-sorted rewriting, which, in turn, is a *special case* of order-sorted rewriting. The obvious logical asymmetry is that *all* results about order-sorted rewriting apply automatically to many-sorted and unsorted rewriting, but *not conversely*. Since most applications to declarative programming and algebraic specification are typed, this makes unsorted treatments unusable in practice for such applications, so that the extra generality is not at all an extravagant caprice but a necessity. The second dimension of generality consists in allowing rewrite rules R that are *conditional* and of the most general kind possible[4] —namely (the natural generalization of) oriented 4-CTRSs [60]. Except for [27, 44, 53, 6, 9, 18, 21], all treatments of equational rewriting I am aware of deal only with unsorted and unconditional rewriting. This makes such treatments unusable in practice for conditional theories. In fact, conditional rewriting substantially increases the expressive power of a language and is for this reason supported by all the well-known rewriting-based languages such as OBJ [32], ASF+SDF [69], ELAN [7], CafeOBJ [25] and Maude [11].

[4] I give in Section 6.2 additional results that apply to the —also very general, but with much better executability properties— special case of (the natural generalization of) strongly deterministic 3-CTRSs [60], whose rules can have extra variables in their condition and right-hand side, but they are instantiated incrementally by matching in the process of solving the condition.

3.1. Preliminaries on Order-Sorted Signatures, Terms and Equations

I follow the standard terminology and notation of term rewriting (see, e.g., [60, 2, 17, 68, 16]) and order-sorted algebra [31, 54]. Readers familiar with such terminology and notation can skip this section and proceed to Section 3.2. Let me recall the notions of order-sorted signature, term, substitution and equation. An order-sorted signature $(\Sigma, S, \leq)$ consists of a poset of sorts $(S, \leq)$ and an $S^* \times S$-indexed family of function symbols sets $\Sigma = \{\Sigma_{s_1 \ldots s_n, s}\}_{(s_1 \ldots s_n, s) \in S^* \times S}$, where $f \in \Sigma_{s_1 \ldots s_n, s}$ is displayed as $f : s_1 \ldots s_n \to s$. To avoid ambiguous terms, Σ should first of all be *sensible*, in the sense that for any two typings $f : s_1 \ldots s_n \to s$ and $f : s'_1 \ldots s'_n \to s'$ of an n-argument function symbol f, if s_i and s'_i are in the same connected component of $(S, \leq)$ for $1 \leq i \leq n$, then s and s' are also in the same connected component; this provides the right notion of *unambiguous* signature at the order-sorted level.[5] Throughout, Σ is also assumed to be *preregular* [31], so that each term t has a least sort, denoted $ls(t)$ (see Example 3 below). Σ is also assumed to be *kind-complete*, that is, for each sort $s \in S$ its connected component in the poset $(S, \leq)$ has a top sort, denoted $[s]$, and for each $f \in \Sigma_{s_1 \ldots s_n, s}$ there is also an $f \in \Sigma_{[s_1] \ldots [s_n], [s]}$. An order-sorted signature can always be extended to a kind-complete one (see Example 3 below). Maude [11] automatically checks preregularity and adds a new "kind" sort $[s]$ at the top of the connected component of each sort $s \in S$ specified by the user, and automatically lifts each operator to the kind level.

Given an S-sorted set $\mathcal{X} = \{\mathcal{X}_s\}_{s \in S}$ of *mutually disjoint* countably infinite sets of variables, $\mathcal{T}_\Sigma(\mathcal{X})_s$ denotes the set of Σ-terms of sort s with variables in $\mathcal{X}$, and $\mathcal{T}_\Sigma(\mathcal{X})$ denotes, ambiguously, both the S-sorted set of all Σ-terms with variables in $\mathcal{X}$, and (provided Σ is sensible) the *free* Σ-algebra on those variables. Similarly, $\mathcal{T}_\Sigma$ denotes both the S-sorted set of all *ground* Σ-terms that have no variables, and (provided Σ is sensible) the initial Σ-algebra. Here is the inductive construction of $\mathcal{T}_\Sigma(\mathcal{X})$:

- $\mathcal{X}_s \subseteq \mathcal{T}_\Sigma(\mathcal{X})_s$
- if $f \in \Sigma_{s_1 \ldots s_n, s}$ and $t_i \in \mathcal{T}_\Sigma(\mathcal{X})_{s_i}$, $1 \leq i \leq n$, then $f(t_1, \ldots, t_n) \in \mathcal{T}_\Sigma(\mathcal{X})_s$ (including the case of constants, where $n = 0$)
- if $s \leq s'$ then $\mathcal{T}_\Sigma(\mathcal{X})_s \subseteq \mathcal{T}_\Sigma(\mathcal{X})_{s'}$.

Σ is said to have *non-empty sorts* iff $\mathcal{T}_{\Sigma,s} \neq \emptyset$ for each sort s. $\mathcal{V}ar(t)$ denotes the set of variables appearing in term t. A *substitution* is an S-sorted[6] mapping $\sigma : \mathcal{X} \longrightarrow \mathcal{T}_\Sigma(\mathcal{X})$. We define its *domain*, denoted $dom(\sigma)$, as the set $dom(\sigma) =$

[5] Ambiguity when Σ is not sensible can be illustrated even when Σ is many-sorted. Consider Σ with sorts A, B, C, a constant a of sort A, and unary operators $f : A \to B$, $f : A \to C$. Σ is not sensible because of the operators $f : A \to B$, $f : A \to C$, and is ambiguous, because the term $f(a)$ has *two* completely unrelated sorts, namely, B and C. In the order-sorted case, "unrelated" means "in different connected components."

[6] Since σ is S-sorted, if x has sort s, then $\sigma(x) \in \mathcal{T}_\Sigma(\mathcal{X})_s$. But since $s' \leq s$ implies $\mathcal{T}_\Sigma(\mathcal{X})_{s'} \subseteq \mathcal{T}_\Sigma(\mathcal{X})_s$, it may very well be that $\sigma(x)$ can also be typed with sorts lower than s.

$\{x \in \mathcal{X} \mid \sigma(x) \neq x\}$. The homomorphic extension $\sigma : \mathcal{T}_\Sigma(\mathcal{X}) \longrightarrow \mathcal{T}_\Sigma(\mathcal{X})$ is also denoted σ, and its application to a term t is denoted $t\sigma$. $\mathcal{P}(t)$ denotes the set of positions [16] of a Σ-term t, and $t|_p$ denotes the *subterm of* t *at position* $p \in \mathcal{P}(t)$. Similarly, $\mathcal{P}_\Sigma(t)$ denotes the *non-variable positions* of t, that is, those $p \in \mathcal{P}(t)$ such that $t|_p \notin \mathcal{V}ar(t)$. A term t with its subterm $t|_p$ replaced by the term t' is denoted by $t[t']_p$.

For a Σ-equation $u = v$ to be well-formed, the sorts of u and v should be in the same connected component of $(S, \leq)$. For B a set of Σ-equations, $=_B$ denotes the provable B-equality relation[7] [31, 54], and $[t]_B$ denotes the equivalence class of t modulo $=_B$. Given a set of Σ-equations B, a substitution σ is a *B-unifier* of an equation $t = t'$ iff $t\sigma =_B t'\sigma$; let $MGU_B(t = t')$ denote a complete set of most general B-unifiers.[8] Likewise, a substitution σ is a *B-match* from t to t' iff $t' =_B t\sigma$. Since the practical interest is in implementable uses of rewriting modulo B, from Section 4 onwards I will assume that B has a finitary B-matching algorithm; that is, an algorithm generating a complete finite set of B-matches from t to t', denoted $Match_B(t, t')$; that is, for any B-match σ there is a $\tau \in Match_B(t, t')$ such that for all $x \in \mathcal{V}ar(t)$ $\sigma(x) =_B \tau(x)$.

An equation $u = v$ is called *sort-preserving* iff for each S-sorted substitution θ we have $ls(u\theta) = ls(v\theta)$. Using substitutions that specialize variables to smaller sorts it can be easily checked whether an equation is sort-preserving.

Example 3. *Several of the above-mentioned order-sorted notions can be illustrated with a simple example. Let Σ have sorts: Nat of natural numbers, NzNat of non-zero naturals, Even of even numbers, and Odd of odd numbers, with subsort inclusions $NzNat, Even, Odd < Nat$, a constant 0 of sort Even, a "subsort-polymorphic" successor operator with typings $s : Nat \to NzNat$, $s : Even \to Odd$, and $s : Odd \to Even$, a predecessor operator $p : NzNat \to Nat$, and an addition function $\_ + \_ : Nat\ Nat \to Nat$.*

The data type of natural numbers with successor, predecessor and addition functions can be defined as the initial algebra $\mathcal{T}_{\Sigma/E}$ of the order-sorted equational theory (Σ, E), where E has just three equations: $p(s(n)) = n$, $n + 0 = n$, and $n + s(m) = s(n + m)$, where n and m are variables of sort Nat.

Since it is easy to check that Σ is a sensible signature, mathematically speaking the initial algebra $\mathcal{T}_{\Sigma/E}$ is a perfectly correct model of the natural numbers in Peano notation for those functions. However, at the operational semantics level this specification of $\mathcal{T}_{\Sigma/E}$ is problematic. *For example, the term $s(s(0))$ has* three *sorts, namely, NzNat using $s : Nat \to NzNat$, Even using the fact that 0 has sort Even, and the typings $s : Even \to Odd$ and $s : Odd \to Even$, and of course Nat using either of the inclusions $NzNat, Even < Nat$.*

But since the sorts NzNat and Even are incomparable *in the above-defined*

[7] Besides the proof systems in [31, 54], the reader can find an explicit definition of $=_B$ as part of the inference system for the relations $\to_{R/B}$ and $\to^\star_{R/B}$ in Section 3.2, and a proof system for *conditional* equality in Section 6.1.

[8] This exactly means that for each B-unifier α of $t = t'$ there exists a B-unifier $\beta \in MGU_B(t = t')$ and a substitution γ such that for each variable x we have $\alpha(x) =_B \gamma(\beta(x))$.

sort hierarchy, this means that Σ *is* not *preregular, which is a bad situation both for parsing terms (no "unique least parse" exists), and for term rewriting purposes. For example, suppose we now define a* non-zero predicate *function* $nz : Nat \to Nat$ *with two equations:* $nz(0) = 0$*, and* $nz(n') = s(0)$*, where the variable* n' *has sort NzNat. How can we now evaluate the predicate* $nz(s(s(0)))$ *by using the above equations oriented as left-to-right rewrite rules* $nz(0) \to 0$*, and* $nz(n') \to s(0)$*? Well, we* cannot *if we have parsed* $s(s(0))$ *as having sort Even. That is, we need to* compute all the lowest parses *for* $s(s(0))$*, i.e., NzNat and Even, to see if with one of them* $nz(s(s(0)))$ *can match the lefthand side* $nz(n')$ *of rule* $nz(n') \to s(0)$*, which is inefficient and cumbersome.*

All these problems can be solved by adding a new sort of non-zero even numbers NzEven, three more subsort declarations: $NzEven < Even$*,* $Odd < NzNat$ *and* $NzEven < NzNat$*, and replacing the declaration* $s : Odd \to Even$ *by the more precise declaration* $s : Odd \to NzEven$*. Then, the so-revised signature, let us call it* Σ'*, becomes* preregular*, that is, every term* t *has now a* unique least sort $ls(t)$*.*

There is a remaining issue. Consider the terms $p(0+s(0))$ *and* $p(p(s(s(0))))$*. The problem with these, perfectly reasonable terms is that they* do not even exist *in the term algebra* $\mathcal{T}_{\Sigma'}$*. That is, these terms cannot even be* parsed*. We would like to* give them the benefit of the doubt *and await until they are simplified with the equations for* p *and* $+$ *oriented as rewrite rules* $p(s(n)) \to n$*,* $n+0 \to n$*, and* $n + s(m) \to s(n + m)$*, to see if they are acceptable or not. They indeed are, because both can be simplified to the term* 0*. Instead, the term* $p(p(s(0)))$ *simplifies to* $p(0)$*, which is an informative* error term*.*

The handling of such, dubious terms can be easily achieved by extending Σ' *to be* kind-complete *by adding a "kind" supersort* $[Nat]$ *above Nat, i.e., with a subsort inclusion* $Nat < [Nat]$*, and lifting all operators to the kind level by adding new declarations:* $s, p, nz : [Nat] \to [Nat]$*, and* $\_ + \_ : [Nat]\ [Nat] \to [Nat]$*. In this way, the above, simplified error term* $p(0)$ *becomes a term of "kind"* $[Nat]$*, which does not have a sort (is not defined) in the original signature* Σ' *defining our data. Maude [11] automatically checks preregularity, and kind-completes any user-defined signature to give the benefit of the doubt to dubious terms like* $p(0+s(0))$*,* $p(p(s(s(0))))$*, and* $p(p(s(0)))$*, so that their final status can be settled after simplifying them with the specification's equations oriented as rewrite rules.*

3.2. Conditional Order-Sorted Rewriting Modulo B

Conditional order-sorted rewriting modulo axioms goes back to [27, 44], where it was assumed that all variables in a rule's condition and righthand side appeared also in its lefthand side. Various extensions to conditional rewriting with membership conditions were developed, including, in the case without axioms, [14, 15, 8], and modulo axioms [18], where conditional R, B-rewriting for *deterministic* theories (see Definition 6) was defined.

I present below a detailed semantics for conditional order-sorted rewriting modulo axioms B. Since the goal is to define such rewriting in full generality, *no assumptions are made on* B in this section. However, in Section 4 various assumptions on B, including regularity and linearity, will be explicitly stated.

The key notion is that of a *conditional order-sorted rewrite theory*, that is, a triple $\mathcal{R} = (\Sigma, B, R)$, with Σ an order-sorted signature, B a set of Σ-equations, and R a set of conditional rewrite rules of the form $l \to r$ *if* $\bigwedge_{i=1..n} u_i \to v_i$, with *no restrictions* on the variables of l, r, or those of the u_i and v_i.

The meaning of $\mathcal{R} = (\Sigma, B, R)$ *need not be equational*; that is, the rules R need not be understood as conditional equations that have been oriented as rewrite rules. Of course for some applications R *can* be understood equationally; for example, Maude's *functional modules* [11] are conditional equational theories that are executed as conditional rewrite theories $\mathcal{R} = (\Sigma, B, R)$ by rewriting modulo B. I further discuss the equational meaning of conditional order-sorted rewrite theories in Section 6. However, in many other applications $\mathcal{R} = (\Sigma, B, R)$ specifies a *concurrent system* whose states are B-equivalence classes $[t]_B$ with t a ground Σ-term [53]. The rules R then specify the possible concurrent transitions of such a system. Maude's *system modules* [11] give such a concurrent system semantics to a theory $\mathcal{R} = (\Sigma, B, R)$, which may, for example, specify the semantics of a concurrent programming language, a process calculus, a network protocol, or a cyber-physical or cell biology system [55].

Something not entirely obvious is how the rewrite relations $\to_{R/B}$ and $\to_{R,B}$ should be *defined* for a conditional rewrite theory $\mathcal{R} = (\Sigma, B, R)$. As explained below, the naive generalization of the unconditional relations would actually be wrong. The most satisfactory way is by means of *inference systems*.[9] I give below an inference system defining both $\to_{R/B}$ and $\to^{\star}_{R/B}$. To ease the exposition I assume a preregular and kind-complete Σ with *non-empty sorts* (i.e., $\mathcal{T}_{\Sigma,s} \neq \emptyset$ for each $s \in S$) . It is of course possible to drop the non-empty sorts requirement, but then all the implicit universal quantification of equations and rules must become *explicit*. This is because without explicit quantification deductions when some sorts are empty can easily become *unsound*. Therefore, sound inference systems for order-sorted conditional equational logic (and even for many-sorted logic) make universal quantification explicit [29, 56, 31, 54]. Similarly, without assuming non-empty sorts, sound inference systems for rewriting logic based on many-sorted, order-sorted, or membership equational logic typing disciplines

[9] Since a single step of conditional rewriting may involve many rewriting steps to evaluate the corresponding instance of the rule's condition, defining the conditional rewrite relation is notoriously complicated, even for *standard* rewriting (without axioms). Perhaps the least complicated classical definition in the standard case is the one given in the advanced textbook [60], where the conditional rewriting relation $\to_R$ is defined as an infinite union of $\bigcup_{n\in\mathbb{N}} \to_{R,n}$ of unconditional rewrite relations. The case of conditional rewriting modulo axioms B is even more complicated. One notorious additional complication is caused by the somewhat subtle fact that, as I will explain in what follows, the reflexive-transitive closure of the relation $\to_{R,B}$ is actually *too weak* to express the satisfaction of a rule's condition. For all these reasons, generalizing a classical definition of standard conditional rewriting such as the one in [60] to the modulo case would not just be *a priori* quite complicated: as I show in Section 3.3, it could be quite treacherous, because the straightforward generalization of the definition in [60] would yield the wrong definition. All this suggests the much simpler possibility of adopting an *inference system approach*, where everything can be spelled out with just three inference rules. This is analogous to why Structural Operational Semantics [62], which also relies on inference systems, is widely used to define computation relations because of its simplicity.

must make universal quantification explicit [9].

Let $\mathcal{R} = (\Sigma, B, R)$ satisfy the above assumptions. Given $t, t' \in \mathcal{T}_\Sigma(\mathcal{X})$ define $t \leftrightarrow_B t'$ iff there is a position p in t, an (S-sorted) substitution σ, and either $u = v \in B$ or $v = u \in B$ such that $t = t[u\sigma]_p$ and $t' = t[v\sigma]_p$. The *order-sorted B-equality relation* $=_B$ is then defined as the reflexive-transitive closure $(\leftrightarrow_B)^*$. The inference rules defining both $\rightarrow_{R/B}$ and $\rightarrow^\star_{R/B}$ then are:

- **Reflexivity**. For each $t, t' \in \mathcal{T}_\Sigma(\mathcal{X})$ such that $t =_B t'$, $\quad \dfrac{}{t \rightarrow^\star_{R/B} t'}$
- **Replacement**. For $l \rightarrow r \;\; \textit{if} \;\; u_1 \rightarrow v_1 \wedge \ldots \wedge u_n \rightarrow v_n$ a rule in R, $t, u, v \in \mathcal{T}_\Sigma(\mathcal{X})$, $p \in \mathcal{P}(t)$, and θ a substitution, such that $u =_B t[l\theta]_p$ and $v =_B t[r\theta]_p$,
$$\frac{u_1\theta \rightarrow^\star_{R/B} v_1\theta \;\; \ldots \;\; u_n\theta \rightarrow^\star_{R/B} v_n\theta}{u \rightarrow_{R/B} v}$$
- **Transitivity**. For $t_1, t_2, t_3 \in \mathcal{T}_\Sigma(\mathcal{X})$,
$$\frac{t_1 \rightarrow_{R/B} t_2 \qquad t_2 \rightarrow^\star_{R/B} t_3}{t_1 \rightarrow^\star_{R/B} t_3}$$

In general, the relation $u \rightarrow_{R/B} v$ may be undecidable, since checking whether $u \rightarrow_{R/B} v$ holds involves searching through the possibly infinite equivalence class $[u]_B$ to find a representative that can be rewritten with R and checking, furthermore, that the result u' of such rewriting belongs to the equivalence class $[v]_B$. For this reason, and for greater efficiency, a much simpler relation $\rightarrow_{R,B}$ is defined, which in the unconditional case becomes decidable for R finite if a finitary B-matching algorithm exists. However, in the conditional case, even with such a B-matching algorithm, the relation $\rightarrow_{R,B}$ may still be undecidable due to the general undecidability of reachability by rewriting. As already mentioned, the key idea about $\rightarrow_{R,B}$ is to replace general B-equalities of the form $u =_B t[l\theta]_p$ by a matching B-equality $t|_p =_B l\theta$ with the subterm actually being rewritten. This completely eliminates any need for searching for a redex in the possibly infinite equivalence class $[u]_B$. Here is the inference system defining both $\rightarrow_{R,B}$ and $\rightarrow^\star_{R,B}$ under the same assumptions on $\mathcal{R}$.

- **Reflexivity**. For each $t, t' \in \mathcal{T}_\Sigma(\mathcal{X})$ such that $t =_B t'$, $\quad \dfrac{}{t \rightarrow^\star_{R,B} t'}$
- **Replacement**. For $l \rightarrow r \;\; \textit{if} \;\; u_1 \rightarrow v_1 \wedge \ldots \wedge u_n \rightarrow v_n$ a rule in R, $t \in \mathcal{T}_\Sigma(\mathcal{X})$, $p \in \mathcal{P}(t)$, and θ a substitution, such that $t|_p =_B l\theta$,
$$\frac{u_1\theta \rightarrow^\star_{R,B} v_1\theta \;\; \ldots \;\; u_n\theta \rightarrow^\star_{R,B} v_n\theta}{t \rightarrow_{R,B} t[r\theta]_p}$$

- **Transitivity**. For $t_1, t_2, t_3 \in \mathcal{T}_\Sigma(\mathcal{X})$,

$$\frac{t_1 \to_{R,B} t_2 \qquad t_2 \to^\star_{R,B} t_3}{t_1 \to^\star_{R,B} t_3}$$

Note that the only difference between the two inference systems just defined is that the **Replacement** rule defining $\to_{R,B}$ is *more restrictive* than the **Replacement** rule defining $\to_{R/B}$. This, of course, is the whole point of defining $\to_{R,B}$, i.e., its simpler implementation and greater efficiency. But this means that, in general, the containment $\to_{R,B} \subseteq \to_{R/B}$ implied by the more restrictive definition of $\to_{R,B}$ may be *strict*. The purpose of later adding strict coherence requirements to $\to_{R,B}$ is precisely to make such a, usually strict, containment harmless. More specifically, the main goal is to find strict coherence conditions making the deductive power of both systems with respect to their respective relations $\to^\star_{R/B}$ and $\to^\star_{R,B}$ the *same*, i.e., $\to^\star_{R/B} = \to^\star_{R,B}$, even though we will still typically have $\to_{R,B} \subset \to_{R/B}$, which is a *wanted* strict containment for efficiency purposes (see Proposition 1).

Example 4. *(Idempotent Semigroups). The theory of* semigroups *has a single binary operator, which can be denoted with empty syntax* $\_\_$ *(juxtaposition), and the singleton set A of equational axioms given by the associativity axiom* $x(yz) = (xy)z$. *A semigroup is called* idempotent *iff it satisfies the equation* $xx = x$. *Here is a simple, yet non-trivial question: is the equational theory of idempotent semigroups* decidable*?*

A simple answer can be given using unconditional rewriting modulo axioms for the special case of commutative *semigroups: the rewrite theory* $\mathcal{R}_{Idemp/AC} = (\{\_\_\}, AC, \{xx \to x\})$*, where AC is obtained by adding to A the commutativity axiom* $xy = yx$ *is not strictly AC-coherent, but using the methods of Section 4.1 and Corollary 3 it can be completed into the strictly AC-coherent theory* $\overline{\mathcal{R}}_{Idemp/AC} = (\{\_\_\}, AC, \{xx \to x, y(xx) \to yx\})$. *Since both rules are size-decreasing and the AC-axioms are size-preserving, the theory* $\overline{\mathcal{R}}_{Idemp/AC}$ *is* terminating. *Futhermore, as a special case of Theorem 7 below,* $\overline{\mathcal{R}}_{Idemp/AC}$ *is locally confluent modulo AC and, by termination,* confluent *modulo AC by easy inspection of its critical pairs. Therefore, as a special case of the more general Church-Rosser Theorem 6 below, the theory of idempotent commutative semigroups is* decidable *by rewriting modulo AC with the theory* $\overline{\mathcal{R}}_{Idemp/AC}$*: two terms are provably equal iff their canonical froms are equal modulo AC.*

The general case of idempotent semigroups is harder. To the best of my knowledge no confluent and terminating set of unconditional rules modulo A is known for this theory, and the paper [65] conjectures that no such (finite) set of rules exists. However, the paper [65] defines a special form of unsorted conditional rewriting that, although somewhat ad-hoc because of involving two different equality relations: the axioms A for the rules and the axioms AC for the condition, can be reformulated in standard conditional equational deduction and conditional rewriting terms using an order-sorted *specification as follows (see also the specification in [11], which is here simplified). We can specify the*

theory of idempotent semigroups as a conditional equational theory expressible in Maude with self-explanatory syntax: `eq` *(equation),* `ceq` *(conditional equation).*

```
fth IDEMPOTENT-SEMIGROUP is
  protecting QID .
  sorts List Set Prop .
  subsorts Qid < List .                        *** Qid's as variables

  op _ _ : List List -> List [assoc] .         *** list concatenation
  op _,_ : Set Set -> Set [assoc comm] .       *** set union
  op {_} : List -> Set .                       *** set of a list
  op tt : -> Prop .                            *** true
  op _~_ : Set Set -> Prop .                   *** equality predicate

  var  S : Set .
  vars L P Q : List .

  eq L L = L .
  ceq L P Q = L Q if {L} ~ {Q} = tt  /\ {L P} ~ {L} = tt .
  eq S , S = S .
  eq {L P} = {L} , {P} .
  eq S ~ S = tt .
endfth
```

Let me explain this equational theory in some detail. By the Theorem of Constants (see, e.g., Lemma 2.3.2 in [35]), we can reduce E-equality between Σ-terms with variables X to E-equality between the exact same terms viewed as ground *terms in the extended signature $\Sigma(X)$, where the variables X have been added as fresh new constants. This is achieved in the above theory by taking X to be the set of quoted identifiers of sort* `Qid` *in Maude's* `QID` *module, and giving the subsort declaration* `Qid < List`*, so that the ground terms of sort* `List` *are exactly the terms with variables in the theory of semigroups. The A axiom is then given by the* `assoc` *declaration for the juxtaposition operator. Likewise, the AC axioms for the set union operator $\_,\_$ are given by the* `assoc comm` *declaration. Let B denote the union of these A and AC axioms. The Maude parser uses these axioms to disregard parentheses. For example, the two parses $L(PQ)$ and $(LP)Q$ of LPQ are considered identical modulo B. Note that the operator $\{\_\}$ turns a list into a finite set. For example,* `{'x 'x 'x 'y 'y 'z}` *becomes the set* `{'x},{'y},{'z}`*. Note, finally, that the operator $\_ \sim \_$ defines an equality predicate for finite sets, so that two set expressions S and S' are provably equal iff we can prove $S \sim S' = tt$. This equality predicate is introduced to make the two equations in the condition of the conditional equation nicely orientable from left to right; this is a well-known method (see the transformation $\mathcal{R} \mapsto \mathcal{R}^{\equiv}$ at the end of Section 6.1) for turning an equational condition like $\{L\} = \{Q\}$ into an equivalent rewrite condition like $\{L\} \sim \{Q\} \to tt$.*

The idempotency equation is of course the first equation. The problem is that this equation, when oriented as a rule $LL \to L$, is not *confluent modulo B. For example, the term* `'x 'y 'x 'y 'z 'y 'x 'y 'z` *can be rewritten with*

$LL \to L$ *modulo* B *to the terms* `'x 'y 'z` *and* `'x 'y 'z 'y 'x 'y 'z`*, which cannot be further rewritten modulo* B *with* $LL \to L$*. The point of adding the second conditional equation is that, when oriented as the conditional rule*

$$L(PQ) \to LQ \;\; \textit{if} \;\; \{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$$

it can be used, together with the idempotency rule, to rewrite the second canonical form `'x 'y 'z 'y 'x 'y 'z` *to the simpler one* `'x 'y 'z` *in just two* R/B*-rewrite steps. Of course, the non-obvious question of whether adding this extra conditional equation can identify more terms than those provable using only the original idempotency equation is* the *crucial question. That this is so, so that for purposes of equational provability of terms of sort* `List` *adding or not this conditional equation makes no difference, is proved in detail by Siekmann and Szabo in [65].*

Consider the rewrite theory $\mathcal{R}_{Idemp.Sgr} = (\Sigma, B, R)$ *with* Σ *the order-sorted signature defined by the above theory (viewing Qids exclusively as new constants),* B *the above-declared axioms for juxtaposition and set union, and* R *the above equations oriented as left-to-right rewrite rules, with the conditional equation oriented as explained above. This theory is* not *strictly* B*-coherent when we interpret the abstract relations* $\to_{R/\!\!/B}$ *and* $\to_{R:B}$ *as, respectively,* $\to_{R/B}$ *and* $\to_{R,B}$ *(more on this interpretation in Section 4).*

To show this it is enough to show that the **Completeness** *property fails. That is, we will be done if we can show that, for example, the term*

`('v 'x)(('y 'z)(('y 'x)('y 'z)))`

can be rewritten with the $\to_{R/B}$ *relation but* not *with the* $\to_{R,B}$ *relation. For the sake of readability I will omit all quotes in quoted identifiers and write the above term as* $(vx)((yz)((yx)(yz)))$*. Note that we have the* B*-equality*

$$(vx)((yz)((yx)(yz))) =_B v(x(yz)(y(x(yz)))).$$

Note also that the subterm $x(yz)(y(x(yz)))$ *matches the lefhand side of the conditional rule* $L(PQ) \to LQ \;\; \textit{if} \;\; \{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$ *with substitution* $\{L \mapsto x(yz), P \mapsto y, Q \mapsto x(yz)\}$*. Therefore, provided we can satisfy the condition*

$$\{x(yz)\} \sim \{x(yz)\} \to^{\star}_{R/B} tt \wedge \{(x(yz))y\} \sim \{x(yz)\} \to^{\star}_{R/B} tt$$

we will be able to use the **Replacement** *rule to perform the rewrite*

$$(vx)((yz)((yx)(yz))) \to_{R/B} v((x(yz))(x(yz))).$$

The first conjunct $\{x(yz)\} \sim \{x(yz)\} \to^{\star}_{R/B} tt$ *can be easily discharged by first using* **Transitivity** *to transform it into the subgoals* $\{x(yz)\} \sim \{x(yz)\} \to_{R/B} tt$ *and* $tt \to^{\star}_{R/B} tt$*, and then discharging the first subgoal using the* **Replacement** *rule with rule* $S \sim S \to tt$ *and substitution* $\{S \mapsto \{x(yz)\}\}$*, and the second subgoal using* **Reflexivity**.

It is also easy to see that, by repeated application of **Transitivity** *followed by* **Replacement** *with the rule* $\{LP\} \to \{L\}, \{P\}$*, we can transform the second conjunct into the subgoal* $\{x\}, \{y\}, \{z\}, \{y\} \sim \{x\}, \{y\}, \{z\} \to^{\star}_{R/B} tt$*, where parentheses are omitted for readability. Let me illustrate the first step in this repeated application of* **Transitivity** *followed by* **Replacement**. *We apply* **Transitivity** *to transform the second conjunct* $\{(x(yz))y\} \sim \{x(yz)\} \to^{\star}_{R/B} tt$ *into the subgoals* $\{(x(yz))y\} \sim \{x(yz)\} \to_{R/B} \{x(yz)\}, \{y\} \sim \{x(yz)\}$ *and* $\{x(yz)\}, \{y\} \sim \{x(yz)\} \to^{\star}_{R/B} tt$*, and then we discharge the first subgoal by* **Replacement** *with rule* $\{LP\} \to \{L\}, \{P\}$ *and substitution* $\{L \mapsto x(yz), P \mapsto y\}$.

Finally, the subgoal $\{x\}, \{y\}, \{z\}, \{y\} \sim \{x\}, \{y\}, \{z\} \to^{\star}_{R/B} tt$ *is easily discharged as follows. We first use* **Transitivity** *to transform it into the subgoals* $\{x\}, \{y\}, \{z\}, \{y\} \sim \{x\}, \{y\}, \{z\} \to_{R/B} \{x\}, \{y\}, \{z\} \sim \{x\}, \{y\}, \{z\}$ *and* $\{x\}, \{y\}, \{z\} \sim \{x\}, \{y\}, \{z\} \to^{\star}_{R/B} tt$*. We can then discharge the first subgoal by* **Replacement** *with rule* $S, S \to S$ *and substitution* $\{S \mapsto \{y\}\}$*. Likewise, we can discharge the second subgoal by first using* **Transitivity** *to transform it into the subgoals* $\{x\}, \{y\}, \{z\} \sim \{x\}, \{y\}, \{z\} \to_{R/B} tt$ *and* $tt \to^{\star}_{R/B} tt$*, which are then respectively discharged by* **Replacement** *with rule* $S \sim S \to tt$ *and by* **Reflexivity**.

The lack of strict B*-coherence of the theory* $\mathcal{R}_{Idemp.Sgr}$ *can now be boiled down to showing that our chosen term* $(vx)((yz)((yx)(yz)))$ *is in* R, B*-normal form. This is so because: (i) the only two rules that can be applied to it are the idempotency rule* $LL \to L$ *and the conditional rule, but (ii).1 there is* no *subterm of the above term that can be shown equal modulo* B *to an instance of the lefthand side* LL *of the idempotency rule, and (ii).2 no match of the lefthand side* $L(PQ)$ *of the conditional rule can satisfy the rule's condition. This is fairly obvious for* B*-matches at the top of the term, since the term* B*-matching* L *must have* v *as a subterm, but* v *can never appear in a* B*-match for* Q*, so that corresponding instance of the condition* $\{L\} \sim \{Q\} \to^{\star}_{R,B} tt$ *can never be satisfied. The only subterms having any* B*-matches at all for* $L(PQ)$ *are* $(yz)((yx)(yz))$ *and* $(yx)(yz)$*. Maude's* `xmatch` *command computes ten* B*-matches for* $(yz)((yx)(yz))$*. Restricting them to the relevant variables* L *and* Q*, and disregarding parentheses, these* B*-matches are:* $\{L \mapsto y, Q \mapsto yxyz\}$, $\{L \mapsto y, Q \mapsto xyz\}$, $\{L \mapsto yz, Q \mapsto xyz\}$, $\{L \mapsto y, Q \mapsto yz\}$, $\{L \mapsto yz, Q \mapsto yz\}$, $\{L \mapsto yzy, Q \mapsto yz\}$, $\{L \mapsto y, Q \mapsto z\}$, $\{L \mapsto yz, Q \mapsto z\}$, $\{L \mapsto yzy, Q \mapsto z\}$ *and* $\{L \mapsto yzyx, Q \mapsto z\}$*. The key observation again is that, except for the fifth* B*-match* $\{L \mapsto yz, Q \mapsto yz\}$*, the* sets *associated to the instances of* L *and* Q *in the remaining nine matches are* different*, so that the condition* $\{L\} \sim \{Q\} \to^{\star}_{R,B} tt$ *can never be satisfied. But the full description of the fifth* B*-match is* $\{L \mapsto yz, P \mapsto yx, Q \mapsto yz\}$*, so that the corresponding instance of the second condition* $\{LP\} \sim \{L\} \to^{\star}_{R,B} tt$ *cannot be satisfied, because the sets* $\{y\}, \{z\}, \{x\}$ *and* $\{y\}, \{z\}$ *are different. For the subterm* $(yx)(yz)$ *we get the following three* B*-matches (again, restricted to* L *and* Q*):* $\{L \mapsto y, Q \mapsto yz\}$, $\{L \mapsto y, Q \mapsto z\}$*, and* $\{L \mapsto yx, Q \mapsto z\}$*. But the sets associated to* L *and* Q *in all these* B*-matches are all different, so that, again, the corresponding instances of the condition* $\{L\} \sim \{Q\} \to^{\star}_{R,B} tt$ *can never be satisfied. Therefore,*

$(vx)((yz)((yx)(yz)))$ *is in* R, B*-normal form, as claimed.*

This leaves the following question unanswered: is the theory of idempotent semigroups decidable? This will be answered in the positive. I will explain: (i) in Example 8 how the above rewrite theory $\mathcal{R}_{Idemp.Sgr}$ *can be extended to a strictly* B*-coherent theory* $\overline{\mathcal{R}}_{Idemp.Sgr}$*, (ii) in Example 10 how* $\overline{\mathcal{R}}_{Idemp.Sgr}$ *is operationally terminating, and (iii) in Example 14 how it is Church-Rosser and provides a decision procedure for the equational theory of idempotent semigroups.*

3.3. Discussion

Note that the only assumption on the conditional order-sorted rewrite theory $\mathcal{R} = (\Sigma, B, R)$ is that the conditional rules in R have *oriented conditions* and are therefore of the form $l \to r$ *if* $\bigwedge_{i=1..n} u_i \to v_i$. When Σ is unsorted and $B = \emptyset$, such theories specialize to the notion of an *oriented 4-CTRS* [60]. A difficulty with such extremely general rewrite theories is their *infinitely-branching non-determinism*, since there can be an infinite number of possible (not B-equivalent) substitutions θ in a single application of the **Replacement** rule. However, as explained in [63], such infinitely-branching non-deterministic rewrite theories can be quite useful, since they can naturally model *open-concurrent systems*, and can become executable using symbolic methods. In Section 6.2, a still very general —yet easily executable— notion of rewrite theory, is discussed. It generalizes the notion of *deterministic 3-CTRS* in [60].

Although quite simple, the two inference systems above hide some subtleties worth explaining. They have to do with the claim made above that the naive generalization of the unconditional relations $\to_{R/B}$, $\to^*_{R/B}$, $\to_{R,B}$ and $\to^*_{R,B}$ would actually be wrong. The two key observations are the following:

1. In standard treatments of unconditional rewriting modulo B, e.g., [38, 3], the relations $\to^*_{R/B}$ and $\to^*_{R,B}$ are defined in the *standard sense*, that is, as the *reflexive-transitive* closures of the respective relations $\to_{R/B}$ and $\to_{R,B}$.

2. However, in the above two inference systems the relations $\to^\star_{R/B}$ and $\to^\star_{R,B}$ are *not* the reflexive-transitive closures of the respective relations $\to_{R/B}$ and $\to_{R,B}$. The perceptive reader may have noticed that I use Latex's `\star` symbol for $\to^\star$, as opposed to the `\ast` symbol typically used for the reflexive-transitive closure $\to^*$, to mark, unobtrusively, the subtle difference. What we actually have (see Fact (2) in Remark 2 below) is $\to^\star_{R/B} = (\to^*_{R/B}; =_B)$, and $\to^\star_{R,B} = (\to^*_{R,B}; =_B)$. That is, the B-equality relation $=_B$ is further composed in both cases because of the corresponding **Reflexivity** rule. And this is *crucial* for the **Replacement** rules to work correctly.

I call $\to^\star_{R/B}$ (resp. $\to^\star_{R,B}$) the R/B*-reachability relation* (resp. the the R, B*-reachability relation*), because that is exactly what these relations are when we view the terms $t \in \mathcal{T}_\Sigma(\mathcal{X})$ as descriptions of *states* of a system satisfying structural axioms B.

Following the standard approach to conditional rewriting (see, e.g., [60]), the *naive* generalization of rewriting modulo B to the conditional case would proceed as follows. One would define $\rightarrow_{R/B}$ as the union $\rightarrow_{R/B} = \bigcup_n \rightarrow_{R/B,n}$, where $\rightarrow_{R/B,0} = \emptyset$, and for each $n \in \mathbb{N}$, we have $\rightarrow_{R/B,n+1} = \rightarrow_{R/B,n} \cup \{(u, v) \mid u =_B l\sigma \ \rightarrow \ r\sigma =_B v \wedge l \rightarrow r \text{ if } \bigwedge_i u_i \rightarrow v_i \in R \wedge \forall i, u_i\sigma \rightarrow^*_{R/B,n} v_i\sigma\}$, where $\rightarrow^*_{R/B,n}$ of course denotes the reflexive-transitive closure. The naive definition of $\rightarrow_{R,B}$ would be entirely analogous. That these are the *wrong* definitions can be illustrated with a simple example.

Example 5. *Consider an unsorted signature with constants a, b, c, unary function symbol f and binary operator $\_\cdot\_$ and let B consist of the commutativity axiom $x \cdot y = y \cdot x$. Let R have just the single conditional rule $f(x \cdot f(y)) \rightarrow c$ if $x \cdot y \rightarrow z \cdot a$. Since f is different from $\_\cdot\_$, this rule is coherent and, indeed, as we shall see in Section 4, strictly coherent. If we were to adopt the above, naive definitions of $\rightarrow_{R/B}$ and $\rightarrow_{R,B}$, the term $f(a \cdot f(b))$ would actually be* irreducible *modulo B. This is because the term $a \cdot b$ that has to be tested in the condition is obviously irreducible modulo B. But for any irreducible term t, if $t \rightarrow^*_{R/B} t'$ and $\rightarrow^*_{R/B}$ is the reflexive-transitive closure of $\rightarrow_{R/B}$, then we must have $t = t'$. Therefore, from $a \cdot b$ we can* never *reach with the reflexive-transitive closure $\rightarrow^*_{R/B}$ any instance of the term $z \cdot a$, even if we make the "right" choice and instantiate z to b.*

This clearly shows that the above, naive definition is wrong. In fact, the term $f(a \cdot f(b))$ *is* reducible to c, since its condition is satisfied in 0 steps by $a \cdot b$ matching modulo commutativity the pattern $z \cdot a$ with matching substitution $z \mapsto b$. Here is, for example, a trace of the execution given by Maude, where `comm` declares the commutativity axiom.

```
mod APORIA is
  sort U .
  ops a b c : -> U .
  op f : U -> U .
  op _._ : U U -> U [comm] .
  vars x y z : U .
  crl f(x . f(y)) => c if x . y => z . a .
endm

Maude> set trace on .
Maude> rewrite f(a . f(b)) .
********** trial #1
crl f(x . f(y)) => c if x . y => a . z .
x --> a
y --> b
z --> (unbound)
********** solving condition fragment
x . y => a . z
********** success for condition fragment
x . y => a . z
```

```
x --> a
y --> b
z --> b
********** success #1
********** rule
crl f(x . f(y)) => c if x . y => a . z .
x --> a
y --> b
z --> b
f(a . f(b))
--->
c
rewrites: 1 in 0ms cpu (0ms real) (2506 rewrites/second)
result U: c
```

In summary, the key point is that the satisfaction of a rule's condition requires *search*, to see whether each $u_i\theta_{i-1}$ can *reach* an instance $v_i\theta_i$ of the pattern v_i. For this, the above **Reflexivity** rules are crucial in two related ways. First, to be able to reach terms in 0 steps, as the above example illustrates. Second, in the case of the $\rightarrow^{\star}_{R,B}$ relation, to reach the terms $v_i\theta_i$ from $u_i\theta_{i-1}$ in n steps, since for $n \geq 1$ the n-step relation $\rightarrow^{n}_{R,B}$ will typically rewrite $u_i\theta_{i-1}$ to a term w_i such that we only have $w_i =_B v_i\theta_i$, and we need a last use of **Reflexivity** (always required by the **Transitivity** rule) to close the gap $w_i =_B v_i\theta_i$.

Remark 2. *The easy proof of the following facts is left to the reader:*

1. $\rightarrow_{R/B} \subseteq \rightarrow^{\star}_{R/B}$, *and* $\rightarrow_{R,B} \subseteq \rightarrow^{\star}_{R,B}$.
2. $u \rightarrow^{\star}_{R/B} v$ *holds iff there is a chain of* $n \geq 0$ *rewrite steps followed by a* B*-equality step of the form* $u \rightarrow_{R/B} u_1 \rightarrow_{R/B} u_2 \ldots u_{n-1} \rightarrow_{R/B} u_n =_B v$. *And* $u \rightarrow^{\star}_{R,B} v$ *holds iff there is a chain of* $n \geq 0$ *rewrite steps followed by a* B*-equality step of the form* $u \rightarrow_{R,B} u'_1 \rightarrow_{R,B} u'_2 \ldots u'_{n-1} \rightarrow_{R,B} u'_n =_B v$.
3. $(\rightarrow^{\star}_{R/B}; \rightarrow^{\star}_{R/B}) \subseteq \rightarrow^{\star}_{R/B}$.

In general the reachability relation $\rightarrow^{\star}_{R,B}$ is *not* transitive; that is, we do not have $(\rightarrow^{\star}_{R,B}; \rightarrow^{\star}_{R,B}) \subseteq \rightarrow^{\star}_{R,B}$. For a simple counterexample consider an unsorted signature with constants a, b, c and a binary AC symbol $\_ \cdot \_$, and R with just one unconditional rule $a \cdot b \rightarrow c$. Then we have

$$(b \cdot a) \cdot ((c \cdot a) \cdot b) \rightarrow_{R,AC} c \cdot ((c \cdot a) \cdot b) =_{AC} (b \cdot a) \cdot (c \cdot c) \rightarrow_{R,AC} c \cdot (c \cdot c).$$

Therefore, $((b \cdot a) \cdot ((c \cdot a) \cdot b), c \cdot (c \cdot c)) \in (\rightarrow^{\star}_{R,AC}; \rightarrow^{\star}_{R,AC})$. But since $c \cdot ((c \cdot a) \cdot b)$ is irreducible by $\rightarrow_{R,AC}$, Fact (2) above ensures $((b \cdot a).((c \cdot a) \cdot b), c \cdot (c \cdot c)) \notin \rightarrow^{\star}_{R,AC}$.

4. Strict Coherence of Conditional Order-Sorted Rewrite Theories

Although the semantics of conditional ordered-sorted rewriting modulo B has been defined in Section 3.2 with no restrictions on B, as mentioned in

Section 2.1, non-linear and/or non-regular axioms B can cause serious anomalies. To avoid such anomalies, and to ensure a reasonable implementability of R, B-rewriting in the order-sorted case, throughout this section in which strict coherence is studied I will assume that the rewrite theory $\mathcal{R} = (\Sigma, B, R)$ has a set B of equational axioms satisfying the following *executability requirements*:

1. they are regular and linear;
2. they are *most general possible*, in the sense that for any $u = v \in B$, the sort of each $x \in \mathcal{V}ar(u = v)$ is the top sort of its connected component in the poset $(S, \leq)$;
3. they are sort-preserving; and
4. $=_B$ is decidable and has a finitary B-matching algorithm.

Let me comment briefly on requirements (1)–(4). A good thought experiment is to ask yourself what they mean in the *many-sorted* and *unsorted* subcases. The obvious answer is that in those subcases requirements (2) and (3) hold trivially, so we are just asking that: (1) the axioms B are regular and linear, which as shown in Section 2.1 is essential for them to be well-behaved and to achieve strict coherence; and (4) that $=_B$ is decidable, which is essential to implement the **Reflexivity** rule, and B has a finitary B-matching algorithm, which is again essential to implement the **Replacement** inference rule for the R, B-rewriting relation. So in these two subcases we are asking the *minimum necessary* for the entire business of strictly B-coherent R, B-rewriting to get off the ground.

What more are we requiring in the general, *order-sorted* case? Requirement (3) is the essential one: B being sort-preserving is extremely useful for correctly performing *order-sorted* rewriting modulo B: when B-matching a subterm t_p against a rule's lefthand side to obtain a matching substitution σ, we need to check that σ is well-sorted, that is, that if a variable x has sort s, then some element in the B-equivalence class $[x\sigma]_B$ has also sort s (or lower). But since B is sort-preserving, this is equivalent to checking $ls(x\sigma) \leq s$, which is a trivial check. Because of its great importance for implementing B-matching both correctly and efficiently, when a module is entered to Maude this property is automatically checked (together with the preregularity property that ensures the the least sort function $ls(t)$ is defined for any term t) by checking the so-called B-*preregularity* of the signature Σ for the axioms B [11].

What about requirement (2)? Under the very reasonable assumption that all subsort-polymorphic operators —for example, all subsort-polymorphic typings of a binary + operator— obey the *same* equational axioms in B, requirement (2) involves no real loss of generality in the context of (3). This is so because we can always *make it hold* by a simple extension of our rewrite theory $\mathcal{R} = (\Sigma, B, R)$ in which a fresh extra sort, called a "kind," is added at the top of each connected component and all operators are "lifted to kinds." In fact, as already pointed out in Example 3, Maude automatically kind-completes any user theory $\mathcal{R} = (\Sigma, B, R)$ in exactly this way. Note that, by B being sort-preserving, all

new terms having those new "kind" sorts as their least sort will *never* be equal modulo B to any terms having any of the original sorts. Also, no variables in the rules in R will ever be able to match such new terms.

Example 6. *(Natural Numbers with Addition). Consider the following order-sorted specification $(\Sigma, E \cup AC)$ defining natural numbers with addition. Σ has a sort Nat, a subsorts NzNat $<$ Nat of non-zero naturals, constants 0 of sort Nat and 1 of sort NzNat, and subsort-polymorphic addition operators $\_+\_ :$ Nat Nat $\to$ Nat and $\_+\_ :$ NzNat Nat $\to$ NzNat. AC are the associativity $(x+y)+z = x+(y+z)$ and commutativity $x+y = y+x$ axioms for $+$, where x, y, z are variables of sort Nat. E consists of the single equation $x+0 = x$.*

The initial algebra $\mathcal{T}_{\Sigma/AC\cup E}$ exactly defines the natural numbers with addition. This specification satisfies the above requirements (1), (2), and (4) (note that (2) holds without any need to "kind complete" the specification). However, this specification fails requirement (3) that the axioms AC should be sort-preserving. *This is because, if we declare a new variable x' of sort NzNat, then the term $x'+y$ has least sort NzNat, but the term $y+x'$ has least sort Nat. As mentioned above, failure of requirement (3) wreaks havoc for matching rules modulo B. Consider that, as done in Example 3, we were to add a non-zero predicate $nz :$ Nat $\to$ Nat defined by the equations: $nz(0) = 0$, and $nz(n') = 1$, where n' has sort NzNat. Let now (Σ, AC, R) denote the rewrite theory with rules $R = \{x+0 \to x, nz(0) \to 0, nz(n') \to 1\}$ and let us try to simplify the term $nz((0+1)+0)$ by R, AC-rewriting with rule $nz(n') \to 1$. Since the least sort of $0+(0+1)$ is Nat, we* cannot *do so* unless *we embark in a costly AC-proof search to see if $0+(0+1)$ could be proved AC-equal to a term of sort NzNat. Indeed, the AC-proof $0+(0+1) \leftrightarrow_{AC} 0+(1+0) \leftrightarrow_{AC} (1+0)+0$ reaches the term $(1+0)+0$, which* does *have sort NzNat. But such a proof search is now a* prerequisite *for testing the sort of each variable before taking any R, AC-rewriting step, which can be prohibitively expensive!*

All these problems can be solved by just adding to Σ the operator declaration $\_+\_ :$ Nat NzNat $\to$ NzNat, which makes the AC axioms sort-preserving.[10] *The effect when evaluating the term $nz((0+1)+0)$ is remarkable: now the attempt to apply the rule $nz(n') \to 1$* automatically succeeds*, because of the* cheap *syntactic check $ls((0+1)+0) =$ NzNat.*

The main goal of this section is to show that strict coherence of the rules R in a conditional order-sorted rewrite theory $\mathcal{R} = (\Sigma, B, R)$ is the key notion to bisimulate the relation $\to_{R/B}$ by means of the much simpler and much more efficient relation $\to_{R,B}$, and to develop methods of *strict coherence completion* that can try to transform $\mathcal{R}$ into a semantically equivalent theory $\overline{\mathcal{R}}$ that is strictly coherent. For this, the notion of rule B-extension plays a key role.

[10] B being sort-preserving can be easily checked by checking $ls(u\rho) = ls(v\rho)$ for each axiom $u = v \in B$ and each ρ ranging over all "sort specialization" substitutions, where some sorts in the variables of $u = v$ may be lowered ("specialized") to smaller sorts. Maude automatically performs this check for any user-specified theory as part of its B-preregularity check [11].

4.1. B-Extensions of Conditional Rules

In the unsorted and unconditional case, a useful technique to achieve coherence modulo B for a set of rewrite rules R is to *extend* the rules R by suitable contexts obtained from the terms in the axioms B. This technique goes back to Peterson and Stickel [61], who introduced the notion of extension, considered the case where B is regular and linear (their Theorem 9.5), and proved that a finite set of extensions suffice when B is any combination of associativity (A), commutativity (C), and associativity-commutativity (AC) axioms (their Theorem 10.5). The technique has been generalized to arbitrary sets of axioms B, e.g., [38, 3]. One attractive feature is its simplicity. An alternative, more complex approach is to attempt some kind of completion based on suitable "critical pairs" between axioms and rules in the style of, e.g., [38, 26]; but the number of B-unifiers between two terms may be infinite for some B. Instead, the technique of extensions only relies on a B-*unifiability* algorithm.

Let me begin by recalling the very simple notion of B-extension in the unconditional case. Given an unconditional rule $l \to r$ and axioms B, its B-*extensions* are all rules of the form $u[l]_p \to u[r]_p$ such that either $u = v$ or $v = u$ is in B, and where p is a non-variable and non-top (i.e., $p \neq \epsilon$) position of u such that $u|_p$ and l are B-*unifiable* (note, however, that no B-unifier is computed to build the extension). The variables of u and l are always renamed if needed to ensure that u and l have no variables in common. The extended rules can themselves be extended; however, for certain axioms B, such as any combination of A, C, and identity (U) axioms, it is easy to show that the extension process reaches a fixpoint in at most two steps, in the sense that the newly generated extensions are all *instances* modulo B of previously generated rules (for combinations of A and C this is the already-mentioned Theorem 9.5 in [61]).

As in the unconditional and unsorted case, the key notions in their conditional order-sorted generalization are those of B-extension and B-subsumption.

Definition 1. *Let $\mathcal{R} = (\Sigma, B, R)$ be a conditional order-sorted rewrite theory, and let $l \to r$ if C be a rule in R, where C abbreviates the rule's condition. Without loss of generality we assume that $\mathcal{V}ar(B) \cap \mathcal{V}ar(l \to r \text{ if } C) = \emptyset$. If this is not the case,* only the variables of B *will be renamed; the variables of $l \to r$ if C will* never *be renamed. We then define the set of B-*extensions[11] *of $l \to r$ if C as the set*[12]*:*

$$Ext_B(l \to r \text{ if } C) = \{u[l]_p \to u[r]_p \text{ if } C \mid u = v \in B \cup B^{-1} \wedge p \in \mathcal{P}_\Sigma(u) - \{\epsilon\} \wedge MGU_B(l, u|_p) \neq \emptyset\}$$

where, by definition, $B^{-1} = \{v = u \mid u = v \in B\}$.

[11] One could consider the rule $l \to r$ *if* C as a trivial extension of itself. The set $Ext_B(l \to r \text{ if } C)$ excludes such a trivial extension and contains only the rule's *proper* extensions. This is useful in practice to avoid looping in the extension closure algorithm described later.

[12] Note that, because of the assumptions that Σ is kind-complete and that all $u = v \in B$ are most general possible and have variables whose sorts are tops of connected components in the sort poset $(S, \leq)$, the terms $u[l]_p$ and $u[r]_p$ are always well-formed Σ-terms.

Given two rules $l \to r$ if C and $l' \to r'$ if C with the same *condition C we say that $l \to r$ if C B-*subsumes[13] *$l' \to r'$ if C iff there is a substitution σ such that: (i) $dom(\sigma) \cap \mathcal{V}ar(C) = \emptyset$, (ii) $l' =_B l\sigma$, and (iii) $r' =_B r\sigma$.*

I now describe in detail an algorithm to compute the *B-extension closure* $\overline{Ext}_B(l \to r \text{ if } C)$ of a conditional rule $l \to r$ *if* C. To avoid non-determinism I assume that the set U of all terms u such that $u = v \in B \cup B^{-1}$ has been linearly ordered. I also assume that for each $u \in U$ the non-variable and non-top positions of u have been linearly ordered.[14] Call such positions *usable* positions. The algorithm maintains two queues, one of *extended* rules (i.e., rules whose extensions have already been computed), and another of *generated* rules (i.e., rules generated by the extension process). Note that, by construction, all rules in the queues will always have the exact same condition C.

The algorithm to compute the set $\overline{Ext}_B(l \to r \text{ if } C)$ begins in an initial state where the queue of extended rules is empty, and the queue of generated rules contains only the rule $l \to r$ *if* C. It then repeatedly performs the following sequence of steps until the queue of generated rules is empty:

1. If the queue of generated rules is empty **stop**; otherwise, let $l' \to r'$ *if* C be the first rule in it.

2. Sequentially (according to the linear order of U and of u's positions), for each $u \in U$ and usable position p, if l' and $u|_p$ are B-unifiable, do the following:

   - Generate the extension $u[l']_p \to u[r']_p$ *if* C.
   - If $u[l']_p \to u[r']_p$ *if* C is B-subsumed by any rule in either of the queues, discard it; otherwise, append it at the end of the queue of generated rules.

3. Once all the terms in U and all usable positions in each $u \in U$ have been tried, dequeue the rule $l' \to r'$ *if* C and append it at the end of the queue of extended rules.

If after repeating the above loop n times the queue of generated rules becomes empty, the set $\overline{Ext}_B(l \to r \text{ if } C)$ is the set of rules in the queue of extended rules of the final state. If the queue of generated rules never becomes empty, we can still think of $\overline{Ext}_B(l \to r \text{ if } C)$ as the infinite set of all rules ever added to the queue of extended rules.

[13] Note that for unconditional rules, since C is empty, we have $\mathcal{V}ar(C) = \emptyset$, so that requirement (i) trivially holds for σ. Therefore, the conditional notion of subsumption yields the usual unconditional notion as a special case.

[14] These two assumption can easily be dropped; they are only useful to obtain a *deterministic algorithm*, so they would typically be made anyway in a sequential implementation.

Given a conditional order-sorted rewrite theory $\mathcal{R} = (\Sigma, B, R)$, let its *B-extension closure* be the theory $\overline{\mathcal{R}} = (\Sigma, B, \overline{R})$, where

$$\overline{R} = \bigcup_{(l \to r \text{ if } C) \in R} \overline{Ext}_B(l \to r \text{ if } C).$$

This theory always exists, but can be infinite if the B-extension closure of some rule in R is infinite.

Is there an algorithm to check whether a conditional theory is already closed under B-extensions? Yes, this is just a B-subsumption check. That is, we *define* $\mathcal{R} = (\Sigma, B, R)$ to be *closed under B-extensions* iff any B-extension of any rule in R is B-subsumed by some rule in R.[15]

Remark 3. *All notions of B-extension I am aware of deal only with* unconditional *rules in an unsorted setting. Definition 1 generalizes those previous notions to the broader class of order-sorted* conditional *rewrite theories. But Definition 1 is not the only possibility: the notion of "closed under B-extensions" could be broadened by adopting a more relaxed notion of B-subsumption that would, for example, allow a rule like* $(a \cdot b) \cdot x \to c \cdot x$ *if* $x > 0 \to$ *true to B-subsume the rule* $a \cdot (b \cdot y) \to c \cdot y$ *if* $y > 0 \to$ *true, where* $a, b, 0$ *and true are constants, and* $\_ \cdot \_$ *is A or AC. A further broadening would allow not just variable renamings in conditions, but general condition instantiations. For example, again with* $\_ \cdot \_$ *A or AC, the rule* $\alpha : (x \cdot a) \cdot (b \cdot y) \to c$ *if* $x \to y$ *would then also subsume the rule* $z \cdot (a \cdot (b \cdot a)) \to c$ *if* $z \to a$*. An even broader notion of subsumption would allow the above rule* α *to also subsume the unconditional rule* $z \cdot (a \cdot (b \cdot z)) \to c$*. Using these broader senses of subsumption, B-extension closures could be computed not just a rule at a time using the sets* $\overline{Ext}_B(l \to r \text{ if } C)$*, but* jointly*, for all the rules R together, as* $\overline{Ext}_B(R)$*.*

To keep technicalities to a minimum, I will not pursue here the details of such possible broadenings. Some generality would indeed be gained; but, as it will become obvious in Section 4.2, much control could be lost for coherence purposes: crucially, the fact that in the **Strict Coherence** *property of Section 2 and its equivalents, the notion of B-extension closure I propose can guarantee that the* same condition instantiation *can be used in two strictly coherent conditional rewrites, where the rules used in those two rewrites belong to the* same

[15] Since, by construction, any B-extension of a rule in $\overline{\mathcal{R}}$ is always B-subsumed by some rule already in $\overline{\mathcal{R}}$, the theory $\overline{\mathcal{R}}$ is of course closed under B-extensions. One would like to say that $\mathcal{R} = (\Sigma, B, R)$ is closed under B-extensions iff $R = \overline{R}$. But, this is not quite right: for efficiency reasons, the algorithm given above to compute $\overline{R}$ is too weak to ensure that $\overline{\overline{R}} = \overline{R}$, so that we only have the containment $\overline{R} \subseteq \overline{\overline{R}}$. For a simple example consider $R = \{a + b \to c, a + b + d \to c + d, a + b + x \to c + x\}$ unsorted and with $+$ AC. Since $a + b + x \to c + x$ extends $a + b \to c$ and subsumes both any further extensions of $a + b \to c$ and $a + b + d \to c + d$, R is closed under B-extensions, but $R \subset \overline{R}$. This happens because the algorithm computes B-extensions *one rule at a time*, so that it does not check if a generated rule is already B-subsumed by some other rule in R, or in the already computed sets $\overline{Ext}_B(l \to r \text{ if } C)$ for some other rules $l \to r$ *if* C in R. By checking B-subsumption in *this* more general sense, the above algorithm could be easily modified to ensure that we always have the fixpoint $\overline{\overline{R}} = \overline{R}$.

set $\overline{Ext}_B(l \to r \text{ if } C)$. *In particular, this means that such rule applications —and, more generally, rule application* attempts*— will have the* same operational termination behavior. *The last example above, where the unconditional rule* $z \cdot (a \cdot (b \cdot z)) \to c$ *was subsumed by the rule* α*, illustrates the total loss of control over operational termination behavior: since the rule is unconditional, it* cannot *loop when evaluating its empty condition. But* α can *loop on some rule instances to which the unconditional rule can also be applied: for example, when applied to the term* $(a \cdot a) \cdot (b \cdot a)$ *in the presence of the additional rules* $a \to b$ *and* $b \to a$.

Example 7. *(B-Extension Closure for Exclusive Or). Recall the exclusive or rewrite theory* $\mathcal{R}_\oplus = (\Sigma_\oplus, B, R)$ *from Example 2, where* B *are the associativity and commutativity (AC) axioms for* $\oplus$*, and* R *are the rewrite rules* $R = \{x \oplus 0 \to x, x \oplus x \to 0\}$. *Note that in the commutativity axiom* $x \oplus y = y \oplus x$ *there are no non-top and non-variable positions. This means that no extensions can be associated to this axiom. The terms in the associativity axiom* $x \oplus (y \oplus z) = (x \oplus y) \oplus z$ *can be ordered from left to right as* $u = x \oplus (y \oplus z)$*, and* $v = (x \oplus y) \oplus z$. *The only non-top and non-variable position* p *of* u *is* $p = 2$. *Likewise, the only such position of* v *is* $p = 1$. *Note, furthermore, that a* $\Sigma_\oplus$*-term is* B*-unifiable with* u_2 *(resp.* v_1*) iff it is of the form* $t \oplus t'$. *Let us now compute the* B*-extensions of* R. *Let us represent the pair of queues of extended and generated rules as* $\alpha_1 : \ldots : \alpha_n \,[\!]\, \beta_1 : \ldots : \beta_m$*, with* $\alpha_1 : \ldots : \alpha_n$ *the queue of extended rules and* $\beta_1 : \ldots : \beta_m$ *that of generated rules, and with* $\emptyset$ *denoting an empty queue. The initial pair of queues for computing the* B*-extensions of the rule* $x \oplus 0 \to x$ *is* $\emptyset \,[\!]\, x \oplus 0 \to x$. *The extension of* $x \oplus 0 \to x$ *using* u *is* $y \oplus (x \oplus 0) \to y \oplus x$*, which is* B*-subsumed by the rule* $x \oplus 0 \to x$ *itself with substitution* $\{x \mapsto y \oplus x\}$. *Likewise, the extension using* v *is* $(x \oplus 0) \oplus z \to x \oplus z$*, again* B*-subsumed by* $x \oplus 0 \to x$ *with substitution* $\{x \mapsto x \oplus z\}$. *Therefore, we obtain the final pair of queues* $x \oplus 0 \to x \,[\!]\, \emptyset$*, so that* $x \oplus 0 \to x$ *has no extensions besides itself.*

Let us now compute the B*-extensions of the rule* $x \oplus x \to 0$. *The initial pair of queues is* $\emptyset \,[\!]\, x \oplus x \to 0$. *The extension of* $x \oplus x \to 0$ *using* u *is* $y \oplus (x \oplus x) \to y \oplus 0$. *Its extension using* v *is* $(x \oplus x) \oplus z \to 0 \oplus z$*, which is* B*-subsumed by the extension using* u. *So in a first iteration we get the pair of queues* $x \oplus x \to 0 \,[\!]\, y \oplus (x \oplus x) \to y \oplus 0$. *But the two extensions of* $y \oplus (x \oplus x) \to y \oplus 0$ *using* u *and* v *are* B*-subsumed by itself, so that we get the final pair of queues* $x \oplus x \to 0 : y \oplus (x \oplus x) \to y \oplus 0 \,[\!]\, \emptyset$. *Therefore, the only new rule added to* $\mathcal{R}_\oplus$ *in its* B*-extension closure* $\overline{\mathcal{R}}_\oplus$ *is the rule* $y \oplus (x \oplus x) \to y \oplus 0$.

Recall that in Example 2 the term $x \oplus (y \oplus x)$ *was* R, B*-irreducible in the theory* $\mathcal{R}_\oplus$. *Now, however, since* $x \oplus (y \oplus x) =_B y \oplus (x \oplus x)$*, it can be* $\overline{R}, B$*-reduced with the extended rule* $y \oplus (x \oplus x) \to y \oplus 0$ *in the extended set of rules* $\overline{R}$ *of* $\overline{\mathcal{R}}_\oplus$ *to* $y \oplus 0$ *with identity* B*-matching substitution, and* $y \oplus 0$ *can then be rewritten to the canonical form* y *with rule* $x \oplus 0 \to x$ *and* B*-substitution* $\{x \mapsto y\}$. *In fact, Corollary 3 below implies that* $\overline{\mathcal{R}}_\oplus$ *is strictly* B*-coherent.*

Note that the argument given above, showing that the rule $x \oplus 0 \to x$ has no AC-extensions besides itself applies *mutatis mutandis* to the theory (Σ, AC, R)

defined in Example 6, where to make the AC axioms sort-preserving Σ contains the additional declaration $\_ + \_ : \mathit{Nat}\ \mathit{NzNat} \to \mathit{NzNat}$, and the rules R include the identity rule for natural number addition with 0, and the two rules defining the non-zero predicate. Since the rules for the non-zero predicate have no AC-extensions at all, this shows that the just-described natural number theory (Σ, AC, R) from Example 6 is *already closed* under AC-extensions.

Example 8. *(B-Extension Closure for Idempotent Semigroups). In Example 4 the rewrite theory $\mathcal{R}_{Idemp.Sgr} = (\Sigma, B, R)$ orienting the equations of the order-sorted theory of idempotent semigroups was shown not to be strictly B-coherent. But we can easily compute its B-completion by extensions $\overline{\mathcal{R}}_{Idemp.Sgr}$ as follows. Note that the two rules obtained by orienting the last two equations in the theory of Example 4 have no B-extensions whatever. As already pointed out for the case of commutative semigroups in Example 4, the rule $S, S \to S$ has the rule $S', (S, S) \to S', S$ as its only B-extension. I leave for the reader to check that the rule $LL \to L$ has three B-extensions, namely, $P(LL) \to PL$, $(LL)Q \to LQ$ and $(P(LL))R \to (PL)R$. Let me explain in detail how the above conditional rule is B-extended. The initial pair of queues is*

$$\emptyset \,[]\, L(PQ) \to LQ \ \ \mathit{if}\ \ \{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$$

The A axiom gives us again terms $u = X(YZ)$ and $v = (XY)Z$. Extending the above rule with u gives us the rule $X(L(PQ)) \to X(LQ)$ if $\{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$, and with v we get the rule $(L(PQ))Z \to (LQ)Z$ if $\{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$. After this first iteration we get a configuration of queues with the original rule in the queue of extended rules, and the two extended rules in the following queue of generated rules:

$$X(L(PQ)) \to X(LQ) \ \ \mathit{if}\ \ \{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt :$$

$$(L(PQ))Z \to (LQ)Z \ \ \mathit{if}\ \ \{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$$

The extension of the first rule with u gives us the rule $X'(X(L(PQ))) \to X'(X(LQ))$ if $\{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$, which is B-subsumed by that first rule with substitution $\{X \mapsto X'X\}$. The extension with v gives us the new rule $(X(L(PQ)))Y \to (X(LQ))Y$ if $\{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$. Therefore, after a second iteration the queue of extended rules has the original rule and the first rule above, and the queue of generated rules has the rules:

$$(L(PQ))Z \to (LQ)Z \ \ \mathit{if}\ \ \{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt :$$

$$(X(L(PQ)))Y \to (X(LQ))Y \ \ \mathit{if}\ \ \{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$$

The extensions with u of the first rule is B-subsumed by the second rule in the queue, and its extension with v is B-subsumed by the first rule itself. Therefore, in the third iteration the queue of generated rules has the original rule and its first two extensions, and the queue of generated rules only has the rule

$$(X(L(PQ)))Y \to (X(LQ))Y \ \ \mathit{if}\ \ \{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$$

But the extensions with u and v of this rule are both B-subsumed by that rule itself. Therefore, the above rule is moved to the queue of extended rules and the queue of generated rules becomes empty, so that the set of extended conditional rules consists of the following four rules:

- $L(PQ) \to LQ$ *if* $\{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$
- $X(L(PQ)) \to X(LQ)$ *if* $\{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$
- $(L(PQ))Z \to (LQ)Z$ *if* $\{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$
- $(X(L(PQ)))Y \to (X(LQ))Y$ *if* $\{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$.

Note that all rules above share the same *condition. This useful property follows from the definitions of B-extension and B-subsumption in Definition 1.*

Recall from Example 4 that the term $(vx)((yz)((yx)(yz)))$ is in R, B-normal form in the rewrite theory $\mathcal{R}_{Idemp.Sgr} = (\Sigma, B, R)$. Now, however, it can be $\overline{R}, B$-reduced to $v((x(yz))(x(yz)))$ in the B-extended theory $\overline{\mathcal{R}}_{Idemp.Sgr} = (\Sigma, B, \overline{R})$ with the above extended rule rule $X(L(PQ)) \to X(LQ)$ if $\{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$ and B-matching substitution $\{X \mapsto v, L \mapsto x(yz), P \mapsto y, Q \mapsto x(yz)\}$. And then $v((x(yz))(x(yz)))$ can be $\overline{R}, B$-reduced to the canonical form $v(x(yz))$, either at the top with the B-extended rule $P(LL) \to PL$, or at position $p = 2$ with the original idempotency rule $LL \to L$. In fact, Corollary 5 below implies that $\overline{\mathcal{R}}_{Idemp.Sgr}$ is strictly B-coherent.

4.2. Strictly Coherent Conditional Theories

As one would expect, the point of computing the B-extension closure $\overline{\mathcal{R}}$ is to obtain a strictly coherent theory. Of course, to really talk about strict coherence one must explain in which sense the abstract relations from Section 2 are *interpreted*. I will show below that for unconditional order-sorted rewrite theories they are interpreted exactly as in the special case of unsorted unconditional theories fully worked out in Example 1.

The case of an order-sorted conditional theory $\mathcal{R} = (\Sigma, B, R)$ is discussed in Section 4.3 below and is more subtle, because the abstract relation $\to_{\{R\}}$ is *not* interpreted as the standard conditional rewrite relation $\to_R$ associated to the rewrite theory $\mathcal{R} = (\Sigma, \emptyset, R)$, i.e., as the special case $B = \emptyset$, where the two relations $\to_{R/\emptyset}$ and $\to_{R,\emptyset}$ coincide and are denoted $\to_R$. However, *in both the unconditional and the conditional case* I will show below that: (i) the abstract relation $\leftrightarrow_B$ is interpreted as the one-step B-equality relation $\leftrightarrow_B$, (ii) the abstract relation $=_B$ is interpreted as the B-equality relation $=_B$, (iii) the abstract relation $\to_{R/\!\!/B}$ is interpreted as the R/B-rewrite relation $\to_{R/B}$, and (iv) the abstract relation $\to_{R:B}$ is interpreted as the R, B-rewrite relation $\to_{R,B}$. In Theorem 2 below and following statements, "strict local coherence" or "strict coherence" should be understood according to the interpretation (i)–(iv) above.

Rather then proving strict coherence in the interpretation (i)–(iv), I prove a stronger, and in fact crucial, result.

Theorem 2. *(Strict Local Coherence of* **Replacement** *Inferences). Let* $\mathcal{R} = (\Sigma, B, R)$ *be a conditional order-sorted rewrite theory where* B *satisfies requirements (1)–(4) at the beginning of Section 4 and is closed under* B*-extensions. Then, for each instance of* **Replacement** *of the form:*

$$\frac{u_1\theta \rightarrow^{\star}_{R,B} v_1\theta \quad \ldots \quad u_n\theta \rightarrow^{\star}_{R,B} v_n\theta}{u \rightarrow_{R,B} v}$$

and each one-step B*-equality proof* $u \leftrightarrow_B u'$ *there is another instance of* **Replacement** *of the form*

$$(\dagger) \quad \frac{u_1\theta' \rightarrow^{\star}_{R,B} v_1\theta' \quad \ldots \quad u_n\theta' \rightarrow^{\star}_{R,B} v_n\theta'}{u' \rightarrow_{R,B} v'}$$

with $v =_B v'$ *and* $\theta(x) = \theta'(x)$ *for each* $x \in \mathcal{V}ar(u_1 \rightarrow v_1 \ \wedge \ldots \wedge \ u_n \rightarrow v_n)$.

Proof. Let the above instance of **Replacement** be obtained by trying to apply[16] to u at position p with substitution θ rule $l \rightarrow r \ \ \mathit{if}\, C \in R$ with $C = u_1 \rightarrow v_1 \wedge \ldots \wedge u_n \rightarrow v_n$. Therefore, $u|_p =_B l\theta$ and $v = u[r\theta]_p$. Similarly, let $u \leftrightarrow_B u'$ be obtained by applying equation $w = w' \in B \cup B^{-1}$ at position q with substitution σ, so that $u|_q = w\sigma$, and $u' = u[w'\sigma]_q$.

The proof is by case analysis on the positions p and q using the *prefix* order $p \leq q$ between positions as strings. That is, $p \leq q$ iff there exists a (possibly empty) string q' such that $p.q' = q$. (1) If neither $p \leq q$ nor $q < p$ the result is trivial, since $u'|_p = u|_p$ and $v|_q = u|_q$, so that we have an application of **Replacement** of the form (†) at position p with $\theta = \theta'$, $v' = u'[r\theta]_p$, and a one-step B-equality proof $v \leftrightarrow_B v'$ at position q with $w = w'$ instantiated with σ. (2) If $p \leq q$ we have $u'|_p =_B u|_p =_B l\theta$ and therefore an application of **Replacement** of the form (†) with $\theta = \theta'$ and $v' = v$. For case (3) ($q < p$) two subcases can be distinguished. Let $x_1, \ldots, x_k$ be the variables in $w = w'$, and $q_1, \ldots, q_k$ their respective positions in w. Since $w = w'$ is linear and regular, the variables $x_1, \ldots, x_k$ occur in respective positions $q'_1, \ldots, q'_k$ in w'. Then either: (i) $q.q_i \leq p$ for some $1 \leq i \leq k$, so that $p = q.q_i.s$, or (ii) $p = q.q'$ for q' a non-variable and non-top position in w. In case (i), $u|_p = u'|_{q.q'_i.s}$ and therefore we have an application of **Replacement** of the form (†) with $\theta = \theta'$ and $v' = (u[w'\sigma]_q)[r\theta]_{q.q'_i.s}$. But this means that we have a one-step B-equality proof $v \leftrightarrow_B v'$ with $w = w'$ and substitution σ' identical to σ except for its value for x_i, which is $\sigma'(x_i) = u|_{q.q_i}[r\theta]_s$. This proves case 3.(i). In case 3.(ii), we have the identity $w|_{q'}\sigma = u|_p$. Since $u|_p =_B l\theta$, $w|_{q'}$ and l are B-unifiable, so that (assuming as always disjoint variables between B and R) the rule $l \rightarrow r \ \mathit{if}\, C$ has a B-extension $w[l]_{q'} \rightarrow w[r]_{q'} \ \mathit{if}\, C$. Since $\mathcal{R}$ is closed under B-extensions, either this extended rule belongs to R or is

[16] Since this is just an inference step, in general it is *not necessarily a rule application* (which would require the condition to be provable). To make this clear, let us call applications of **Replacement** "*rule application attempts*."

subsumed by a rule in R. I prove first the easier case where the extended rule belongs to R. Let θ' extend θ over $\mathcal{V}ar(w[l]_{q'} \to w[r]_{q'} \text{ if } C)$ by defining for each $x \in \mathcal{V}ar(w) - \mathcal{V}ar(w|_{q'})$ $\theta'(x) = \sigma(x)$. This gives us the equalities $u'|_q =_B u|_q =_B w[l]_{q'}\theta'$ and therefore, since θ' extends θ and instantiates the variables of C in the exact same manner, we have an application of **Replacement** of the form (†) at position q, with $v' = u'[w[r]_{q'}\theta']_q$. But, by the definition of θ' and the fact that $u' = u[u'|_q]_q$, it is easy to check that we have the actual term identities $u'[w[r]_{q'}\theta']_q = u[w[r]_{q'}\theta']_q = u[r\theta] = v$, thus proving the requirements for the inference step (†). This leaves us with the remaining case when there is a rule $l' \to r'$ *if* C in R subsuming $w[l]_{q'} \to w[r]_{q'}$ *if* C. That is, there is a substitution τ with $dom(\tau) \cap \mathcal{V}ar(C) = \emptyset$ such that $l'\tau =_B w[l]_{q'}$ and $r'\tau =_B w[r]_{q'}$. But then we have $u'|_q =_B u|_q =_B w[l]_{q'}\theta' =_B l'\tau\theta'$, and since $dom(\tau) \cap \mathcal{V}ar(C) = \emptyset$ and θ' extends θ, $\tau\theta'$ instantiates the variables of C in the exact same manner as θ. Therefore, we have an application of **Replacement** of the form (†) at position q with $v' = u'[r'\tau\theta']_q$, and the chain of equalities $u'[r'\tau\theta']_q =_B u'[w[r]_{q'}\theta']_q = v$, proving again the requirements for the inference step (†), as desired. □

Since a rewrite step $u \to_{R,B} v$ is just an application of **Replacement** for which the condition can be proved, and when $u \leftrightarrow_B u'$ the similar application of **Replacement** ensured by (†) in Theorem 2 has the *same condition*, we obtain also a rewrite $u' \to_{R,B} v'$ with $v =_B v'$. That is, we obtain the **Strict Local Coherence** property of the relation $\to_{R,B}$.

An easy induction on the number of steps in an equality proof $u =_B u'$ then gives us:

Corollary 2. *(Strict Coherence of* **Replacement** *Inferences). Let $\mathcal{R} = (\Sigma, B, R)$ be a conditional order-sorted rewrite theory where B satisfies requirements (1)–(4) at the beginning of Section 4 and is closed under B-extensions. Then, for each instance of* **Replacement** *of the form:*

$$\frac{u_1\theta \to^\star_{R,B} v_1\theta \quad \ldots \quad u_n\theta \to^\star_{R,B} v_n\theta}{u \to_{R,B} v}$$

and each B-equality proof $u =_B u'$ there is another instance of **Replacement** *of the form*

$$(\ddagger) \qquad \frac{u_1\theta' \to^\star_{R,B} v_1\theta' \quad \ldots \quad u_n\theta' \to^\star_{R,B} v_n\theta'}{u' \to_{R,B} v'}$$

with $v =_B v'$ and $\theta(x) = \theta'(x)$ for each $x \in \mathcal{V}ar(u_1 \to v_1 \ \wedge \ldots \wedge \ u_n \to v_n)$. □

Since a rewrite step $u \to_{R,B} v$ is just an application of **Replacement** for which the condition can be proved, and when $u =_B u'$ the similar application of **Replacement** ensured by (‡) in Corollary 2 has the *same condition*, we obtain also a rewrite $u' \to_{R,B} v'$ with $v =_B v'$. That is, we obtain the **Strict Coherence** property of the relation $\to_{R,B}$.

Given an order-sorted signature Σ and a set R of conditional Σ rules, we can *define* the relation $\to_R$ —in the *standard* CTRS sense (see [60]), but generalized to the order-sorted case— as the relation $\to_R = \to_{R/\emptyset} = \to_{R,\emptyset}$ for the

order-sorted conditional rewrite theory $(\Sigma, \emptyset, R)$. For $\mathcal{R} = (\Sigma, B, R)$ we then have obvious inclusions $\rightarrow_R \subseteq \rightarrow_{R,B} \subseteq \rightarrow_{R/B}$. Furthemore, if $\mathcal{R} = (\Sigma, B, R)$ is an *unconditional* order-sorted rewrite theory (i.e., all rules in R have an empty condition), routine inspection of the corresponding inference systems shows that we have $\rightarrow_{R/B} = (=_B; \rightarrow_R; =_B)$. Therefore, if $\mathcal{R}$ is an order-sorted unconditional rewrite theory closed under B-extensions, we can interpret the abstract relation $\rightarrow_{\{R\}}$ as the rewrite relation $\rightarrow_R$. This, together with the above interpretations (i)–(iv) for the other four abstract relations, means that all abstract relations from Section 2 are interpreted exactly as done for the special case of unsorted unconditional rewrite theory in Example 1. Therefore, using Theorem 1, the fact that $\rightarrow_{R,B}$ satisfies the **Strict Coherence** property immediately gives us:

Corollary 3. *Let $\mathcal{R} = (\Sigma, B, R)$ be an unconditional order-sorted rewrite theory where B satisfies requirements (1)–(4) at the beginning of Section 4 and is closed under B-extensions. Then $\mathcal{R}$ satisfies the* **Completeness**, **Bisimulation**, **Strict Coherence** *and* **Strict Local Coherence** *properties.*

4.3. Interpreting the Abstract Relations for Conditional Theories

The case when $\mathcal{R} = (\Sigma, B, R)$ is closed under B-extensions but is actually *conditional* is more subtle. One might easily think that, interpreting the abstract relations from Section 2 exactly as done in Example 1 for the special case of unsorted and unconditional rewrite theories, all the equivalences in Theorem 1 immediately apply to $\mathcal{R}$, but, remarkably, this is actually *false*.

We have indeed been using the notations $\rightarrow_{R/B}$ and $\rightarrow_{R,B}$, and we just defined above the notation $\rightarrow_R$ with its standard CTRS meaning. But in the conditional case we *cannot* in general interpret the abstract relations in Section 2 as we did in Example 1, for the simple reason that those abstract relations *must* satisfy certain conditions (e.g., $\rightarrow_{R/\!\!/B} = (=_B; \rightarrow_{\{R\}}; =_B)$), which are in general *violated* under that interpretation for conditional rewrite theories.

We indeed have inclusions $\rightarrow_R \subseteq \rightarrow_{R,B} \subseteq \rightarrow_{R/B}$. But what in general we do *not* have is the identity $\rightarrow_{R/B} = (=_B; \rightarrow_R; =_B)$. In fact, when $\mathcal{R}$ is not closed under B-extensions we do not even have the identity $\rightarrow_{R/B} = (=_B; \rightarrow_{R,B}; =_B)$. Here is a simple counterexample. Consider the unsorted rewrite theory $\mathcal{R}$ whose signature Σ has constants a, b, c, d, e and a binary AC symbol $\_ \cdot \_$ and whose rules R are the two rules: $c \cdot d \rightarrow e$, and $a \cdot b \rightarrow c$ *if* $(c \cdot a) \cdot d \rightarrow a \cdot e$. The rules R are simple enough to allow a succinct set-theoretic characterization of the relations $\rightarrow_R$, $\rightarrow_{R,AC}$, and $\rightarrow_{R/AC}$, namely:

- $\rightarrow_R = \{(u, u[e]_p) \in \mathcal{T}_\Sigma(\mathcal{X})^2 \mid p \in \mathcal{P}(u) \wedge u|_p = c \cdot d\}$,
- $\rightarrow_{R,AC} = \{(u, u[e]_p) \in \mathcal{T}_\Sigma(\mathcal{X})^2 \mid p \in \mathcal{P}(u) \wedge u|_p =_{AC} c \cdot d\}$, and
- $\rightarrow_{R/AC} = (=_{AC}; \rightarrow_R; =_{AC}) \cup \{(u', v') \in \mathcal{T}_\Sigma(\mathcal{X})^2 \mid (\exists u \in \mathcal{T}_\Sigma(\mathcal{X}))(\exists p \in \mathcal{P}(u))\ u =_{AC} u' \wedge u|_p = a \cdot b \wedge v' =_{AC} u[c]_p\}$.

Note, in particular, that $a \cdot b \to_{R/AC} c$, but $a \cdot b \not\to_{R,AC} c$. It follows clearly from the above set-theoretic characterizations that $\to_{R/AC} \neq (=_{AC}; \to_R; =_{AC})$ and $\to_{R/AC} \neq (=_{AC}; \to_{R,AC}; =_{AC})$. Consider now the AC-extension closure $\overline{\mathcal{R}}$, where $\overline{R}$ is obtained by adding to R the two extension rules: $(c \cdot d) \cdot x \to e \cdot x$, and $(a \cdot b) \cdot y \to c \cdot y$ *if* $(c \cdot a) \cdot d \to a \cdot e$. We have $\to_{\overline{R}} = \to_R$, and $\to_{\overline{R}/AC} = \to_{R/AC}$. Therefore, we still have $\to_{\overline{R}/AC} \neq (=_{AC}; \to_{\overline{R}}; =_{AC})$. Therefore, it is *impossible* to interpret in $\overline{\mathcal{R}}$ the abstract relations from Section 2 as we did in Example 1 for unsorted and unconditional rewrite theories and we can still do for *unconditional* order-sorted rewrite theories by Corollary 3.

So, what is the *right* interpretation of the abstract relations from Section 2 in the general, conditional case? As pointed out at the beginning of Section 4.2, four out of the five abstract relations can be interpreted exactly as in Example 1, according to clauses (i)–(iv). But that leaves out the issue of how to interpret the remaining abstract relation $\to_{\{R\}}$. This issue can be settled by specifying the remaining clause: (v) the abstract relation $\to_{\{R\}}$ is interpreted as the concrete relation $\to_{R,B}$. Therefore, in the conditional case the two different abstract relations $\to_{\{R\}}$ and $\to_{R:B}$ are interpreted by the *same* concrete relation. That this is the "right" interpretation —because it preserves the abstract properties postulated for the abstract relations— follows from the proposition below.

Proposition 1. *Let $\mathcal{R} = (\Sigma, B, R)$ be a conditional order-sorted rewrite theory where B satisfies requirements (1)–(4) at the beginning of Section 4 and $\mathcal{R}$ is closed under B-extensions. Then $\mathcal{R}$ satisfies:*

1. $\to^\star_{R/B} = \to^\star_{R,B}$, *and*
2. $\to_{R/B} = (=_B; \to_{R,B}; =_B)$.

Proof. The inclusions $\to^\star_{R/B} \supseteq \to^\star_{R,B}$, and $\to_{R/B} \supseteq (=_B; \to_{R,B}; =_B)$ are obvious; we just need to prove the opposite inclusions. The proof is by induction on the size of the closed proof tree of each rewrite of the form $u \to_{R/B} v$ or of the form $u \to^\star_{R/B} v$, where the tree size is the number of goal occurrences. The base cases are either an application of **Reflexivity**, where we obviously have $u \to^\star_{R,B} v \Leftrightarrow u \to^\star_{R/B} v$, or an application of **Replacement** with an *unconditional* rule $l \to r \in R$, so that $u \to_{R/B} v$ is obtained as $u =_B u' \to_R u'[r\theta]_p =_B v$, where $u|_p = l\theta$, which clearly shows $u =_B; \to_{R,B}; =_B v$. The inductive cases are: (i) **Replacement** applied with a *conditional* rule in R, where the induction hypothesis about (1) applied to the $\to^\star_{R/B}$ rewrites in the condition allows us to conclude as for unconditional **Replacement** that $u =_B; \to_{R,B}; =_B v$, and (ii) an instance of **Transitivity** of the form:

$$\frac{u \to_{R/B} w \qquad w \to^\star_{R/B} v}{u \to^\star_{R/B} v}$$

By the induction hypothesis we have $u =_B u' \to_{R,B} w' =_B w$ for some u', w'. Since $\to_{R,B}$ satisfies the **Strict Coherence** property, we have a rewrite $u \to_{R,B}$

w'' with $w'' =_B w'$, and therefore with $w'' =_B w$. Recall now Fact (2) in Remark 2 about $w \to^\star_{R/B} v$. Using the transitivity of $=_B$ and the fact that $\to_{R/B} = (=_B; \to_{R/B})$, it is then easy to prove by induction on the number $n \geq 0$ of $\to_{R/B}$-steps in $w \to^\star_{R/B} v$, that if it has a proof tree T of size k, then $w'' \to^\star_{R/B} v$ also has a proof tree T' of size k, so that the induction hypothesis applies and we get $w'' \to^\star_{R,B} v$, which together with $u \to_{R,B} w''$ gives us by **Transitivity** the desired result $u \to^\star_{R,B} v$. □

Recall from the discussion right after Remark 2 that, in general, the reachability relation $\to^\star_{R,B}$ need not be transitive. However, as an immediate consequence of (1) above and Fact (3) in Remark 2, we obtain:

Corollary 4. *Let $\mathcal{R} = (\Sigma, B, R)$ be a conditional order-sorted rewrite theory where B satisfies requirements (1)–(4) at the beginning of Section 4 and $\mathcal{R}$ is closed under B-extensions. Then the relation $\to^\star_{R,B}$ is transitive.*

In particular, the above corollary applies to any *unconditional* $\mathcal{R}$ closed under B-extensions and satisfying requirements (1)–(4).

Now that the way has been cleared for applying the abstract strict coherence results in Section 2 to a *conditional* order-sorted theory $\mathcal{R} = (\Sigma, B, R)$ closed under B-extensions by interpreting the five abstract relations according to the above clauses (i)–(v), we immediately get:

Corollary 5. *Let $\mathcal{R} = (\Sigma, B, R)$ be a conditional order-sorted rewrite theory where B satisfies requirements (1)–(4) at the beginning of Section 4 and $\mathcal{R}$ is closed under B-extensions. Then $\mathcal{R}$ satisfies the* **Completeness**, **Bisimulation**, **Strict Coherence** *and* **Strict Local Coherence** *properties. Furthermore, if $u =_B u'$ and $u \to_{R,B} v$ at position p with a rule $l \to r$ if $C \in R$ and with substitution θ, then there exists a term v' such that $u' \to_{R,B} v'$ at some position q with a rule $l' \to r'$ if $C \in R$ and with a substitution θ' such that: (i) $v =_B v'$, and (ii) for all $x \in \mathcal{V}ar(C)$ $x\theta = x\theta'$.* □

Corollary 6. *Let $\mathcal{R} = (\Sigma, B, R)$ be as in Corollary 5. Given any chain of $n \geq 0$ R, B-rewrite steps followed by a B-equality step of the form, $u \to_{R,B} u_1 \to_{R,B} u_2 \ldots u_{n-1} \to_{R,B} u_n =_B v$, where at each step a rule $l_i \to r_i$ if $C_i \in R$ has been applied with substitution θ_i, and given any term u' such that $u =_B u'$, there is another chain of $n \geq 0$ rewrite steps followed by a B-equality step of the form, $u' \to_{R,B} u'_1 \to_{R,B} u'_2 \ldots u'_{n-1} \to_{R,B} u'_n =_B v'$, such that: (i) $u_i =_B u'_i$, $1 \leq i \leq n$ and $v =_B v'$, where at each step a rule $l'_i \to r'_i$ if $C_i \in R$ has been applied with substitution θ'_i such that for all $x \in \mathcal{V}ar(C_i)$ $x\theta_i = x\theta'_i$, $1 \leq i \leq n$.* □

Obviously, it also follows from Corollary 5 that $\to_{R,B}$ can be used to rewrite in equivalence classes using the equivalence:

$$[t]_B \to_{R/B} [t']_B \quad \Leftrightarrow \quad (\exists u)\, t \to_{R,B} u \wedge u =_B t'.$$

Note that if $\mathcal{R} = (\Sigma, B, R)$ is closed under B-extensions and therefore strictly B-coherent, we know from Corollary 1 that $\rightarrow_{R/B}$ and $\rightarrow_{R,B}$ are *equi-terminating*. However, since looping can also happen when evaluating conditions, termination of the rewrite relations $\rightarrow_{R/B}$ and $\rightarrow_{R,B}$ is a clearly *insufficient* notion of conditional termination. The appropriate notion is that of *operational termination* [18, 48]. Therefore, the relevant question is whether the R/B and R, B inference systems are *operationally equi-terminating*.

5. Operational Equi-Termination of R/B and R, B Rewriting

5.1. Operational Termination of Declarative Programs

The central idea of declarative programming can be summarized by the identities: (i) *program = logical theory*, and (ii) *computation = deduction*. Different declarative languages correspond to different computational logics $\mathcal{L}$ defined by inference rules parameterized by each *program*, that is, by each *theory* $\mathcal{S}$ in $\mathcal{L}$. The traditional TRS approach to termination assumes a single relation $\rightarrow_R$ that terminates iff it has no infinite chains, that is, iff it is well-founded. Although this idea fits well the case of unconditional rewriting, it breaks down for conditional rewriting,[17] and of course for many other computational logics, which may not involve any rewriting at all. A natural way to express termination for a declarative program $\mathcal{S}$ —i.e., a theory— in a general computational logic $\mathcal{L}$ is as *absence of infinite inference*. This is the key intuition formalized below by the notion of operational termination [18, 48]. To make the paper self-contained, I summarize below the main general notions, illustrating them for the case of conditional rewriting modulo axioms B. I follow closely —with some additional explanations— the presentation in [18].

The axiomatic context in which operational termination is expressed is the theory of general logics [52]; more specifically, its inference aspect, which is captured by the notion of *entailment system*. For our present purposes, all we need to assume is that:

1. Theories $\mathcal{S}$ in a logic $\mathcal{L}$ belong to a set of theories $Th_{\mathcal{L}}$, so that $\mathcal{S} \in Th_{\mathcal{L}}$. For example, an order-sorted conditional rewrite theory $\mathcal{R} = (\Sigma, B, R)$ belongs to the set of theories of *two* logics: (i) $OSRL(R/B)$, based on the relations $\rightarrow_{R/B}$ and $\rightarrow^{\star}_{R/B}$, and (ii) $OSRL(R, B)$, based on the relations $\rightarrow_{R,B}$ and $\rightarrow^{\star}_{R,B}$.

2. For each theory $\mathcal{S} \in Th_{\mathcal{L}}$ there is a set $Form_{\mathcal{L}}(\mathcal{S})$ of *formulas* of $\mathcal{S}$. For example, for $(\Sigma, B, R) \in Th_{OSRL(R/B)}$ (resp. $(\Sigma, B, R) \in Th_{OSRL(R,B)}$) we have:

   $Form_{OSRL(R/B)}(\Sigma, B, R) =$
   $\{t \rightarrow_{R/B} t' \mid t, t' \in \mathcal{T}_{\Sigma}(\mathcal{X})_{[s]}, s \in S\} \cup \{t \rightarrow^{\star}_{R/B} t' \mid t, t' \in \mathcal{T}_{\Sigma}(\mathcal{X})_{[s]}, s \in S\}$

[17] Because of the additional possibility of looping when evaluating a rule's condition.

$Form_{OSRL(R,B)}(\Sigma, B, R) =$
$\{t \to_{R,B} t' \mid t, t' \in \mathcal{T}_\Sigma(\mathcal{X})_{[s]}, s \in S\} \cup \{t \to^\star_{R,B} t' \mid t, t' \in \mathcal{T}_\Sigma(\mathcal{X})_{[s]}, s \in S\}$
where S is the set of sorts of the signature Σ.

3. Each theory $\mathcal{S} \in Th_\mathcal{L}$ has an associated set of *inference rules* $\mathcal{I}_\mathcal{L}(\mathcal{S})$, where each inference rule $\iota \in \mathcal{I}_\mathcal{L}(\mathcal{S})$ is a scheme specifying a (possibly infinite) set of pairs $(\vec{\phi}, \varphi)$, called its *instances*, and denoted $\frac{\vec{\phi}}{\varphi}$, where $\vec{\phi} \in Form_\mathcal{L}(\mathcal{S})^*$ and $\varphi \in Form_\mathcal{L}(\mathcal{S})$.

   For example, for $(\Sigma, B, R) \in Th_{OSRL(R/B)}$ (resp. $(\Sigma, B, R) \in Th_{OSRL(R,B)}$) the corresponding inference systems are those specified for $\to_{R/B}$ and $\to^\star_{R/B}$ (resp. $\to_{R,B}$ and $\to^\star_{R,B}$) in Section 3.2.

The key proof-theoretic notion in such a logic $\mathcal{L}$ is that of a *proof tree*.

Definition 2. *The set of (finite) proof trees for a theory $\mathcal{S}$ in a logic $\mathcal{L}$ and the head of a proof tree are defined inductively as follows. A* proof tree *is*

- *either an* open goal*, simply denoted as φ, where φ is a formula for $\mathcal{S}$; then, we define head$(\varphi) = \varphi$,*
- *or a* non-atomic *tree with φ as its head, denoted as*

$$\frac{T_1 \quad \cdots \quad T_n}{\varphi} \quad (\iota)$$

  where φ is a formula for $\mathcal{S}$, ι is an inference rule in $\mathcal{I}_\mathcal{L}(\mathcal{S})$, and $T_1, \ldots, T_n$ are proof trees such that

$$\frac{head(T_1) \quad \cdots \quad head(T_n)}{\varphi}$$

  is an instance of ι.

We say that a proof tree is closed *whenever it is finite and contains no open goals. If T is a closed proof tree for $\mathcal{S}$, we then write $\mathcal{S} \vdash head(T)$, and call head$(T)$ a* theorem *of $\mathcal{S}$.*

Notice the difference between φ, an open goal, and $\overline{\varphi}$, a goal closed by a rule ι without premisses.

Increasing chains of (finite) proof trees can give rise to infinite proof trees.

Definition 3. *A proof tree T is a* proper prefix *of a proof tree T' if there are one or more open goals $\varphi_1, \ldots, \varphi_n$ in T such that T' is obtained from T by replacing each φ_i by a non-atomic proof tree T_i having φ_i as its head. We denote the proper prefix relation as $T \subset T'$.*

An infinite proof tree *is an infinite increasing chain of finite trees, that is, a sequence $\{T_i\}_{i \in \mathbb{N}}$ such that for all i, $T_i \subset T_{i+1}$.*

We characterize the proof trees with computational meaning (those which are computed by an *interpreter* for $\mathcal{L}$ which solves goals in a proof tree bottom-up and from left to right), by means of the notion of well-formed proof tree.

Definition 4. *We say that a proof tree T is* well-formed *if it is either an open goal, or a closed proof tree, or a proof tree of the form*

$$\frac{T_1 \quad \cdots \quad T_n}{\varphi} \quad (\iota)$$

where for each j T_j is itself well-formed, and there is $i \leq n$ such that T_i is not closed, for any $j < i$ T_j is closed, and each of the T_{i+1} ,...,T_n is an open goal. An infinite proof tree is well-formed *if it is an ascending chain of well-formed finite proof trees. $\mathcal{S}$ is called* operationally terminating *if no infinite well-formed proof tree for $\mathcal{S}$ exists.*

Operational termination intuitively means that, given an initial goal, an interpreter that solves goals bottom-up and from left to right will either succeed in finite time in producing a closed proof tree, or will fail in finite time, not being able to close or extend further any of the possible proof trees, after exhaustively searching all such proof trees.

Lack of operational termination means that there is an infinite *well-formed* proof tree. It follows easily from the above definition that an infinite well-formed proof tree has a *single* infinite branch so that: (i) to the *left* of that infinite branch all maximal subtrees are *finite* and *closed* —that is, there are no *open goals* to the left of the infinite branch— and (ii) all leaves to the *right* of that infinite branch are *goals* that were introduced by an inference step corresponding to an ascending step in the single infinite branch, but *were never expanded* by some subsequent inference step. This is as it should be: since well-formed trees are those built by an interpreter that tries to prove goals *from left to right*, an infinite branch will happen precisely when the interpreter having, say, n goals to solve, has already solved the first $k < n$ goals, but loops (builds an infinite branch) when trying to solve goal $k + 1$. This also means that the remaining goals $k + 2, \ldots, n$, if any, will *never* be explored, and the same will happen (in general for a different k) at each step up the infinite branch.

The following observation, made by one of the paper's anonymous referees, may also be helpful: an operationally terminating theory may have many *harmless* infinite non-well-formed proof trees. For example, for t any term in *any* rewrite theory, we can always apply **Transitivity** forever, without ever trying to solve the first goal generated at each step, and thus build the nonsensical infinite proof tree:

$$\frac{t \rightarrow_{R,B} t \qquad \dfrac{\vdots}{t \rightarrow^{\star}_{R,B} t}}{t \rightarrow^{\star}_{R,B} t}$$

Note that this tree is *not* well-formed precisely because of the presence of *open goals* of the form $t \rightarrow_{R,B} t$ to the left of the infinite branch.

Example 9. *Consider the unsorted rewrite theory* $(\Sigma, \emptyset, R)$ *from [48], where* Σ *has two constants,* a *and* b*, and a unary function symbol* f*, and* R *has the single conditional rule* $a \to b$ *if* $f(a) \to b$*. It is easy to see that the rewrite relation* $\to_R$ *is the empty set and is therefore trivially well-founded. That is, there are no* $\to_R$*-chains at all, so that the rewrite relation is trivially* terminating.

However, $(\Sigma, \emptyset, R)$ *is* not *operationally terminating, because when we try to rewrite the term* a *we* loop *when trying to satisfy the rule's condition and generate the following infinite well-formed proof tree:*

$$\dfrac{\dfrac{\dfrac{\dfrac{\vdots}{f(a) \to_R^\star b}}{f(a) \to_R f(b)} \quad f(b) \to_R^\star b}{f(a) \to_R^\star b}}{a \to_R b}$$

For the inference systems of the logics $OSRL(R/B)$ and $OSRL(R, B)$, as already mentioned in Section 3.3, the key challenge for an interpreter is how to guess the substitution θ in the **Replacement** rule, which may require using *symbolic constraints*. As further discussed in Section 6.2, a much easier to implement interpreter for $OSRL(R, B)$ treats the —still very general— special case where the rules in the conditional order-sorted rewrite theory $\mathcal{R}$ generalize those of a deterministic 3-CTRS [60], so that the substitution θ can be computed *incrementally*, as each subgoal in the condition gets solved.

Example 10. *(Operational Termination of the Rewrite Theory of Idempotent Semigroups). The rewrite theory* $\overline{\mathcal{R}}_{Idemp.Sgr}$ *from Example 8 is strictly* B*-coherent. Furthermore, it belongs to the just-mentioned class of rewrite theories generalizing deterministic 3-CTRS. In fact, since it has no extra variables in the conditions of its conditional rules other than those in the rule's lefthand side, there is no need at all to incrementally compute the substitution* θ *to evaluate a rule application:* θ *is just the* B*-matching substitution obtained by* B*-matching a subterm of the subject term to the rule's lefthand side.*

Therefore, if we can show that $\overline{\mathcal{R}}_{Idemp.Sgr}$ *is operationally terminating in the logic* $OSRL(R, B)$*, by the equi-termination result in Theorem 3 below we will have proved that* $\overline{\mathcal{R}}_{Idemp.Sgr}$ *is also operationally terminating in the logic* $OSRL(R/B)$*.*

By the characterization of operational termination for order-sorted conditional R, B*-rewriting in [49], we will be done if we can show that* $\overline{\mathcal{R}}_{Idemp.Sgr}$ *is* quasi-decreasing modulo B*. But since* $\overline{\mathcal{R}}_{Idemp.Sgr}$ *has rules with no extra variables in their conditions, by [60] our task is even simpler: we will be done if we can show that* $\overline{\mathcal{R}}_{Idemp.Sgr}$ *is* decreasing *(modulo* B*) in the sense of [60]. In particular, we will be done if we can find a* B*-compatible well-founded relation* $>$ *closed under substitutions, having the subterm property (more on this below), and such that for each rule* $l \to r$ *if* $\bigwedge_{i \in I} l_i \to r_i$ *in* R *we have* $l > r$*, and* $l > l_i$ *for each* $i \in I$*.*

We can define $>$ *using a* polynomial interpretation *of each* $f \in \Sigma$ *of* n *arguments as an* n*-argument polynomial function* $[f] : \mathbb{N}_{\geq 3}^n \to \mathbb{N}_{\geq 3}$ *with natural number coefficients on the set* $\mathbb{N}_{\geq 3}$ *of natural numbers greater than or equal to 3. The interpretation* $f \mapsto [f]$ *for each* $f \in \Sigma$ *is then homomorphically extended to an interpretation* $t \mapsto [t]$ *on* Σ*-tems in the usual way, so that* $[f(u_1, \ldots, u_n)] = [f] \circ ([u_1], \ldots, [u_n])$*, where* $\_ \circ \_$ *denotes function composition, and* $([u_1], \ldots, [u_n])$ *denotes the* tupling *of the polynomial functions* $[u_1], \ldots, [u_n]$*. We then define* $t > t'$ *iff we have the functional inequality* $[t] > [t']$*. Since the arguments are naturals greater or equal to* 3*, for any such polynomial function* $p(x_1, \ldots, x_n)$ *we have* $p(x_1, \ldots, x_n) > x_i$*,* $1 \leq i \leq n$*. This ensures that for any such polynomial interpretation, give a term* $f(u_1, \ldots, u_n)$ *we always have the* subterm property $[f(u_1, \ldots, u_n)] > [u_i]$*,* $1 \leq i \leq n$*.*

The polynomial interpretation $f \mapsto [f]$ *we can use is as follows: we interpret juxtaposition* XY *as polynomial multiplication* $X \cdot Y$*, set union* S, S' *as polynomial addition* $S + S'$*, the set equality predicate* $S \sim S'$ *also as polynomial addition* $S + S'$*, the list-to-set operation* $\{X\}$ *as the polynomial* $2X$*, and the truth constant* tt *as 3. Note that, since polynomial addition and multiplication are associative-commutative, this order is trivially compatible with the axioms* B *in* $\overline{\mathcal{R}}_{Idemp.Sgr}$*, that is, it does not depend on the representative chosen in a* B*-equivalence class. Note also that, because of its homomorphic extension to terms,* $>$ *is closed under substitutions, i.e.,* $t > t' \Rightarrow t\sigma > t'\sigma$ *for any substition* σ*. Finally, it is easy to check that: (i) for all rules* $l \to r$ *if* $\bigwedge_{i \in I} l_i \to r_i$ *in* R *we have* $l > r$ *and* $l > l_i$ *for each* $i \in I$ *under this interpretation.*

Let me illustrate how this works for a couple of rules. For the idempotency rule $LL \to L$ *this boils down to the fact that for polynomial functions over the natural numbers greater than or equal to 3 we have* $L \cdot L > L$*. For one of the* B*-extended conditional rules, for example the rule* $X(L(PQ)) \to X(LQ)$ *if* $\{L\} \sim \{Q\} \to tt \wedge \{LP\} \sim \{L\} \to tt$*, this boils down to the fact that for polynomial functions over the natural numbers greater than or equal to 3 we have: (a)* $X \cdot L \cdot P \cdot Q > X \cdot L \cdot P$*, (b)* $X \cdot L \cdot P \cdot Q > 2L + 2Q$*, and (c)* $X \cdot L \cdot P \cdot Q > 2(L \cdot P) + 2L = 2L \cdot (P + 1)$*.*

5.2. Proof of Operational Equi-Termination of R/B *and* R, B *Rewriting*

Given a conditional order-sorted rewrite theory $\mathcal{R} = (\Sigma, B, R)$ an interpreter evaluating goals for $\mathcal{R}$ in the logic $OSRL(R, B)$ is much easier to implement and much more efficient than an interpreter evaluating similar goals in the logic $OSRL(R/B)$. Of course, without the strict coherence of the rules R an interpreter for $\mathcal{R}$ in the logic $OSRL(R, B)$ would be *incomplete*. So we should in any case assume that $\mathcal{R}$ is closed under B-extensions and therefore strictly coherent. But, assuming that, what can we say about operational termination? Is it the same in both logics?

Theorem 3. *Let* $\mathcal{R} = (\Sigma, B, R)$ *be closed under the* B*-extensions. Then* $\mathcal{R}$ *is operationally terminating in* $OSRL(R/B)$ *iff* $\mathcal{R}$ *is operationally terminating in* $OSRL(R, B)$*.*

PROOF. The proof of the ($\Rightarrow$) direction is straightforward. Just notice that, by systematically changing each goal $u \rightarrow_{R,B} u'$, or $v \rightarrow^{\star}_{R,B} v'$, in a well-formed proof tree for $\mathcal{R}$ in $OSRL(R, B)$ into a corresponding goal $u \rightarrow_{R/B} u'$, or $v \rightarrow^{\star}_{R/B} v'$, we obtain a well-formed proof tree for $\mathcal{R}$ in $OSRL(R/B)$. This is because, up to such renaming of goals, the corresponding instances of the **Reflexivity** and **Transitivity** rules are the same in both logics; and the **Replacement** rule is *more restrictive* in $OSRL(R, B)$ than in $OSRL(R/B)$. Therefore, if $\mathcal{R}$ is operationally terminating in $OSRL(R/B)$ there are no well-formed infinite proof trees for $\mathcal{R}$ in $OSRL(R/B)$ and, *a fortiori*, no well-formed infinite proof trees for $\mathcal{R}$ in $OSRL(R, B)$.

To prove the ($\Leftarrow$) direction we reason by contradiction and assume that $\mathcal{R}$ is operationally terminating in $OSRL(R, B)$ but there is a well-formed infinite proof tree for $\mathcal{R}$ in $OSRL(R/B)$. We will reach the desired contradiction if we can then show that there is a well-formed infinite proof tree for $\mathcal{R}$ in $OSRL(R, B)$.

Let $\{T_i\}_{i\in\mathbb{N}}$ be the well-formed infinite proof tree for $\mathcal{R}$ in $OSRL(R/B)$. Without loss of generality we may assume that the well-formed finite tree inclusions $T_i \subset T_{i+1}$ are such that T_0 is an open goal, and for each $i \in \mathbb{N}$, T_{i+1} is just the proper prefix of T_i obtained by a one-step expansion of the *current goal* ϕ of T_i, denoted $\phi = \mathit{cgoal}(T_i)$, by an instance of an inference rule having ϕ as its head; where for a non-closed finite proof tree we define: (i) $\mathit{cgoal}(\varphi) = \varphi$ for an open goal, and (ii) $\mathit{cgoal}(\frac{T_1 \quad \cdots \quad T_n}{\varphi}) = \mathit{cgoal}(T_j)$, where T_j is the leftmost non-closed proof tree in the sequence $T_1 \quad \cdots \quad T_n$. We can obtain the desired contradiction by a sequence of lemmas.

The proof of the following lemma is by induction on n and is left to the reader.

Lemma 1. *Let $\{T_i\}_{i\in\mathbb{N}}$ be a well-formed infinite proof tree for $\mathcal{R}$ in $OSRL(R/B)$ satisfying the assumptions above. Then for each $i \in \mathbb{N}$ all open goals in T_i other than $\mathit{cgoal}(T_i)$ are of the form $u \rightarrow^{\star}_{R/B} v$ for some u, v.* □

Call a well-formed finite tree T for $\mathcal{R}$ in $OSRL(R/B)$ $\rightarrow^{\star}$*current* iff $\mathit{cgoal}(T)$ is of the form $u \rightarrow^{\star}_{R/B} v$ for some u, v.

Lemma 2. *Let $\{T_i\}_{i\in\mathbb{N}}$ be a well-formed infinite proof tree for $\mathcal{R}$ in $OSRL(R/B)$ satisfying the assumptions above. Then for each $i \in \mathbb{N}$ there is a smallest $j > i$ such that T_j is $\rightarrow^{\star}$*current*.*

PROOF. Let $\mathit{cgoal}(T_i) = t \rightarrow_{R/B} t'$. Since the only rule that can be applied to it is **Replacement**, if the rule applied is unconditional, then the result follows from Lemma 1; and if it is conditional, then $\mathit{cgoal}(T_{i+1}) = u_1\theta \rightarrow^{\star}_{R/B} v_1\theta$ for some θ, where $u_1 \rightarrow v_1$ is the first goal in the rule's condition. Instead, if $\mathit{cgoal}(T_i) = t \rightarrow^{\star}_{R/B} t'$, and we then apply **Reflexivity**, the result again follows from Lemma 1. Otherwise, we have applied an instance of **Transitivity** of the form $\frac{t \rightarrow_{R/B} t'' \quad t'' \rightarrow^{\star}_{R/B} t'}{t \rightarrow^{\star}_{R/B} t'}$, and $\mathit{cgoal}(T_{i+1}) = t \rightarrow_{R/B} t''$. But then T_{i+2} must be

obtained by an application of **Replacement** to expand $t \rightarrow_{R/B} t''$, and the above reasoning shows that T_{i+2} is $\rightarrow^\star$*current*. □

As a consequence of the above two lemmas we then obtain:

Corollary 7. *Let $\{T_i\}_{i\in\mathbb{N}}$ be a well-formed infinite proof tree for $\mathcal{R}$ in OSRL(R/B) satisfying the assumptions above. The set $\{T_0\} \cup \{T_i \mid i > 0,\ T_i \ \rightarrow^\star$ current$\}$ is an infinite chain of well-formed trees, which we can denote $\{T_{\alpha(i)}\}_{i\in\mathbb{N}}$ for a monotonically increasing function α with $\alpha(0) = 0$. Furthermore, for each $i > 0$ all open goals in $T_{\alpha(i)}$ are of the form $u \rightarrow^\star_{R/B} v$ for some u, v.* □

The next useful notion is that of *B-similarity* between a proof tree T for $\mathcal{R}$ in *OSRL*(R/B) and a proof tree U for $\mathcal{R}$ in *OSRL*(R, B), denoted $T \approx_B U$ and defined inductively as follows: (i) for open goals we have $(u \rightarrow_{R/B} v) \approx_B (u' \rightarrow_{R,B} v')$ (resp. $(u \rightarrow^\star_{R/B} v) \approx_B (u' \rightarrow^\star_{R,B} v')$) iff $u =_B u'$ and $v =_B v'$; (ii) for a proof tree T for $\mathcal{R}$ in *OSRL*(R/B) of the form $\frac{T_1 \quad \cdots \quad T_n}{\varphi}\,(\iota)$, a proof tree U for $\mathcal{R}$ in *OSRL*(R, B) satisfies $T \approx_B U$ iff U is of the form $\frac{U_1 \quad \cdots \quad U_n}{\phi}\,(\iota)$, with $\varphi \approx_B \phi$ and $T_i \approx_B U_i$, $1 \leq i \leq n$.

The desired contradiction then follows from the following lemma:

Lemma 3. *Let $\{T_{\alpha(i)}\}_{i\in\mathbb{N}}$ be as in Corollary 7. Then there is a well-formed infinite proof tree for $\mathcal{R}$ in OSRL(R, B) of the form $\{U_{\alpha(i)}\}_{i\in\mathbb{N}}$ such that for each $i \in \mathbb{N}$ $T_{\alpha(i)} \approx_B U_{\alpha(i)}$.*

Proof. We distinguish two cases, depending on $T_{\alpha(0)} = T_0$.

Case 1: T_0 is of the form $u \rightarrow_{R/B} v$, then $T_{\alpha(1)} = T_1$ must be obtained by application of **Replacement** with a rule $l \rightarrow r \ \ if \ \ u_1 \rightarrow v_1 \wedge \ldots \wedge u_n \rightarrow v_n$ in R, so that T_1 is of the form:

$$\frac{u_1\theta \rightarrow^\star_{R/B} v_1\theta \quad \ldots \quad u_n\theta \rightarrow^\star_{R/B} v_n\theta}{u \rightarrow_{R/B} v}$$

with $t, u, v \in \mathcal{T}_\Sigma(\mathcal{X})$, $p \in \mathcal{P}(t)$, and θ a substitution, such that $u =_B t[l\theta]_p$ and $v =_B t[r\theta]_p$.

But since $t[l\theta]_p \rightarrow_R t[r\theta]_p$ is a special case of $t[l\theta]_p \rightarrow_{R,B} t[r\theta]_p$, since $\mathcal{R} = (\Sigma, B, R)$ is closed under B-extensions, we can apply Corollary 2, so that there is a rule $l' \rightarrow r' \ \ if \ \ u_1 \rightarrow v_1 \wedge \ldots \wedge u_n \rightarrow v_n$ in R and a proof tree U_1 for $\mathcal{R}$ in *OSRL*(R, B) of the form:

$$\frac{u_1\theta' \rightarrow^\star_{R,B} v_1\theta' \quad \ldots \quad u_n\theta' \rightarrow^\star_{R,B} v_n\theta'}{u \rightarrow_{R,B} v'}$$

with $v =_B v'$ and $\theta(x) = \theta'(x)$ for each $x \in \mathcal{V}ar(u_1 \rightarrow v_1 \ \wedge \ldots \wedge \ u_n \rightarrow v_n)$. Therefore, choosing $U_0 = u \rightarrow_{R,B} v'$ we have $T_0 \approx_B U_0$, and $T_1 \approx_B U_1$.

Case 2: T_0 is of the form $u \rightarrow^\star_{R/B} v$. We then choose $U_0 = u \rightarrow^\star_{R,B} v$.

To prove the lemma, call an ascending finite chain of well-formed proof trees for $\mathcal{R}$ in *OSRL*(R, B), $\{V_i\}_{0\leq i\leq n}$, *extensible* iff there is a well-formed proof tree

V_{n+1} obtained from V_n by extending $cgoal(V_n)$ by a well-formed proof tree. Then the existence of an infinite chain of well-formed proof trees $\{U_{\alpha(i)}\}_{i\in\mathbb{N}}$ such that for each $i \in \mathbb{N}$ $T_{\alpha(i)} \approx_B U_{\alpha(i)}$ is equivalent to the existence of an "infinite chain of extensible finite chains" $\{U_0\} \subset \{U_{\alpha(i)}\}_{0\leq i\leq 1} \subset \ldots \{U_{\alpha(i)}\}_{0\leq i\leq n} \subset \ldots$ such that for each $i \in \mathbb{N}$ $T_{\alpha(i)} \approx_B U_{\alpha(i)}$. We now reason by contradiction. Suppose such a chain of extensible finite chains with the required B-similarity property does not exist. In particular, it does not exist for our choices of U_0 and $U_{\alpha(1)}$ in **Case 1**, and for our choice of U_0 in **Case 2**. This means that, for such choices, there is a finite chain $\{U_{\alpha(i)}\}_{0\leq i\leq n}$ with $T_{\alpha(i)} \approx_B U_{\alpha(i)}$, $0 \leq i \leq n$, such that either no extension to a $U_{\alpha(n+1)}$ exists, or for any such extension $U_{\alpha(n+1)}$ we have $T_{\alpha(n+1)} \not\approx_B U_{\alpha(n+1)}$. Note that, in both **Case 1** and **Case 2**, $U_{\alpha(n)}$ is $\rightarrow^\star$*current*. Let $cgoal(U_{\alpha(n)}) = u' \rightarrow^\star_{R,B} v'$. Then, since $T_{\alpha(n)} \approx_B U_{\alpha(n)}$, we have $cgoal(T_{\alpha(n)}) = u \rightarrow^\star_{R/B} v$ with $u =_B u'$ and $v =_B v'$, and $T_{\alpha(n+1)}$ is obtained by either an application of **Reflexivity**, which, can then also be applied to $u' \rightarrow^\star_{R,B} v'$, yielding the contradiction of an extension $U_{\alpha(n+1)}$ with $T_{\alpha(n+1)} \approx_B U_{\alpha(n+1)}$; or $T_{\alpha(n+1)}$ is obtained by an application of **Transitivity** of the form $\frac{u\rightarrow_{R/B} w \;\; w\rightarrow^\star_{R/B} v}{u\rightarrow^\star_{R/B} v}$ followed by an application of **Replacement** to expand $u \rightarrow_{R/B} w$ to

$$\frac{u_1\theta \rightarrow^\star_{R/B} v_1\theta \;\; \ldots \;\; u_n\theta \rightarrow^\star_{R/B} v_n\theta}{u \rightarrow_{R/B} w}$$

But, applying again Corollary 2, we then get an application of **Replacement**

$$\frac{u_1\theta' \rightarrow^\star_{R,B} v_1\theta' \;\; \ldots \;\; u_n\theta' \rightarrow^\star_{R,B} v_n\theta'}{u' \rightarrow_{R,B} w'}$$

with $w =_B w'$ and $\theta(x) = \theta'(x)$ for each $x \in \mathcal{V}ar(u_1 \rightarrow v_1 \;\wedge \ldots \wedge\; u_n \rightarrow v_n)$. But this means that we also have an application of **Transitivity** of the form $\frac{u'\rightarrow_{R,B} w' \;\; w'\rightarrow^\star_{R,B} v'}{u'\rightarrow^\star_{R,B} v'}$ which, followed by the application of **Replacement** to the goal $u' \rightarrow_{R,B} w'$, yields the contradiction of a well-formed proof tree $U_{\alpha(n+1)}$ with $T_{\alpha(n+1)} \approx_B U_{\alpha(n+1)}$. □

This finishes the proof of the theorem. □

6. The Church-Rosser Property

As mentioned in Section 3.2, the meaning of $\mathcal{R} = (\Sigma, B, R)$ need not be equational. However, it *can* be equational, and many applications to functional programming, theorem proving, unification theory, and equational logic use an equational interpretation. Given an unconditional equational theory (Σ, E) we can orient its equations E from left to right as rewrite rules $\vec{E}$ to obtain a TRS $(\Sigma, \vec{E})$. The Church-Rosser property is then the key property giving us a

complete method of *reducing* equational reasoning to rewriting. $(\Sigma, \vec{E})$ is said to be *Church-Rosser* iff for any Σ-terms t, t' we have the equivalence:

$$t =_E t' \;\Leftrightarrow\; t \downarrow_{\vec{E}} t',$$

where the *joinability relation* $t \downarrow_{\vec{E}} t'$ is defined by the equivalence:

$$(\natural)\;\; t \downarrow_{\vec{E}} t' \;\;\Leftrightarrow\;\; (\exists u)\, t \rightarrow^*_{\vec{E}} u \wedge t' \rightarrow^*_{\vec{E}} u.$$

It is well-known and very easy to prove that $(\Sigma, \vec{E})$ is confluent iff it is Church-Rosser (this is called the *Church-Rosser Theorem*).

In the present setting, the obvious generalization of the Church-Rosser property assumes a *conditional* order-sorted equational theory of the form $(\Sigma, E \cup B)$, where the equations B are unconditional. One would then like to call the conditional order-sorted rewrite theory $(\Sigma, B, \vec{E})$ *Church-Rosser modulo* B iff it satisfies the equivalence:

$$t =_{E \cup B} t' \;\Leftrightarrow\; t \downarrow^{\star}_{\vec{E}/B} t',$$

where the $\star$-joinability modulo B relation $t \downarrow^{\star}_{\vec{E}/B} t'$ is defined in the "obvious" way by replacing in $(\natural)$ $t \downarrow_{\vec{E}} t'$ by $t \downarrow^{\star}_{\vec{E}/B} t'$, and $\rightarrow^*_{\vec{E}}$ by $\rightarrow^{\star}_{\vec{E}/B}$. I say "obvious" in quotes since, as pointed out in Section 3.3, the relation $\rightarrow^{\star}_{\vec{E}/B}$ is *not* the reflexive-transitive closure of $\rightarrow_{\vec{E}/B}$. One then would also like to prove a *Church-Rosser Theorem modulo* B, showing that, under suitable conditions, $(\Sigma, B, \vec{E})$ is Church-Rosser modulo B iff it is *confluent* modulo B, where the "obvious" definition of confluence modulo B is that for all u, v, v' such that $u \rightarrow^{\star}_{\vec{E}/B} v$ and $u \rightarrow^{\star}_{\vec{E}/B} v'$ there is a term w such that $v \rightarrow^{\star}_{\vec{E}/B} w$ and $v' \rightarrow^{\star}_{\vec{E}/B} w$ (see Definition 5, Lemma 5, and Remark 4 below for a full justification of these "obvious" definitions of joinability and confluence modulo B, and for a comparison with other similar definitions in the literature).

In contrast with the trivial proof of the Church-Rosser Theorem in the unconditional case and without axioms, proving a *conditional* Church-Rosser Theorem *modulo* axioms B is nontrivial. To begin with, if the equations E are conditional, it is not entirely obvious what the rules $\vec{E}$ mean, since we have to deal with their conditions; also, even in the unsorted case, the notion of confluence modulo B is somewhat more subtle (see [38], and Definition 5, Lemma 5 and Remark 4 below for the general case). Furthermore, the easy road to prove the Church-Rosser Theorem via abstract reduction relations is now blocked by the simple fact that, whereas in the unconditional case both equational reasoning and rewriting can be reduced to stringing rewrite steps together, in the conditional case —as made clear in Section 3.3– we need to deal with *inference systems*, both for equational reasoning and for rewriting, whose mutual relationships are considerably less obvious. All this makes the trivial proof of the Church-Rosser Theorem in the unconditional case non-trivial for conditional theories. Indeed, even for CTRSs (i.e., Σ unsorted and $B = \emptyset$), it is well-known that the Church-Rosser theorem

does not hold for an arbitrary confluent theory $\mathcal{R}$ without imposing additional conditions on $\mathcal{R}$ (see [72] and Example 11 below).

Of course, one of the main motivations for a Church-Rosser Theorem is to make equational theories *decidable*; however, as further discussed in Section 6.2, the issue of decidability of the equality relation by rewriting is also nontrivial and particularly subtle in the conditional case. For all these reasons, in Sections 6.1 and 6.2 I state two increasingly stronger versions of a conditional Church-Rosser Theorem modulo axioms B. Furthermore, in Section 6.2 I briefly discuss checkable conditions for confluence, and therefore for the Church-Rosser property, when $\mathcal{R} = (\Sigma, B, R)$ is what I call a *strongly deterministic* conditional rewrite theory. This fully connects the results in [21] with those in this paper.

6.1. The Church-Rosser Property for Conditional Rewriting Modulo B

Given a sensible order-sorted signature Σ, a Σ-*conditional equation* is an implication formula $u_1 = v_1 \wedge \ldots \wedge u_n = v_n \Rightarrow t = t'$, hereafter written:

$$t = t' \ \ \textit{if} \ \ u_1 = v_1 \wedge \ldots \wedge u_n = v_n,$$

where $t = t'$ and $u_1 = v_1, \ldots, u_n = v_n$ are Σ-equations. A *conditional equational theory* is then a pair (Σ, E), with Σ a sensible order-sorted signature, and E a set of conditional Σ-equations.

To simplify the exposition and avoid explicit universal quantification, I will assume throughout that Σ has non-empty sorts (recall from Section 3.1 that we also always assume Σ to be kind-complete, preregular, and sensible; in the unsorted and many-sorted cases all this just boils down to Σ having non-empty sorts and being unambiguous).

Here is a simple inference system for conditional order-sorted equational logic under the above assumptions on Σ. Given an order-sorted conditional equational theory (Σ, E), its *theorems* are those Σ-equations that can be derived by finite application of the following inference rules:

- **Reflexivity**. For each Σ-term t, $\dfrac{}{t = t}$
- **Replacement**. For either $(t = t' \ \ \textit{if} \ \ u_1 = v_1 \wedge \ldots \wedge u_n = v_n) \in E$, or $(t' = t \ \ \textit{if} \ \ u_1 = v_1 \wedge \ldots \wedge u_n = v_n) \in E$, substitution θ, and Σ-term u with position p such that $u|_p = t\theta$,

$$\frac{u_1\theta = v_1\theta \ \ \ldots \ \ u_n\theta = v_n\theta}{u = u[t'\theta]_p}$$

- **Transitivity**. For t_1, t_2, t_3 Σ-terms,

$$\frac{t_1 = t_2 \qquad t_2 = t_3}{t_1 = t_3}$$

It is easy to check that, under the above assumptions on Σ, this inference system is equivalent to similar inference systems in [31, 54], which have been proved *sound and complete* with respect to the model-theoretic semantics provided by order-sorted algebras as models of conditional order-sorted equational theories.

The next order of business is to relate equational logic to rewriting. We can begin with a very simple question: can equational deduction in a conditional order-sorted equational theory be *reduced* to conditional rewriting deduction for *any* such theory? The answer is in the affirmative, as follows. Assume that the conditional equational theory is of the form $(\Sigma, E \cup B)$, where the equations B are unconditional (note that we can choose $B = \emptyset$ as a special case). Our desired order-sorted conditional rewrite theory simulating $(\Sigma, E \cup B)$ is of the form $(\Sigma, B, \overset{\leftrightarrow}{E})$, where $\overset{\leftrightarrow}{E} = \overrightarrow{E} \cup \overleftarrow{E}$, and

- $\overrightarrow{E} = \{t \to t' \ \textit{if} \ u_1 \to v_1 \wedge \ldots \wedge u_n \to v_n \mid (t = t' \ \textit{if} \ u_1 = v_1 \wedge \ldots \wedge u_n = v_n) \in E\}$
- $\overleftarrow{E} = \{t' \to t \ \textit{if} \ u_1 \to v_1 \wedge \ldots \wedge u_n \to v_n \mid (t = t' \ \textit{if} \ u_1 = v_1 \wedge \ldots \wedge u_n = v_n) \in E\}$.

Note the left-to-right orientation for the conditions in both cases.

The symbol $\vdash$ will always denote *provability* in the given inference system. Thus, $(\Sigma, E) \vdash u = v$ means that the equation $u = v$ is provable in the equational theory (Σ, E), and $(\Sigma, B, R) \vdash u \to^\star_{R/B} v$ means that $u \to^\star_{R/B} v$ is provable in the rewrite theory (Σ, B, R). The key lemma reducing conditional equational deduction to conditional rewriting is then,

Lemma 4. *We have the equivalence:*

$$(\Sigma, E \cup B) \vdash u = v \ \Leftrightarrow \ (\Sigma, B, \overset{\leftrightarrow}{E}) \vdash u \to^\star_{\overset{\leftrightarrow}{E}/B} v.$$

Proof. Since each rewrite proof can be viewed as an equality proof carried out in a more restricted inference system, the ($\Leftarrow$) implication is easy and left to the reader. To prove the ($\Rightarrow$) implication we reason by structural induction on the equational proof trees. The case of an equality $t = t$ obtained by **Reflexivity** follows trivially from the rewriting **Reflexivity** rule. The case of a proof of $t_1 = t_3$ by **Transitivity** follows by induction using Fact (3) in Remark 2. The case of a proof of $u = u[t'\theta]_p$ by **Replacement** follows by induction from rewriting **Replacement** by applying the corresponding conditional rule in either $\overrightarrow{E}$ or $\overleftarrow{E}$ to u at position p with substitution θ. □

Since the relations $\to^\star_{R/B}$ and $\to^\star_{R,B}$ are not reflexive-transitive closures, a few words should be said about confluence and joinability. Recall the notation $\to^*_{R/B}$ and $\to^*_{R,B}$ for the respective reflexive-transitive closures of $\to_{R/B}$ and $\to_{R,B}$. The relevant notions are summarized in the following definition:

Definition 5. *Given a conditional order-sorted rewrite theory $\mathcal{R} = (\Sigma, B, R)$, two terms $u, v \in \mathcal{T}_\Sigma(\mathcal{X})$ are called:*

1. *R/B-*joinable*, denoted $u \downarrow_{R/B} v$, (resp. R, B-*joinable*, denoted $u \downarrow_{R,B} v$) iff there exist $w, w' \in \mathcal{T}_\Sigma(\mathcal{X})$ such that $u \to^*_{R/B} w =_B w' \ {}_{R/B}\overset{*}{\leftarrow} v$ (resp. such that $u \to^*_{R,B} w =_B w' \ {}_{R,B}\overset{*}{\leftarrow} v$).*

2. *$\star$-R/B-*joinable*, denoted $u \downarrow^\star_{R/B} v$, (resp. $\star$-R, B-*joinable*, denoted $u \downarrow^\star_{R,B} v$) iff there exists $w \in \mathcal{T}_\Sigma(\mathcal{X})$ such that $u \to^\star_{R/B} w \ {}_{R/B}\overset{\star}{\leftarrow} v$ (resp. such that $u \to^\star_{R,B} w \ {}_{R,B}\overset{\star}{\leftarrow} v$).*

The relation $\to_{R/B}$ (resp. $\to_{R,B}$) is called:

1. Confluent *iff for each $u, v, t \in \mathcal{T}_\Sigma(\mathcal{X})$, $u \ {}_{R/B}\overset{*}{\leftarrow} t \to^*_{R/B} v$ implies $u \downarrow_{R/B} v$ (resp. $u \ {}_{R,B}\overset{*}{\leftarrow} t \to^*_{R,B} v$ implies $u \downarrow_{R,B} v$).*

2. $\star$-Confluent *iff for each $u, v, t \in \mathcal{T}_\Sigma(\mathcal{X})$, $u \ {}_{R/B}\overset{\star}{\leftarrow} t \to^\star_{R/B} v$ implies $u \downarrow^\star_{R/B} u$ (resp. $u \ {}_{R,B}\overset{\star}{\leftarrow} t \to^\star_{R,B} v$ implies $u \downarrow^\star_{R,B} v$).*

Note the extra B-equality $w =_B w'$ needed in the notions of R/B-joinability and R, B-joinability, and therefore on the notions of R/B-confluence and R, B-confluence, which makes such notions non-standard. By contrast, no such extra B-equality $w =_B w'$ is needed for $\star$-R/B-joinability, $\star$-R, B-joinability, $\star$-R/B-confluence, and $\star$-R, B-confluence, which makes such notions more natural. Their key relationships can be summarized in the following easy lemma:

Lemma 5. *Given a conditional order-sorted rewrite theory $\mathcal{R} = (\Sigma, B, R)$,*

1. *For each $u, v \in \mathcal{T}_\Sigma(\mathcal{X})$, $u \downarrow_{R/B} v \Leftrightarrow u \downarrow^\star_{R/B} v$, and $u \downarrow_{R,B} v \Leftrightarrow u \downarrow^\star_{R,B} v$.*

2. *$\to_{R/B}$ is confluent iff it is $\star$-confluent.*

3. *$\to_{R,B}$ $\star$-confluent implies $\to_{R,B}$ confluent, but the converse does not hold in general.*

Furthermore, if B satisfies the assumptions in Section 4 and $\mathcal{R}$ is closed under B-extensions we have:

1. *$\forall u, v \in \mathcal{T}_\Sigma(\mathcal{X})$ $u \downarrow_{R/B} u \Leftrightarrow u \downarrow^\star_{R/B} u \Leftrightarrow u \downarrow^\star_{R,B} u \Leftrightarrow u \downarrow_{R,B} u$, and*

2. *$\to_{R/B}$ confluent iff $\to_{R/B}$ $\star$-confluent iff $\to_{R,B}$ $\star$-confluent iff $\to_{R,B}$ confluent.*

For a counterexample showing that in general $\to_{R,B}$ confluent does not imply $\to_{R,B}$ $\star$-confluent, consider an unsorted signature with constants a, b, c, d binary AC symbol $\_\cdot\_$ and R with just one rule $a \cdot b \to c$. It is easy to check that $\to_{R,AC}$ is confluent. However, we have $(c \cdot a) \cdot b \ {}_{R,AC}\overset{\star}{\leftarrow} c \cdot (a \cdot b) \to^\star_{R,AC} c \cdot c$, where both $(c \cdot a) \cdot b$ and $c \cdot c$ are $\to_{R,AC}$-irreducible and obviously $(c \cdot a) \cdot b \neq_{AC} c \cdot c$.

Remark 4. *Lemma 5 implictly answers the question of how confluence modulo B can be understood as confluence of an* abstract relation. *The* abstract *notion of confluence is that we have a set T and a binary relation $\Rightarrow$ on T such that for each $u, v, v' \in T$ such that $u \Rightarrow^* v$ and $u \Rightarrow^* v'$ there is a $w \in T$ such that $v \Rightarrow^* w$ and $v' \Rightarrow^* w$, where $\Rightarrow^*$ denotes the reflexive-transitive closure of $\Rightarrow$.*

The essential point is that, even when $\mathcal{R} = (\Sigma, B, R)$ is closed under B-extensions, satisfies conditions (1)–(4), and is confluent *modulo B in any of the above senses (which become* equivalent *under those assumptions), in general we* cannot *interpret the abstract relation $\Rightarrow$ as either $\rightarrow_{R/B}$ or $\rightarrow_{R,B}$.*

Let $\mathcal{R} = (\Sigma, C, R)$ be unsorted with constants a, b, c, d and a binary operator $+$, and let C be the commutativity axiom $x + y = y + x$. Let R consist of the rules $x + a \rightarrow c + x$, and $y + b \rightarrow d + y$. $\mathcal{R}$ is already closed under C-extensions, because no extensions exist for commutativity. It is also easy to check that $\mathcal{R}$ is terminating, because the sum of the number of occurrences of a's and b's in a term decreases by rewriting; and that it is locally $\rightarrow_{R,C}$-confluent by critical pair inspection. Therefore, $\mathcal{R}$ is $\rightarrow_{R,C}$-confluent (see Theorem 7 below). Yet, $\rightarrow_{R,C}$ cannot *interpret a confluent abstract relation $\Rightarrow$. Indeed, we have rewrites $b + a \rightarrow_{R,C} c + b \rightarrow_{R,C} d + c$ and $b + a \rightarrow_{R,C} d + a \rightarrow_{R,C} c + d$. However, there is* no *term w such that $d + c \rightarrow^*_{R,C} w$ and $c + d \rightarrow^*_{R,C} w$. Actually, both $d + c$ and $c + d$ are R, C- and R/C-irreducible. The killer fact is that $=_C \not\subseteq \rightarrow^*_{R,C}$. Since $\rightarrow_{R,C} \subseteq \rightarrow_{R/C}$ and $=_C \not\subseteq \rightarrow^*_{R/C}$, this also shows that $\rightarrow_{R/C}$ cannot interpret a confluent abstract relation $\Rightarrow$ either.*

So, what is the point? There are three: (i) since by Fact (3) in Remark 2 the relation $\rightarrow^\star_{R/B}$ is transitive *(and obviously reflexive), if $\rightarrow_{R/B}$ is confluent, then the relation $\rightarrow^\star_{R/B}$ can interpret an abstract confluent relation $\Rightarrow$; (ii) if furthermore $\mathcal{R}$ is closed under B-extensions and satisfies conditions (1)–(4), then by Corollary 4 the relation $\rightarrow^\star_{R,B}$ is also* transitive *(and obviously reflexive), so the relation $\rightarrow^\star_{R,B}$ can also interpret an abstract confluent relation $\Rightarrow$; and (iii) if we instead interpret T as the quotient algebra $\mathcal{T}_\Sigma(\mathcal{X})/=_B$ and $\rightarrow_{R/B}$ is confluent, then the induced relation $\rightarrow_{R/B}$ on $\mathcal{T}_\Sigma(\mathcal{X})/=_B$ can indeed interpret an abstract confluent relation $\Rightarrow$ on $\mathcal{T}_\Sigma(\mathcal{X})/=_B$.*

As already mentioned, the Church-Rosser Theorem does *not* hold in general for confluent theories in the conditional case, unless some additional conditions are imposed on $\mathcal{R}$. The problem is illustrated by the following simple example from [72]:

Example 11. *Consider the CTRS $\mathcal{R} = (\Sigma, R)$ with signature Σ having just three constants a, b, c and R having the rules $a \rightarrow c$ and $b \rightarrow c$ if $c \rightarrow a$. $\mathcal{R}$ is confluent.*[18] *For the conditional equational theory (Σ, E) with E having the equations $a = c$ and $b = c$ if $c = a$, we obviously have $E \vdash a = b$. However, b is $\mathcal{R}$-irreducible, and a can only be rewritten to c, so that $a \not\downarrow_R b$. That is, $\mathcal{R}$ fails to have the Church-Rosser property.*

[18] Since here $\rightarrow_R = \rightarrow_{R/\emptyset} = \rightarrow_{R,\emptyset}$, this means confluent in *all* the senses of above Definition 5, which here coincide.

The Church-Rosser property for CTRSs (called there *logicality*) has been carefully studied in [72], where several sufficient conditions on $\mathcal{R}$ ensuring the Church-Rosser Theorem are given. Let me briefly summarize a class of CTRSs enjoying the Church-Rosser property proposed (among other classes) in [72]. Given a CTRS (Σ, R), call a Σ-term t *R-irreducible* iff there is no t' such that $t \rightarrow_R t'$. Likewise, let us call a substitution θ *R-irreducible* iff $x\theta$ is R-irreducible for each $x \in dom(\theta)$. Finally, call a Σ-term t *strongly R-irreducible* iff $t\theta$ is R-irreducible for each R-irreducible θ. Furthermore, we call (Σ, R) *weakly terminating* iff for each Σ-term t there is an R-irreducible t' such that $t \rightarrow_R^* t'$. The class in question is that of all CTRSs $\mathcal{R} = (\Sigma, \vec{E})$ associated to conditional equational theories (Σ, E) such that $\mathcal{R}$ is confluent and weakly terminating, and for each rewrite rule $l \rightarrow r$ *if* $\bigwedge_{i=1..n} u_i \rightarrow v_i$ in $\vec{E}$ all the v_j, $1 \leq j \leq n$ are strongly R-irreducible. They enjoy the Church-Rosser property. That is, we have:

Theorem 4. *[72] If the CTRS $\mathcal{R}$ satisfies the above conditions, then for any Σ-terms t, t' we have the equivalence:*

$$t =_E t' \;\Leftrightarrow\; t \downarrow_{\vec{E}} t'.$$

The above class of CTRSs can be naturally generalized to conditional order-sorted theories. Given a conditional order-sorted rewrite theory $\mathcal{R} = (\Sigma, B, R)$, we call a Σ-term t *R/B-irreducible* iff there is no t' such that $t \rightarrow_{R/B} t'$. Likewise, we call a substitution θ *R/B-irreducible* iff $x\theta$ is R/B-irreducible for each $x \in dom(\theta)$. We call a Σ-term t *strongly R/B-irreducible* iff $t\theta$ is R/B-irreducible for each R/B-irreducible θ. And we call $\mathcal{R} = (\Sigma, B, R)$ *weakly terminating* modulo B iff for each Σ-term t there is an R/B-irreducible t' such that $t \rightarrow_{R/B}^{\star} t'$.

Since we are in an order-sorted setting, a further property, satisfied automatically by unsorted and many-sorted rewrite theories, is also relevant, namely, sort-decreasingness. $\mathcal{R} = (\Sigma, B, R)$ is called *sort-decreasing* modulo B iff whenever $t \rightarrow_{R/B} t'$ we have $ls(t) \geq ls(t')$. A checkable sufficient condition for the sort-decreasingness of $\mathcal{R} = (\Sigma, B, R)$ is that: (i) B is sort-preserving, and (ii) for all rules $l \rightarrow r$ *if* $\bigwedge_{i=1..n} u_i \rightarrow v_i$ in R and all "sort specializations" ρ (i.e., sort-lowering substitutions ρ such that for all $x : s$ in $dom(\rho)$ we have $\rho : x : s \mapsto x' : s'$ with $s \geq s'$) the property $ls(l\rho) \geq ls(r\rho)$ holds. The importance of the requirement that the rules R are sort-decreasing to ensure even confluence and to obtain a good Church-Rosser type of correspondence between $\vec{E}, B$-rewriting and provable $E \cup B$-equality is dramatically underscored —even for unconditional equations with $B = \emptyset$— by the following, simple example.

Example 12. *Let (Σ, E) be the unconditional order-sorted equational theory with Σ having three sorts, A, B, and $Pred$, with $A < B$, constants a of sort A, b of sort B, and tt of sort $Pred$, and a unary "predicate" $p : B \rightarrow Pred$, and where E has two equations: $a = b$, and $p(x) = tt$, with x of sort A.*

Then $\vec{E}$ gives us the rewrite theory $(\Sigma, \emptyset, \{a \rightarrow b, p(x) \rightarrow tt\})$. This theory is clearly terminating, and, furthermore, there are no critical pairs between the

rules in R. So, one would reasonably expect that: (i) the rules R are confluent; and (ii) we have a Church-Rosser kind of equivalence $u =_E v \;\Leftrightarrow\; u \downarrow_{\vec{E}} v$. *But both (i) and (ii) fail miserably. R is* not *confluent, because the term* $p(a)$ *has* two *canonical forms, namely,* tt *and* $p(b)$. *And the Church-Rosser property obviously fails, because we have* $p(b) =_E tt$, *but* $p(b) \not\downarrow_{\vec{E}} tt$.

All these problems go away, so that (i) and (ii) hold and (what is also very important, the check that all critical pairs can be joined ensures local confluence (see [21])) by just reorienting the rule $a \to b$ *as the rule* $b \to a$. *This makes the rules sort-decreasing, terminating, and, again, not having any critical pairs. Therefore, as a consequence of Theorem 7 below, they are confluent and, furthermore, satisfy the Church-Rosser property.*

Here is the main theorem, generalizing Theorem 4 above:

Theorem 5. *(Church-Rosser Theorem modulo B). Let* $\mathcal{R} = (\Sigma, B, \vec{E})$, *associated to a conditional equational theory* $(\Sigma, E \cup B)$, *be such that* $\to_{\vec{E}/B}$ *is sort-decreasing and weakly terminating modulo B, and for each rewrite rule* $l \to r$ *if* $\bigwedge_{i=1..n} u_i \to v_i$ *in* $\vec{E}$ *all the* v_j, $1 \leq j \leq n$, *are strongly* $\vec{E}/B$*-irreducible. Then* $\to_{\vec{E}/B}$ *is confluent iff for any* Σ*-terms* t, t' *we have the equivalence:*

$$(\spadesuit) \quad t =_{E \cup B} t' \;\Leftrightarrow\; t \downarrow_{\vec{E}/B} t'.$$

Proof. Since we have the theory inclusion $(\Sigma, B, \vec{E}) \subseteq (\Sigma, B, \overset{\leftrightarrow}{E})$, Lemma 4 easily gives us the $(\Leftarrow)$ implication in the above equivalence $(\spadesuit)$. Therefore, all we need to prove is that, under the given assumptions, $\mathcal{R}$ is confluent iff the implication $t =_{E \cup B} t' \;\Rightarrow\; t \downarrow_{\vec{E}/B} t'$ holds. Suppose the implication holds. We then need to prove that $u \;{}_{R/B}^{*}\!\leftarrow t \to^*_{R/B} v$ implies $u \downarrow_{R/B} v$. But, again by $(\Sigma, B, \vec{E}) \subseteq (\Sigma, B, \overset{\leftrightarrow}{E})$, this follows easily from Lemma 4, which gives us $u =_{E \cup B} v$.

Suppose now that $\mathcal{R}$ is confluent. We need to prove that $t =_{E \cup B} t' \;\Rightarrow\; t \downarrow_{\vec{E}/B} t'$ holds. Call a set G of Σ-equations *deduction closed* iff $G \vdash u = v$ implies $(u = v) \in G$, where $\vdash$ is the provability relation for the inference system of order-sorted conditional equational logic presented above. It is easy to prove that confluence and the other assumptions on $\mathcal{R}$ make the set of equations

$$=_{\downarrow_{\vec{E}/B}} \;=\; \{u = v \mid u, v \in \mathcal{T}_\Sigma(\mathcal{X}) \;\wedge\; u \downarrow_{\vec{E}/B} v\}$$

deduction-closed. We will be done if we prove $=_{E \cup B} \subseteq =_{\downarrow_{\vec{E}/B}}$.

Recall the notion of free order-sorted Σ/G-algebra $\mathcal{T}_{\Sigma/G}(\mathcal{X})$ on $\mathcal{X}$ for G a set of either conditional or unconditional equations [31, 54]. For each top sort $[s]$ in a connected component of the poset $(S, \leq)$ define $\mathcal{T}_{\Sigma/G}(\mathcal{X})_{[s]} = \mathcal{T}_\Sigma(\mathcal{X})_{[s]}/=_G$, and for any other sort s in the same connected component define $\mathcal{T}_{\Sigma/G}(\mathcal{X})_s = \{[t]_G \in \mathcal{T}_{\Sigma/G}(\mathcal{X})_{[s]} \mid \exists u \in [t]_G \; s.t. \; ls(u) = s\}$. The operations Σ for $\mathcal{T}_{\Sigma/G}(\mathcal{X})$ are defined in the usual way by operating on representatives of G-equivalence classes. We have two such free algebras on $\mathcal{X}$, namely, $\mathcal{T}_{\Sigma/E \cup B}(\mathcal{X})$, associtated to $E \cup B$, and $\mathcal{T}_{\Sigma/=_{\downarrow_{\vec{E}/B}}}(\mathcal{X})$, associated to the deduction-closed equations $=_{\downarrow_{\vec{E}/B}}$.

The desired inclusion $=_{E \cup B} \subseteq =_{\downarrow_{\vec{E}/B}}$ will follow easily from the completeness and freeness theorems for order-sorted algebra (see Theorem 3.1 and Corollary 3.3 in [31]) if we show that $\mathcal{T}_{\Sigma/=_{\downarrow_{\vec{E}/B}}}(\mathcal{X}) \models E \cup B$. The satisfaction $\mathcal{T}_{\Sigma/=_{\downarrow_{\vec{E}/B}}}(\mathcal{X}) \models B$ follows trivially from the **Reflexivity** rule for rewriting modulo B. We just need to show that $\mathcal{T}_{\Sigma/=_{\downarrow_{\vec{E}/B}}}(\mathcal{X}) \models E$. Let $l = r \ \mathit{if} \ \bigwedge_{i=1..n} u_i = v_i$ be a conditional equation in E. We need to show that for each sort-preserving assignment $a : \mathcal{X} \longrightarrow \mathcal{T}_{\Sigma/=_{\downarrow_{\vec{E}/B}}}(\mathcal{X})$ such that $\overline{a}(u_i) = \overline{a}(v_i)$, $1 \leq i \leq n$ we have $\overline{a}(l) = \overline{a}(r)$, where $\overline{a}$ is the homomorphic extension of a. But note that a can always be described as $a(x) = [\theta(x)]_{=_{\downarrow_{\vec{E}/B}}}$ for some (by definition sort-preserving) substitution θ. Therefore, we just need to show that $u_i\theta \downarrow_{\vec{E}/B} v_i\theta$, $1 \leq i \leq n$ implies $l\theta \downarrow_{\vec{E}/B} r\theta$.

Since $\mathcal{R}$ is weakly terminating and sort-decreasing modulo B, we have also an (again sort-preserving thanks to sort-decreasingness) substitution $\theta!_{\vec{E}/B}$ defined for each x by: $\theta!_{\vec{E}/B}(x) = \theta(x)!_{\vec{E}/B}$, where, by definition, $\theta(x)!_{\vec{E}/B}$ is an R/B-irreducible term reachable from $\theta(x)$ thanks to the weak termination modulo B assumption.

Since $u_i\theta \downarrow_{\vec{E}/B} v_i\theta$, $1 \leq i \leq n$ and $\rightarrow_{R/B}$ is confluent, we also have $(u_i\theta!_{\vec{E}/B}) \downarrow_{\vec{E}/B} (v_i\theta!_{\vec{E}/B})$, $1 \leq i \leq n$, which by $\theta!_{\vec{E}/B}$ R/B-irreducible substitution and the v_i strongly R/B-irreducible yields $(u_i\theta!_{\vec{E}/B}) \rightarrow^{\star}_{\vec{E}/B} (v_i\theta!_{\vec{E}/B})$. Therefore, by **Replacement** we get: $(l\theta!_{\vec{E}/B}) \rightarrow_{\vec{E}/B} (r\theta!_{\vec{E}/B})$. But since $(l\theta) \rightarrow^{\star}_{\vec{E}/B} (l\theta!_{\vec{E}/B})$, and $(r\theta) \rightarrow^{\star}_{\vec{E}/B} (r\theta!_{\vec{E}/B})$, we get $l\theta \downarrow_{\vec{E}/B} r\theta$, as desired. □

The class of theories described in Theorem 5 is very general. It includes, in particular, all confluent, sort-decreasing and weakly terminating modulo B rewrite theories that interpret a conditional equation $l = r \ \mathit{if} \ \bigwedge_{i=1..n} u_i = v_i$ as a conditional rewrite rule $l \rightarrow r \ \mathit{if} \ \bigwedge_{i=1..n} u_i \downarrow v_i$. I have not even bothered discussing such theories because, up to a simple transformation, they can be reduced to rewrite theories with oriented conditions. The transformation $\mathcal{R} \mapsto \mathcal{R}^{\equiv}$ in question adds: (i) a new sort *Truth* with a constant tt in a new connected component; (ii) for each top sort $[s]$ of each connected component of the poset of sorts of $\mathcal{R}$ an operator $\_ \equiv \_ : [s]\ [s] \longrightarrow \mathit{Truth}$ and a rewrite rule $x \equiv x \rightarrow tt$, with x of sort $[s]$. Then each rule $l \rightarrow r \ \mathit{if} \ \bigwedge_{i=1..n} u_i \downarrow v_i$ in $\mathcal{R}$ is mapped to the rule $l \rightarrow r \ \mathit{if} \ \bigwedge_{i=1..n} u_i \equiv v_i \rightarrow tt$ in $\mathcal{R}^{\equiv}$. Note that tt is $\mathcal{R}^{\equiv}$-irreducible and obviously strongly $\mathcal{R}^{\equiv}$-irreducible.

6.2. Strongly Deterministic Rewrite Theories and Decidability Issues

I am particularly interested in the use of confluent order-sorted rewrite theories as *functional programs* having an initial algebra semantics as equational theories; and on conditions ensuring that such an initial algebra semantics *agrees* with their operational semantics by rewriting. Furthermore, for any practical applications, the effective implementability of such programs and, when possible, their good decidability properties are paramount.

However, all these desirable properties are not available in general for CTRSs in the class proposed in [72] and shown there to satisfy Theorem 4. A fortiori, they are not available in the even broader class of confluent conditional order-sorted theories satisfying Theorem 5. The reasons why such good properties are lacking include the following:

1. Since no restrictions are given on the variables appearing in the condition and the righthand side of a rule $l \to r$ *if* $\bigwedge_{i=1..n} u_i \to v_i$, guessing which substitution θ to use when applying the **Replacement** rule becomes quite difficult, since in general an infinite number of choices for θ may exist. This makes implementations of rewriting difficult, so that symbolic methods may be needed.

2. A second problem, further discussed below, is that the class of CTRSs in Theorem 4 and, a fortiori, the class of confluent conditional order-sorted theories satisfying Theorem 5 contain all kinds of *monsters*; that is, theories where one's computational intuitions break down (see Example 13 below for a simple example of a monster theory).

3. This leads to a third problem of *decidability*, with two closely-related manifestations. One is that, although mathematically the results in Theorems 4 and 5 ensure the existence of initial algebras, such initial algebras are in general *undecidable* data types lacking a good computational correspondence between mathematical, initial algebra semantics, and operational semantics by rewriting [41, 4, 49]. A second, related manifestation is that, in spite of the equivalence $t =_{E \cup B} t' \;\Leftrightarrow\; t \downarrow_{\vec{E}/B} t'$ ensured by Theorem 5, the equality relation $=_{E \cup B}$ is in general *undecidable*. This nullifies one of the key advantages of the unconditional Church-Rosser property in the weakly terminating case under B-coherence and finitary B-matching algorithm assumptions, namely, the *decidability* of the equality relation $t =_{E \cup B} t'$ by comparing for B-equality the $\vec{E}, B$-irreducible terms $t!_{\vec{E},B}$ and $t'!_{\vec{E},B}$.

The main theme of this section is the study of additional requirements on a rewrite theory $\mathcal{R}$ overcoming the just-mentioned problems (1)–(3). A first step towards overcoming Problem (1) while remaining within the class of confluent theories of Theorem 5 is restricting the rewrite theories $\mathcal{R} = (\Sigma, B, \vec{E})$ to be *strongly deterministic* in the following sense:

Definition 6. *Let $\mathcal{R} = (\Sigma, B, R)$ be a conditional order-sorted rewrite theory. A rule $l \to r$ if $\bigwedge_{i=1..n} u_i \to v_i$ in R is called* deterministic *iff: (i) $\forall j \in [1..n], \mathcal{V}ar(u_j) \subseteq \mathcal{V}ar(l) \cup \bigcup_{k<j} \mathcal{V}ar(v_k)$, and (ii) $\mathcal{V}ar(r) \subseteq \mathcal{V}ar(l) \cup \bigcup_{j \leq n} \mathcal{V}ar(v_j)$. $\mathcal{R}$ is* deterministic *iff all its rules are so.*

A determinsitic rewrite theory $\mathcal{R} = (\Sigma, B, R)$ is called strongly deterministic *if, in addition, each rewrite rule $l \to r$ if $\bigwedge_{i=1..n} u_i \to v_i$ in R is such that the v_i, $1 \leq i \leq n$, are strongly R/B-irreducible.*

Note that when Σ is unsorted and $B = \emptyset$, a deterministic (resp. strongly deterministic) conditional rewrite theory specializes to the notion of a *deterministic* (resp. *strongly deterministic*) *3-CTRS* [60].

The key intuition about both deterministic 3-CTRSs and deterministic order-sorted rewrite theories is that the extra variables in the righthand side and the condition of a rule $l \to r$ *if* $\bigwedge_{i=1..n} u_i \to v_i$ are incrementally instantiated by matching. That is, the conditions are solved *from left to right* by rewriting each $u_i\sigma$ to an instance (modulo B in our case) of the pattern v_i, thus incrementally extending the domain of σ, which can then be used to start solving the next condition by rewriting $u_{i+1}\sigma$.

Specifically, if we assume that B is regular and linear, the deterministic rules R are closed under B-extensions, and there is a finitary B-matching algorithm, rewriting with R modulo B can be bisimulated as R, B-rewriting. The way the substitution θ in the **Replacement** rule of the inference system for $\to_{R,B}$ and $\to^{\star}_{R,B}$ is computed in practice is to choose a position $p \in \mathcal{P}(t)$, and a θ_0 in the finite set $Match_B(l, t_p)$ and then extend θ_0 to θ incrementally by trying to satisfy each of the conditions in the rule from left to right. The crucial point is that we can start the search process to satisfy the first condition with the term $u_1\theta_0$. If that first condition can be satisfied, the instantiation of the extra variables in the pattern v_1 gives us an extended substitution θ_1 with which we can start the search for satisfying the second condition from the term $u_2\theta_1$, and so on. Various examples of deterministic and strongly determinisitc theories can be found in both [60] and, for rewrite theories, in [11]. Note that the requirement of strong determinism is essential for the Church-Rosser property, since the CTRS in Example 11 is confluent, terminating and deterministic, but *not* strongly deterministic.

Strong determinism, however, does not solve Problem (2), that is, the presence of "monsters" theories, where the usual computational intuitions break down. Such monsters lurk also within the class of strongly deterministic theories.

Example 13. *Here is an extremely simple monster, namely, the CTRS $\mathcal{R}$ with constants a, b, c and the single rule $a \to b$ if $a \to c$. Since its rewrite relation is empty, it is trivially confluent and terminating and, furthermore, since c is strongly irreducible, it is strongly deterministic. In particular, this theory belongs to the class described in Theorem 4 above and, a fortiori, to the class of confluent theories described in Theorem 5. What is wrong with this CTRS is that all terms are R-irreducible, yet, an interpreter trying to evaluate a will* loop forever*! Salvador Lucas and I argue in [49] that calling a a* normal form *of this CTRS is a bad joke, because the intuitive idea of a normal form is that it is the* result *of the normalization process; that is, of rewriting a term until no more reductions are possible; but this is precisely what we* cannot *do with a.*

This is just the tip of the iceberg. The broader problem is that there are strongly deterministic "monster" CTRSs $\mathcal{R}$ for which the set $Irr(\mathcal{R})$ of $\mathcal{R}$-irreducible terms is *not* recursively enumerable [41, 4, 68]. So there is no hope,

given an irreducible term t, to *know* that it is irreducible; and therefore no hope in general to know when a computation, that has already reached t, *terminates*. All this means that, in their full generality, the notions of irreducible term and of weakly terminating CTRS are highly problematic, since they violate all the usual intuitions and expectations about both irreducibility and termination. Furthermore, since irreducibility in general is undecidable, all hopes to *decide* equality of two terms by evaluating both to irreducible terms and comparing them for B-equality evaporate.

As argued in [49], the root of these problems is that in the entire literature on CTRSs two *different* notions —which are identical for TRSs but completely different for CTRSs— have been conflated: (i) that of an *R-irreducible* term; and (ii) that of a term in *R-normal form*. Conflating these two notions causes all kinds of *aporias*. In reality, given a CTRS $\mathcal{R}$ we can distinguish *two* sets, one contained in the other: $\mathsf{NF}(\mathcal{R}) \subseteq \mathit{Irr}(\mathcal{R})$, where $\mathsf{NF}(\mathcal{R})$ is the set of *normal forms* of $\mathcal{R}$. That is, every normal form is an irreducible term, but some irreducible terms such as, for example, the constant a in the last example are *not* normal forms. So, what is a normal form?

Definition 7 (Normal form, normal theory, weak normalization). *[49] Given an order-sorted conditional rewrite theory $\mathcal{R} = (\Sigma, B, R)$, a term t is called a* normal form *iff: (i) it is $\mathcal{R}$-irreducible (that is, there is no term u such that $t \rightarrow_{R/B} u$); and (ii) there are no infinite well-formed proof trees whose root has the form $t \rightarrow_{R/B} u$ for u arbitrary. That is, all proof attempts to perform a one-step R/B-rewrite on t fail in finite time.*

Let $\mathsf{NF}(\mathcal{R})$ denote the set of normal forms of $\mathcal{R}$, and $\mathit{Irr}(\mathcal{R})$ the set of irreducible terms of $\mathcal{R}$. $\mathcal{R}$ is called normal[19] *iff the inclusion $\mathsf{NF}(\mathcal{R}) \subseteq \mathit{Irr}(\mathcal{R})$ is an equality, i.e., iff every irreducible term is a normal form. If $\mathcal{R}$ is not normal, we call it* abnormal.

If every term s has a normal form, i.e., $s \rightarrow^{\star}_{R/B} t$ for some normal form t, then $\mathcal{R}$ is called weakly operationally terminating *(or* weakly normalizing*). Note that any weakly normalizing $\mathcal{R}$ is normal.*

All this leads to the notion of a *convergent* (resp. *weakly convergent*) conditional rewrite theory $\mathcal{R}$, which extends to the order-sorted, conditional and modulo cases the good properties of convergent theories in the unsorted and unconditional case.

Definition 8. *A strongly deterministic conditional rewrite theory $\mathcal{R} = (\Sigma, B, R)$ satisfying requirements (1)–(4) at the beginning of Section 4 is called* convergent *(resp.* weakly convergent*) iff R is: (i) sort-decreasing; (ii) closed under*

[19] Note that this meaning of "normal" is in open conflict with the definition of a *normal CTRS* (see, e.g., [60]) as a CTRS whose rewrite rules R are all of the form $l \rightarrow r$ *if* $\bigwedge_{i=1..n} u_i \rightarrow v_i$ with each v_j ground and, not only R-irreducible, but, furthermore, R_u-irreducible, where R_u is the set of unconditional rules obtained from R by dropping all conditions. Since these two meanings of "normal" are so different, no confusion should arise.

B-extensions; (iii) confluent modulo B (in all *senses, which coincide by Lemma 5); and (iv) operationally terminating (resp. weakly normalizing) modulo B. If confluence holds just for ground terms, $\mathcal{R}$ is called* weakly ground convergent*.*

Problems (1)–(3) can be overcome by weakly convergent theories in the following way:

Theorem 6. *(Church Rosser Theorem modulo B with Decidable Equality). Let $\mathcal{R} = (\Sigma, B, \vec{E})$, associated to a conditional equational theory $(\Sigma, E \cup B)$, be strongly deterministic, sort-decreasing, closed under B-extensions, and weakly normalizing modulo B. Then $\mathcal{R}$ is confluent modulo B iff for any Σ-terms t, t' we have the equivalence:*

$$t =_{E \cup B} t' \;\Leftrightarrow\; t \downarrow_{\vec{E}/B} t'.$$

Furthermore, for any weakly convergent, and therefore Church-Rosser modulo B, $\mathcal{R} = (\Sigma, B, \vec{E})$, if E is finite the equality relation $t =_{E \cup B} t'$ is decidable *by checking the B-equality $t!_{\vec{E}/B} =_B t'!_{\vec{E}/B}$.*

Proof. $\mathcal{R} = (\Sigma, B, \vec{E})$ belongs to the class of theories in Theorem 5, so that its confluence modulo B holds iff the above equivalence holds.

Suppose now that $\mathcal{R} = (\Sigma, B, \vec{E})$ is weakly convergent and therefore by the above argument is Church-Rosser modulo B. By the finiteness of E and of B-matches, plus strong determinism (which allows incremental computation of substitutions used in **Replacement** steps), given a term t, its R, B-normal form $t!_{\vec{E},B}$ can be effectively computed up to B-equality by enumerating all rewrite proofs of goals of the form $t \rightarrow^{\star}_{R,B} u$ for any u in increasing order of proof size until encountering a proof of $t \rightarrow^{\star}_{R,B} u$ with u a term for which no further R, B-rewrites are possible, which exists and can be effectively identified by the weak normalization assumption. This, plus the decidability of $=_B$, makes $t!_{\vec{E},B} =_B t'!_{\vec{E},B}$ decidable. But since confluence modulo B gives us the equivalence $t \downarrow_{\vec{E}/B} t' \;\Leftrightarrow\; t!_{\vec{E},B} =_B t'!_{\vec{E},B}$, this makes $t =_{E \cup B} t'$ decidable, as desired. □

Corollary 8. *(Agreement between Mathematical and Operational Semantics). Let $\mathcal{R} = (\Sigma, B, \vec{E})$, associated to a conditional equational theory $(\Sigma, E \cup B)$, be weakly ground convergent with E finite. Define the* canonical term algebra *$\mathcal{C}_{\vec{E},B}$ with $\mathcal{C}_{\vec{E}/B,s} = \{[t!_{\vec{E},B}]_B \mid t \in \mathcal{T}_\Sigma \;\wedge\; ls(t!_{\vec{E},B}) \leq s\}$ for each $s \in S$, and with operations $f : s_1 \ldots, s_n \longrightarrow s$ mapping each $([t_1!_{\vec{E},B}]_B, \ldots, [t_n!_{\vec{E},B}]_B)$ to the B-equivalence class $[f(t_1!_{\vec{E},B}, \ldots, t_n!_{\vec{E},B})!_{\vec{E},B}]_B$. Then $\mathcal{C}_{\vec{E},B}$ is a computable algebra*[20] *and we have an isomorphism of algebras:*

$$\mathcal{T}_{\Sigma/E \cup B} \;\cong\; \mathcal{C}_{\vec{E},B}.$$

[20] That is, an algebra where both the operations and the equality predicate are computable functions.

This corollary manifests the full agreement between the initial algebra semantics furnished by $\mathcal{T}_{\Sigma/E\cup B}$ and the operational semantics by reduction to canonical form furnished by $\mathcal{C}_{\vec{E},B}$. Note that if we give a term t for evaluation to an interpreter, the result that should be returned by the interpreter[21] is precisely $t!_{\vec{E},B}$. Therefore, the canonical term algebra $\mathcal{C}_{\vec{E},B}$ is precisely (up to B-equality) the algebra of *values* obtained by normalization modulo B and therefore a perfect algebraic summary of the operational semantics of the functional program $\mathcal{R} = (\Sigma, B, \vec{E})$.

Weakly convergent rewrite theories are a very general class of functional programs with good computational and decidability properties, including the above agreement between their mathematical and operational semantics. But how can we *check* that a conditional rewrite theory is convergent or weakly convergent? Closure under B-extensions is easy to check by the methods presented in this paper. As already mentioned, easily checkable sufficient conditions for sort-decreasingness (and tools [21]) also exist. Proof methods and tools to show operational termination of order-sorted rewrite theories have been developed in, e.g., [18, 20, 19, 49]; and sufficient conditions for a theory being normal have been studied in [49]. Checking of confluence modulo regular and linear axioms B with a finitary unification algorithm under the sort-decreasingness, operational termination and closure under B-extensions assumptions, and a tool supporting such checking for various combinations of associativity, commutativity and identity axioms have been documented in [21].

In the terminology of the present work, and taking into account that the operational termination of a sort-decreasing, deterministic rewrite theory $\mathcal{R} = (\Sigma, B, R)$ is equivalent to its being quasi-decreasing modulo B [49], we can rephrase Theorem 2 in [21] as follows:

Theorem 7. *[21] Let $\mathcal{R} = (\Sigma, B, R)$ be sort-decreasing, strongly deterministic, closed under B-extensions, and operationally terminating. Then $\mathcal{R}$ is confluent (and therefore convergent) iff all its conditional critical pairs are joinable.*

By definition, the conditional critical pairs of $\mathcal{R}$ are the implications of the form:

$$C\,\sigma \wedge C'\sigma \Rightarrow (l[r']_p)\sigma = r\sigma$$

obtained from (possibly renamed) conditional rewrite rules $l \to r \ \textit{if}\ C$ and $l' \to r' \ \textit{if}\ C'$ in R such that $\mathcal{V}ar(l \to r \ \textit{if}\ C) \cap \mathcal{V}ar(l' \to r' \ \textit{if}\ C') = \emptyset$ and $l|_p\sigma =_B l'\sigma$, for some nonvariable position $p \in \mathcal{P}(l)$ and B-unifier $\sigma \in \mathit{Unif}_B(l|_p, l')$. Such a conditional critical pair is called *joinable* iff for each substitution μ such that $(C\,\sigma \wedge C'\sigma)\mu$ holds in $\mathcal{R}$ we have $(l[r']_p)\sigma\mu \downarrow_{R,B} r\sigma\mu$.

[21] For a convergent rewrite theory this is exactly the case. For a weakly convergent rewrite theory various evaluation strategies —including the inefficient one sketched out in the proof of Theorem 6— are possible, depending on $\mathcal{R}$. As explained in [49], several such strategies are supported by Maude under various assumptions on $\mathcal{R}$.

Theorem 7 above generalizes to the order-sorted and modulo cases a similar method for checking confluence of quasi-reductive, strongly deterministic CTRSs in [1].

Example 14. *(Decidability of the Exclusive Or and the Idempotent Semigroup Theories). As a direct corollary of Theorem 7 above, we obtain the decidability of the equational theories of exclusive or and of idempotent semigroups by the R, B-rewriting methods developed in this paper.*

First of all, note that all rules in the closure under B-extensions $\overline{\mathcal{R}}_{\oplus}$ of the exclusive or rewrite theory in Example 7 are size-decreasing, and the AC axioms in that theory preserve the size of terms. Therefore, $\overline{\mathcal{R}}_{\oplus}$ is terminating and, being unconditional, also operationally terminating. It is also easy to check that all critical pairs in $\overline{\mathcal{R}}_{\oplus}$ are joinable. This can be automatically checked using various tools for proving local termination of (possibly conditional) rewrite theories modulo commonly used axioms B such as, for example, Maude's Church-Rosser Checker [21].

The operational termination of the closure under B-extensions $\overline{\mathcal{R}}_{Idemp.Sgr}$ of the conditional rewrite theory of idempotent semigroups has already been proved in Example 10. We just need to show that its conditional critical pairs are joinable. This is harder, because in this case the axioms B include the associativity without commutativity of the semigroup operation (juxtaposition), and associative unification is in general infinitary. Fortunately, this check of conditional critical pairs under associativity has already been done by Siekmann and Szabo in [65]. More precisely, what they actually prove is the joinability of the conditional critical pairs under the $\rightarrow_{R/B}$ relation, which implies that of the $\rightarrow_{R,B}$ relation because of strict coherence. This is achieved by a careful analysis of all the overlaps that are possible for such rules. The local confluence for the remaining auxiliary rules in $\overline{\mathcal{R}}_{Idemp.Sgr}$, which are all unconditional, can be checked by computing critical pairs (B-unification becomes finitary for auxiliary symbols) and then checking the joinability of each of those pairs.

In summary, therefore, we obtain that both these theories are not only Church-Rosser, but also decidable *by R, B-rewriting to canonical form.*

7. Related Work and Conclusions

The most obviously related work are the various studies on unconditional equational rewriting, e.g., [36, 45, 47, 46, 61, 37, 38, 3, 39, 59, 43, 26]. For conditional rewriting modulo axioms, earlier work includes, e.g., [27, 44, 53, 6, 9, 18, 21]. Only the last three papers [9, 18, 21] considered rules with extra variables in their conditions and righthand sides. The paper by Bockmayr [6] considered conditional rewriting modulo B with the R, B-relation, but without studying B-extensions, and only under the assumptions of no extra variables in a rule's condition and of the simplifying termination [60] of R modulo B. In this work, conditional rewriting modulo axioms B has been considered in its fullest generality, namely, for order-sorted conditional rewrite theories with no restrictions whatsoever on either B or the rule's variables. However, due

to the problematic nature of non-regular or non-linear axioms B, closure under B-extensions has been studied only for regular and linear axioms B. The fact that the conditional rewrite rules can have extra variables in their conditions and righthand sides makes coherence issues more subtle, since our being able to rewrite a term u to a term v with a given rule satisfying a certain condition with a certain substitution leaves unclear how, for another term $u' =_B u$, we can find a second rule and a second substitution satisfying that second rule's condition and rewriting u' to v' with $v' =_B v$. To the best of my knowledge this paper provides the first systematic study of coherence issues for rewriting modulo axioms with conditional rules with no restrictions whatsoever about their variables. B-extensions and conditional coherence could certainly be generalized to more general sets of axioms B; but, for the reasons already given in the paper and the additional reasons given below in the discussion of strong coherence, I see no compelling practical reasons to embark in such a generalization.

As already mentioned, in the unsorted and unconditional case, equi-termination of R/B- and R, B-rewriting under what here is called the **Completeness** property, was shown in [26]. This result has been here broadly generalized to the operational equi-termination of R/B- and R, B-rewriting in the order-sorted and conditional case, assuming closure under B-extensions. This of course ensures **Completeness** and all other equivalent strict coherence properties. The study of such equivalent notions of strict coherence was initiated in [20, 21].

A non-trivial question is what to do to rewrite modulo axioms B when the equational axioms B fall outside the case of regular and linear equations and could even be conditional. As already mentioned, a generalization of the R, B-rewriting relation in the style of, e.g., [37, 38, 3] is possible (at least for B unconditional), but brings with it considerable technical difficulties and limitations: for example, bisimulation and equi-termination are no longer possible. However, a different alternative exists, namely, *allowing two rewrite relations*. Within an equational logic setting, an early proposal in this direction was made by Marché [50]. In the broader, not necessarily equational setting of rewriting logic, the strong coherence ideas initiated in [70] and substantially generalized in [21] have matured to a point where there is now ample evidence through many examples, language implementations, and tools such as the Maude Coherence Checker described in [21], supporting the claim that allowing two rewrite relations provides a much more flexible method of achieving, in an effectively computable way, the effect of rewriting *modulo* a very broad class of equational axioms, including conditional ones.

The general idea is to decompose a rewrite theory $\mathcal{R}$ as $\mathcal{R} = (\Sigma, E \cup B, R)$, where: (i) B are regular and linear equations, but the equations E can be conditional; (ii) $(\Sigma, B, \vec{E})$ is a convergent conditional rewrite theory exactly in the sense of Definition 8; and (iii) R are not necessarily equational and possibly conditional rewrite rules closed under B-extensions whose conditions are given an equational meaning and are solved by rewriting with $\vec{E}$ modulo B. There are, therefore, two rewrite relations, namely, a convergent equational one, $\rightarrow_{\vec{E},B}$, and a not necessarily equational one $\rightarrow_{R,B}$, which uses $\rightarrow_{\vec{E},B}$ as an auxiliary relation

to evaluate its conditions. The right property ensuring the effect of rewriting with R modulo $E \cup B$ is the *strong coherence* of the rules R with the equational rules $\vec{E}$ modulo B [70, 21]. Furthermore, the conditions under which strong coherence can be achieved can be substantially relaxed for initial models where the equations E are sufficiently complete with respect to a subsignature $\Omega \subseteq \Sigma$ of constructor symbols [21]. The work presented here is actually directly relevant for strong coherence in the conditional case, since completeness of the relations $\rightarrow_{\vec{E},B}$ and $\rightarrow_{R,B}$ requires both $\vec{E}$ and R to be closed under B-extensions.

In conclusion, this work has developed the foundations of conditional rewriting modulo axioms under very general assumptions about the type structure and the kinds of conditional rules allowed. This generality is not a caprice: it is needed and used in actual applications to rule-based languages and in formal specification and reasoning tools. But such generality should not obscure the obvious fact that, even in the unconditional case, new concepts and results are provided: the notion of strict coherence, and the specialization of all the subsequent results in the paper to the unconditional case afford a considerably simpler conceptual setting for rewriting modulo axioms in the (for all purposes most practical) case of regular and linear axioms B, than that provided by more general but considerably more complex approaches such as [37, 38, 3]. Issues such as operational equi-termination, the Church-Rosser Theorem, and executability and decidability have also been studied in detail.

Acknowledgements. Francisco Durán and Salvador Lucas were involved in the joint early development of the strict coherence ideas in [20, 21]. I thank Nachum Dershowitz, Santiago Escobar, Jean-Pierre Jouannaud, Hélène Kirchner, Salvador Lucas, and the anonimous referees, for their comments and suggestions, which have helped me prepare a substantially improved version of the paper. This work has been partially supported by NSF Grant CNS 13-19109.

References

[1] J. Avenhaus and C. Loría-Sáenz. On conditional rewrite systems with extra variables and deterministic logic programs. In F. Pfenning, editor, *Logic Programming and Automated Reasoning, 5th International Conference, LPAR 1994, Proceedings*, volume 822 of *Springer LNCS*, pages 215–229, 1994.

[2] F. Baader and T. Nipkow. *Term Rewriting and All That.* Cambridge University Press, 1998.

[3] L. Bachmair and N. Dershowitz. Completion for rewriting modulo a congruence. *Theor. Comput. Sci.*, 67(2&3):173–201, 1989.

[4] J. A. Bergstra and J. W. Klop. Conditional rewrite rules: Confluence and termination. *J. Comput. Syst. Sci.*, 32(3):323–362, 1986.

[5] M. Bidoit and P. D. Mosses. *CASL User Manual - Introduction to Using the*

Common Algebraic Specification Language, volume 2900 of *Lecture Notes in Computer Science*. Springer, 2004.

[6] A. Bockmayr. Conditional narrowing modulo of set of equations. *Appl. Algebra Eng. Commun. Comput.*, 4:147–168, 1993.

[7] P. Borovanský, C. Kirchner, H. Kirchner, and P.-E. Moreau. ELAN from a rewriting logic point of view. *Theoretical Computer Science*, 285:155–185, 2002.

[8] A. Bouhoula, J.-P. Jouannaud, and J. Meseguer. Specification and proof in membership equational logic. *Theoretical Computer Science*, 236:35–132, 2000.

[9] R. Bruni and J. Meseguer. Semantic foundations for generalized rewrite theories. *Theor. Comput. Sci.*, 360(1-3):386–414, 2006.

[10] H.-J. Bürckert. *A Resolution Principle for a Logic with Restricted Quantifiers*, volume 568 of *Lecture Notes in Computer Science*. Springer, 1991.

[11] M. Clavel, F. Durán, S. Eker, J. Meseguer, P. Lincoln, N. Martí-Oliet, and C. Talcott. *All About Maude – A High-Performance Logical Framework*. Springer LNCS Vol. 4350, 2007.

[12] A. G. Cohn. A more expressive formulation of many sorted logic. *J. Autom. Reasoning*, 3(2):113–200, 1987.

[13] A. G. Cohn. Taxonomic reasoning with many-sorted logics. *Artif. Intell. Rev.*, 3(2-3):89–128, 1989.

[14] H. Comon. Completion of rewrite systems with membership constraints. part I: deduction rules. *J. Symb. Comput.*, 25(4):397–419, 1998.

[15] H. Comon. Completion of rewrite systems with membership constraints. part II: constraint solving. *J. Symb. Comput.*, 25(4):421–453, 1998.

[16] N. Dershowitz and J.-P. Jouannaud. Rewrite systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume Formal Models and Sematics (B), pages 244–320. Elsevier, 1990.

[17] N. Dershowitz and D. Plaisted. Rewriting. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, volume I, chapter 9, pages 535–610. Elsevier, 2001.

[18] F. Durán, S. Lucas, C. Marché, J. Meseguer, and X. Urbain. Proving operational termination of membership equational programs. *Higher-Order and Symbolic Computation*, 21(1-2):59–88, 2008.

[19] F. Durán, S. Lucas, and J. Meseguer. Methods for proving termination of rewriting-based programming languages by transformation. *Electr. Notes Theor. Comput. Sci.*, 248:93–113, 2009.

[20] F. Durán, S. Lucas, and J. Meseguer. Termination modulo combinations of equational theories. In *Frontiers of Combining Systems, 7th International Symposium, FroCoS 2009, Trento, Italy, September 16-18, 2009. Proceedings*, volume 5749 of *Lecture Notes in Computer Science*, pages 246–262. Springer, 2009.

[21] F. Durán and J. Meseguer. On the Church-Rosser and coherence properties of conditional order-sorted rewrite theories. *J. Algebraic and Logic Programming*, 81:816–850, 2012.

[22] S. Eker. Associative-commutative rewriting on large terms. In R. Nieuwenhuis, editor, *RTA*, volume 2706 of *Lecture Notes in Computer Science*, pages 14–29. Springer, 2003.

[23] S. Escobar, R. Sasse, and J. Meseguer. Folding variant narrowing and optimal variant termination. *J. Algebraic and Logic Programming*, 81:898–928, 2012.

[24] A. M. Frisch. The substitutional framework for sorted deduction: Fundamental results on hybrid reasoning. *Artif. Intell.*, 49(1-3):161–198, 1991.

[25] K. Futatsugi and R. Diaconescu. *CafeOBJ Report*. World Scientific, 1998.

[26] J. Giesl and D. Kapur. Dependency pairs for equational rewriting. In *RTA 2001*, volume 2051 of *Lecture Notes in Computer Science*, pages 93–108. Springer, 2001.

[27] J. Goguen, J.-P. Jouannaud, and J. Meseguer. Operational semantics of order-sorted algebra. In *Proc. ICALP 1985*, volume 194, pages 221–231. Springer LNCS, 1985.

[28] J. Goguen and J. Meseguer. Equality, types, modules and (why not?) generics for logic programming. *Journal of Logic Programming*, 1(2):179–210, 1984.

[29] J. Goguen and J. Meseguer. Completeness of many-sorted equational logic. *Houston Journal of Mathematics*, 11(3):307–334, 1985. Preliminary version in: *SIGPLAN Notices*, July 1981, Volume 16, Number 7, pages 24-37.

[30] J. Goguen and J. Meseguer. Unifying functional, object-oriented and relational programming with logical semantics. In B. Shriver and P. Wegner, editors, *Research Directions in Object-Oriented Programming*, pages 417–477. MIT Press, 1987.

[31] J. Goguen and J. Meseguer. Order-sorted algebra I: Equational deduction for multiple inheritance, overloading, exceptions and partial operations. *Theoretical Computer Science*, 105:217–273, 1992.

[32] J. Goguen, T. Winkler, J. Meseguer, K. Futatsugi, and J.-P. Jouannaud. Introducing OBJ. In *Software Engineering with OBJ: Algebraic Specification in Action*, pages 3–167. Kluwer, 2000.

[33] A. E. Haxthausen. Order-sorted algebraic specifications with higher-order functions. *Theor. Comput. Sci.*, 183(2):157–185, 1997.

[34] J. Hendrix and J. Meseguer. Order-sorted equational unification revisited. *Electr. Notes Theor. Comput. Sci.*, 290:37–50, 2012.

[35] W. Hodges. *A Shorter Model Theory*. Cambridge UP, 1997.

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

[37] J.-P. Jouannaud. Confluent and coherent equational term rewriting systems: Application to proofs in abstract data types. In G. Ausiello and M. Protasi, editors, *CAAP*, volume 159 of *Lecture Notes in Computer Science*, pages 269–283. Springer, 1983.

[38] J.-P. Jouannaud and H. Kirchner. Completion of a set of rules modulo a set of equations. *SIAM Journal of Computing*, 15:1155–1194, November 1986.

[39] J.-P. Jouannaud and J. Li. Church-Rosser properties of normal rewriting. In P. Cégielski and A. Durand, editors, *CSL*, volume 16 of *LIPIcs*, pages 350–365. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2012.

[40] J.-P. Jouannaud and M. Muñoz. Termination of a set of rules modulo a set of equations. In *CADE 1984*, volume 170 of *Lecture Notes in Computer Science*, pages 175–193. Springer, 1984.

[41] S. Kaplan. Conditional rewrite rules. *Theoretical Computer Science*, 33:175–193, 1984.

[42] C. Kirchner. Order-sorted equational unification. Technical Report 954, INRIA Lorraine & LORIA, Nancy, France (Dec. 1988).

[43] C. Kirchner and H. Kirchner. Rewriting, Solving, Proving. Unpublished book draft available online, 2006.

[44] C. Kirchner, H. Kirchner, and J. Meseguer. Operational semantics of OBJ3. In *Proc. ICALP 1988*, volume 317, pages 287–301. Springer LNCS, 1988.

[45] D. Lankford and A. Ballantyne. Decision procedures for simple equational theories with a commutative axiom: complete sets of commutative reductions. Technical Report ATP-35, Department of Mathematics and Computer Science, Univ. of Texas, Austin, 1977.

[46] D. Lankford and A. Ballantyne. Decision procedures for simple equational theories with commutative-associative axioms: complete sets of commutative-associative reductions. Technical Report ATP-39, Department of Mathematics and Computer Science, Univ. of Texas, Austin, 1977.

[47] D. Lankford and A. Ballantyne. Decision procedures for simple equational theories with permutative axioms: complete sets of permutative reductions. Technical Report ATP-37, Department of Mathematics and Computer Science, Univ. of Texas, Austin, 1977.

[48] S. Lucas, C. Marché, and J. Meseguer. Operational termination of conditional term rewriting systems. *Information Processing Letters*, 95(4):446–453, 2005.

[49] S. Lucas and J. Meseguer. Normal forms and normal theories in conditional rewriting. *J. Log. Algebr. Meth. Program.*, 85(1):67–97, 2016.

[50] C. Marché. Normalised rewriting and normalised completion. In *Proc. LICS'94*, pages 394–403. IEEE, 1994.

[51] N. Martí-Oliet and J. Meseguer. Inclusions and subtypes II: Higher-order case. *J. Logic and Computation*, 6:541–572, 1996.

[52] J. Meseguer. General logics. In H.-D. E. et al., editor, *Logic Colloquium'87*, pages 275–329. North-Holland, 1989.

[53] J. Meseguer. Conditional rewriting logic as a unified model of concurrency. *Theoretical Computer Science*, 96(1):73–155, 1992.

[54] J. Meseguer. Membership algebra as a logical framework for equational specification. In *Proc. WADT'97*, pages 18–61. Springer LNCS 1376, 1998.

[55] J. Meseguer. Twenty years of rewriting logic. *J. Algebraic and Logic Programming*, 81:721–781, 2012.

[56] J. Meseguer and J. Goguen. Initiality, induction and computability. In M. Nivat and J. Reynolds, editors, *Algebraic Methods in Semantics*, pages 459–541. Cambridge University Press, 1985.

[57] J. Meseguer, J. Goguen, and G. Smolka. Order-sorted unification. *J. Symbolic Computation*, 8:383–413, 1989.

[58] P. D. Mosses. *CASL Reference Manual, The Complete Documentation of the Common Algebraic Specification Language*, volume 2960 of *Lecture Notes in Computer Science*. Springer, 2004.

[59] P. Narendran, M. Subramanian, and Q. Guo. Observations on equational rewriting. Unpublished manuscript, ca. 1994.

[60] E. Ohlebusch. *Advanced Topics in Term Rewriting*. Springer Verlag, 2002.

[61] G. E. Peterson and M. E. Stickel. Complete sets of reductions for some equational theories. *Journal of the Association Computing Machinery*, 28(2):233–264, 1981.

[62] G. D. Plotkin. A structural approach to operational semantics. *Journal of Logic and Algebraic Programming*, 60-61:17–139, 2004. Previously published as technical report DAIMI FN-19, Computer Science Department, Aarhus University, 1981.

[63] C. Rocha, J. Meseguer, and C. Muñoz. Rewriting Modulo SMT and Open System Analysis. In *Proc. WRLA 2014*, volume 8663, pages 247–262. Springer LNCS, 2014.

[64] M. Schmidt-Schauss. *Computational aspects of order-sorted logic with term declarations*. Springer LNCS 395, 1989.

[65] J. Siekmann and P. Szabó. A Noetherian and confluent rewrite system for idempotent semigroups. *Semigroup Forum*, 25:83–110, 1982.

[66] G. Smolka and H. Aït-Kaci. Inheritance hierarchies: Semantics and unification. *J. Symb. Comput.*, 7(3/4):343–370, 1989.

[67] G. Smolka, W. Nutt, J. Goguen, and J. Meseguer. Order-sorted equational computation. In M. Nivat and H. Aït-Kaci, editors, *Resolution of Equations in Algebraic Structures*, volume 2, pages 297–367. Academic Press, 1989.

[68] TeReSe. *Term Rewriting Systems*. Cambridge University Press, 2003.

[69] A. van Deursen, J. Heering, and P. Klint. *Language Prototyping: An Algebraic Specification Approach*. World Scientific, 1996.

[70] P. Viry. Equational rules for rewriting logic. *Theoretical Computer Science*, 285:487–517, 2002.

[71] C. Walther. A mechanical solution of Schubert's steamroller by many-sorted resolution. *Artif. Intell.*, 26(2):217–224, 1985.

[72] T. Yamada, J. Avenhaus, C. Loría-Sáenz, and A. Middeldorp. Logicality of conditional rewrite systems. *Theor. Comput. Sci.*, 236(1-2):209–232, 2000.