Normalized Completion Revisited

Sarah Winkler* **and Aart Middeldorp**

Institute of Computer Science
University of Innsbruck, Austria
{sarah.winkler,aart.middeldorp}@uibk.ac.at

Abstract

Normalized completion (Marché 1996) is the last development in a line of research that extends Knuth-Bendix completion to completion modulo theories. If successful, a normalized completion procedure computes a rewrite system that allows to decide the validity problem using normalized rewriting. In this paper we present a completeness result for normalized completion, adapt critical pair criteria to the setting of normalized completion, and show how normalized completion procedures can benefit from AC-termination tools instead of relying on a fixed AC-compatible reduction order. We outline our implementation of this approach in the completion tool mascott and present experimental results.

1998 ACM Subject Classification F.4.2 Grammars and Other Rewriting Systems

Keywords and phrases term rewriting, completion

Digital Object Identifier 10.4230/LIPIcs.xxx.yyy.p

1 Introduction

Since the landmark paper of Knuth and Bendix [14], completion has evolved as a basic deduction method in theorem proving, computer algebra and computational logic. Various generalizations have been proposed to deal efficiently with common algebraic theories. The theory of associativity and commutativity (AC) has been incorporated in [19, 27]. For general theories $\mathcal{T}$ where $\mathcal{T}$-unification is finitary and the subterm ordering modulo $\mathcal{T}$ is well-founded, extensions have been presented in [11, 4]. These limitations on the theory have been partially overcome by constrained completion [12], which allows e.g. for completion modulo AC with a unit element, but excludes other theories such as abelian groups.

Normalized completion [21, 23] constitutes the last result in this line of research. It has three advantages over earlier proposals. (1) It allows completion modulo any theory $\mathcal{T}$ that can be represented as an AC-convergent rewrite system $\mathcal{S}$. (2) Critical pairs need not be computed for the theory $\mathcal{T}$, which may not be finitary or even have a decidable unification problem. Instead, any theory between AC and $\mathcal{T}$ can be used. (3) The AC-compatible reduction order used to establish termination need not be compatible with $\mathcal{T}$. This is beneficial for theories such as AC with a unit element where no $\mathcal{T}$-compatible reduction order can possibly exist.

Normalized completion is thus applicable to many common theories such as AC augmented with axioms for unit elements, idempotency or nilpotency, but also to groups and rings. It also generalizes Buchberger's algorithm for computing Gröbner bases [24]. Compared to previous completion techniques, it improves efficiency if the input theory includes a subtheory for which an AC-convergent presentation is known. In computing less critical pairs by focusing on a particular theory, the approach shares advantages with specialized theorem

* The first author is supported by a DOC-fFORTE grant of the Austrian Academy of Sciences.

submitted to the 21st EACSL Annual Conferences on Computer Science Logic.
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

proving techniques with built-in equational theories (e.g. [10, 26, 29]). Such sophisticated reasoning techniques are efficient in proving equational consequences.

The focus of this paper is to transform a given theory into a convergent system, in order to obtain a decision procedure for the theory which also allows to disprove equational consequences. Besides, convergent systems have applications in e.g. algebraic proof mining [30].

In this paper we establish a new completeness result for normalized completion. Although the techniques used in the proof are similar to ones for standard completion [5], the setting of normalized completion involves some subtleties. In contrast to [23], we thus make all AC-steps explicit to enhance clarity. Moreover we study how critical pair criteria can be incorporated to limit equational consequences, which has been identified as an issue for future work in [22]. For this purpose we also define a weaker notion of fairness. State-of-the-art implementations of normalized completion such as C*i*ME require the input of a suitable AC-compatible reduction order. This parameter is critical for success, but hard to determine in advance. We tackle this problem by applying the by now well-understood combination of two approaches: (1) termination tools replace fixed reduction orders as proposed in [31], and (2) back-tracking is avoided by keeping different orientations of equations. This combined multi-completion approach with termination tools has been investigated for standard completion [34], ordered completion [32] and AC-completion [33]. We present novel convergent systems obtained with normalized multi-completion using termination tools.

The remainder of this paper is structured as follows. Preliminaries on equational reasoning and rewriting are given in Section 2. In Section 3 we recall normalized completion, present our completeness result, and describe critical pair criteria in the setting of normalized completion. Section 4 describes the extension with termination tools. In Section 5 we give a short description of our tool mascott, outlining the multi-completion approach and some implementation details. We also present experimental results. In Section 6 we conclude with some topics for future work. Due to a lack of space, some (proof) details can be found in the appendix.

2 Preliminaries

We assume familiarity with term rewriting and Knuth-Bendix completion [2], and recall only some central notions. Let $\mathcal{F}$ be a signature. We consider term rewrite systems (TRSs) $\mathcal{R}$ over $\mathcal{F}$. If the associated rewrite relation $\to_\mathcal{R}$ is well-founded, we write $s \to^!_\mathcal{R} t$ if s rewrites to a normal form t. If t is the only normal form of s, we also write $t = s{\downarrow_\mathcal{R}}$. We also consider (symmetric) equational systems $\mathcal{E}$ over $\mathcal{F}$ with associated equational theory $=_\mathcal{E}$. If $u \approx v$ is an equation in $\mathcal{E}$ we write $u \simeq v$ to denote $u \approx v$ or $v \approx u$. Let $\mathcal{F}_\text{AC} \subseteq \mathcal{F}$ be a set of binary function symbols. The equational system AC contains equations $x + (y + z) \approx (x + y) + z$ and $x + y \approx y + x$ for all symbols $+ \in \mathcal{F}_\text{AC}$. We denote equivalence modulo AC by $\leftrightarrow^*_\text{AC}$. Formally, a term s rewrites to t at position p using a rule $\ell \to r$ and a substitution σ, denoted by $s \to^{p,\sigma}_{\ell\to r/\text{AC}} t$, if i) $s|_p \leftrightarrow^*_\text{AC} \ell\sigma$ and $t = s[r\sigma]_p$, or ii) $\text{root}(\ell) = +$ for some symbol $+ \in \mathcal{F}_\text{AC}$, and we have $s|_p \leftrightarrow^*_\text{AC} (\ell + x)\sigma$ and $t = s[(r + x)\sigma]_p$ for some variable x which does not occur in ℓ. We write $s \to_{\mathcal{R}/\text{AC}} t$ if $s \to^{p,\sigma}_{\ell\to r/\text{AC}} t$ for some rule $\ell \to r$ in $\mathcal{R}$, substitution σ, and position p.

A TRS $\mathcal{R}$ *terminates modulo* AC whenever the relation $\to_{\mathcal{R}/\text{AC}}$ is well-founded. To establish AC-termination we will consider AC-compatible reduction orders $\succ$, i.e., reduction orders that satisfy $\leftrightarrow^*_\text{AC} \cdot \succ \cdot \leftrightarrow^*_\text{AC} \subseteq \succ$. The TRS $\mathcal{R}$ is *convergent modulo AC* if it terminates modulo AC and the relation $\leftrightarrow^*_{\text{AC}\cup\mathcal{R}}$ coincides with $\to^*_\mathcal{R} \cdot \leftrightarrow^*_\text{AC} \cdot \leftarrow^*_\mathcal{R}$.

Consider an equational system $\mathcal{T}$ which allows to compute complete sets of unifiers. For a rewrite rule $\ell \to r$ in $\mathcal{R}$ and a variable-disjoint equation $u \simeq v$ in $\mathcal{T}$ such that ℓ and a proper non-variable subterm $u|_p$ of u are $\mathcal{T}$-unifiable, $u[\ell]_p \to u[r]_p$ is a $\mathcal{T}$-extended rule [27]. The set of all $\mathcal{T}$-extended rules of $\mathcal{R}$ is denoted by $\mathrm{EXT}_{\mathcal{T}}(\mathcal{R})$. An overlap is a quadruple $\langle \ell_2 \to r_2, p, \ell_1 \to r_1 \rangle_\Sigma$ consisting of rewrite rules $\ell_1 \to r_1$, $\ell_2 \to r_2$, a function position p in ℓ_1, and a complete set Σ of $\mathcal{T}$-unifiers of $\ell_1|_p$ and ℓ_2. We denote by $\mathrm{CP}_{\mathcal{T}}(\mathcal{R})$ the set of $\mathcal{T}$-critical pairs among rules in $\mathcal{R} \cup \mathrm{EXT}_{\mathcal{T}}(\mathcal{R})$.

In normalized completion, we consider a fixed rewrite system $\mathcal{S}$ and a pair $(\mathcal{E}, \mathcal{R})$ of equations $\mathcal{E}$ and rewrite rules $\mathcal{R}$. An *equational proof step* $s \leftrightarrow_e^{p,\sigma} t$ in $(\mathcal{S}, \mathcal{E}, \mathcal{R})$ is an *AC-step* (*equality step*) if e or e^{-1} is an equation in AC ($\mathcal{E}$) applied from left to right at position p in s with substitution σ. A proof step $s \leftrightarrow_{\ell\to r}^{p,\sigma} t$ is a *rewrite step* if $s = u[\ell\sigma]_p$ and $t = u[r\sigma]_p$ for some term u with position p and substitution σ and rewrite rule $\ell \to r$ in $\mathcal{R}$ or $\mathcal{S}$. In this case also $t \leftrightarrow_{r\leftarrow\ell}^{p,\sigma} s$ is a rewrite proof step. We call a proof step an $\mathcal{R}$-rewrite ($\mathcal{S}$-rewrite) step if it is a rewrite step using a rule in $\mathcal{R}$ ($\mathcal{S}$).

We sometimes write $s \leftrightarrow t$ to express the existence of some proof step, omitting the position p, substitution σ and equation or rule e. An *equational proof* P of an equation $t_0 \approx t_n$ is a finite sequence

$$t_0 \xleftrightarrow[e_0]{p_0} t_1 \xleftrightarrow[e_1]{p_1} \cdots \xleftrightarrow[e_{n-1}]{p_{n-1}} t_n \tag{1}$$

of equational proof steps. It has a *subproof* Q, denoted by $P[Q]$, if Q is a sequence $t_i \leftrightarrow \cdots \leftrightarrow t_j$ with $0 \leqslant i \leqslant j \leqslant n$. For a term u with position q, a substitution σ, and a proof P of the shape (1) we write $u[P\sigma]_q$ to denote the sequence

$$u[t_0\sigma]_q \xleftrightarrow[e_0]{qp_0} u[t_1\sigma]_q \xleftrightarrow[e_1]{qp_1} \cdots \xleftrightarrow[e_{n-1}]{qp_{n-1}} u[t_n\sigma]_q$$

which is again an equational proof. A *proof order* $\ggg$ is a well-founded order on equational proofs such that i) $P \ggg Q$ implies $u[P\sigma]_p \ggg u[Q\sigma]_p$ for all substitutions σ and terms u with position p, and ii) $P \ggg P'$ implies $Q[P] \ggg Q[P']$ for all proofs P, P' and Q.

In the sequel we will consider a fixed theory $\mathcal{T}$ that is representable as an AC-convergent rewrite system $\mathcal{S}$.[1] So

$$\xleftrightarrow[\mathcal{T}]{*} \;=\; \xrightarrow[\mathcal{S}/\mathrm{AC}]{!} \cdot \xleftrightarrow[\mathrm{AC}]{*} \cdot \xleftarrow[\mathcal{S}/\mathrm{AC}]{!}$$

For example, for the theory ACU consisting of an AC-operator $+$ with unit $\mathsf{0}$, we have $\mathcal{T} = \{x + (y + z) \approx (x + y) + z,\ x + y \approx y + x,\ x + \mathsf{0} \approx x\}$ and $\mathcal{S} = \{x + \mathsf{0} \to x\}$. Note that the representation $\mathcal{S}$ need not be unique.

We now define normalized rewriting as in [23] but use a different notation to distinguish it from the by now established notation for relative rewriting.

▶ **Definition 1.** Two terms s and t admit an $\mathcal{S}$*-normalized* $\mathcal{R}$*-rewrite step* if

$$s \xrightarrow[\mathcal{S}/\mathrm{AC}]{!} s' \xrightarrow[\ell\to r/\mathrm{AC}]{p} t \tag{2}$$

for some rule $\ell \to r$ in $\mathcal{R}$ and position p. We write

$$s \xrightarrow[\ell\to r\backslash\mathcal{S}]{p} t$$

[1] To avoid confusion we differentiate between the theory and its AC-convergent representation, although both are denoted by S in [23].

deduce	$\dfrac{\mathcal{E}, \mathcal{R}}{\mathcal{E} \cup \{s \approx t\}, \mathcal{R}}$	if $s \approx t \in \mathrm{CP}_L(\mathcal{R})$
normalize	$\dfrac{\mathcal{E} \cup \{s \approx t\}, \mathcal{R}}{\mathcal{E} \cup \{s{\downarrow} \approx t{\downarrow}\}, \mathcal{R}}$	
orient	$\dfrac{\mathcal{E} \cup \{s \simeq t\}, \mathcal{R}}{\mathcal{E} \cup \Theta(s,t), \mathcal{R} \cup \Psi(s,t)}$	if $s = s{\downarrow}$, $t = t{\downarrow}$, and $s \succ t$
delete	$\dfrac{\mathcal{E} \cup \{s \approx t\}, \mathcal{R}}{\mathcal{E}, \mathcal{R}}$	if $s \leftrightarrow^*_{\mathrm{AC}} t$
simplify	$\dfrac{\mathcal{E} \cup \{s \simeq t\}, \mathcal{R}}{\mathcal{E} \cup \{s \simeq u\}, \mathcal{R}}$	if $t \to_{\mathcal{R}\backslash\mathcal{S}} u$
compose	$\dfrac{\mathcal{E}, \mathcal{R} \cup \{s \to t\}}{\mathcal{E}, \mathcal{R} \cup \{s \to u\}}$	if $t \to_{\mathcal{R}\backslash\mathcal{S}} u$
collapse	$\dfrac{\mathcal{E}, \mathcal{R} \cup \{t \to s\}}{\mathcal{E} \cup \{u \approx s\}, \mathcal{R}}$	if $t \xrightarrow[\ell \to r\backslash\mathcal{S}]{p,\sigma} u$ for $\ell \to r \in \mathcal{R}$ with $t \mathrel{\triangleright_{\mathrm{AC}}} \ell$ or $s \succ r\sigma$

Figure 1 Normalized completion $\mathcal{N}_\mathcal{T}$.

for (2). Moreover, we write $s \to_{\mathcal{R}\backslash\mathcal{S}} t$ if $s \xrightarrow[\ell \to r\backslash\mathcal{S}]{p} t$ for some rule $\ell \to r$ in $\mathcal{R}$ and position p.

Given a TRS $\mathcal{R}$, equational proofs of the form $s \to^!_{\mathcal{R}\backslash\mathcal{S}} \cdot \leftrightarrow^*_\mathcal{T} \cdot \leftarrow^!_{\mathcal{R}\backslash\mathcal{S}} t$ will play a special role and are called *rewrite proofs*.

3 Normalized Completion

We consider an AC-compatible reduction order $\succ$ such that $\mathcal{S} \subseteq \succ$. Thus for any TRS $\mathcal{R}$ satisfying $\mathcal{R} \subseteq \succ$, the normalized rewrite relation $\to_{\mathcal{R}\backslash\mathcal{S}}$ is well-founded [23]. From now on we write $t{\downarrow}$ for $t{\downarrow}_{\mathcal{S}/\mathrm{AC}}$ and $s{\downarrow}_p$ for $s[u{\downarrow}]_p$ where $u = s|_p$. We let $c(s,p,t)$ denote the multiset $\{s\}$ if $s{\downarrow}_p = s$ and $\{s,t\}$ otherwise. We write $\triangleright_{\mathrm{AC}}$ to denote the relation $\leftrightarrow^*_{\mathrm{AC}} \cdot \triangleright$ [3], where $\triangleright$ denotes the proper encompassment relation.

In Figure 1 we recall the inference system of normalized completion as presented in [23]. In the deduce rule, L denotes some fixed theory such that $\mathrm{AC} \subseteq L \subseteq \mathcal{T}$.[2] In the orient rule, the function Θ maps a term pair to a sets of equations, while Ψ maps a term pair to a set of rewrite rules. Together they form an $\mathcal{S}$*-normalizing pair* [23, Definition 3.5]. A general $\mathcal{S}$-normalizing pair can always be defined as follows: $\Psi(u,v) = \{u \to v\}$ and $\Theta(u,v)$ consists of all AC-critical pairs between $u \to v$ and a rule in $\mathcal{S}$ for which the unifying substitution θ is $\mathcal{S}$-irreducible. However, depending on $\mathcal{T}$ more efficient, in the sense that less critical pairs have to be computed, normalizing pairs may exist [23, Section 4]. For example, for the theory ACU where an AC-operator $+$ has some unit element $\mathsf{0}$, the set $\Theta_{\mathrm{ACU}}(u,v)$ can be restricted to all equations $u\theta \approx v\theta$ in $\Theta(u,v)$ such that u contains a subterm $x + u'$ for some variable x and term u', where θ is the substitution $\{x \mapsto \mathsf{0}\}$. To simplify the presentation, we

[2] Thus if $\mathcal{T}$ itself is not decidable and finitary with respect to unification, one can simply use AC for L. On the other hand, for example the set of unifiers obtained from ACU or ACUI unification are typically much smaller than those obtained from AC unification.

restrict ourselves to runs using the general $\mathcal{S}$-normalizing pair, although we believe that the results also hold for specialized versions such as the aforementioned ACU-normalizing pair.

An inference sequence $(\mathcal{E}_0, \varnothing) \vdash (\mathcal{E}_1, \mathcal{R}_1) \vdash (\mathcal{E}_2, \mathcal{R}_2) \vdash \cdots$ with respect to $\mathcal{N}_\mathcal{T}$ is called a *run* with persistent equations $\mathcal{E}_\omega = \bigcup_i \bigcap_{j>i} \mathcal{E}_j$ and rules $\mathcal{R}_\omega = \bigcup_i \bigcap_{j>i} \mathcal{R}_j$. We write $(\mathcal{E}_0, \varnothing) \vdash^* (\mathcal{E}_\alpha, \mathcal{R}_\alpha)$ to express that the run has length α, where $\alpha = \omega$ if it is not finite. A run *succeeds* if $\mathcal{E}_\omega$ is empty, otherwise it *fails*.

3.1 Fairness and Correctness

Our relaxed notion of fairness is based on the definition of the underlying proof order, which we thus recall here. It is also required to establish correctness of critical pair criteria.

▶ **Definition 2.** The *complexity* of an equational proof step is defined as follows:

$$C(s \xleftrightarrow[\ell \approx r]{p,\sigma} t) = (\{s{\downarrow_p}, t{\downarrow_p}\}, \{s,t\}, \bot, \bot) \qquad C(s \xleftrightarrow[\mathrm{AC}]{} t) = (\bot, \{s\}, \bot, \bot)$$

$$C(s \xrightarrow[\ell \to r]{p,\sigma} t) = (c(s,p,t), \{s\}, \ell\sigma, r\sigma) \qquad C(s \xrightarrow[\mathcal{S}]{u \to v} t) = (\bot, \{s\}, \bot, \bot)$$

Complexities are compared by the lexicographic combination of $\succ_{\mathsf{mul}}$, $\succ_{\mathsf{mul}}$ modulo AC, $\rhd_{\mathrm{AC}}$ and $\succ$, where $\bot$ is considered minimal in all of these orderings. The complexity of an equational proof is the multiset consisting of the complexities of its steps. The *proof order* $\ggcurly$ on equational proofs is the multiset extension of the order on complexities of proof steps. Moreover, we write $P \Rightarrow Q$ if $P \ggcurly Q$ and P and Q prove the same equation.

Note that we define the third component of the complexity of a rewrite step to be $\ell\sigma$, rather than ℓ as in [23]. This modification does not harm the correctness proof, i.e., every inference step still results in a decrease with respect to the proof order (actually only the collapse rule is affected). This modification is required to incorporate critical pair criteria. The following persistence lemma expresses that any equation which has an equational proof P in some state of a run also has a persisting proof Q in the limit, and Q is smaller or equal than P. Using the fact that inference steps in $\mathcal{N}_\mathcal{T}$ result in a decrease with respect to the proof ordering [23, Figures 6 and 7], it can be proved in the same way as [5, Corollary 3.2].

▶ **Lemma 3.** *Consider a run $(\mathcal{E}_0, \mathcal{R}_0) \vdash (\mathcal{E}_1, \mathcal{R}_1) \vdash (\mathcal{E}_2, \mathcal{R}_2) \vdash \cdots$ and let P be a proof in $(\mathcal{S}, \bigcup_i \mathcal{E}_i, \bigcup_i \mathcal{R}_i)$. Then there exists a proof Q in $(\mathcal{S}, \mathcal{E}_\omega, \mathcal{R}_\omega)$ such that $P \Rightarrow^= Q$.* ◀

Fairness captures the important property of runs that whenever some inference step can achieve progress then progress is eventually made.

▶ **Definition 4.** A run $(\mathcal{E}_0, \mathcal{R}_0) \vdash (\mathcal{E}_1, \mathcal{R}_1) \vdash (\mathcal{E}_2, \mathcal{R}_2) \vdash \cdots$ is *fair* if for every non-rewrite proof P in $(\mathcal{S}, \mathcal{E}_\omega, \mathcal{R}_\omega)$, for which there exists an inference step $(\mathcal{E}_\omega, \mathcal{R}_\omega) \vdash (\mathcal{E}', \mathcal{R}')$ and a proof P' in $(\mathcal{S}, \mathcal{E}', \mathcal{R}')$ satisfying $P \Rightarrow P'$, there also exists a proof Q in $(\mathcal{S}, \mathcal{E}_i, \mathcal{R}_i)$ for some $i \geqslant 0$ such that $P \Rightarrow Q$.

Note that our definition is less restrictive than the original one, which is essential to incorporate critical pair criteria (see Section 3.3). The original definition [23] constitutes a sufficient criterion for fairness in our sense, provided that all AC-critical pairs between rules in $\mathcal{R}_\omega$ and $\mathcal{S}$ have a smaller proof. This is e.g. the case if the general normalizing pair is used.

▶ **Lemma 5.** *Any non-failing run which uses the general $\mathcal{S}$-normalizing pair and satisfies $\mathrm{CP}_L(\mathcal{R}_\omega) \subseteq \bigcup_i \mathcal{E}_i$ is fair.*

Proof. Let γ be a run $(\mathcal{E}_0, \mathcal{R}_0) \vdash (\mathcal{E}_1, \mathcal{R}_1) \vdash (\mathcal{E}_2, \mathcal{R}_2) \vdash \cdots$ such that $\mathcal{E}_\omega = \varnothing$ and $\mathrm{CP}_L(\mathcal{R}_\omega) \subseteq \bigcup_i \mathcal{E}_i$. We have to show that whenever a proof P in $(\mathcal{S}, \varnothing, \mathcal{R}_\omega)$ contains one of the patterns $s \leftarrow_{\mathcal{R}_\omega \backslash \mathcal{S}} \cdot \rightarrow_{\mathcal{R}_\omega \backslash \mathcal{S}} t$ or $s \leftarrow_{\mathcal{R}_\omega \backslash \mathcal{S}} \cdot \rightarrow_{\mathcal{S}/\mathrm{AC}} t$ then there exists a proof Q in $(\mathcal{S}, \varnothing, \mathcal{R}_\omega)$ such that $P \Rightarrow Q$. If such a pattern originates from a non-overlap or a variable overlap, then by [23, Figures 1, 3 and 4] the corresponding peak can be replaced by a subproof $s \rightarrow^*_{\mathcal{R}_\omega \backslash \mathcal{S}} \cdot \leftarrow^*_{\mathcal{R}_\omega \backslash \mathcal{S}} t$ or $s \rightarrow^*_{\mathcal{S}/\mathrm{AC}} \cdot \leftarrow^*_{\mathcal{R}_\omega \backslash \mathcal{S}} t$, and this replacement yields a proof which is smaller than P.

If $P = P[P']$ for a subproof P' of the shape $s \leftarrow_{\mathcal{R}_\omega \backslash \mathcal{S}} \cdot \rightarrow_{\mathcal{R}_\omega \backslash \mathcal{S}} t$ such that $s \approx t$ is an instance $C[u\sigma] \simeq C[v\sigma]$ of an L-critical pair $u \approx v$ then by assumption $u \approx v \in \mathcal{E}_i$ for some $i \geqslant 0$. By definition of $\ggcurly$, the proof $s \leftrightarrow_{u \simeq v} t$ is smaller than P'. By Lemma 3, there exists a proof Q' in $(\mathcal{S}, \varnothing, \mathcal{R}_\omega)$ such that $s \leftrightarrow_{u \simeq v} t \Rightarrow Q'$ holds. Thus the proof $Q = P[Q']$ in $(\mathcal{S}, \varnothing, \mathcal{R}_\omega)$ satisfies $P \Rightarrow Q$.

Finally, assume $P = P[P']$ for a subproof P' of the shape $s \leftarrow_{\ell \to r \backslash \mathcal{S}} \cdot \rightarrow_{\mathcal{S}/\mathrm{AC}} t$ such that $s \approx t$ is an instance $C[u\sigma] \simeq C[v\sigma]$ of an AC-critical pair between $\ell \to r$ and a rule in $\mathcal{S}$. According to [23, Figure 8], in a run using the general normalizing pair there exists a proof Q such that $P' \Rightarrow Q$. Again, by Lemma 3 there is a proof Q' in $(\mathcal{S}, \varnothing, \mathcal{R}_\omega)$ such that $Q \Rightarrow^* Q'$. Thus $P' \Rightarrow Q'$, and as $\Rightarrow$ is closed under proof contexts, $P = P[P']$ entails $P \Rightarrow P[Q']$. ◀

In the sequel a TRS $\mathcal{R}$ is called *$\mathcal{T}$-convergent* for $\mathcal{E}$ if $\rightarrow_{\mathcal{R} \backslash \mathcal{S}}$ is terminating and the relations $\leftrightarrow^*_{\mathcal{E} \cup \mathcal{T}}$ and $\rightarrow^*_{\mathcal{R} \backslash \mathcal{S}} \cdot \leftrightarrow^*_{\mathcal{T}} \cdot \leftarrow^*_{\mathcal{R} \backslash \mathcal{S}}$ coincide. As mentioned earlier, we restrict ourselves to general $\mathcal{S}$-normalizing pairs, and thus recall correctness only for this special case.

▶ **Theorem 6.** *Let $(\mathcal{E}, \varnothing) \vdash^* (\varnothing, \mathcal{R})$ be a fair and non-failing $\mathcal{N}_\mathcal{T}$ run using general $\mathcal{S}$-normalizing pairs. Then $\mathcal{R}$ is $\mathcal{T}$-convergent for $\mathcal{E}$.* ◀

Correctness as stated in [23, Theorem 3.8] assumes the more restrictive fairness condition that all critical pairs in $\mathrm{CP}_L(\mathcal{R}_\omega)$ are deduced. But its proof can easily be adapted to our relaxed definition as it is based on the proof normalization approach applied in [5], where the notion of fairness corresponds to Definition 4.

3.2 Completeness

Completeness of normalized completion can be proved in a similar fashion as completeness of standard completion. First we state an easy consequence of the definition of $\mathcal{T}$-convergence.

▶ **Lemma 7.** *Let $\mathcal{R}$ be $\mathcal{T}$-convergent for $\mathcal{E}$. If $s \leftrightarrow^*_{\mathcal{E} \cup \mathcal{T}} t$ then there exists a proof*

$$s \xrightarrow[\mathcal{R}\backslash\mathcal{S}]{!} \cdot \xrightarrow[\mathcal{S}/\mathrm{AC}]{!} u \xleftrightarrow[\mathrm{AC}]{*} v \xleftarrow[\mathcal{S}/\mathrm{AC}]{!} \cdot \xleftarrow[\mathcal{R}\backslash\mathcal{S}]{!} t \tag{3}$$

Moreover, the terms u and v are unique up to AC-equivalence.

Proof. Let s' and t' be normal forms of s and t with respect to $\rightarrow_{\mathcal{R} \backslash \mathcal{S}}$. By $\mathcal{T}$-convergence of $\mathcal{R}$, there exists a proof $s' \rightarrow^*_{\mathcal{R} \backslash \mathcal{S}} \cdot \leftrightarrow^*_{\mathcal{T}} \cdot \leftarrow^*_{\mathcal{R} \backslash \mathcal{S}} t'$. As s' and t' are $\rightarrow_{\mathcal{R} \backslash \mathcal{S}}$-irreducible, we must have $s' \leftrightarrow^*_{\mathcal{T}} t'$. Because $\mathcal{S}$ is AC-convergent, $s' \rightarrow^!_{\mathcal{S}/\mathrm{AC}} u \leftrightarrow^*_{\mathrm{AC}} v \leftarrow^!_{\mathcal{S}/\mathrm{AC}} t'$ must hold. Note that u and v are irreducible in $\rightarrow_{\mathcal{S}/\mathrm{AC}}$, and by the definition of normalized rewriting, also in $\rightarrow_{\mathcal{R} \backslash \mathcal{S}}$. Now suppose that u' also satisfies (3). Then we have $u \leftrightarrow^*_{\mathcal{E} \cup \mathcal{T}} u'$, so there is a proof of $u \approx u'$ of the form (3). Since also u' must be irreducible with respect to $\rightarrow_{\mathcal{R} \backslash \mathcal{S}}$ and $\rightarrow_{\mathcal{S}/\mathrm{AC}}$, we have $u \leftrightarrow^*_{\mathrm{AC}} u'$. A symmetric argument shows that v is unique modulo AC. ◀

▶ **Theorem 8.** *Assume $\mathcal{R}$ is a finite $\mathcal{T}$-convergent system for $\mathcal{E}$ and let $\succ$ be an AC-reduction order that contains $\mathcal{R}$ and $\mathcal{S}$. Then any fair and non-failing run from $\mathcal{E}$ using $\succ$ will produce a $\mathcal{T}$-convergent system in finitely many steps.*

Proof. Let $\mathcal{R}'$ denote the system obtained from $\mathcal{R}$ after replacing each right-hand side r by r' such that $r \to^!_{\mathcal{R}\backslash\mathcal{S}} \cdot \to^!_{\mathcal{S}/\mathrm{AC}} r'$. An argument similar to the one used in the proof of Lemma 7 shows that r' is unique. The system $\mathcal{R}'$ is terminating because it is contained in $\succ$. Moreover, the relation $\to^!_{\mathcal{R}\backslash\mathcal{S}} \cdot \to^!_{\mathcal{S}/\mathrm{AC}}$ is contained in $\to^!_{\mathcal{R}'\backslash\mathcal{S}} \cdot \to^!_{\mathcal{S}/\mathrm{AC}}$. Let $s \leftrightarrow^*_{\mathcal{E}\cup\mathcal{T}} t$. As $\mathcal{R}$ is $\mathcal{T}$-convergent, there exists an equational proof of $s \approx t$ of the form (3) by Lemma 7. Thus there is also a proof

$$s \xrightarrow[\mathcal{R}'\backslash\mathcal{S}]{!} \cdot \xrightarrow[\mathcal{S}/\mathrm{AC}]{!} \cdot \xleftrightarrow[\mathrm{AC}]{*} \cdot \xleftarrow[\mathcal{S}/\mathrm{AC}]{!} \cdot \xleftarrow[\mathcal{R}'\backslash\mathcal{S}]{!} t \tag{4}$$

Hence $\mathcal{R}'$ is $\mathcal{T}$-convergent for $\mathcal{E}$. Now consider a non-failing $\mathcal{N}_\mathcal{T}$ run γ which starts from $(\mathcal{E}, \varnothing)$ and uses $\succ$. Let $\mathcal{R}_\omega$ be the set of persistent rules. Let $\ell \to r$ be a rule in $\mathcal{R}'$. Since $\ell \approx r$ belongs to the equational theory of $\mathcal{T} \cup \mathcal{E}$, it has a persistent equational proof P after a finite number of steps in γ. Note that r must be $\to_{\mathcal{R}_\omega\backslash\mathcal{S}}$-irreducible. If $r \to_{\mathcal{R}_\omega\backslash\mathcal{S}} r'$ for some term r' then $r \succ r'$ and $r \leftrightarrow^*_{\mathcal{E}\cup\mathcal{T}} r'$, so $\mathcal{R}'$ admits an $\mathcal{S}$-normalized rewrite proof for $r \approx r'$. As r is irreducible in $\to_{\mathcal{R}\backslash\mathcal{S}}$ and $\to_{\mathcal{S}\backslash\mathrm{AC}}$, this proof must have the form $r \leftrightarrow^*_{\mathrm{AC}} \cdot \leftarrow^!_{\mathcal{S}/\mathrm{AC}} \cdot \leftarrow^!_{\mathcal{R}'\backslash\mathcal{S}} r'$. As $\mathcal{R}' \cup \mathcal{S} \subseteq \succ$ and $\succ$ is AC-compatible, this contradicts $r \succ r'$. It follows that the proof P must have the form $\ell \to^+_{\mathcal{R}_\omega\backslash\mathcal{S}} \cdot \to^*_{\mathcal{S}/\mathrm{AC}} \cdot \leftrightarrow^*_{\mathrm{AC}} r$. Let $\mathcal{Q}$ denote the set of rules in $\mathcal{R}_\omega$ required for all these rewrite proofs for rules of $\mathcal{R}'$. The system $\mathcal{Q}$ is obviously terminating, and derived in γ in finitely many steps. We claim that $\mathcal{Q}$ is $\mathcal{T}$-convergent for $\mathcal{E}$. First note that by the definition of normalized rewriting the relations $\to^*_{\mathcal{S}/\mathrm{AC}} \cdot \leftrightarrow^*_{\mathrm{AC}} \cdot \to_{\mathcal{Q}\backslash\mathcal{S}}$ and $\to_{\mathcal{Q}\backslash\mathcal{S}}$ coincide. Thus the inclusion $\to_{\mathcal{R}'\backslash\mathcal{S}} \subseteq \to^*_{\mathcal{Q}\backslash\mathcal{S}} \cdot \to^*_{\mathcal{S}/\mathrm{AC}} \cdot \leftrightarrow^*_{\mathrm{AC}}$ entails

$$\xrightarrow[\mathcal{R}'\backslash\mathcal{S}]{*} \subseteq \xrightarrow[\mathcal{Q}\backslash\mathcal{S}]{*} \cdot \xrightarrow[\mathcal{S}/\mathrm{AC}]{*} \cdot \xleftrightarrow[\mathrm{AC}]{*} \tag{5}$$

Hence for every proof $u \leftrightarrow^*_{\mathcal{E}\cup\mathcal{T}} v$ there exists an equational proof of the form (4), and due to (5) and the inclusion $\to^*_{\mathcal{S}/\mathrm{AC}} \cdot \leftrightarrow^*_{\mathrm{AC}} \cdot \to^*_{\mathcal{S}/\mathrm{AC}} \cdot \leftrightarrow^*_{\mathrm{AC}} \subseteq \to^*_{\mathcal{S}/\mathrm{AC}} \cdot \leftrightarrow^*_{\mathrm{AC}}$, also a valley proof

$$u \xrightarrow[\mathcal{Q}\backslash\mathcal{S}]{!} \cdot \xrightarrow[\mathcal{S}/\mathrm{AC}]{!} \cdot \xleftrightarrow[\mathrm{AC}]{*} \cdot \xleftarrow[\mathcal{S}/\mathrm{AC}]{!} \cdot \xleftarrow[\mathcal{Q}\backslash\mathcal{S}]{!} v \tag{6}$$

using rules in $\mathcal{Q}$ is possible. ◀

3.3 Critical Pair Criteria

We will now adapt critical pair criteria to the setting of normalized completion. A critical pair criterion CPC is a mapping from sets of equations to sets of equations such that CPC($\mathcal{E}$) is a subset of CP($\mathcal{E}$). Intuitively, CPC($\mathcal{E}$) contains those critical pairs that are considered redundant. A run $(\mathcal{E}_0, \mathcal{R}_0) \vdash (\mathcal{E}_1, \mathcal{R}_1) \vdash (\mathcal{E}_2, \mathcal{R}_2) \vdash \cdots$ using general normalizing pairs and a reduction order $\succ$ is *fair with respect to* CPC if for every peak P associated with a critical pair in $\mathrm{CP}(R_\omega) \setminus \bigcup_i \mathrm{CPC}(\mathcal{R}_i \cup \mathcal{E}_i)$ there exists a proof Q in $(\mathcal{S}, \bigcup_i \mathcal{E}_i, \bigcup_i \mathcal{R}_i)$ such that $P \Rightarrow Q$. A critical pair criterion CPC is *correct* if a non-failing run is fair in the general sense whenever it is fair with respect to CPC. Clearly, correct critical pair criteria allow to filter out unnecessary critical pairs without compromising completeness.

An equational proof P that has the form of a peak $s \leftarrow u \to t$ is *composite* in $(\mathcal{S}, \mathcal{E}, \mathcal{R})$ if there exist terms $u_0, \ldots, u_{n+1}$ where $s = u_0$ and $t = u_{n+1}$ and proofs $P_0, \ldots, P_n$ in $(\mathcal{S}, \mathcal{E}, \mathcal{R})$ such that P_i proves $u_i \approx u_{i+1}$ and $P \ggcurly P_i$ holds for all $1 \leqslant i \leqslant n$. The *compositeness criterion* returns all critical pairs among equations in $\mathcal{E}$ for which the associated overlaps are composite.

▶ **Lemma 9.** *The compositeness criterion is correct.*

Proof. Consider a run $(\mathcal{E}_0, \varnothing) \vdash (\mathcal{E}_1, \mathcal{R}_1) \vdash (\mathcal{E}_2, \mathcal{R}_2) \vdash \cdots$ which uses general normalizing pairs and is fair with respect to the compositeness criterion (CC) such that $\mathcal{E}_\omega = \varnothing$. We show that it is also fair in the general sense. To this end we argue that for every non-rewrite proof P in $(\mathcal{S}, \varnothing, \mathcal{R}_\omega)$ there exists a proof Q in $(\mathcal{S}, \varnothing, \mathcal{R}_\omega)$ such that $P \Rightarrow Q$. If P does not contain an instance of a critical pair in $\mathrm{CC}(\mathcal{R}_i \cup \mathcal{E}_i)$ for some $i \geqslant 0$ then we obtain a smaller proof in $(\mathcal{S}, \varnothing, \mathcal{R}_\omega)$ as in the proof of Lemma 5. So assume P contains a subproof $C[Q\sigma]$ such that Q is a peak corresponding to a critical pair $s \approx t$ in $\mathrm{CC}(\mathcal{R}_i \cup \mathcal{E}_i)$. By definition of CC, $(\mathcal{S}, \mathcal{E}_i, \mathcal{R}_i)$ allows for proofs $Q_0, \ldots, Q_n$ such that $Q' = Q_0; \ldots; Q_n$ proves $s \approx t$ and $Q \ggcurly Q_i$ holds for all $1 \leqslant i \leqslant n$. By Definition 2 we have $Q \ggcurly Q'$, and as Q and Q' prove the same equation also $Q \Rightarrow Q'$. As $\Rightarrow$ is closed under contexts and substitutions, replacing $C[Q\sigma]$ by $C[Q'\sigma]$ in P yields an proof P' such that $P \Rightarrow P'$. According to Lemma 3 a proof which is smaller or equal than P' is possible using rules and equations in $\mathcal{R}_\omega \cup \mathcal{T}$. ◀

This general criterion is hard to apply in practice, but several special cases can be checked efficiently. Consider an overlap o of the form $\langle \ell_2 \to r_2, p, \ell_1 \to r_1 \rangle_\Sigma$ giving rise to the set of critical pairs

$$\ell_1\sigma[r_2\sigma]_p \xleftarrow[\ell_2 \to r_2]{p} \ell_1\sigma[\ell_2\sigma]_p \xleftrightarrow[\mathcal{T}]{*} \ell_1\sigma \xrightarrow[\ell_1 \to r_1]{\epsilon} r_1\sigma \tag{7}$$

such that $\sigma \in \Sigma$. We let $u = \ell_1\sigma$, $u' = \ell_1\sigma[\ell_2\sigma]_p$, $s = r_1\sigma$ and $t = \ell_1\sigma[r_2\sigma]_p$. The cost of the overlap proof (7) is $c(P) = \{(c(u, \epsilon, s), \{u\}, u, s), (c(u', p, t), \{u'\}, u'|_p, t|_p)\} \cup c_\mathcal{T}(P)$ according to Definition 2, where $c_\mathcal{T}(P)$ corresponds to the cost of the subproof $u' \leftrightarrow^*_\mathcal{T} u$.

$\mathcal{S}$-reducibility. Assume u is $\mathcal{S}$/AC-reducible to some term v. We thus have a proof $P_1; P_2$ of $\ell_1\sigma[r_2\sigma]_p \approx r_1\sigma$ which is equivalent to P, where

$$P_1 = \ell_1\sigma[r_2\sigma]_p \xleftarrow[\ell_2 \to r_2]{p} u' \xleftrightarrow[\mathcal{T}]{*} u \xrightarrow[\mathcal{S}/\mathrm{AC}]{} v \qquad\qquad P_2 = v \xleftarrow[\mathcal{S}/\mathrm{AC}]{} u \xrightarrow[\ell_1 \to r_1]{\epsilon} r_1\sigma$$

As u is $\mathcal{S}$/AC-reducible we have $c(u, \epsilon, s) = \{u, s\}$, such that the complexities amount to

$$\begin{aligned} c(P) &= \{(\{u,s\}, \{u\}, u, s), (c(u', p, t), \{u'\}, u'|_p, t|_p)\} \cup c_\mathcal{T}(P) \\ c(P_1) &= \{(c(u', p, t), \{u'\}, u'|_p, t|_p), (\bot, \{u\}, \bot, \bot)\} \cup c_\mathcal{T}(P) \cup c_\mathrm{AC}(P_1) \\ c(P_2) &= \{(\bot, \{u\}, \bot, \bot), (\{u,s\}, \{u\}, u, s)\} \cup c_\mathrm{AC}(P_2) \end{aligned}$$

where $c_\mathrm{AC}(P_i)$ corresponds to the complexities of possibly required AC-steps in the $\mathcal{S}$/AC-step from u to v. Note that the complexities of AC-steps are smaller than all complexity tuples in $c(P)$. By Definition 2 we have $P \ggcurly P_1$ and $P \ggcurly P_2$ such that the overlap o is composite. A symmetric argument shows compositeness of any critical pair where u' is $\mathcal{S}$/AC-reducible.

In the sequel we thus assume that both u and u' are in $\mathcal{S}$/AC-normal form. Hence $u' \xleftrightarrow[\mathrm{AC}]{*} u$ by the AC-convergence of $\mathcal{S}$. Now assume that u' can be reduced to some term v by an $\mathcal{R}$-step using a rule $\ell_3 \to r_3$ at position q, such that $(\ell_3 \to r_3, q)$ is different from $(\ell_1 \to r_1, \epsilon)$ and $(\ell_2 \to r_2, p)$. Thus there are proofs

$$P_1 = \ell_1\sigma[r_2\sigma]_p \xleftarrow[\ell_2 \to r_2]{p} u' \xrightarrow[\ell_3 \to r_3]{q} v \qquad\qquad P_2 = v \xleftarrow[\ell_3 \to r_3]{q} u' \xleftrightarrow[\mathrm{AC}]{*} u \xrightarrow[\ell_1 \to r_1]{\epsilon} r_1\sigma$$

such that $P_1; P_2$ proves the same equation as P.

Primality. If $p < q$, the primality criterion proposed in [13] is applicable. As u and u' are in $\mathcal{S}$-normal form, the proof complexities amount to

$$\begin{aligned} c(P) &= \{(\{u\},\{u\},u,s),(\{u'\},\{u'\},u'|_p,t|_p)\} \cup c_{\mathrm{AC}}(P) \\ c(P_1) &= \{(\{u'\},\{u'\},u'|_p,t|_p),(\{u'\},\{u'\},u'|_q,v|_q)\} \\ c(P_2) &= \{(\{u\},\{u\},u,s),(\{u'\},\{u'\},u'|_q,v|_q)\} \cup c_{\mathrm{AC}}(P_2) \end{aligned}$$

Since $q \neq \epsilon$, $u \rhd_{\mathrm{AC}} u'|_q$ and thus $P \ggg P_1$. Furthermore, $(\{u'\},\{u'\},u'|_p,t|_p)$ exceeds $(\{u'\},\{u'\},u'|_q,v|_q)$ as $u'|_p \rhd u'|_q$ such that $P \ggg P_2$. It follows that P is composite. (Note that at this point our modification in the proof order is required.) As in standard completion, the unblockedness criterion [5] forms a special case of the primality criterion. This criterion considers all critical pairs superfluous where $x\sigma$ is $\mathcal{R}$/AC-reducible for some variable x occurring in ℓ_1 or ℓ_2.

Connectedness. Also a normalized completion variant of the connectedness criterion [16] can be defined. If both proofs P_1 and P_2 are either not proper overlaps or the respective critical pair was already considered, then there exist equivalent proofs P_1' and P_2' which do not involve the term u. Hence $P \ggg P_1'$ and $P \ggg P_2'$ holds since all terms occurring in the complexity tuples of P_1' and P_2' are strictly smaller than u.

Combining criteria. Since the three criteria described above capture special cases of compositeness they can also be combined. For a critical pair $s \approx t$ originating from an overlap o, one can thus perform the following checks:

1. If u or u' is not in $\mathcal{S}$-normal form, the critical pair is composite.
2. Otherwise, if u' is reducible with some $\mathcal{R}$/AC-step strictly below the position p of the overlap then $s \approx t$ is composite.
3. Otherwise, one checks whether u or u' is (in addition to the rewrite steps involved in the overlap) $\mathcal{R}$/AC-reducible with some rule $\ell_3 \to r_3$ at position q. If there exists such a reduction such that for both $i \in \{1,2\}$ the rules $\ell_i \to r_i$ and $\ell_3 \to r_3$ do either not form a proper overlap, or the corresponding critical pair was already considered then $s \approx t$ is composite.

4 Normalized Completion with Termination Tools

Classical Knuth-Bendix completion requires a fixed reduction order as input. To avoid fixing this critical parameter from the very beginning and obtain a greater variety of usable orders, Wehrman *et al.* [31] proposed *completion with termination tools*. In this section we take a similar approach to normalized completion.

The inference rules in Figure 2 describe $\mathcal{T}$-normalized completion with termination tools (abbreviated $\mathcal{N}_{\mathcal{T}}^{\mathsf{TT}}$). Note that $\mathsf{collapse}_2$ simulates the $\mathsf{collapse}$ rule of $\mathcal{N}_{\mathcal{T}}$ when $t \rhd_{\mathrm{AC}} \ell$ does not hold (and hence $t \doteq_{\mathrm{AC}} \ell$ holds) but $s \succ r\sigma$ is satisfied.

A sequence $(\mathcal{E}_0,\varnothing,\varnothing) \vdash (\mathcal{E}_1,\mathcal{R}_1,\mathcal{C}_1) \vdash (\mathcal{E}_2,\mathcal{R}_2,\mathcal{C}_2) \vdash \cdots$ of inference steps in $\mathcal{N}_{\mathcal{T}}^{\mathsf{TT}}$ is called a *run*. Before proving the correctness of $\mathcal{N}_{\mathcal{T}}^{\mathsf{TT}}$, we illustrate the inference system on a concrete example.

▶ **Example 10.** Consider the initial set of equations $\mathcal{E}_0 = \{\mathsf{a} + x \approx \mathsf{b} + \mathsf{g}(\mathsf{a})\}$, where $+$ is an AC symbol with unit $\mathsf{0}$, such that the theory $\mathcal{T}$ can be represented as $\mathcal{S} = \{x + \mathsf{0} \to x\}$. Note that the given equation cannot be oriented with an order enjoying the subterm property. Thus any completion tool restricted to simplification orders such as AC-RPO [28] or AC-KBO [15] fails immediately. In contrast, employing termination tools using e.g. AC-dependency pairs

deduce	$\dfrac{\mathcal{E}, \mathcal{R}, \mathcal{C}}{\mathcal{E} \cup \{s \approx t\}, \mathcal{R}, \mathcal{C}}$	if $s \approx t \in \mathrm{CP}_L(\mathcal{R})$
normalize	$\dfrac{\mathcal{E} \cup \{s \approx t\}, \mathcal{R}, \mathcal{C}}{\mathcal{E} \cup \{s{\downarrow} \approx t{\downarrow}\}, \mathcal{R}, \mathcal{C}}$	
orient	$\dfrac{\mathcal{E} \cup \{s \simeq t\}, \mathcal{R}, \mathcal{C}}{\mathcal{E} \cup \Theta(s,t), \mathcal{R} \cup \Psi(s,t), \mathcal{C} \cup \{s \to t\}}$	if $s = s{\downarrow}$, $t = t{\downarrow}$, and $\mathcal{C} \cup \{s \to t\} \cup \mathcal{S}$ is AC-terminating
delete	$\dfrac{\mathcal{E} \cup \{s \approx t\}, \mathcal{R}, \mathcal{C}}{\mathcal{E}, \mathcal{R}, \mathcal{C}}$	if $s \leftrightarrow^*_{\mathrm{AC}} t$
simplify	$\dfrac{\mathcal{E} \cup \{s \simeq t\}, \mathcal{R}, \mathcal{C}}{\mathcal{E} \cup \{s \simeq u\}, \mathcal{R}, \mathcal{C}}$	if $t \to_{\mathcal{R}\backslash\mathcal{S}} u$
compose	$\dfrac{\mathcal{E}, \mathcal{R} \cup \{s \to t\}, \mathcal{C}}{\mathcal{E}, \mathcal{R} \cup \{s \to u\}, \mathcal{C}}$	if $t \to_{\mathcal{R}\backslash\mathcal{S}} u$
collapse	$\dfrac{\mathcal{E}, \mathcal{R} \cup \{t \to s\}, \mathcal{C}}{\mathcal{E} \cup \{u \approx s\}, \mathcal{R}, \mathcal{C}}$	if $t \to_{\mathcal{R}\backslash\mathcal{S}} u$ using $\ell \to r \in \mathcal{R}$ with $t \rhd_{\mathrm{AC}} \ell$
collapse$_2$	$\dfrac{\mathcal{E}, \mathcal{R} \cup \{t \to s\}, \mathcal{C}}{\mathcal{E} \cup \{u \approx s\}, \mathcal{R}, \mathcal{C} \cup \{s \to r\theta\}}$	if $t \to_{\mathcal{R}\backslash\mathcal{S}} u$ using $\ell \to r \in \mathcal{R}$ with $t \doteq_{\mathrm{AC}} l$ and renaming θ, and $\mathcal{C} \cup \{s \to r\theta\} \cup \mathcal{S}$ is AC-terminating

Figure 2 Normalized completion with termination checks $\mathcal{N}^{\mathsf{TT}}_{\mathcal{T}}$.

allows to orient the equation $\mathsf{a} + x \approx \mathsf{b} + \mathsf{g}(\mathsf{a})$. When using ACU-normalizing pairs, this results in the state

$$\mathcal{E}_1: \quad \mathsf{a} + 0 \approx \mathsf{b} + \mathsf{g}(\mathsf{a}) \qquad \mathcal{R}_1: \quad \mathsf{a} + x \to \mathsf{b} + \mathsf{g}(\mathsf{a}) \qquad \mathcal{C}_1: \quad \mathsf{a} + x \to \mathsf{b} + \mathsf{g}(\mathsf{a})$$

After normalizing $\mathsf{a} + 0$ to a, we have

$$\mathcal{E}_2: \quad \mathsf{a} \approx \mathsf{b} + \mathsf{g}(\mathsf{a}) \qquad \mathcal{R}_2: \quad \mathsf{a} + x \to \mathsf{b} + \mathsf{g}(\mathsf{a}) \qquad \mathcal{C}_2: \quad \mathsf{a} + x \to \mathsf{b} + \mathsf{g}(\mathsf{a})$$

Since $\mathcal{C}_2 \cup \{\mathsf{b} + \mathsf{g}(\mathsf{a}) \to \mathsf{a}\}$ terminates, we may perform an orient step:

$$\mathcal{E}_3: \qquad \mathcal{R}_3: \quad \begin{array}{r} \mathsf{a} + x \to \mathsf{b} + \mathsf{g}(\mathsf{a}) \\ \mathsf{b} + \mathsf{g}(\mathsf{a}) \to \mathsf{a} \end{array} \qquad \mathcal{C}_3: \quad \begin{array}{r} \mathsf{a} + x \to \mathsf{b} + \mathsf{g}(\mathsf{a}) \\ \mathsf{b} + \mathsf{g}(\mathsf{a}) \to \mathsf{a} \end{array}$$

In a subsequent compose step, the new rule can be used to reduce the first one:

$$\mathcal{E}_4: \qquad \mathcal{R}_4: \quad \begin{array}{r} \mathsf{a} + x \to \mathsf{a} \\ \mathsf{b} + \mathsf{g}(\mathsf{a}) \to \mathsf{a} \end{array} \qquad \mathcal{C}_4: \quad \begin{array}{r} \mathsf{a} + x \to \mathsf{b} + \mathsf{g}(\mathsf{a}) \\ \mathsf{b} + \mathsf{g}(\mathsf{a}) \to \mathsf{a} \end{array}$$

Three applications of deduce yield the state

$$\mathcal{E}_7: \quad \begin{array}{r} \mathsf{a} + \mathsf{g}(\mathsf{a}) \approx \mathsf{a} + \mathsf{a} \\ \mathsf{a} + \mathsf{a} \approx \mathsf{a} + \mathsf{b} \\ \mathsf{a} + \mathsf{a} \approx \mathsf{a} \end{array} \qquad \mathcal{R}_7: \quad \begin{array}{r} \mathsf{a} + x \to \mathsf{a} \\ \mathsf{b} + \mathsf{g}(\mathsf{a}) \to \mathsf{a} \end{array} \qquad \mathcal{C}_7: \quad \begin{array}{r} \mathsf{a} + x \to \mathsf{b} + \mathsf{g}(\mathsf{a}) \\ \mathsf{b} + \mathsf{g}(\mathsf{a}) \to \mathsf{a} \end{array}$$

Since all terms in $\mathcal{E}_7$ simplify to a, the resulting trivial equations can be deleted such that the run succeeds. Since all critical pairs among rules in $\mathcal{R}_7$ were already deduced, the run is also fair. Thus the system $\mathcal{R}_7 = \{\mathsf{a} + x \to \mathsf{a},\ \mathsf{b} + \mathsf{g}(\mathsf{a}) \to \mathsf{a}\}$ is $\mathcal{T}$-convergent for $\mathcal{E}_0$.

The proof of the following simulation result can be found in the appendix. It relies on the fact that if $\mathcal{C}$ is some TRS such that $\mathcal{C} \cup \mathcal{S}$ terminates modulo AC then the relation $\to^+_{(\mathcal{C}\cup\mathcal{S})/\mathrm{AC}}$ is an AC-compatible reduction order.

▶ **Lemma 11.** *1. For every finite run* $(\mathcal{E}_0, \varnothing, \varnothing) \vdash^* (\mathcal{E}_n, \mathcal{R}_n, \mathcal{C}_n)$ *in* $\mathcal{N}^{\mathcal{TT}}_{\mathcal{T}}$ *there exists an* $\mathcal{N}_{\mathcal{T}}$ *run* $(\mathcal{E}_0, \varnothing) \vdash^* (\mathcal{E}_n, \mathcal{R}_n)$ *using the AC-compatible reduction order* $\to^+_{(\mathcal{C}_n\cup\mathcal{S})/\mathrm{AC}}$.
2. Every $\mathcal{N}_{\mathcal{T}}$ *run* $(\mathcal{E}_0, \varnothing) \vdash^* (\mathcal{E}_\alpha, \mathcal{R}_\alpha)$ *using an AC-compatible reduction order* $\succ$ *is simulated in an* $\mathcal{N}^{\mathcal{TT}}_{\mathcal{T}}$ *run* $(\mathcal{E}, \varnothing, \varnothing) \vdash^* (\mathcal{E}_\alpha, \mathcal{R}_\alpha, \mathcal{C}_\alpha)$ *such that* $\mathcal{C}_\alpha \subseteq \succ$.

▶ **Corollary 12.** *For any finite non-failing and fair* $\mathcal{N}^{\mathcal{TT}}_{\mathcal{T}}$ *run* $(\mathcal{E}, \varnothing, \varnothing) \vdash^* (\varnothing, \mathcal{R}, \mathcal{C})$ *the system* $\mathcal{R}$ *is* $\mathcal{T}$*-convergent for* $\mathcal{E}$.

Proof. According to Lemma 11, the same TRS $\mathcal{R}$ can be derived in a run of normalized completion using the reduction order $\to^+_{(\mathcal{C}\cup\mathcal{S})/\mathrm{AC}}$. This run is fair and non-failing. Theorem 6 yields that $\mathcal{R}$ is $\mathcal{T}$-convergent for $\mathcal{E}$. ◀

5 Implementation Details and Experimental Results

5.1 Multi-Completion

In completion with termination tools, the orient rule leaves a choice if the considered equation can be oriented in both directions. As the appropriate orientation of an equation is hard to predict, it is beneficial to keep track of multiple orientations. Thus, in our tool mascott we implemented a *multi-completion* variant of normalized completion with termination tools, following the approach suggested for completion with multiple reduction orders [17]. The basic idea is to simulate multiple $\mathcal{N}^{\mathcal{TT}}_{\mathcal{T}}$ processes in parallel, but share common inferences to gain efficiency. Here a process corresponds to a sequence of decisions on how to orient equations. In our implementation, we model a process as a bit string. The initial process is denoted by ϵ. A formal description of this approach can be found in the appendix. Here we content ourselves with giving an example.

▶ **Example 13.** We consider the system CGA describing an abelian group with a group action ϕ on itself such that two endomorphisms f and g commute with respect to ϕ, as described by the following set of equations $\mathcal{E}$

$$\begin{array}{ccc} \mathsf{f}(\mathsf{e}) \approx \mathsf{e} & x \cdot x^{-1} \approx \mathsf{e} & \mathsf{f}(x \cdot y) \approx \mathsf{f}(x) \cdot \mathsf{f}(y) \\ \mathsf{g}(\mathsf{e}) \approx \mathsf{e} & & \mathsf{g}(x \cdot y) \approx \mathsf{g}(x) \cdot \mathsf{g}(y) \\ \phi(\mathsf{e}, x) \approx x & \phi(x, \phi(y, z)) \approx \phi(x \cdot y, z) & \phi(\mathsf{f}(x), \mathsf{g}(y)) \approx \phi(\mathsf{g}(y), \mathsf{f}(x)) \end{array}$$

together with the theory $\mathcal{T} = \{x \cdot y \approx y \cdot x,\ (x \cdot y) \cdot z \approx x \cdot (y \cdot z),\ x \cdot \mathsf{e} \approx x\}$. Several equations are orientable in both directions. A multi-completion run thus gives rise to a process tree, where each branch corresponds to a possible sequence of orientations. Part of the process tree developed in a run on CGA run is shown in Figure 3. Note that the equation $\phi(\mathsf{f}(x), \mathsf{g}(y)) \approx \phi(\mathsf{g}(y), \mathsf{f}(x))$ cannot be oriented with AC-RPO or AC-compatible polynomial interpretations. Hence e.g. C*i*ME[3] cannot succeed, but by using muterm [1] for termination

[3] We compared with C*i*ME 3.0.2, see `http://cime.lri.fr` and [7].

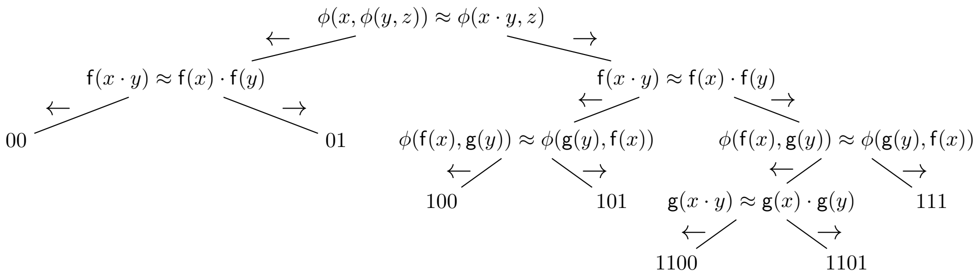

Figure 3 Part of the process tree developed in a run on CGA where process 1101 succeeds.

checks, mascott can produce the following AC-convergent system

$$\begin{array}{rcc}
\mathsf{f}(\mathsf{e}) \to \mathsf{e} & \mathsf{f}(x) \cdot \mathsf{f}(y) \to \mathsf{f}(x \cdot y) & \mathsf{i}(\mathsf{f}(x)) \to \mathsf{f}(\mathsf{i}(x)) \\
\mathsf{g}(\mathsf{e}) \to \mathsf{e} & \mathsf{g}(x) \cdot \mathsf{g}(y) \to \mathsf{g}(x \cdot y) & \mathsf{i}(\mathsf{g}(x)) \to \mathsf{g}(\mathsf{i}(x)) \\
x \cdot x^{-1} \to \mathsf{e} & x \cdot \mathsf{e} \to x & \mathsf{i}(\mathsf{e}) \to \mathsf{e} \\
\phi(\mathsf{e}, x) \to x & \phi(\mathsf{f}(x), \mathsf{e}) \to \mathsf{f}(x) & \phi(x, \mathsf{f}(y)) \to \phi(\mathsf{f}(y) \cdot x, \mathsf{e}) \\
\phi(x, \phi(y, z)) \to \phi(x \cdot y, z) & \phi(\mathsf{g}(x), \mathsf{e}) \to \mathsf{g}(x) & \phi(x, \mathsf{g}(y)) \to \phi(\mathsf{g}(y) \cdot x, \mathsf{e}) \\
\mathsf{i}(\mathsf{i}(x)) \to x & \mathsf{i}(x \cdot y) \to \mathsf{i}(x) \cdot \mathsf{i}(y) &
\end{array}$$

5.2 Implementation

We extended our tool mascott [33] to handle normalized multi-completion with termination tools. While the basic control loop remained the same, some changes had to be made to apply normalized completion. First of all, an AC-convergent TRS $\mathcal{S}$ representing the theory $\mathcal{T}$ is fixed and all terms are kept in $\mathcal{S}$-normalized form. The TRS $\mathcal{S}$ can be supplied by the user, otherwise mascott detects an applicable theory automatically (currently ACU, groups and rings are supported, besides AC). The orient inference had to be changed to add equations in the Θ component. Currently we always compute AC-critical pairs and apply general $\mathcal{S}$-normalizing pairs in orient steps, independent of the theory $\mathcal{T}$. In order to limit the number of nodes, the critical pair criteria described in Section 3.3 were implemented. As in the previous version of mascott, termination checks required in orient inference steps may be performed by an external termination tool supporting AC-termination. Alternatively, mascott can now also apply AC-RPO [28] internally. Further details can be found in [33] or obtained from the mascott website.[4]

5.3 Experiments

To evaluate our approach we ran mascott on problems collected from a number of different sources. All of the following tests were performed on an Intel Core Duo running at a clock rate of 1.4 GHz with 2.8 GB of main memory. Termination checks were done with muterm, and the primality critical pair criterion was used. The global timeout and the timeout for each termination check were set to 300 and 2 seconds, respectively.

In Table 1 we compare the results obtained with mascott applying different theories $\mathcal{T}$ (AC, AC with unit (ACU) and the theory of abelian groups (AG)) as well as automatic

[4] http://cl-informatik.uibk.ac.at/software/mascott

	mascott								CiME[3]
	AC		ACU		AG		auto		
	(1)	(2)	(1)	(2)	(1)	(2)	(1)	(2)	(1)
[a]abelian groups (AG)	1.6	77	2.4	61	0.1	5	0.1	5	0.05
AG + homomorphism	181.7	928	173.5	993	4.8	104	4.8	104	0.05
[c]G0	1.9	82	1.9	70	0.1	8	0.1	8	?
[c]G1	∞		∞		12.4	49	12.5	49	?
[a]arithmetic	14.9	503	15.1	483	–		13.8	483	?
[a]AC-ring with unit	22.9	501	28.5	466	7.2	301	0.1	9	0.1
[b]binary arithmetic	2.9	199	2.8	185	–		3.0	185	?
[b]ternary arithmetic	18.1	816	17.3	781	–		17.3	781	?
Example 10	0.3	26	0.2	17	–		0.3	26	?
Example 13	∞		∞		15.4	486	15.2	486	?
Example 14	∞		∞		216.7	457	145.1	400	?
[a]semiring	3.3	209	3.6	192	–		3.5	193	0.1
[d]sum	1.4	4	1.5	5	–		1.4	4	?
completed systems	10		10		7		13		4

Table 1 Comparison of mascott using different theories.

theory detection.[5] The superscripts attached to problems indicate their source: a refers to [8] and b refers to [25], c is associated with finite group representations in [20], and d refers to [18]. The remaining examples were added by the authors. Columns (1) list the total time in seconds while columns (2) give the number of nodes created during the run. The symbol ∞ marks a timeout, and – indicates that the theory is not applicable. In line with [23], we observed that completion with respect to larger theories $\mathcal{T}$ is typically faster. Only in some cases such as the ring problem ACU-normalized completion is slower than AC-normalized completion, due to an unfortunate selection sequence. As expected, CiME is much faster if an appropriate reduction order is supplied as input. But as already mentioned, such a reduction order is hard to determine in advance, and in some cases no usable AC-RPO or polynomial interpretation exists. This is e.g. the case for Example 13, where mascott is able to find an ACU-convergent system in a bit more than one hour, and for the example given below.

Concerning critical pair criteria, we found that the primality criterion decreased the total number of nodes by nearly 40%, which reduces the computation time by about 25%. $\mathcal{S}$-reducibility does not filter out any critical pairs if completion modulo ACU is performed. For normalized completion modulo group theory, very few redundant critical pairs are detected. The connectedness criterion was found to be comparatively expensive, and also the combined criterion could not achieve the same performance gain as the simpler primality criterion due to the additional effort of testing the criterion. Complete tables and more details on experimental results can be obtained from the website.

▶ **Example 14.** Consider ring theory with two commuting multiplicative mappings as defined by AC-axioms for + together with the set of equations

$$x + 0 \approx x \qquad \mathsf{f}(1) \approx 1 \qquad x \cdot (y + z) \approx (x \cdot y) + (x \cdot z)$$
$$x + (-x) \approx 0 \qquad \mathsf{g}(1) \approx 1 \qquad (x + y) \cdot z \approx (x \cdot z) + (x \cdot z)$$

[5] The mascott website further includes a comparison with the version of mascott described in [33].

$$1 \cdot x \approx x \qquad \mathsf{f}(x \cdot y) \approx \mathsf{f}(x) \cdot \mathsf{f}(y) \qquad (x \cdot y) \cdot z \approx x \cdot (y \cdot z)$$
$$x \cdot 1 \approx x \qquad \mathsf{g}(x \cdot y) \approx \mathsf{g}(x) \cdot \mathsf{g}(y) \qquad \mathsf{f}(x) \cdot \mathsf{g}(y) \approx \mathsf{g}(y) \cdot \mathsf{f}(x)$$

Our tool computes a convergent system using normalized completion modulo group theory/ring theory in 216.7/145.1 seconds producing 457/400 nodes, respectively. Normalized completion modulo AC and ACU as well as the previous version of mascott yields a timeout. Note that the equation $\mathsf{f}(x) \cdot \mathsf{g}(y) \approx \mathsf{g}(y) \cdot \mathsf{f}(x)$ cannot be oriented with AC-RPO or AC-compatible polynomial interpretations. Hence no suitable input for C*i*ME is known.

6 Conclusion

We presented completeness and uniqueness results for $\mathcal{S}$-normalized completion. Critical pair criteria for this setting were presented and proved correct using a relaxed notion of fairness. We also showed how the use of termination tools supporting AC-termination can replace fixed reduction orders. Thus a user does not need to fix an AC-compatible reduction order in advance, a suitable ordering is instead found automatically. We implemented $\mathcal{S}$-normalized multi-completion with termination tools in mascott to evaluate our approach. As expected, a convergent system is obtained (considerably) faster if a convergent system for a subtheory is known in advance. Furthermore, we showed that the use of termination tools adds power to normalized completion in that new convergent systems can be constructed.

As future work, we want to investigate completeness and critical pair criteria for arbitrary normalizing pairs as defined in [23, Definition 3.5]. We also plan to improve our implementation in several respects. Unification algorithms and optimized normalizing pairs for specific theories are expected to generate fewer critical pairs (cf. [6]). Also, more powerful termination techniques that could be used internally will improve performance. Note that the union $\mathcal{C} \cup \mathcal{S}$ terminates modulo AC if and only if the relative termination problems $(\mathcal{S} \cup \mathcal{C})/(\mathrm{AC} \cup \mathrm{AC}^{-1})$ or, equivalently [9], $\mathcal{C}/(\mathcal{S} \cup \mathrm{AC} \cup \mathrm{AC}^{-1})$ and $\mathcal{S}/(\mathrm{AC} \cup \mathrm{AC}^{-1})$ are terminating. It could thus be investigated whether the use of termination tools supporting relative termination pays off.

References

1 B. Alarcón, R. Gutiérrez, J. Iborra, and S. Lucas. Proving termination of context-sensitive rewriting with MU-TERM. In *6th PROLE*, volume 188 of *ENTCS*, pages 105–115, 2007.

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

3 L. Bachmair. *Canonical Equational Proofs*. Progress in Theoretical Computer Science. Birkhäuser, 1991.

4 L. Bachmair and N. Dershowitz. Completion for rewriting modulo a congruence. *Theoretical Computer Science*, 67(2,3):173–201, 1989.

5 L. Bachmair and N. Dershowitz. Equational inference, canonical proofs, and proof orderings. *Journal of the ACM*, 41(2):236–276, 1994.

6 A. Boudet, E. Contejean, and C. Marché. AC-complete unification and its application to theorem proving. In *7th RTA*, volume 1103 of *LNCS*, pages 18–32, 1996.

7 E. Contejean and C. Marché. C*i*ME: Complet*i*on modulo *E*. In *7th RTA*, volume 1103 of *LNCS*, pages 416–419, 1996.

8 W. Gehrke. Detailed catalogue of canonical term rewrite systems generated automatically. Technical report, RISC Linz, 1992.

9 A. Geser. *Relative Termination*. PhD thesis, Universität Passau, 1990.

10 G. Godoy and R. Nieuwenhuis. Paramodulation with built-in abelian groups. In *LICS 2000*, pages 413–424. IEEE Computer Society, 2000.

11 J.-P. Jouannaud and H. Kirchner. Completion of a set of rules modulo a set of equations. *SIAM Journal of Computation*, 15(4):1155–1194, 1986.

12 J.-P. Jouannaud and C. Marché. Termination and completion modulo associativity, commutativity and identity. *Theoretical Computer Science*, 104(1):29–51, 1992.

13 D. Kapur, D.R. Musser, and P. Narendran. Only prime superpositions need be considered in the Knuth-Bendix completion procedure. *JSC*, 6(1):19–36, 1988.

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

15 K. Korovin and A. Voronkov. An AC-compatible Knuth-Bendix order. In *19th CADE*, volume 2741 of *LNAI*, pages 47–59, 2003.

16 W. Küchlin. A confluence criterion based on the generalised Newman lemma. In *2nd EUROCAL*, volume 204 of *LNCS*, pages 390–399, 1985.

17 M. Kurihara and H. Kondo. Completion for multiple reduction orderings. *Journal of Automated Reasoning*, 23(1):25–42, 1999.

18 K. Kusakari. *AC-Termination and Dependency Pairs of Term Rewriting Systems.* PhD thesis, JAIST, 2000.

19 D. Lankford and A.M. Ballantyne. Decision procedures for simple equational theories with commutative-associative axioms: Complete sets of commutative-associative reductions. Technical Report ATP-39, University of Texas, 1977.

20 S. Linton and D. Shand. Some group theoretic examples with completion theorem provers. *Journal of Automated Reasoning*, 17(2):145–169, 1996.

21 C. Marché. Normalised rewriting and normalised completion. In *LICS 1994*, pages 394–403. IEEE Computer Society, 1994.

22 C. Marché. *Réécriture modulo une théorie présentée par un système convergent et décidabilité du problème du mot dans certaines classes de théories équationnelles.* PhD thesis, Université Paris-Sud, Orsay, France, 1995.

23 C. Marché. Normalized rewriting: An alternative to rewriting modulo a set of equations. *JSC*, 21(3), 1996.

24 C. Marché. Normalized rewriting: An unified view of Knuth-Bendix completion and Gröbner bases computation. *Progress in Computer Science and Applied Logic*, 15:193–208, 1998.

25 C. Marché and X. Urbain. Modular and incremental proofs of AC-termination. *JSC*, 38(1):873–897, 2004.

26 R. Nieuwenhuis and A. Rubio. Paramodulation-based theorem proving. In *Handbook of Automated Reasoning*, pages 371–443. Elsevier Science Publishers, 2001.

27 G.E. Peterson and M.E. Stickel. Complete sets of reductions for some equational theories. *Journal of the ACM*, 28(2):233–264, 1981.

28 A. Rubio. A fully syntactic AC-RPO. *Information and Computation*, 178(2):515–533, 2002.

29 U. Waldmann. Cancellative abelian monoids and related structures in refutational theorem proving (parts I and II). *JSC*, 33(6):777–861, 2002.

30 I. Wehrman and A. Stump. Mining propositional simplification proofs for small validating clauses. In *3rd PDPAR*, volume 144 of *ENTCS*, pages 79–91, 2005.

31 I. Wehrman, A. Stump, and E.M. Westbrook. Slothrop: Knuth-Bendix completion with a modern termination checker. In *17th RTA*, volume 4098 of *LNCS*, pages 287–296, 2006.

32 S. Winkler and A. Middeldorp. Termination tools in ordered completion. In *5th IJCAR*, volume 6173 of *LNAI*, pages 518–532, 2010.

33 S. Winkler and A. Middeldorp. AC completion with termination tools (system description). In *23rd CADE*, volume 6803 of *LNAI*, pages 492–498, 2011.

34 S. Winkler, H. Sato, A. Middeldorp, and M. Kurihara. Multi-completion with termination tools. *Journal of Automated Reasoning*, 2012. Accepted for publication. Available from the mascott website.

A Appendix

A.1 Proof of Lemma 11

Proof.

1. Note that all TRSs $\mathcal{C}_i \cup \mathcal{S}$ are AC-terminating. The relations $\to^+_{(\mathcal{C}_i \cup \mathcal{S})/\mathrm{AC}}$ are thus AC-compatible reduction orders, which we abbreviate by $\succ_i$. We prove the claim by induction on n, which is trivial for $n = 0$. For an $\mathcal{N}^{\mathsf{TT}}_{\mathcal{T}}$ run $(\mathcal{E}_0, \varnothing, \varnothing) \vdash^* (\mathcal{E}_n, \mathcal{R}_n, \mathcal{C}_n) \vdash (\mathcal{E}_{n+1}, \mathcal{R}_{n+1}, \mathcal{C}_{n+1})$, the induction hypothesis yields a normalized completion run $(\mathcal{E}_0, \mathcal{R}_0) \vdash^* (\mathcal{E}_n, \mathcal{R}_n)$ using reduction order $\succ_n$. Since constraint rules are never removed we have $\mathcal{C}_n \subseteq \mathcal{C}_{n+1}$, so the same run can be obtained with $\succ_{n+1}$. Case distinction on the applied $\mathcal{N}^{\mathsf{TT}}_{\mathcal{T}}$ rule shows that a step $(\mathcal{E}_n, \mathcal{R}_n) \vdash (\mathcal{E}_{n+1}, \mathcal{R}_{n+1})$ using $\succ_{n+1}$ is possible in $\mathcal{N}_{\mathcal{T}}$: If orient added the rule $s \to t$ then $s \succ_{n+1} t$ by definition, so $\mathcal{N}_{\mathcal{T}}$ can apply orient as well. If collapse$_2$ was applied, $s \succ_{n+1} r\theta$ such that a corresponding $\mathcal{N}_{\mathcal{T}}$ step is possible. In all remaining cases the step can obviously be simulated by the corresponding $\mathcal{N}_{\mathcal{T}}$ rule as no conditions on the order are involved.
2. We apply (transfinite) induction on α. For $\alpha = 0$ the claim is trivially satisfied by setting $\mathcal{C}_0 = \varnothing$. If $\alpha = n+1$ the induction hypothesis yields a $\mathcal{N}^{\mathsf{TT}}_{\mathcal{T}}$ run $(\mathcal{E}_0, \varnothing, \varnothing) \vdash^* (\mathcal{E}_n, \mathcal{R}_n, \mathcal{C}_n)$ such that $\mathcal{C}_n \subseteq \succ$. An easy case distinction on the last inference step $(\mathcal{E}_n, \mathcal{R}_n) \vdash (\mathcal{E}_{n+1}, \mathcal{R}_{n+1})$ shows that using $\succ$ for AC-termination checks allows for a corresponding $\mathcal{N}^{\mathsf{TT}}_{\mathcal{T}}$ step: If the applied inference rule is orient we have $\mathcal{E}_n = \mathcal{E}_{n+1} \cup \{s \simeq t\}$ and $\mathcal{R}_{n+1} = \mathcal{R}_n \cup \{s \to t\}$ such that $s \succ t$. Thus also $\mathcal{C}_n \cup \{s \to t\} \subseteq \succ$ is satisfied, ensuring AC-termination of the system $\mathcal{C}_n \cup \{s \to t\} \cup \mathcal{S}$. Hence the $\mathcal{N}^{\mathsf{TT}}_{\mathcal{T}}$ inference rule orient can be applied to obtain $(\mathcal{E}_n, \mathcal{R}_n, \mathcal{C}_n) \vdash (\mathcal{E}_n \setminus \{s \simeq t\}, \mathcal{R}_n \cup \{s \to t\}, \mathcal{C}_n \cup \{s \to t\})$. If the inference step $(\mathcal{E}_n, \mathcal{R}_n) \vdash (\mathcal{E}_{n+1}, \mathcal{R}_{n+1})$ applies collapse then a rewrite rule $t \to s$ is transformed into an equation $u \approx s$ using a rule $\ell \to r$ with a substitution θ. If $t \rhd_{\mathrm{AC}} l$ holds then a collapse step can be applied with $\mathcal{C}_{n+1} = \mathcal{C}_n$. Otherwise, $s \succ r\theta$ needs to be satisfied. Consequently, $\mathcal{C}_n \cup \{s \to r\theta\} \cup \mathcal{S}$ must be AC-terminating, so collapse$_2$ is applicable. In the remaining cases one can set $\mathcal{C}_{n+1} = \mathcal{C}_n$ and replace the applied rule by its $\mathcal{N}^{\mathsf{TT}}_{\mathcal{T}}$ counterpart since no conditions on the order are involved. If $\alpha = \omega$ then, by the induction hypothesis, for all runs $(\mathcal{E}_0, \varnothing) \vdash^* (\mathcal{E}_n, \mathcal{R}_n)$ with $n < \omega$ we have $(\mathcal{E}_0, \varnothing, \varnothing) \vdash^* (\mathcal{E}_n, \mathcal{R}_n, \mathcal{C}_n)$ and $\mathcal{C}_n \subseteq \succ$. Since the definitions of $\mathcal{E}_\omega$ and $\mathcal{R}_\omega$ in $\mathcal{N}_{\mathcal{T}}$ and $\mathcal{N}^{\mathsf{TT}}_{\mathcal{T}}$ coincide, for $\mathcal{C}_\omega = \bigcup_i \mathcal{C}_i$ the sequence $(\mathcal{E}_0, \varnothing, \varnothing) \vdash^* (\mathcal{E}_\omega, \mathcal{R}_\omega, \mathcal{C}_\omega)$ is a valid $\mathcal{N}^{\mathsf{TT}}_{\mathcal{T}}$ run and $\mathcal{C}_\omega \subseteq \succ$ holds.

◀

A.2 Multi-Completion with Termination Tools

Here we present the inference system $\mathcal{M}^{\mathsf{TT}}$ of normalized multi-completion with termination tools. The inference rules operate on so-called nodes, which are tuples $\langle s : t, R_0, R_1, E, C_0, C_1\rangle$ containing as data two terms s and t and as labels sets of processes R_0, R_1, E, C_0, C_1 such that $R_0 \cup C_0$, $R_1 \cup C_1$ and E are mutually disjoint. The process sets R_0, R_1 are called rewrite labels, E is the equation label and C_0, C_1 are the constraint labels. The node $\langle s : t, R_0, R_1, E, C_0, C_1\rangle$ is identified with $\langle t : s, R_1, R_0, E, C_1, C_0\rangle$. To mimic deduction of critical pairs and inter-reduction, new nodes are created while (in the latter case) others get removed, and orientation is simulated by moving processes between the different labels of a node. In any case, progress can be achieved for multiple processes at once.

Figure 4 displays the inference rules of $\mathcal{M}^{\mathsf{TT}}$ as implemented in mascott. For the sake of simplicity we assume that $\Psi(s,t) = \{s \to t\}$, as is the case for most theories under

orient
$$\frac{N \cup \{\langle s : t, R_0, R_1, E, C_0, C_1\rangle\}}{\begin{array}{r}\mathit{split}_U(N) \cup \{\langle s : t, R_0 \cup R_{lr}, R_1 \cup R_{rl}, E', C_0 \cup R_{lr}, C_1 \cup R_{rl}\rangle\} \\ \cup\, \{\langle u : v, \varnothing, \varnothing, R_{lr}, \varnothing, \varnothing\rangle \mid u \approx v \in \Theta(s,t)\} \\ \cup\, \{\langle u : v, \varnothing, \varnothing, R_{rl}, \varnothing, \varnothing\rangle \mid u \approx v \in \Theta(t,s)\}\end{array}}$$

if $s = s{\downarrow}$, $t = t{\downarrow}$, $E_{lr}, E_{rl} \subseteq E$, $E' = E \setminus (E_{lr} \cup E_{rl})$, $C[N,p] \cup \mathcal{S} \cup \{s \to t\}$ is AC-terminating for all $p \in E_{lr}$ and $C[N,p] \cup \mathcal{S} \cup \{t \to s\}$ is AC-terminating for all $p \in E_{rl}$, $U = E_{lr} \cap E_{rl}$, $E_{lr} \cup E_{rl} \neq \varnothing$, $R_{lr} = (E_{lr} \setminus E_{rl}) \cup \{p0 \mid p \in U\}$, and $R_{rl} = (E_{rl} \setminus E_{lr}) \cup \{p1 \mid p \in U\}$

delete
$$\frac{N \cup \{\langle s : t, \varnothing, \varnothing, E, \varnothing, \varnothing\rangle\}}{N}$$

if $s \leftrightarrow^*_{\mathrm{AC}} t$ and $E \neq \varnothing$

deduce
$$\frac{N}{N \cup \{\langle s : t, \varnothing, \varnothing, R \cap R', \varnothing, \varnothing\rangle\}}$$

if there exist nodes $\langle \ell : r, R, \ldots\rangle$ and $\langle \ell' : r', R', \ldots\rangle$ in N such that $s \approx t \in \mathrm{CP}_{(}\ell \to r, \ell' \to r')$ and $R \cap R' \neq \varnothing$

normalize
$$\frac{N \cup \{\langle s : t, R_0, R_1, E, C_0, C_1\rangle\}}{N \cup \{\langle s{\downarrow} : t{\downarrow}, R_0, R_1, E, C_0, C_1\rangle\}}$$

rewrite$_1$
$$\frac{N \cup \{\langle s : t, R_0, R_1, E, C_0, C_1\rangle\}}{\begin{array}{l}N \cup \{\langle s : t, R_0 \setminus R, R_1, E \setminus R, C_0, C_1\rangle\} \\ \quad \cup\, \{\langle s : u, R_0 \cap R, \varnothing, (R_1 \cup E) \cap R, \varnothing, \varnothing\rangle\} \\ \quad \cup\, \{\langle s : r\theta, \varnothing, \varnothing, \varnothing, R_1 \cap R, \varnothing\rangle\}\end{array}}$$

if $\langle \ell : r, R \cup R', \ldots\rangle \in N$, $t \xrightarrow[\ell \to r \setminus \mathcal{S}]{\theta} u$, $t \doteq \ell$, $C[N,p] \cup \mathcal{S} \cup \{s \to r\theta\}$ is AC-terminating for all $p \in R_1 \cap R$, and $R \cap (R_0 \cup R_1 \cup E) \neq \varnothing$

rewrite$_2$
$$\frac{N \cup \{\langle s : t, R_0, R_1, E, C_0, C_1\rangle\}}{\begin{array}{l}N \cup \{\langle s : t, R_0 \setminus R, R_1 \setminus R, E \setminus R, C_0, C_1\rangle\} \\ \quad \cup\, \{\langle s : u, R_0 \cap R, \varnothing, (R_1 \cup E) \cap R, \varnothing, \varnothing\rangle\}\end{array}}$$

if $\langle \ell : r, R, \ldots\rangle \in N$, $t \xrightarrow[\ell \to r \setminus \mathcal{S}]{} u$, $t \rhd \ell$, and $R \cap (R_0 \cup R_1 \cup E) \neq \varnothing$

Figure 4 Normalized multi-completion with termination tools ($\mathcal{M}^{\mathsf{TT}}$).

consideration. Otherwise, an orient step needs to add all nodes $\langle \ell : r, R_{lr}, \varnothing, \varnothing, \varnothing, \varnothing\rangle$ such that $\ell \to r \in \Psi(s,t)$ and $\langle \ell' : r', R_{rl}, \varnothing, \varnothing, \varnothing, \varnothing\rangle$ such that $\ell' \to r' \in \Psi(t,s)$, but do not add R_{lr} and R_{rl} to rewrite labels of the node with data $s : t$. Here, for a node $n = \langle s : t, R_0, R_1, E, C_0, C_1\rangle$ and a process p, the constraint projection $C[n,p]$ is defined as $\{s \to t\}$ if $p \in C_0$, $\{t \to s\}$ if $p \in C_1$ and $\varnothing$ otherwise. This is extended to a set of nodes N by setting $C[N,p] = \bigcup_{n \in N} C[n,p]$.

The rewrite rule projection $R[N,p]$ is defined analogous to $C[N,p]$. A sequence of $\mathcal{M}^{\mathsf{TT}}$ inference steps $N_0 \vdash N_1 \vdash N_2 \vdash \cdots$ is again called a run. Given a set of equations $\mathcal{E}$, the initial node set $N_0 = N_{\mathcal{E}}$ consists of all nodes $\langle s : t, \varnothing, \varnothing, \{\epsilon\}, \varnothing, \varnothing\rangle$ such that $s \approx t$ is in $\mathcal{E}$. A finite run γ of the form $N_0 \vdash^* N$ succeeds for process p if p occurs in N but not in any equation label of a node in N. If γ succeeds for no process in N then it fails.

Similar as for standard completion [34], it can be shown that every $\mathcal{M}^{\mathsf{TT}}$ run γ of the form $N_0 \vdash^* N$ corresponds to a valid $\mathcal{N}_{\mathcal{T}}^{\mathsf{TT}}$ run γ_p for every process p occurring in N. An $\mathcal{M}^{\mathsf{TT}}$ run is fair for a process p if the projected run γ_p is fair. We conclude by stating a correctness result for $\mathcal{M}^{\mathsf{TT}}$.

▶ **Theorem 15.** *Let $N_{\mathcal{E}}$ be the initial node set for a set of equations $\mathcal{E}$ and let γ be a finite non-failing $\mathcal{M}^{\mathsf{TT}}$ run of the form $N_{\mathcal{E}} \vdash^* N$ which uses general $\mathcal{S}$-normalizing pairs and is fair for some process p occurring in N. Then $R[N,p]$ is $\mathcal{T}$-convergent.* ◀