Notions of Knowledge in Combinations of Theories Sharing Constructors

Serdar Erbatur, Andrew Marshall, Christophe Ringeissen

Notions of Knowledge in Combinations of Theories Sharing Constructors

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

[1] Ludwig-Maximilians-Universität, München (Germany)
[2] University of Mary Washington (USA)
[3] LORIA – INRIA Nancy-Grand Est (France)

Abstract. One of the most effective methods developed for the analysis of security protocols is an approach based on equational reasoning and unification. In this approach, it is important to have the capability to reason about the knowledge of an intruder. Two important measures of this knowledge, defined modulo equational theories, are deducibility and static equivalence. We present new combination techniques for the study of deducibility and static equivalence in unions of equational theories sharing constructors. Thanks to these techniques, we obtain new modularity results for the decidability of deducibility and static equivalence. In turn, this should allow for the analysis of protocols involving combined equational theories which previous disjoint combination methods could not address due to their non-disjoint axiomatization.

1 Introduction

The formal analysis of security protocols is a large area of research, with one of its primary starting points the paradigm developed by Dolev and Yao [16] in which equational theories play a central role. This field of research has resulted in the development of several automated tools for the analysis of security issues in protocols, including [3, 8, 14, 19, 21, 23, 26]. Unification procedures and their combinations are widely used in such tools, e.g., a disjoint combination procedure [5, 24] is the basic engine of Cl-AtSe [26]. This disjoint combination procedure has been extended to solve satisfiability problems in non-disjoint hierarchical intruder theories [10]. Verifying the security of protocols requires the development of specific decision procedures to reason about the knowledge that an attacker may have. Two important measures of this knowledge, which are useful and widely used, are *deducibility* [20, 22] and *static equivalence* [1]. Informally, deducibility is the question of whether an attacker, given their deductive capabilities and a set of messages representing their knowledge, can compute another message representing 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

* This work has received funding from the European Research Council (ERC) under the H2020 research and innovation program (grant agreement No 645865-SPOOC).

which we need to be able to decide more than deducibility. For some protocols, in addition to deducibility, we would like to know if an attacker 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. One measure of this is static equivalence, which is a critical measure of the capability of the protocol to maintain indistinguishability between different runs.

Much work has gone into investigating and developing decision procedures for the questions of deducibility and static equivalence [1, 7, 12, 15]. The equational theories of interest are usually defined as unions of several simpler sub-theories. In these cases, it is quite natural to try to proceed in a modular way by combining the decision procedures already available for the sub-theories. This combination problem has been investigated in the analysis of sequent calculi [25] for deducibility, and saturation-based decision procedures [13] for both deducibility and static equivalence. However, these contributions [13, 25] are restricted to the disjoint case, where sub-theories are signature-disjoint. Until now, the non-disjoint case remained unexplored. One difficulty in this study is that the sub-theories often share some axioms. For example, encryption and decryption axioms are often found in such equational theories. The approach of just removing the axioms from one theory can often lead to a dead end: it may no longer be possible to reuse any existing decision procedure for the theory if an axiom is removed. Furthermore, along with these shared axioms are often found function symbols, such as pairing, which are also shared between the combined theories. It is possible that the shared function symbol appear in some shared axioms and in some non-shared. Thus, the non-disjoint case offers more complexities.

The approach developed in [13] to solve the disjoint case for deducibility and static equivalence is based on locality principles, to restrict the application of saturation-based decision procedures to the finitely many terms occurring in the problem. Instead, we follow an approach based on the tuning of some combination techniques which are instrumental to prove the combination procedures for deducibility and static equivalence. From our point of view, this combination approach leads to simpler and shorter proofs.

Along the lines of previous works on non-disjoint combination [6, 17, 18], we focus on equational theories sharing *constructors*. An originality of our approach is the ability to consider both shared constructors and shared axioms as those defining the access to the components of a constructor. In the first portion of the paper, we clearly define the class of combined theories we consider. A combined term rewrite system is used to identify the constructors. This term rewrite system is useful to state results showing that some decision procedures known for sub-theories can be reused without loss of completeness for the combined theory. In particular, we are interested in solving some restricted context unification problems related to deducibility and static equivalence. The proposed combination procedures purify the problems by replacing, as usual, alien subterms with fresh names. This reduction by purification is correct if the problems are first transformed in an appropriate way: the knowledge specified by the prob-

lems must be completed before purification. These transformations are borrowed from the ones initiated in [13] for the disjoint case.

Outline. Section 2 presents the background information for this paper. Section 3 develops the new combination results for non-disjoint equational theories sharing constructors. In Section 4 we apply the results from Section 3 to the two knowledge questions from security protocols, deducibility in Section 4.1, and static equivalence in Section 4.2.

2 Preliminaries

We use the standard notation of equational logic and term rewriting [4]. As in [1] we use some concepts, such as names, borrowed from the applied pi calculus [2].

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. Given any $\Sigma' \subseteq \Sigma$, A term t is said to be Σ'-*rooted* if $t(\epsilon) \in \Sigma'$. A *context*, C, is a first-order term with "holes", or distinguished variable that occur only once. We may write $C\,[x_1,\ \ldots,\ x_n]$, to illustrate that the context C contains n distinguished variables.

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.

A substitution σ is an endomorphism of $T(\Sigma \cup N, X)$ with only finitely many variables not mapped to themselves. Application of a substitution σ to a term t (resp. a substitution θ) is written $t\sigma$ (resp. $\theta\sigma$). The domain of σ is $Dom(\sigma) = \{x \in X \mid x\sigma \neq x\}$. The range of σ is $Ran(\sigma) = \{x\sigma \mid x \in Dom(\sigma)\}$. Given a substitution $\sigma = \{x_1 \mapsto t_1, \ldots, x_m \mapsto t_m\}$, we have $Dom(\sigma) = \{x_1, \ldots, x_m\}$ and $Ran(\sigma) = \{t_1, \ldots, t_m\}$. When θ and σ are two substitutions with disjoint domains and with only ground terms in their ranges, $\theta\sigma = \theta \cup \sigma$.

A *term rewrite system* (TRS) is a pair (Σ, R), where Σ is a signature and R is a 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 $\rightarrow_R$. A TRS R is *confluent* if, whenever $t \rightarrow_R^* s_1$ and $t \rightarrow_R^* s_2$, there exists a term w such that $s_1 \rightarrow_R^* w$ and $s_2 \rightarrow_R^* w$. 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 is *finite* if its set of rules is finite. From now on, a finite TRS is denoted by a calligraphic letter, say $\mathcal{R}$. Given a finite TRS $(\Sigma, \mathcal{R})$, $\mathtt{D}(\mathcal{R}) = \{l(\epsilon) \mid l \rightarrow r \in \mathcal{R}\}$ and $\mathtt{C}(\mathcal{R}) = \Sigma \backslash \mathtt{D}(\mathcal{R})$. A finite convergent TRS $\mathcal{R}$ is said to be *subterm convergent* if for any $l \rightarrow r \in \mathcal{R}$, r is either a strict subterm of l or a constant. An equational theory is *subterm convergent* if it is presented by a subterm convergent TRS. Both deducibility and static equivalence are known to be decidable in subterm convergent theories [1].

3 Combination of Theories

In this section we begin with an example from security protocol analysis to help elucidate the new presentation of non-disjoint combination below.

Example 1. Consider the following equational theories:

$$T_1 = \left\{ \begin{array}{l} enc(\langle x, y\rangle, z) = \langle enc(x, z), enc(y, z)\rangle \\ dec(\langle x, y\rangle, z) = \langle dec(x, z), dec(y, z)\rangle,\ dec(enc(x, y), y) = x \end{array} \right\}$$

$T_2 = \{h(\langle x, y\rangle, z) = \langle h(x, z), h(y, z)\rangle\}$, $T_3 = \{fst(\langle x, y\rangle) = x, snd(\langle x, y\rangle) = y\}$. The theories $E_1 = T_1 \cup T_3$ and $E_2 = T_2 \cup T_3$ are two theories of homomorphism studied respectively in [1] and in [11].

In the above example, if one wishes to ask questions about the combined theory, $E = E_1 \cup E_2$, then there are several problems.

First, there is the shared symbol, $\langle\rangle$, which, if the equalities are oriented from left to right, is a shared constructor. In addition, there are two particular shared destructors, again via a left-to-right orientation, $fst(\langle x, y\rangle) = x$, and $snd(\langle x, y\rangle) = y$. This problem, having one or more axioms which are exactly the same but in two different presentations of two different equational theories, is common in theories arising from security protocols. For example, the axioms in T_3 are common, just like $dec(enc(x, y), y) = x$. To proceed by combination techniques we could try to consider E as the union of three theories T_1, T_2 and T_3. However, this union would still share the symbol $\langle\rangle$ and thus we couldn't rely on current combination methods. Furthermore, there is an additional problem of the availability of decision procedures for these *three* theories. Often, we are trying to combine two or more theories for which we have decision procedures available to obtain a decision procedure for the combined theory, E in our example. If we remove equalities from a presentation, we are not guaranteed to still have a decision procedure available. For example, deducibility has been studied for E_1 and E_2, but has not been studied for $E_1 \setminus T_3$. Therefore, we consider a new method of non-disjoint combination which allows us to combine E_1 and E_2 and maintain decidability of such questions as deducibility and static equivalence.

Before continuing to the details let us briefly outline some of the key topics needed to achieve the combination results. In the following, the combined theory E is handled thanks to a combined convergent TRS R sharing constructors (cf. Definition 1). The purification of ground terms is processed by constant abstraction, which is formally defined via a bijection between R-normal forms and fresh names (cf. Definition 2). Fortunately, we can use layer-reduced forms as a computable alternative to R-normal forms (cf. Definition 3). The knowledge problems we focus on are expressed using the notion of frame defined as a ground substitution together with a set of restricted names. A completion mechanism is required to achieve all the knowledge encoded by a frame (cf. Definition 5). As shown in Section 4, our combination methods are based on the constant abstraction of completed frames.

3.1 Constructor-Sharing Theories

Let us formally describe the combined theories we are interested in. We focus on combinations of theories $T_1 \cup \cdots \cup T_n$ for which shared function symbols can be interpreted as *constructors*. To formalize the notion of constructor, it is convenient to rely on a term rewrite system. However, not every equational theory can be equivalently presented by a term rewrite system. Fortunately, it is always possible to rely on a ground term rewrite system that could be obtained by unfailing completion [5]. More directly, this term rewrite system and the related constructors are defined below with respect to a reduction ordering used to orient heterogeneous ground instances of T_i-equalities.

Definition 1. *Let T_i be an equational Ω_i-theory for $i = 1, \ldots, n$. Consider the signature $\Sigma = \Omega_1 \cup \cdots \cup \Omega_n$ and the equational Σ-theory $E = T_1 \cup \cdots \cup T_n$. Let $>$ be a Noetherian reduction ordering on $T(\Sigma \cup V)$ (i.e., stable by context) such that V denotes a (sufficiently large) finite set of free constants (including names) which are minimal w.r.t $>$. Consider a (possibly infinite) set of Ω_i-equalities A_i such that:*

- *For any $l = r \in A_i$ such that $l(\epsilon), r(\epsilon) \in \Omega_i$, and any substitution ψ such that $Ran(\psi) \subseteq T(\Sigma \cup V)$, we have $l\psi > r\psi$ or $r\psi > l\psi$.*
- *For any $l = x \in A_i$ such that $l(\epsilon) \in \Omega_i$, x is a variable, and any substitution ψ such that $Ran(\psi) \subseteq T(\Sigma \cup V)$, we have $l\psi > x\psi$.*
- *the TRS $R_i = \{l\psi \to r\psi \mid l = r \in A_i, l\psi > r\psi, Ran(\psi) \subseteq T(\Sigma \cup V)\}$ is convergent on $T(\Sigma \cup V)$ and $=_{R_i}$ is $=_{T_i}$ on $T(\Sigma \cup V)$.*

A function symbol $f \in \Sigma$ is a constructor *of R_i if for any terms $t_1, \ldots, t_m$ in $T(\Sigma \cup V)$, $f(t_1, \ldots, t_m) \downarrow_{R_i} = f(t_1 \downarrow_{R_i}, \ldots, t_m \downarrow_{R_i})$. E is said to be* constructor-sharing (w.r.t $>$) *if for any $i, j \in \{1, \ldots n\}$, $i \neq j$, function symbols in $\Omega_i \cap \Omega_j$ are constructors of both R_i and R_j. In that case, $R = R_1 \cup \cdots \cup R_n$ is the* combined TRS *of E and $SC = \bigcup_{i \neq j} \Omega_i \cap \Omega_j$ is the set of* shared constructors *of E.*

There are several ways to consider appropriate A_i's following Definition 1. In general, A_i can be chosen as the set of all Ω_i-equalities $l = r$ such that $l =_{T_i} r$.

In the following example, we detail the prominent case of theories presented by finite convergent term rewrite systems.

Example 2. In Definition 1, consider T_i is presented by a finite convergent TRS $\mathcal{R}_i$ for $i = 1, \ldots, n$, such that $\mathcal{R}_1 \cup \cdots \cup \mathcal{R}_n$ is terminating for a reduction ordering on (the set of terms with variables) $T(\Sigma, X)$. Then, rules in $\mathcal{R}_i$ can be used to build A_i and R_i-normal forms are computable by $\mathcal{R}_i$-normalization. Assume that for any $i, j \in \{1, \ldots n\}$, $i \neq j$, we have $\Omega_i \cap \Omega_j \subseteq \mathtt{C}(\mathcal{R}_i) \cap \mathtt{C}(\mathcal{R}_j)$. Let $\mathcal{R} = \mathcal{R}_1 \cup \cdots \cup \mathcal{R}_n$. Then, the equational theory of $\mathcal{R}$ is constructor-sharing, where normal forms are computable by $\mathcal{R}$-normalization.

Proposition 1. *If E is a constructor-sharing Σ-theory, then its combined TRS R is a convergent TRS such that $=_R$ is $=_E$ on $T(\Sigma \cup V)$.*

In Definition 1, note that the constructors of any R_i remain constructors of R. If a term is R-reducible, then it is R-reducible by a rule whose left hand-side is $(\Sigma \backslash SC)$-rooted.

Assumption 1 *Consider a constructor-sharing theory $E = T_1 \cup \cdots \cup T_n$ as in Definition 1, its combined TRS R, and its set of shared constructors SC. We assume E is split into two non-disjoint theories E_1 and E_2, defined as follows. Let $K_1, K_2 \subset \{1, \ldots, n\}$ such that $K_1 \cup K_2 = \{1, \ldots, n\}$. For $i = 1, 2$, consider the signature $\Sigma_i = \bigcup_{k \in K_i} \Omega_k$ and the Σ_i-theory $E_i = \bigcup_{k \in K_i} T_k$. So, $\Sigma = \Sigma_1 \cup \Sigma_2$, $E = E_1 \cup E_2$ and both E_1, E_2 include the equational theory $\bigcup_{k \in K_1 \cap K_2} T_k$. From now on, the R-normal form of any t is simply denoted by $t\!\downarrow$.*

We illustrate the above notion of constructor-sharing theories with several examples. These examples originate from theories studied in the security protocol analysis literature.

Example 3. (Example 1 continued). Assumption 1 holds with $A_i = T_i$ for $i = 1, 2, 3$. Indeed, the left-to-right orientation of equalities in A_i leads to the TRS $\mathcal{R}_i$ for $i = 1, 2, 3$. The TRS $\mathcal{R} = \bigcup_{i=1}^{3} \mathcal{R}_i$ is convergent, where $\langle\rangle$ is a constructor of each TRS $\mathcal{R}_i$ for $i = 1, 2, 3$. Thus, E is a constructor-sharing theory.

Example 4. Consider the following equational theories:

$$T_1 = \left\{ \begin{array}{l} fst(\langle x, y\rangle) = x,\ snd(\langle x, y\rangle) = y \\ adec(aenc(x, pk(y), z), y) = x,\ dec(enc(x, y), y) = x \\ check_1(sign(x, y), pk(y)) = ok,\ msg(sign(x, y)) = x \end{array} \right\}$$

$$T_2 = \left\{ \begin{array}{l} open(commit(x, y), y) = x,\ getpk(host(x)) = x \\ unblind(blind(x, y), y) = x \\ unblind(sign(blind(x, y), z), y) = sign(x, z) \\ check_2(sign(x, y), pk(y)) = x \end{array} \right\}$$

The theories T_1 and T_2 are used for modeling respectively strong secrecy [9] and blind signatures in e-voting protocols [1]. Let $E_1 = T_1, E_2 = T_2$ and $E = E_1 \cup E_2$. For the same reasons as in Example 1, E is a constructor-sharing theory, where

sign and *pk* are the shared constructors. Alternatively, it is also possible to remove the axioms $check_i(sign(x,y),pk(y)) = \ldots$ from T_i $(i = 1,2)$ and to consider a third theory, say $\{check(sign(x,y),pk(y)) = x\}$, that would be shared as the theory T_3 in Example 1.

3.2 Equational Proofs in Combined Theories

A modular approach is possible due to the close relationship between combined equational proofs (modulo E) and pure ones (modulo E_1 and E_2). To state this relationship, we use a well-known notion, called *abstraction* [5]. In our context, impure terms are abstracted by free constants, via a bijection denoted by π.

Definition 2 (Constant Abstraction). *Let $\mathcal{C}$ be a set of (free) constants such that V and $\mathcal{C}$ are disjoint. Let $\pi : \{t\downarrow \mid t \in T(\Sigma \cup V), t\downarrow \notin V\} \longrightarrow \mathcal{C}$ be a bijection called a* constant abstraction with range $\mathcal{C}$. *For $i = 1,2$, the* i-abstraction *of t is denoted by t^{π_i} and defined as follows:*

- *If $t \in V$, then $t^{\pi_i} = t$.*
- *If $t = f(t_1, \ldots, t_n)$ and $f \in \Sigma_i$, then $t^{\pi_i} = f(t_1^{\pi_i}, \ldots, t_n^{\pi_i})$.*
- *Otherwise (t is $\Sigma\backslash\Sigma_i$-rooted), $t^{\pi_i} = \pi(t\downarrow)$ if $t\downarrow \notin V$, else $t^{\pi_i} = t\downarrow$.*

An inverse mapping of π *is a mapping $\pi^{-1} : \mathcal{C} \longrightarrow (T(\Sigma \cup V)\backslash V)$ such that $\pi(\pi^{-1}(c)\downarrow) = c$ for any $c \in \mathcal{C}$.*

Given a signature Ω, $Alien_\Omega(t)$ denotes the set of maximal subterms of t rooted by a function symbol in $\Sigma\backslash\Omega$. $Alien_{\Sigma_i}(t)$ is abbreviated into $Alien_i(t)$. The terms in $Alien_i(t)$ are called the *i-alien subterms* of t. Given a substitution σ, $Alien_i(\sigma) = \bigcup_{x \in Dom(\sigma)} Alien_i(x\sigma)$. The set of *alien subterms* of t is $Alien(t) = Alien_1(t) \cup Alien_2(t)\backslash\{t\}$.

Lemma 1. *Let t be an arbitrary term such that its i-alien subterms are R-normalized. If t is R-reducible, then there exists a term t' such that $t \to_R t'$ and $(t)^{\pi_i} =_{E_i} (t')^{\pi_i}$ where the i-alien subterms of t' are R-normalized.*

Proof. Assume t is a term such that terms in $Alien_i(t)$ are R-normalized for some $i \in \{1,2\}$.

If t is variable, then t is R-irreducible. If t is $\Sigma\backslash\Sigma_i$-rooted, then $Alien_i(t) = \{t\}$, and so t is R-irreducible by assumption.

Let us now assume t is Σ_i-rooted. Then, the redex position, p, in t occurs necessarily above the i-alien subterms. Hence, without loss of generality, there is some $l\psi \to r\psi \in R$ such that $t|_p = l\psi$, $t' = t[r\psi]_p$, $Alien_i(t') \subseteq Alien_i(t)$ and $l =_{E_i} r$ where l, r are i-pure terms. On the one hand, we have $(t^{\pi_i})|_p = (t|_p)^{\pi_i} = l\psi^{\pi_i}$. On the other hand, $(t')^{\pi_i} = (t[r\psi]_p)^{\pi_i} = t^{\pi_i}[r\psi^{\pi_i}]_p$. Since $l =_{E_i} r$, we have $l\psi^{\pi_i} =_{E_i} r\psi^{\pi_i}$. Therefore, $t^{\pi_i} =_{E_i} t^{\pi_i}[r\psi^{\pi_i}]_p = (t')^{\pi_i}$. □

Lemma 1 can be applied inductively to obtain the following result.

Lemma 2. *Let t be a term such that its i-alien subterms are normalized. Then $t^{\pi_i} =_{E_i} (t\downarrow)^{\pi_i}$.*

The notion of *layer-reduced form* [17] aims at providing a computable term with the same "theory shape" as the R-normal form.

Definition 3. *A term t is in* layer-reduced form *if*

- $t \in V$, *or*
- $t = f(t_1, \ldots, t_n)$, $f \in SC$ *and* $t_1, \ldots, t_n$ *are in layer-reduced form, or*
- $t(\epsilon), t\downarrow(\epsilon) \in \Omega_i \backslash SC$ *and the terms in* $Alien_{\Omega_i}(t)$ *are in layer-reduced form.*

A substitution σ is in layer-reduced form *if $x\sigma$ is in layer-reduced form for any $x \in Dom(\sigma)$.*

Example 5. (Example 1 continued). Consider $t = dec(\langle enc(x, y), enc(x, z)\rangle, y)$. The terms $\langle x, dec(enc(x, z), y)\rangle$, x, and $dec(enc(x, z), y)$ are layer-reduced forms of respectively t, $fst(t)$ and $snd(t)$.

As stated below, a layer-reduced form is computable provided that a particular case of match-equations is decidable for each Ω_i-theory T_i involved in the combined theory $E = E_1 \cup E_2 = \bigcup_{i=1}^{n} T_i$. A *$SC$-rooted pattern T_i-matching problem* is any match-equation $f(X_1, \ldots, X_m) =^?_{T_i} t$ where $f \in SC$, $X_1, \ldots, X_m$ are pairwise distinct variables, and t is a ground Ω_i-term. Of course, SC-rooted pattern T_i-matching is decidable if T_i-matching is decidable. To get the decidability of SC-rooted pattern T_i-matching, another sufficient condition is to assume that T_i is presented by a finite convergent TRS $\mathcal{R}_i$.

Proposition 2. *([17]) It is possible to compute an E-equal layer-reduced form of any term if SC-rooted pattern T_i-matching and T_i-equality is decidable for each $i = 1, \ldots, n$.*

In Example 2, layer-reduced forms are computable by $\mathcal{R}$-normalization. Lemma 1 can again be applied inductively to rephrase Lemma 2.

Lemma 3. *Let t be a term such that its i-alien subterms are in layer-reduced form. Then $t^{\pi_i} =_{E_i} (t\downarrow)^{\pi_i}$.*

3.3 Frames in Combined Theories

Along the lines of Lemma 3, we present a new result which will be instrumental in proving the correctness of combination methods for knowledge problems popular in protocol analysis, namely, the deduction and the static equivalence. These problems are defined using the notion of frame to express the intruder knowledge. A *frame*, $\phi = \nu\tilde{n}.\sigma$, consists of a finite set of restricted names, $\tilde{n}$, and a substitution σ such that $Ran(\sigma)$ contains only ground terms. This definition is borrowed from the applied pi-calculus [2] and more insight behind the definition is given is Section 4.

We say that a frame $\phi = \nu\tilde{n}.\sigma$ is in layer-reduced form if σ is in layer-reduced form. Given a term t, $St(t) = \{t\} \cup \bigcup_{a \in Alien(t)} St(a)$. For a set of terms T, $St(T) = \bigcup_{t \in T} St(t)$ and $fn(T) = \bigcup_{t \in T} fn(t)$. For a substitution σ, $St(\sigma) = St(Ran(\sigma))$. The set of terms $T \cup Ran(\sigma)$ is abbreviated into $T \sqcup \sigma$.

Definition 4. *Let $\phi = \nu\tilde{n}.\sigma$ be a frame, and t a ground term. We denote $\phi \Vdash_E t$ 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.*

Abstraction constants are particular restricted names. When a constant abstraction is performed to a get a pure problem, only finitely many terms are abstracted and only finitely many fresh names are introduced. For sake of simplicity, we assume that this finite set of abstraction constants is already included in the set of restricted names ($\tilde{n}$) of the considered frame. Thanks to this assumption, the i-abstraction of a frame can be defined without introducing new names to be restricted: they are already restricted. Also, we can assume without loss of generality that constants abstracting terms not in the knowledge problem are not restricted. All these assumptions can be formalized as follows.

Assumption 2 *Consider a finite set F of frames in layer-reduced form and a finite set T of terms in layer-reduced form. Let $U_\sigma = St(T \sqcup \sigma)\backslash V$ for each $(\nu\tilde{n}.\sigma) \in F$ and a bijection ρ from $(\bigcup_{(\nu\tilde{n}.\sigma)\in F} U_\sigma)/=_E$ to a set of fresh constants. We assume that each frame $(\nu\tilde{n}.\sigma) \in F$ is equipped with a constant abstraction π with range $\mathcal{C}$ such that $\pi(t\downarrow) = \rho([t]_E)$ if $t \in U_\sigma$ and $\tilde{n} \cap \mathcal{C} = \{\rho([t]_E) \mid t \in U_\sigma\}$.*

Assumption 2 is not restrictive. It clarifies the relationship between restricted names $\tilde{n}$ and constants $\mathcal{C}$ used by the constant abstraction.

Definition 5. *Under Assumption 2 introducing F and T, let $\phi = \nu\tilde{n}.\sigma$ be any frame in F. The* completion *of ϕ is the frame $\phi_* = \nu\tilde{n}.\sigma_*$ where*

$$\sigma_* = \sigma\{\chi_t \mapsto t \mid t \in St(T \sqcup \sigma) \cup \tilde{n}, \phi \Vdash_E t, t \notin Ran(\sigma)\}$$

such that the fresh variables χ_t are bijectively mapped to the terms t.
*The i-*abstraction *of ϕ_* is the frame $\phi_*^{\pi_i} = \nu\tilde{n}.\sigma_*^{\pi_i}$ where*

$$\sigma_*^{\pi_i} = \{x \mapsto (x\sigma_*)^{\pi_i} \mid x \in Dom(\sigma_*)\}.$$

The completion of a frame always exists. We will see in Section 4.1 how to compute it.

Example 6. Consider the theory E introduced in Example 1, $T = \emptyset$ and the frame $\phi = \nu\tilde{n}.\sigma$ where $\tilde{n}\backslash\mathcal{C} = \{s_1, k_1, k_2\}$ and $\sigma = \{x_1 \mapsto \langle s_2, k_1\rangle,\ x_2 \mapsto enc(s_1, k_1),\ x_3 \mapsto h(s_2, k_2)\}$. Note that only the names s_1, k_1, and k_2 are restricted but not s_2. We have $St(\sigma) = \{\langle s_2, k_1\rangle,\ enc(s_1, k_1),\ h(s_2, k_2)\}$ and $Ran(\sigma_*)\backslash Ran(\sigma)$ includes s_1 and k_1, since we have $dec(x_2, snd(x_1))\sigma =_E s_1$ and $snd(x_1)\sigma =_E k_1$. Later, we will be able to check that $\phi \not\Vdash_E k_2$.

Theorem 1. *Let $\phi_* = \nu\tilde{n}.\sigma_*$ be the completion of any frame $\phi = \nu\tilde{n}.\sigma$ following Assumption 2. For any term t such that $fn(t) \cap \tilde{n} = \emptyset$, there exists an i-pure term t_i such that $fn(t_i) \cap \tilde{n} = \emptyset$ and $((t\sigma_*)\downarrow)^{\pi_i} =_{E_i} t_i\sigma_*^{\pi_i}$.*

Proof. Proof by induction on the *theory height* of t, formally defined as follows:

– $ht(t) = 1 + \max_{a \in Alien_i(t)} ht(a)$ if t is Σ_i-rooted,
– $ht(t) = 1 + \max_{a \in Alien_j(t)} ht(a)$ if t is $\Sigma_j \backslash \Sigma_i$-rooted for $j \neq i$.

If t is i-pure, then $((t\sigma_*)\downarrow)^{\pi_i} =_{E_i} (t\sigma_*)^{\pi_i}$ by Lemma 2, where $(t\sigma_*)^{\pi_i} = t\sigma_*^{\pi_i}$. In that case, we can define $t_i = t$. Let us now assume that t is not i-pure.

(1) Consider the case t is a Σ_i-rooted term $C_i[a_1, \ldots, a_n]$ where $a_1, \ldots a_n$ are the i-aliens of t. Let $(t\sigma_*)_\downarrow$ be the term obtained from $t\sigma_*$ by replacing its i-alien subterms by their normal forms. We have:

$$\begin{aligned}((t\sigma_*)\downarrow)^{\pi_i} &=_{E_i} ((t\sigma_*)_\downarrow)^{\pi_i} \quad \text{(by Lemma 2)}\\ &= C_i[((a_1\sigma_*)\downarrow)^{\pi_i}, \ldots, ((a_n\sigma_*)\downarrow)^{\pi_i}]\\ &=_{E_i} C_i[a_{1,i}\sigma_*^{\pi_i}, \ldots, a_{n,i}\sigma_*^{\pi_i}] \quad \text{(by induction hypothesis)}\\ &= (C_i[a_{1,i}, \ldots, a_{n,i}])\sigma_*^{\pi_i}\end{aligned}$$

where $C_i[a_{1,i}, \ldots, a_{n,i}]$ is an i-pure term satisfying the name restriction.

(2) Consider the case t is $\Sigma_j \backslash \Sigma_i$-rooted for $j \neq i$.
   – If $(t\sigma_*)\downarrow$ is Σ_i-rooted, then there exists some Σ_i-rooted term t' such that $(t\sigma_*)\downarrow$ is equal to $(t'\sigma_*)\downarrow$, $ht(t') \leq ht(t)$ and t' satisfies the name restriction. Then, the rest of the proof follows the case (1).
   – If $(t\sigma_*)\downarrow$ is $\Sigma_j \backslash \Sigma_i$-rooted, then t_i is given as follows:
      (i) if $(t\sigma_*)\downarrow = s\downarrow$ with $s \in St(T \sqcup \sigma) \cup \tilde{n}$, then $\phi \vdash_E s$ and there exists some $x \in Dom(\sigma_*)$ such that $x\sigma_* = s$, and we define $t_i = x$;
      (ii) otherwise, t_i is defined as the abstraction constant $((t\sigma_*)\downarrow)^{\pi_i}$. This fresh constant cannot occur in $\tilde{n}$: otherwise, it would mean that (i) is satisfied. □

4 Application to Two Notions of Knowledge in Protocols

We now apply the results from Section 3 to two questions on knowledge in protocol analysis; *deduction* and *static equivalence*. We begin by reviewing some background material on protocols and how knowledge can be represented in their analysis. As mentioned in Section 3, the applied pi calculus and frames are used to model attacker knowledge [2]. 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 $\phi = \nu\tilde{n}.\sigma$. This allows us to not only keep the set of messages known by the attacker but also the variables in the domain of σ allow for the consideration of each term and the tracking of the order of transmission of each term. That is, it represents the order in which these messages/terms were obtained and transmitted. We also need to model such cryptographic concepts as nonces, keys, and publicly known values. We do this by using names, which are essentially free constants. Here also, we need to track the names which the attacker knows, such as public values, and the names which the attacker does not know, such as freshly generated nonces. $\tilde{n}$ consists of a finite set of restricted names. The intuition is that these names represent freshly generated names which remain secret from the attacker.

4.1 Deduction Problem

The first combination problem we consider is the problem of deduction. That is, given a frame ϕ, representing the knowledge of the attacker, can a ground term M be deduced from ϕ? We denote the deduction of M from ϕ modulo E by $\phi \vdash_E M$. Deduction is axiomatized by the inference system given in Figure 1.

$$\frac{}{\nu\tilde{n}.\sigma \vdash_E M} \text{ if } \exists x \in Dom(\sigma) \text{ s.t. } x\sigma = M$$

$$\frac{}{\nu\tilde{n}.\sigma \vdash_E s} \text{ if } s \notin \tilde{n}$$

$$\frac{\phi \vdash_E M_1, \ \ldots, \phi \vdash_E M_k}{\phi \ \vdash_E f(M_1, \ \ldots, \ M_k)} \text{ if } f \in \Sigma$$

$$\frac{\phi \vdash_E M}{\phi \ \vdash_E M'} \text{ if } M =_E M'$$

Fig. 1. Deduction Axioms

However, a useful characterization of deduction has been given in [1], relating $\vdash_E$ to $\Vdash_E$ (introduced in Definition 4).

Lemma 4. *([1]) $\phi \vdash_E M$ iff $\phi \Vdash_E M$.*

Lemma 5. *Under Assumption 2 introducing F and T, let $\phi = \nu\tilde{n}.\sigma$ be any frame in F. For any $M \in St(T \sqcup \sigma) \cup \tilde{n}$, we have $\phi \vdash_E M$ if and only if $(\phi_*)^{\pi_1} \vdash_{E_1} M^{\pi_1}$ or $(\phi_*)^{\pi_2} \vdash_{E_2} M^{\pi_2}$.*

Proof. The if-direction is simple. Let us focus on the only-if direction. By definition of ϕ_*, we have that $\phi \vdash_E M$ if and only if $\phi_* \vdash_E M$. Suppose t is a Σ_i-rooted term for some $i \in \{1, 2\}$ such that $t\sigma_* =_E M$ with $fn(t) \cap \tilde{n} = \emptyset$. By Lemma 3 and Theorem 1, we get an E_i-equality $t_i\sigma_*^{\pi_i} =_{E_i} ((t\sigma_*)\downarrow)^{\pi_i} = (M\downarrow)^{\pi_i} =_{E_i} M^{\pi_i}$ where $t_i\sigma_*^{\pi_i}$ is an i-pure term. By construction, the fresh constants introduced in t_i are not restricted, i.e. $fn(t_i) \cap \tilde{n} = \emptyset$. □

Notice that Lemma 5 now provides a modular method for computing σ_*.

Corollary 1. *(Computing σ_*) Assume the deduction problem modulo E_i is decidable for each $i = 1, 2$. Under Assumption 2 introducing F and T, let $\phi = \nu\tilde{n}.\sigma$ be any frame in F. The completion ϕ_* is computable and the range of σ_* is the set S such that $Ran(\sigma) \subseteq S \subseteq St(T \sqcup \sigma) \cup \tilde{n}$ and $M \in S$ if and only if $(\nu\tilde{n}.S)^{\pi_1} \vdash_{E_1} M^{\pi_1}$ or $(\nu\tilde{n}.S)^{\pi_2} \vdash_{E_2} M^{\pi_2}$.*

Proof. If $M \in St(T \sqcup \sigma) \cup \tilde{n}$, then Lemma 5 applies: we have $M \in Ran(\sigma_*)$ iff $\phi \vdash_E M$ iff $(\phi_*)^{\pi_i} \vdash_{E_i} M^{\pi_i}$ for some $i = 1, 2$, where $\phi_* = \nu\tilde{n}.Ran(\sigma_*)$. □

Example 7. (Example 1 continued).
Assume $t_1 = \langle enc(h(a,b),c), enc(enc(c,d),c)\rangle$, $t_2 = \langle h(h(a,b),c), h(enc(c,d),c)\rangle$ and $t_3 = \langle enc(h(a,b),a), enc(enc(c,d),a)\rangle$. Let $T = \{t_1, t_2, t_3\}$ and the frame $\phi = \nu\tilde{n}.\sigma$ where $\tilde{n}\backslash\mathcal{C} = \{a,b,c\}$ and $\sigma = \{X \mapsto \langle h(a,b), enc(c,d)\rangle\}$. The completion ϕ_* is $\nu\tilde{n}.\sigma\{X_1 \mapsto c, X_2 \mapsto h(a,b), X_3 \mapsto enc(c,d), X_4 \mapsto t_1, X_5 \mapsto t_2\}$.

Directly from Lemma 5, we obtain our main result on the deduction problem.

Theorem 2. *Let $E = E_1 \cup E_2$ be a constructor-sharing theory following Assumption 1. The deduction problem modulo E is decidable if the deduction problem modulo E_i is decidable for each $i = 1, 2$.*

By applying Theorem 2, the deduction problem is decidable in a modular way for the combined theories given in Examples 1 and 4.

Example 8. (Example 7 continued). The following terms are deducible in ϕ modulo E since they occur in $Ran(\phi_*)$: c, with the recipe $dec(snd(X), d)$; t_1, with the recipe $enc(X, dec(snd(X), d))$; t_2, with the recipe $h(X, dec(snd(X), d))$. The term t_3 occurs in $T\backslash Ran(\phi_*)$ and so $\phi \not\vdash_E t_3$.

4.2 Static Equivalence

Another form of knowledge is the ability to tell if two frames are *statically equivalent* modulo E, sometimes also called *indistiguishability*. Two terms s and t are *equal* in a frame $\phi = \nu\tilde{n}.\sigma$ modulo an equational theory E, denoted $(s =_E t)\phi$, iff $s\sigma =_E t\sigma$, and $\tilde{n} \cap (fn(s) \cup fn(t)) = \emptyset$. 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 $Dom(\sigma) = Dom(\tau)$ and for all terms s and t, we have $(s =_E t)\phi$ iff $(s =_E t)\psi$.

Given an equational Σ-theory E and a frame ϕ, $Eq_E(\phi)$ denotes the set of Σ-equalities $s = t$ such that $(s =_E t)\phi$. Thanks to the above notation, given $\phi = \nu\tilde{n}.\sigma$ and $\psi = \nu\tilde{n}.\tau$, we have $\phi \approx_E \psi$ if and only if $Dom(\sigma) = Dom(\tau)$ and $Eq_E(\phi) = Eq_E(\psi)$.

Definition 6. *Let $\phi = \nu\tilde{n}.\sigma$ be any frame following Assumption 2 (with $T = \emptyset$). A pair $(s = t, \phi)$ is an* equality candidate *of ϕ if $(fn(s) \cup fn(t)) \cap \tilde{n} = \emptyset$. An equality candidate $(s_i = t_i, \phi_i)$ is i-pure if s_i, t_i and ϕ_i are i-pure.*

Let $\phi_* = \nu\tilde{n}.\sigma_*$ be the completion of ϕ. An equality candidate $(s = t, \phi_*)$ is an E-instance *of an i-pure equality candidate $(s_i = t_i, \phi_*^{\pi_i})$ if there exists some substitution μ such that $s\sigma_* =_E (s_i\sigma_*)\mu$ and $t\sigma_* =_E (t_i\sigma_*)\mu$.*

We now state the relationship between $Eq_E(\phi_*)$ and $Eq_{E_i}(\phi_*^{\pi_i})$. First, any E-instance of any equality in $Eq_{E_i}(\phi_*^{\pi_i})$ leads to an equality in $Eq_E(\phi_*)$.

Lemma 6 (Soundness). *If $s_i = t_i \in Eq_{E_i}(\phi_*^{\pi_i})$, then for any E-instance $(s = t, \phi_*)$ of $(s_i = t_i, \phi_*^{\pi_i})$, we have $s = t \in Eq_E(\phi_*)$.*

Conversely, any equality in $Eq_E(\phi_*)$ is the E-instance of some equality in $Eq_{E_i}(\phi_*^{\pi_i})$:

Lemma 7 (Completeness). *If $s = t \in Eq_E(\phi_*)$, then there exists $s_i = t_i \in Eq_{E_i}(\phi_*^{\pi_i})$ such that $(s = t, \phi_*)$ is an E-instance of $(s_i = t_i, \phi_*^{\pi_i})$.*

Proof. if $s\sigma_* =_E t\sigma_*$, then $(s\sigma_*)\downarrow = (t\sigma_*)\downarrow$. By Theorem 1, we have for any $i = 1, 2$, $s_i(\sigma_*)^{\pi_i} =_{E_i} ((s\sigma_*)\downarrow)^{\pi_i} = ((t\sigma_*)\downarrow)^{\pi_i} =_{E_i} t_i(\sigma_*)^{\pi_i}$ where s_i and t_i are i-pure terms satisfying the name restriction. Moreover, we have $s_i(\sigma_*)^{\pi_i}\pi^{-1} =_E (s_i\sigma_*)\pi^{-1} =_E s\sigma_*$ and $t_i(\sigma_*)^{\pi_i}\pi^{-1} =_E (t_i\sigma_*)\pi^{-1} =_E t\sigma_*$. Consequently, $(s = t, \phi_*)$ is an E-instance of $(s_i = t_i, \phi_*^{\pi_i})$, where $s_i = t_i \in Eq_{E_i}(\phi_*^{\pi_i})$. □

Lemma 8. *Let $\phi = \nu\tilde{n}.\sigma$ and $\psi = \nu\tilde{n}.\tau$ be any two frames following Assumption 2 (with $T = \emptyset$). We have $\phi_* \approx_E \psi_*$ iff $(\phi_*)^{\pi_1} \approx_{E_1} (\psi_*)^{\pi_1}$ and $(\phi_*)^{\pi_2} \approx_{E_2} (\psi_*)^{\pi_2}$.*

Proof. Follows from Lemma 6 and Lemma 7. □

To reduce any static equivalence problem $\phi \approx_E \psi$ into a static equivalence problem of completed frames $\phi_* \approx_E \psi_*$, we still need an additional form of frame extension introducing recipes [13].

Definition 7. *Let $\phi = \nu\tilde{n}.\sigma$ be a frame. A term t is* compatible *with ϕ if $fn(t) \cap \tilde{n} = \emptyset$ and $t\sigma$ is ground. Let Π be a set of terms compatible with ϕ. The Π-*extension *of ϕ is the frame $\Pi\phi = \nu\tilde{n}.\{\chi_t \mapsto t \mid t \in \Pi\}\sigma$.*

Given a term d E-deducible in ϕ, $rcp_\phi(d)$ denotes an admissible recipe of d in ϕ. By extension, given a set D of E-deducible terms in ϕ, $Rcp_\phi(D) = \{rcp_\phi(d) \mid d \in D\}$. If the deduction problem modulo E is decidable, then it is always possible to compute an admissible recipe of d. A brute force method consists in enumerating all possible admissible terms until a recipe r satisfying $r\sigma =_E d$ is found. It is also possible to proceed in a modular way. If the decision procedures known for the deduction problems modulo E_1 and E_2 are indeed "recipe-producing", then the combination method in Section 4.1 can easily be adapted to get a "recipe-producing" decision procedure for the deduction problem modulo $E = E_1 \cup E_2$.

In a way similar to [13], the recipes are used to define a set Π of admissible terms. Then, two new extended frames respectively E-equal to $\Pi\phi$ and $\Pi\psi$ are considered. Formally, two frames $\phi = \nu\tilde{n}.\sigma$ and $\phi' = \nu\tilde{n}.\sigma'$ are said to be *E-equal*, denoted by $\phi =_E \phi'$, if $Dom(\sigma) = Dom(\sigma')$ and $x\sigma =_E x\sigma'$ for any $x \in Dom(\sigma)$.

Lemma 9. *Let $\phi = \nu\tilde{n}.\sigma$, $\psi = \nu\tilde{n}.\tau$, $\bar{\phi} =_E \Pi\phi$, $\bar{\psi} =_E \Pi\psi$ be any frames following Assumption 2 (with $T = \emptyset$), where*

$$\Pi = St(Rcp_\phi(Ran(\sigma_*)\backslash Ran(\sigma)) \cup Rcp_\psi(Ran(\tau_*)\backslash Ran(\tau)))$$

Then, we have (i) $(\bar{\phi})_* = \bar{\phi}$ and $(\bar{\psi})_* = \bar{\psi}$; (ii) $\phi \approx_E \psi$ if and only if $\bar{\phi} \approx_E \bar{\psi}$.

Proof. Let us first prove that $(\bar{\phi})_* = \bar{\phi}$. The set of terms $St(\bar{\sigma}) \cup \tilde{n}$ is a superset of $St(\sigma) \cup \tilde{n}$:

- For any $t \in St(\sigma) \cup \tilde{n}$, we have $t \in Ran(\bar{\sigma}_*)$ implies $t \in Ran(\sigma_*)$ and then $t \in Ran(\bar{\sigma})$;

– For any other term $\bar{t} \in St(\bar{\sigma})$, $\bar{t} \in Ran(\bar{\sigma})$ since $St(\Pi) = \Pi$.

Hence, $Ran(\bar{\sigma}_*) \subseteq Ran(\bar{\sigma})$ and so $(\bar{\phi})_* = \bar{\phi}$. Similarly, we prove that $(\bar{\psi})_* = \bar{\psi}$. Let us now prove that $\phi \approx_E \psi$ if and only if $\bar{\phi} \approx_E \bar{\psi}$:

– (If direction) If $\phi \not\approx_E \psi$, then there exists $s = t$ such that $s = t \in Eq(\phi), s = t \notin Eq_E(\psi)$ or $s = t \notin Eq(\phi), s = t \in Eq_E(\psi)$, where $fv(s = t) \cap (Dom(\bar{\sigma}) \backslash Dom(\sigma)) = \emptyset$.
– (Only-if direction) Assume $\phi \approx_E \psi$. For any $s = t \in Eq_E(\bar{\phi})$, there exist, by definition of $\bar{\phi}$, two terms s', t' such that $s'\sigma =_E s\bar{\sigma} =_E t\bar{\sigma} =_E t'\sigma$ and $(fn(s') \cup fn(t')) \cap \tilde{n} = \emptyset$. Hence $s' = t' \in Eq_E(\phi)$ and so, by assumption, $s' = t' \in Eq_E(\psi)$. Eventually, we have $s\bar{\tau} =_E s'\tau =_E t'\tau =_E t\bar{\tau}$, which means that $s = t \in Eq_E(\bar{\psi})$. □

Theorem 3. *Let $E = E_1 \cup E_2$ be a constructor-sharing theory following Assumption 1. The static equivalence modulo E and the deduction problem modulo E are both decidable if the static equivalence modulo E_i and the deduction problem modulo E_i are both decidable for each $i = 1, 2$.*

Proof. By Lemma 8, Lemma 9 and Theorem 2. □

Example 9. (Example 1 continued). In [1], the authors introduce locally stable theories and locally finite theories, where respectively the deduction problem and the static equivalence are proven to be decidable. As an example, the theory E_1 is shown in [1] to be both locally stable and locally finite. By reusing the same proof technique as in [1], we can show in a similar way that E_2 is both locally stable and locally finite. Then Theorem 3 allows us to get the decidability of the static equivalence and the deduction problem modulo $E = E_1 \cup E_2$.

Example 10. (Example 4 continued). The decidability of the static equivalence and the deduction problem modulo $E = E_1 \cup E_2$ follows from Theorem 3, since the static equivalence and the deduction problem are decidable modulo E_1 (because E_1 is subterm convergent) and modulo E_2 [1].

Example 11. (Example 1 continued). Let $T = \emptyset$, $\phi = \nu\tilde{n}.\sigma$, $\psi = \nu\tilde{n}.\tau$ where $\tilde{n}\backslash\mathcal{C} = \{k_1,\ k_2\}$, $\sigma = \{x_1 \mapsto \langle enc(h(a,a),k_1), h(c,c)\rangle,\ x_2 \mapsto k_2\}$ and $\tau = \{x_1 \mapsto \langle enc(h(b,b),k_1), h(c,c)\rangle,\ x_2 \mapsto k_2\}$. By computing the completions, we have $\phi_* = \nu\tilde{n}.\sigma\{x_3 \mapsto enc(h(a,a),k_1),\ x_4 \mapsto h(c,c)\}$ and $\psi_* = \nu\tilde{n}.\tau\{x'_3 \mapsto enc(h(b,b),k_1),\ x_4 \mapsto h(c,c)\}$. Consider the set of admissible recipes $\Pi = \{fst(x_1), snd(x_1)\}$. By Definition 7, we use fresh variables, say x_3, x_4, to denote the respective instances of $fst(x_1), snd(x_1)$ in $\Pi\phi$ and in $\Pi\psi$. Thus, we have $\bar{\phi} = \nu\tilde{n}.\sigma\{x_3 \mapsto enc(h(a,a),k_1),\ x_4 \mapsto h(c,c)\} =_E \Pi\phi$ and $\bar{\psi} = \nu\tilde{n}.\tau\{x_3 \mapsto enc(h(b,b),k_1),\ x_4 \mapsto h(c,c)\} =_E \Pi\psi$. Notice that given $\bar{\phi}$ and $\bar{\psi}$ there is still no recipe for which subterms of $h(a,a)$ and $h(b,b)$ can be moved to the root of a term modulo E. Therefore, one would need to use the subterms $enc(h(a,a),k_1)$ and $enc(h(b,b),k_1)$ to distinguish two terms s and t such that $s = t \in Eq_E(\bar{\phi})$ but $s = t \notin Eq_E(\bar{\psi})$. However, this would violate the restriction

that $\tilde{n} \cap (fn(t) \cup fn(s)) = \emptyset$. Hence, $\bar{\phi} \approx_E \bar{\psi}$, and by Lemma 9, $\phi \approx_E \psi$. One can verify that $(\bar{\phi})^{\pi_1} \approx_{E_1} (\bar{\psi})^{\pi_1}$ and $(\bar{\phi})^{\pi_2} \approx_{E_2} (\bar{\psi})^{\pi_2}$, where

$$\begin{array}{l}
(\bar{\phi})^{\pi_1} = \nu\tilde{n}.\{x_1 \mapsto \langle enc(a', k_1), c'\rangle,\ x_2 \mapsto k_2,\ x_3 \mapsto enc(a', k_1),\ x_4 \mapsto c'\} \\
(\bar{\psi})^{\pi_1} = \nu\tilde{n}.\{x_1 \mapsto \langle enc(b', k_1), c'\rangle,\ x_2 \mapsto k_2,\ x_3 \mapsto enc(b', k_1),\ x_4 \mapsto c'\} \\
(\bar{\phi})^{\pi_2} = \nu\tilde{n}.\{x_1 \mapsto \langle e_a, h(c, c)\rangle,\ x_2 \mapsto k_2,\ x_3 \mapsto e_a,\ x_4 \mapsto h(c, c)\} \\
(\bar{\psi})^{\pi_2} = \nu\tilde{n}.\{x_1 \mapsto \langle e_b, h(c, c)\rangle,\ x_2 \mapsto k_2,\ x_3 \mapsto e_b,\ x_4 \mapsto h(c, c)\}.
\end{array}$$

Now consider a small modification to the frames. Let $\phi = \nu\tilde{n}.\sigma$, $\psi = \nu\tilde{n}.\tau$ such that $\sigma = \{x_1 \mapsto \langle enc(h(a, a), k_1), h(c, c)\rangle,\ x_2 \mapsto k_1\}$ and $\tau = \{x_1 \mapsto \langle enc(h(b, b), k_1), h(c, c)\rangle,\ x_2 \mapsto k_1\}$. Now it seems to be the same situation as above. However, $\phi_* = \nu\tilde{n}.\sigma\{x_3 \mapsto enc(h(a, a), k_1),\ x_4 \mapsto h(c, c),\ x_5 \mapsto h(a, a)\}$ and $\psi_* = \nu\tilde{n}.\tau\{x_3' \mapsto enc(h(b, b), k_1),\ x_4 \mapsto h(c, c),\ x_5' \mapsto h(b, b)\}$. The set of admissible recipes is $\Pi = \{fst(x_1), snd(x_1), dec(fst(x_1), x_2)\}$. By Definition 7, we use fresh variables, say x_3, x_4, x_5, to denote the respective instances of $fst(x_1), snd(x_1), dec(fst(x_1), x_2)$ in $\Pi\phi$ and in $\Pi\psi$. Then, $\bar{\phi} = \nu\tilde{n}.\sigma_1\{x_3 \mapsto enc(h(a, a), k_1),\ x_4 \mapsto h(c, c),\ x_5 \mapsto h(a, a)\} =_E \Pi\phi$ and $\bar{\psi} = \nu\tilde{n}.\sigma_2\{x_3 \mapsto enc(h(b, b), k_1),\ x_4 \mapsto h(c, c),\ x_5 \mapsto h(b, b)\} =_E \Pi\psi$. Notice, if $s = x_5$ and $t = h(a, a)$, then $s = t \in Eq_E(\bar{\phi})$ and $s = t \notin Eq_E(\bar{\psi})$. Hence, $\bar{\phi} \not\approx_E \bar{\psi}$, and by Lemma 9, $\phi \not\approx_E \psi$. One can verify that $(\bar{\phi})^{\pi_2} \not\approx_{E_2} (\bar{\psi})^{\pi_2}$, where

$$\begin{array}{l}
(\bar{\phi})^{\pi_2} = \nu\tilde{n}.\{x_1 \mapsto \langle e_a, h(c, c)\rangle, x_2 \mapsto k_1, x_3 \mapsto e_a, x_4 \mapsto h(c, c), x_5 \mapsto h(a, a)\} \\
(\bar{\psi})^{\pi_2} = \nu\tilde{n}.\{x_1 \mapsto \langle e_b, h(c, c)\rangle, x_2 \mapsto k_1, x_3 \mapsto e_b, x_4 \mapsto h(c, c), x_5 \mapsto h(b, b)\}.
\end{array}$$

5 Conclusion

This paper presents new *non-disjoint* combination results for both deduction and static equivalence. That is, if the deduction and static equivalence problems are decidable for two *constructor sharing* theories E_1 and E_2 (following Assumption 1), then they are decidable for the theory $E_1 \cup E_2$. The procedure does not, however, require such properties as locally stable or locally finite [1]. While these are very useful for both obtaining decision procedures and combination (i.e., check for local stability in $E_1 \cup E_2$), our approach is applicable to theories which may not have such properties.

This new approach requires that the frames are extended with some finitely many deducible (sub)terms. For the deduction problem, the notion of completion is sufficient. For the static equivalence, another form of frame extension, introducing recipes, is required to get a modular decision procedure. Thus, it nicely illustrates some of the differences between the two problems.

A natural future work is to study how we could move beyond the sharing of absolutely free constructors, e.g., to allow Associative-Commutative constructors. One possible approach we are investigating is related to our work on hierarchical combination [18].

Acknowledgments: We would like to thank Véronique Cortier and Steve Kremer for the thoughtful comments and discussions.

References

1. M. Abadi and V. Cortier. Deciding knowledge in security protocols under equational theories. *Theor. Comput. Sci.*, 367(1-2):2–32, 2006.
2. M. Abadi and C. Fournet. Mobile values, new names, and secure communication. In *Proceedings of the 28th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, POPL'01, pages 104–115, New York, NY, USA, 2001. ACM.
3. A. Armando, D. A. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuéllar, P. H. Drielsma, P. Héam, O. Kouchnarenko, J. Mantovani, S. Mödersheim, D. von Oheimb, M. Rusinowitch, J. Santiago, M. Turuani, L. Viganò, and L. Vigneron. The AVISPA tool for the automated validation of internet security protocols and applications. In K. Etessami and S. K. Rajamani, editors, *Computer Aided Verification, 17th International Conference, CAV 2005, Edinburgh, Scotland, UK, Proceedings*, volume 3576 of *Lecture Notes in Computer Science*, pages 281–285. Springer, 2005.
4. F. Baader and T. Nipkow. *Term rewriting and all that.* Cambridge University Press, New York, NY, USA, 1998.
5. F. Baader and K. U. Schulz. Unification in the union of disjoint equational theories: Combining decision procedures. *Journal of Symbolic Computation*, 21(2):211 – 243, 1996.
6. F. Baader and C. Tinelli. Deciding the word problem in the union of equational theories. *Inf. Comput.*, 178(2):346–390, 2002.
7. M. Baudet, V. Cortier, and S. Delaune. YAPA: A generic tool for computing intruder knowledge. *ACM Trans. Comput. Log.*, 14(1):4, 2013.
8. B. Blanchet. 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*, pages 82–96. IEEE Computer Society, 2001.
9. R. Chadha, V. Cheval, Ştefan Ciobâcă, and S. Kremer. Automated verification of equivalence properties of cryptographic protocols. *ACM Trans. Comput. Log.*, 17(4):23:1–23:32, 2016. Available as Research Report at https://hal.inria.fr.
10. Y. Chevalier and M. Rusinowitch. Hierarchical combination of intruder theories. *Inf. Comput.*, 206(2-4):352–377, 2008.
11. H. Comon-Lundh and R. Treinen. Easy intruder deductions. In N. Dershowitz, editor, *Verification: Theory and Practice, Essays Dedicated to Zohar Manna on the Occasion of His 64th Birthday*, volume 2772 of *Lecture Notes in Computer Science*, pages 225–242. Springer, 2003.
12. B. Conchinha, D. A. Basin, and C. Caleiro. FAST: an efficient decision procedure for deduction and static equivalence. In M. Schmidt-Schauß, editor, *Proceedings of RTA 2011, Novi Sad, Serbia*, volume 10 of *LIPIcs*, pages 11–20. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2011.
13. V. Cortier and S. Delaune. Decidability and combination results for two notions of knowledge in security protocols. *Journal of Automated Reasoning*, 48(4):441–487, 2010.
14. C. J. F. Cremers. The Scyther tool: Verification, falsification, and analysis of security protocols. In A. Gupta and S. Malik, editors, *Computer Aided Verification, 20th International Conference, CAV 2008, Princeton, NJ, USA, July 7-14, 2008, Proceedings*, volume 5123 of *Lecture Notes in Computer Science*, pages 414–418. Springer, 2008.

15. Ştefan Ciobâcă, S. Delaune, and S. Kremer. Computing knowledge in security protocols under convergent equational theories. *J. Autom. Reasoning*, 48(2):219–262, 2012.
16. D. Dolev and A. C. Yao. On the security of public key protocols (extended abstract). In *22nd Annual Symposium on Foundations of Computer Science, Nashville, Tennessee, USA, 28-30 October 1981*, pages 350–357. IEEE Computer Society, 1981.
17. E. Domenjoud, F. Klay, and C. Ringeissen. Combination techniques for non-disjoint equational theories. In A. Bundy, editor, *Automated Deduction - CADE-12, 12th International Conference on Automated Deduction, Nancy, France, June 26 - July 1, 1994, Proceedings*, volume 814 of *Lecture Notes in Computer Science*, pages 267–281. Springer, 1994.
18. S. Erbatur, D. Kapur, A. M. Marshall, P. Narendran, and C. Ringeissen. Hierarchical combination. In M. P. Bonacina, editor, *Automated Deduction - CADE-24 - 24th International Conference on Automated Deduction, Lake Placid, NY, USA, June 9-14, 2013. Proceedings*, volume 7898 of *Lecture Notes in Computer Science*, pages 249–266. Springer, 2013.
19. S. Escobar, C. A. Meadows, and J. Meseguer. Maude-NPA: Cryptographic protocol analysis modulo equational properties. In *Foundations of Security Analysis and Design V, FOSAD 2007/2008/2009 Tutorial Lectures*, volume 5705 of *Lecture Notes in Computer Science*, pages 1–50. Springer, 2007.
20. J. Millen and V. Shmatikov. Constraint solving for bounded-process cryptographic protocol analysis. In *Proceedings of the 8th ACM Conference on Computer and Communications Security*, CCS'01, pages 166–175, New York, NY, USA, 2001. ACM.
21. S. Mödersheim and L. Viganò. 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*, volume 5705 of *Lecture Notes in Computer Science*, pages 166–194. Springer, 2009.
22. L. C. Paulson. The inductive approach to verifying cryptographic protocols. *Computer Security*, 6:85128, 1998.
23. B. Schmidt, S. Meier, C. J. F. Cremers, and D. A. Basin. Automated Analysis of Diffie-Hellman Protocols and Advanced Security Properties. In S. Chong, editor, *25th IEEE Computer Security Foundations Symposium, CSF 2012, Cambridge, MA, USA, June 25-27, 2012*, pages 78–94. IEEE Computer Society, 2012.
24. M. Schmidt-Schauß. Unification in a combination of arbitrary disjoint equational theories. *Journal of Symbolic Computation*, 8:51–99, July 1989.
25. A. Tiu, R. Goré, and J. E. Dawson. A proof theoretic analysis of intruder theories. *Logical Methods in Computer Science*, 6(3), 2010.
26. M. Turuani. The CL-Atse protocol analyser. In F. Pfenning, editor, *Term Rewriting and Applications, 17th International Conference, RTA 2006, Seattle, WA, USA, August 12-14, 2006, Proceedings*, volume 4098 of *Lecture Notes in Computer Science*, pages 277–286. Springer, 2006.