Mathematical Structures in Computer Science (2020), **30**, pp. 683–709
doi:10.1017/S0960129520000031

CAMBRIDGE UNIVERSITY PRESS

PAPER

Computing knowledge in equational extensions of subterm convergent theories

Serdar Erbatur[1] Andrew M. Marshall[2] and Christophe Ringeissen[3,*]

[1]University of Texas at Dallas, Richardson, TX, USA, [2]University of Mary Washington, Fredericksburg, VA, USA and [3]Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France

*Corresponding author. Email: Christophe.Ringeissen@loria.fr

(Received 28 February 2019; revised 4 October 2019; accepted 18 January 2020; first published online 2 March 2020)

Abstract

We study decision procedures for two knowledge problems critical to the verification of security protocols, namely the intruder deduction and the static equivalence problems. These problems can be related to particular forms of context matching and context unification. Both problems are defined with respect to an equational theory and are known to be decidable when the equational theory is given by a subterm convergent term rewrite system (TRS). In this work, we extend this to consider a subterm convergent TRS defined modulo an equational theory, like Commutativity. We present two pairs of solutions for these important problems. The first solves the deduction and static equivalence problems in rewrite systems modulo shallow theories such as Commutativity. The second provides a general procedure that solves the deduction and static equivalence problems in subterm convergent systems modulo syntactic permutative theories, provided a finite measure is ensured. Several examples of such theories are also given.

Keywords: Equational theories; decision procedures; security protocols; deduction; static equivalence

1. Introduction

The formal analysis of security protocols is nowadays widely investigated. One of its starting points is the paradigm initiated by Dolev and Yao (1981) where term algebras modulo some equational theories are used to represent messages built over cryptographic primitives.

Several automated tools for the analysis of security issues in protocols have been developed, including Armando *et al.* (2005); Blanchet (2016, 2001); Chadha *et al.* (2016); Cheval *et al.* (2018b); Ciobâcă *et al.* (2012); Cremers (2008); Escobar *et al.* (2007); Mödersheim and Viganò (2009); Schmidt *et al.* (2012); and Turuani (2006). All these tools make use of decision procedures related to constraint solving in term algebras.

Verifying the security of protocols requires the development of specific decision procedures to reason about the knowledge of an intruder. Two important measures of this knowledge are *(intruder) deduction* (Millen and Shmatikov 2001; Paulson 1998) and *static equivalence* (Abadi and Cortier 2006). The deduction problem is the question of whether an intruder, given their deductive capability and a sequence of messages representing their knowledge, can obtain some secret. This is a critical measure of the capability of the protocol to maintain secrets. Deducibility is needed for many questions about the security of protocols. However, there are some questions for which we need to be able to decide more than deducibility. For some protocols, in addition to deducibility, we would like to determine whether an intruder can distinguish between different

runs of the protocol. For example, in protocols which attempt to transmit encrypted votes, we would like to know if, to the attacker, two different votes are indistinguishable. Static equivalence measures this property.

Much work has gone into investigating and developing decision procedures for the deduction and the static equivalence problems (Abadi and Cortier 2006; Ayala-Rincón *et al.*, 2017; Baudet *et al.*, 2013; Ciobâcă *et al.*, 2012; Conchinha *et al.*, 2011; Erbatur *et al.*, 2017). In this line of research, the security protocols are often represented by equational theories usually defined as unions of several simpler sub-theories. In this paper, we focus on decision procedures for the deduction problem and the static equivalence problem in equational theories $T \cup E$ where T and E are possibly non-disjoint. Until now, decision procedures for these problems have been obtained under the following assumptions:

— T is given by a subterm convergent term rewrite system (TRS), and E is empty (Abadi and Cortier 2006);
— T and E are disjoint (Cortier and Delaune 2010) and both deduction and static equivalence are decidable in T and in E;
— T and E share only constructors (Erbatur *et al.*, 2017), and both deduction and static equivalence are decidable in T and in E.

In this paper, we investigate a new scenario:

— T is given by a TRS R which is both subterm and convergent modulo E, and E is an arbitrary equational theory.

We then have two cases, based on E. The first case follows our preliminary results in Erbatur *et al.* (2018) where we are able to show that the methods of Abadi and Cortier (2006) can be extended to rewrite systems that are both subterm and convergent modulo E for a simple but significant class of E theories. In the second case, the previous method is insufficient and a new approach is developed that extends to a broader class of E theories.

We focus on permutative theories E, such as the Commutativity $C = \{x + y = y + x\}$ or the Associativity–Commutativity $AC = \{(x + y) + z = x + (y + z), x + y = y + x\}$. Permutative theories are commonly used as background theories E in TRSs modulo E. In a permutative theory, the number of occurrences of a symbol on the left-hand side of an axiom is equal to the number of occurrences on the right. These theories also have a number of nice properties, such as being finite, and having decidable word and matching problems. However, the unification problem in general is undecidable in permutative theories even if there are important particular permutative theories with a decidable unification problem, such as C and AC. We investigate the following two classes of permutative theories:

— The first class corresponds to *shallow permutative* theories. In that particular case, a permutative theory must be shallow, meaning that a variable can only occur at depth 1 in the axioms of the theory. Thus, C is a typical example of a shallow permutative theory. We show that this class of theories admits decision procedures for deduction and static equivalence which are based on some reductions to the empty theory. In that simple case, it is possible to reuse the same proof techniques as the ones developed for the case of subterm convergent term rewrite systems (Abadi and Cortier 2004, 2006).
— The second class consists of *syntactic permutative* theories. A syntactic theory admits a complete unification procedure defined as a non-necessarily terminating extension of syntactic unification with finitely many additional mutation rules. According to Kirchner and Klay (1990), any permutative theory with a finitary unification problem is indeed syntactic permutative. Thus, AC is syntactic permutative, but it is not shallow due to the Associativity axiom. For this general case of syntactic permutative theories, we develop a new approach

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

based on the computation of a Complete set of E-Matched Terms, *CMT* for short. We show that E-matched terms relate to an appropriate notion of E-variant. Here, an E-variant is a generalization of a normalized variant, as defined in the literature, in the restricted case where E is given by a convergent rewrite system (Comon and Delaune 2005). We present a method called MTG to generate a *CMT*. In general, MTG is not necessarily terminating. However, we identify a class of syntactic permutative theories, namely the permutative theories closed by paramodulation, for which MTG always terminates by computing a finite *CMT*. Thus, MTG is instrumental in showing that any permutative theory closed by paramodulation has the Finite Equational Variant Property, defined here and denoted by FEVP. Even if the MTG procedure does not terminate, it may still be possible to find a finite *CMT* for all the left-hand sides of a rewrite system R. In this case, it is then possible to define an appropriate notion of size of R modulo E, which is of prime interest to solve the knowledge problems in $R \cup E$. Actually we show that both deduction and static equivalence in $R \cup E$ are decidable if they are decidable in E, the size of R modulo E is computable, and R is subterm and convergent modulo E. Compared to the shallow case, this second approach substantially differs from the ones developed in Abadi and Cortier (2004, 2006). However, we are able to reuse some combination techniques introduced for solving the knowledge problems in unions of theories (Cortier and Delaune 2010; Erbatur *et al.*, 2017). Indeed, our reduction methods require that we implement decision procedures for deduction and static equivalence in the combination of E with additional free function symbols. Thankfully, this is always possible due to the combination result in Cortier and Delaune (2010).

1.1 Main contributions

The primary contributions of this paper are:

- A new method that solves both the deduction and static equivalence problems in $R \cup E$ where R is a subterm rewrite system and R is convergent modulo a shallow permutative theory E (cf. Theorem 1).
- The introduction of the notion of complete set of E-matched terms (*CMT*) and a method called MTG generating a *CMT* for the case E is syntactic permutative. If MTG always terminates by computing a finite *CMT*, then there are two important consequences: (1) E has the Finite Equational Variant Property (FEVP) considered in this paper, and (2) the size of R modulo E is computable for any rewrite system R. We show that MTG is terminating for the class of permutative theories closed by paramodulation, and so all these theories have the FEVP (cf. Theorem 2).
- A new method that solves both the deduction and static equivalence problems in $R \cup E$ if R is a subterm rewrite system, R is convergent modulo a syntactic permutative theory E, the size of R modulo E is computable, and both deduction and static equivalence are decidable in E (cf. Theorem 3). Compared to the simple case of a shallow permutative theory E, notice that the general case of a syntactic permutative theory E requires some additional assumptions: the computability of the size of R modulo E and the decidability of both deduction and static equivalence in E.

1.2 Plan of the paper

The paper is organized as follows. The concepts and notations used in this paper can be found in Section 2. The equational theories we focus on are exemplified in Section 3. These theories are of the form $R \cup E$, where the rewrite system R is both subterm and convergent modulo a syntactic permutative theory E. Section 4 presents the decision procedures for deduction and static equivalence in $R \cup E$ for the particular case in which E is shallow permutative. Starting from Section 5, we investigate the general case in which E is syntactic permutative. At the beginning of Section 5, we introduce the notion of complete set of E-matched terms (*CMT*) and discuss how a (finite) *CMT*

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

can be related to a (finite) complete set of E-variants. The MTG method for generating a *CMT* is presented in Section 5.1. As shown in Section 5.2, MTG terminates and generates a finite *CMT* for the class of permutative theories closed by paramodulation. Section 6 introduces the key notion of size of R modulo E, which is well defined when each left-hand side of R admits a finite *CMT*. Then, we show how to reduce any deduction (resp., static equivalence) problem in $R \cup E$ into a deduction (resp., static equivalence) problem in E, provided that the size of R modulo E is computable. For both classes of permutative theories, shallow (Section 4) and syntactic (Section 6), we detail the correctness proofs of the related decision procedures. Finally, we discuss in Section 7 some possible lines of future work.

2. Preliminaries

We assume that the reader is familiar with equational logic and term rewriting. We use the standard notations as presented in Baader and Nipkow (1989). In addition, as in Abadi and Cortier (2006), we use some concepts, such as names and frames, borrowed from the applied pi calculus (Abadi and Fournet 2001).

2.1 Terms and substitutions

Given a first-order signature Σ, a set of *names* is a countable set of (free) constants N, such that $\Sigma \cap N = \emptyset$. Given a (countable) set of variables X, the set of $(\Sigma \cup N)$-terms over X is denoted by $T(\Sigma \cup N, X)$. The set of variables in a term t is denoted by $fv(t)$, and the set of names in t is denoted by $fn(t)$. A term t is *ground* if $fv(t) = \emptyset$. For any position p in a term t (including the root position ϵ), $t(p)$ denotes the symbol at position p, $t|_p$ denotes the subterm of t at position p, and $t[u]_p$ denotes the term t in which $t|_p$ is replaced by u. The size of a term t is denoted by $|t|$ and defined in the usual way as follows: $|f(t_1, \ldots, t_n)| = 1 + \sum_{i=1}^{n} |t_i|$ if f is a n-ary function symbol with $n \geq 1$, $|c| = 1$ if $c \in N$, and $|x| = 0$ if $x \in X$. Given any $\Sigma' \subseteq \Sigma$, a term t is said to be Σ'-*rooted* if $t(\epsilon) \in \Sigma'$. A *context*, s, is a first-order term with "holes" or distinguished variables that occur only once. We may write $s[x_1, \ldots, x_n]$, to illustrate that the context s contains n distinguished variables.

A substitution σ is an endomorphism of $T(\Sigma \cup N, X)$ with only finitely many variables not mapped to themselves, denoted by $\sigma = \{x_1 \mapsto t_1, \ldots, x_m \mapsto t_m\}$. Application of a substitution σ to a term t is written as $t\sigma$. Given two substitutions θ and σ, the composition $\sigma \circ \theta$ is the substitution denoted here by $\theta\sigma$ and defined such that $x(\theta\sigma) = (x\theta)\sigma$ for any $x \in X$. The domain of σ is $Dom(\sigma) = \{x \in X \mid x\sigma \neq x\}$. The range of σ is $Ran(\sigma) = \{x\sigma \mid x \in Dom(\sigma)\}$. When θ and σ are two substitutions with disjoint domains and only ground terms in their ranges, then $\theta\sigma = \theta \cup \sigma$. Given a substitution σ and a finite set of variables $V \subseteq X$, the restriction of σ to V is the substitution denoted by $\sigma_{|V}$ such that $x\sigma_{|V} = x\sigma$ for any $x \in V$ and $x\sigma_{|V} = x$ for any $x \in X \backslash V$.

2.2 Equational theories

Given a set E of Σ-axioms (i.e., pairs of Σ-terms, denoted by $l = r$), the *equational theory* $=_E$ is the congruence closure of E under the law of substitutivity. For any Σ-term t, the equivalence class of t with respect to $=_E$ is denoted by $[t]_E$. Since $\Sigma \cap N = \emptyset$, the Σ-equalities in E do not contain any names in N. A theory E is *trivial* if $x =_E y$, for two distinct variables x and y. In this paper, all the considered theories are assumed non-trivial.

An E-unification problem with free constants in N is a set of $\Sigma \cup N$-equations $P = \{s_1 =^? t_1, \ldots, s_m =^? t_m\}$. The set of variables in P is denoted by $fv(P)$. A solution to P, called an E-*unifier*, is a substitution σ such that $s_i\sigma =_E t_i\sigma$ for all $1 \leq i \leq m$. A substitution σ is *more general modulo* E than θ on a set of variables V, denoted as $\sigma \leq_E^V \theta$, if there is a substitution τ such that $x\sigma\tau =_E x\theta$ for all $x \in V$. A *Complete Set of E-Unifiers* of P, denoted by $CSU_E(P)$, is a set of substitutions such that each $\sigma \in CSU_E(P)$ is an E-unifier of P, and for each E-unifier θ of P, there exists $\sigma \in CSU_E(P)$ such that $\sigma \leq_E^{fv(P)} \theta$. E-unification is said to be *finitary* if any E-unification problem

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

P admits a finite $CSU_E(P)$. An E-unification problem $P = \{x_1 =^? t_1, \ldots, x_m =^? t_m\}$ is a *solved form* if $x_1, \ldots, x_m$ are variables occurring once in P, and in that case, the corresponding substitution $\mu_P = \{x_1 \mapsto t_1, \ldots, x_m \mapsto t_m\}$ is an E-unifier of P such that $\{\mu_P\}$ is a $CSU_E(P)$. When E is empty, E-unification is called syntactic unification and has the property of being unitary: if two terms s and t are syntactically unifiable, then there exists a $CSU_E(s =^? t)$ of cardinality 1 whose element is called a most general unifier of $s =^? t$ denoted by $mgu(s, t)$.

A match-equation over $\Sigma \cup N$ is a $\Sigma \cup N$-equation $s =^? t$ such that t is ground, also denoted by $s \leq^? t$. An E-matching problem (with free constants in N) is a set of match-equations over $\Sigma \cup N$. An E-word problem (with free constants in N) is a set of ground $\Sigma \cup N$-equations. Thus, any E-matching problem and any E-word problem is defined as a particular E-unification problem with free constants. As a usual practice, a variable x may also occur in the right-hand side of a match-equation. In that case, the variable x is said to be a *subject* variable, which is considered as a free constant in the related unification problem, meaning that any E-unifier σ must satisfy the additional constraint $x\sigma = x$.

Let us introduce the different classes of theories considered in the paper. An axiom $l = r$ is *regular* if l and r have the same set of variables. An axiom $l = r$ is *collapse-free* if l and r are non-variable terms. An equational theory is *regular* (resp., *collapse-free*) if all its axioms are regular (resp., collapse-free). An equational theory E is *finite* if for each term t, there are only finitely many terms s such that $t =_E s$. Matching in finite theories is finitary. A finite theory is necessarily regular and collapse-free. A sufficient condition to get a finite theory is to assume that E is permutative. An equational theory E is *permutative* if for each axiom $l = r$ in E, l and r contain the same symbols with the same number of occurrences. Well-known theories such as Associativity ($A = \{(x + y) + z = x + (y + z)\}$), Commutativity ($C = \{x + y = y + x\}$), and Associativity–Commutativity ($AC = A \cup C$) are permutative theories. The word problem and matching are both decidable in finite theories, and so in permutative theories. However, unification in permutative theories is undecidable in general (Schmidt-Schauß 1989).

2.3 Notions of knowledge

The applied pi calculus and frames are used to model attacker knowledge (Abadi and Fournet 2001). In this model, the set of messages or terms which the attacker knows, and which could have been obtained from observing one or more protocol sessions, are the set of terms in $Ran(\sigma)$ of the frame $\phi = \nu\tilde{n}.\sigma$, where σ is a substitution such that the terms of $Ran(\sigma)$ are ground. We also need to model cryptographic concepts such as nonces, keys, and publicly known values. We do this by using names, which are essentially free constants. We need to track the names the attacker knows, such as public values, as well as the names the attacker does not know, such as freshly generated nonces. In a frame $\phi = \nu\tilde{n}.\sigma$, $\tilde{n}$ consists of a finite set of restricted names and represents names which remain secret from the attacker. The set of names occurring in a term t is denoted by $fn(t)$.

Given a frame $\phi = \nu\tilde{n}.\sigma$ and a term t, $t\phi$ denotes by a slight abuse of notation the term $t\sigma$. We say that a term t *satisfies the name restriction (of ϕ)* if $fn(t) \cap \tilde{n} = \emptyset$. In this paper, we start with the Abadi and Cortier notation, introduced below, for deduction and static equivalence. However, deduction and related problems have been studied before Abadi and Cortier (2006), see for example Amadio and Lugiez (2000).

Definition 1 (Deduction). *Let $\phi = \nu\tilde{n}.\sigma$ be a frame, and t a ground term. We say that t is* deduced from ϕ modulo E, *denoted by $\phi \vdash_E t$, if there exists a term s such that $s\sigma =_E t$ and $fn(s) \cap \tilde{n} = \emptyset$. The term s is called a* recipe *of t in ϕ modulo E. When ϕ and E are clear from the context, a recipe of t is usually denoted by ζ_t.*

Another form of knowledge is the ability to tell if two frames are *statically equivalent* modulo E, sometimes also called *indistinguishability*.

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

Definition 2 (Static Equivalence). *Two terms s and t are* equal *in a frame $\phi = \nu\tilde{n}.\sigma$ modulo an equational theory E, denoted $(s =_E t)\phi$, if $s\sigma =_E t\sigma$, and $\tilde{n} \cap (fn(s) \cup fn(t)) = \emptyset$. The set of all equalities $s = t$ such that $(s =_E t)\phi$ is denoted by $Eq(\phi)$. Given a set of equalities Eq, the fact that $(s =_E t)\phi$ for any $s = t \in Eq$ is denoted by $\phi \models Eq$. Two frames $\phi = \nu\tilde{n}.\sigma$ and $\psi = \nu\tilde{n}.\tau$ are* statically equivalent modulo *E, denoted as $\phi \approx_E \psi$, if $Dom(\sigma) = Dom(\tau)$, $\phi \models Eq(\psi)$ and $\psi \models Eq(\phi)$.*

Both deduction and static equivalence are known to be decidable in subterm convergent rewrite systems (Abadi and Cortier 2006). In this paper, we lift these results to term rewrite systems that are subterm convergent modulo some permutative theories.

2.4 Term rewrite systems

A term rewrite system (TRS, for short) is a pair (Σ, R), where Σ is a signature and R is a finite set of rewrite rules of the form $l \to r$, such that l, r are Σ-terms, l is not a variable and $fv(r) \subseteq fv(l)$. When the signature is clear from the context, a TRS is simply denoted by R. A term s *rewrites* to a term t, denoted by $s \to_R t$ (or simply $s \to t$), if there exists a position p of s, a rule $l \to r \in R$, and a substitution σ such that $s|_p = l\sigma$ and $t = s[r\sigma]_p$. A term s is a *normal form with respect to the relation* $\to_R$ (or simply a normal form), if there is no term t such that $s \to_R t$. This notion is lifted to substitutions as follows: a substitution σ is *normalized* if, for every variable x in the domain of σ, $x\sigma$ is a normal form. A TRS R is *terminating* if there are no infinite reduction sequences with respect to $\to_R$. A TRS R is *confluent* if, whenever $t \to_R^* s_1$ and $t \to_R^* s_2$, there exists a term w such that $s_1 \to_R^* w$ and $s_2 \to_R^* w$. A confluent and terminating TRS is called *convergent*. In a convergent TRS R, any term t admits a unique R-normal form denoted by $t{\downarrow_R}$. A TRS R is said to be *subterm* if for any $l \to r \in R$, r is either a strict subterm of l or a ground R-irreducible term. A TRS is *subterm convergent* if it is both subterm and convergent. An equational theory E is *subterm convergent* if there exists a subterm convergent TRS R such that $=_E$ is $\leftrightarrow_R^*$. The size of a TRS R is denoted by $|R|$ and defined as follows: $|R| = \max_{\{l \to r \in R\}} |l|$. Since a variable cannot occur as the left-hand side of any rule in R, we have that $|R| \geq 1$ for any non-empty TRS R. When R is empty, we define $|R| = 1$.

Let us now introduce the notion of equational rewriting, also called class rewriting (Jouannaud and Kirchner 1986). Given a TRS R and an equational theory E, the rewrite relation of R modulo E is defined as follows: $s \to_{R,E} t$ if there exist some position p in s, some rule $l \to r \in R$ and a substitution μ such that $s|_p =_E l\mu$ and $t = s[r\mu]_p$. The TRS R is said to be *E-convergent* if the relation $=_E \circ \to_R \circ =_E$ is terminating and $\longleftrightarrow_{R \cup E}^* \subseteq \to_{R,E}^* \circ =_E \circ \leftarrow_{R,E}^*$. In an E-convergent TRS R, any term t admits a unique R-normal form modulo E denoted by $t{\downarrow_{R,E}}$, and for any terms s and t, we have $s \longleftrightarrow_{R \cup E}^* t$ iff $(s{\downarrow_{R,E}}) =_E (t{\downarrow_{R,E}})$. In this paper, we focus on the following E-convergent TRSs.

Definition 3. *A* subterm E-convergent *TRS is a TRS which is both subterm and E-convergent.*

A reduction ordering is a well-founded ordering on terms closed under context and substitution. Consider an inference system $\mathcal{I}$ whose each inference rule is of the form $e_1, \ldots, e_n \vdash e_{n+1}$ where both the premises $e_1, \ldots, e_n$ and the conclusion e_{n+1} are equalities. Let us assume a reduction ordering $<$ with the additional property of being total on ground terms. To define a notion of redundancy with respect to $<$, we extend $<$ to an ordering on equalities using the multiset extension of $<$ to compare equalities viewed as multisets of terms. In a set of ground equalities E, an inference with premises in E is said to be *redundant* if its conclusion follows from the equalities of E that are smaller than its largest premise. In an arbitrary set of equalities E, an inference with premises in E is *redundant* if it is redundant in the set of all ground instances of E. A set of equalities is *saturated* with respect to $\mathcal{I}$ if all the inferences of $\mathcal{I}$ with premises in E are redundant.

2.5 Syntactic theories

A theory E is *syntactic* if it has a finite *resolvent presentation* S, that is a finite set of equational axioms S such that each equality $t =_E u$ has an equational proof $t \leftrightarrow_S^* u$ with at most one step $\leftrightarrow_S$

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

MatchDec $\{f(s_1,\ldots,s_m) \leq^? f(t_1,\ldots,t_m)\} \cup P \vdash \{s_1 \leq^? t_1,\ldots,s_m \leq^? t_m\} \cup P$

MatchMut $\{f(s_1,\ldots,s_m) \leq^? g(t_1,\ldots,t_n)\} \cup P$
$\vdash \{r_1 \leq^? t_1,\ldots,r_n \leq^? t_n,\ s_1 =^? l_1,\ldots,s_m =^? l_m\} \cup P$
where $f(l_1,\ldots,l_m) = g(r_1,\ldots,r_n)$ is a fresh variant of an axiom in S.

Rep $\{ x \leq^? u,\ t =^? t' \} \cup P \vdash \{ x \leq^? u,\ t =^? t'\{x \mapsto u\} \} \cup P$
where $x \in fv(t')$.

RemEq $\{t =^? t'\} \cup P \vdash \{t \leq^? t'\} \cup P$
where t' is ground.

Merge $\{x \leq^? t,\ x \leq^? s\} \cup P \vdash \{x \leq^? t,\ s \leq^? t\} \cup P$

Notice that all possible derivations must be explored when **MatchMut** is applicable in the case $f \neq g$ or when both **MatchDec** and **MatchMut** are applicable on the same match-equation in the case $f = g$.

Figure 1. MSP matching procedure for syntactic permutative theories

applied at the root position. The theories *C* and *AC* are permutative and syntactic (Kirchner and Klay 1990).

The interest of syntactic theories is to admit a mutation-based unification procedure that bears similarities with the standard syntactic unification procedure. In addition to the classical decomposition rule, additional mutation rules are needed. This leads to a mutation-based unification procedure which is not necessarily terminating for all syntactic theories. When restricting to the matching problem, it is possible to get termination for a large class of theories of practical interest. Actually, a mutation-based matching algorithm for the class of syntactic permutative theories has been presented in Nipkow (1990). In Figure 1, we consider a rule-based description of this matching algorithm borrowed from Ringeissen (2019). It will be applied in Section 6 to possibly compute a finite representation of terms matched by the left-hand sides of a TRS.

Lemma 1. *Assume S is a finite resolvent presentation of any syntactic permutative theory E. The* MSP *inference system given in Figure 1 provides a sound, complete, and terminating E-matching procedure: the set of computed solved forms corresponds to a complete set of solutions.*

Mutation-based unification algorithms are known for some important subclasses of syntactic theories, such as shallow theories (Comon *et al.*, 1994), and theories closed by paramodulation (Lynch and Morawska 2002). These particular syntactic theories play a central role in the paper and are defined as follows.

A theory *E* is *shallow* if variables can only occur at a depth at most 1 in axioms of *E*. For instance, *C* is a shallow theory but *AC* is not.

To define the property of being closed by paramodulation, we rely on the notion of saturation introduced at the end of Section 2.4 with respect to a reduction ordering $<$ which is assumed to be total on ground terms. An equational theory *E* is *closed by paramodulation* if *E* is a finite set of equalities saturated with respect to the inference system including the single rule

$$\textbf{Paramodulation}\ s[l'] = t, \quad l = r \vdash s[r]\sigma = t\sigma$$

where $l = r$ is a fresh instance of an equality in *E*, $\sigma = mgu(l', l)$, $l\sigma \not< r\sigma$ and l' is not a variable. Applying the paramodulation rule to an equality in *E* produces a new equality which is then added to the set. Thus, an equational theory is saturated by paramodulation when, after exhaustive applications of the above rule, no further non-redundant equalities are added (see Section 2.4 for the

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

description of redundant). The set is finitely saturated by paramodulation if only a finite number of new, non-redundant equalities are added.

The theory C is closed by paramodulation, while AC is not. The following permutative theories are not shallow but closed by paramodulation:

— $\{i(x) + y = i(y) + x\}$,
— $\{i(x) + y = i(y) + x,\ x * i(y) = y * i(x)\}$.

As shown in Lynch and Morawska (2002), the class of theories closed by paramodulation admits a terminating mutation-based unification procedure. For these theories, applying mutation on terms previously introduced by mutation will only generate superfluous unifiers. To take into account this property, the idea initiated in Lynch and Morawska (2002) is to consider a marking notation, terms boxing, such that terms generated by mutation are *boxed* and no further mutations are applied on boxed terms. Thus, any mutation reduces a complexity measure counting the number of unboxed symbols, where a symbol is said to be *boxed* if it occurs in a boxed term. This complexity measure is used to prove termination of a mutation-based unification procedure for theories closed by paramodulation (Lynch and Morawska 2002). Along the lines of Lynch and Morawska (2002), it is possible to obtain a matching algorithm for permutative theories closed by paramodulation called MSP_B and defined in the same way as MSP except **MatchMut** which is replaced by its boxed version:

$$\textbf{MatchMut}_B\ \{f(s_1, \ldots, s_m) \leq^? g(t_1, \ldots, t_n)\} \cup P \vdash \{\boxed{r_1} \leq^? t_1, \ldots, \boxed{r_n} \leq^? t_n,\ s_1 =^? \boxed{l_1}, \ldots, s_m =^? \boxed{l_m}\} \cup P$$

where $f(s_1, \ldots, s_m)$ is unboxed and $f(l_1, \ldots, l_m) = g(r_1, \ldots, r_n)$ is a fresh variant of an axiom in S.

Compared to MSP, the interest of MSP_B is to generate a reduced search space without loss of completeness for permutative theories closed by paramodulation.

3. Subterm Equational Convergent Rewrite Systems

From now on, let us consider $(\Omega \cup \Sigma, RE) = (\Omega \cup \Sigma, R \cup E)$ where $(\Omega \cup \Sigma, R)$ is a TRS modulo a permutative theory (Σ, E) such that $\Omega \cap \Sigma = \emptyset$ and R is subterm E-convergent. Hence, E can be a permutative theory such as C or AC. The fact that R is E-convergent implies the uniqueness of normal forms modulo E and the decidability of the word problem modulo RE: for any terms s and t, we have $s =_{RE} t$ iff $(s \downarrow_{R,E}) =_E (t \downarrow_{R,E})$. In the following, a term or a substitution is said to be normalized if it is normalized w.r.t $\rightarrow_{R,E}$, and a frame is normalized if its substitution is normalized.

In the rest of this section, we present some examples of theories RE such that R is subterm E-convergent (Section 3.1) and we introduce the notion of subterms modulo E (Section 3.2).

3.1 Examples

3.1.1 A theory for a messaging protocol

Let us start with a theory used in practice to model a group messaging protocol. The Asynchronous Racheting Tree protocol specified in Cohn *et al.* (2018) has been studied in Nguyen (2019) using ProVerif (Cheval *et al.* 2018a; Blanchet 2016). The goal of this protocol is to provide encrypted group messaging by maintaining some strong security guarantees. For this protocol, the theory modeling the intruder is defined in Nguyen (2019) as a combination $R_{ENC} \cup K$ where R_{ENC} and K are as follows:

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

$$R_{ENC} = \left\{ \begin{array}{r} adec(aenc(m, pk(sk)), sk) \to m \\ getmsg(sign(m, sk)) \to m \\ checksign(sign(m, sk), m, pk(sk)) \to ok \\ sdec(senc(m, k), k) \to m \end{array} \right\}$$

$$K = \left\{ keyexch(x, pk(x'), y, pk(y')) = keyexch(x', pk(x), y', pk(y)) \right\}$$

The combination $R_{ENC} \cup K$ is non-disjoint since the function symbol pk is shared by R_{ENC} and K. More precisely, pk satisfies an appropriate notion of shared constructor, and so the combination method described in Erbatur *et al.* (2017) applies to $R_{ENC} \cup K$. Another possibility is to apply the reduction method presented in Section 6 using the fact that R_{ENC} is subterm K-convergent and K is a permutative theory closed by paramodulation.

3.1.2 Non-associative sub-theories of Abelian groups and combinations

The theory of Abelian groups is $AG = R_{AG} \cup AC(*)$ where R_{AG} denotes the following $AC(*)$-convergent TRS:

$$R_{AG} = \left\{ \begin{array}{r} x * e \to x \\ x * i(x) \to e \\ x * (y * i(x)) \to y \\ i(i(x)) \to x \\ i(e) \to e \\ i(x * y) \to i(x) * i(y) \end{array} \right\}$$

R_{AG} is not subterm due to the rule $i(x * y) \to i(x) * i(y)$. Note that AG is an example of monoidal theory. Hence, the decidability of deduction and static equivalence in AG follows from the fact that these problems are decidable in monoidal theories (Cortier and Delaune 2010).

The theory of Abelian Pre-Groups is $APG = R_{APG} \cup C(*)$ where R_{APG} denotes the following subterm $C(*)$-convergent *TRS*:

$$R_{APG} = \{x * e \to x,\ x * i(x) \to e,\ i(i(x)) \to x,\ i(e) \to e\}$$

In Yang *et al.* (2014), *APG* was considered as an approximation to deal with unification in homomorphic encryption over Abelian groups.

We can actually extend the definition of *APG* to include an approximation of associativity in the following way. Define the theory of Abelian Pre-Groups with Associative Approximation by $APGAA = R_{APGAA} \cup C(*)$ where R_{APGAA} is the following subterm $C(*)$-convergent *TRS*:

$$R_{APGAA} = R_{APG} \cup \{(i(n) * x) * n \to x,\ (n * x) * i(n) \to x\}$$

This theory is a mono-sorted version of the theory of Abelian Pre-Groups with Associative Approximation studied in Yang *et al.* (2014). As with *APG*, the motivation of studying *APGAA* in Yang *et al.* (2014) is to approximate the theory of homomorphic encryption over Abelian groups in order to solve the unification problem using a variant-based approach (modulo $C(*)$). Note that *APG* and *APGAA* are not monoidal theories, despite the fact that *AG* is.

The modular exponentiation theory can be defined as an extension of *AG* with two additional axioms:

$$exp(exp(x, y), z) = exp(x, y * z),$$
$$exp(x, e) = x.$$

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

This theory is motivated by the Diffie-Hellman exponentiation and has been studied in Chevalier *et al.* (2003). The corresponding TRS

$$\{exp(exp(x, y), z) \to exp(x, y * z), \quad exp(x, e) \to x\} \cup R_{AG}$$

is $AC(*)$-convergent but not subterm. It is possible to get interesting subterm approximations of modular exponentiation by considering *APG* and *APGAA* instead of *AG*. In that direction, the following two TRSs:

$$\{exp(exp(x, y), i(y)) \to x, \quad exp(x, e) \to x\} \cup R_{APG},$$
$$\{exp(exp(x, y), i(y)) \to x, \quad exp(x, e) \to x\} \cup R_{APGAA}$$

are subterm $C(*)$-convergent. To take into account more equational properties of modular exponentiation, these last two TRSs are more generally convergent modulo the enlarged background theory $\{exp(exp(x, y), z) = exp(exp(x, z), y)\} \cup C(*)$, which is actually a syntactic permutative theory and more precisely a permutative theory closed by paramodulation.

3.1.3 Exclusive Or, its non-associative sub-theories, and combinations

The theory of Exclusive Or is $ACUN = R_{\oplus} \cup AC(\oplus)$ where $R_{\oplus}$ denotes the following subterm $AC(\oplus)$-convergent TRS:

$$R_{\oplus} = \{x \oplus 0 \to x,\ x \oplus x \to 0,\ x \oplus (x \oplus y) \to y\}$$

ACUN is another example of monoidal theory, and so it follows that both deduction and static equivalence are decidable in *ACUN* (Cortier and Delaune 2010). Interestingly, the decidability of these problems was already shown in Abadi and Cortier (2006) for *ACUN* by using some *ad-hoc* decision procedures. The deduction problem for *ACUN* was also successfully studied in Comon and Shmatikov (2003), Chevalier *et al.* (2005). By omitting the Associativity axiom, we get that $\{x \oplus 0 \to x, x \oplus x \to 0\}$ is subterm $C(\oplus)$-convergent.

Exclusive Or often appears as a sub-theory in the axiomatization of protocols. For example, the following TRS is an axiomatization for the Needham-Schroeder-Lowe protocol:

$$R_{NSL} = \{pk(x, sk(x, y)) \to y,\ sk(x, pk(x, y)) \to y\} \cup R_{\oplus}$$

Here, the operators *pk* and *sk* are used to model public key encryption. The unification problem in $R_{NSL} \cup AC(\oplus)$ was studied in Sasse *et al.* (2011). Notice that R_{NSL} is subterm $AC(\oplus)$-convergent. Other examples of a disjoint combination of $R_{\oplus}$ with a subterm convergent TRS have been considered in Dreier *et al.* (2018).

3.1.4 Quasigroups and loops

A quasigroup is a binary groupoid, $(Q, *)$, such that every equation of the form $x * y = z$ has a unique solution whenever two of the elements x, y, and z are specified. These have been axiomatized by the following identities (Hullot 1980):

$$x \setminus (xy) = y,\ x(x \setminus y) = y,\ (yx)/x = y,\ (y/x)x = y$$

where $\setminus$ and $/$ are the left and right division operations and are used to denote the unique solutions of the equation $x * y = z$, i.e, $y = x \setminus z$ and $x = z/y$. A loop is a quasigroup with a unit element. When the commutative axiom is added for multiplication, we do not need two unique divisors and can replace them by a system with a single divisor, here denoted by $x|y$. For example, with the following axiomatization:

$$\{(x * y)|x = y,\ x * (y|x) = y\} \cup \{x * y = y * x\}$$

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

the corresponding subterm $C(*)$-convergent rewrite system is

$$\{(x * y)|x \to y,\ x * (y|x) \to y,\ y|(y|x) \to x\}.$$

If we add a unit element to obtain a loop, we could start with the following axiomatization:

$$\{(x * y)|x = y,\ x * (y|x) = y,\ x * 1 = x\} \cup \{x * y = y * x\}$$

and the corresponding subterm $C(*)$-convergent rewrite system is

$$\{(x * y)|x \to y,\ x * (y|x) \to y,\ x * 1 \to x,\ y|1 \to y,\ x|x \to 1,\ y|(y|x) \to x\}.$$

3.1.5 Rewrite systems with E-constructors

Some classical examples of subterm rewrite systems can be easily adapted to model theories including commutative or associative-commutative symbols. For instance, let us briefly mention the following rewrite systems:

(i) $\{occ(x + k, k) \to ok\}$
(ii) $\{rm(x + k, k) \to x\}$
(iii) $\{dec(enc(x, k + y), k) \to x\}$
(iv) $\{dec(enc(x, k), k + y) \to x\}$

For any of these four rewrite systems, one can check that

— the symbol $+$ is a constructor, that is, it does not appear at the root of any left-hand side;
— the system is subterm $AC(+)$-convergent;
— the system is subterm $C(+)$-convergent.

3.2 Subterms modulo

In the case of subterm convergent TRSs (modulo the empty theory), it is sufficient for the deduction decision procedure to compute deducible terms just from the set of subterms occurring in the set of terms of the frame. That is, no new terms need to be added. When considering a non-empty theory E, we have to introduce an extended notion of subterm to capture the fact that matching modulo E is now performed when applying a rewrite step modulo E. Recall that E is assumed to be permutative. While this may seem somewhat restrictive, it allows for the consideration of theories such as AC and C which are found in a large number of security protocols.

Given a term t, $St(t)$ is the smallest set of terms including t such that

— if $u' =_E u$ and $u \in St(t)$, then $u' \in St(t)$,
— if $u \in St(t)$ and p is a non-root position of u, then $u|_p \in St(t)$.

Notice that $St(t)$ is finite since E is permutative. For a set of terms T, $St(T) = \bigcup_{t \in T} St(t)$, and for a substitution σ, $St(\sigma) = St(Ran(\sigma))$.

Example 1. Consider the theory APG defined in Section 3.1.2 and the following frames where $\tilde{n} = \{k_1, k_2\}$:

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

$$\phi = \nu\tilde{n}.\{x_1 \mapsto i(k_1 * a),\ x_2 \mapsto a * k_2\}$$
$$\psi = \nu\tilde{n}.\{x_1 \mapsto k_1 * a,\ x_2 \mapsto a * k_2\}$$
$$\phi' = \nu\tilde{n}.\{x_1 \mapsto i(k_1 * a),\ x_2 \mapsto a * k_1\}$$
$$\psi' = \nu\tilde{n}.\{x_1 \mapsto i(k_1),\ x_2 \mapsto i(k_1 * k_2)\}$$

According to the above definition of St, we get the following set of terms.

$$St(\phi) = \{i(k_1 * a), i(a * k_1), a * k_1, k_1 * a, a, k_1, a * k_2, k_2 * a, k_2\}$$
$$St(\psi) = \{k_1 * a, a * k_1, a * k_2, k_2 * a, k_1, k_2, a\}$$
$$St(\phi') = \{i(k_1 * a), i(a * k_1), a * k_1, k_1 * a, k_1, a\}$$
$$St(\psi') = \{i(k_1), i(k_1 * k_2), i(k_2 * k_1), k_1 * k_2, k_2 * k_1, k_1, k_2\}$$

Proposition 1. *For any terms t, t', $t =_E t'$ implies $St(t) = St(t')$, and for any position p in t, $St(t|_p) \subseteq St(t)$.*

The following result states that we cannot generate a new term outside $St(t)$ by rewriting terms in $St(t)$ (except the ground right-hand sides of R).

Lemma 2. *If $l\sigma =_E t$, then for any position p of l, $(l|_p)\sigma \in St(t)$.*

Proof. By structural induction on l.

If l is a variable, this is trivial since the only possible position is ϵ and $l|_\epsilon = l$.

Assume $l = f(l_1, \ldots, l_m)$ and σ is a substitution such that $f(l_1, \ldots, l_m)\sigma =_E t$.

If there is an equational step at the root position, then there exist some terms $g_1, \ldots, g_m$ such that $l_1\sigma =_E g_1, \ldots, l_m\sigma =_E g_m$ and $f(g_1, \ldots, g_m) =_E t$. By definition of $St(t)$ and Proposition 1, the terms $g_1, \ldots, g_m$ are in $St(t)$, and so $l_1\sigma, \ldots, l_m\sigma \in St(t)$.

If there is no equational step at the root position, then t is of the form $f(t_1, \ldots, t_m)$ and $l_1\sigma =_E t_1, \ldots, l_m\sigma =_E t_m$. By definition of $St(t)$ and Proposition 1, the terms $t_1, \ldots, t_m$ are in $St(t)$, and so $l_1\sigma, \ldots, l_m\sigma \in St(t)$. □

4. Decision Procedures for Shallow Permutative Theories

In this section, we construct new decision procedures for deduction and static equivalence where E is a shallow permutative theory, for example, E is Commutativity. We start by considering deduction which will also be needed when considering the problem of static equivalence.

4.1 Deduction

The decision procedure for the deduction problem requires the computation of some finite deducible terms defining the so-called *completion* of a given frame.

Definition 4. *Let $\phi = \nu\tilde{n}.\sigma$ be a normalized frame. The set of terms $D_*(\phi)$ is the smallest set D such that:*

(1) $Ran(\sigma) \subseteq D$,
(2) if $t_1, \ldots, t_n \in D$ and $f(t_1, \ldots, t_n) \in St(\sigma)$ then $f(t_1, \ldots, t_n) \in D$,
(3) if $t \in D$, $t' \in St(\sigma)$, $t =_E t'$, then $t' \in D$,
(4) if there is a root reduction $s[\bar{d}] \rightarrow^{\epsilon}_{R,E} t$ where $|s| \leq |R|$, $fn(s) \cap \tilde{n} = \emptyset$, $\bar{d} \in D$ and $t \in St(\sigma)$, then $t \in D$.

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

Let $\sigma_* = \sigma\{\chi_u \mapsto u \mid u \in D_*(\phi)\backslash Ran(\sigma)\}$ *where* χ_u *is a fresh variable. The frame* $\phi_* = \nu\tilde{n}.\sigma_*$ *is called the* completion of ϕ with respect to contexts bounded by $|R|$. *Given a recipe* ζ_u *for each* $u \in D_*(\phi)\backslash Ran(\sigma)$, *the substitution* $\{\chi_u \mapsto \zeta_u \mid u \in D_*(\phi)\backslash Ran(\sigma)\}$ *is called a* recipe substitution *of* ϕ *and is denoted by* ζ_ϕ.

Example 2. Consider the frames ϕ and ψ from Example 1. Now let us compute the sets $D_*(\phi)$ and $D_*(\psi)$, and recipe substitutions ζ_ϕ and ζ_ψ. One can check that $D_*(\phi) = \{i(k_1 * a),\ a * k_2,\ a,\ i(a * k_1),\ k_2 * a,\ k_1 * a,\ a * k_1\}$ and $D_*(\psi) = \{k_1 * a,\ a * k_2,\ a,\ a * k_1,\ k_2 * a\}$. The symbol $a \in St(\sigma)$ is contained in these sets due to the second item of Definition 4. However, we cannot use this rule for elements of $\tilde{n}$, which rules out k_1 and k_2. In addition, e is not contained in these sets since $e \notin St(\phi)$. Therefore, for $\phi = \nu\tilde{n}.\sigma$, we get:

$$\sigma_* = \sigma\{x_3 \mapsto i(a * k_1),\ x_4 \mapsto k_2 * a,\ x_5 \mapsto a,\ x_6 \mapsto k_1 * a,\ x_7 \mapsto a * k_1\}$$
$$\zeta_\phi = \{x_3 \mapsto x_1,\ x_4 \mapsto x_2,\ x_5 \mapsto a,\ x_6 \mapsto i(x_1),\ x_7 \mapsto i(x_1)\}$$

For the frame $\psi = \nu\tilde{n}.\tau$, we obtain:

$$\tau_* = \tau\{x_3 \mapsto a * k_1,\ x_4 \mapsto k_2 * a,\ x_5 \mapsto a\}$$
$$\zeta_\psi = \{x_3 \mapsto x_1,\ x_4 \mapsto x_2,\ x_5 \mapsto a\}$$

The decision procedure is based on the following reduction lemma, using the facts that the completion is computable and the deduction problem is decidable in the empty equational theory.

Lemma 3 (Deduction). *Let* $RE = R \cup E$ *where* R *is any subterm E-convergent TRS and* E *is any shallow permutative theory. For any normalized frame* ϕ *and any normalized term* t, *we have that* $\phi \vdash_{RE} t$ *if and only if* $\phi_* \vdash t$.

Proof. See Section 4.3. □

Example 3. (Example 1 continued.)

- The term $(k_2 * a) * (k_1 * a)$ is deduced from ϕ modulo *APG* since it is deduced from ϕ_* thanks to the recipe $x_4 * x_6$.
- The term k_1 is deduced from ψ' modulo *APG*, thanks to the recipe $i(x_1)$.
- The term $k_1 * (k_2 * k_1)$ is deduced from ψ' modulo *APG*, thanks to the recipe $i(x_1) * i(x_2)$.

4.2 Static equivalence

The decision procedure for the static equivalence is based on the computation of small equalities bounded by the size of R.

Definition 5. *Let* $\phi = \nu\tilde{n}.\sigma$ *be a normalized frame. Consider the following sets of terms:* $Bt(R) = \{t \mid |t| \leq |R|\}$; *and* $Gr(R) = \{r \mid l \to r \in R, r \text{ is ground}\}$. *Given a recipe substitution* ζ_ϕ *of* ϕ *as introduced in Definition 4, the set* $Eq^B_\zeta(\phi)$ *is the set of equalities* $t\zeta_\phi = t'\zeta_\phi$ *such that* $(t\zeta_\phi =_{RE} t'\zeta_\phi)\phi$ *and* $t, t' \in Bt(R) \cup Gr(R)$.

Example 4. Let us look at Eq^B_ζ for some of the frames from Example 1. Since these sets can be large, but finite, we will not list every equation in the set. Let us consider $Eq^B_\zeta(\phi)$. First are all the equalities that consist of terms, t and t', of size 0, that is, variables:

$$x_1 = x_1,\ x_2 = x_2,\ x_1 = x_3,\ x_2 = x_4 \ldots$$

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

Notice that each of these satisfies Definition 5. For example, $(x_2\zeta_\phi)\sigma = a * k_2 =_{RE} k_2 * a = \sigma(\zeta_\phi x_4)$. We also have equalities between size 1 terms:

$$i(x_1) = i(x_1),\ i(x_2) = i(x_2),\ \ldots\ i(x_1) = i(x_3),\ \ldots$$

$$x_1 * x_1 = x_1 * x_1,\ \ldots,\ x_1 * x_2 = x_3 * x_4 \ldots$$

The same applies for these equalities. For example, $((x_1 * x_2)\zeta_\phi)\sigma = i(k_1 * a) * (a * k_2) =_{RE} i(a * k_1) * (k_2 * a) = ((x_3 * x_4)\zeta_\phi)\sigma$. We need to also include mixed sized equalities such as between size 0 and size 1 terms:

$$x_3 = i(x_5),\ i(x_3) = x_1,\ i(x_3) = x_5,\ e * x_7 = x_7\ \ldots$$

Note that equalities such as $x_7 * k_1 = x_1$ are not included since $fn(x_7 * k_1) \cap \tilde{n} \neq \emptyset$.

$|R| = 2$ thus the final and largest, in terms of term size, set of equalities is between two terms of size 2:

$$e * x_1 = e * x_1,\ e * x_1 = e * x_3,\ \ldots,\ i(i(x_1)) = i(i(x_3)), \ldots$$

To get a decision procedure, it remains to show that checking small equalities defined by Eq_ζ^B is sufficient to prove the static equivalence of the two input frames. Note that the check of each of these equalities is effective since the *RE*-equality is decidable.

Lemma 4 (Static Equivalence). *Let $RE = R \cup E$ where R is any subterm E-convergent TRS and E is any shallow permutative theory. For any normalized frames ϕ and ψ, we have that $\phi \approx_{RE} \psi$ iff $\psi \models Eq_\zeta^B(\phi)$ and $\phi \models Eq_\zeta^B(\psi)$.*

Proof. See Section 4.3. □

Example 5. Consider the frames ϕ, ϕ', and ψ from Example 1. We have $\phi \not\approx_{RE} \phi'$ since $i(x_1) =_{RE} x_2 \in Eq_\zeta^B(\phi')$ and $i(x_1)\phi \neq_{RE} x_2\phi$. For the two frames ϕ and ψ, we can show $\phi \approx_{RE} \psi$ by checking that $\psi \models Eq_\zeta^B(\phi)$ and $\phi \models Eq_\zeta^B(\psi)$.

According to the above reduction lemmas (Lemmas 3 and 4), we obtain the following result:

Theorem 1. *Let $RE = R \cup E$ where E is any shallow permutative theory and R is any subterm E-convergent TRS. Then, deduction and static equivalence are decidable in RE.*

To prove both reduction lemmas (Lemmas 3 and 4) and so Theorem 1, we reuse the same approach as in Abadi and Cortier (2004, 2006), by applying two technical lemmas introduced in Section 4.3, namely Lemma 7 for E and Lemma 9 for R modulo E. To prove these lemmas, we use some properties satisfied by a shallow permutative theory E. With shallow permutative theories, we have identified a class of theories E for which we can apply exactly the same approach as in Abadi and Cortier (2004, 2006) to get new decidability results for equational rewrite systems which are both subterm and E-convergent.

Theorem 1 applies to the subterm C-convergent rewrite systems, such as the ones listed in Section 3.1. The rewrite system can be empty, which means that the deduction and the static equivalence problems are decidable in shallow permutative theories. Commutativity is perhaps the most popular shallow permutative axiom, but obviously it is not the only one, for example, $f(x, y, z) = f(z, x, y)$ and $x + 0 = 0 + x$ are also shallow permutative. Moreover, a union of shallow permutative theories remains shallow permutative, and so Theorem 1 can be directly applied to this union of theories, for instance to handle a union of several commutative symbols.

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

4.3 Correctness proofs

Recall that, given a frame $\phi = \nu\tilde{n}.\sigma$ and a term t, $t\phi$ denotes by a slight abuse of notation the term $t\sigma$.

Remark 1. When we use the notation $t\phi$, we assume that a variable x does not occur in t if $x\phi$ is a ground term occurring in the axioms of E. This can be assumed without loss of generality since that term, say t', can be used as a subterm of t since t' satisfies the name restriction of ϕ.

Lemma 5. *Assume E is shallow permutative. For any term s satisfying the name restriction, if $s\phi_* =_E u$, then there exists a term t satisfying the name restriction of ϕ_* such that $u = t\phi_*$ and $|s| = |t|$.*

Proof. Let us focus on an equational step $s\phi_* \leftrightarrow_E u$. Then, the generalization to $s\phi_* \leftrightarrow_E^* u$ can be easily proved by induction on the length of the derivation.

If the equational step $\leftrightarrow_E$ is applied at a position of a non-variable term s, then ϕ_* belongs to the substitution part of the equational step because E is shallow permutative, and the term u can be expressed as a term $t\phi_*$ with $|t| = |s|$. Note that, due to the restrictions on E, an equational step $\leftrightarrow_E$ will not increase the size of a term.

Otherwise, the equational step $\leftrightarrow_E$ is necessarily applied in ϕ_*, which means that there exists a variable x at a position p of s such that $x\phi_* \leftrightarrow_E u|_p$. By definition of ϕ_*, there exists a variable y such that $y\phi_* = u|_p$. Therefore, we have $s\phi_* \leftrightarrow_E u = (s[y]_p)\phi_*$, and we can choose $t = (s[y]_p)$. □

Lemma 6. *Assume E is shallow permutative. For any non-variable terms $s = f(\bar{s})$ and $t = g(\bar{t})$ satisfying the name restriction, if $s\phi_* =_E t\phi_*$ then*

- *$f = g$ and $\bar{s}\phi_* =_E \bar{t}\phi_*$,*
- *or there exist terms $\bar{l}, \bar{r}, \bar{u}, \bar{v}$ and a substitution μ such that $\bar{l}, \bar{r}$ are either variables or ground terms, $\bar{u}, \bar{v}$ satisfy the name restriction of ϕ, $|\bar{u}| = |\bar{s}|$, $|\bar{v}| = |\bar{t}|$, $\bar{s}\phi_* =_E \bar{u}\phi_* = \bar{l}\mu$, $\bar{t}\phi_* =_E \bar{v}\phi_* = \bar{r}\mu$, and $f(\bar{l}) \longleftrightarrow_E^{\epsilon} g(\bar{r})$.*

Proof. Since E is a shallow theory, it has a resolvent presentation which remains shallow. Therefore, if $s\phi_* =_E t\phi_*$, then

- $s\phi_* = f(\bar{s}\phi_*)$, $t\phi_* = f(\bar{t}\phi_*)$, and $\bar{s}\phi_* =_E \bar{t}\phi_*$,
- or $s\phi_* = f(\bar{s}\phi_*)$, $t\phi_* = g(\bar{t}\phi_*)$ and there exist terms $\bar{l}, \bar{r}$ and a substitution μ such that $\bar{l}, \bar{r}$ are either variables or ground terms, $\bar{s}\phi_* =_E \bar{l}\mu$, $\bar{t}\phi_* =_E \bar{r}\mu$, and $f(\bar{l}) \longleftrightarrow_E^{\epsilon} g(\bar{r})$. By Lemma 5, there exist terms $\bar{u}$ and $\bar{v}$ satisfying the name restriction such that $|\bar{u}| = |\bar{s}|$ $|\bar{v}| = |\bar{t}|$ and $\bar{u}\phi_* = \bar{l}\mu$, $\bar{v}\phi_* = \bar{r}\mu$. □

The following lemma corresponds to Lemma 3 in the appendix of Abadi and Cortier (2004).

Lemma 7. *Let $RE = R \cup E$ where R is any subterm E-convergent TRS and E is any shallow permutative theory. For any terms s and t satisfying the name restriction, if $s\phi_* =_E t\phi_*$ and $\psi \models Eq_\zeta^B(\phi)$, then $(s\zeta_\phi)\psi =_{RE} (t\zeta_\phi)\psi$.*

Proof. By induction on $|s| + |t|$.

- Base case: if $|s|$ and $|t|$ are less than $|R|$, then it is true by definition of Eq_ζ^B.
- Inductive step:

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

(A) Consider $s = f(\bar{s})$ and $t = g(\bar{t})$. By applying Lemma 6, two cases are possible:
  (i) $f = g$ and $\bar{s}\phi_* =_E \bar{t}\phi_*$. By applying the induction hypothesis, we get $(\bar{s}\zeta_\phi)\psi =_E (\bar{t}\zeta_\phi)\psi$, and so $(s\zeta_\phi)\psi = f((\bar{s}\zeta_\phi)\psi) =_E f((\bar{t}\zeta_\phi)\psi) = (t\zeta_\phi)\psi$.
  (ii) By Lemma 6, we have $\bar{s}\phi_* =_E \bar{u}\phi_* = \bar{l}\mu$ and $\bar{t}\phi_* =_E \bar{v}\phi_* = \bar{r}\mu$ such that $f(\bar{l}) \overset{\epsilon}{\longleftrightarrow}_E g(\bar{r})$. By applying the induction hypothesis, we get $(\bar{s}\zeta_\phi)\psi =_{RE} (\bar{u}\zeta_\phi)\psi$ and $(\bar{t}\zeta_\phi)\psi =_{RE} (\bar{v}\zeta_\phi)\psi$. The terms $\bar{l}, \bar{r}$ are either variables or ground terms, and so $u_i\phi_* = x\mu = v_j\phi_*$ for each variable x in $\bar{l}, \bar{r}$. By the induction hypothesis, we get $(u_i\zeta_\phi)\psi =_{RE} (v_j\zeta_\phi)\psi$. Hence, there exists a substitution μ' such that
  – $(\bar{s}\zeta_\phi)\psi =_{RE} (\bar{u}\zeta_\phi)\psi =_{RE} \bar{l}\mu'$
  – $(\bar{t}\zeta_\phi)\psi =_{RE} (\bar{v}\zeta_\phi)\psi =_{RE} \bar{r}\mu'$
  Consequently, we have
  – $(s\zeta_\phi)\psi = f((\bar{s}\zeta_\phi)\psi) =_{RE} f((\bar{u}\zeta_\phi)\psi) =_{RE} f(\bar{l}\mu')$
  – $(t\zeta_\phi)\psi = g((\bar{t}\zeta_\phi)\psi) =_{RE} g((\bar{v}\zeta_\phi)\psi) =_{RE} g(\bar{r}\mu')$
  where $f(\bar{l}\mu') =_E g(\bar{r}\mu')$.

(B) Consider $s = f(s_1, \ldots, s_r)$ and t is a variable x. Assume $x\phi_* =_E s\phi_*$. We have that $f(s_1\phi_*, \ldots, s_r\phi_*) =_E x\phi_*$. Let $N_i = s_i\phi_*$ for $i = 1, \ldots, r$, and $M = x\phi_*$. Since $f(N_1, \ldots, N_r) =_E M$ and $M \in St(\phi)$, then $N_i \in St(\phi)$. Since $N_i = s_i\phi_*$ and $N_i \in St(\phi)$, we have that $N_i \in Ran(\phi_*)$, and so there exists some recipe ζ_{N_i}. Since $M =_E f(N_1, \ldots, N_r)$, we have $\zeta_M\phi =_E f((\zeta_{N_1}\phi), \ldots, (\zeta_{N_r}\phi))$, and $\zeta_M\psi =_{RE} f((\zeta_{N_1}\psi), \ldots, (\zeta_{N_r}\psi))$ by assumption on ψ.
Since $N_i =_E s_i\phi_*$, we have that $\zeta_{N_i}\psi =_{RE} (s_i\zeta_\phi)\psi$ by the induction hypothesis. Then,

$$(x\zeta_\phi)\psi = \zeta_M\psi =_{RE} (f(s_1, \ldots, s_r)\zeta_\phi)\psi = (s\zeta_\phi)\psi$$ □

The Lemma 4 given in the appendix of Abadi and Cortier (2004) can be adapted as follows:

Lemma 8. *Let $RE = R \cup E$ where R is any subterm E-convergent TRS and E is any shallow permutative theory. For any term s satisfying the name restriction and any term t such that $s\phi_* \to_R t$, there exists a term u satisfying the name restriction such that $t = u\phi_*$ and for any frame ψ such that $\psi \models Eq^B_\zeta(\phi)$, $(s\zeta_\phi)\psi =_{RE} (u\zeta_\phi)\psi$.*

Proof. (i) Let us first assume that the rewrite step occurs at the root position. Suppose $s\phi_* = l\mu$ with $l \to r \in R$. There are two possibilities:
— Assume there exists some substitution μ' such that $s = l\mu'$. The substitution μ' satisfies the name restriction of ϕ since s satisfies it. We have $\mu = \mu'\phi_*$, and so $s\phi_* = l\mu'\phi_* \to_R r\mu'\phi_*$, where $r\mu'$ satisfies the name restriction of ϕ thanks to the form of rules in R. Moreover, for any frame ψ with the same name restriction as the one of ϕ, the same rewrite step applies on $s\zeta_\phi\psi = l\mu'\zeta_\phi\psi$ and we get $s\zeta_\phi\psi = l\mu'\zeta_\phi\psi \to_R r\mu'\zeta_\phi\psi$.
— Otherwise, it is impossible to have $|s| > |R|$ and $s\phi_* = l\mu$. Consequently, $|s| \leq |R|$ and only two cases are possible for the rewrite rule $l \to r$ since R is subterm E-convergent:
  – If r is a ground term, then $(s\zeta_\phi)\phi =_{RE} s\phi_* \to_R r = r\phi_* =_{RE} (r\zeta_\phi)\phi$ (where r satisfies the name restriction of ϕ). By definition of Eq^B_ζ and by assumption on ψ, we have $(s\zeta_\phi)\psi =_{RE} r =_{RE} (r\zeta_\phi)\psi$.
  – If r is a subterm of l, then by definition of ϕ_* (cf. Definition 4(4)), there exists some variable x such that $(s\zeta_\phi)\phi =_{RE} s\phi_* \to_R x\phi_* =_{RE} (x\zeta_\phi)\phi$. By definition of Eq^B_ζ and by assumption on ψ, we have $(s\zeta_\phi)\psi =_{RE} (x\zeta_\phi)\psi$.

(ii) Let us now assume that the rewrite step occurs below the root position. There exists a position $p \neq \epsilon$ such that $s'\phi_* = (s'\phi_*)[s\phi_*]_p$ with $s\phi_* \overset{\epsilon}{\to}_R t$. By the case (*i*) above, there exists a term u such that $t = u\phi_*$ and $(s\zeta_\phi)\psi =_{RE} (u\zeta_\phi)\psi$. Then, we have

$$s'\phi_* \to_R (s'\phi_*)[t]_p = (s'\phi_*)[u\phi_*]_p = (s'[u]_p)\phi_*$$

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

and

$$(s'\zeta_\phi)\psi = (s'\zeta_\phi\psi)[s\zeta_\phi\psi]_p =_{RE} (s'\zeta_\phi\psi)[u\zeta_\phi\psi]_p = (s'[u]_p)\zeta_\phi\psi$$ □

This lemma can now be extended as follows using Lemma 7.

Lemma 9. *Let $RE = R \cup E$ where R is any subterm E-convergent TRS and E is any shallow permutative theory. For any term s satisfying the name restriction and any term t with $s\phi_* \to_{R,E} t$, there exists a term u satisfying the name restriction such that $t =_E u\phi_*$. In addition, for any frame ψ such that $\psi \models Eq_\zeta^B(\phi)$, $(s\zeta_\phi)\psi =_{RE} (u\zeta_\phi)\psi$.*

Proof. If $s\phi_* \to_{R,E} t$, then (according to Lemma 5) there exists a term s' satisfying the name restriction such that $s\phi_* =_E s'\phi_*$ and $s'\phi_* \to_R t$. By Lemma 7, we have $(s\zeta_\phi)\psi =_{RE} (s'\zeta_\phi)\psi$. By Lemma 8, there exists a term u satisfying the name restriction such that $(s'\zeta_\phi)\psi =_{RE} (u\zeta_\phi)\psi$. Consequently, we get $(s\zeta_\phi)\psi =_{RE} (u\zeta_\phi)\psi$. □

We are now ready to prove the two reduction lemmas, namely Lemmas 3 and 4. For both lemmas, we prove the non-obvious direction:

— **Lemma 3.**

Proof. (Only if direction) Assume $s\phi_* =_{RE} t$ where t is normalized. According to Lemma 9, there exists a rewrite proof of the form

$$s\phi_* \to_{R,E} \circ =_E \cdots \to_{R,E} \circ =_E s'\phi_* =_E t$$

where s' satisfies the name restriction of ϕ_*, equivalently the name restriction of ϕ. □

— **Lemma 4.**

Proof. (If direction) Let $Eq_\zeta(\phi)$ be the set of all equalities $s\zeta_\phi = t\zeta_\phi$ such that $(s\zeta_\phi =_{RE} t\zeta_\phi)\phi$. Consider any $s\zeta_\phi = t\zeta_\phi \in Eq_\zeta(\phi)$. According to Lemma 9, there exists a rewrite proof of the form

$$(s\zeta_\phi)\phi =_{RE} s\phi_* \to_{R,E} \circ =_E \cdots \to_{R,E} \circ =_E s'\phi_*$$

$$(t\zeta_\phi)\phi =_{RE} t\phi_* \to_{R,E} \circ =_E \cdots \to_{R,E} \circ =_E t'\phi_*$$

where $s'\phi_* =_E t'\phi_*$ and s', t' satisfy the name restriction of ϕ_* (equivalently, the name restriction of ϕ).
By Lemma 9, we have $(s\zeta_\phi)\psi =_{RE} (s'\zeta_\phi)\psi$ and $(t\zeta_\phi)\psi =_{RE} (t'\zeta_\phi)\psi$. By Lemma 7, we have $(s'\zeta_\phi)\psi =_{RE} (t'\zeta_\phi)\psi$. Hence, $(s\zeta_\phi)\psi =_{RE} (t\zeta_\phi)\psi$, which means that $\psi \models Eq_\zeta(\phi)$.
In a symmetric way, we can show that $\phi \models Eq_\zeta(\psi)$. Then, we can conclude since

$$\phi \approx_{RE} \psi \text{ iff } \psi \models Eq_\zeta(\phi) \text{ and } \phi \models Eq_\zeta(\psi)$$ □

5. Equational Variants in Syntactic Permutative Theories

In this section, we start investigating the possibility to go beyond the case E is shallow permutative, by considering E is syntactic permutative, for example, E is Associativity–Commutativity, and R is E-convergent. In this general case, we need to consider an additional finiteness assumption introduced below. The investigated approach relies on the possibility of computing a finite representation of all the terms that are matched modulo E by the left-hand sides of the TRS R.

Definition 6. *An* E-variant *of a term l is a pair (t, σ) such that $t =_E l\sigma$ and $Dom(\sigma)$ is included in $fv(l)$. Given two E-variants (u, θ) and (v, γ) of a term l, (u, θ) is* more general *than (v, γ), denoted by $(u, \theta) \leq_E (v, \gamma)$ if there exists a substitution τ such that $u\tau =_E v$ and $\theta\tau =_E \gamma$. A complete set*

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

of E-variants of l, denoted by $CV_E(l)$*, is a set of E-variants of l such that for any E-variant* (v, γ) *of l, there exists* $(u, \theta) \in CV_E(l)$ *such that* $(u, \theta) \leq_E (v, \gamma)$*. The equational theory E is said to have the Finite Equational Variant Property* (FEVP, for short) *if any term admits a finite complete set of E-variants.*

Equational variants are analogous to variants defined with respect to a convergent TRS (Comon and Delaune 2005; Escobar *et al.*, 2012; Meseguer 2018). When R denotes a convergent TRS, a R-variant of l is defined in the literature as a pair (t, σ) such that $t = (l\sigma)\downarrow_R$ and t, σ are both R-normalized. Given two R-variants (u, θ) and (v, γ) of l, (u, θ) is said to be more general than (v, γ) if there exists a substitution τ such that $u\tau = v$ and $\theta\tau = \gamma$. A complete set of R-variants of l is defined in the same way as in Definition 6: it is a set $CV_R(l)$ of R-variants of l such that for any R-variant (v, γ) of l, there exists $(u, \theta) \in CV_R(l)$ such that (u, θ) is more general than (v, γ). Then, R is said to have the Finite Variant Property (FVP, for short) if any term admits a finite complete set of R-variants.

Proposition 2. *If a convergent TRS has the FVP, then its equational theory has the FEVP.*

The following notion of complete set of E-matched terms is instrumental to show the FEVP.

Definition 7. *A term t is said to be* E-matched by l *if there exists some substitution* σ *such that* (t, σ) *is an E-variant of l. The set of terms E-matched by l is denoted by* $MT_E(l)$*. A* complete set of terms E-matched by l *is a subset of* $MT_E(l)$ *denoted by* $CMT_E(l)$ *such that for any* $t \in MT_E(l)$*, there exist* $t' \in CMT_E(l)$*, and a substitution* μ *satisfying the following property:* $t =_E t'\mu$ *and for any E-variant* (t, σ) *of l, there exists an E-variant* (t', σ') *of l such that* $\sigma =_E \sigma'\mu$*.*

Given a non-empty TRS R, a complete set of terms E-matched by R *is* $CMT_E(R) = \{t \mid t \in CMT_E(l), l \to r \in R\}$.

Unsurprisingly, the finiteness of $CMT_E(l)$ for each term l suffices to show the FEVP:

Proposition 3. *Assume E is permutative. For any term l, if* $CMT_E(l)$ *is finite, then* $\{(t, \sigma) \mid t \in CMT_E(l), l\sigma =_E t\}$ *is a finite* $CV_E(l)$*.*

Proof. Since E is assumed to be permutative, the set of substitutions σ such that $l\sigma =_E t$ is finite for any terms l, t. □

When E is the empty theory and l is any term, the singleton $\{l\}$ is a $CMT_E(l)$. When E is an arbitrary theory, the singleton $\{l\}$ can be a $CMT_E(l)$ for some particular terms l, for example, $CMT_E(x + 0) = \{x + 0\}$ for $E = AC(+)$. As stated below, the singleton $\{l\}$ is always a $CMT_E(l)$ under a simple assumption that bears some similarities with the unique matching property used in Chevalier and Rusinowitch (2008) to get decidability of ground intruder systems corresponding to deduction problems in a hierarchical combination of theories.

Proposition 4. *If l is any variable, then* $\{l\}$ *is a* $CMT_E(l)$*. If l is any non-variable term and for any term t in* $MT_E(l)$*, the match-equation* $l \leq_E^? t$ *admits a unique solution modulo E, then* $\{l\}$ *is a* $CMT_E(l)$*.*

Proof. Consider the identity substitution ε. For any term l, $l\varepsilon = l$, and so (l, ε) is an E-variant of l.

- Let l be any variable. For any E-variant (t, σ) of l, we have $t =_E l\sigma$ and $\sigma =_E \varepsilon\sigma$ where (l, ε) is an E-variant of l.
- Let l be any non-variable term. By assumption, for any term t in $MT_E(l)$, there exists a unique substitution σ such that $t =_E l\sigma$ and $\sigma =_E \varepsilon\sigma$.

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

$\mathbf{Merge}_{SV}$ $\quad \{x \leq^? y\} \cup P \vdash P\sigma$
where x, y are subject variables, $\sigma = \{x \mapsto y\}$,
if no rule from MSP is applicable.

$\mathbf{Inst}_{SV}$ $\quad \{s \leq^? x\} \cup P$
$\vdash (\{s \leq^? x\} \cup P)\sigma$
where x is a subject variable and s is rooted by a function symbol, $\sigma = \{x \mapsto f(x_1, \ldots, x_n)\}$, f is a function symbol, $x_1, \ldots, x_n$ are fresh pairwise distinct variables,
if no rule from MSP $\cup \{\mathbf{Merge}_{SV}\}$ is applicable.

Figure 2. MTG additional rules

By definition of a complete set of E-matched terms, this implies in both cases that $\{l\}$ is a $CMT_E(l)$. □

The case of a variable l being easily solved, we focus below on the case where l is a non-variable term.

5.1 The MTG *procedure for computing a complete set of matched terms*

We now study a general procedure that, when terminating, computes a finite $CMT_E(l)$. Consider the inference system MTG defined by the set of rules in MSP given in Figure 1 plus the two additional rules $\mathbf{Merge}_{SV}$, $\mathbf{Inst}_{SV}$ given in Figure 2.

Lemma 10. *Let l be an arbitrary non-variable term and x a fresh variable. Assume that any* MTG*-derivation starting from $\{l \leq^? x\}$ is terminating. Then, a $CMT_E(l)$ is given by the finite set MT of terms $x\sigma_P$ where*

- *P is a solved form such that $\{l \leq^? x\} \vdash^*_{\mathsf{MTG}} P$,*
- *σ_P is the composition of all substitutions applied by $\mathbf{Inst}_{SV}$ or $\mathbf{Merge}_{SV}$ in the derivation $\{l \leq^? x\} \vdash^*_{\mathsf{MTG}} P$.*

Proof. Given any solved form P such that $\{l \leq^? x\} \vdash^*_{\mathsf{MTG}} P$, let μ_P be the corresponding substitution and $\mu_{P|fv(l)}$ the restriction of μ_P to $fv(l)$. For any term $x\sigma_P \in MT$, $(x\sigma_P, \mu_{P|fv(l)})$ is an E-variant of l. Consequently, $MT \subseteq MT_E(l)$.

To show that MT is a complete set of E-matched terms, consider any term $t \in MT$ and the set SF of solved forms computed by MSP with $\{l \leq^? t\}$ as input. For any substitution θ, $\{P\theta \mid P \in SF\}$ corresponds to the set of solved forms computed by MSP with $\{l \leq^? t\theta\}$ as input. Since MSP is an E-matching algorithm, $\{\sigma\theta \mid \sigma \in CSU_E(l \leq^? t)\}$ is a $CSU_E(l \leq^? t\theta)$. Therefore, MT is a $CMT_E(l)$. □

We show in the following example that MTG may not terminate when E is the Associativity–Commutativity.

Example 6. It is difficult to get terminating MTG-derivations when $E = AC(+)$. Let us consider $l = (a + x)$. Starting from $a + x \leq^? x_0$, the only possibility is to apply $\mathbf{Inst}_{SV}$, leading to $a + x \leq^? x_1 + x_2$. The theory $E = AC(+)$ admits a resolvent presentation that consists of seven axioms. One of these axioms is $w_1 + (w_2 + w_3) = (w_1 + w_2) + w_3$. By applying **MatchMut** with this axiom, we get

$$a =^? w_1, x =^? w_2 + w_3, w_1 + w_2 \leq^? x_1, w_3 \leq^? x_2$$

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

The repeated application of **RemEq** and **Rep** leads to

$$w_1 \leq^? a, x =^? w_2 + x_2, a + w_2 \leq^? x_1, w_3 \leq^? x_2.$$

It contains $a + w_2 \leq^? x_1$ which is a renaming of the input, and so the MTG procedure loops in that case.

5.2 Termination of MTG *and closure under paramodulation*

As stated above, the MTG procedure is not guaranteed to terminate for an arbitrary E. However, there are classes of permutative theories for which the MTG procedure is guaranteed to terminate. One such class of permutative theories are those closed under paramodulation, which we prove below.

Lemma 11. *Consider* $\mathsf{MTG}_B = \mathsf{MSP}_B \cup \{\mathbf{Merge}_{SV}, \mathbf{Inst}_{SV}\}$ *where* MSP_B *is defined in Section 2.5 and* $\mathbf{Merge}_{SV}$, $\mathbf{Inst}_{SV}$ *are given in Figure 2. Let E be any permutative theory closed by paramodulation. For any non-variable term l and any fresh variable x, any* MTG_B*-derivation starting from* $\{l \leq^? x\}$ *is terminating and* MTG_B *provides a* $CMT_E(l)$ *in the same way as the one described in Lemma 10 for* MTG.

Proof. To prove termination, let us analyze the interaction between MSP_B and the additional rules $\mathbf{Merge}_{SV}$ and $\mathbf{Inst}_{SV}$:

- After the application of $\mathbf{Merge}_{SV}$, no MSP_B rule can be fired. Only $\mathbf{Inst}_{SV}$ can be possibly fired after the exhaustive application of $\mathbf{Merge}_{SV}$.
- After the application of $\mathbf{Inst}_{SV}$ on $s \leq^? x$, some MSP_B rule may be fired, and there are two possible cases:
  - If s is rooted by an unboxed symbol occurrence, then **MatchDec** or $\mathbf{MatchMut}_B$ applies and in both cases the number of unboxed symbol occurrences is strictly decreasing.
  - If s is rooted by a boxed symbol occurrence, then necessarily **MatchDec** applies and the number of unboxed symbol occurrences is not increasing but the multiset of sizes of terms is strictly decreasing.

The completeness follows from the proof of Lemma 10 and the fact that MSP_B is an E-matching algorithm.

□

According to Lemma 11 and Proposition 3, we get the following result:

Theorem 2. *If E is a permutative theory closed by paramodulation, then E has the FEVP.*

Example 7. Let us consider a small example of computing a $CMT_E(l)$, where l is $x + i(a)$ and $E = \{x + i(y) = i(y) + x\}$. We start by applying the MTG_B procedure on the input $\{x + i(a) \leq^? x_0\}$. Initially, no rules from MSP apply and only $\mathbf{Inst}_{SV}$ applies, resulting in three substitutions: $\{x_0 \mapsto x_1 + x_2\}$, $\{x_0 \mapsto i(x_1)\}$, and $\{x_0 \mapsto a\}$. Thus, each substitution causes a branch in the computation of the procedure. The branches generated by $\{x_0 \mapsto i(x_1)\}$, and $\{x_0 \mapsto a\}$ lead to a failure. Let us follow the branch corresponding to $\{x_0 \mapsto x_1 + x_2\}$ and the match-equation $x + i(a) \leq^? x_1 + x_2$. Now rules from MSP apply, including as a first rule **MatchDec** resulting in $\{x \leq^? x_1,\ i(a) \leq^? x_2\}$. Continuing in this fashion, we reach a solved form corresponding to the E-matched term $x_1 + i(a)$ where $x_1 + i(a) = x_0\{x_0 \mapsto x_1 + x_2\}\{x_2 \mapsto i(x_3)\}\{x_3 \mapsto a\}$. In addition to **MatchDec**, $\mathbf{MatchMut}_B$

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

also applies on $x + i(a) \leq^? x_1 + x_2$ and leads to

$$\left\{x =^? \boxed{x'},\ i(a) =^? \boxed{i(y')},\ \boxed{i(y')} \leq^? x_1,\ \boxed{x'} \leq^? x_2\right\}.$$

Then, $\mathbf{Inst}_{SV}$ can be applied on $\boxed{i(y')} \leq^? x_1$ with the substitution $\{x_1 \mapsto i(x_3)\}$. Following this branch, we get another solved form corresponding to the E-matched term $i(a) + x_2$ where $i(a) + x_2 = x_0\{x_0 \mapsto x_1 + x_2\}\{x_1 \mapsto i(x_3)\}\{x_3 \mapsto a\}$.

6. Decision Procedures for Syntactic Permutative Theories

From now on, we assume that E is syntactic permutative and R is a subterm E-convergent TRS admitting a finite $CMT_E(R)$, where $CMT_E(R)$ is introduced in Definition 7. Under this finiteness assumption, it is possible to define an appropriate notion of size for R modulo E.

Definition 8. *Given a non-empty TRS R admitting a finite $CMT_E(R)$, the* size of R modulo E, *denoted by $|R|$, is defined as follows:*

$$|R| = \max\{|t| \mid t \in CMT_E(R)\}.$$

We refer below to some notions introduced in Section 4 with respect to $|R|$, such as the completion of a frame (Definition 4) and a set of terms of size bounded by $|R|$ (Definition 5). These notions are defined in the same way in the context of this section, using now the size $|R|$ given in Definition 8.

Remark 2. The size of R modulo E is computable if and only if there exists a finite and computable $CMT_E(R)$.

Remark 3. When E is the empty theory, the size of R modulo E coincides with the size of R defined in Section 2 since $\{l \mid l \to r \in R\}$ is a $CMT_E(R)$.

In the following, we present reduction methods from $RE = R \cup E$ to the combined theory $\emptyset \cup E$ where $\emptyset$ denotes the empty Ω-theory. According to the combination result in Cortier and Delaune (2010), it is always possible to obtain decision procedures for both deduction and static equivalence in $\emptyset \cup E$ from the ones existing in E alone. This explains why $\emptyset \cup E$ is often simply denoted by E in the following two subsections. In Abadi and Cortier (2006), it has been observed that the decidability of static equivalence entails the decidability of deduction, provided that the signature includes a unary free function symbol. In fact, the encoding presented in Abadi and Cortier (2006) can be easily generalized to any non-constant free function symbol. Usually Ω contains at least a non-constant function symbol, and so the decidability of static equivalence in $\emptyset \cup E$ implies the decidability of deduction in E. Thus, we could be tempted to focus our attention on static equivalence only. However, as illustrated in Abadi and Cortier (2006) and Cortier and Delaune (2010), a decision procedure for the static equivalence usually requires a decision procedure for the deduction. In an analogous way, we first focus on deduction as a first step toward a decision procedure for the static equivalence.

6.1 Deduction

The decision procedure for the deduction problem in RE is based on the following reduction lemma.

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

Lemma 12 (Deduction). *Let $RE = R \cup E$ where E is any syntactic permutative theory and R is any subterm E-convergent TRS with a computable size of R modulo E. For any normalized frame ϕ and any normalized term t, we have that $\phi \vdash_{RE} t$ if and only if $\phi_* \vdash_E t$.*

Proof. See Section 6.3. □

Example 8. Consider $R = \{i(x) + i(a) \to x\}$, $E = \{i(x) + i(y) = i(y) + i(x)\}$. Let $\phi = \nu\{k\}.\{x_1 \mapsto i(a), x_2 \mapsto a + i(k)\}$ and $\psi = \nu\{k\}.\{x_1 \mapsto i(a), x_2 \mapsto i(k)\}$. One can check that $D_*(\phi) = \{i(a), a + i(k), a\}$ and $D_*(\psi) = \{i(a), i(k), a, k\}$. Once these sets of deducible terms are computed, the frames can be completed and used to reduce the deduction problem modulo RE to the deduction problem modulo E. For example, notice that the term $i(k) + k$ is deducible from ψ_* but not from ϕ_*.

6.2 Static equivalence

In a way similar to what is done for disjoint combinations (Cortier and Delaune 2010), we extend the input frames with the instantiation of recipes of all deducible terms occurring in the completions.

Definition 9. *Let $\phi = \nu\tilde{n}.\sigma$ be a frame. Let Π be a set of terms t such that $t\sigma$ is ground and t satisfies the name restriction of ϕ. The* Π-extension *of ϕ is the frame $\Pi\phi = \nu\tilde{n}.\{\chi_t \mapsto t \mid t \in \Pi\}\sigma$.*

Lemma 13. *Given any normalized frames $\phi = \nu\tilde{n}.\sigma$ and $\psi = \nu\tilde{n}.\tau$ such that $Dom(\sigma) = Dom(\tau)$, let $\bar{\phi} = (\Pi\phi)\downarrow_{R,E}$, $\bar{\psi} = (\Pi\psi)\downarrow_{R,E}$ where $\Pi = St(Ran(\zeta_\phi) \cup Ran(\zeta_\psi))$. Then, we have (i) $(\bar{\phi})_* = \bar{\phi}$ and $(\bar{\psi})_* = \bar{\psi}$; (ii) $\phi \approx_{RE} \psi$ if and only if $\bar{\phi} \approx_{RE} \bar{\psi}$.*

Proof.

(i) Notice that $(\sigma_*)_* = \sigma_*$. This is due to the fact that $St(\sigma_*) = St(\sigma)$. Therefore, completing the frame $\bar{\phi}$ (or $\bar{\psi}$) does not add new terms.

(i) Consider $\bar{\phi} = \nu\tilde{n}.\bar{\sigma}$, $\bar{\psi} = \nu\tilde{n}.\bar{\tau}$ and the substitution $\pi = \{\chi_t \mapsto t \mid t \in \Pi\}$. By definition, $\bar{\sigma} = \pi\sigma$ and $\bar{\tau} = \pi\tau$. Let us now prove the two directions:
   — Assume $\bar{\phi} \approx_{RE} \bar{\psi}$. Consider any terms s and t satisfying the name restriction of ϕ. We can assume without loss of generality that $(fv(s) \cup fv(t)) \cap Dom(\pi) = \emptyset$. The restriction of $\bar{\sigma}$ to $Dom(\sigma)$ coincides with σ, and so $(s =_{RE} t)\phi$ iff $(s =_{RE} t)\bar{\phi}$. Since $\bar{\phi} \approx_{RE} \bar{\psi}$, $(s =_{RE} t)\bar{\phi}$ iff $(s =_{RE} t)\bar{\psi}$. The restriction of $\bar{\tau}$ to $Dom(\tau)$ coincides with τ, and so $(s =_{RE} t)\bar{\psi}$ iff $(s =_{RE} t)\psi$. Thus, $\phi \approx_{RE} \psi$.
   — Assume $\phi \approx_{RE} \psi$. Consider any terms s and t satisfying the name restriction of $\bar{\phi}$. By definition of $\bar{\phi}$, $(s =_{RE} t)\bar{\phi}$ iff $(s\pi =_{RE} t\pi)\phi$. Since $\phi \approx_{RE} \psi$, $(s\pi =_{RE} t\pi)\phi$ iff $(s\pi =_{RE} t\pi)\psi$. By definition of $\bar{\psi}$, $(s\pi =_{RE} t\pi)\psi$ iff $(s =_{RE} t)\bar{\psi}$. Thus, $\bar{\phi} \approx_{RE} \bar{\psi}$. □

Example 9. Assume $R = \{i(i(x)) \to x\}$ and $E = \emptyset$. Consider the frames $\phi = \nu\{k\}.\{x \mapsto i(k)\}$ and $\psi = \nu\{k\}.\{x \mapsto k\}$. One can observe that $\phi \approx_{RE} \psi$. For this static equivalence problem, $\Pi = \{i(x), x\}$ where $i(x)$ is the recipe of k in ϕ. According to the definition of $\bar{\phi}$ and $\bar{\psi}$ introduced in Lemma 13, we have $\bar{\phi} = \nu\{k\}.\{x \mapsto i(k), x' \mapsto k, x'' \mapsto i(k)\}$ and $\bar{\psi} = \nu\{k\}.\{x \mapsto k, x' \mapsto i(k), x'' \mapsto k\}$. Again, one can observe that $\bar{\phi} \approx_{RE} \bar{\psi}$.

Example 10. Continuing Example 8, $\Pi = \{a, x_1 + x_2, x_1, x_2\}$ where $x_1 + x_2$ is the recipe of k in ψ. Then, $\bar{\phi} = \nu\{k\}.\{x_1 \mapsto i(a), x_2 \mapsto a + i(k), x_3 \mapsto a, x_4 \mapsto i(a) + (a + i(k)), x_5 \mapsto i(a), x_6 \mapsto a + i(k)\}$ and $\bar{\psi} = \nu\{k\}.\{x_1 \mapsto i(a), x_2 \mapsto i(k), x_3 \mapsto a, x_4 \mapsto k, x_5 \mapsto i(a), x_6 \mapsto i(k)\}$.

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

The decision procedure for the static equivalence computes small equalities obtained by considering a finite set of contexts derived from the left-hand sides of R.

Definition 10. *Let $\phi = \nu\tilde{n}.\sigma$ be a normalized frame. Consider the sets of terms $Bt(R) = \{t \mid |t| \leq |R|\}$ where $|R|$ is given in Definition 8 and $Gr(R)$ introduced in Definition 5. The set $Eq^B(\phi)$ is the set of equalities $t = t'$ such that $(t =_{RE} t')\phi$ and $t, t' \in Bt(R) \cup Gr(R)$.*

Example 11. Continuing Example 10, we obtain $Eq^B(\bar{\phi}) = \{i(a) = i(a),\ i(x_1) = i(x_1),\ i(x_3) = i(x_3),\ a = a,\ a + a = a + a,\ \ldots\}$ and $Eq^B(\bar{\psi}) = \{i(a) = i(a),\ i(x_1) = i(x_1),\ i(x_3) = i(x_3),\ i(x_4) = i(x_4),\ x_4 = x_4,\ a = a,\ \ldots,\ x_4 = x_2 + x_1,\ \ldots\}$.

To get a decision procedure, it remains to show that checking small equalities defined by Eq^B are sufficient to prove the static equivalence of the two input frames. Note that the check of each of these equalities is effective since the RE-equality is decidable.

The decision procedure for static equivalence in RE is based on the following reduction lemma:

Lemma 14 (Static Equivalence). *Let $RE = R \cup E$ where E is any syntactic permutative theory and R is any subterm E-convergent TRS with a computable size of R modulo E. For any normalized frames $\bar{\phi}$ and $\bar{\psi}$ introduced in Lemma 13, we have $\bar{\phi} \approx_{RE} \bar{\psi}$ iff $\bar{\psi} \models Eq^B(\bar{\phi})$ and $\bar{\phi} \models Eq^B(\bar{\psi})$ and $\bar{\phi} \approx_E \bar{\psi}$.*

Proof. See Section 6.3. □

Example 12. Continuing Example 11, notice that $x_4 = x_2 + x_1 \in Eq^B(\bar{\psi})$. However, $x_4 = x_2 + x_1 \notin Eq^B(\bar{\phi})$. In other words, $\bar{\phi} \not\models Eq^B(\bar{\psi})$. Therefore, the frames are not statically equivalent. This is due to the fact that in the second frame, ψ, the adversary would have knowledge of k but not so in the first frame, ϕ. Thus, the adversary is able to use this knowledge to distinguish the frames.

According to the above reduction lemmas, we get the following result.

Theorem 3. *Let $RE = R \cup E$ where E is any syntactic permutative theory, R is any subterm E-convergent TRS with a computable size of R modulo E and both deduction and static equivalence are decidable in E. Then, both deduction and static equivalence are decidable in RE.*

The proof of Theorem 3 is given below.

6.3 Correctness proofs

Let us first rephrase the Lemma 4 proved in the appendix of Abadi and Cortier (2004) by using the frames $\bar{\phi}$ and $\bar{\psi}$. One can notice that we use a definition for Eq^B which is more refined than the rough one considered in Abadi and Cortier (2004). Instead of considering contexts whose sizes are bounded by the maximal size of the left-hand sides in R as in Abadi and Cortier (2004), our definition takes into account only the contexts for which the frame is needed for being matched by some left-hand side.

Lemma 15. *Let $RE = R \cup E$ where E is any syntactic permutative theory and R is any subterm E-convergent with a computable size of R modulo E. Assume $\bar{\phi} \approx_E \bar{\psi}$ and $\bar{\psi} \models Eq^B(\bar{\phi})$. For any term s satisfying the name restriction and for any term t such that $s\bar{\phi} \to_{R,E} t$, there exists a term u satisfying the name restriction such that $t =_E u\bar{\phi}$ and $s\bar{\psi} =_{RE} u\bar{\psi}$.*

Proof. The structure of the proof is similar to the one developed for Lemma 8.

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

(i) Let us first assume that the rewrite step occurs at the root position. Suppose $s\bar{\phi} =_E l\mu$ with $l \to r \in R$. There are two possibilities:
— Assume there exist some term s_c in $CMT_E(l)$ and a substitution θ such that $s =_E s_c\theta$. Both s_c and θ satisfy the name restriction of $\bar{\phi}$ since s satisfies it. By definition of a $CMT_E(l)$, for the E-variant $(s\bar{\phi}, \mu)$ of l, there exists some E-variant (s_c, θ') of l such that $\mu =_E \theta'(\theta\bar{\phi})$. The substitution θ' satisfies the name restriction of $\bar{\phi}$ since s_c satisfies it. Let $\mu' = \theta'\theta$. We have $\mu =_E \mu'\bar{\phi}$ where μ' satisfies the name restriction of $\bar{\phi}$. Thus, we have $s\bar{\phi} =_E l\mu'\bar{\phi} \to_R r\mu'\bar{\phi}$, where $r\mu'$ satisfies the name restriction of $\bar{\phi}$ thanks to the form of rules in R. Moreover, for any frame $\bar{\psi}$ with the same name restriction as the one of $\bar{\phi}$, the same rewrite step applies and we get $l\mu'\bar{\psi} \to_R r\mu'\bar{\psi}$, where $r\mu'$ satisfies the name restriction. If $\bar{\phi} \approx_E \bar{\psi}$, then $s\bar{\phi} =_E l\mu'\bar{\phi}$ implies $s\bar{\psi} =_E l\mu'\bar{\psi}$. Consequently, $s\bar{\psi} =_E l\mu'\bar{\psi} \to_R r\mu'\bar{\psi}$.
— Otherwise, if $|s| > |R|$ and $s\bar{\phi} =_E l\mu$, then there would be a contradiction with the fact that $CMT_E(l)$ is a complete set of terms E-matched by l. Thus, we have necessarily $|s| \leq |R|$, and only two cases are possible for the rewrite rule $l \to r$ since R is subterm E-convergent:
– If r is a ground term, then $s\bar{\phi} \to_{R,E} r = r\bar{\phi}$ (where r satisfies the name restriction of $\bar{\phi}$). By definition of Eq^B and by assumption on $\bar{\psi}$, we have $s\bar{\psi} =_{RE} r = r\bar{\psi}$.
– If r is a subterm of l, then by definition of $\bar{\phi}$, there exists some variable x such that $s\bar{\phi} \to_{R,E} x\bar{\phi}$. By definition of Eq^B and by assumption on $\bar{\psi}$, we have $s\bar{\psi} =_{RE} x\bar{\psi}$.

(ii) Let us now assume that the rewrite step occurs below the root position. There exists a position $p \neq \epsilon$ such that $s'\bar{\phi} = (s'\bar{\phi})[s\bar{\phi}]_p$ with $s\bar{\phi} \to^{\epsilon}_{R,E} t$. By the case (*i*) above, there exists a term u such that $t =_E u\bar{\phi}$ and $s\bar{\psi} =_{RE} u\bar{\psi}$. Then, we have

$$s'\bar{\phi} \to_{R,E} (s'\bar{\phi})[t]_p =_E (s'\bar{\phi})[u\bar{\phi}]_p = (s'[u]_p)\bar{\phi}$$

and

$$s'\bar{\psi} = (s'\bar{\psi})[s\bar{\psi}]_p =_{RE} (s'\bar{\psi})[u\bar{\psi}]_p = (s'[u]_p)\bar{\psi}$$ □

We are now ready to prove the two reduction lemmas, namely Lemmas 12 and 14. For both lemmas, we prove the non-obvious direction:

— **Lemma 12.**

Proof. (Only if direction) Assume $s\bar{\phi} =_{RE} t$ where t is normalized. According to Lemma 15, there exists a rewrite proof of the form

$$s\bar{\phi} \to_{R,E} \circ =_E \cdots \to_{R,E} \circ =_E s'\bar{\phi} =_E t$$

where s' satisfies the name restriction of $\bar{\phi}$. By choosing $\psi = \phi$, we have $\bar{\phi} =_E \phi_*$ and so

$$s\phi_* =_E s\bar{\phi} \to_{R,E} \circ =_E \cdots \to_{R,E} \circ =_E s'\bar{\phi} =_E s'\phi_* =_E t$$ □

— **Lemma 14.**

Proof. (If direction) Assume $s\bar{\phi} =_{RE} t\bar{\phi}$. According to Lemma 15, there exists a rewrite proof of the form

$$s\bar{\phi} \to_{R,E} \circ =_E \cdots \to_{R,E} \circ =_E s'\bar{\phi}$$

$$t\bar{\phi} \to_{R,E} \circ =_E \cdots \to_{R,E} \circ =_E t'\bar{\phi}$$

where $s'\bar{\phi} =_E t'\bar{\phi}$ and s', t' satisfy the name restriction of $\bar{\phi}$.
By Lemma 15, we have $s\bar{\psi} =_{RE} s'\bar{\psi}$ and $t\bar{\psi} =_{RE} t'\bar{\psi}$. By assumption, we have $\bar{\phi} \approx_E \bar{\psi}$, and so $s'\bar{\phi} =_E t'\bar{\phi}$ implies $s'\bar{\psi} =_E t'\bar{\psi}$. Consequently, $s\bar{\psi} =_{RE} t\bar{\psi}$. □

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

7. Conclusion

We have shown how to lift the existing decidability results on knowledge in subterm convergent rewrite systems to rewrite systems defined modulo certain equational theories E. In particular, we have lifted the decidability results to the case in which E is shallow permutative, for example, the Commutativity theory C. Furthermore, we have studied a notion of complete set of E-matched terms, *CMT* for short, such that if a subterm convergent equational rewrite system R admits a finite *CMT* for each of its left-hand sides, then the size of R modulo E can be defined. In addition, the knowledge problems in $R \cup E$ are decidable if they are decidable modulo E and the size of R modulo E is computable. We have also developed a procedure (MTG) such that if the procedure terminates, a finite *CMT* is guaranteed. The MTG procedure is terminating for a large class of syntactic permutative theories, namely permutative theories closed by paramodulation. Even if the MTG procedure is not terminating, it is possible to have a finite *CMT*. For example, in general, the *AC* case does not produce a terminating MTG procedure. However, a particular subterm *AC*-convergent rewrite system can admit a computable size of R modulo *AC* and consequently the knowledge problems are decidable in $R \cup AC$.

The next step would be to explore a relaxing of the notion of subterm convergent while maintaining the decidability results for the knowledge problems. This would be useful since many axiomatizations of protocols are close but not completely subterm convergent. For example, consider the following set of axioms:[1]

$$\begin{aligned} d(e(x, y), y) &\to x \\ d(e(x, y\&z), y) &\to e(x, z) \\ d(e(x, y), y\&z) &\to d(x, z) \\ d(e(x, y\&z), y\&v) &\to d(e(x, z), v) \end{aligned}$$

Notice that this theory is not completely subterm convergent. However, it is close in that all the right-hand sides are either subterms or homeomorphic embeddings of the left-hand sides. If the notion of subterm could be extended to such cases, then it may be possible to solve the knowledge problem in the $C(\&)$-convergent form of this theory.

More generally, a natural continuation of this work is to study the use of equational (E-convergent) rewrite systems for extending the complete but non-necessarily terminating procedures that have been designed for the knowledge problems in standard (convergent) rewrite systems (Baudet *et al.*, 2013; Ciobâcă *et al.*, 2012). Due to the interest of *AC* in protocol verification, it would be useful to develop such an engine with the capability to handle *AC*-convergent rewrite systems.

Another challenge is to study the knowledge problems in combinations of the form $R \cup E \cup T$ where R is an E-convergent TRS and T is an arbitrary theory sharing with R only the function symbols of E, like for instance R is $AC(+)$-convergent and $T = \{h(x + y) = h(x) + h(y)\}$. In this direction, it would be interesting to extend our combination results (Erbatur *et al.*, 2017) on theories sharing absolutely free constructors to the case of theories sharing constructors modulo a theory E such as *AC*.

Acknowledgements. We are very grateful to the reviewers: the paper has been significantly improved thanks to their suggestions. We would like to thank Véronique Cortier and Steve Kremer for fruitful comments and discussions. This work has received funding from the European Research Council (ERC) under the H2020 research and innovation program (grant agreement no. 645865-SPOOC).

Note

1 These axioms are a fragment of a larger theory studied in Yang *et al.* (2014), modeling encryption and decryption in a multiset of keys.

https://doi.org/10.1017/S0960129520000031 Published online by Cambridge University Press

References

Abadi, M. and Cortier, V. (2004). *Deciding knowledge in security protocols under equational theories*. Research Report RR-5169, INRIA.

Abadi, M. and Cortier, V. (2006). Deciding knowledge in security protocols under equational theories. *Theoretical Computer Science* **367** (1–2) 2–32.

Abadi, M. and Fournet, C. (2001). Mobile values, new names, and secure communication. In: *Proceedings of the 28th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages* (POPL 2001), ACM, 104–115.

Armando, A., Basin, D. A., Boichut, Y., Chevalier, Y., Compagna, L., Cuéllar, J., Drielsma, P. H., Héam, P., Kouchnarenko, O., Mantovani, J., Mödersheim, S., von Oheimb, D., Rusinowitch, M., Santiago, J., Turuani, M., Viganò, L. and Vigneron, L. (2005). The AVISPA tool for the automated validation of internet security protocols and applications. In: Etessami, K. and Rajamani, S. K. (eds.) *17th International Conference on Computer Aided Verification, CAV 2005, Edinburgh, Scotland, UK, Proceedings*, Lecture Notes in Computer Science, vol. 3576, Springer, 281–285.

Amadio, R. and Lugiez, D. (2000). On the reachability problem in cryptographic protocols. In: *Proceedings of CONCUR 2000 — Concurrency Theory*, Springer, 380–394.

Ayala-Rincón, M., Fernández, M. and Nantes-Sobrinho, D. (2017). Intruder deduction problem for locally stable theories with normal forms and inverses. *Theoretical Computer Science* **672** 64–100.

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

Baudet, M., Cortier, V. and Delaune, S. (2013). YAPA: A generic tool for computing intruder knowledge. *ACM Transactions on Computational Logic* **14** (1) 4.

Blanchet, B. (2016). Modeling and verifying security protocols with the applied Pi calculus and ProVerif. *Foundations and Trends in Privacy and Security* **1** (1–2) 1–135.

Blanchet, B. (2001). An efficient cryptographic protocol verifier based on prolog rules. In :*14th IEEE Computer Security Foundations Workshop (CSFW-14 2001), 11–13 June 2001, Cape Breton, Nova Scotia, Canada*. IEEE Computer Society, 82–96.

Chadha, R., Cheval, V., Ciobâcă, S. and Kremer, S. (2016). Automated verification of equivalence properties of cryptographic protocols. *ACM Transactions on Computational Logic* **17** (4) 23:1–23:32.

Cheval, V., Cortier, V. and Turuani, M. (2018). A little more conversation, a little less action, a lot more satisfaction: global states in ProVerif. In: *31st IEEE Computer Security Foundations Symposium, CSF 2018, Oxford, United Kingdom*, IEEE Computer Society, 344–358.

Cheval, V., Kremer, S. and Rakotonirina, I. (2018). The DEEPSEC prover. In: Chockler, H. and Weissenbacher, G. (eds.) *Computer Aided Verification – 30th International Conference, CAV 2018, Held as Part of the Federated Logic Conference, FloC 2018, Oxford*, Lecture Notes in Computer Science, vol. 10982, Springer, 28–36.

Chevalier, Y., Küsters, R., Rusinowitch, M. and Turuani, M. (2003). Deciding the security of protocols with Diffie-Hellman exponentiation and products in exponents. In Pandya, P. K. and Radhakrishnan, J. (eds.) *FSTTCS 2003: Foundations of Software Technology and Theoretical Computer Science, 23rd Conference, Mumbai, India*, Lecture Notes in Computer Science, vol. 2914, Springer, 124–135.

Chevalier, Y., Küsters, R., Rusinowitch, M. and Turuani, M. (2005). An NP decision procedure for protocol insecurity with XOR. *Theoretical Computer Science* **338** 247–274.

Chevalier, Y. and Rusinowitch, M. (2008). Hierarchical combination of intruder theories. *Information and Computation* **206** (2–4) 352–377.

Ciobâcă, S., Delaune, S. and Kremer, S. (2012). Computing knowledge in security protocols under convergent equational theories. *Journal of Automated Reasoning* **48** (2) 219–262.

Cohn-Gordon, K., Cremers, C., Garratt, L., Millican, J. and Milner, K. (2018). On ends-to-ends encryption: Asynchronous group messaging with strong security guarantees. In: *Conference on Computer and Communications Security, CCS 2018, Toronto, Canada*, ACM, 1802–1819.

Comon-Lundh, H. and Delaune, S. (2005). The finite variant property: How to get rid of some algebraic properties. In: Giesl, J. (ed.) *Rewriting Techniques and Applications*, Lecture Notes in Computer Science, vol. 3467, Springer, 294–307.

Comon, H., Haberstrau, M. and Jouannaud, J.-P. (1994). Syntacticness, cycle-syntacticness, and shallow theories. *Information and Computation* **111**(1) 154–191.

Comon-Lundh, H. and Shmatikov, V. (2003). Intruder deductions, constraint solving and insecurity decision in presence of exclusive or. In: *18th Annual IEEE Symposium on Logic in Computer Science* (LICS 2003), 271–280.

Conchinha, B., Basin, D. A. and Caleiro, C. (2011). FAST: An efficient decision procedure for deduction and static equivalence. In: Schmidt-Schauß, M. (ed.) *Proceedings of RTA 2011, Novi Sad, Serbia*, LIPIcs, vol. 10, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 11–20.

Cortier, V. and Delaune, S. (2010). Decidability and combination results for two notions of knowledge in security protocols. *Journal of Automated Reasoning* **48**(4) 441–487.

Cremers, C. J. F. (2008). The Scyther tool: Verification, falsification, and analysis of security protocols. In: Gupta, A. and Malik, S. (eds.) *20th International Conference on Computer Aided Verification (CAV 2008), Princeton, NJ, USA, July 7–14, 2008, Proceedings*, Lecture Notes in Computer Science, vol. 5123, Springer, 414–418.

Dolev, D. and Yao, A. C. (1981). On the security of public key protocols (extended abstract). In: *22nd Annual Symposium on Foundations of Computer Science, Nashville, Tennessee, USA*, IEEE Computer Society, 350–357.

Dreier, J., Hirschi, L., Radomirovic, S. and Sasse, R. (2018). Automated unbounded verification of stateful cryptographic protocols with exclusive or. In: *31st IEEE Computer Security Foundations Symposium, CSF 2018, Oxford, United Kingdom*, IEEE Computer Society, 359–373.

Erbatur, S., Marshall, A. M. and Ringeissen, C. (2017). Notions of knowledge in combinations of theories sharing constructors. In: de Moura, L. (ed.) *Automated Deduction – CADE-26, 26th International Conference on Automated Deduction, Gothenburg, Sweden, Proceedings*, Lecture Notes in Computer Science, vol. 10395, Springer, 60–76.

Erbatur, S., Marshall, A. M. and Ringeissen, C. (2018). Knowledge problems in equational extensions of subterm convergent theories. In: *32nd International Workshop on Unification (UNIF-2018), Oxford, UK*.

Escobar, S., Meadows, C. A. and Meseguer, J. (2007). Maude-NPA: Cryptographic protocol analysis modulo equational properties. In: *Foundations of Security Analysis and Design V, FOSAD 2007/2008/2009 Tutorial Lectures*, Lecture Notes in Computer Science, vol. 5705, Springer, 1–50.

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

Hullot, J. (1980). Canonical forms and unification. In: Bibel, W. and Kowalski, R. A. (eds.) *5th Conference on Automated Deduction, Les Arcs, France, July 8–11, 1980, Proceedings*, Lecture Notes in Computer Science, vol. 87, Springer, 318–334.

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

Kirchner, C. and Klay, F. (1990). Syntactic theories and unification. In: *Fifth Annual IEEE Symposium on Logic in Computer Science* (LICS 1990), 270–277.

Lynch, C. and Morawska, B. (2002). Basic syntactic mutation. In: Voronkov, A. (ed.) *Automated Deduction – CADE-18, 18th International Conference on Automated Deduction, Proceedings*, Lecture Notes in Computer Science, vol. 2392, Springer, 471–485.

Meseguer, J. (2018). Variant-based satisfiability in initial algebras. *Science of Computer Programming* **154** 3–41.

Millen, J. and Shmatikov, V. (2001). Constraint solving for bounded-process cryptographic protocol analysis. In: *Proceedings of the 8th ACM Conference on Computer and Communications Security* (CCS 2001), ACM, 166–175.

Mödersheim, S. and Viganò, L. (2009). The open-source fixed-point model checker for symbolic analysis of security protocols. In: *Foundations of Security Analysis and Design V, FOSAD 2007/2008/2009 Tutorial Lectures*, Lecture Notes in Computer Science, vol. 5705, Springer, 166–194.

Nguyen, K. (2019). Formal verification of a messaging protocol. Work done under the supervision of Vincent Cheval and Véronique Cortier.

Nipkow, T. (1990). Proof transformations for equational theories. In: *Fifth Annual IEEE Symposium on Logic in Computer Science* (LICS 1990), 278–288.

Paulson, L. C. (1998). The inductive approach to verifying cryptographic protocols. *Computer Security* **6**(1–2) 85–128.

Ringeissen, C. (2019). Building and combining matching algorithms. In: Lutz, C., Sattler, U., Tinelli, C., Turhan, A. Y. and Wolter, F. (eds.) *Description Logic, Theory Combination, and All That – Essays Dedicated to Franz Baader on the Occasion of His 60th Birthday*, Lecture Notes in Computer Science, vol. 11560, Springer, 523–541.

Sasse, R., Escobar, S., Meadows, C., Meseguer, J. (2011). Protocol analysis modulo combination of theories: A case study in Maude-NPA. In: *Proceedings of International Workshop on Security and Trust Management*, Springer, 163–178.

Schmidt, B., Meier, S., Cremers, C. J. F. and Basin, D. A. (2012). Automated analysis of Diffie-Hellman protocols and advanced security properties. In: Chong, S. (ed.) *25th IEEE Computer Security Foundations Symposium, CSF 2012, Cambridge, MA, USA, June 25-27, 2012*, IEEE Computer Society, 78–94.

Schmidt-Schauß, M. (1989). Unification in permutative equational theories is undecidable. *Journal of Symbolic Computation* **8** (4) 415–421.

Turuani, M. (2006). The CL-Atse protocol analyser. In: Pfenning, F. (ed.) *Term Rewriting and Applications, 17th International Conference, RTA 2006, Seattle, WA, USA, August 12–14, 2006, Proceedings*, Lecture Notes in Computer Science, vol. 4098, Springer, 277–286.

Yang, F., Escobar, S., Meadows, C., Meseguer, J. and Narendran, P. (2014). Theories of homomorphic encryption, unification, and the finite variant property. In: *Proceedings of the 16th International Symposium on Principles and Practice of Declarative Programming* (PPDP 2014), ACM, 123–133.

Cite this article: Erbatur S, Marshall AM and Ringeissen C (2020). Computing knowledge in equational extensions of subterm convergent theories. *Mathematical Structures in Computer Science* **30**, 683–709. https://doi.org/10.1017/S0960129520000031