Ground Confluence of Order-sorted Conditional Specifications Modulo Axioms

Francisco Durán

Universidad de Málaga, Málaga, Spain

José Meseguer

University of Illinois, Urbana-Champaign, IL, USA

Camilo Rocha

Pontificia Universidad Javeriana, Cali, Colombia

Abstract

Terminating functional programs should be *deterministic*, i.e., should evaluate to a *unique* result, regardless of the evaluation order. For equational functional programs such determinism is exactly captured by the *ground confluence* property. For operationally terminating conditional equations this is equivalent to *ground local confluence*, which follows from *local confluence*. Checking local confluence by computing critical pairs is the *standard* way to check ground confluence. The problem is that some perfectly reasonable equational programs are *not* locally confluent and it can be very hard or even impossible to make them so by adding more equations. We propose three methods, called Methods 1–3, that can be synergistically combined to prove an order-sorted conditional specifications modulo axioms B ground locally confluent. Method 1 applies the strategy proposed in [14] to use non-joinable critical pairs as *completion hints* to either achieve local confluence or reduce the number of critical pairs. Method 2 uses the *inductive joinability inference system* proposed in this paper to try to prove the critical pairs remaining after applying Method 1 ground joinable. It can furthermore show ground local confluence of the original specification. Method 3 is *hierarchical* in nature: it can be used to prove the ground local confluence of a *conditional* equational specification whose conditions belong to a *subspecification* that has already been proved ground confluent and operationally terminating, and that is conservatively extended by the overall specification in an appropriate sense. These methods apply to order-sorted and possibly conditional equational programs modulo axioms such as, e.g., Maude functional modules. We show their effectiveness in proving the ground confluence of non-trivial examples that have eluded previous proof attempts.

Keywords: equational programs, ground confluence, order-sorted specifications, rewriting modulo axioms, inductive joinability proof methods, Maude.

1. Introduction

Functional programs should be *deterministic*; that is, if they terminate for a given input, they should return a *unique* value, regardless of the evaluation order. *Ground confluence* is the precise characterization of such determinism for functional equational programs associated to equational theories of the form $\mathcal{E} = (\Sigma, E \uplus B)$, were B are structural axioms and E are, possibly conditional, equations that are executed as rewrite rules $\overrightarrow{E}$ *modulo* B. Therefore, for execution purposes, all the relevant information is contained in the rewrite theory $\mathcal{R}_{\mathcal{E}} = (\Sigma, B, \overrightarrow{E})$. Since ground confluence is essential both for correct execution and for almost any form of formal verification about properties of $\mathcal{E}$ and $\mathcal{R}_{\mathcal{E}}$, methods to prove ground confluence are very important.

The *standard method* to do so for a (possibly conditional) terminating equational program $\mathcal{R}_{\mathcal{E}} = (\Sigma, B, \overrightarrow{E})$ is to: (i) prove that it is *operationally terminating* (and if Σ is order-sorted, also *sort decreasing*); and then (ii) since operational termination plus local confluence imply confluence, prove the *stronger* property that $\mathcal{R}_{\mathcal{E}}$ is locally confluent (modulo B). This tends to work well in many cases, but not always. The thorny issue addressed in this paper is what to do when this standard method does not work. We describe in what follows three such methods. The first goes back to [14] and is not new. However it is a good starting point for the other two new methods.

1.1. Method 1: Incremental Joining of Critical Pairs

In [14], the wild goose chase for a convergent specification by attempting a Knuth-Bendix completion of $\mathcal{E}$ was explicitly discouraged, since it can often lead to an infinite loop and, even if it were to eventually succeed, can result in a highly bloated and hard to understand specification. Instead, the following *incremental strategy* in the spirit of Knuth-Bendix was suggested: since failure of a proof of local confluence will generate a set of unjoinable *critical pairs* characterizing the most general cases in which rules cannot be shown confluent, such critical pairs can be used as very useful *hints* for a user to try to either: (i) orient a critical pair as a rule and add it to the specification; or (ii) if the critical pair has the form $C[u] = C[v]$ with C a common context, orient instead $u = v$ and add it to the specification; or (iii) *generalize* $u = v$ in cases (i) and (ii) into a more general $u' = v'$ that has $u = v$ as a substitution instance and add an oriented version of $u' = v'$ to the specification. In this way, we obtain a new specification $\mathcal{R}_{\mathcal{E}'} = (\Sigma, B, \overrightarrow{E} \uplus \overrightarrow{G})$, where $\overrightarrow{G}$ are the new oriented equations added by methods (i)–(iii). If $\mathcal{R}_{\mathcal{E}'}$ is locally confluent, operationally terminating, and sort-decreasing, we are done; otherwise, we can *iterate* the process with the critical pairs obtained for $\mathcal{R}_{\mathcal{E}'}$.

In practice, this incremental strategy works reasonably well, but not always. Furthermore, it raises the following unsolved questions:

1. Have we *changed* the initial algebra semantics? That is, do the original $\mathcal{R}_{\mathcal{E}}$ and its extension $\mathcal{R}_{\mathcal{E}'}$ have the same initial algebra when viewed as equational theories? If only additions of type (i) are made, this is always true; but additions of type (ii)–(iii) are often needed in practice.

2. Was the *original* specification $\mathcal{R}_{\mathcal{E}}$ already ground confluent? That is, can we use $\mathcal{R}_{\mathcal{E}'}$ as the proverbial "Wittgenstein's ladder" that we can kick away *after* we have proved its local confluence?

3. What do we do if we run into a *wall*? For instance, the "wall" of having an equation $u = v$ obtained by methods (i)–(iii) above that *cannot* be oriented because it would lead to non-termination.

1.2. Method 2: Inductive Proof of Ground Joinability of Critical Pairs

Method 2, the first new method proposed in this paper, answers the above three unsolved questions, for which Method 1 provides no support. It can achieve proofs of ground confluence for highly non-trivial theories for which it is impossible to achieve a Knuth-Bendix completion with a finite set of critical pairs. Furthermore, although also transformational in nature, Method 2 can *show* that the original specification was already ground confluent. In a nutshell, the more general and powerful strategy of Method 2 is based on three steps:

1. Use the above-described strategy of Method 1 from [14] as far as it can go.

2. If one hits the wall of non-orientability for some critical pairs (Question 3), or otherwise Method 1 seems to have reached its practical limits without achieving joinability, prove the *ground joinability* of the remaining critical pairs —which are hopefully a smaller set than the original set of critical pairs— by the *inductive joinability inference system*[1] presented in this work.

3. To ensure preservation of the initial algebra semantics (Question 1) and the ground confluence of the original specification (Question 2), we can use the same inductive methods to prove ground joinability of all the equations added along the first step. Of course, one could skip the first step altogether and merge the second and third steps into one; but this may require a considerably bigger effort, since the whole point of taking the first step is to *greatly reduce* the number of pairs to be proved ground joinable. Furthermore, the user may have made an *actual mistake* in the original specification $\mathcal{R}_{\mathcal{E}}$, so that the second and third steps become meaningless. In such a case, the first step can be quite helpful in identifying such mistakes and help the user *restart* the process with a new specification.

The example we use to illustrate the use and effectiveness of Method 2 is worth discussing briefly. It is a specification `HF-SETS` of *hereditarily finite sets*, i.e., sets whose elements are themselves finite sets, whose elements . . ., and so on. It is well-known that hereditarily finite sets provide a model of set theory minus the axiom of infinity, and that all of finitary mathematics, including arithmetic, can be carried out within `HF-SETS` [7]. Therefore, proving the ground confluence of `HF-SETS` (which is also terminating) amounts to giving a functional program whose initial model is a computable model of set theory minus the axiom of infinity. We show that, indeed, the

[1] We specify the inference system for *unconditional* specifications, whose critical pairs are likewise unconditional. However, our inductive joinability inference system can be extended to the conditional case, so that conditional critical pairs can likewise be proved ground joinable. In fact, later in the paper, we use one of the inference rules in a conditional setting.

roadblock envisioned in Question 3 (a critical pair that is intrinsically unorientable, yet it is ground joinable) is met in this example. Furthermore, Method 2 shows that the HF-SETS specification as originally given is ground joinable.

1.3. Method 3: Hierarchical Proof of Ground Joinability of Conditional Critical Pairs

Conditional equational specifications are intrinsically more complex than unconditional ones. For joinability this extra complexity clearly shows up: given a terminating unconditional specification it is *decidable* whether it is confluent, since this holds if and only if all critical pairs are joinable. Instead, for operationally terminating conditional specifications it is in general *undecidable* whether they are confluent. We may, for example, have a conditional critical pair whose condition is unsatisfiable and therefore causes no confluence problems; but proving such unsatisfiability may be undecidable. In practical terms this means that for proving ground confluence in the conditional case, as Charlie Brown would put it, we need all the help we can get.

The first line of help is that, as pointed out in Footnote 1, Method 2 does naturally extend to the conditional case, so we can use it to inductively prove ground joinability of conditional critical pairs. But there is more. In this paper we present an additional new method, Method 3, that is based on a *hierarchical* idea: the given, operationally terminating specification, say, $(\Sigma, E \cup B)$, where the conditional equations E are used as oriented rewrite rules modulo axioms B, may have a subspecification $(\Sigma_0, E_0 \cup B_0)$ that has already been shown to be ground convergent and, furthermore: (i) the conditions in all equations in $E \setminus E_0$ are Σ_0-conditions and remain so after applying any substitutions; and (ii) $(\Sigma, E \cup B)$ conservatively extends $(\Sigma_0, E_0 \cup B_0)$ in the rewriting sense of not introducing any new rewrites among Σ_0-terms. Then we *can* bootstrap ourselves quite well for proving that $(\Sigma, E \cup B)$ is ground convergent. To begin with, we can use the fact that $(\Sigma_0, E_0 \cup B_0)$ is convergent and apply the Church-Rosser Theorem to reason about the satisfiability/unsatisfiability of conditions in conditional critical pairs at the inductive equational logic level, that is, in the initial algebra $T_{\Sigma_0/E_0 \cup B_0}$, where inductive theorem proving techniques —plus satisfiability decision procedures such as, for example, variant satisfiability [27]— can greatly help. Furthermore, we can combine: (a) inductive theorem proving in $T_{\Sigma_0/E_0 \cup B_0}$, (b) variable abstraction to abstract the maximal Σ_0-subterms in the Σ-terms u and v of a conditional critical pair, say, $D \Rightarrow u \downarrow v$, and (c) rewriting techniques in a theory extending $(\Sigma, E \cup B)$ with the abstraction variables as constants and with (oriented) inductive lemmas proved under the assumption D understood equationally, to prove that $D \Rightarrow u \downarrow v$ is ground joinable.

In practice, the opportunities for applying Method 3 to order-sorted conditional specifications are quite common, because specifications are developed in a modular way and also because it is quite common that the conditions in conditional equations only involve a subset of functions —for example some predicates— that often belong to already-defined submodules. The chicken-and-egg conundrum that Method 3 solves is that, *unless* we have already proved a specification ground convergent we *cannot* use the Church-Rosser Theorem and inductive equational reasoning techniques to prove ground joinability of conditional critical pairs. This conundrum is solved not for $(\Sigma, E \cup B)$ but for the convergent subspecification $(\Sigma_0, E_0 \cup B_0)$ thanks to the hierarchical nature of Method 3. But if Method 3 succeeds, it is then also solved for $(\Sigma, E \cup B)$, so that if $(\Sigma, E \cup B)$ is itself later extended, say, by $(\Sigma, E \cup B) \subset (\Sigma', E' \cup B')$, then Method 3 could again be applied to prove $(\Sigma', E' \cup B')$ ground convergent.

We illustrate the use and effectiveness of Method 3 by solving a research challenge that has been open since the 1980's, namely, proving the ground confluence of an order-sorted conditional specification of the rational numbers. In this case, $(\Sigma, E \cup B)$ is the specification of the rationals, and $(\Sigma_0, E_0 \cup B_0)$ the subspecification of the integers.

Paper organization. Preliminaries are gathered in Section 2. Method 1 is illustrated in Section 3 by a hereditarily finite sets specification that does indeed run into a non-orientability wall. Method 2's inductive joinability inference system for ground confluence is presented and proved sound in Section 4, and is illustrated by proving the inductive joinability of the non-orientable critical pair from Section 3, thus illustrating the Second Step. The effectiveness of Method 2 is further illustrated for the `HF-SETS` example in Section 5. Method 3, its foundations, and its application to proving an order-sorted specification of the rational numbers ground confluent are presented in Section 6. Some related work and conclusions are discussed in Section 7. Various auxiliary results needed for the mechanized proofs and for the proofs of ground convergence of `HF-SETS` example and of the rational numbers specification are presented in the Appendices.

Comparison with the Conference Paper [15]. In comparison with the conference paper [15], which this paper substantially extends, the following are new contributions:

1. The appendices containing all auxiliary proofs for the mechanized inductive proofs of Method 2 and for the `HF-SETS` example are new.
2. Section 6 on Method 3 is entirely new, and so are the various appendices needed for the proofs of the rational numbers example.
3. The specification of the rational numbers example is also entirely new. Previous such specifications extended the Peano natural numbers. Instead, the current specification uses natural and integer addition with an associative-commutative + operator with 0 as unit for the entire number hierarchy, where +, 0 and 1 are the constructors for the naturals. It has several important advantages, including the existence of quite large subspecifications `NAT-FVP` and `INT-FVP` that have the finite variant property [8, 18], and where satisfiability of quantifier-free formulas in their initial algebras is decidable [27].
4. The Abstract and the Introduction and Related Work and Conclusions sections have been substantially extended.

To make the main ideas of the paper more easily accessible, some technical details and lengthy proofs have been made available in a technical report in [**?**].

2. Preliminaries

Notation on terms, term algebras, and equational theories is used as, e.g., in [21]. An *order-sorted signature* Σ is a tuple $\Sigma = (S, \leq, F)$ with a finite poset of sorts $(S, \leq)$

and set of function symbols F typed with sorts in S. The binary relation $\equiv_{\leq}$ denotes the equivalence relation $(\leq \cup \geq)^{+}$ generated by $\leq$ on S and its point-wise extension to strings in S^{*}. The function symbols in F can be subsort-overloaded. For any sort $s \in S$, the expression $[s]$ denotes the connected component of s, that is, $[s] = [s]_{\equiv_{\leq}}$. A *top sort* in Σ is a sort $s \in S$ such that for all $s' \in [s]$, $s' \leq s$. Let $X = \{X_s\}_{s\in S}$ be an S-indexed family of disjoint variable sets with each X_s countably infinite. For any $s \in S$, let $X_{\leq s} = \bigcup_{s' \in S \wedge s' \leq s} X_{s'}$. The *set of terms of sort s* and the *set of ground terms of sort s* are denoted, respectively, by $T_{\Sigma}(X)_s$ and $T_{\Sigma,s}$; similarly, $T_{\Sigma}(X)$ and T_{Σ} denote, respectively, the set of terms and the set of ground terms. $\mathcal{T}_{\Sigma}(X)$ and $\mathcal{T}_{\Sigma}$ denote the corresponding order-sorted Σ-term algebras. All order-sorted signatures are assumed *preregular* [21], i.e., each Σ-term t has a unique *least sort* $ls(t) \in S$ s.t. $t \in T_{\Sigma}(X)_{ls(t)}$. It is also assumed that Σ has *nonempty sorts*, i.e., $T_{\Sigma,s} \neq \emptyset$ for each $s \in S$. The *set of variables* of t is written $vars(t)$ and for a list of terms $t_1, \ldots, t_n$, $vars(t_1, \ldots, t_n) = vars(t_1) \cup \cdots \cup vars(t_n)$.

A *substitution* is an S-indexed mapping $\theta \in [X \longrightarrow T_{\Sigma}(X)]$ that is different from the identity only for a finite subset of X and such that $\theta(x) \in T_{\Sigma}(X)_s$ if $x \in X_s$, for any $x \in X$ and $s \in S$. The expression $\theta|_Y$ denotes the restriction of θ to a family of variables $Y \subseteq X$. The *domain* of θ, denoted $dom(\theta)$, is the subfamily of X such that $x \in dom(\theta)$ iff $\theta(x) \neq x$, for each $x \in X$. If $dom(\theta) = \{x_1, \ldots, x_n\}$ we write $\theta = \{x_1 \mapsto \theta(x_1), \ldots, x_n \mapsto \theta(x_n)\}$. The *range* of θ is the set $ran(\theta) = \bigcup\{vars(\theta(x)) \mid x \in dom(\theta)\}$. Substitutions extend homomorphically to terms in the natural way. A substitution θ is called *ground* iff $ran(\theta) = \emptyset$. The application of a substitution θ to a term t is denoted by $t\theta$ and the composition (in diagrammatic order) of two substitutions θ_1 and θ_2 is denoted by $\theta_1\theta_2$, so that $t\theta_1\theta_2$ denotes $(t\theta_1)\theta_2$. A *context* C is a λ-term of the form $C = \lambda x_1, \ldots, x_n.c$ with $c \in T_{\Sigma}(X)$ and $\{x_1, \ldots, x_n\} \subseteq vars(c)$; it can be viewed as an n-ary function $(t_1, \ldots, t_n) \mapsto C(t_1, \ldots, t_n) = c\theta$, where $\theta(x_i) = t_i$ for $1 \leq i \leq n$ and $\theta(x) = x$ for $x \notin \{x_1, \ldots, x_n\}$.

An *equational theory* is a tuple (Σ, E), with Σ an order-sorted signature and E a finite collection of (possibly conditional) Σ-equations. An equational theory $\mathcal{E} = (\Sigma, E)$ induces the congruence relation $=_{\mathcal{E}}$ on $T_{\Sigma}(X)$ defined for $t, u \in T_{\Sigma}(X)$ by $t =_{\mathcal{E}} u$ iff $\mathcal{E} \vdash t = u$, where $\mathcal{E} \vdash t = u$ denotes $\mathcal{E}$-provability by the deduction rules for order-sorted equational logic in [25]. For the purpose of this paper, such inference rules, which are analogous to those of many-sorted equational logic, are even simpler thanks to the assumption that Σ has nonempty sorts, which makes unnecessary the explicit treatment of universal quantifiers. The expressions $\mathcal{T}_{\mathcal{E}}(X)$ and $\mathcal{T}_{\mathcal{E}}$ (also written $\mathcal{T}_{\Sigma/E}(X)$ and $\mathcal{T}_{\Sigma/E}$) denote the quotient algebras induced by $=_{\mathcal{E}}$ on the term algebras $\mathcal{T}_{\Sigma}(X)$ and $\mathcal{T}_{\Sigma}$, respectively. $\mathcal{T}_{\Sigma/E}$ is called the *initial algebra* of (Σ, E).

We assume acquaintance with the usual notions of position p in a term t, subterm $t|_p$ at position p, and term replacement $t[u]_p$ at position p (see, e.g., [9]). A *rewrite theory* is a tuple $\mathcal{R} = (\Sigma, E, R)$ with (Σ, E) an order-sorted equational theory and R a finite set of possibly conditional Σ-rules, with conditions being a conjunction of Σ-equalities. A rewrite theory $\mathcal{R}$ induces a rewrite relation $\rightarrow_{\mathcal{R}}$ on $T_{\Sigma}(X)$ defined for every $t, u \in T_{\Sigma}(X)$ by $t \rightarrow_{\mathcal{R}} u$ iff there is a rule $(l \rightarrow r \textit{ if } \phi) \in R$, a term t', a position p in t', and a substitution $\theta : X \longrightarrow T_{\Sigma}(X)$ satisfying $t =_E t' = t'[l\theta]_p$, $u =_E t'[r\theta]_p$, and $(\Sigma, E) \vdash \phi\theta$. The tuple $\mathcal{T}_{\mathcal{R}} = (\mathcal{T}_{\Sigma/E}, \rightarrow^{\star}_{\mathcal{R}})$, where, by definition, $\rightarrow^{\star}_{\mathcal{R}} = \rightarrow^{+}_{\mathcal{R}} \cup =_E$, where $\rightarrow^{+}_{\mathcal{R}}$ denotes the transitive closure of $\rightarrow_{\mathcal{R}}$, is called the *initial reachability model of* $\mathcal{R}$ [4].

In this paper we will mostly focus on rewrite theories of the form $\mathcal{R}_{\mathcal{E}} = (\Sigma, B, \overrightarrow{E})$ associated to an equational theory $\mathcal{E} = (\Sigma, E \uplus B)$, were: (i) B are decidable structural axioms whose equations $u = v \in B$ are linear (no repeated variables in either u or v) and regular (same variables in u and v), for which a matching algorithm exists, and (ii) the possibly conditional rewrite rules $\overrightarrow{E}$ are *strictly B-coherent* [26]. Under such assumptions, the rewrite relation $t \to_{\mathcal{R}_{\mathcal{E}}} u$ holds iff there exists u' such that $u' =_B u$, and $t \to_{\overrightarrow{E},B} u'$, where, by definition, $t \to_{\overrightarrow{E},B} u'$ iff there exists a rule $(l \to r \ if \ \phi) \in \overrightarrow{E}$, a position p in t and a substitution θ such that $t|_p =_B l\theta$, $u' = t[r\theta]_p$, and $\mathcal{R}_{\mathcal{E}} \vdash \phi\theta$. We will assume throughout that the rules $\overrightarrow{E}$ are *always strictly* B-coherent. We finally assume that the axioms B are: (i) *sort-preserving*, i.e., for each $(u = v) \in B$ and substitution σ we have $ls(u\sigma) = ls(v\sigma)$; and (ii) *term-size preserving*[2], i.e., if $t =_B t'$, then $|t| = |t'|$.

Additional requirements are needed to make an equational theory $\mathcal{E} = (\Sigma, E \uplus B)$ *admissible* as an equational program, i.e., for making $\mathcal{R}_{\mathcal{E}} = (\Sigma, B, \overrightarrow{E})$ *executable* in languages such as Maude [6]. In this paper, besides the above assumptions about B and $\overrightarrow{E}$, we assume that the rules in $\overrightarrow{E}$ are sort-decreasing, operationally terminating, and ground confluent modulo B. The rewrite rules $\overrightarrow{E}$ are *sort decreasing* modulo B iff for each $(t \to u \ if \ \gamma) \in \overrightarrow{E}$ and substitution θ, $ls(t\theta) \geq ls(u\theta)$ if $\mathcal{R}_{\mathcal{E}} \vdash \gamma\theta$. Instead, the rules $\overrightarrow{E}$, assuming they are weakly terminating, are called *ground sort-decreasing* iff for each ground term t there exists a term t' such that $t \to^{!}_{\overrightarrow{E},B} t'$ (where $\to^{!}_{\overrightarrow{E},B}$ denotes rewriting to $\overrightarrow{E}, B$-normal form) such that $ls(t) \geq ls(t')$. $\mathcal{R}_{\mathcal{E}} = (\Sigma, B, \overrightarrow{E})$ is *operationally terminating* modulo B [10] iff there is no infinite well-formed proof tree in $(\Sigma, B, \overrightarrow{E})$. Call $t, t' \in T_{\Sigma}(X)$ *joinable* in $\mathcal{R}_{\mathcal{E}}$, denoted $t \downarrow_{\mathcal{R}_{\mathcal{E}}} t'$ iff there exist u, v such that $t \to^{*}_{\overrightarrow{E},B} u$, $t' \to^{*}_{\overrightarrow{E},B} v$, and $u =_B v$. Call $\mathcal{R}_{\mathcal{E}} = (\Sigma, B, \overrightarrow{E})$ *confluent* (resp., *ground confluent*) modulo B iff for all $t, t_1, t_2 \in T_{\Sigma}(X)$ (resp., for all $t, t_1, t_2 \in T_{\Sigma}$), if $t \to^{*}_{\overrightarrow{E},B} t_1$ and $t \to^{*}_{\overrightarrow{E},B} t_2$, then $t_1 \downarrow_{\mathcal{R}_{\mathcal{E}}} t_2$. For $\mathcal{R}_{\mathcal{E}} = (\Sigma, B, \overrightarrow{E})$ to have good executability properties as a terminating equational program, the following requirements are needed : (a) ground sort decreasingness, (b) operational termination, (c) ground confluence, and (d) *strong determinism*, defined as follows:

Definition 1. *Given a rewrite theory $\mathcal{R} = (\Sigma, B, R)$, a rule $l \to r$ if $\bigwedge_{i=1..n} u_i \to v_i$ in R is said to be* deterministic *iff (i) $\forall j \in [1..n]$, $vars(u_j) \subseteq vars(l) \cup \bigcup_{k<j} vars(v_k)$, and (ii) $vars(r) \subseteq vars(l) \cup \bigcup_{j \leq n} vars(v_j)$. $\mathcal{R}$ is* deterministic *iff all its rules are so. A term t is called* strongly irreducible *with respect to R modulo B (or* strongly R, B-irreducible*) iff $t\sigma$ is an R, B-normal form for every normalized substitution σ. A deterministic rewrite theory $\mathcal{R}$ is called* strongly deterministic *iff for every rule $l \to r$ if $\bigwedge_{i=1..n} u_i \to v_i$ in R each v_i is strongly R, A-irreducible.*

[2] For combinations of associativity, commutativity, and identity axioms, this last condition only rules out identity axioms. However, both for termination and confluence analysis purposes, identity axioms can always be turned into convergent rewrite rules modulo associativity and/or commutativity axioms, as explained in [12].

Note that conditions (a)–(d) essentially corresponds to the notion of an *admissible* Maude functional module in the sense of [6, Section 4.6]. That is, an admissible conditional order-sorted Maude functional specification can be transformed into an equivalent strongly deterministic rewrite theory by a very simple procedure, in which equations are oriented as rewrite rules and equational conditions (ordinary ones and so called matching equations) are transformed into rewrite conditions (see [13] for a detailed algorithm). If conditions (a)–(d) are met, we call $\mathcal{R}_{\mathcal{E}}$ *ground convergent*. $\mathcal{R}_{\mathcal{E}}$ is called *convergent* if it satisfies the stronger requirements of sort-decreasingness, operational termination, and ground confluence.

For a detailed inference system describing the operational semantics of convergent and ground convergent rewrite theories, as well as the fact that convergent rewrite theories satisfy the *Church-Rosser property*, that is, the equivalence: $\mathcal{E} \vdash t = t'$ iff $t \downarrow_{\mathcal{R}_{\mathcal{E}}} t'$, see [24]. Note also that, for $\mathcal{R}_{\mathcal{E}}$ ground convergent, Theorem 3 in [24] can be easily adapted to yield a *ground Church-Rosser* equivalence $\mathcal{E} \vdash t = t'$ iff $t \downarrow_{\mathcal{R}_{\mathcal{E}}} t'$, where now t, t' are ground terms.

Given a condition $C = u_1 \to v_1 \wedge \ldots \wedge u_n \to v_n$, with $Z = vars(C)$, and a substitution $\theta \in [Z \to T_{\Sigma}(X)]$, we write $(\Sigma, B, \overrightarrow{E}) \models C\theta$ iff for each i, $1 \leq i \leq n$, there exists a w_i such that $u_i\theta \longrightarrow^{*}_{\overrightarrow{E},B} w_i \wedge w_i =_B v_i\theta$. Note that in a strongly deterministic rewrite theory a rewrite step with a rule $l \to r$ *if* C and $\overrightarrow{E}, B$-normalized substitution θ can take place if and only if $(\Sigma, B, \overrightarrow{E}) \models C\theta$.

3. Method 1: An Equational Specification for Hereditarily Finite Sets

When checking the confluence of an equational specification, the CRC tool [17, 14] provides as result a set of critical pairs that cannot be joined automatically by its built-in heuristics. They are proof obligations that can either be proved joinable or used as guidance for modifying the input specification. The methodology proposed in [14] for using the CRC tool suggests that critical pairs can help in identifying theorems of the original specification which, when added to it, may lead to a confluent or ground confluent specification. However, as the example of HF-SETS presented in this section shows, the analysis of critical pairs to modify a specification, though a useful first strategy, may be insufficient to make the specification ground confluent. Other techniques, such as the ones presented in Section 4, may be needed.

Consider the specification of hereditarily finite sets in Figure 1, namely, of finite sets whose elements are all hereditarily finite sets (see, e.g., [23]). The recursive definition of well-founded hereditary sets has the empty set as the base case and if $s_1, \ldots, s_k$ are hereditarily finite, then so is $\{s_1, \ldots, s_k\}$. These sets play a key role in axiomatic set theory because they are a model of all the axioms of set theory except for the axiom of infinity. Furthermore, as the methods developed in this work will show, the initial model of the HF-SETS specification below is a *consistent* model of set theory without the axiom of infinity.

The Church-Rosser check of the HF-SETS module using the CRC tool says that the specification is sort-decreasing, but it cannot show that it is locally confluent, returning eight critical pairs as proof obligations. At this point, there are two alternatives: either

```
fmod HF-SETS is
  protecting BOOL-OPS .
  sorts Magma Set .
  subsort Set < Magma .
  op _,_ : Magma Magma -> Magma [ctor assoc comm] .
  op {_} : Magma -> Set [ctor] .
  op {} : -> Set [ctor] .

  vars M M’ : Magma .
  vars S S’ : Set .

  eq [01]: M, M = M .

  op _in_ : Magma Set -> Bool .                    *** set membership
  eq [11]: S in {S} = true .
  eq [12]: S in {} = false .
  eq [13]: {} in {{M}} = false .
  eq [14]: {M} in {{}} = false .
  eq [15]: {M} in {{M’}} = M in {M’} and M’ in {M} .
  eq [16]: S in {S’, M} = S in {S’} or S in {M} .
  eq [17]: (S, M) in S’ = (S in S’) and (M in S’) .

  op _~_ : Set Set -> Bool .                       *** set equality
  eq [21]: S ~ S’ = (S <= S’) and (S’ <= S) .

  op _<=_ : Set Set -> Bool .                      *** set containment
  eq [31]: {} <= S = true .
  eq [32]: {M} <= S = M in S .

  op _U_ : Set Set -> Set [assoc comm] .           *** union
  eq [41]: S U {} = S .
  eq [42]: {M} U {M’} = {M, M’} .
  eq [43]: S U {M} U {M’} = S U {M, M’} .

  op P : Set -> Set .                             *** powerset
  eq [51]: P({}) = {{}} .
  eq [52]: P({S}) = {{},{S}} .
  eq [53]: P({S, M}) = P({M}) U augment(P({M}), S) .

  op augment : Set Set -> Set .                   *** augmentation
  eq [61]: augment({}, S) = {} .
  eq [62]: augment({S}, S’) = {{S’} U S} .
  eq [63]: augment({M, M’}, S) = augment({M}, S) U augment({M’}, S) .

  op _&_ : Set Set -> Set .                       *** intersection
  eq [71]: {} & S = {} .
 ceq [72]: {S} & S’ = {S} if S in S’ = true .
 ceq [73]: {S} & S’ = {} if S in S’ = false .
 ceq [74]: {S, M} & S’ = {S} U ({M} & S’) if S in S’ = true .
 ceq [75]: {S, M} & S’ = {M} & S’ if S in S’ = false .

  op <_;_> : Set Set -> Set .                     *** ordered pairs
  op <_;_> : Magma Magma -> Magma .               *** extension to magmas
  eq [91]: < S ; S’ > = {S, {S, S’}} .
  eq [92]: < S ; S’, M > = {S, {S, S’}}, < S ; M > .
  eq [93]: < S, M ; M’ > = < S ; M’ >, < M ; M’ > .

  op _X_ : Set Set -> Set .                       *** cartesian product
  eq [101]: {} X S = {} .
  eq [102]: S X {} = {} .
  eq [103]: {M} X {M’} = {< M ; M’ >} .
endfm
```

Figure 1: Equational specification of hereditarily finite sets in Maude

(i) we try to prove the ground joinability of these critical pairs to conclude that the specification is locally ground confluent, or (ii) we follow the iterative strategy proposed in [14] to get a locally confluent specification or at least reduce the number of critical pairs for which a proof of joinability is necessary. In the rest of this section, we explore the second alternative. The first alternative will be revisited after the second one is exhausted (both are useful) in Section 5.

The following one is one of the critical pairs returned by the check:

```
cp HF-SETS1123 for 11 and 15
  true = M':Magma in {M':Magma} .
```

It comes from the overlap of equations 11 and 15. Although there are equations for all possible instances of the term `M in {M}`, Maude cannot reduce it as magmas. We can attempt adding equations to reduce it as follows:

```
fmod HF-SETS-0 is
  protecting HF-SETS .
  vars M M' : Magma .
  eq [18]: M in {M} = true .
  eq [19]: M in {M', M} = true .
endfm
```

A check of the Church-Rosser property for HF-SETS-0 returns seven critical pairs. Let us consider one of these critical pairs:

```
cp HF-SETS-095 for 01 and 63
  augment({M':Magma}, S:Set) = augment({M':Magma}, S:Set) U augment({M':Magma}, S:Set) .
```

This critical pair comes from the overlap of equations 01 and 63. Indeed, this critical pair cannot be further reduced because there is no idempotency equation for the union operator on sets. We can see the same problem in other four of the critical pairs reported by the tool. Although S U S = S could be proven in HF-SETS-0, there is the alternative option of extending the specification with an idempotency equation for set union.

```
fmod HF-SETS-1 is
  protecting HF-SETS-0 .
  var S : Set .
  eq [44]: S U S = S .
endfm
```

The Church-Rosser checker tool produces the following output for HF-SETS-1:

```
The following critical pairs must be proved joinable:
  cp HF-SETS-118 for 53 and 53
     P({#6:Magma}) U augment(P({#6:Magma}), S:Set) U augment(P({#6:Magma}) U
     augment(P({#6:Magma}), S:Set), #1:Set)
   = P({#6:Magma}) U augment(P({#6:Magma}), #1:Set) U augment(P({#6:Magma}) U
     augment(P({#6:Magma}), #1:Set), S:Set).
  cp HF-SETS-1355 for 01 and 53
     P({#3:Magma}) U augment(P({#3:Magma}), S:Set)
   = P({#3:Magma}) U augment(P({#3:Magma}), S:Set) U augment(P({#3:Magma}) U
     augment(P({#3:Magma}), S:Set), S:Set).
  The module is sort-decreasing.
```

A careful study of these critical pairs suggests the need for an equation to apply augment over the union operator.

```
fmod HF-SETS-2 is
  protecting HF-SETS-1 .
  vars S S' T : Set .
  eq [64]: augment(S U S', T) = augment(S, T) U augment(S', T) .
endfm
```

The number of critical pairs gets further decreased in HF-SETS-2, but two remain:

```
The following critical pairs must be proved joinable:
  cp HF-SETS-218 for 53 and 53
     P({#6:Magma}) U augment(P({#6:Magma}), S:Set) U augment(P({#6:Magma}), #1:Set) U
     augment(augment(P({#6:Magma}), S:Set), #1:Set)
   = P({#6:Magma}) U augment(P({#6:Magma}), S:Set) U augment(P({#6:Magma}), #1:Set) U
     augment(augment(P({#6:Magma}), #1:Set), S:Set).
  cp HF-SETS-2411 for 01 and 53
     P({#3:Magma}) U augment(P({#3:Magma}), S:Set)
   = P({#3:Magma}) U augment(P({#3:Magma}), S:Set)
                   U augment(augment(P({#3:Magma}), S:Set), S:Set).
  The module is sort-decreasing.
```

The second critical pair suggests the need for an equation handling the repeated application of the augment operator.

```
fmod HF-SETS-3 is
  protecting HF-SETS-2 .
  vars S T : Set .
  eq [65]: augment(augment(S, T), T) = augment(S, T) .
endfm
```

However, one critical pair remains in HF-SETS-3:

```
Church-Rosser check for HF-SETS-3
The following critical pairs must be proved joinable:
  cp HF-SETS-318 for 53 and 53
     P({#6:Magma})U augment(P({#6:Magma}),S:Set)U augment(P({#6:Magma}),#1:Set)U
     augment(augment(P({#6:Magma}),S:Set),#1:Set)
   = P({#6:Magma})U augment(P({#6:Magma}),S:Set)U augment(P({#6:Magma}),#1:Set)U
     augment(augment(P({#6:Magma}),#1:Set),S:Set).
  The module is sort-decreasing.
```

It is not obvious at all how to eliminate this critical pair, since adding the equation

```
 eq augment(augment(S, S'), T) = augment(augment(S, T), S') .
```

would make the specification *non-terminating*. This suggests that the second approach, i.e., the strategy of trying to complete the specification by analyzing the unjoinable critical pairs has now been exhausted. However, the original problem has now been reduced to a *single* critical pair. At this point, the best approach is to prove the *inductive joinability* of the critical pair HF-SETS-318 obtained in the check of HF-SETS-3, and thus conclude that the specification is ground locally confluent. Section 4 presents techniques for carrying out such inductive proofs. Indeed, it will also present results showing that the *original specification was already ground confluent!*, without the need for the extra equations added in the process. The specification is terminating. Indeed, the MTT tool [11, 17] is able to find termination proofs for all the versions of the HF-SETS module, and specifically for HF-SETS-3 (see [**?** , Appendix B]). A proof of the sufficient completeness of the specification can be found in [**?** , Appendix C].

Finally, note that if an added equation comes from orienting a critical pair, it is a logical consequence of the specification and therefore the new specification has the *same* initial model of the old one. Although the additional equations added during the process may not be those obtained from critical pairs as such, proving that they are ground joinable is enough to show that they are actually *inductive lemmas*, and therefore —as explained in more detail in Theorem 6 in Section 4— that they both preserve the initial algebra semantics *and* can be *removed* from the original specification.

4. Method 2: Proving Ground Joinability

This section presents inductive techniques for proving ground joinability for rewrite theories associated to equational specifications. These techniques are presented as meta-theorems about the ground reachability relation induced by a rewrite theory and are used to justify the inference system also presented in this section.

Definition 2. *Let $\mathcal{R}$ be a rewrite theory with signature $\Sigma = (S, \leq, F)$ and $t, u \in T_\Sigma(X)_s$ for some $s \in S$. The terms t and u are called:*

1. $\mathcal{R}$-joinable*, written $\mathcal{R} \vdash (\forall X)\, t \downarrow u$, iff there is $v \in T_\Sigma(X)_s$ such that $\mathcal{R} \vdash (\forall X)\, t \rightarrow^\star v$ and $\mathcal{R} \vdash (\forall X)\, u \rightarrow^\star v$.*

2. ground $\mathcal{R}$-joinable*, written $\mathcal{R} \Vdash (\forall X)\, t \downarrow u$, iff $\mathcal{R} \vdash t\theta \downarrow u\theta$ for all ground substitutions $\theta \in [X \longrightarrow T_\Sigma]$.*

The authors of [30] investigate constructor-based inductive techniques for proving ground joinability. They distinguish two notions of constructors for a rewrite theory $\mathcal{R}$, namely, one for the equations and another one for the rules in $\mathcal{R}$.

Definition 3 (Defs. 5 and 6 [30])**.** *Let $\mathcal{R} = (\Sigma, E, R)$ be a rewrite theory with underlying equational theory $\mathcal{E} = (\Sigma, E)$. A* constructor signature pair for $\mathcal{R}$ *is a pair (Υ, Ω) of order-sorted subsignatures $\Upsilon = (S, \leq, F_\Upsilon) \subseteq \Omega = (S, \leq, F_\Omega)$. The sets of terms $T_\Upsilon = \{T_{\Upsilon,s}\}_{s \in S}$ and $T_\Omega = \{T_{\Omega,s}\}_{s \in S}$ are called, respectively,* E-constructor terms *and* $\mathcal{R}$-constructor terms*. The rewrite theory $\mathcal{R}$ is called:*

1. E-sufficiently complete relative to Ω *iff $(\forall s \in S)(\forall t \in T_{\Sigma,s})(\exists u \in T_{\Omega,s})\; \mathcal{E} \vdash t = u$.*

2. $\mathcal{R}$-sufficiently complete relative to Υ *iff $(\forall s \in S)(\forall t \in T_{\Sigma,s})(\exists v \in T_{\Upsilon,s})\; \mathcal{R} \vdash t \rightarrow^\star v$.*

3. sufficiently complete relative to (Υ, Ω) *iff (1) and (2) hold.*

The notion of sufficient completeness for a rewrite theory $\mathcal{R}$ relative to a constructor signature pair (Υ, Ω) is that $\Omega \subseteq \Sigma$ are the constructors for the equations and $\Upsilon \subseteq \Omega$ the constructors for the rules, thus including the standard concept of constructor for equational specifications as a special case. The intuition behind equational constructor terms is that any ground Σ-term should be *provably equal* to a term in T_Ω and for rewrite constructors that any Σ-term should be *rewritable* to a term in T_Υ.

It is sufficient to consider all $\mathcal{R}$-constructor terms in $T_{\Upsilon,s}$ when inducting on a variable x of sort s, for a proof on inductive joinability in $\mathcal{R}$ to be sound.

Theorem 1 (Thm. 6 [30])**.** *Let $\mathcal{R}$ be a rewrite theory with signature $\Sigma = (S, \leq, F)$ and $t, u \in T_\Sigma(X)_s$ for some $s \in S$. If $\mathcal{R}$ is sufficiently complete relative to the constructor signature pair (Υ, Ω), then $\mathcal{R} \Vdash (\forall X)\, t \downarrow u$ iff $(\forall \eta \in [X \longrightarrow T_\Upsilon])\; \mathcal{R} \vdash t\eta \downarrow u\eta$.*

$$\frac{\mathcal{R} \vdash (\forall X)\, t \downarrow u}{\mathcal{R} \Vdash (\forall X)\, t \downarrow u}\ \textsc{Join} \qquad \frac{\mathcal{R} \Vdash (\forall X)\, t \downarrow u}{\mathcal{R} \Vdash (\forall X)\, C[t] \downarrow C[u]}\ \textsc{Ctx} \qquad \frac{\mathcal{R} \Vdash (\forall X)\, t \downarrow u}{\mathcal{R} \Vdash (\forall X)\, t\theta \downarrow u\theta}\ \textsc{Gral}$$

Figure 2: Inference rules for proving joinability for a rewrite theory $\mathcal{R}$ by rewrite-based reasoning, and inductive reasoning for contexts and substitution instances.

Figure 2 presents the Join, Ctx and Gral inference rules for proving joinability for a rewrite theory $\mathcal{R}$, respectively, by rewrite-based reasoning, inductive reasoning under contexts, and generalization. The soundness of the Join rule is straightforward to obtain, while Theorem 2 justifies the soundness of the Ctx and Gral rules. This result can be used to simplify the complexity of terms to be joinable if they share a common context.

Theorem 2. *Let $\mathcal{R}$ be a rewrite theory with signature $\Sigma = (S, \leq, F)$ and $C[t], C[u] \in T_\Sigma(X)_s$ for some $s \in S$. If $\mathcal{R} \Vdash (\forall X)\, t \downarrow u$, then:*

1. *$\mathcal{R} \Vdash (\forall X)\, C[t] \downarrow C[u]$;*
2. *$\mathcal{R} \Vdash (\forall X)\, t\theta \downarrow u\theta$, for any substitution $\theta \in [X \longrightarrow T_\Sigma(X)]$.*

Proof. The two properties follow from the fact that the rewrite relation $\rightarrow_{\mathcal{R}}$ is closed under contexts and substitutions. □

Since the goal is to prove ground joinability of a rewrite theory of the form $\mathcal{R}_\mathcal{E} = (\Sigma, B, \overrightarrow{E})$ associated to an equational theory $\mathcal{E} = (\Sigma, E \uplus B)$, such as that for hereditarily finite sets presented in Section 3, the most appropriate notion of constructor is that of $\mathcal{R}_\mathcal{E}$-constructors. More precisely, a constructor signature pair for $\mathcal{R}_\mathcal{E}$ has always the form (Υ, Σ) because the only equations in $\mathcal{R}_\mathcal{E}$ are the axioms B not associated to any rewriting. Hence, $\mathcal{R}_\mathcal{E}$ sufficient completeness is *always* relative only to Υ. One more remark is important for what follows. As pointed out in Section 2, we assume that $\mathcal{R}_\mathcal{E} = (\Sigma, B, \overrightarrow{E})$ is admissible (except for its ground confluence, which may remain to be proved). In particular this means that $\mathcal{R}_\mathcal{E}$ is strictly B-coherent in the sense of [26]. Therefore, the two notions of joinability (resp. ground joinability) involved, namely the one in Def. 2, and that defined in terms of the rewrite relation $\rightarrow_{\overrightarrow{E},B}$ in Section 2 actually coincide (see [26]). We will implicitly use this agreement between both notions in what follows.

Reasoning about ground joinability requires inductive inference support, e.g., in the form of a constructor-based scheme using finite generating sets.

Definition 4. *Let $\mathcal{E} = (\Sigma, E \uplus B)$ be an equational theory, with $\Sigma = (S, \leq, F)$, such that the rewrite theory $\mathcal{R}_\mathcal{E}$ is weakly terminating, ground sort-decreasing, and has subsignature Υ of $\mathcal{R}_\mathcal{E}$-constructors. Further, let $s \in S$. A set $G_s \subseteq T_{\Upsilon,s}(X)$ is a* (finite) generating set for s modulo B *iff G_s is finite, $G_s \cap X = \emptyset$, and*

$$T_{\Upsilon/B,s} = \bigcup_{w \in G_s} \{[w\sigma]_B \mid \sigma \in [vars(w) \longrightarrow T_\Upsilon]\}.$$

The following induction scheme is sound for inferring ground joinability in $\mathcal{R}_\mathcal{E}$.

Theorem 3. *Let $\mathcal{R}_{\mathcal{E}}$ be a weakly terminating and ground sort-decreasing rewrite theory, with signature $\Sigma = (S, \leq, F)$ and subsignature Υ of $\mathcal{R}_{\mathcal{E}}$-constructors. Moreover, let $t, u \in T_{\Sigma}(X)$, $x \in vars(t, u) \cap X_s$ for some $s \in S$, and G_s a generating set for s modulo B, such that (without loss of generality) $vars(G_s) \cap vars(t, u) = \emptyset$. Then:*

$$\textit{If}\ \ \mathcal{R}_{\mathcal{E}} \Vdash (\forall X) \bigwedge_{w \in G_s} \left[\bigwedge_{y \in vars(w) \cap X_{\leq s}} (t \downarrow u)\{x \mapsto y\} \right] \Rightarrow (t \downarrow u)\{x \mapsto w\},$$

then $\mathcal{R}_{\mathcal{E}} \Vdash (\forall X)\, t \downarrow u$.

Proof. By contradiction. Suppose the antecedent holds, but there is a ground substitution $\sigma \in [vars(t, u) \longrightarrow T_{\Sigma}]$ such that $\mathcal{R}_{\mathcal{E}} \nvdash (t \downarrow u)\sigma$. Note, however, that by $\overrightarrow{E}$ being strict B-coherent and G_s being a generating set for s modulo B, σ is always of the form $\sigma =_B \{x \mapsto w\}\tau$, for some $w \in G_s$ and substitution τ, and then we have

$$\mathcal{R}_{\mathcal{E}} \nvdash (t \downarrow u)\sigma \quad \text{iff} \quad \mathcal{R}_{\mathcal{E}} \nvdash (t \downarrow u)\{x \mapsto w\}\tau.$$

Consider now the non-empty set of ground terms

$$\{w\tau \mid w \in G_s \ \wedge\ \tau \in [Y_w \longrightarrow T_{\Sigma}] \ \wedge\ \mathcal{R}_{\mathcal{E}} \nvdash (t \downarrow u)\{x \mapsto w\}\tau\}$$

where $Y_w = (vars(t, u) \setminus \{x\}) \cup vars(w)$. Pick $w\tau_0$ of smallest term size possible in the above set. By the strict B-coherence of $\overrightarrow{E}$ and the assumption that the axioms B are size-preserving, this means that for any ground substitution $\sigma \in [vars(t, u) \longrightarrow T_{\Sigma}]$, such that $\mathcal{R}_{\mathcal{E}} \nvdash (t \downarrow u)\sigma$, we must have $|\sigma(x)| \geq |w\tau_0|$. In particular, since $w \cap X = \emptyset$, this means that for each $y \in vars(w) \cap X_{\leq s}$ we must have $|\tau_0(y)| < |w\tau_0|$ and therefore $\mathcal{R}_{\mathcal{E}} \vdash (t \downarrow u)\{x \mapsto y\}\tau_0$. But, by hypothesis this implies $\mathcal{R}_{\mathcal{E}} \vdash (t \downarrow u)\{x \mapsto w\}\tau_0$, a contradiction. □

It is also sound to reason about ground joinability in $\mathcal{R}_{\mathcal{E}}$ using case analysis based on the $\mathcal{R}_{\mathcal{E}}$-constructor signature Υ.

Theorem 4. *Let $\mathcal{R}_{\mathcal{E}}$ be a weakly terminating and ground sort-decreasing rewrite theory, with signature $\Sigma = (S, \leq, F)$ and subsignature Υ of $\mathcal{R}_{\mathcal{E}}$-constructors. Moreover, let $t, u \in T_{\Sigma}(X)$, $x \in vars(t, u) \cap X_s$ for some $s \in S$, and G_s a generating set for s modulo B, such that (without loss of generality) $vars(G_s) \cap vars(t, u) = \emptyset$. Then:*

$$\mathcal{R}_{\mathcal{E}} \Vdash (\forall X)\, t \downarrow u \quad \textit{iff} \quad \mathcal{R}_{\mathcal{E}} \Vdash (\forall X) \bigwedge_{w \in G_s} (t \downarrow u)\{x \mapsto w\}.$$

Proof. If $\mathcal{R}_{\mathcal{E}} \Vdash (\forall X)\, t \downarrow u$, then clearly $\mathcal{R}_{\mathcal{E}} \Vdash (\forall X) \bigwedge_{w \in G_s} (t \downarrow u)\{x \mapsto w\}$. For the proof in the opposite direction, let $\sigma \in [X \longrightarrow T_{\Sigma}]$ be such that $\mathcal{R}_{\mathcal{E}} \nvdash (t \downarrow u)\sigma$: the goal is to show that $\mathcal{R}_{\mathcal{E}} \nVdash (\forall X) \bigwedge_{w \in G_s} (t \downarrow u)\{x \mapsto w\}$, for some $w \in G_s$. Since G_s is a generating set for the sort s and $x \in X_s$, then there is $w \in G_s$ and $\rho \in [X \longrightarrow T_{\Sigma}]$ such that $\sigma(x) =_B w\rho$. Let $\sigma' = \sigma_{|vars(t,u) \setminus \{x\}} \uplus \rho$ and observe that σ' is well-defined because of the assumption $vars(G_s) \cap vars(t, u) = \emptyset$. Furthermore, observe:

$$\begin{aligned}
(t \downarrow u)\sigma &= (t \downarrow u)\{x \mapsto \sigma(x)\}\sigma_{|vars(t,u) \setminus \{x\}} \\
&=_B (t \downarrow u)\{x \mapsto w\rho\}\sigma_{|vars(t,u) \setminus \{x\}} \\
&= (t \downarrow u)\{x \mapsto w\}(\sigma_{|vars(t,u) \setminus \{x\}} \uplus \rho) \\
&= (t \downarrow u)\{x \mapsto w\}\sigma'.
\end{aligned}$$

Hence, by the strict B-coherence of $\overrightarrow{E}$, we must have $\mathcal{R}_{\mathcal{E}} \nVdash (\forall X) \bigwedge_{w \in G_s} (t \downarrow u)\{x \mapsto w\}$. □

$$\frac{\mathcal{R}_{\mathcal{E}} \Vdash (\forall X)\ \bigwedge_{w\in G_s} \left[\bigwedge_{y\in vars(w)\cap X_{\leq s}} (t \downarrow u)\{x \mapsto y\} \right] \Rightarrow (t \downarrow u)\{x \mapsto w\}}{\mathcal{R}_{\mathcal{E}} \Vdash (\forall X)\, t \downarrow u}\ \textsc{GSInd}$$

$$\frac{\mathcal{R}_{\mathcal{E}} \Vdash (\forall X)\ \bigwedge_{w\in G_s} (t \downarrow u)\{x \mapsto w\}}{\mathcal{R}_{\mathcal{E}} \Vdash (\forall X)\, t \downarrow u}\ \textsc{CtorCases}$$

Figure 3: Inference rules for proving ground joinability for a rewrite theory $\mathcal{R}_{\mathcal{E}}$ with $\mathcal{R}_{\mathcal{E}}$-constructors Υ by induction relative to the generating set G_s and by constructor-based case analysis on a variable $x \in vars(t, u) \cap X_s$.

This concludes the inference system for proving ground joinability. However, an important practical issue remains: how should the checking of $\mathcal{R} \vdash (\forall X)\, t \downarrow u$ used in inference rule Join be best *mechanized*? After all, $t \downarrow u$ is a somewhat *complex* relation, involving existential quantification. This issue can be satisfactorily addressed by means of a program transformation $\mathcal{R}_{\mathcal{E}} \mapsto \mathcal{R}_{\mathcal{E}}^{\approx}$ that extends the possibly conditional and operationally terminating rewrite theory $\mathcal{R}_{\mathcal{E}}$, associated to an equational theory $\mathcal{E} = (\Sigma, E \uplus B)$, to a theory $\mathcal{R}_{\mathcal{E}}^{\approx}$ with: (i) a new sort *Prop* with constant `tt` and (ii) a new operator $\_ \approx \_$ with the rule $x \approx x \to tt$, such that

$$\mathcal{R}_{\mathcal{E}} \vdash (\forall X)\, t \downarrow u \quad \text{iff} \quad \mathcal{R}_{\mathcal{E}}^{\approx} \vdash (\forall X)\, t \approx u \to^{\star} tt.$$

Since the right side of the equivalence is a *reachability property* and the transformation $\mathcal{R}_{\mathcal{E}} \mapsto \mathcal{R}_{\mathcal{E}}^{\approx}$ preserves operational termination, the theory $\mathcal{R}_{\mathcal{E}}^{\approx}$ and Maude's *search* command can be used to check that $\mathcal{R} \vdash (\forall X)\, t \downarrow u$. This is used in the Example 1 below, where the binary function symbol `join` implements the operator $\_ \approx \_$. The precise description of the $\mathcal{R}_{\mathcal{E}} \mapsto \mathcal{R}_{\mathcal{E}}^{\approx}$ transformation is given in Appendix D.

Example 1. *Recall from Section 3 the only critical pair output by the CRC tool for the HF-SETS-3 specification; the goal is to prove:*

$$\textit{HF-SETS-3} \Vdash (\forall M{:}Magma; S, T{:}Set)\, t(M, S, T) \downarrow u(M, S, T)$$

where

$$\begin{aligned} t(M, S, T) &= P(\{M\}) \cup augment(P(\{M\}), S) \cup augment(P(\{M\}), T) \\ &\quad \cup augment(augment(P(\{M\}), S), T) \\ u(M, S, T) &= P(\{M\}) \cup augment(P(\{M\}), S) \cup augment(P(\{M\}), T) \\ &\quad \cup augment(augment(P(\{M\}), T), S) \end{aligned}$$

By the Ctx *rule it suffices to prove:*

$$\begin{aligned} &\textit{HF-SETS-3} \Vdash (\forall M{:}Magma; S, T{:}Set) \\ &\qquad augment(augment(P(\{M\}), S), T) \downarrow augment(augment(P(\{M\}), T), S) \end{aligned}$$

Moreover, since $P(\{M\})$ has sort Set*, this statement can be proved by considering a* *stronger property, namely, by using the* GRAL *rule and proving:*

$$\textsf{HF-SETS-3} \Vdash (\forall S, S', T : \mathit{Set})\, \mathit{augment}(\mathit{augment}(S', S), T) \downarrow \mathit{augment}(\mathit{augment}(S', T), S)$$

This proof obligation can be dealt with by using the CTORCASES *rule on $S' \in X_{\textsf{Set}}$ with generating set $G_{\textsf{Set}} = \{\{\}, \{M\}\}$ and $M \in X_{\textsf{Magma}}$. This rule application results in the following two proof obligations:*

$$\textsf{HF-SETS-3} \Vdash (\forall S, T : \mathit{Set})\, \mathit{augment}(\mathit{augment}(\{\}, S), T) \downarrow \mathit{augment}(\mathit{augment}(\{\}, T), S)$$

$$\begin{aligned}\textsf{HF-SETS-3} \Vdash{} & (\forall S, T : \mathit{Set}; M : \mathit{Magma})\\ & \mathit{augment}(\mathit{augment}(\{M\}, S), T) \downarrow \mathit{augment}(\mathit{augment}(\{M\}, T), S)\end{aligned}$$

The first proof obligation can be discharged by a search command in $\mathcal{R}^{\approx}_{\textsf{HF-SETS-3}}$:

```
search in HF-SETS-3-REACH :
  join(augment(augment({{}}, S), T), augment(augment({{}}, T), S)) =>! tt .
Solution 1 (state 1)
```

The second proof obligation can be handled using the GSIND *rule on $M \in X_{\textsf{Magma}}$ with generating set $G_{\textsf{Magma}} = \{S', (S', M')\}$, $S' \in X_{\textsf{Set}}$, and $M' \in X_{\textsf{Magma}}$:*

$$\begin{aligned}\textsf{HF-SETS-3} \vdash{} & (\forall S, S', T : \mathit{Set})\\ & \mathit{augment}(\mathit{augment}(\{S'\}, S), T) \downarrow \mathit{augment}(\mathit{augment}(\{S'\}, T), S)\\ \textsf{HF-SETS-3} \vdash{} & (\forall S, S', T : \mathit{Set}; M' : \mathit{Magma})\\ & \psi \Rightarrow \mathit{augment}(\mathit{augment}(\{S', M'\}, S), T) \downarrow \mathit{augment}(\mathit{augment}(\{S', M'\}, T), S)\end{aligned}$$

where ψ is the formula:

$$\begin{aligned}& \mathit{augment}(\mathit{augment}(\{S'\}, S), T) \downarrow \mathit{augment}(\mathit{augment}(\{S'\}, T), S) \wedge\\ & \mathit{augment}(\mathit{augment}(\{M'\}, S), T) \downarrow \mathit{augment}(\mathit{augment}(\{M'\}, T), S).\end{aligned}$$

For the first one of these two proof obligations, a proof can be found as follows:

```
search in HF-SETS-3-REACH :
  join(augment(augment({S'}, S), T), augment(augment({S'}, T), S)) =>! tt .
Solution 1 (state 14)
```

For the second proof obligation, it suffices to rewrite both terms in the consequent of the implication and use the second conjunct in ψ, together with the JOIN *and* CTX*, to join the resulting terms:*

```
search in HF-SETS-3-REACH : augment(augment({M',S'}, S), T) =>! X:Set .
Solution 1 (state 6)
X:Set --> {S' U {S,T}} U augment(augment({M'}, S), T)


search in HF-SETS-3-REACH : augment(augment({M',S'}, T), S) =>! X:Set .
Solution 1 (state 6)
X:Set --> {S' U {S,T}} U augment(augment({M'}, T), S)
```

Therefore, all critical pairs of HF-SETS-3 *are ground joinable; hence,* HF-SETS-3 *is ground convergent, as desired.*

But is the original specification HF-SETS itself ground convergent? That is, can the extra equations in HF-SETS-3 just be used as *scaffolding* and then be removed as unnecessary? The following result shows that, if the successive addition of oriented equalities leads us to a ground convergent theory and such equalities are ground joinable, then the added equations are indeed unnecessary. The main idea is that, starting from an equational specification $\mathcal{E}_0$, if a sequence of equational theories $\mathcal{E}_0 \subseteq \mathcal{E}_1 \subseteq \cdots \subseteq \mathcal{E}_n$ can be built by incrementally adding new equations (e.g., suggested by the analysis of critical pairs between the equations), and if the new equations added at each step can be shown ground joinable, then the ground confluence of $\mathcal{E}_n$ implies the ground confluence of each $\mathcal{E}_i$, and in particular of $\mathcal{E}_0$.

Theorem 5. *Let $(\Sigma, E_0 \uplus B) \subseteq (\Sigma, E_1 \uplus B)$ where $\overrightarrow{E_0}, B$ is sufficiently complete with respect to a subsignature Ω, $(\Sigma, E_1 \uplus B)$ is ground convergent, $\to_{\overrightarrow{E_0},B} |_\Omega \; = \; \to_{\overrightarrow{E_1},B} |_\Omega$, and all equations in $E_1 - E_0$ are ground E_0, B-joinable. Then,*

$$\left(\to^!_{\overrightarrow{E_0},B}; =_B\right)|_{T_\Sigma} = \left(\to^!_{\overrightarrow{E_1},B}; =_B\right)|_{T_\Sigma}.$$

That is, the normal forms of the rewriting relation modulo B restricted to the initial term algebra T_Σ coincide.

Proof. First of all note that, since $\overrightarrow{E_0} \subseteq \overrightarrow{E_1}$, $(\Sigma, B, \overrightarrow{E_0})$ is operationally terminating. Consider some $t \in T_\Sigma$ and rewrite $t \to^!_{\overrightarrow{E_1},B} u$. Since $\overrightarrow{E_0}, B$ is sufficiently complete and $\to_{\overrightarrow{E_0},B} |_\Omega \; = \; \to_{\overrightarrow{E_1},B} |_\Omega$, $u \in T_\Omega$. If all rules applied in the chain are in $\overrightarrow{E_0}$, then the chains obviously coincide. Otherwise, let us consider the first rewrite step using a rule in $\overrightarrow{E_1} - \overrightarrow{E_0}$:

$$\begin{tikzcd}
t \arrow[r, "*", "\overrightarrow{E_0},B"'] & {} \arrow[rrr, "\overrightarrow{E_1}-\overrightarrow{E_0},B"'] \arrow[dr, "!", "\overrightarrow{E_0},B"'] & & & {} \arrow[r, "!", "\overrightarrow{E_1},B"'] \arrow[dl, "!"', "\overrightarrow{E_0},B"] & u \\
& & v \arrow[r, equal, "B"'] & w & &
\end{tikzcd}$$

First, we have $v =_B w$ by ground joinability of equations in $E_1 - E_0$. Then, by the assumption that $\to_{\overrightarrow{E_0},B} |_\Omega = \to_{\overrightarrow{E_1},B} |_\Omega$, u and w are in E_1, B-canonical form, and by the ground confluence of $\overrightarrow{E_1}, B$ we must have $u =_B w$. Therefore, we can conclude that $\to^!_{\overrightarrow{E_0},B}; =_B |_{T_\Sigma} = \to^!_{\overrightarrow{E_1},B}; =_B |_{T_\Sigma}$, as desired. □

Theorem 6. *Suppose $(\Sigma, E_0 \uplus B) \subseteq \ldots \subseteq (\Sigma, E_n \uplus B)$, with $n \geq 0$, such that $\overrightarrow{E_0} \uplus B$ is sufficiently complete with respect to a subsignature Ω, $\overrightarrow{E_n} \uplus B$ is ground convergent, $(\to_{\overrightarrow{E_0},B})|_\Omega = (\to_{\overrightarrow{E_n},B})|_\Omega$, and all $E_{i+1} - E_i$ are ground E_i-joinable modulo B. Then, each $(\Sigma, E_i \uplus B)$ is ground convergent, for $0 \leq i \leq n$. Furthermore, all theories in the chain have the same initial algebra.*

Proof. By induction on n. It is trivial for $n = 0$. Suppose it true for n, and let us prove it true for $n + 1$. Given a chain $(\Sigma, E_0 \uplus B) \subseteq (\Sigma, E_1 \uplus B) \subseteq \ldots \subseteq (\Sigma, E_n \uplus B)$, by the induction hypothesis —plus the fact that $(\Sigma, E_0 \uplus B)$ sufficiently complete makes $(\Sigma, E_1 \uplus B)$ so as well— we get that $(\Sigma, E_1 \uplus B)$ is ground convergent. The proof that $(\Sigma, E_0 \uplus B)$ is ground convergent is as follows. Since $(\Sigma, E_1 \uplus B)$ is ground convergent, $(\Sigma, E_0 \uplus B)$ is *a fortiori* sort-decreasing and operationally terminating, so all we need to prove is its ground confluence. But since, by Theorem 5, $\rightarrow^{!}_{\overrightarrow{E_0},B} ; =_B |_{T_\Sigma} = \rightarrow^{!}_{\overrightarrow{E_1},B} ; =_B |_{T_\Sigma}$, the following diagram proves ground confluence of $\rightarrow_{\overrightarrow{E_0},B}$:

$$\begin{tikzcd}
 & & t \arrow[dll, "\overrightarrow{E_0},B"', "*"] \arrow[ddl, "\overrightarrow{E_1},B", "!"'] \arrow[ddr, "\overrightarrow{E_1},B"', "!"] \arrow[drr, "\overrightarrow{E_0},B", "*"'] & & \\
u \arrow[dr, "\overrightarrow{E_0},B"', "!"] & = & & = & v \arrow[dl, "\overrightarrow{E_0},B", "!"'] \\
 & u' \arrow[rr, dotted, no head, "B"'] & & v' &
\end{tikzcd}$$

Note that $u' =_B v'$ by ground confluence of $\rightarrow_{\overrightarrow{E_1},B}$.

Finally, we already know by the Induction Hypothesis that all the theories

$$(\Sigma, E_1 \uplus B) \subseteq \cdots \subseteq (\Sigma, E_n \uplus B)$$

have the same initial algebra, and, by ground-joinability of $E_1 - E_0$, that

$$\mathcal{T}_{\Sigma/E_0 \uplus B} \models E_1 - E_0.$$

Therefore, we also get $\mathcal{T}_{\Sigma/E_0 \uplus B} = \mathcal{T}_{\Sigma/E_1 \uplus B}$, as desired. □

Theorem 6 justifies the view of the new equations suggested by critical pairs obtained, say, from the CRC tool, as hints for extending our original specification as "scaffolding" that can be abandoned *after* we have reached a ground convergent extension $(\Sigma, E_n \uplus B)$. Going back to the example in Section 3, once the HF-SETS-3 module has been proven ground convergent, we can conclude that the original HF-SETS specification is also ground convergent, *provided* we can show that the equations added at stage $i + 1$ were ground joinable relative to stage i. This is shown to be the case in Section 5 by providing proofs of ground joinability for the five equations added in HF-SETS-0, HF-SETS-1, HF-SETS-2, and HF-SETS-3 in Section 3.

5. Proving the Ground Convergence of HF-SETS with Method 2

The goal of this section is to conclude that the equational specification HF-SETS presented in Section 3 is ground convergent, and therefore that its initial model is a model of set theory without the axiom of infinity. The key tools for achieving this goal are the inference system for inductive joinability and Theorem 6, both presented in Section 4. By knowing that $\mathcal{R}_{\text{HF-SETS-3}}$ is terminating (see [**?** , Appendix B]), sort decreasing (Section 3), and that HF-SETS is sufficiently complete (see [**?** , Appendix C]), the conditions in Theorem 6 apply and we just need to show the ground joinability of the added equations.

That is, since HF-SETS-3 is ground convergent and the theory inclusions

$$\text{HF-SETS} \subseteq \text{HF-SETS-0} \subseteq \text{HF-SETS-1} \subseteq \text{HF-SETS-2} \subseteq \text{HF-SETS-3}$$

satisfy the requirements of Theorem 6, it suffices to prove

$$\text{HF-SETS} \Vdash (\forall M\!:\!\mathit{Magma})\, M \in \{M\} \downarrow \mathit{true}$$

$$\text{HF-SETS} \Vdash (\forall M, M'\!:\!\mathit{Magma})\, M \in \{M, M'\} \downarrow \mathit{true}$$

$$\text{HF-SETS-0} \Vdash (\forall S : \mathit{Set})\, S \cup S \downarrow S$$

$$\text{HF-SETS-1} \Vdash (\forall S, S', T\!:\!\mathit{Set})\, \mathit{augment}(S \cup S', T) \downarrow \mathit{augment}(S, T) \cup \mathit{augment}(S', T)$$

$$\textsf{HF-SETS-2} \Vdash (\forall S, T : \mathit{Set})\, \mathit{augment}(\mathit{augment}(S, T), T) \downarrow \mathit{augment}(S, T)$$

in order to conclude that HF-SETS is ground convergent. In what follows, detailed proofs are provided for the last three proof obligations. The first two properties can be proved by following a similar approach.

The third proof obligation is dealt with by using the CtorCases rule on $S \in X_{\mathsf{Set}}$ with generating set $G_{\mathsf{Set}} = \{\{\}, \{M\}\}$ and $M \in X_{\mathsf{Magma}}$:

$$\textsf{HF-SETS-0} \Vdash \{\} \cup \{\} \downarrow \{\}$$

$$\textsf{HF-SETS-0} \Vdash (\forall M : \mathit{Magma})\, \{M\} \cup \{M\} \downarrow \{M\}$$

These two proof obligations can be automatically discharged by Maude in $\mathcal{R}^{\approx}_{\textsf{HF-SETS-0}}$:

```
search in HF-SETS-0-REACH : join({} U {}, {}) =>! tt .
Solution 1 (state 2)


search in HF-SETS-0-REACH : join({M} U {M}, {M}) =>! tt .
Solution 1 (state 3)
```

Next, for the fourth proof obligation, several inference steps are needed. First, the CtorCases rule is used on S_{Set} with generating set $G_{\mathsf{Set}} = \{\{\}, \{M\}\}$ and $M \in X_{\mathsf{Magma}}$, resulting in the following proof obligations:

$$\textsf{HF-SETS-1} \Vdash (\forall S', T : \mathit{Set})\, \mathit{augment}(\{\} \cup S', T) \downarrow \mathit{augment}(\{\}, T) \cup \mathit{augment}(S', T)$$

$$\textsf{HF-SETS-1} \Vdash (\forall S', T : \mathit{Set}; M : \mathit{Magma})$$
$$\mathit{augment}(\{M\} \cup S', T) \downarrow \mathit{augment}(\{M\}, T) \cup \mathit{augment}(S', T)$$

For the second one of these two proof obligations, the CtorCases rule on $S' \in X_{\mathsf{Set}}$ with generating set $H'_{\mathsf{Set}} = \{\{\}, \{M'\}\}$ and $M' \in X_{\mathsf{Magma}}$ is used; this transforms the second proof obligation in the following two proof obligations:

$$\textsf{HF-SETS-1} \Vdash (\forall T : \mathit{Set}; M : \mathit{Magma})$$
$$\mathit{augment}(\{M\} \cup \{\}, T) \downarrow \mathit{augment}(\{M\}, T) \cup \mathit{augment}(\{\}, T)$$

$$\textsf{HF-SETS-1} \Vdash (\forall T : \mathit{Set}; M, M' : \mathit{Magma})$$
$$\mathit{augment}(\{M\} \cup \{M'\}, T) \downarrow \mathit{augment}(\{M\}, T) \cup \mathit{augment}(\{M'\}, T)$$

The remaining three proof obligations can be automatically discharged by Maude in $\mathcal{R}^{\approx}_{\textsf{HF-SETS-1}}$ as follows:

```
search in HF-SETS-1-REACH : join(augment({} U S', T), augment({}, T) U augment(S', T)) =>! tt .
Solution 1 (state 6)


search in HF-SETS-1-REACH :
  join(augment({} U {M}, T), augment({}, T) U augment({M}, T)) =>! tt .
Solution 1 (state 6)


search in HF-SETS-1-REACH :
  join(augment({M} U {M'}, T), augment({M}, T) U augment({M'}, T)) =>! tt .
Solution 1 (state 3)
```

The fifth, and last proof obligation, is dealt with by using the CtorCases rule on $S \in X_{\mathsf{Set}}$ with generating set $G_{\mathsf{Set}} = \{\{\}, \{M\}\}$ and $M \in X_{\mathsf{Magma}}$. This rule application results in the following two proof obligations:

$$\textsf{HF-SETS-2} \Vdash (\forall T : \mathit{Set})\, \mathit{augment}(\mathit{augment}(\{\}, T), T) \downarrow \mathit{augment}(\{\}, T)$$

$$\textsf{HF-SETS-2} \Vdash (\forall T\!:\!Set;\, M\!:\!Magma)$$
$$augment(augment(\{M\},T),T) \downarrow augment(\{M\},T)$$

The first proof obligation can be discharged automatically:

```
search in HF-SETS-2-REACH : join(augment(augment({}, T), T), augment({}, T)) =>! tt .
Solution 1 (state 4)
```

The remaining proof obligation can be handled with the help of the GSInd rule with generating set $G_{\textsf{Magma}} = \{S', (S', M')\}$, $S' \in X_{\textsf{Set}}$ and $M' \in X_{\textsf{Magma}}$:

$$\textsf{HF-SETS-2} \Vdash (\forall S', T\!:\!Set)\, augment(augment(\{S'\},T),T) \downarrow augment(\{S'\},T)$$
$$\textsf{HF-SETS-2} \Vdash (\forall\ S', T\!:\!Set;\, M'\!:\!Magma)$$
$$\psi \Rightarrow augment(augment(\{S', M'\},T),T) \downarrow augment(\{S', M'\},T)$$

where ψ is the formula:

$$augment(augment(\{S'\},T),T) \downarrow augment(\{S'\},T)$$
$$\wedge\ augment(augment(\{M'\},T),T) \downarrow augment(\{M'\},T).$$

These two proof obligations can be solved with the help of Maude:

```
search in HF-SETS-2-REACH : join(augment(augment({S'}, T), T), augment({S'}, T)) =>! tt .
Solution 1 (state 10)


search in HF-SETS-2-REACH : augment(augment({M',S'}, T), T) =>! X:Set .
Solution 1 (state 7)
X:Set --> {S' U {T}} U augment(augment({M'}, T), T)


search in HF-SETS-2-REACH : augment({M',S'}, T) =>! X:Set .
Solution 1 (state 2)
X:Set --> {S' U {T}} U augment({M'}, T)
```

Note that the terms obtained by the last two search commands can be joined by assuming ψ.

The initial goal has now been reached. Namely, since all the equations added in the process of building the tower of theory inclusions

$$\textsf{HF-SETS} \subseteq \textsf{HF-SETS-0} \subseteq \textsf{HF-SETS-1} \subseteq \textsf{HF-SETS-2} \subseteq \textsf{HF-SETS-3}$$

have been shown ground joinable, Theorem 6 guarantees that the equational specification HF-SETS for hereditarily finite sets is ground convergent.

6. Method 3: Hierarchical Conditional Ground Joinability Techniques

So far we have only considered *unconditional* ground joinability problems. However, since we allow equational theories to be conditional and their associated rewrite theories to be *strongly deterministic*, in general to prove that a rewrite theory is ground confluent will require us to prove that certain *conditional critical pairs* are *ground joinable*. To make the exposition self-contained, let us recall the basic notions from [14], where a detailed exposition of order-sorted conditional confluence issues can be found.

Definition 5. *Given an order-sorted equational specification* $(\Sigma, B, \overrightarrow{E})$ *that is strongly deterministic in the sense explained in Section 2, and given (possibly renamed) conditional rewrite rules* $l \to r$ *if* C *and* $l' \to r'$ *if* C' *in* $\overrightarrow{E}$ *such that* $vars(l \to r \text{ if } C) \cap vars(l' \to r' \text{ if } C') = \emptyset$ *and* $l|_p\sigma =_B l'\sigma$*, for some nonvariable position* $p \in \mathcal{P}(l)$ *and* B*-unifier* $\sigma \in Unif_B(l_p, l')$*, then the triple*

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

is called a conditional critical pair *(CCP). Assuming that* $(\Sigma, B, \overrightarrow{E})$ *is weakly terminating, we call such a CCP* ground-joinable *iff for each* ground substitution θ *of all variables appearing in such a CCP and such that* $(\Sigma, B, \overrightarrow{E}) \models (C\sigma \wedge C'\sigma)\theta$ *we have* $(\Sigma, B, \overrightarrow{E}) \models (l\sigma[r'\sigma]_p)\theta \downarrow r\sigma\theta$*.*

A key question is what proof techniques can be used to prove that a CCP is ground joinable. Certainly, it will be so if we can show that it is *context-joinable* or *unfeasible* in the sense explained in [2, 14], since then the CCP is actually joinable or unfeasible. But since these techniques aim at proving the stronger property of joinability (they are actually used in Maude's Church-Rosser Checker [14]), they are usually insufficient for proving ground joinability.

Two complementary options suggest themselves quite naturally. On the one hand, we can *generalize* the ground joinability inference system of Section 4 to reason about inductive *conditional* joinability. On the other, we can develop *hierarchical methods* that allow us to reason in a *modular* way. Suppose that the order-sorted equational specification $(\Sigma, B, \overrightarrow{E})$ imports an equational subspecification $(\Sigma_0, B_0, \overrightarrow{E_0})$ in a *conservative* way—to be made precise in what follows. Furthermore suppose that: (i) $(\Sigma_0, B_0, \overrightarrow{E_0})$ has already been shown ground convergent; and (ii) the CCP in question has been obtained by overlapping rules $l \to r$ *if* C and $l' \to r'$ *if* C' in $\overrightarrow{E}$ such that their conditions C and C' are Σ_0-conditions. Then, under appropriate requirements, we may be able to *use* the fact that the ground version of the Church-Rosser Theorem holds for the equational specification $(\Sigma_0, B_0, \overrightarrow{E_0})$ to *reason inductively* about the properties of the Σ_0-subterms appearing in the two sides of the CCP under the assumption that its Σ_0-condition holds for a given ground substitution to prove the CCP's ground joinability.

6.1. A Rational Numbers Example

To give an intuition for what this hierarchical proof technique can look like, let us consider the RAT-ACU module in Figure 4 that specifies the rational numbers. RAT-ACU imports a sub-specification INT-ACU of the integers that has already been proved not just ground convergent, but actually *convergent*. As we shall see, INT-ACU is imported by RAT-ACU in a conservative manner to be made precise later. The entire number hierarchy specification can be found in Appendix E.

Proving the ground confluence of RAT-ACU or a similar algebraic specification of the field of rational numbers is an outstanding open problem in algebraic specification. First of all, note that, to avoid the problem of division by zero, the use of a subsort NzRat < Rat is essential. Second, since the rationals are such an important data type, order-sorted specifications of them have always been used in OBJ [20], CafeOBJ [19], and Maude [6], and have always been considered as important benchmarks in order-sorted algebra (see, e.g., the Appendix in [21]). However, proving the ground confluence of an order-sorted specification of the rationals has eluded previous efforts since the 1980s.

In the RAT-ACU specification there is only one conditional rule, namely, RAT-ACU-02, which puts a fraction in canonical form by dividing its numerator and denominator by their greater common denominator (gcd). Since this conditional rule packs a lot of arithmetic knowledge into a single punch, it is not surprising at all that it generates quite a number of critical pairs that cannot be joined by standard techniques, because such arithmetic knowledge is only *implicit* in the equational theories RAT-ACU and INT-ACU. Furthermore, it is implicit *not* as *equational reasoning* knowledge, but as *inductive* knowledge about the *initial algebras* of INT-ACU and RAT-ACU, namely, $\mathbb{Z}$ and $\mathbb{Q}$. That is why standard rewrite techniques are insufficient to prove the CCPs that are generated joinable: they are only *ground* joinable.

For example, the overlap of rules RAT-ACU-02 and RAT-ACU-12 generates the following conditional critical pair[3]:

$$\mathsf{gcd}(x_2, x_1) > 1 \to^* \mathsf{true} \Rightarrow \frac{x_3 * \mathsf{quot}(x_1, \mathsf{gcd}(x_2, x_1))}{\mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_1))} \downarrow \frac{x_1 * x_3}{x_2}$$

That any ground instance of the two fractions in the critical pair can be simplified to the same fraction can be understood as a relatively simple arithmetic property, since dividing two numbers by their gcd makes the quotients relative prime to each other. However, both fractions may require further steps of simplification by rule RAT-ACU-02 to reach their canonical forms. The point is that all these arithmetic facts are *inductive knowledge* about integer arithmetic, that is, about the *initial algebra* $\mathbb{Z}$ of the equational specification INT-ACU. The good news is that INT-ACU is convergent[4] and, a fortiori, ground convergent. So "all we need" is to exploit the (ground version of the) Church-Rosser Theorem for INT-ACU to prove this CCP ground joinable under the assumption that its condition holds for a given ground substitution. But before we can do this, some technical notions and results are needed.

6.2. Hierarchical Conditional Ground Confluence Techniques

Let us begin with the definition of conservative extension relative to a set of sorts.

Definition 6. *Let* $(\Sigma_0, E_0 \uplus B_0) \subseteq (\Sigma, E \uplus B)$ *and let* S'_0 *be a subset of sorts in* Σ_0. *The rewrite theory* $(\Sigma, B, \vec{E})$ *is called a* conservative extension of $(\Sigma_0, B_0, \vec{E}_0)$ relative to S'_0 *iff:*

1. $((\to_{E,B} |_{T_{\Sigma_0}(X)}); =_B) \; = \; ((\to_{E_0,B_0}); =_{B_0})$ *and*

2. $T_\Sigma(X)|_{S'_0} = T_{\Sigma_0}(X)|_{S'_0}$.

The following theorem states that if $(\Sigma_0, E_0 \uplus B_0) \subseteq (\Sigma, E \uplus B)$ is a conservative extension then the satisfiability of Σ_0-conditions may be evaluated in $(\Sigma_0, E_0 \uplus B_0)$.

[3] The critical pair is named RAT-ACU5552 in the output generated by the CRC tool, see [**?** , Appendix G].

[4] The output of the CRC tool of the checks of confluence and sort-decreasingness of the INT-ACU module can be found in [**?** , Appendix H].

```
fmod RAT-ACU is
  protecting INT-ACU .
  sorts Rat NzRat PosRat NzPosRat .
  subsort Int < Rat .
  subsorts Nat NzPosRat < PosRat < Rat .
  subsorts NzInt < NzRat < Rat .
  subsorts NzNat < NzPosRat < NzRat PosRat .
  op _/_ : NzRat NzRat -> NzRat [prec 31] .        *** division
  op _/_ : NzInt NzNat -> NzRat [ctor prec 31] .
  op _/_ : NzNat NzNat -> NzPosRat [ctor prec 31] .
  op -_ : Rat -> Rat .
  op _+_ : Rat Rat -> Rat [assoc comm id: 0 prec 33] .
  op _+_ : PosRat PosRat -> PosRat [assoc comm id: 0 prec 33] .
  op _*_ : Rat Rat -> Rat [assoc comm prec 31] .
  op _*_ : NzRat NzRat -> NzRat [assoc comm prec 31] .
  op |_| : Rat -> PosRat .                          *** absolute value
  op _~_ : Rat Rat -> Bool [comm] .                 *** equality predicate
  op _>_ : Rat Rat -> Bool .

  vars I’ J’ : NzInt .
  vars R’ S’ : NzRat .
  vars N’ M’ : NzNat .

  eq [RAT-ACU-01]: 0 / N’ = 0 .
 ceq [RAT-ACU-02]: J’ / M’ = quot(J’, gcd(J’, M’)) / quot(M’, gcd(J’, M’))
     if gcd(J’, M’) > 1 = true .                    *** canonical form of fraction
  eq [RAT-ACU-03]: R’ / 1 = R’ .
  eq [RAT-ACU-04]: I’ / - N’ = - I’ / N’ .
  eq [RAT-ACU-05]: I’ / (J’ / M’) = (I’ * M’) / J’ .
  eq [RAT-ACU-06]: (I’ / N’) / J’ = I’ / (N’ * J’) .
  eq [RAT-ACU-07]: (I’ / N’) / (J’ / M’) = (I’ * M’) / (N’ * J’) .
  eq [RAT-ACU-08]: - (I’ / N’) = - I’ / N’ .
  eq [RAT-ACU-09]: J’ + (I’ / N’) = ((J’ * N’) + I’) / N’ .
  eq [RAT-ACU-10]: (I’ / N’) + (J’ / M’) = ((I’ * M’) + (J’ * N’)) / (N’ * M’) .
  eq [RAT-ACU-11]: (I’ / N’) * 0 = 0 .
  eq [RAT-ACU-12]: (I’ / N’) * J’ = (I’ * J’) / N’ .
  eq [RAT-ACU-13]: (I’ / N’) * (J’ / M’) = (I’ * J’) / (N’ * M’) .
  eq [RAT-ACU-14]: | I’ / N’ | = | I’ | / N’ .
  eq [RAT-ACU-15]: 0 ~ R’ = false .
  eq [RAT-ACU-16]: (I’ / N’) ~ J’ = I’ ~ (J’ * N’) .
  eq [RAT-ACU-17]: (I’ / N’) ~ (J’ / M’) = (I’ * M’) ~ (J’ * N’) .
  eq [RAT-ACU-18]: 0 > N’ / M’ = false .
  eq [RAT-ACU-19]: 0 > - N’ / M’ = true .
  eq [RAT-ACU-20]: N’ / M’ > 0 = true .
  eq [RAT-ACU-21]: - N’ / M’ > 0 = false .
  eq [RAT-ACU-22]: I’ > (J’ / M’) = (I’ * M’) > J’ .
  eq [RAT-ACU-23]: (I’ / N’) > J’ = I’ > (J’ * N’) .
  eq [RAT-ACU-24]: (I’ / N’) > (J’ / M’) = (I’ * M’) > (J’ * N’) .
endfm
```

Figure 4: RAT-ACU module specifying the rational numbers

Theorem 7. *Let* $(\Sigma, B, \overrightarrow{E})$ *be a conservative extension of the rewrite theory* $(\Sigma_0, B_0, \overrightarrow{E}_0)$ *relative to the subset of sorts* S'_0 *of* Σ_0*. If for each conditional rule*

$$(l \to r \ if\ C) \in \overrightarrow{E} \setminus \overrightarrow{E}_0$$

where C *is a* Σ_0*-condition, and letting* $Y = vars(C)$*, we have* $sorts(Y) \subseteq S'_0$*, then for each substitution* $\theta \in [Y \longrightarrow T_\Sigma(X)]$

$$(\Sigma, B, \overrightarrow{E}) \models C\theta \ \Leftrightarrow\ (\Sigma_0, B_0, \overrightarrow{E_0}) \models C\theta.$$

Proof. First of all note that, by the conservativity assumption relative to S'_0, we have $[Y \to T_\Sigma(X)] = [Y \to T_{\Sigma_0}(X|_{S'_0})$. But then, for each $u \to^* v$ in the condition C and $u, v \in T_{\Sigma_0}(X)$, and $\theta \in \left[Y \longrightarrow T_{\Sigma_0}(X)\,|_{S'_0}\right]$, by the conservativity assumption and the fact that $(\Sigma, B, \overrightarrow{E})$ and $(\Sigma_0, B_0, \overrightarrow{E}_0)$ are always assumed to be strictly B-coherent we also have

$$\exists w\ (u\theta \to^*_{\overrightarrow{E},B} w =_B v\theta) \ \Leftrightarrow\ \exists w'\ (u\theta \to^*_{\overrightarrow{E_0},B_0} w' =_{B_0} v\theta)$$

□

The following result provides a mechanism for checking whether an extension is conservative.

Theorem 8. *Let* $(\Sigma_0, B_0, \overrightarrow{E_0}) \subseteq (\Sigma, B, \overrightarrow{E})$ *be an extension of rewrite theories, and* S'_0 *a subset of the sorts* S_0 *in* Σ_0 *such that*

(a) $\forall u, v \in T_{\Sigma_0}(X_{S_0}),\ u =_B v \Leftrightarrow u =_{B_0} v$*, and*

(b) $T_\Sigma(X)\,|_{S'_0} = T_{\Sigma_0}(X_{S_0})\,|_{S'_0}$.

Furthermore, assume that for each $l \to r\ if\ C$ *in* $\overrightarrow{E} \setminus \overrightarrow{E}_0$*, with* $Z = vars(l \to r\ if\ C)$ *and* $Z_0 = vars(C)$*,* C *is a* Σ_0*-condition such that* $sorts(Z_0) \subseteq S'_0$*. Then, if for each such rule, and each substitution* $\theta \in [Z \longrightarrow T_\Sigma(X)]$ *such that* $l\theta \in T_{\Sigma_0}(X_{S_0})$ *and* $(\Sigma, B, \overrightarrow{E}) \models C\theta$ *there is* $(l' \to r'\ if\ C') \in R_0$ *with* $Z' = vars(l' \to r'\ if\ C')$ *and* $\gamma \in [Z' \longrightarrow T_{\Sigma_0}(X\,|_{S_0})]$ *such that* $l\theta =_{B_0} l'\gamma$*,* $r\theta =_{B_0} r'\gamma$*, and* $(\Sigma_0, B_0, \overrightarrow{E_0}) \models C'\gamma$*, then*

$$\Big((\to_{\overrightarrow{E},B}|_{T_{\Sigma_0}(X_{S_0})}); =_B \Big) = \Big(\to_{\overrightarrow{E_0},B_0}; =_{B_0} \Big).$$

Proof. We of course have $\to_{\overrightarrow{E_0},B_0} \subseteq \to_{\overrightarrow{E},B}|_{T_{\Sigma_0}(X)}$, so we only need to show that for any $u, v, w \in T_{\Sigma_0}(X)$, $u \to_{R,B} v =_B w$ implies $u \to_{R_0,B_0} v' =_{B_0} w$. But $u \to_{R,B} v$ means that there is a rule $l \to r\ if\ C \in R$, with $Z = vars(l \to r\ if\ C)$, a position p, and a substitution $\theta \in [Z \longrightarrow T_\Sigma(X)]$ such that $u\,|_p =_B l\theta \in T_{\Sigma_0}(X\,|_{S_0})$, $v \equiv u[r\theta]_p$, and $(\Sigma, B, \overrightarrow{E}) \models C\theta$. By the Theorem's hypothesis we have a rule $l' \to r'\ if\ C' \in R_0$, with $Z' = vars(l' \to r'\ if\ C')$, and substitution $\gamma \in [Z' \longrightarrow T_{\Sigma_0}(X\,|_{S_0})]$ such that $u\,|_p =_B l\theta =_{B_0} l'\gamma$, $r\theta =_{B_0} r'\gamma$, and $(\Sigma_0, B_0, \overrightarrow{E_0}) \models C'\gamma$. But this then means that $u \to_{R_0,B_0} v'$, where $v' \equiv u[r'\gamma]_p =_{B_0} v =_B w$. But, by hypothesis, $u, v \in T_{\Sigma_0}(X\,|_{S_0})$, and therefore, by *(a)* we have $v =_{B_0} w$, proving $v' =_B w$, as desired. □

Finally, Theorem 9 provides a way to handle CCPs whose conditions belong to a sub-specification that is conservatively extended.

Theorem 9. *Let $\mathcal{R}_0 = (\Sigma_0, B_0, \overrightarrow{E_0}) \subseteq (\Sigma, B, \overrightarrow{E}) = \mathcal{R}$ be a conservative extension relative to a set S'_0 of sorts of Σ_0 such that B and $B_0 \subseteq B$ are linear and regular axioms having unification algorithms, $\mathcal{R}_0$ is ground convergent, and $\mathcal{R}$ is ground sort-decreasing, and operationally terminating.*[5] *Let*

$$D \;\Rightarrow\; u \downarrow v \qquad (\dagger)$$

be a conditional critical pair in $\mathcal{R}$ with D a Σ_0-condition whose variables have all sorts in S'_0. Let $D^{=}$ be the conjunction of equalities associated to the conjunction of rewrite conditions D. Then,

(1) If $\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \neg(\exists D^{=})$, where $\exists D^{=}$ is the existential closure of D, then the critical pair $(\dagger)$ can be discarded.

(2) Let $Y = vars(D \;\Rightarrow\; u \downarrow v)$, and let $p_1, \ldots, p_n$, resp. $q_1, \ldots, q_m$, be disjoint positions in u, resp. v, such that $ls(u|_{p_i}) \in S'_0$, $1 \leq i \leq n$, resp. $ls(v|_{q_j}) \in S'_0$, $1 \leq j \leq m$. Let then $z_1, \ldots, z_n$, resp. $z_{n+1}, \ldots, z_{n+m}$ be fresh new variables of such sorts that can be used as abstraction variables *that abstract the terms $u|_{p_i}$, resp. $v|_{q_j}$. Then, let*

$$A \equiv z_1 = u|_{p_1} \wedge \ldots \wedge z_n = u|_{p_n} \wedge z_{n+1} = v|_{q_1} \wedge \ldots \wedge z_{n+m} = v|_{q_m}$$

be such a collection of abstraction equations, and

$$A_{aux} \equiv z_{n+m+1} = w_1 \wedge \ldots \wedge z_{n+m+m'} = w_{m'}$$

where $w_l \in T_{\Sigma_0}(Y)|_{S'_0}$, $1 \leq l \leq m'$, and $Z = \{z_1, \ldots, z_{n+m+m'}\}$ are variables of the corresponding least sorts, so that they are all contained in S'_0. Assume that

(i)

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \forall (Y \cup Z)\Big((D^{=} \wedge A \wedge A_{aux}) \Rightarrow \bigvee_{k \in K} G_k\Big)$$

where, for each $k \in K$, G_k is a set of equations such that if $t = t' \in G_k$, then $t, t' \in T_{\Sigma_0}(Z)$, and either $t' \in Z$, or $t' \in T_{\Sigma_0}$ and t' is in $\overrightarrow{E_0}, B_0$-normal form.

(ii) For each $k \in K$,

$$(\Sigma(Z), B, \overrightarrow{E} \cup \overrightarrow{G_k}) \models u[z_1, \ldots, z_n]_{p_1 \ldots p_n} \downarrow v[z_{n+1}, \ldots, z_{n+m}]_{q_1 \ldots q_m}$$

where $\Sigma(Z)$ is the signature obtained by adding the variables Z as fresh new constants, and $\overrightarrow{G_k}$ are the rules $t \to t'$ for each $t = t' \in G_k$.

Then the conditional critical pair $(\dagger)$ is ground joinable.

(3) If each non-joinable critical pair in $\mathcal{R}$ can either be discarded using (1), or shown ground joinable using (2), then $\mathcal{R}$ is ground convergent.

Before proving Theorem 9, let us illustrate in detail the use of its case (2) with an example.

Example 2. *Let $\mathcal{R}_0 = \mathcal{R}_{\textsf{INT-ACU}}$ be the rewrite theory associated to* INT-ACU, *and $\mathcal{R} = \mathcal{R}_{\textsf{RAT-ACU}}$ the rewrite theory associated to* RAT-ACU,[6] *$S'_0 = \{$*Int*, *NzInt*, *Nat*, *NzNat*$\}$, and*

[5] And, as all other theories, always assumed to be strongly deterministic and strictly B-coherent.

[6] The critical pair in the example is labelled RAT-ACU5552 in the output given by the CRC tool. It comes from the overlap of equations RAT-ACU-02 and RAT-ACU-12. The INT-ACU and RAT-ACU specifications, together with the rest of the modules in the hierarchy, can be found in Appendix E.

consider again the conditional critical pair

$$\mathit{gcd}(x_2, x_1) > 1 \to^* \mathit{true} \Rightarrow x_3 * \frac{\mathit{quot}(x_1, \mathit{gcd}(x_2, x_1))}{\mathit{quot}(x_2, \mathit{gcd}(x_2, x_1))} \downarrow \frac{x_1 * x_3}{x_2}$$

where x_1 and x_3 have sort NzInt*, and x_2 has sort* NzNat*. The lefthand side term positions are* 1, 2.1 *and* 2.2*, and the righthand side term positions are* 1 *and* 2*. That is, all Σ_0-subterms will be abstracted out. Then,*

$$\begin{aligned}
A \equiv \{\ & z_1 = x_3, && (01)\\
& z_2 = \mathit{quot}(x_1, \mathit{gcd}(x_2, x_1)), && (02)\\
& z_3 = \mathit{quot}(x_2, \mathit{gcd}(x_2, x_1)), && (03)\\
& z_4 = x_1 * x_3, && (04)\\
& z_5 = x_2\ \} && (05)\\
A_{aux} \equiv \{\ & z_6 = x_3 * \mathit{quot}(x_1, \mathit{gcd}(x_2, x_1)), && (06)\\
& z_7 = \mathit{quot}(x_1 * x_3, \mathit{gcd}(x_2, x_1 * x_3)), && (07)\\
& z_8 = \mathit{quot}(x_2, \mathit{gcd}(x_2, x_1 * x_3)), && (08)\\
& z_9 = \mathit{gcd}(x_3 * \mathit{quot}(x_1, \mathit{gcd}(x_2, x_1)), \mathit{quot}(x_2, \mathit{gcd}(x_2, x_1)))\ \} && (09)\\
G_0 \equiv \{\ & \mathit{gcd}(z_4, z_5) > 1 = \mathit{true}\ \} && (10)\\
G_1 \equiv\ G_0\ \cup \{\ & \mathit{gcd}(z_6, z_3) = z_9, && (11)\\
& z_9 > 1 = \mathit{true}, && (12)\\
& \mathit{quot}(z_6, z_9) = z_7, && (13)\\
& \mathit{quot}(z_3, z_9) = z_8\ \} && (14)\\
G_2 \equiv\ G_0\ \cup \{\ & z_6 = z_7, && (15)\\
& z_3 = z_8\ \} && (16)
\end{aligned}$$

It is then easy to see that in both $(\Sigma(Z), B, \overrightarrow{E} \cup \overrightarrow{G_1})$ and $(\Sigma(Z), B, \overrightarrow{E} \cup \overrightarrow{G_2})$ we have $z_1 * (z_2 / z_3) \to^* z_7 / z_8$ and $z_4 / z_5 \to^* z_7 / z_8$ and therefore $z_1 * (z_2 / z_3) \downarrow z_4 / z_5$.

For $\overrightarrow{G_0}$, since by (10) *the condition $\mathit{gcd}(z_4, z_5) > 1 \to^* \mathit{true}$ is satisfied, we can simplify z_4 / z_5 with the conditional rule* RAT-ACU-02 *and further simplify it as follows:*

$$\frac{z_4}{z_5} \longrightarrow \frac{\mathit{quot}(z_4, \mathit{gcd}(z_5, z_4))}{\mathit{quot}(z_5, \mathit{gcd}(z_5, z_4))} \overset{(07)}{\underset{(08)}{\longrightarrow}} \frac{z_7}{z_8}$$

Moreover, $z_1 * (z_2 / z_3)$ can also in both cases be simplified using rule RAT-ACU-12 *as*

$$z_1 * \frac{z_2}{z_3} \longrightarrow \frac{z_1 * z_2}{z_3} \overset{(06)}{\longrightarrow} \frac{z_6}{z_3}$$

Notice that since for x of sort NzInt *and y of sort* NzNat *we have the inductive theorems*

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \mathit{quot}(x, \mathit{gcd}(x, y)) : \mathit{NzInt}$$

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \mathit{quot}(y, \mathit{gcd}(x, y)) : \mathit{NzNat}$$

we have z_2 : NzInt *and z_3 :* NzNat*.*

The proof for $\overrightarrow{G_1}$ is easy to complete, since the condition $\mathit{gcd}(z_6, z_3) > 1 \to^* \mathit{true}$ is satisfied by the rewrite sequence

$$\mathit{gcd}(z_6, z_3) > 1 \overset{(09)}{\longrightarrow} z_9 > 1 \overset{(12)}{\longrightarrow} \mathit{true}$$

and therefore we can also simplify z_6 / z_3 with the conditional rule RAT-ACU-02 *and further simplify it as follows:*

$$\frac{z_6}{z_3} \longrightarrow \frac{\mathit{quot}(z_6, \mathit{gcd}(z_6, z_3))}{\mathit{quot}(z_3, \mathit{gcd}(z_6, z_3))} \overset{(09)}{\underset{(09)}{\longrightarrow}} \frac{\mathit{quot}(z_6, z_9)}{\mathit{quot}(z_3, z_9)} \overset{(13)}{\underset{(14)}{\longrightarrow}} \frac{z_7}{z_8}$$

as desired. The proof for $\overrightarrow{G_2}$ is trivially completed using (15) *and* (16).

Letting $D \equiv \mathsf{gcd}(x_2, x_1) > \mathsf{1} \to^* \mathsf{true}$ we need to show that

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \forall(Y \cup Z)\ (D^= \wedge A \wedge A_{aux} \Rightarrow G_1 \vee G_2)$$

But to prove this it is enough to show

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \forall(Y \cup Z)\ (D^= \wedge A \wedge A_{aux} \Rightarrow G)$$

where

$$\begin{aligned} G \equiv \{\ & \mathsf{gcd}(z_6, z_3) = z_9, \\ & \mathsf{gcd}(z_4, z_5) > \mathsf{1} = \mathsf{true}, \\ & \mathsf{quot}(z_6, z_9) = z_7, \\ & \mathsf{quot}(z_3, z_9) = z_8\ \}. \end{aligned}$$

This is so because

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models z_9 = \mathsf{1} \vee z_9 > \mathsf{1} = \mathsf{true}$$

is an inductive theorem for z_9, of sort NzNat, provable by variant satisfiability, so that

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models G \Leftrightarrow ((G \wedge z_9 = \mathsf{1}) \vee (G \wedge z_9 > \mathsf{1} = \mathsf{true})) \Rightarrow G_1 \vee G_2.$$

Let us show that G holds assuming $D^=$, A, and A_{aux} do. Interpret x_1, x_2, x_3 as $\mathsf{a}_1, \mathsf{a}_2, \mathsf{a}_3 \in \mathcal{T}_{\Sigma_0/E_0 \cup B_0}$ having sorts NzInt, NzNat, and NzInt, respectively. First of all, note that there is only one possible interpretation for the variables $z_1 \ldots z_9$ as constants $b_1 \ldots b_9$ making the hypothesis $D^= \wedge A \wedge A_{aux}$ true. Since $\mathcal{R}_0$ is convergent, we can take as $\mathcal{T}_{\Sigma_0/E_0 \cup B_0}$ the canonical term algebra $C_{\Sigma_0/\overrightarrow{E_0},B_0}$, and can assume that $\mathsf{a}_1, \mathsf{a}_2, \mathsf{a}_3$ have the form $\mathsf{a}_1 = [t_1 !_{\overrightarrow{E_0},B_0}]_{B_0}$, $\mathsf{a}_2 = [t_2 !_{\overrightarrow{E_0},B_0}]_{B_0}$, and $\mathsf{a}_3 = [t_3 !_{\overrightarrow{E_0},B_0}]_{B_0}$, where $t !_{\overrightarrow{E_0},B_0}$ denotes the $\overrightarrow{E_0}, B_0$-canonical form of term t. Therefore, for the hypothesis to hold we must, for example, have z_1 interpreted as

$$b_1 = [\ t_3\ !_{\overrightarrow{E_0},B_0}\ ]_{B_0},$$

z_2 as

$$b_2 = [\ \mathsf{quot}(t_1, \mathsf{gcd}(t_2, t_1))\ !_{\overrightarrow{E_0},B_0}\ ]_{B_0}$$

and so on for all other variables $z_3 \ldots z_9$, i.e., we just substitute the x_i for the t_i in A and A_{aux} and obtain the B_0-equivalence class of each righthand side's $\overrightarrow{E_0}, B_0$-normal form. We just need to show that under such an interpretation the corresponding interpretation of G holds. By abuse of language, let the arithmetic expressions for the b_i denote the corresponding (B_0-equivalence classes of) $\overrightarrow{E_0}, B_0$-normal forms. From A and A_{aux}, $\mathsf{gcd}(b_6, b_2) = b_9$ trivially holds in $C_{\Sigma_0/\overrightarrow{E_0},B_0}$, and $\mathsf{gcd}(a_1, a_2) > \mathsf{1} = \mathsf{true}$ implies $\mathsf{gcd}(a_1 * a_3, a_2) > \mathsf{1} = \mathsf{true}$. Then, we just need to show that $\mathsf{quot}(b_6, b_9) = b_7$ and $\mathsf{quot}(b_2, b_9) = b_8$ do too using some inductive properties of the initial algebra $\mathcal{T}_{\Sigma_0/E_0 \cup B_0}$, namely, $\mathbb{Z}$.

Let us abbreviate the predicate $\exists k\ m = n * k$ as $n \mid m$, which can be read as n divides to m. If $n \mid m$ and $n \mid m'$ we sometimes write $n \mid m, m'$. Since $\mathsf{gcd}(a_2, a_1) \mid \mathsf{gcd}(a_2, a_1 * a_3)$, then $\mathsf{gcd}(a_2, a_1 * a_3) = c * \mathsf{gcd}(a_2, a_1)$ for some c. But then, since $\mathsf{gcd}(a_2, a_1) \mid a_2, a_1$, and $\mathsf{gcd}(a_2, a_1 * a_3) \mid a_2, a_1 * a_3$, from A and A_{aux} we get:

$$b_6 * \mathsf{gcd}(a_2, a_1) = a_1 * a_3 = b_7 * \mathsf{gcd}(a_2, a_1 * a_3) = b_7 * c * \mathsf{gcd}(a_2, a_1) \tag{a}$$

$$b_3 * \mathsf{gcd}(a_2, a_1) = a_2 = b_8 * \mathsf{gcd}(a_2, a_1 * a_3) = b_8 * c * \mathsf{gcd}(a_2, a_1) \tag{b}$$

And since for x, y, z of sort NzInt *we have the inductive theorem*

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models x * z = y * z \Rightarrow x * y$$

from the above (a) *and* (b) *we get $b_6 = b_7 * c$ and $b_3 = b_8 * c$. But since* gcd(b_7, b_8) = 1 *this means that*

$$b_9 = \mathsf{gcd}(b_6, b_3) = c$$

and therefore that quot(b_6, b_9) $= b_7$ *and* quot(b_3, b_9) $= b_8$*, as desired. We just need to show that*

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models (G \wedge z_9 > 1 = \mathit{true}) \Rightarrow G_1$$
$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models (G \wedge z_9 = 1) \Rightarrow G_2$$

But this is easy to check and is left to the reader.

Before proving Theorem 9 we need an auxiliary lemma.

Lemma 1. *Let $\mathcal{R}$ and $\mathcal{R}_0$ be rewrite theories as in Theorem 9, and let the CCP variables Y, the abstraction variables Z, and the equations G_k, $k \in K$, be as in (2) in Theorem 9. Then, for any $t, t' \in T_{\Sigma(Z)}(Y)$, and each $\overrightarrow{E_0}, B_0$-normalized substitution $\delta \in [Z \longrightarrow T_{\Sigma_0}]$, $\delta = \delta!_{\overrightarrow{E_0},B_0}$, such that $\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \wedge G_k\delta$, if $t\delta =_B t'$ and $t \rightarrow^*_{\overrightarrow{E} \cup \overrightarrow{G_k},B} w$, then there exists a rewrite sequence $t' \rightarrow^*_{\overrightarrow{E},B} w'$ such that $w\delta =_B w'$.*

Proof. By induction on the length n of the rewrite sequence $t \rightarrow^*_{\overrightarrow{E} \cup \overrightarrow{G_k},B} w$. For $n = 0$ the result follows trivially. Assume the result true for rewrite sequences of length smaller or equal than n, and let $t \rightarrow^*_{\overrightarrow{E} \cup \overrightarrow{G_k},B} t_1 \rightarrow_{\overrightarrow{E} \cup \overrightarrow{G_k},B} w$ be a rewrite sequence of length $n + 1$. By the induction hypothesis we then have $t' \rightarrow^*_{\overrightarrow{E},B} t'_1$ such that $t_1\delta =_B t'_1$. Suppose that the last step uses a rule $(l \rightarrow r \; \mathit{if} \; C) \in \overrightarrow{E}$. Although the rewrite $t_1 \rightarrow_{\overrightarrow{E},B} w$ is performed in the theory $(\Sigma(Z), B, \overrightarrow{E})$, where the variables Z are constants, the exact same rewrite step and proof of the condition can be performed in $(\Sigma, B, \overrightarrow{E})$ and therefore we have also a rewrite $t_1\delta \rightarrow_{\overrightarrow{E},B} w\delta$ and, by the strict B-coherence of $(\Sigma, B, \overrightarrow{E})$, a rewrite $t'_1 \rightarrow_{\overrightarrow{E},B} w'$ such that $w\delta =_B w'$, as desired. Suppose instead that we have a rewrite step $t_1 \rightarrow_{\overrightarrow{G_k},B} w$ with a rule $(l \rightarrow r) \in \overrightarrow{G_k}$. Then, since r is either a variable in Z or a Σ_0-ground term in $\overrightarrow{E_0}, B_0$-normal form, $\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models G_k\delta$, and $\mathcal{R}_0$ is ground convergent, by the ground Church-Rosser Theorem we must have $l\delta \rightarrow^*_{\overrightarrow{E_0},B_0} l\delta!_{\overrightarrow{E_0},B_0} =_{B_0} r\delta$. Therefore, there is a rewrite sequence $t_1\delta \rightarrow^*_{\overrightarrow{E_0},B_0} w_1$ with $w_1 =_B w\delta$ and, by strict B-coherence of $\mathcal{R}$, also a rewrite sequence $t'_1 \rightarrow^*_{\overrightarrow{E_0},B_0} w'$ such that $w' =_B w_1 =_B w\delta$, as desired. This finishes the proof of the lemma. □

Let us now prove Theorem 9.

Proof. The proof of *(1)* is easy, because if $\mathcal{R}_0 \models D\theta$ for some ground substitution θ, we obviously have $\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models D^{=}\theta$. Therefore, $\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \neg(\exists D^{=})$ means that there is no ground substitution θ such that $\mathcal{R}_0 \models D\theta$, and therefore the conditional critical pair can be discarded.

Let us now prove *(2)*. That is, assuming that *(2)-(i)* and *(2)-(ii)* hold, we need to prove that the critical pair (†) is ground confluent. Let $\theta \in [Y \longrightarrow T_\Sigma]$ be any ground substitution such that $\mathcal{R} \models D\theta$. We have to show that $u\theta \downarrow_{\overrightarrow{E},B} v\theta$ holds. By the assumptions on D and $\mathcal{R}_0 \subseteq \mathcal{R}$ being a conservative extension this means that $\mathcal{R}_0 \models D\theta$, which obviously implies $\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models D^{=}\theta$. But, since the unique interpretation of the variables Y in $\mathcal{T}_{\Sigma_0/E_0 \cup B_0}$ provided by θ extends in a unique way up to B_0-equivalence to a unique interpretation of the variables Z making $A \wedge A_{aux}$ true in $\mathcal{T}_{\Sigma_0/E_0 \cup B_0}$, we can then choose a $\overrightarrow{E_0}, B_0$-normalized substitution $\delta \in [Z \longrightarrow T_{\Sigma_0}]$, i.e., $\delta = \delta!_{\overrightarrow{E_0},B}$, such that $\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models (A \wedge A_{aux})\ (\theta \uplus \delta)$. Note that, by the Ground Church-Rosser Theorem for $\mathcal{R}_0$, this means that for each $z = t$ in $A \cup A_{aux}$ we must have $t\theta \rightarrow^{*}_{\overrightarrow{E_0},B_0} t\theta!_{\overrightarrow{E_0},B} =_{B_0} \delta(z)$. But then, by *(i)* we must have $\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \bigvee_{k \in K} \wedge G_k\delta$, and therefore there is a $k \in K$ such that $\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \wedge G_k\delta$ holds. But by *(2)-(ii)* we have rewrite sequences

$$\begin{tikzcd}
u[z_6, \ldots, z_n]_{p_1 \ldots p_n} \arrow[dr, "*", "\overrightarrow{E} \cup \overrightarrow{G_k},B"'] & & v[z_{n+1}, \ldots, z_{n+m}]_{q_1 \ldots q_m} \arrow[dl, "*"', "\overrightarrow{E} \cup \overrightarrow{G_k},B"] \\
& w_1 =_B w_2 &
\end{tikzcd}$$

and therefore, since $u[z_6, \ldots, z_n]_{p_1 \ldots p_n}(\theta \uplus \delta) = u\theta[\delta(z_6), \ldots, \delta(z_n)]_{p_1 \ldots p_n}$, and $v[z_{n+1}, \ldots, z_{n+m}]_{q_1 \ldots q_m}(\theta \uplus \delta) = v\theta[\delta(z_{n+1}), \ldots, \delta(z_{n+m})]_{q_1 \ldots q_m}$, rewrite sequences

$$\begin{tikzcd}
u\theta[\delta(z_6), \ldots, \delta(z_n)]_{p_1 \ldots p_n} \arrow[dr, "*", "\overrightarrow{E} \cup \overrightarrow{G_k},B"'] & & v\theta[\delta(z_{n+1}), \ldots, \delta(z_{n+m})]_{q_1 \ldots q_m} \arrow[dl, "*"', "\overrightarrow{E} \cup \overrightarrow{G_k},B"] \\
& w_1(\theta \uplus \delta) =_B w_2(\theta \uplus \delta) &
\end{tikzcd}$$

But note that for each z_i, $1 \leq i \leq n$, z_{n+j}, $1 \leq j \leq n$, since δ is $\overrightarrow{E_0}, B_0$-normalized, we have

$$u\theta\,|_{p_i} \rightarrow^{*}_{\overrightarrow{E_0},B_0} u\theta\,|_{p_i}\ !_{\overrightarrow{E_0},B_0} =_{B_0} \delta(z_i)$$

$$v\theta\,|_{q_j} \rightarrow^{*}_{\overrightarrow{E_0},B_0} v\theta\,|_{q_j}\ !_{\overrightarrow{E_0},B} =_{B_0} \delta(z_{n+j})$$

and therefore

$$u\theta \rightarrow^{*}_{\overrightarrow{E_0},B_0} u' =_{B_0} u\theta[\delta(z_6), \ldots, \delta(z_n)]_{p1 \ldots p_n}$$

$$v\theta \rightarrow^{*}_{\overrightarrow{E_0},B_0} v' =_{B_0} v\theta[\delta(z_{n+1}), \ldots, \delta(z_{n+m})]_{q1 \ldots q_m}$$

We will be done if we show that there are rewrite sequences $u' \rightarrow^{*}_{\overrightarrow{E},B} w'_1$ and $v' \rightarrow^{*}_{\overrightarrow{E},B} w'_2$ such that $w'_1 =_B w_1(\theta \uplus \delta)$ and $w'_2 =_B w_2(\theta \uplus \delta)$. But note that

$$u' = (u[(u|_{p_1}\theta)!_{\overrightarrow{E_0},B_0}, \ldots, (u|_{p_n}\theta)!_{\overrightarrow{E_0},B_0}]_{p_1 \ldots p_n})\theta =_{B_0} (u[z_6, \ldots, z_n]_{p1 \ldots p_n}\delta)\theta,$$

$$v' = (v[(v|_{q_1}\theta)!_{\overrightarrow{E_0},B_0}, \ldots, (v|_{q_m}\theta)!_{\overrightarrow{E_0},B_0}]_{q_1 \ldots q_m})\theta =_{B_0} (v[z_{n+1}, \ldots, z_{n+m}]_{q_1 \ldots q_m}\delta)\theta,$$

$$w_1(\theta \uplus \delta) = (w_1\delta)\theta, \mathit{and}$$

$$w_2(\theta \uplus \delta) = (w_2\delta)\theta.$$

Therefore, by Lemma 1, the ground instance of the CCP satisfying its condition can be joined, as desired. □

6.3. Confluence of the RAT-ACU *module*

If we check the Church-Rosser property of the RAT-ACU module (Appendix E), we get several proof obligations, specifically, 36 conditional critical pairs for confluence and 33 conditional memberships for sort decreasingness. The result given by the tool can be found in [**?** , Appendix G]. Proofs for the membership assertions can be found in [**?** , Appendix I].

In addition to the check for RAT-ACU, [**?** , Appendix G] presents the output of checking each of the modules in the hierarchy. Interestingly, modules BOOL-FVP, NAT-FVP, NAT-ACU, INT-FVP, and INT-ACU are proved locally confluent and sort decreasing. Since they are also terminating, they are convergent, and therefore, we can apply the hierarchical techniques developed in the previous sections for proving the conditional critical pairs for RAT-ACU.

Given the specifications in Appendix E, let $\mathcal{R}_0 = \mathcal{R}_{\mathsf{INT\text{-}ACU}}$ be the rewrite theory associated to INT-ACU, and $\mathcal{R} = \mathcal{R}_{\mathsf{RAT\text{-}ACU}}$ the rewrite theory associated to RAT-ACU. We provide in this section proofs for the joinability or unsatisfiability of a representative selection of the critical pairs output by the CRC tool on input RAT-ACU. These critical pairs were chosen to show different situations found when completing the proofs and most of the techniques in the overall proofs are present in these examples. The detail of all proofs can be found in [**?** , Appendix H].

Given variables I', J' and K' of sort NzInt and N' of sort NzNat we have the following inductive lemmas:

$$\mathcal{T}_{\Sigma_0/E_0\cup B_0} \models I' + \text{ - } N' = \text{ - }(\text{ - } I' + N') \quad \text{(L01)}$$
$$\mathcal{T}_{\Sigma_0/E_0\cup B_0} \models \mathsf{quot}(I', \mathsf{gcd}(I', N')) : \mathsf{NzInt} \quad \text{(L02)}$$
$$\mathcal{T}_{\Sigma_0/E_0\cup B_0} \models \mathsf{gcd}(\text{ - } I', J') = \mathsf{gcd}(I', J') \quad \text{(L03)}$$
$$\mathcal{T}_{\Sigma_0/E_0\cup B_0} \models \mathsf{gcd}(I' + J' * N', N') = \mathsf{gcd}(I', N') \quad \text{(L04)}$$
$$\mathcal{T}_{\Sigma_0/E_0\cup B_0} \models I' = \mathsf{quot}(I', \mathsf{gcd}(I', J')) * \mathsf{gcd}(I', J') \quad \text{(L05)}$$
$$\mathcal{T}_{\Sigma_0/E_0\cup B_0} \models I' * K' = J' * K' \Leftrightarrow I' = J' \quad \text{(L06)}$$
$$\mathcal{T}_{\Sigma_0/E_0\cup B_0} \models \mathsf{quot}(N', \mathsf{gcd}(I', N')) : \mathsf{NzNat} \quad \text{(L07)}$$

In what follows, we let $S'_0 = \{\mathsf{Int}, \mathsf{NzInt}, \mathsf{Nat}, \mathsf{NzNat}\}$.

6.3.1. Conditional critical pair RAT-ACU1000

The critical pair RAT-ACU1000 is the following one:

$$\mathsf{gcd}(x_2, x_1) > 1 \to^* \mathsf{true} \Rightarrow \frac{x_2 + x_1 * x_3}{x_3} \downarrow x_1 + \frac{\mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_3))}{\mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3))}$$

where x_1 and x_3 have sort NzNat, and x_2 has sort NzInt. It comes from the overlap of equations INT-FVP-02 and RAT-ACU-02.

Let us consider the following sets of axioms:

$$A \equiv \{\, z_1 = x_2 + x_1 * x_3, \quad \text{(1000-01)}$$
$$z_2 = x_3, \quad \text{(1000-02)}$$

$$z_3 = x_1, \tag{1000-03}$$
$$z_4 = \mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_3)), \tag{1000-04}$$
$$z_5 = \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3)) \}, \tag{1000-05}$$
$$A_{aux} \equiv \{\, z_6 = \mathsf{0}, \tag{1000-06}$$
$$z_7 = x_1 * \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3)) + \mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_3)), \tag{1000-07}$$
$$z_8 = \mathsf{gcd}(x_2 + x_1 * x_3, x_3), \tag{1000-08}$$
$$z_9 = \mathsf{quot}(x_2 + x_1 * x_3, \mathsf{gcd}(x_2 + x_1 * x_3, x_3)), \tag{1000-09}$$
$$z_{10} = \mathsf{quot}(x_3, \mathsf{gcd}(x_2 + x_1 * x_3, x_3)) \tag{1000-10}$$
$$z_{11} = \mathsf{gcd}(x_1 * \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3)) + \mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_3)), \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3))) \tag{1000-11}$$
$$z_{12} = \mathsf{quot}(x_1 * \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3)) + \mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_3)), \mathsf{gcd}(x_1 * \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3)) + \mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_3)), \mathsf{qu} \tag{1000-12}$$
$$z_{13} = \mathsf{quot}(\mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3)), \mathsf{gcd}(x_1 * \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3)) + \mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_3)), \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3)))) \tag{1000-13}$$
$$G^1_{aux} \equiv \{\, z_3 * z_5 + z_4 = z_7 \,\} \tag{1000-14}$$
$$G_1 \equiv G^1_{aux} \cup$$
$$\{\, z_1 = \mathsf{0}, \tag{1000-15}$$
$$z_7 = \mathsf{0} \,\} \tag{1000-16}$$
$$G^2_{aux} \equiv G^1_{aux} \cup$$
$$\{\, z_1 \neq \mathsf{0}, \tag{1000-17}$$
$$z_9 \neq \mathsf{0} \,\} \tag{1000-18}$$
$$G^3_{aux} \equiv G^2_{aux} \cup$$
$$\{\, \mathsf{gcd}(z_1, z_2) = z_8, \tag{1000-19}$$
$$z_8 > \mathsf{1} = \mathsf{true}, \tag{1000-20}$$
$$\mathsf{quot}(z_1, z_8) = z_9, \tag{1000-21}$$
$$\mathsf{quot}(z_2, z_8) = z_{10} \,\} \tag{1000-22}$$
$$G^4_{aux} \equiv G^2_{aux} \cup$$
$$\{\, \mathsf{gcd}(z_7, z_5) = z_{11}, \tag{1000-23}$$
$$z_{11} > \mathsf{1} = \mathsf{true}, \tag{1000-24}$$
$$\mathsf{quot}(z_7, z_{11}) = z_{12}, \tag{1000-25}$$
$$\mathsf{quot}(z_5, z_{11}) = z_{13} \,\} \tag{1000-26}$$
$$G_2 \equiv G^3_{aux} \cup$$
$$\{\, z_9 = z_7, \tag{1000-27}$$
$$z_{10} = z_5 \,\} \tag{1000-28}$$
$$G_3 \equiv G^4_{aux} \cup$$
$$\{\, z_{12} = z_1, \tag{1000-29}$$
$$z_{13} = z_2 \,\} \tag{1000-30}$$
$$G_4 \equiv \{\, z_9 = z_{12}, \tag{1000-31}$$
$$z_{10} = z_{13} \,\} \tag{1000-32}$$

It is then easy to see that in all $(\Sigma(Z), B, \overrightarrow{E} \cup \overrightarrow{G_i})$, $i = 1 \cdots 4$, we have $z_1 \,/\, z_2 \downarrow z_3 + z_4 \,/\, z_5$.

In $\overrightarrow{G^1_{aux}}$, by (1000-03), (1000-04), and (1000-05), we can simplify $z_3 + z_4 \,/\, z_5$ with the rule RAT-ACU-09 as follows:

$$z_3 + \frac{z_4}{z_5} \xrightarrow{\mathsf{RAT\text{-}ACU\text{-}09}} \frac{z_3 * z_5 + z_4}{z_5} \xrightarrow{(1000\text{-}07)} \frac{z_7}{z_5}$$

Notice that by Lemmas (L02) and (L07) we have $z_4 : \mathsf{NzInt}$ and $z_5 : \mathsf{NzNat}$.

We can complete the proof for $\overrightarrow{G_1}$ with the following simplification steps leading both $z_1 \mathbin{/} z_2$ and $z_7 \mathbin{/} z_5$ to z_6:

$$\frac{z_1}{z_2} \xrightarrow{\text{(1000-15)}} \frac{\mathsf{0}}{z_2} \xrightarrow{\text{(1000-06)}} \frac{z_6}{z_2} \xrightarrow{\textsf{RAT-ACU-01}} z_6 \xleftarrow{\textsf{RAT-ACU-01}} \frac{z_6}{z_5} \xleftarrow{\text{(1000-06)}} \frac{\mathsf{0}}{z_5} \xleftarrow{\text{(1000-16)}} \frac{z_7}{z_5}$$

In $\overrightarrow{G^3_{aux}}$, by (1000-19) and (1000-20), the condition $\mathsf{gcd}(z_1, z_2) > \mathsf{1} \to^* \mathsf{true}$ is satisfied, and therefore we can simplify $z_1 \mathbin{/} z_2$ with the conditional rule RAT-ACU-02 and further simplify it as follows:

$$\frac{z_1}{z_2} \xrightarrow{\textsf{RAT-ACU-02}} \frac{\mathsf{quot}(z_1, \mathsf{gcd}(z_1, z_2))}{\mathsf{quot}(z_2, \mathsf{gcd}(z_1, z_2))} \xrightarrow[\text{(1000-08)}]{\text{(1000-08)}} \frac{\mathsf{quot}(z_1, z_8)}{\mathsf{quot}(z_2, z_8)} \xrightarrow[\text{(1000-22)}]{\text{(1000-21)}} \frac{z_9}{z_{10}}$$

In $\overrightarrow{G^4_{aux}}$, by (1000-23) and (1000-24), the condition $\mathsf{gcd}(z_7, z_5) > \mathsf{1} \to^* \mathsf{true}$ is satisfied, and therefore we can simplify $z_7 \mathbin{/} z_5$ with the conditional rule RAT-ACU-02 and further simplify it as follows:

$$\frac{z_7}{z_5} \xrightarrow{\textsf{RAT-ACU-02}} \frac{\mathsf{quot}(z_7, \mathsf{gcd}(z_5, z_7))}{\mathsf{quot}(z_5, \mathsf{gcd}(z_5, z_7))} \xrightarrow[\text{(1000-11)}]{\text{(1000-11)}} \frac{\mathsf{quot}(z_7, z_{11})}{\mathsf{quot}(z_5, z_{11})} \xrightarrow[\text{(1000-26)}]{\text{(1000-25)}} \frac{z_{12}}{z_{13}}$$

The proof is then trivially completed for $\overrightarrow{G_i}$, $i = 2 \ldots 4$.

Letting $D \equiv \mathsf{gcd}(x_2, x_3) > \mathsf{1} \to^* \mathsf{true}$ we need to show that

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \forall(Y \cup Z)\ (D^= \wedge A \wedge A_{aux} \Rightarrow G_1 \vee G_2 \vee G_3 \vee G_4)$$

But to prove this it is enough to show

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models \forall(Y \cup Z)\ (D^= \wedge A \wedge A_{aux} \Rightarrow G)$$

where

$$\begin{aligned} G \equiv \{\ & z_3 * z_5 + z_4 = z_7, \\ & \mathsf{gcd}(z_1, z_2) = z_8, \\ & \mathsf{gcd}(z_7, z_5) = z_{11}, \\ & \mathsf{quot}(z_1, z_8) = \mathsf{quot}(z_7, z_{11}), \\ & \mathsf{quot}(z_2, z_8) = \mathsf{quot}(z_5, z_{11})\ \} \end{aligned}$$

This is so because

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models I = \mathsf{0} \vee I \neq \mathsf{0}$$

$$\mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models N' = \mathsf{1} \vee N' > \mathsf{1} = \mathsf{true}$$

are inductive theorems for N' of sort NzNat and I of sort Int, provable by variant satisfiability, so that

$$\begin{aligned} \mathcal{T}_{\Sigma_0/E_0 \cup B_0} \models G \Leftrightarrow G \wedge (&(z_1 = \mathsf{0} \vee z_1 \neq \mathsf{0}) \\ &\wedge (z_8 = 1 \vee z_8 > \mathsf{1} = \mathsf{true}) \\ &\wedge (z_{11} = 1 \vee z_{11} > \mathsf{1} = \mathsf{true})) \Rightarrow G_1 \vee G_2 \vee G_3 \vee G_4. \end{aligned}$$

Let us show that G holds assuming $D^=$, A, and A_{aux}. Interpret $x_1 \ldots x_3$ as $\mathsf{a}_1 \ldots \mathsf{a}_3 \in \mathcal{T}_{\Sigma_0/E_0 \cup B_0}$ having sorts NzInt, NzInt, and NzNat, respectively. Let constants $b_1 \ldots b_{10}$ be the unique interpretation for the variables $z_1 \ldots z_{10}$ making the hypothesis $D^= \wedge A \wedge A_{aux}$ true. Let $\mathcal{T}_{\Sigma_0/E_0 \cup B_0}$ be the canonical term algebra $C_{\Sigma_0/\overrightarrow{E_0},B_0}$, and $\mathsf{a}_1 \ldots \mathsf{a}_3$ have the form $\mathsf{a}_i = [t_i!_{\overrightarrow{E_0},B_0}]_{B_0}$, where $t_i!_{\overrightarrow{E_0},B_0}$ denotes the $\overrightarrow{E_0}, B_0$-canonical form of term t_i. Let us interpret variables z_i as values b_i by substituting the x_i for the t_i in A and A_{aux} and obtain the B_0-equivalence class of each righthand side's $\overrightarrow{E_0}, B_0$-normal form. Let us now show under such interpretation that the corresponding interpretation of G holds.

First, notice that $b_1 = \mathsf{0} \Leftrightarrow b_7 = \mathsf{0}$ is a consequence of the following sequence of equivalences:

$$\begin{aligned}
& z_1 = 0 \\
\Leftrightarrow\ & x_2 + x_1 * x_3 = 0 \\
\Leftrightarrow\ & z_4 * \mathsf{gcd}(x_2, x_3) + x_1 * z_5 * \mathsf{gcd}(x_2, x_3) = 0 \\
\Leftrightarrow\ & (z_4 + x_1 * z_5) * \mathsf{gcd}(x_2, x_3) = 0 \\
\Leftrightarrow\ & z_4 + x_1 * z_5 = 0 \\
\Leftrightarrow\ & z_7 = 0
\end{aligned}$$

$b_3 * b_5 + b_4 = b_7$, $\mathsf{gcd}(b_1, b_2) = b_8$, and $\mathsf{gcd}(b_7, b_5) = b_{11}$ trivially hold. From A and A_{aux}, given Lemmas (L04) and (L05), we have:

$$\begin{aligned}
z_9 * \mathsf{gcd}(x_2, x_3) &= \mathsf{quot}(z_1, z_8) * \mathsf{gcd}(x_2, x_3) \\
&= \mathsf{quot}(x_2 + x_1 * x_3, \mathsf{gcd}(x_2 + x_1 * x_3, x_3)) * \mathsf{gcd}(x_2, x_3) \\
&= \mathsf{quot}(x_2 + x_1 * x_3, \mathsf{gcd}(x_2 + x_1 * x_3, x_3)) * \mathsf{gcd}(x_2 + x_1 * x_3, x_3) \\
&= x_2 + x_1 * x_3 \\
&= \mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_3)) * \mathsf{gcd}(x_2, x_3) + x_1 * \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3)) * \mathsf{gcd}(x_2, x_3) \\
&= (\mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_3)) + x_1 * \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3))) * \mathsf{gcd}(x_2, x_3) \\
&= (z_4 + z_3 * z_5) * \mathsf{gcd}(x_2, x_3) \\
&= z_7 * \mathsf{gcd}(x_2, x_3) \\
&= \mathsf{quot}(z_7, z_{11}) * z_{11} * \mathsf{gcd}(x_2, x_3) \\
&= z_{12} * \mathsf{gcd}(x_2, x_3) \\
z_{10} * \mathsf{gcd}(x_2, x_3) &= \mathsf{quot}(z_2, z_8) * \mathsf{gcd}(x_2, x_3) \\
&= \mathsf{quot}(x_3, \mathsf{gcd}(x_2 + x_1 * x_3, x_3)) * \mathsf{gcd}(x_2, x_3) \\
&= \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3)) * \mathsf{gcd}(x_2, x_3) \\
&= x_3 \\
&= \mathsf{quot}(x_3, \mathsf{gcd}(x_2, x_3)) * \mathsf{gcd}(x_2, x_3) \\
&= z_5 * \mathsf{gcd}(x_2, x_3) \\
&= \mathsf{quot}(z_5, z_{11}) * z_{11} * \mathsf{gcd}(x_2, x_3) \\
&= z_{13} * \mathsf{gcd}(x_2, x_3)
\end{aligned}$$

From the above, by Lemma (L06), we get $b_9 = b_{12}$ and $b_{10} = b_{13}$, as desired.

6.3.2. Conditional critical pair RAT-ACU1530

The critical pair RAT-ACU1530 is the following one:

$$\mathsf{gcd}(1, x_1) > 1 \rightarrow^* \mathsf{true} \Rightarrow x_1 + x_2 \downarrow x_2 + \frac{\mathsf{quot}(x_1, \mathsf{gcd}(1, x_1))}{\mathsf{quot}(1, \mathsf{gcd}(1, x_1))}$$

where x_1 have sort NzInt, and x_2 has sort [Rat]. It comes from the overlap of equations RAT-ACU-03 and RAT-ACU-02.

Our first observation is that the condition of the critical pair is a Σ_0-condition whose variables have all sorts in S'_0. Since such condition is unsatisfiable, the critical pair may be discarded.

6.3.3. Conditional critical pair RAT-ACU3326

The critical pair RAT-ACU3326 is the following one:

$$\mathsf{gcd}(x_2, x_1) > 1 \rightarrow^* \mathsf{true} \Rightarrow \frac{x_1 + \text{ - } x_2}{x_2} \downarrow \text{ - } 1 + \frac{\mathsf{quot}(x_1, \mathsf{gcd}(x_1, x_2))}{\mathsf{quot}(x_2, \mathsf{gcd}(x_1, x_2))}$$

where x_1 has sort NzInt, and x_2 has sort NzNat. It comes from the overlap of equations INT-FVP-05 and RAT-ACU-02.

Let us consider the following sets of axioms:

$A \equiv \{\ z_1 = x_1 + \text{ - } x_2,$ (3326-01)

$$
\begin{align}
& z_2 = x_2, \tag{3326-02}\\
& z_3 = \text{ - }1, \tag{3326-03}\\
& z_4 = \mathsf{quot}(x_1, \mathsf{gcd}(x_1, x_2)), \tag{3326-04}\\
& z_5 = \mathsf{quot}(x_2, \mathsf{gcd}(x_1, x_2)) \,\} \tag{3326-05}\\
A_{aux} \equiv \{\; & z_6 = \mathsf{gcd}(x_1 + \text{ - } x_2, x_2), \tag{3326-06}\\
& z_7 = \mathsf{quot}(x_1 + \text{ - } x_2, \mathsf{gcd}(x_1 + \text{ - } x_2, x_2)), \tag{3326-07}\\
& z_8 = \mathsf{quot}(x_2, \mathsf{gcd}(x_1 + \text{ - } x_2, x_2)), \tag{3326-08}\\
& z_9 = \mathsf{quot}(x_1 + \text{ - } x_2, \mathsf{gcd}(\text{ - } x_1 + x_2, x_2)), \tag{3326-09}\\
& z_{10} = \mathsf{quot}(x_2, \mathsf{gcd}(\text{ - } x_1 + x_2, x_2)), \tag{3326-10}\\
& z_{11} = \mathsf{quot}(x_1 + \text{ - } x_2, \mathsf{gcd}(\text{ - } x_1, x_2)), \tag{3326-11}\\
& z_{12} = \mathsf{quot}(x_2, \mathsf{gcd}(\text{ - } x_1, x_2)), \tag{3326-12}\\
& z_{13} = \text{ - }1 * \mathsf{quot}(x_2, \mathsf{gcd}(x_1, x_2)) + \mathsf{quot}(x_1, \mathsf{gcd}(x_1, x_2)), \tag{3326-13}\\
& z_{14} = \text{ - }\mathsf{quot}(x_2, \mathsf{gcd}(x_1, x_2)) + \mathsf{quot}(x_1, \mathsf{gcd}(x_1, x_2)), \tag{3326-14}\\
& z_{15} = \mathsf{quot}(\text{ - } x_2, \mathsf{gcd}(x_1, x_2)) + \mathsf{quot}(x_1, \mathsf{gcd}(x_1, x_2)), \tag{3326-15}\\
& z_{16} = \mathsf{quot}(\text{ - } x_2 + x_1, \mathsf{gcd}(x_1, x_2)) \,\} \tag{3326-16}\\
G \equiv \{\; & z_6 > 1 = \mathsf{true}, \tag{3326-17}\\
& z_7 = z_9, \tag{3326-18}\\
& z_8 = z_{10}, \tag{3326-19}\\
& z_9 = z_{11}, \tag{3326-20}\\
& z_{10} = z_{12}, \tag{3326-21}\\
& z_{11} = z_{16}, \tag{3326-22}\\
& z_{12} = z_5, \tag{3326-23}\\
& z_3 * z_5 + z_4 = z_{13} \tag{3326-24}\\
& z_{13} = z_{14}, \tag{3326-25}\\
& z_{14} = z_{15}, \tag{3326-26}\\
& z_{15} = z_{16} \,\} \tag{3326-27}
\end{align}
$$

It is then easy to see that in $(\Sigma(Z), B, \overrightarrow{E} \cup \overrightarrow{G})$ we have $z_1 \mathbin{/} z_2 \rightarrow^* z_{16} \mathbin{/} z_5$ and $z_3 + z_4 \mathbin{/} z_5 \rightarrow^* z_{16} \mathbin{/} z_5$ and therefore $z_1 \mathbin{/} z_2 \downarrow z_3 + z_4 \mathbin{/} z_5$.

Since by (3326-06) and (3326-17) the condition $\mathsf{gcd}(z_1, z_2) > 1 \rightarrow^* \mathsf{true}$ is satisfied, we can simplify $z_1 \mathbin{/} z_2$ with the conditional rule RAT-ACU-02 and further simplify it as follows:

$$\frac{z_1}{z_2} \longrightarrow \frac{\mathsf{quot}(z_1, \mathsf{gcd}(z_1, z_2))}{\mathsf{quot}(z_2, \mathsf{gcd}(z_1, z_2))} \xrightarrow[(3326\text{-}08)]{(3326\text{-}07)} \frac{z_7}{z_8} \xrightarrow[(3326\text{-}19)]{(3326\text{-}18)} \frac{z_9}{z_{10}} \xrightarrow[(3326\text{-}21)]{(3326\text{-}20)} \frac{z_{11}}{z_{12}} \xrightarrow[(3326\text{-}23)]{(3326\text{-}22)} \frac{z_{16}}{z_5}$$

$z_3 + z_4 \mathbin{/} z_5$ can also be simplified using rule RAT-ACU-09 and further simplify it as

$$z_3 + \frac{z_4}{z_5} \longrightarrow \frac{z_3 * z_5 + z_4}{z_5} \xrightarrow{(3326\text{-}24)} \frac{z_{13}}{z_5} \xrightarrow{(3326\text{-}25)} \frac{z_{14}}{z_5} \xrightarrow{(3326\text{-}26)} \frac{z_{15}}{z_5} \xrightarrow{(3326\text{-}27)} \frac{z_{16}}{z_5}$$

Notice that by Lemmas (L02) and (L07) we have $z_4 : \mathsf{NzInt}$ and $z_5 : \mathsf{NzNat}$.

Let us show that G holds assuming $D^=$, A, and A_{aux} do. Interpret x_1, x_2 as $\mathsf{a}_1, \mathsf{a}_2 \in \mathcal{T}_{\Sigma_0/E_0 \cup B_0}$ having sorts NzInt and NzNat, respectively. Let constants $b_1 \ldots b_{16}$ be the unique interpretation for the variables $z_1 \ldots z_{16}$ making the hypothesis $D^= \wedge A \wedge A_{aux}$ true. Let $\mathcal{T}_{\Sigma_0/E_0 \cup B_0}$ be the canonical term algebra $C_{\Sigma_0/\overrightarrow{E_0}, B_0}$, and $\mathsf{a}_1, \mathsf{a}_2$ have the form $\mathsf{a}_i = [t_i!_{\overrightarrow{E_0}, B_0}]_{B_0}$, where $t_i!_{\overrightarrow{E_0}, B_0}$ denotes the $\overrightarrow{E_0}, B_0$-canonical form of term t_i. Let us interpret variables z_i as values b_i by substituting the x_i for the t_i in A and A_{aux} and obtain the B_0-equivalence class of each righthand side's $\overrightarrow{E_0}, B_0$-normal form. Let us now show under such interpretation that the corresponding interpretation of G holds.

We have the following rewrite sequence

$$\begin{aligned}\mathsf{gcd}(x_1, x_2) > 1 = \mathsf{true} &\xrightarrow{\text{(L03)}} \mathsf{gcd}(\text{ - } x_1, x_2) > 1 = \mathsf{true}\\ &\xrightarrow{\text{(L04)}} \mathsf{gcd}(\text{ - } x_1 + x_2, x_2) > 1 = \mathsf{true}\\ &\xrightarrow{\text{(L01)}} \mathsf{gcd}(\text{ - } (\text{ - } x_1 + x_2), x_2) > 1 = \mathsf{true}\\ &\xrightarrow{\text{(L03)}} \mathsf{gcd}(x_1 + \text{ - } x_2, x_2) > 1 = \mathsf{true}\end{aligned}$$

and therefore $b_6 > 1$. (3326-18)...(3326-23) and (3326-25)...(3326-27) are also consequences of Lemmas (L01), (L03), and (L04). Finally, $b_3 * b_5 + b_4 = b_{13}$ trivially holds.

6.3.4. Conditional critical pair RAT-ACU5555

The critical pair RAT-ACU5555 is:

$$\mathsf{gcd}(x_2, x_1) > 1 \to^* \mathsf{true} \Rightarrow x_3 * x_4 * \frac{\mathsf{quot}(x_1, \mathsf{gcd}(x_2, x_1))}{\mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_1))} \downarrow x_4 * \frac{x_1 * x_3}{x_2}$$

where x_1 and x_3 have sort NzInt, x_2 has sort NzNat, and x_4 has sort [Rat]. It comes from the overlap of equations RAT-ACU-12 and RAT-ACU-02.

Although the inference rules for proving joinability in Section 4 were introduced for the non-conditional case, the context Ctx rule can be lifted to a restricted form in a conditional setting that is useful for the example at hand.

$$\frac{\mathcal{R} \Vdash D \Rightarrow (\forall X)\, t \downarrow u}{\mathcal{R} \Vdash D \Rightarrow (\forall X)\, C[t] \downarrow C[u]}\ \text{SC-Ctx}$$

A proof of the joinability of the critical pair may be provided using the same techniques as for the previous critical pairs of the RAT-ACU specification. However, this is a case of joinability under a context; the conditional critical pair RAT-ACU5552 was proven joinable in Example 2.

$$\mathsf{gcd}(x_2, x_1) > 1 \to^* \mathsf{true} \Rightarrow x_3 * \frac{\mathsf{quot}(x_1, \mathsf{gcd}(x_2, x_1))}{\mathsf{quot}(x_2, \mathsf{gcd}(x_2, x_1))} \downarrow \frac{x_1 * x_3}{x_2}$$

7. Related Work and Conclusion

In [3], A. Bouhoula proposes an inference system for simultaneously checking the sufficient completeness and ground confluence of constructor-based equational specifications. His approach computes a pattern tree for every defined symbol and identifies a set of proof obligations whose inductive validity has to be checked: if they all are inductive theorems, then the specification is both sufficiently complete and ground confluent; otherwise, it outputs a counterexample. The main difference between the two approaches is that the one presented in this paper can handle both conditional specifications and reasoning modulo axioms, while [3] does not support reasoning modulo axioms. More recently, Nakamura et al. [29] propose incremental techniques for proving termination, confluence, and sufficient completeness of OBJ specifications. Their inference system is also based on the notion of constructor subsignatures, handles conditional equations, and provides sufficient conditions for ensuring such an incremental extension in a modular way. However, for ground confluence, their method has been developed for extensions that preserve the set of critical pairs relative to the extended specification.

Different tools and techniques have been proposed for proving and disproving confluence. Tools such as CSI [28] or ACP [1] are automatic confluence provers for first-order rewrite systems. These tools implement different criteria for proving both confluence and non-confluence.

This work has addressed a thorny and important problem in reasoning about equational programs and algebraic specifications with an initial algebra semantics: the fact that in practice a substantial number of such programs and specifications are *perfectly reasonable* and there is nothing wrong with them, yet they are not locally confluent and therefore fall outside the scope of the standard methods to prove them ground convergent. As the HF-SETS example has shown, it would be quite mistaken to assume that, since our program is perfectly reasonable, we should be able to complete it in some Knuth-Bendix-like fashion. This need not be the case since, as HF-SETS has shown, we can hit a non-orientability "wall" that cannot be surpassed by standard completion methods.

We have proposed a general methodology to help verify the ground convergence of a given equational program based on the synergistic combination of three methods, called Methods 1–3. Method 1, going back to [14], uses unjoinable critical pairs as *hints* for transforming the original specification by adding new rules suggested by such critical pairs to try to make the specification locally confluent or to, at least, reduce the number of critical pairs. Method 2 uses *inductive joinability* proof methods to show the remaining critical pairs ground joinable. Furthermore, using the same inductive joinability proof techniques, Method 2 can prove that the *original* specification was already ground convergent and that its initial algebra semantics has been preserved by its subsequent extensions using Method 1. Method 3 is *hierarchical* in nature: it can be used to prove the ground local confluence of a *conditional* equational specification whose conditions belong to a *subspecification* that has already been proved ground confluent and operationally terminating and that is conservatively extended by the overall specification in an appropriate sense. These methods apply to a very general class of functional programs, namely, to operationally terminating conditional order-sorted specifications modulo axioms such as associativity and/or commutativity and/or identity. In particular, any operationally terminating order-sorted functional module in Maude can benefit from these methods. The HF-SETS and RAT-ACU programs show that Methods 1–3 can be effective in proving the ground confluence of highly non-trivial functional programs. In particular, the proof of ground confluence of an order-sorted specification of the rationals had remained open for decades.

Future work suggested by this work includes: (i) mechanization of the inductive joinability inference system of Method 2 in its most general form, i.e., for conditional specification; (ii) mechanization of Method 3; (iii) integration of Methods 2–3 within the Maude Formal Environment; this integration will be important both to facilitate the use of the methods and to discharge associated proof obligations such as standard joinability of critical pairs, proofs of operational termination, and inductive proofs of equational properties; (iv) further experimentation with these methods on a rich collection of examples; and (v) development of additional proof techniques extending or complementing those presented here as suggested by further experiments.

Acknowledgments. The first author was partially supported by Spanish MINECO/FEDER project TIN2014-52034-R and Univ. Málaga, Campus de Excelencia Internacional Andalucía Tech. The second author was partially supported by NRL under contract number N00173-17-1-G002. The third author was partially supported by CAPES, Colciencias, and INRIA via the STIC AmSud project "EPIC: EPistemic Interactive Concurrency" (Proc. No 88881.117603/2016-01).

References

[1] Aoto, T., Yoshida, J., Toyama, Y.. Proving confluence of term rewriting systems automatically. In: Treinen, R., editor. Rewriting Techniques and Applications, 20th International Conference, RTA. Springer; volume 5595 of *Lecture Notes in Computer Science*; 2009. p. 93–102.

[2] Avenhaus, J., Loría-Sáenz, C.. On conditional rewrite systems with extra variables and deterministic logic programs. In: Pfenning, F., editor. Logic Programming and Automated Reasoning, 5th International Conference, LPAR 1994, Proceedings. Springer; volume 822 of *Lecture Notes in Computer Science*; 1994. p. 215–229.

[3] Bouhoula, A.. Simultaneous checking of completeness and ground confluence for algebraic specifications. ACM Transactions on Computational Logic 2009;10(3):1–33. doi:10.1145/1507244.1507250.

[4] Bruni, R., Meseguer, J.. Semantic foundations for generalized rewrite theories. Theoretical Computer Science 2006;360(1-3):386–414. doi:10.1016/j.tcs.2006.04.012.

[5] Clavel, M., Durán, F., Eker, S., Lincoln, P., Martí-Oliet, N., Meseguer, J., Talcott, C.L.. All About Maude - A High-Performance Logical Framework, How to Specify, Program and Verify Systems in Rewriting Logic. volume 4350 of *Lecture Notes in Computer Science*. Springer, 2007. doi:10.1007/978-3-540-71999-1.

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

[7] Cohen, P.. Set Theory and the Continuum Hypothesis. W.A. Benjamin, 1966.

[8] Comon-Lundh, H., Delaune, S.. The finite variant property: How to get rid of some algebraic properties. In: Giesl, J., editor. Term Rewriting and Applications, 16th International Conference, RTA. Springer; volume 3467 of *Lecture Notes in Computer Science*; 2005. p. 294–307. URL: https://doi.org/10.1007/978-3-540-32033-3_22. doi:10.1007/978-3-540-32033-3_22.

[9] Dershowitz, N., Jouannaud, J.P.. Rewrite systems. In: van Leeuwen, J., editor. Handbook of Theoretical Computer Science, Vol. B. North-Holland; 1990. p. 243–320.

[10] Durán, F., Lucas, S., Marché, C., Meseguer, J., Urbain, X.. Proving operational termination of membership equational programs. Higher-Order and Symbolic Computation 2008;21(1-2):59–88. doi:10.1007/s10990-008-9028-2.

[11] Durán, F., Lucas, S., Meseguer, J.. MTT: the maude termination tool (system description). In: Armando, A., Baumgartner, P., Dowek, G., editors. Automated Reasoning, 4th International Joint Conference, IJCAR. Springer; volume 5195 of *Lecture Notes in Computer Science*; 2008. p. 313–319. URL: https://doi.org/10.1007/978-3-540-71070-7_27. doi:10.1007/978-3-540-71070-7_27.

[12] Durán, F., Lucas, S., Meseguer, J.. Termination modulo combinations of equational theories. In: Ghilardi, S., Sebastiani, R., editors. Frontiers of Combining Systems, 7th International Symposium, FroCoS. Springer; volume 5749 of *Lecture Notes in Computer Science*; 2009. p. 246–262.

[13] Durán, F., Meseguer, J.. A Church-Rosser checker tool for conditional order-sorted equational Maude specifications (long version); 2010. Available at http://maude.lcc.uma.es/CRChC.

[14] Durán, F., Meseguer, J.. On the Church-Rosser and coherence properties of conditional order-sorted rewrite theories. Journal of Logic and Algebraic Programming 2012;81(7-8):816–850.

[15] Durán, F., Meseguer, J., Rocha, C.. Proving ground confluence of equational specifications modulo axioms. In: Rusu, V., editor. Rewriting Logic and Its Applications - 12th International Workshop, WRLA. Springer; volume 11152 of *Lecture Notes in Computer Science*; 2018. p. 184–204. URL: https://doi.org/10.1007/978-3-319-99840-4_11. doi:10.1007/978-3-319-99840-4_11.

[16] Durán, F., Meseguer, J., Rocha, C.. Proving Ground Confluence of Equational Specifications Modulo Axioms. Technical Report 2142/99548; University of Illinois; Urbana, USA; 2018.

[17] Durán, F., Rocha, C., Álvarez, J.M.. Towards a Maude Formal Environment. In: Agha, G., Danvy, O., Meseguer, J., editors. Formal Modeling: Actors, Open Systems, Biological Systems. Springer; volume 7000 of *Lecture Notes in Computer Science*; 2011. p. 329–351. doi:10.1007/978-3-642-24933-4_17.

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

[19] Futatsugi, K., Diaconescu, R.. CafeOBJ Report. World Scientific, 1998.

[20] Goguen, J., Winkler, T., Meseguer, J., Futatsugi, K., Jouannaud, J.P.. Introducing OBJ. In: Software Engineering with OBJ: Algebraic Specification in Action. Kluwer; 2000. p. 3–167.

[21] Goguen, J.A., Meseguer, J.. Order-sorted algebra I: Equational deduction for multiple inheritance, overloading, exceptions and partial operations. Theoretical Computer Science 1992;105(2):217–273. doi:10.1016/0304-3975(92)90302-V.

[22] Hendrix, J., Meseguer, J., Ohsaki, H.. A sufficient completeness checker for linear order-sorted specifications modulo axioms. In: Automated Reasoning. Springer; 2006. p. 151–155.

[23] Hrbacek, K., Jech, T.J.. Introduction to Set Theory. 3rd ed.; volume 45 of *Monographs and textbooks in pure and applied mathematics*. M. Dekker, 1999.

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

[25] Meseguer, J.. Membership algebra as a logical framework for equational specification. In: Goos, G., Hartmanis, J., Leeuwen, J., Presicce, F.P., editors. Recent Trends in Algebraic Development Techniques. Springer; volume 1376; 1998. p. 18–61. doi:10.1007/3-540-64299-4_26.

[26] Meseguer, J.. Strict coherence of conditional rewriting modulo axioms. Theoretical Computer Science 2017;672:1–35.

[27] Meseguer, J.. Variant-based satisfiability in initial algebras. Sci Comput Program 2018;154:3–41.

[28] Nagele, J., Felgenhauer, B., Middeldorp, A.. CSI: new evidence - A progress report. In: de Moura, L., editor. Automated Deduction, 26th International Conference on Automated Deduction, CADE. Springer; volume 10395 of *Lecture Notes in Computer Science*; 2017. p. 385–397.

[29] Nakamura, M., Ogata, K., Futatsugi, K.. Incremental Proofs of Termination, Confluence and Sufficient Completeness of OBJ Specifications. In: Iida, S., Meseguer, J., Ogata, K., editors. Specification, Algebra, and Software. Springer; volume 8373 of *Lecture Notes in Computer Science*; 2014. p. 92–109. doi:10.1007/978-3-642-54624-2_5.

[30] Rocha, C., Meseguer, J.. Constructors, Sufficient Completeness, and Deadlock Freedom of Rewrite Theories. In: Fermüller, C.G., Voronkov, A., editors. Logic for Programming, Artificial Intelligence, and Reasoning - 17th International Conference, LPAR-17. Springer; volume 6397 of *Lecture Notes in Computer Science*; 2010. p. 594–609. doi:10.1007/978-3-642-16242-8_42.

Appendix A. Checking $\mathcal{R}_{\mathcal{E}} \vdash (\forall X)\, t \downarrow u$

Let $\mathcal{R}_{\mathcal{E}} = (\Sigma, B, \overrightarrow{E})$ with $\Sigma = (S, \leq, F)$ be the rewrite theory obtained from $\mathcal{E} = (\Sigma, E \uplus B)$, and let $\mathcal{R}_{\mathcal{E}}^{\approx} = (\Sigma^{\approx}, B, \overrightarrow{E}^{\approx})$ extend $\mathcal{R}_{\mathcal{E}}$ by:

1. extending $(S, \leq)$ to $(S^{\approx}, \leq^{\approx})$ by adding to each connected component $[s] \in S/{\equiv_{\leq}}$ a top sort $[s]$ with $s' \leq [s]$ for each $s' \in [s]$;
2. adding a fresh new sort *Prop* with constant *tt*;
3. adding for each $[s] \in S/{\equiv_{\leq}}$ an operator
$$\_ \approx \_ : [s]\ [s] \longrightarrow Prop$$
4. adding to $\overrightarrow{E}$ the rules
$$\{x : [s] \approx x : [s] \to tt \mid [s] \in S/{\equiv_{\leq}}\}.$$

Lemma 2. *For any $t, u \in T_{\Sigma}(X)$ with $[ls(t)] = [ls(u)]$:*
$$\mathcal{R}_{\mathcal{E}} \vdash (\forall X)\, t \downarrow u \quad \textit{iff} \quad \mathcal{R}_{\mathcal{E}}^{\approx} \vdash (\forall X)\, (t \approx u) \to^{*} tt.$$

For $\mathcal{R}_{\mathcal{E}}$ operationally terminating, $\mathcal{R}_{\mathcal{E}} \vdash (\forall X)\, t \downarrow u$ can be effectively checked in Maude by executing in the system module *mod* $\mathcal{R}_{\mathcal{E}}^{\approx}$ *endm* the search command:
$$search\ t \approx u \Rightarrow!\ tt\,.$$
giving us a decision procedure for deciding $\mathcal{R}_{\mathcal{E}} \vdash (\forall X)\, t \downarrow u$.

Note that the above result applies not just for $\mathcal{E} = (\Sigma, E \uplus B)$ an *unconditional* theory, but also for $\mathcal{E}$ *conditional* and satisfying the requirements in [14], namely, when $\mathcal{R}_{\mathcal{E}}$ is:

1. strongly deterministic;
2. strictly coherent modulo B; and
3. operationally terminating.

Therefore, reasoning about joinability in $\mathcal{R}_{\mathcal{E}}$ can be done under conditions (1)–(3) also for conditional theories and have the equivalence
$$\mathcal{R}_{\mathcal{E}} \vdash (\forall X)\, t \downarrow u \quad \text{iff} \quad \mathcal{R}_{\mathcal{E}}^{\approx} \vdash (\forall X)\, (t \approx u) \to^{*} tt.$$
and the implementation in Maude by *search* applying as well to conditional theories satisfying (1)–(3). In particular, this applies to the checking of joinability for the conditional theories of hereditarily finite sets in Section 4, which have some conditional equations for set intersection.

Appendix B. Specification of a Number Hierarchy

The following modules specify natural numbers, integers and rationals. The specification of natural and integer numbers have been divided between the sub-specification with the finite variant property (-FVP) and the rest of the ACU specification (-ACU).

```
fmod NAT-FVP is
  protecting TRUTH-VALUE .
  sorts Nat NzNat Zero .
  subsorts Zero NzNat < Nat .
  op 0 : -> Zero [ctor] .
  op 1 : -> NzNat [ctor] .
  op _+_ : Nat Nat -> Nat [ctor assoc comm id: 0 prec 33] .
  op _+_ : NzNat NzNat -> NzNat [ctor assoc comm id: 0 prec 33] .
  op p : NzNat -> Nat .                             *** predecessor
  op d : Nat Nat -> Nat [comm] .                    *** symmetric difference
  op _\_ : Nat Nat -> Nat .                         *** monus
  op _~_ : Nat Nat -> Bool [comm] .                 *** equality predicate
  op _>_ : Nat Nat -> Bool .

  vars N M : Nat .
  vars N' M' : NzNat .

  eq [NAT-FVP-01]: p(N + 1) = N [variant] .
  eq [NAT-FVP-02]: d(N + M, N) = M [variant] .
  eq [NAT-FVP-03]: (N + M) \ N = M [variant] .
  eq [NAT-FVP-04]: N \ (N + M) = 0 [variant] .
  eq [NAT-FVP-05]: N ~ N = true [variant] .
  eq [NAT-FVP-06]: (N + M') ~ N = false [variant] .
  eq [NAT-FVP-07]: M + N + 1 > N = true [variant] .
  eq [NAT-FVP-08]: N > N + M = false [variant] .
  eq [NAT-FVP-09]: N' > 0 = true [variant] .
endfm

fmod NAT-ACU is
  protecting NAT-FVP .
  op quot : Nat NzNat -> Nat .                      *** quotient
  op gcd : NzNat NzNat -> NzNat [comm] .            *** greatest common divisor
  op _*_ : Nat Nat -> Nat [assoc comm prec 31] .
  op _*_ : NzNat NzNat -> NzNat [assoc comm prec 31] .

  vars N M : Nat .
  vars N' M' K' : NzNat .
  var  N? : [Nat] .

  eq [NAT-ACU-01]: quot(N' + M', M') = 1 + quot(N', M')  .
  eq [NAT-ACU-02]: quot(M', M') = 1 .
  eq [NAT-ACU-03]: quot(M, N' + M) = 0 .
  eq [NAT-ACU-04]: gcd(N', N') = N' .
  eq [NAT-ACU-05]: gcd(N' + M', M') = gcd(N', M') .
  eq [NAT-ACU-06]: N? * 0 = 0 .
  eq [NAT-ACU-07]: N? * 1 = N? .
  eq [NAT-ACU-08]: N' * (M' + K') = (N' * M') + (N' * K') .
endfm

fmod INT-FVP is
  protecting NAT-FVP .
  sorts NzNeg NzInt Int .
  subsorts Nat NzNeg < Int .
  subsorts NzNat NzNeg < NzInt < Int .
  op -_ : NzNat -> NzNeg [ctor] .
  op -_ : Int -> Int .
  op -_ : NzInt -> NzInt .
  op _+_ : Int Int -> Int [assoc comm id: 0 prec 33] .
  op _+_ : NzInt NzInt -> NzInt [assoc comm id: 0 prec 33] .
  op |_| : Int -> Nat .                             *** absolute value
  op |_| : NzInt -> NzNat .
  op p : Int -> Int .                               *** predecessor
  op _~_ : Int Int -> Bool [comm] .                 *** equality predicate
  op _>_ : Int Int -> Bool .

  vars N' M' : NzNat .
  var  N : Nat .
  var  I' : NzInt .
  eq [INT-FVP-01]: - 0 = 0 [variant] .
  eq [INT-FVP-02]: - - I’ = I’ [variant] .
  eq [INT-FVP-03]: | N | = N [variant] .
  eq [INT-FVP-04]: | - N’ | = N’ [variant] .
  eq [INT-FVP-05]: p(0) = - 1 [variant] .
  eq [INT-FVP-06]: p(- N’) = -(N’ + 1) [variant] .
  eq [INT-FVP-07]: - N’ ~ - M’ = N’ ~ M’ [variant] .
  eq [INT-FVP-08]: - N’ ~ N = false [variant] .
  eq [INT-FVP-09]: - N’ > - M’ = M’ > N’ [variant] .
  eq [INT-FVP-10]: - N’ > N = false [variant] .
  eq [INT-FVP-11]: N > - N’ = true [variant] .
endfm

fmod INT-ACU is
  protecting INT-FVP .
  protecting NAT-ACU .
  op quot : Int NzInt -> Int .
  op gcd : NzInt NzInt -> NzNat [comm] .
  op _*_ : Int Int -> Int [assoc comm prec 31] .
  op _*_ : NzInt NzInt -> NzInt [assoc comm prec 31] .

  vars I’ J’ H’ : NzInt .
  vars N’ M’ : NzNat .
  var  Q : NzNeg .

  eq [INT-ACU-01]: quot(0, Q) = 0 .
  eq [INT-ACU-02]: quot(- N’, M’) = - quot(N’, M’) .
  eq [INT-ACU-03]: quot(N’, - M’) = - quot(N’, M’) .
  eq [INT-ACU-04]: quot(- N’, - M’) = quot(N’, M’) .
  eq [INT-ACU-05]: gcd(- N’, I’) = gcd(N’, I’) .
  eq [INT-ACU-06]: I’ * - J’ = - (I’ * J’) .
  eq [INT-ACU-07]: I’ * (J’ + H’) = (I’ * J’) + (I’ * H’) .
  eq [INT-ACU-08]: - I’ + - J’ = - (I’ + J’) .
  eq [INT-ACU-09]: - (I’ + - J’) = - I’ + J’ .
endfm

fmod RAT-ACU is
  protecting INT-ACU .
  sorts Rat NzRat PosRat NzPosRat .
  subsort Int < Rat .
  subsorts Nat NzPosRat < PosRat < Rat .
  subsorts NzInt < NzRat < Rat .
  subsorts NzNat < NzPosRat < NzRat PosRat .
  op _/_ : NzRat NzRat -> NzRat [prec 31] .         *** division
  op _/_ : NzInt NzNat -> NzRat [ctor prec 31] .
  op _/_ : NzNat NzNat -> NzPosRat [ctor prec 31] .
  op -_ : Rat -> Rat .
  op _+_ : Rat Rat -> Rat [assoc comm id: 0 prec 33] .
  op _+_ : PosRat PosRat -> PosRat [assoc comm id: 0 prec 33] .
  op _*_ : Rat Rat -> Rat [assoc comm prec 31] .
  op _*_ : NzRat NzRat -> NzRat [assoc comm prec 31] .
  op |_| : Rat -> PosRat .                          *** absolute value
  op _~_ : Rat Rat -> Bool [comm] .                 *** equality predicate
  op _>_ : Rat Rat -> Bool .

  vars I’ J’ : NzInt .
  vars R’ S’ : NzRat .
  vars N’ M’ : NzNat .

  eq [RAT-ACU-01]: 0 / N’ = 0 .
 ceq [RAT-ACU-02]: J’ / M’ = quot(J’, gcd(J’, M’)) / quot(M’, gcd(J’, M’))
     if gcd(J’, M’) > 1 = true .                    *** canonical form of fraction
  eq [RAT-ACU-03]: R’ / 1 = R’ .
  eq [RAT-ACU-04]: I’ / - N’ = - I’ / N’ .
  eq [RAT-ACU-05]: I’ / (J’ / M’) = (I’ * M’) / J’ .
  eq [RAT-ACU-06]: (I’ / N’) / J’ = I’ / (N’ * J’) .
  eq [RAT-ACU-07]: (I’ / N’) / (J’ / M’) = (I’ * M’) / (N’ * J’) .
  eq [RAT-ACU-08]: - (I’ / N’) = - I’ / N’ .
  eq [RAT-ACU-09]: J’ + (I’ / N’) = ((J’ * N’) + I’) / N’ .
  eq [RAT-ACU-10]: (I’ / N’) + (J’ / M’) = ((I’ * M’) + (J’ * N’)) / (N’ * M’) .
  eq [RAT-ACU-11]: (I’ / N’) * 0 = 0 .
  eq [RAT-ACU-12]: (I’ / N’) * J’ = (I’ * J’) / N’ .
  eq [RAT-ACU-13]: (I’ / N’) * (J’ / M’) = (I’ * J’) / (N’ * M’) .
  eq [RAT-ACU-14]: | I’ / N’ | = | I’ | / N’ .
  eq [RAT-ACU-15]: 0 ~ R’ = false .
  eq [RAT-ACU-16]: (I’ / N’) ~ J’ = I’ ~ (J’ * N’) .
  eq [RAT-ACU-17]: (I’ / N’) ~ (J’ / M’) = (I’ * M’) ~ (J’ * N’) .
  eq [RAT-ACU-18]: 0 > N’ / M’ = false .
  eq [RAT-ACU-19]: 0 > - N’ / M’ = true .
  eq [RAT-ACU-20]: N’ / M’ > 0 = true .
  eq [RAT-ACU-21]: - N’ / M’ > 0 = false .
  eq [RAT-ACU-22]: I’ > (J’ / M’) = (I’ * M’) > J’ .
  eq [RAT-ACU-23]: (I’ / N’) > J’ = I’ > (J’ * N’) .
  eq [RAT-ACU-24]: (I’ / N’) > (J’ / M’) = (I’ * M’) > (J’ * N’) .
endfm
```