Conditional Narrowing Modulo in Rewriting Logic and Maude*

Luis Aguirre, Narciso Martí-Oliet, Miguel Palomino, and Isabel Pita

Facultad de Informática, Universidad Complutense de Madrid, Spain
{luisagui, narciso, miguelpt, ipandreu}@ucm.es

Abstract. This work studies the relationship between verifiable and computable answers for reachability problems in rewrite theories with an underlying membership equational logic. These problems have the form $(\exists \bar{x}) s(\bar{x}) \rightarrow^* t(\bar{x})$, with $\bar{x}$ some variables, or a conjunction of several of these subgoals. A calculus that solves this kind of problems has been developed and proved correct. Given a reachability problem in a rewrite theory, this calculus can compute any normalized answer that can be checked by rewriting, or a more general one. Special care has been taken in the calculus to keep membership information attached to each term, using this information whenever possible.

Keywords: Maude, narrowing, reachability, rewriting logic, unification, membership equational logic

1 Introduction

Rewriting logic is a computational logic that has been around for more than twenty years [Mes90], whose semantics [BM06] has a precise mathematical meaning allowing mathematical reasoning for property proving, providing a more flexible framework for the specification of concurrent systems. It turned out that it can express both concurrent computation and logical deduction, allowing its application in many areas such as automated deduction, software and hardware specification and verification, security, etc. One important property of rewriting logic is reflection [CM96]. Intuitively, reflection means representing a logic's metalevel at the object level, allowing the definition of strategies that guide rule application in an object-level theory.

Reachability problems have the form $(\exists \bar{x}) s(\bar{x}) \rightarrow^* t(\bar{x})$, with $\bar{x}$ some variables, or a conjunction of several of these subgoals. They can be solved by model checking methods for finite state spaces. A technique known as *narrowing* [Fay78] that was first proposed as a method for solving equational goals (*unification*), has been extended to cover also reachability goals [MT07], leaving equational goals as a special case of reachability goals. In recent years the idea of *variants of a term* has been applied to narrowing. A strategy for order-sorted unconditional

* Research supported by MINECO Spanish project StrongSoft (TIN2012–39391–C04–04) and Comunidad de Madrid program PROMETIDOS (S2009/TIC-1465).

rewrite theories known as *folding variant narrowing* [ESM12], which computes a complete set of variants of any term, has been developed by Escobar, Sasse and Meseguer, allowing unification *modulo* a set of equations and axioms. The strategy terminates on any input term on those systems enjoying the *finite variant property*, and it is *optimally terminating*. It is being used for cryptographic protocol analysis [MT07], with tools like Maude-NPA [EMM05], termination algorithms modulo axioms [DLM+08], and algorithms for checking confluence and coherence of rewrite theories modulo axioms, such as the Church-Rosser (CRC) and the Coherence (ChC) Checkers for Maude [DM12].

This work explores narrowing for membership conditional rewrite theories, going beyond the scope of folding variant narrowing which works on order-sorted unconditional rewrite theories. A calculus that computes answers to reachability problems in membership conditional rewrite theories has been developed and proved correct with respect to idempotent normalized answers.

The work is structured as follows: in Section 2 all needed definitions and properties for rewriting and narrowing are introduced. Section 3 introduces the first part of the narrowing calculus, the one that deals with equational unification. Section 4 introduces the part of the calculus dealing with reachability and its proof of correctness. Section 5 shows the calculus at work. In Section 6, related work, conclusions and current lines of investigation for this work are presented. An extended version of this paper, with all the missing proofs, can be found at `http://maude.sip.ucm.es/cnarrowing/`, together with a previous version of this work with transformation rules and a prototype.

2 Preliminaries

We assume familiarity with rewriting logic [BM06]. There are several language implementations of rewriting logic, including Maude [CDE+07]. Rewriting logic is parameterized by an underlying equational logic. In Maude's case this logic is membership equational logic [Mes97].

2.1 Tower of Hanoi example

Throughout this paper the Tower of Hanoi puzzle will be used as a motivating example to explain the definitions in a less abstract way. We have `Rods` `a`, `b` and `c`, and `Disks` `1`, `2`, `3` and `4` which can slide onto any `Rod`. We call a `Rod` with zero or more stacked `Disks` (written juxtaposed) a `Tower`. If smaller `Disks` are always stacked on top of bigger `Disks` we have a `ValidTower` (abbreviated `VT`). A set of valid towers (written separated by commas) is a `State` (abbreviated `St`). A `move` between a `Pair` of towers (written separated by a $-$ symbol) is defined by the rules: 1) only one `Disk` may be moved at a time, 2) each move consists of taking the upper `Disk` from one `Tower` and placing it on top of another `Tower`, and 3) `Disk` X may be placed on top of `Disk` Y only if X is smaller than Y (written $X < Y = \texttt{t}$, where `t` is the *true* `Boolean` value). The goal of the puzzle is to reach a desired `State` from a given initial `State`.

2.2 Membership equational logic

A *membership equational logic* (MEL) *signature* [BM06] is a triple $\Sigma = (K, \Omega, S)$, with K a set of *kinds*, $\Omega = \{\Sigma_{w;k}\}_{(w;k)\in K^* \times K}$ a many-kinded algebraic signature, and $S = \{S_k\}_{k\in K}$ a K-kinded family of disjoint sets of sorts. For simplicity, we only allow overloading of operators whenever the result belongs in the same kind. The kind of a sort s is denoted by $[s]$. The sets $T_{\Sigma,s}$, $T_\Sigma(X)_s$, $T_{\Sigma,k}$ and $T_\Sigma(X)_k$ denote, respectively, the set of ground Σ-terms with sort s, the set of Σ-terms with sort s over the set X of *sorted* variables, the set of ground Σ-terms with kind k and the set of Σ-terms with kind k over the set X of *sorted* variables. We write T_Σ, $T_\Sigma(X)$ for the corresponding term algebras. $vars(t) \subseteq X$ denotes the set of variables in $t \in T_\Sigma(X)$.

In the Tower of Hanoi puzzle, $\Sigma = (K, \Omega, S)$ is: K=$\{\mathtt{TS}, \mathtt{P}, \mathtt{D}, \mathtt{B}\}$,
Ω=$\{\cdot_{\mathtt{D\ TS;TS}},\ ,_{\mathtt{TS\ TS;TS}},\ -_{\mathtt{TS\ TS;P}},\ \mathtt{move}_{\mathtt{P;P}},\ <_{\mathtt{D\ D;B}}\}$, S=$\{S_{\mathtt{TS}}, S_{\mathtt{P}}, S_{\mathtt{D}}, S_{\mathtt{B}}\}$, $S_{\mathtt{D}}$=$\{\mathtt{Disk}\}$,
$S_{\mathtt{TS}}$=$\{\mathtt{Rod}, \mathtt{VT}, \mathtt{Tower}, \mathtt{St}\}$, $S_{\mathtt{P}}$=$\{\mathtt{Pair}\}$, $S_{\mathtt{D}}$=$\{\mathtt{Disk}\}$, $S_{\mathtt{B}}$=$\{\mathtt{Boolean}\}$.
$\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$, $\{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{4}\}$, and $\{\mathtt{t}\}$ are the *atoms* with sort `Rod`, `Disk`, and `Boolean` respectively.

Positions in a term t: we represent the root of t as ϵ and the other positions as strings of nonzero natural numbers in the usual way, considering t as a tree. The set of positions of a term is written $Pos(t)$. $t|_p$ is the subtree below position p. $t[u]_p$ is the replacement in t of the subterm at position p with term u.

A *substitution* $\sigma : Y \to T_\Sigma(X)$ is a function from a finite set of sorted variables $Y \subseteq X$ to $T_\Sigma(X)$ such that $\sigma(y)$ has the same or lower sort as that of the variable $y \in Y$ ($s_1 \leq s_2$, formally defined in the next paragraph). Substitutions are written as σ=$\{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$ where $Dom(\sigma)$=$\{x_1, \ldots, x_n\}$ and $Ran(\sigma)$=$\bigcup_{i=1}^n vars(t_i)$. The identity substitution is id. The restriction of σ to a set of variables V is $\sigma|_V$. Composition of two substitutions is denoted by $\sigma\sigma'$. For substitutions σ and σ' where $Dom(\sigma) \cap Dom(\sigma')$=$\emptyset$, we denote their union by $\sigma \cup \sigma'$.

A MEL theory [BM06] is a pair $(\Sigma, \mathcal{E})$, where Σ is a MEL signature and $\mathcal{E}$ is a finite set of MEL sentences, either conditional equations or conditional memberships of the forms:

$$(\forall X)\ t{=}t' \ \mathtt{if} \bigwedge_i A_i, \quad (\forall X)\ t{:}s \ \mathtt{if} \bigwedge_i A_i$$

for $t, t' \in T_\Sigma(X)_k$ and $s \in S_k$, the latter stating that t is a term of sort s, provided the condition holds, and each A_i can be of the form $t{=}t'$, $t{:}s$ or $t{:=}t'$ (a *matching* equation). Matching equations are treated as ordinary equations, but they impose a limitation in the syntax of admissible MEL theories, as we will see. We also admit unconditional sentences in $\mathcal{E}$. Order-sorted (*sugared*) notation $s_1 \leq s_2$ can be used instead of $(\forall x{:}[s_1])\ x{:}s_2$ `if` $x{:}s_1$. An operator declaration $f : s_1 \times \cdots \times s_n \to s$ corresponds to declaring f at the kind level and giving the membership axiom $(\forall x_1{:}[s_1], \ldots, x_n{:}[s_n])\ f(x_1, \ldots, x_n){:}s$ `if` $\bigwedge_{1\leq i\leq n} x_i{:}s_i$. Given a MEL sentence ϕ, we denote by $\mathcal{E} \vdash \phi$ that ϕ can be deduced from $\mathcal{E}$ using the rules in Figure 1, where $=$ can be either $=$ or $:=$ as explained before [BM12]. The rules of Figure 1 specify a sound and complete calculus. A MEL

$$\frac{t \in T_\Sigma(X)}{(\forall X)t = t}\ \texttt{Reflexivity} \qquad \frac{(\forall X)t = t'}{(\forall X)t' = t}\ \texttt{Symmetry}$$

$$\frac{(\forall X)t_1 = t_2 (\forall X)t_2 = t_3}{(\forall X)t_1 = t_3}\ \texttt{Transitivity} \qquad \frac{(\forall X)t' {:} s \quad (\forall X)t {=} t'}{(\forall X)t {:} s}\ \texttt{Membership}$$

$$\frac{f \in \Sigma_{k_1 \cdots k_n, k} \ (\forall X)t_i {=} t_i' \quad t_i, t_i' {\in} T_\Sigma(X)_{k_i}, 1 {\leq} i {\leq} n}{(\forall X) f(t_1, \ldots, t_n) = f(t_1', \ldots, t_n')}\ \texttt{Congruence}$$

$$\frac{((\forall X)\ A_0\ \texttt{if}\ \bigwedge_i A_i) {\in} E \quad \theta {:} X {\to} T_\Sigma(Y) \quad (\forall Y) A_i\theta}{(\forall Y) A_0\theta}\ \texttt{Replacement}$$

Fig. 1. Deduction rules for membership equational logic.

theory $(\Sigma, \mathcal{E})$ has an *initial algebra*, denoted by $T_{\Sigma/\mathcal{E}}$, whose elements are the equivalence classes $[t]_\mathcal{E} \subseteq T_\Sigma$ of ground terms identified by the equations in $\mathcal{E}$.

The MEL theory for the Tower of Hanoi puzzle consists of $\Sigma = (K, \Omega, S)$ and the following set $\mathcal{E}$ of MEL sentences where we omit the universal quantifiers:

$X : \texttt{St}$ *if* $X : \texttt{VT}$; $X : \texttt{Tower}$ *if* $X : \texttt{VT}$; $X : \texttt{St}$ *if* $X : \texttt{Rod}$; $X : \texttt{Tower}$ *if* $X : \texttt{Rod}$;
$X : \texttt{St}$ *if* $X : \texttt{Rod}$; $X : \texttt{VT}$ *if* $X : \texttt{Rod}$; $XY : \texttt{Tower}$ *if* $X : \texttt{Disk} \wedge Y : \texttt{Tower}$;
$X, Y : \texttt{St}$ *if* $X : \texttt{St} \wedge Y : \texttt{St}$; $X, Y = Y, X$; $(X, Y), Z = X, (Y, Z)$;
$X - Y : \texttt{Pair}$ *if* $X : \texttt{Tower} \wedge Y : \texttt{Tower}$; $X - Y = Y - X$;
$X < Y : \texttt{Boolean}$ *if* $X : \texttt{Disk} \wedge Y : \texttt{Disk}$; $XR : \texttt{VT}$ *if* $X : \texttt{Disk} \wedge R : \texttt{Rod}$;
$XYT : \texttt{VT}$ *if* $X : \texttt{Disk} \wedge Y : \texttt{Disk} \wedge T : \texttt{Tower} \wedge X {<} Y = \texttt{t} \wedge YT : \texttt{Vt}$;
$\texttt{1} < \texttt{2} = \texttt{t}$; $\texttt{1} < \texttt{3} = \texttt{t}$; $\texttt{1} < \texttt{4} = \texttt{t}$; $\texttt{2} < \texttt{3} = \texttt{t}$; $\texttt{2} < \texttt{4} = \texttt{t}$; $\texttt{3} < \texttt{4} = \texttt{t}$;
$\texttt{move}(XT - R) = T - XR$ *if* $X : \texttt{Disk} \wedge T : \texttt{Tower} \wedge R : \texttt{Rod}$;
$\texttt{move}(XT - YT') = T - XYT'$ *if* $X : \texttt{Disk} \wedge Y : \texttt{Disk} \wedge T : \texttt{Tower} \wedge$
$\wedge\ T' : \texttt{Tower} \wedge \texttt{X} < \texttt{Y} = \texttt{t}$; $\texttt{move}(X) : \texttt{Pair}$ *if* $X : \texttt{Pair}$.

A single `Disk` stacked on a `Rod` is always a `ValidTower`. For multiple `Disks`, we compare them recursively. The operator `move` distinguishes between two cases: if one `Tower` is empty, i.e. a `Rod`, then we can stack any `Disk` on it; else the sizes of the top `Disks` on each `Tower` must be compared (`<`) and we can stack the smaller one on top of the other.

2.3 Rewriting logic

A rewrite theory $\mathcal{R} = (\Sigma, \mathcal{E}, R)$ consists of a MEL theory $(\Sigma, \mathcal{E})$ together with a finite set R of *conditional rewrite rules* each of which has the form

$$(\forall X)\ l \to r\ \texttt{if}\ \bigwedge_i p_i {=} q_i \wedge \bigwedge_j w_j {:} s_j \wedge \bigwedge_k l_k \to r_k,$$

where l, r are Σ-terms of the same kind and $=$ can be either $=$ or $:=$. Rewrite rules can also be unconditional.

Such a rewrite rule specifies a *one-step transition* from a state $t[l\theta]_p$ to the state $t[r\theta]_p$, denoted by $t[l\theta]_p \to^1_R t[r\theta]_p$, provided the condition holds. The subterm $t|_p$ is called a *redex*.

$$\frac{t \in T_\Sigma(X)}{(\forall X) t \to t}\ \texttt{Reflexivity} \qquad \frac{(\forall X) t_1 \to t_2,\ (\forall X) t_2 \to t_3}{(\forall X) t_1 \to t_3}\ \texttt{Transitivity}$$

$$\frac{f \in \Sigma_{k_1 \cdots k_n, k}\ (\forall X) t_i \to t_i' \quad t_i, t_i' \in T_\Sigma(X)_{k_i}, 1 \leq i \leq n}{(\forall X) f(t_1, \ldots, t_n) \to f(t_1', \ldots, t_n')}\ \texttt{Congruence}$$

$$((\forall X)\ l \to r\ \texttt{if}\ \bigwedge_i p_i = q_i \wedge \bigwedge_j w_j : s_j \wedge \bigwedge_k l_k \to r_k) \in R$$

$$\frac{\theta : X \to T_\Sigma(Y) \quad \bigwedge_i \mathcal{E} \vdash (\forall Y) p_i\theta = q_i\theta \quad \bigwedge_j \mathcal{E} \vdash (\forall Y) w_j\theta : s_j \quad \bigwedge_k (\forall Y) l_k\theta \to r_k\theta}{(\forall Y) l\theta \to r\theta}\ \texttt{Replace}$$

Fig. 2. Deduction rules for rewrite theories.

In the example, R has as only element the conditional rewrite rule: $D, E \to F, G\ if\ D : \texttt{Tower} \wedge E : \texttt{Tower} \wedge F - G := \texttt{move}(D - E) \wedge F : \texttt{Tower} \wedge G : \texttt{Tower}$.

F and G are new variables on the right side of the rule. They are instantiated by matching on the conditional part of the rule.

The inference rules in Figure 2 for rewrite theories can infer all possible computations in the system specified by $\mathcal{R}$ [BM12]. We can *reach* a state v from a state u if we can prove $\mathcal{R} \vdash u \to v$.

The relation $\to^1_{R/\mathcal{E}}$ on $T_\Sigma(X)$ is $=_\mathcal{E} \circ \to^1_R \circ =_\mathcal{E}$. $\to^1_{R/\mathcal{E}}$ on $T_\Sigma(X)$ induces a relation $\to^1_{R/\mathcal{E}}$ on $T_{\Sigma/\mathcal{E}}(X)$, the equivalence relation modulo $\mathcal{E}$, by $[t]_\mathcal{E} \to^1_{R/\mathcal{E}} [t']_\mathcal{E}$ iff $t \to^1_{R/\mathcal{E}} t'$. The transitive (resp. transitive and reflexive) closure of $\to^1_{R/\mathcal{E}}$ is denoted $\to^+_{R/\mathcal{E}}$ (resp. $\to^*_{R/\mathcal{E}}$). We say that a term t is $\to_{R/\mathcal{E}}$*-irreducible* (or just $R/\mathcal{E}$-irreducible) if there is no term t' such that $t \to^1_{R/\mathcal{E}} t'$.

A rewrite rule $l \to r$ `if` *cond*, is *sort-decreasing* if for each substitution σ, we have that for any sort s if $l\sigma \in T_\Sigma(X)_s$ and $(cond)\sigma$ is verified implies $r\sigma \in T_\Sigma(X)_s$. A Σ-equation $t = t'$ is *regular* if $Var(t) = Var(t')$. It is *sort-preserving* if for each substitution σ, we have $t\sigma \in T_\sigma(X)_s$ implies $t'\sigma \in T_\sigma(X)_s$ and vice versa.

A substitution is called $\mathcal{E}$*-normalized* (or normalized) if $x\sigma$ is $\mathcal{E}$-irreducible for all $x \in V$.

The relation $\to^1_{R/\mathcal{E}}$ is *terminating* if there are no infinite rewriting sequences. The relation $\to^1_{R/\mathcal{E}}$ is *operationally terminating* if there are no infinite well-formed proof trees. The relation $\to^1_{R/\mathcal{E}}$ is *confluent* if whenever $t \to^*_{R/\mathcal{E}} t'$ and $t \to^*_{R/\mathcal{E}} t''$, there exists a term t''' such that $t' \to^*_{R/\mathcal{E}} t'''$ and $t'' \to^*_{R/\mathcal{E}} t'''$. In a confluent, terminating, sort-decreasing, membership rewrite theory, for each term $t \in T_\Sigma(X)$, there is a unique (up to $\mathcal{E}$-equivalence) $R/\mathcal{E}$-irreducible term t' obtained by rewriting to *canonical* form, denoted by $t \to^!_{R/\mathcal{E}} t'$, or $t \downarrow_{R/\mathcal{E}}$ when t' is not relevant, which we call $can_{R/\mathcal{E}}(t)$.

2.4 Executable rewrite theories

For a rewrite theory $\mathcal{R} = (\Sigma, \mathcal{E}, R)$, whether a one step rewrite $t \rightarrow^1_{R/\mathcal{E}} t'$ holds is undecidable in general. We impose additional conditions, similar to those required for functional and system modules in Maude, under which we can decide if $t \rightarrow^1_{R/\mathcal{E}} t'$ holds. We decompose $\mathcal{E}$ into a disjoint union $E \cup A$, with A a set of equational axioms (such as associativity, and/or commutativity, and/or identity). We define the relation $\rightarrow^1_{E,A}$ on $T_\Sigma(X)$ as follows: $t \rightarrow^1_{E,A} t'$ if there is a position $\omega \in Pos(t)$, an equation $l = r$ `if` $cond \in E$, and a substitution σ such that $t|_\omega =_A l\sigma$ (A-matching), $(cond)\sigma$ is satisfied, and $t' = t[r\sigma]_\omega$. The relation $\rightarrow^1_{R,A}$ is similarly defined. We define $\rightarrow^1_{R\cup E,A}$ as $\rightarrow^1_{R,A} \cup \rightarrow^1_{E,A}$. A rewrite theory $\mathcal{R} = (\Sigma, E \cup A, R)$ is *executable* if each kind k in Σ is nonempty, E, A, and R are finite and the following conditions hold:

1. E and R are *admissible* [CDE+07]. Then we have a *deterministic 3-CTRS* [Ohl02]. Any new variable in the conditions will be instantiated by matching. New variables are distinguished in Maude by using a := symbol instead of = in the condition. They appear on the left terms of these *matching equations*. Conditions in deterministic 3-CTRS's *must* be solved in left to right order.
2. Equality modulo A is decidable and there exists a finite *matching algorithm modulo A*.
3. The equations in E are *sort-decreasing*, and *terminating and confluent modulo A* when we consider them as oriented rules, where $\rightarrow^1_{E/A}$ is defined in the same way as we did for $\rightarrow^1_{R/\mathcal{E}}$.
4. $\rightarrow_{E,A}$ is *coherent with A*, i.e., $\forall t_1, t_2, t_3$ we have $t_1 \rightarrow^+_{E,A} t_2$ and $t_1 =_A t_3$ implies $\exists t_4, t_5$ such that $t_2 \rightarrow^*_{E,A} t_4, t_3 \rightarrow^+_{E,A} t_5$ and $t_4 =_A t_5$ [MT07].

$$\begin{array}{ccc} t_1 & \rightarrow^+_{E,A} & t_2 \\ A & & \\ t_3 & \longrightarrow^+_{E,A} & \end{array} \qquad \begin{array}{ccc} \rightarrow^*_{E,A} & t_4 \\ & A \\ & t_5 \end{array}$$

5. $\rightarrow_{R,A}$ is *$\mathcal{E}$-consistent with A*, i.e., $\forall t_1, t_2, t_3$ we have $t_1 \rightarrow_{R,A} t_2$ and $t_1 =_A t_3$ implies $\exists t_4$ such that $t_3 \rightarrow_{R,A} t_4$ and $t_2 =_\mathcal{E} t_4$. Also $\rightarrow_{R,A}$ is *$\mathcal{E}$-consistent with $\rightarrow_{E,A}$*, i.e., $\forall t_1, t_2, t_3$ we have $t_1 \rightarrow_{R,A} t_2$ and $t_1 \rightarrow^*_{E,A} t_3$ implies $\exists t_4, t_5$ such that $t_3 \rightarrow^*_{E,A} t_4$ and $t_4 \rightarrow_{R,A} t_5$ and $t_2 =_\mathcal{E} t_5$. In both cases the $\rightarrow_{R,A}$ rewriting steps from t_3 and t_4 must be performed with the *same* rule that was applied to t_1 [MT07].

$$\begin{array}{ccc} t_1 & \rightarrow_{R,A} & t_2 \\ A & & \mathcal{E} \\ t_3 & \rightarrow_{R,A} & t_4 \end{array} \qquad \begin{array}{ccccc} t_1 & & \longrightarrow_{R,A} & & t_2 \\ \downarrow^*_{E,A} & & & & \mathcal{E} \\ t_3 & \rightarrow^*_{E,A} & t_4 & \rightarrow_{R,A} & t_5 \end{array}$$

(a) $\mathcal{E}$-consistency of $\rightarrow_{R,A}$ with A (b) $\mathcal{E}$-consistency of $\rightarrow_{R,A}$ with $\rightarrow_{E,A}$

Technically, what coherence means is that the weaker relation $\rightarrow^1_{E,A}$ becomes semantically equivalent to the stronger relation $\rightarrow^1_{E/A}$, so we can decide $t \rightarrow^1_{R/\mathcal{E}}$

t' by finding t'' such that $can_{E,A}(t) \to_R^1 t''$ and $can_{E,A}(t') =_A can_{E,A}(t'')$, which is decidable, since the number of rules is finite and A-matching is decidable and finite.

Under these conditions we can implement $\to_{R/\mathcal{E}}$ on terms using $\to_{R \cup E,A}$ [MT07]. This lemma links $\to_{R/\mathcal{E}}$ with $\to_{E,A}$ and $\to_{R,A}$. Patrick Viry gave a proof for unsorted unconditional rewrite theories [Vir94], which can easily be lifted to our membership conditional case.

Lemma 1. *Let $\mathcal{R} = (\Sigma, \mathcal{E}, R)$ be an executable rewrite theory, that is, it has all the properties specified in Section 2.4. Then $t_1 \to_{R/\mathcal{E}} t_2$ if and only if $t_1 \to_{E,A}^* \to_{R,A} t_3$ for some $t_3 =_{\mathcal{E}} t_2$.*

The rewrite theory for the Tower of Hanoi puzzle is executable if we decompose $\mathcal{E}$ in the following way: the set A has as elements the associative equation and the commutative equations in $\mathcal{E}$; the set E has as elements the rest of equations and all memberships in $\mathcal{E}$, and we add to R the following rule needed for $\mathcal{E}$-consistency:
$D, E, S \to F, G, S \; if \; D : \texttt{Tower} \wedge E : \texttt{Tower} \wedge S : \texttt{State} \wedge \; F - G := \texttt{move}(D - E) \wedge$
$\wedge F : \texttt{Tower} \wedge G : \texttt{Tower}$.

2.5 Unification

Given a rewrite theory $\mathcal{R} = (\Sigma, \mathcal{E}, R)$, a *$\Sigma$-equation* is an expression of the form $t = t'$ where $t, t' \in T_\Sigma(X)_s$ for an appropriate s. The *$\mathcal{E}$-subsumption* preorder $\ll_{\mathcal{E}}$ on $T_\Sigma(X)_s$ is defined by $t \ll_{\mathcal{E}} t'$ if there is a substitution σ such that $t =_{\mathcal{E}} t'\sigma$. For substitutions σ, ρ and a set of variables V we define $\sigma|_V \ll_{\mathcal{E}} \rho|_V$ if there is a substitution η such that $\sigma|_V =_{\mathcal{E}} (\rho\eta)|_V$. Then we say that ρ is more general than σ with respect to V. When V is not specified, we assume that $V = Dom(\sigma) = Dom(\rho)$ and we say that ρ is more general than σ.

A *system of equations* F is a conjunction of the form $t_1 = t_1' \wedge \ldots \wedge t_n = t_n'$ where for $1 \leq i \leq n$, $t_i = t_i'$ is a Σ-equation. We define $Var(F) = \bigcup_i Var(t_i) \cup Var(t_i')$. An *$\mathcal{E}$-unifier* for F is a substitution σ such that $t_i\sigma =_{\mathcal{E}} t_i'\sigma$ for $1 \leq i \leq n$. For $V = Var(F) \subseteq W$, a set of substitutions $CSU_{\mathcal{E}}^W(F)$ is said to be a *complete set of unifiers modulo $\mathcal{E}$* of F away from W if

- each $\sigma \in CSU_{\mathcal{E}}^W(F)$ is an $\mathcal{E}$-unifier of F;
- for any $\mathcal{E}$-unifier ρ of F there is a $\sigma \in CSU_{\mathcal{E}}^W(F)$ such that $\rho|_V \ll_{\mathcal{E}} \sigma|_V$;
- for all $\sigma \in CSU_{\mathcal{E}}^W(F)$, $Dom(\sigma) \subseteq V$ and $Ran(\sigma) \cap W = \emptyset$.

An $\mathcal{E}$-unification algorithm is *complete* if for any given system of equations it generates a complete set of $\mathcal{E}$-unifiers, which may not be finite. A unification algorithm is said to be *finite* and complete if it terminates after generating a finite and complete set of solutions.

2.6 Reachability goals

Given a rewrite theory $\mathcal{R} = (\Sigma, \mathcal{E}, R)$, a *reachability goal* G is a conjunction of the form $t_1 \to^* t_1' \wedge \ldots \wedge t_n \to^* t_n'$ where for $1 \leq i \leq n$, $t_i, t_i' \in T_\Sigma(X)_{s_i}$ for

appropriate s_i. We define $Var(G) = \bigcup_i Var(t_i) \cup Var(t'_i)$. A substitution σ is a *solution* of G if $t_i\sigma \rightarrow^*_{R/\mathcal{E}} t'_i\sigma$ for $1 \leq i \leq n$. We define $\mathrm{E}(G)$ to be the system of equations $t_1 = t'_1 \wedge \ldots \wedge t_n = t'_n$. We say σ is a *trivial solution* of G if it is an $\mathcal{E}$-unifier for $\mathrm{E}(G)$. We say G is trivial if the identity substitution id is a trivial solution of G.

For goals $G : t_1 \rightarrow^* t_2 \wedge \ldots \wedge t_{2n-1} \rightarrow^* t_{2n}$ and $G' : t'_1 \rightarrow^* t'_2 \wedge \ldots \wedge t'_{2n-1} \rightarrow^* t'_{2n}$ we say $G =_{\mathcal{E}} G'$ if $t_i =_{\mathcal{E}} t'_i$ for $1 \leq i \leq 2n$. We say $G \rightarrow_R G'$ if there is an odd i such that $t_i \rightarrow_R t'_i$ and for all $j \neq i$ we have $t_j = t'_j$. That is, G and G' differ only in one subgoal ($t_i \rightarrow t_{i+1}$ vs $t'_i \rightarrow t_{i+1}$), but $t_i \rightarrow t'_i$, so when we rewrite t_i in G to t'_i we get G'. The relation $\rightarrow_{R/\mathcal{E}}$ over goals is defined as $=_{\mathcal{E}} \circ \rightarrow_R \circ =_{\mathcal{E}}$.

2.7 Narrowing

Let t be a Σ-term and W be a set of variables such that $Var(t) \subseteq W$. The R, A-*narrowing* relation on $T_\Sigma(X)$ is defined as follows: $t \rightsquigarrow_{p,\sigma,R,A} t'$ if there is a non-variable position $p \in Pos_\Sigma(t)$, a rule $l \rightarrow r$ `if` *cond* in R, properly renamed, such that $Var(l) \cap W = \emptyset$, and a unifier $\sigma \in CSU_A^{W'}(t|_p = l)$ for $W' = W \cup Var(l)$, such that $t' = (t[r]_p)\sigma$ and $(cond)\sigma$ holds. Similarly E, A-narrowing and $R \cup E, A$-narrowing relations are defined.

2.8 Associated rewrite theory

Any executable Mel theory $(\Sigma, E \cup A)$ has a corresponding rewrite theory $\mathcal{R}_E = (\Sigma', A, R_E)$ associated to it [DLM+08]: we add a fresh new kind *Truth* with a constant tt to Σ, and for each kind $k \in K$ an operator $eq : k\ k \rightarrow Truth$. $\top$ represents a conjunction of any number of tt's. There are rules $eq(x{:}k, x{:}k) \rightarrow tt$ for each kind $k \in K$. For each conditional equation or membership in E the set R_E has a conditional rule or membership of the form

$$t \rightarrow t' \ \textit{if}\ A_1^\bullet \wedge \ldots \wedge A_n^\bullet \qquad t{:}s \ \textit{if}\ A_1^\bullet \wedge \ldots \wedge A_n^\bullet$$

where if A_i is a membership then $A_i^\bullet{=}A_i$, if $A_i \equiv t_i{:=}t'_i$ then $A_i^\bullet$ is $t'_i{\rightarrow}t_i$, and if $A_i \equiv t{=}t'$ then $A_i^\bullet$ is $eq(t, t'){\rightarrow}tt$.

Systems of equations in $(\Sigma, E \cup A)$ with form $G \equiv \bigwedge_{i=1}^m (s_i = t_i)$ become reachability goals in $\mathcal{R}_E$ of the form $\bigwedge_{i=1}^m eq(s_i, t_i) \rightarrow tt$. A substitution σ is a solution of G if there are derivations for $\bigwedge_{i=1}^m (s_i\sigma = t_i\sigma)$, or $\bigwedge_{i=1}^m eq(s_i\sigma, t_i\sigma)$ rewrites to $\top$.

The *inference rules for membership rewriting in* $\mathcal{R}_E$ are the ones in Figure 3, adapted from [DLM+08, Fig. 4, p. 12], where the rules are defined for context-sensitive membership rewriting.

3 Conditional narrowing modulo unification

Narrowing allows us to assign values to variables in such a way that a reachability goal holds. We implement narrowing using a calculus that has the following properties:

$$\frac{t_1 \to^1 t_2,\ t_2 \to t_3}{t_1 \to t_3}\ \texttt{Transitivity} \qquad \frac{t \to^1 t', t' : s}{t : s}\ \texttt{Subject Reduction}$$

$$\frac{t =_A t'}{t \to t'}\ \texttt{Reflexivity} \qquad \frac{t_i \to^1 t_i'}{f(t_1, \ldots, t_i, \ldots, t_n) \to^1 f(t_1, \ldots, t_i', \ldots, t_n)}\ \texttt{Congruence}$$

$$\frac{t \to t' \text{ if } A_1^\bullet \ldots A_n^\bullet \in \mathcal{R}_E \text{ and } u =_A t\sigma \quad A_1^\bullet\sigma \ldots A_n^\bullet\sigma}{u \to^1 t'\sigma}\ \texttt{Replacement}$$

$$\frac{t : s \text{ if } A_1^\bullet \ldots A_n^\bullet \in \mathcal{R}_E \text{ and } u =_A t\sigma \quad A_1^\bullet\sigma \ldots A_n^\bullet\sigma}{u : s}\ \texttt{Membership}$$

Fig. 3. Inference rules for membership rewriting.

1. If σ is an $R/\mathcal{E}$-normalized idempotent solution for a reachability goal G, the calculus can compute a more general answer $\sigma \ll_{\mathcal{E}} \sigma'$ for G.
2. If the calculus computes an answer σ for G, then σ is a solution for G.

That is, we want to compute a complete set of answers for G, a set that includes a generalization of any possible solution for G, with respect to $R/\mathcal{E}$-normalized substitutions.

We are going to split this task into two subtasks: first we will solve the part of the calculus that deals with unification; second, we will solve the part that deals with reachability.

3.1 Calculus rules for unification

We assume we are working with an executable rewrite theory named M. We refer to the set of equations and memberships in M as E, to the set of rules as R and to the set of axioms as A. We also assume that we have an A-unification algorithm that returns a *CSU* for any pair of terms.

A unification equation is a term $s{:}S = t{:}T$, which is a shorthand for the system of equations $s = t \wedge s = X_S \wedge t = Y_T$ (we will also write $s = t$, $s{:}S$, $t{:}T$). This means that we intend to unify s and t, with resulting sorts S and T respectively. A unification goal is a sequence (understood as conjunction) of unification equations.

Admissible goals, or simply goals, are any sequence of $s{:}S{=}t{:}T$, $s{:}S{:=}t{:}T$, $s{:}S{\to}t{:}T$, $s{:}S{\to^1}t{:}T$ and $t{:}T$. Any condition in an equation, of the form $s{=}t$ or $s{:=}t$ is turned into an admissible goal by adding inferred sorts to it. If any term s is a variable or a constant, we use the sort of s as inferred sort. If the term is of the form $f(\bar{s})$, we use the kind of any membership for f.

Our calculus is defined by the following set of inference rules derived from those in Figure 3. The first two rules, $[u]$ and $[x]$, transform *equational* problems into *rewriting* problems modulo axioms, rule $[u]$ playing the part of the added rules $eq(x{:}k, x{:}k) \to tt$ in the associated rewrite theory; rule $[n]$ describes one step of unification narrowing where the conditions on the applied rule are turned

into subgoals and the instantiated right side of the rule ($r\theta$) is required to have a sort which is a common subsort of S and T; rule $[t]$ allows us to apply several unification narrowing steps; rule $[i]$ decomposes a term allowing rule $[n]$ to be applied to any subterm of it; rule $[r]$ allows instantiation of variables on unificable terms; rule $[m1]$ solves the membership problem for variables, and rules $[s]$ and $[m2]$ for the rest of terms, using the membership conditions in E:

- $[u]$ *unification*

$$\frac{s{:}S = t{:}T, G'}{s{:}S' \to X_{S'}{:}S', t{:}S' \to X_{S'}{:}S', G'}$$

where $X_{S'}$ fresh variable, $S' \leq S, S' \leq T$.

- $[x]$ *matching*

$$\frac{s{:}S := t{:}T, G'}{t{:}S' \to s{:}S', G'}$$

where $S' \leq S, S' \leq T$.

- $[n]$ *narrowing*

$$\frac{s{:}S \to^1 X{:}T, G'}{((c,)X{:}S', G')\rho\theta}$$

where s is not a variable, $(c)eq\ l{=}r\ (if\ c) \in E$ has fresh variables, $S' \leq S, S' \leq T$, $\theta \in CSU_A(s = l)$, $\rho{=}\{X \mapsto r\}$.

- $[t]$ *transitivity*

$$\frac{s{:}S \to t{:}T, G'}{s{:}S' \to^1 X_{S'}{:}S', X_{S'}{:}S' \to t{:}S', G'}$$

where $X_{S'}$ fresh variable, $S' \leq S, S' \leq T$.

- $[i]$ *imitation*

$$\frac{f(\bar{s}{:}\bar{S}){:}S \to^1 X{:}T, G'}{G'\theta, s_i{:}S_i \to^1 X'_{S_i}{:}S_i, X\theta{:}S', G''\theta}$$

with $X \notin Var(s), \theta = \{X \mapsto f((s_1, \ldots, s_{i-1}, X'_{S_i}{:}S_i, s_{i+1}, \ldots, s_n))\}$, X'_{S_i} fresh variable, $S' \leq S, S' \leq T$.

- $[r]$ *removal of equations*

$$\frac{s{:}S \to t{:}T, G'}{(G', s{:}S', G')\theta}$$

with $\theta \in CSU_A(s = t)$, $S' \leq S, S' \leq T$

- $[s]$ *subject reduction*

$$\frac{s{:}S, G'}{s{:}[S] \to^1 X_S{:}S, G'}$$

X_S fresh variable.

– [$m1$] *membership*

$$\frac{X_S{:}T, G'}{(G')\theta}$$

where $\theta = \{X_S \mapsto X'_{S'}\}$ with $X'_{S'}$ fresh variable and $S' \leq S, S' \leq T$.

– [$m2$] *membership*

$$\frac{s{:}S, G'}{((c,)\, G')\theta}$$

where $(c)mb\ t{:}T$ (*if* c) is a fresh variant, with $T \leq S$, of a (conditional) membership in E, and $\theta \in CSU_A(s = t)$.

From a unification equation u a derivation is made applying rules of the calculus. If the derivation ends in the empty goal, denoted by $\square$, then the composition of the substitutions used on each derivation step, restricted to those variables appearing in u, is a computed answer for u.

Theorem 1. *The calculus for unification is sound and weakly complete.*

That is, given a unification goal G, if $G \rightsquigarrow^*_\sigma \square$ then $G\sigma$ can be derived, so σ is a solution for G in $\rightarrow_{E/A}$, and if ρ is an E/A-normalized idempotent answer of G ($G\rho \rightarrow^*_{E/A} \top$), then there is ρ' idempotent, with $\rho \ll_A \rho'$, such that $G \rightsquigarrow_{\rho'} \square$.

4 Reachability by conditional narrowing

Conditional narrowing relies on conditional unification. As we have used the symbol $\rightarrow$ in the calculus rules for unification, we will use a different symbol $\Rightarrow$ in the calculus rules for reachability. Our goal, given a reachability problem $\bigwedge_i s_i{:}S_i \Rightarrow t_i{:}T_i$, is to find a solution σ (ground or not) such that $\bigwedge_i s_i\sigma{:}S_i \Rightarrow_{R/\mathcal{E}} t_i\sigma{:}T_i$. For executable rewrite theories this is equivalent to $\bigwedge_i s_i\sigma{:}S_i \Rightarrow_{R\cup E,A} \bigwedge_i t_i\sigma{:}T_i$. These new calculus rules deal with the $\rightsquigarrow_{R,A}$ part. Narrowing, we call it *replacement* here, takes place only at position ϵ of terms, thanks to new transitivity and imitation calculus rules.

Reachability goals are any sequence (understood as conjunction) of subgoals of the form $s{:}S \Rightarrow t{:}T$. Admissible goals, or simply goals, are now extended to be any sequence of $s{:}S \Rightarrow t{:}T$, $s{:}S \Rightarrow^1 t{:}T$, $s{:}S = t{:}T$, $s{:}S \rightarrow t{:}T$, $s{:}S \rightarrow^1 t{:}T$, $s{:}S \rightarrow^1 t{:}T$, $s{:}S := t{:}T$ and $t{:}T$. If the calculus derives the empty goal from a reachability goal G with a substitution σ, then σ is a computed answer for G.

As for unification, any reachability subgoal in our calculus of the form of $s{:}S \Rightarrow^{(1)} t{:}T$ is equivalent to the admissible goal $s \Rightarrow^{(1)} t$, $s{:}S$, $t{:}T$.

4.1 Calculus rules for reachability

Reachability by conditional narrowing is achieved using the calculus rules presented in Section 3, extended with the following calculus rules, based on the deduction rules for rewrite theories in Figure 2. Rule [X] solves reachability problems by unification; rule [R] applies one step of reachability narrowing; rule

$[T]$ enables reachability narrowing modulo and multiple steps of reachability narrowing. It is a direct consequence a 1 ; rule $[I]$ allows us to imitate narrowing at non root term positions, replacing the rewriting rule for congruence, that can now be achieved by transitivity and imitation. Recall that narrowing steps for reachability ($\Rightarrow^1$), which are generated by rule $[T]$, impose no sort within the given kind on the right side of the step:

- $[X]$ *reflexivity*

$$\frac{s{:}S \Rightarrow t{:}T, G'}{s{:}S = t{:}T, G'}$$

- $[R]$ *replacement*

$$\frac{s{:}S \Rightarrow^1 X_{[S]}{:}[S], G'}{(s{:}S, (c,), G')\rho\theta}$$

where s is not a variable, $(c)rl\ l \Rightarrow r\ (if\ c)$ is a fresh variant of a (conditional) rule in R, $\rho = \{X_{[S]} \mapsto r\}$, $\theta \in CSU_A(s = l)$.

- $[T]$ *transitivity*

$$\frac{s{:}S \Rightarrow t{:}T, G'}{s{:}S \rightarrow X'_S{:}S, X'_S{:}S \Rightarrow^1 X''_{[S]}{:}[S], X''_{[S]}{:}[S] \Rightarrow t{:}T, G'}$$

where X'_S and $X''_{[S]}$ are fresh variables.

- $[I]$ *imitation*

$$\frac{f(\bar{s}{:}\bar{S}){:}S \Rightarrow^1 X_{[S]}{:}[S], G'}{s_i{:}S_i \Rightarrow^1 X'_{S_i}{:}S_i, f(\bar{s}{:}\bar{S}){:}S, G'\theta}$$

where $X_{[S]} \notin vars(s), \theta = \{X_{[S]} \mapsto f((s_1, \ldots, X'_{S_i}{:}S_i, \ldots, s_n))\}$, X'_{S_i} fresh variable.

From a reachability goal r a derivation is made applying rules of the calculus. Each application of the *reflexivity* rule generates a unification equation. These unification equations as well as any generated membership goals must be solved using the calculus rules for unification. If the derivation ends with an *empty goal*, written $\square$, then the composition of the substitutions used on each derivation step, restricted to those variables appearing in r, is a computed answer for r.

Theorem 2. *The calculus for reachability is sound and weakly complete.*

That is, given a reachability goal G, if $G \rightsquigarrow^*_\sigma \square$ then $G\sigma$ can be derived, so σ is a solution for G in $\rightarrow_{R/\mathcal{E}}$, and if θ is an $R/\mathcal{E}$-normalized idempotent answer for a reachability problem G in $\rightarrow_{R/\mathcal{E}}$, then there is σ idempotent, with $\theta \ll_{\mathcal{E}} \sigma$, such that $G \rightsquigarrow^*_\sigma \square$.

Proof. We prove correctness of the calculus for reachability with respect to $R/\mathcal{E}$-normalized (equivalently $R \cup E, A$) idempotent substitutions for the executable rewrite theory $\mathcal{R} = (\Sigma, \mathcal{E}, R)$ in $\rightarrow_{R/\mathcal{E}}$.

1. Soundness: By structural induction on the calculus rule for reachability applied.

2. Completeness: We prove that for $R/\mathcal{E}$-normalized idempotent answers $\Rightarrow^1$ solves $\rightarrow^1_{R,A}$ reachability problems and $\Rightarrow$ solves $\rightarrow^*_{R/\mathcal{E}}$ reachability problems, according to [MT07, Theorem 3] and Lemma 1. Then it follows that if θ is an $R/\mathcal{E}$-normalized idempotent answer for a reachability problem G in $\rightarrow_{R/\mathcal{E}}$, then there is σ idempotent, with $\theta \ll_{\mathcal{E}} \sigma$, such that $G \rightsquigarrow^*_\sigma \square$. Inferred sorts are treated as in the proof of completeness of the calculus for unification (see extended version). We don't show the inferred sorts here.
   (a) We prove that if $s\rho \rightarrow^1_{R,A} t$ then $s \Rightarrow^1 t' \rightsquigarrow^*_\sigma \square$, with $\rho \ll_{\mathcal{E}} \sigma$ and $t \ll_{\mathcal{E}} t'$. By definition there is a position p in $s\rho$, a rule $l \rightarrow r$ *if* $c \in R$ and a matching θ such that $s\rho|_p = l\theta$, $c\theta$ can be derived and $t \equiv (s\rho)[r\theta]_p$. By the same reasoning we used for the completeness of the calculus for unification, p must be a nonvariable position in s. Otherwise ρ would not be $R/\mathcal{E}$-normalized. From $s \Rightarrow^1 X$, by imitation we can reach position p, turning our reachability problem into $s|_p \Rightarrow^1 X^p$ with $\eta = \{X \mapsto s[X^p]_p\}$. Applying replacement, as $s\rho|_p = l\theta$, there is $\sigma (\equiv \rho' \cup \theta') \in CSU_A(s\rho|_p = l)$, with $\rho \ll_{\mathcal{E}} \rho'$, $\theta \ll_{\mathcal{E}} \theta'$ and $t' \equiv X\eta\sigma \equiv (s\rho')[r\theta']_p$.
   It is important to remember, again, that ACU-coherence completion allows A-unification of the left term of the ACU-coherence completed version of the rule, l, with the whole $s\rho|_p$ whenever the original left term l can be A-unified with some subterm of a recombination of $s\rho|_p$.
   (b) We prove that if $s\rho \rightarrow^*_{R/\mathcal{E}} t\rho$, ρ is a solution, then $s \Rightarrow t \rightsquigarrow^*_\sigma \square$, with $\rho \ll_{\mathcal{E}} \sigma$. We distinguish two cases:
      – Reflexive case: $s\rho =_{\mathcal{E}} t\rho$. Then $s \Rightarrow t \rightsquigarrow_{[X]} s = t \rightsquigarrow^*_\sigma \square$, with $\rho \ll_{\mathcal{E}} \sigma$ by correctness of the calculus for unification.
      – Rest of the cases. According to [MT07, Lemmas 7 and 8] and the Lemma in Section 2.4 it suffices to show that $(\rightsquigarrow^*_{E,A} \rightsquigarrow_{R,A})^+ =_{\mathcal{E}}$ is implemented by $\Rightarrow$. This is done in the transitivity rule

$$\frac{s{:}S \Rightarrow t{:}T, G'}{s{:}S \rightarrow X'_S{:}S, X'_S{:}S \Rightarrow^1 X''_{[S]}{:}[S], X''_{[S]}{:}[S] \Rightarrow t{:}T, G'}$$

   $s{:}S \rightarrow X'_S$ implements $\rightsquigarrow^*_{E,A}$ as proved in the calculus for unification. $X'_S{:}S \Rightarrow^1 X''_{[S]}{:}[S]$ implements $\rightsquigarrow_{R,A}$ as proved in the previous point. $X''_{[S]}{:}[S] \Rightarrow t{:}T$ allows iteration (the $^+$ part) through several uses of the transitivity rule ending with the $=_{\mathcal{E}}$ part through the use of the reflexivity rule, which is the only rule that enables us to exit the loop generated by the transitivity rule.
   Finally, correct typing is ensured because $s{:}S$ and $t{:}T$ are included as conditions.

5 Example

As an example of our calculus we use the specification of the Tower of Hanoi puzzle in Section 2 and the reachability problem $(3T^0_T, b, c){:}S \Rightarrow (a, b, T^1_T){:}S$, where from a `State` composed of one `Tower` with `Disk` 3 on top of it and two `Towers`

with `Rods` b and c alone respectively we want to reach a `State` composed of two `Towers` with `Rods` a and b alone respectively and another `Tower`. The subindex of each variable means its type (sort or kind) and we write D, R, V, T, P, S instead of `Disk`, `Rod`, `ValidT`, `Tower`, `Pair`, `State` for readability.:

1. $\underline{(3T^0_T, b, c){:}S \Rightarrow (a, b, T^1_T){:}S} \rightsquigarrow_{[T]}$
   Transitivity decomposes reachability into several rewriting narrowing steps.
2. $\underline{(3T^0_T, b, c){:}S \to X^1_S{:}S}, X^1_S{:}S \Rightarrow^1 X^2_{[S]}{:}[S], X^2_{[S]}{:}[S] \Rightarrow (a, b, T^1_T){:}S$
   $\rightsquigarrow_{[r],\{T^0_T \mapsto a, X^1_S \mapsto (3a,b,c)\}} T^0_T$ is instantiated through rule $[r]$.
3. $\underline{(3a, b, c){:}S}, (3a, b, c){:}S \Rightarrow^1 X^2_{[S]}{:}[S], X^2_{[S]}{:}[S] \Rightarrow (a, b, T^1_T){:}S$
   We focus on the first subgoal.
4. $\underline{(3a, b, c){:}S} \rightsquigarrow_{[m2], S^1_{[S]}, S^2_{[S]}{:}S \ \mathit{if}\ S^1_{[S]}{:}S \wedge S^2_{[S]}{:}S, \{S^1_{[S]} \mapsto (3a,b), S^2_{[S]} \mapsto c\}}$
5. $\underline{c{:}S}, (3a, b){:}S \rightsquigarrow \ldots$
6. $\underline{3a{:}S} \rightsquigarrow_{[m2], X_{[D]} R_{[R]}{:}V \ \mathit{if}\ X_{[D]}{:}D \wedge R_{[R]}{:}R, \{X_{[D]} \mapsto 3, R_{[R]} \mapsto a\}}$. OK because $V \leq S$.
7. $\underline{3{:}D, a{:}R} \rightsquigarrow \ldots$ similar to previous steps. First subgoal finished.
8. $(3a, b, c){:}S \Rightarrow^1 X^2_{[S]}{:}[S], X^2_{[S]}{:}[S] \Rightarrow (a, b, T^1_T){:}S$. We focus on the first subgoal.
9. $\underline{(3a, b, c){:}S \Rightarrow^1 X^2_{[S]}{:}[S]} \rightsquigarrow_{[R], D_{[T]}, E_{[T]}, X_{[S]} \to F_{[T]}, G_{[T]}, X_{[S]} \ \mathit{if}}$
   $D_{[T]}{:}T \wedge E_{[T]}{:}T \wedge X_{[S]}{:}S \wedge F_{[T]}{:}T \wedge G_{[T]}{:}T \wedge F_{[T]} - G_{[T]} := \mathit{move}(D_{[T]} - E_{[T]}),$
   $\theta = \{D_{[T]} \mapsto 3a, E_{[T]} \mapsto c, X_{[S]} \mapsto b\}, \rho = \{X^2_{[S]}{:}[S] \mapsto F_{[T]}, G_{[T]}, X_{[S]}\}$ Narrowing step.
10. $\underline{(3a, b, c){:}S, 3a{:}T, c{:}T, b{:}S}, (F_{[T]} - G_{[T]}){:}[P] := \mathit{move}(3a - c){:}[P] \rightsquigarrow \ldots$
11. $\underline{F_{[T]} - G_{[T]}{:}[P] := \mathit{move}(3a - c){:}[P]} \rightsquigarrow_{[x]}$
12. $\underline{\mathit{move}(3a - c){:}[P] \to F_{[T]} - G_{[T]}{:}[P]} \rightsquigarrow_{[t]}$
    Transitivity decomposes unification into several unification narrowing steps.
13. $\underline{\mathit{move}(3a - c){:}[P] \to^1 Y_{[P]}{:}[P], Y_{[P]}{:}[P] \to F_{[T]} - G_{[T]}{:}[P]} \rightsquigarrow_{[n],}$
    $\mathit{move}(X_{[D]} T_{[T]} - R_{[R]}) = T_{[T]} - X_{[D]} R_{[R]} \ \mathit{if}\ X_{[D]}{:}D \wedge T_{[T]}{:}T \wedge R_{[R]}{:}R,$
    $\theta = \{X_{[D]} \mapsto 3, T_{[T]} \mapsto a, R_{[R]} \mapsto c\}, \rho = \{Y_{[P]} \mapsto T_{[T]} - X_{[D]} R_{[R]}\}$
    Unification narrowing step. $Y_{[P]}$ is instantiated to a ground term.
14. $\underline{a - 3c{:}[P], 3{:}[D], a{:}[T], c{:}[R]}, a - 3c{:}[P] \to F_{[T]} - G_{[T]}{:}[P] \rightsquigarrow \ldots$
15. $\underline{a - 3c{:}[P] \to F_{[T]} - G_{[T]}{:}[P]} \rightsquigarrow_{[r], \theta_1 = \{F_{[T]} \mapsto a, G_{[T]} \mapsto 3c\}}$ Removal of equations.
16. $\underline{a - 3c{:}[P]} \rightsquigarrow \ldots$ We omit this and go back to the second subgoal on step 8.
17. $\underline{(a, 3c, b) : [S] \Rightarrow (a, b, T^1_T){:}S} \rightsquigarrow_{[X]} \ldots$
18. $\underline{(a, 3c, b) : S \to X_S{:}S}, (a, b, T^1_T){:}S \to X_S{:}S \rightsquigarrow_{[r], \{X_S \mapsto (a,3c,b)\}}$
19. $\underline{(a, 3c, b) : S}, (a, b, T^1_T){:}S \to (a, 3c, b){:}S \rightsquigarrow \ldots$
20. $\underline{(a, b, T^1_T){:}S \to (a, 3c, b){:}S} \rightsquigarrow_{[r], \{T^1_T \mapsto 3c\}} T^1_T$ is instantiated through rule $[r]$.
21. $\underline{(a, b, 3c) : S} \rightsquigarrow \ldots \square$

From the substitutions in steps 2 and 20 the answer $\{T^1_T \mapsto 3c, T^0_T \mapsto a\}$ is computed. The calculus has found the solution $(3a, b, c){:}S \Rightarrow (a, b, 3c){:}S$ which is an instance of the given reachability problem $(3T^0_T, b, c){:}S \Rightarrow (a, b, T^1_T){:}S$.

6 Related work, conclusions and future work

A classic reference in equational conditional narrowing modulo is the work of Bockmayr [Boc93]. The topic is addressed here for Church-Rosser equational CTRS with empty axioms, but non terminating axioms (like ACU) are not allowed. Non conditional narrowing modulo order-sorted equational logics is covered by Meseguer and Thati [MT07], the reference for recent development in this area, which is actively being used for cryptographic protocol analysis. This work is partially based on the work of Viry [Vir94] where $R/\mathcal{E}$ rewriting is defined in terms of R, A and E, A for unsorted rewrite theories. Another topic addressed by the present work, membership equational logic, is defined by Meseguer [Mes97]. An equivalent rewrite system for MEL theories is presented by Durán, Lucas et al. [DLM+08], allowing unification by rewriting. Strategies, which also play a main role in narrowing, have been studied by Antoy, Echahed and Hanus [AEH94]. Their needed narrowing strategy, for inductively sequential rewrite systems, generates only narrowing steps leading to a computed answer. Recently Escobar, Sasse and Meseguer [ESM12] have developed the concepts of variant and folding variant, a narrowing strategy for order-sorted unconditional rewrite theories that terminates on those theories having the *finite variant property*. As an extension to rewrite theories Bruni and Meseguer [BM06] have defined *generalized rewrite theories* that support context-sensitive rewriting, thus allowing rewrites only on certain positions of terms.

In this work we have developed a narrowing calculus for unification in membership equational logic and a narrowing calculus for reachability in rewrite theories with an underlying membership equational logic. The main features in these calculi are that they make use of membership information whenever possible, reducing the state space, and also that they only allow steps leading to a different state, no mutual cancelling steps are allowed. The calculi have been proved correct. This work is part of a bigger effort where we attempt to explore the possibilities of performing conditional narrowing with constraint solvers. A transformation for rules and goals that will make both calculi strongly complete is under study. Strong completeness of reachability for topmost rewrite theories, Russian dolls configurations and linear theories are also under study. Finally, decidability of the calculus for unification in the case of *operationally terminating* [LM09] MEL theories with a finitary and complete A-unification algorithm, using the required strategy for deterministic 3-CTRS's of solving subgoals from left to right, is being studied.

Our current line of investigation also intends to study the extension of the calculi to handle constraints and their connection with external constraint solvers for domains such as finite domains, integers, Boolean values, etc., that could greatly improve the performance of any implementation. We also plan on the extension of the calculi, adding support for generalized rewrite theories. Better strategies that may help reducing the state space will also be studied. All the improvements will have new sets of transformation rules that will allow their implementation on Maude.

References

[AEH94] S. Antoy, R. Echahed, and M. Hanus. A needed narrowing strategy. In H.-J. Boehm, B. Lang, and D. M. Yellin, editors, *POPL*, pages 268–279. ACM Press, 1994.

[BM06] R. Bruni and J. Meseguer. Semantic foundations for generalized rewrite theories. *Theoretical Computer Science*, 360(1-3):386–414, 2006.

[BM12] K. Bae and J. Meseguer. Model checking LTLR formulas under localized fairness. In F. Durn, editor, *WRLA*, volume 7571 of *Lecture Notes in Computer Science*, pages 99–117. Springer, 2012.

[Boc93] A. Bockmayr. Conditional narrowing modulo a set of equations. *Applicable Algebra in Engineering, Communication and Computing*, 4:147–168, 1993.

[CDE+07] M. Clavel, F. Durán, S. Eker, P. Lincoln, N. Martí-Oliet, J. Meseguer, and C. Talcott. *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, July 2007.

[CM96] M. Clavel and J. Meseguer. Reflection and strategies in rewriting logic. *Electronic Notes in Theoretical Computer Science*, 4:126–148, 1996.

[DLM+08] F. Durán, S. Lucas, C. Marché, J. Meseguer, and X. Urbain. Proving operational termination of membership equational programs. *Higher-Order and Symbolic Computation*, 21(1-2):59–88, 2008.

[DM12] F. Durán and J. Meseguer. On the Church-Rosser and coherence properties of conditional order-sorted rewrite theories. *The Journal of Logic and Algebraic Programming*, 81(7-8):816–850, 2012.

[EMM05] S. Escobar, C. Meadows, and J. Meseguer. A rewriting-based inference system for the NRL protocol analyzer: grammar generation. In V. Atluri, P. Samarati, R. Küsters, and J. C. Mitchell, editors, *FMSE*, pages 1–12. ACM, 2005.

[ESM12] S. Escobar, R. Sasse, and J. Meseguer. Folding variant narrowing and optimal variant termination. *The Journal of Logic and Algebraic Programming*, 81(7-8):898–928, 2012.

[Fay78] M.J. Fay. *First-order Unification in an Equational Theory.* University of California, 1978.

[LM09] S. Lucas and J. Meseguer. Operational termination of membership equational programs: the order-sorted way. *Electr. Notes Theor. Comput. Sci.*, 238(3):207–225, 2009.

[Mes90] J. Meseguer. Rewriting as a unified model of concurrency. In J.C.M. Baeten and J.W. Klop, editors, *CONCUR '90 Theories of Concurrency: Unification and Extension*, volume 458 of *Lecture Notes in Computer Science*, pages 384–400. Springer, 1990.

[Mes97] J. Meseguer. Membership algebra as a logical framework for equational specification. In Francesco Parisi-Presicce, editor, *WADT*, volume 1376 of *Lecture Notes in Computer Science*, pages 18–61. Springer, 1997.

[MT07] J. Meseguer and P. Thati. Symbolic reachability analysis using narrowing and its application to verification of cryptographic protocols. *Higher-Order and Symbolic Computation*, 20(1-2):123–160, 2007.

[Ohl02] E. Ohlebusch. *Advanced topics in term rewriting.* Springer, 2002.

[Vir94] P. Viry. Rewriting: An effective model of concurrency. In C. Halatsis, D. G. Maritsas, G. Philokyprou, and S. Theodoridis, editors, *PARLE*, volume 817 of *Lecture Notes in Computer Science*, pages 648–660. Springer, 1994.