Security protocols, constraint systems, and group theories

Stéphanie Delaune, Steve Kremer, Daniel Pasaila

Security protocols, constraint systems, and group theories *

Stéphanie Delaune[1], Steve Kremer[2], and Daniel Pasaila[1,3]

[1] LSV, CNRS & ENS Cachan & INRIA Saclay Île-de-France, France
[2] LORIA, INRIA Nancy Grand Est, France
[3] Google, Inc.

Abstract. When formally analyzing security protocols it is often important to express properties in terms of an adversary's inability to distinguish two protocols. It has been shown that this problem amounts to deciding the equivalence of two constraint systems, *i.e.*, whether they have the same set of solutions. In this paper we study this equivalence problem when cryptographic primitives are modeled using a group equational theory, a special case of monoidal equational theories. The results strongly rely on the isomorphism between group theories and rings. This allows us to reduce the problem under study to the problem of solving systems of equations over rings. We provide several new decidability and complexity results, notably for equational theories which have applications in security protocols, such as exclusive or and Abelian groups which may additionally admit a unary, homomorphic symbol.

1 Introduction

Automated verification methods used for the analysis of security protocols have been shown extremely successful in the last years. They have for instance been able to discover flaws in the Single Sign On Protocols used in Google Apps [5]. In 2001, J. Millen and V. Shmatikov [19] have shown that confidentiality properties can be encoded as satisfiability of a constraint system. This approach has been widely studied and extended both in terms of the supported cryptographic primitives and security properties (*e.g.* [11, 7]).

Recently, many works have concentrated on indistinguishability properties, which state that two slightly different protocols *look the same* to an adversary who interacts with either one of the protocols. The notion of indistinguishability can be modelled using equivalences from cryptographic calculi (*e.g.* [3, 2]) and are useful to model a variety of properties such as resistance to guessing attacks in password based protocols [7] as well as anonymity like properties in various applications [16, 4]. More generally, indistinguishability allows one to model security by the means of ideal systems, which are correct by construction [3]. In

* The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC grant agreement n° 258865, project ProSecure, and the project JCJC VIP ANR-11-JS02-006.

2005, M. Baudet has shown that the equivalence of traces can again be encoded using constraint systems: instead of deciding whether a constraint system is satisfiable one needs to decide whether two constraint systems have the same set of solutions. M. Baudet [7], and later Y. Chevalier and M. Rusinowitch [10], have proven the equivalence of two constraint systems decidable when cryptographic primitives are modelled by a subterm convergent equational theory. Subsequently more practical procedures have been implemented in prototype tools [8, 22].

Our contributions. We continue the study of the problem of deciding the equivalence of constraint systems used to model security protocols. In particular we consider the case where cryptographic primitives are modelled using a *group theory*. Group theories are a special case of monoidal theories which have been extensively studied by F. Baader and W. Nutt [20, 6] who have provided a complete survey of unification in these theories. Group theories include theories for exclusive or and Abelian groups. These theories are useful to model many security protocols (see [13]), as well as for modeling low level properties of encryption schemes and chaining modes.

More precisely we provide several new decidability and complexity results for the equivalence of constraint systems. We consider exclusive or and Abelian Groups which may also contain a unary homomorphic symbol. Our results rely on an encoding of the problem in systems of equations on a ring associated to the equational theory under study.

We may note that these equational theories have been previously studied for deciding the satisfiability of constraint systems [17] and for the static equivalence problem [12]. To the best of our knowledge these are however the first results to decide equivalence of constraint systems for these theories, which in contrast to static equivalence considers the presence of a fully active adversary. We also note that studying group theories may seem very restricted since they do not contain the equational theories for classical operators like encryption or signatures. However, combination results for disjoint equational theories for the problems of satisfiability of constraint systems [9] and static equivalence [12] have already been developed and we are confident that similar results can be obtained for equivalence properties.

Outline of the paper. In Section 2 we recall some basic notation and the central notion of group theory. Then, in Section 3, we introduce the notion of constraint systems and define the two problems we are interested in. The sections 4, 5, and 6 are devoted to the study of the satisfiability and equivalence problems. Our results are summarized in Section 6. Detailed proofs of our results can be found in [15].

2 Preliminaries

2.1 Terms

A *signature* Σ consists of a finite set of *function symbols*, each with an arity. A function symbol with arity 0 is a *constant symbol*. We assume that $\mathcal{N}$ is an

infinite set of *names* and $\mathcal{X}$ an infinite set of *variables*. The concept of names is borrowed from the applied pi calculus [2] and is used to model fresh, secret values. Let $\mathcal{A}$ be a set of atoms which may consist of names and variables. We denote by $\mathcal{T}(\Sigma, \mathcal{A})$ the set of *terms* over $\Sigma \cup \mathcal{A}$. We write $n(t)$ (resp. $v(t)$) for the set of names (resp. variables) that occur in the term t. A term is *ground* if it does not contain any variable. A *substitution* σ is a mapping from a finite subset of $\mathcal{X}$ called its domain and written $dom(\sigma)$ to $\mathcal{T}(\Sigma, \mathcal{N} \cup \mathcal{X})$. Substitutions are extended to endomorphisms of $\mathcal{T}(\Sigma, \mathcal{X})$ as usual. We use a postfix notation for their application.

2.2 Group theories

Equational theories are very useful for modeling the algebraic properties of the cryptographic primitives. Given a signature Σ, an equational theory E is a set of equations (*i.e.*, a set of unordered pairs of terms in $\mathcal{T}(\Sigma, \mathcal{X})$). Given two terms u and v such that $u, v \in \mathcal{T}(\Sigma, \mathcal{N} \cup \mathcal{X})$, we write $u =_{\mathsf{E}} v$ if the equation $u = v$ is a consequence of E. In this paper, we are particularly interested in the class of group theories, a special case of monoidal theories introduced by F. Baader [6] and W. Nutt [20]. It captures many theories with AC properties, which are known to be difficult to deal with.

Definition 1 (group theory). *A theory* E *over* Σ *is called a* group theory *if it satisfies the following properties:*

1. *The signature* Σ *contains a binary function symbol* $+$*, a unary symbol* $-$ *and a constant symbol* 0*. All other function symbols in* Σ *are unary.*
2. *The symbol* $+$ *is associative-commutative with unit* 0 *and inverse* $-$*. This means that the equations* $x + (y + z) = (x + y) + z$*,* $x + y = y + x$*,* $x + 0 = x$ *and* $x + (-x) = 0$ *are in* E*.*
3. *Every unary function symbol* $\mathsf{h} \in \Sigma$ *is an endomorphism for* $+$ *and* 0*,* i.e. $\mathsf{h}(x + y) = \mathsf{h}(x) + \mathsf{h}(y)$ *and* $\mathsf{h}(0) = 0$*.*

Note that a group theory on a given signature Σ may contain arbitrary additional equalities over Σ. The only requirement is, that at least the laws given above hold. By abuse of notation we sometimes write $t_1 - t_2$ for $t_1 + (-t_2)$.

Example 1. Suppose $+$ is a binary function symbol and 0 a constant. Moreover assume that the others symbols, *i.e* $-$, h, are unary symbols. The equational theories below are group theories.

- The theory ACUN (*exclusive or*) over $\Sigma = \{+, 0\}$ which consist of the axioms for associativity $(x + y) + z = x + (y + z)$ and commutativity $x + y = y + x$ (AC), unit $x + 0 = x$ (U) and Nilpotency $x + x = 0$ (N).[4]
- The theory AG (*Abelian groups*) over $\Sigma = \{+, -, 0\}$ which is generated by the axioms (AC), (U) and $x + -(x) = 0$ (Inv). Note that the equations $-(x + y) = -(x) + -(y)$ and $-0 = 0$ are consequences of the others.

[4] We here omit to explicit the inverse symbol $-$ as it acts as the identity, *i.e.* $-x = x$.

– The theories ACUNh over $\Sigma = \{+, \mathsf{h}, 0\}$ and AGh over $\Sigma = \{+, -, \mathsf{h}, 0\}$: these theories correspond to the ones described above extended by the homomorphism laws (h) for the symbol h, *i.e.*, $\mathsf{h}(x + y) = \mathsf{h}(x) + \mathsf{h}(y)$ and $\mathsf{h}(0) = 0$.

Other examples of monoidal and group theories can be found in [20].

2.3 Group theories define rings

Group theories have an algebraic structure which are *rings*.

Definition 2 (ring). *A* ring *is a set $\mathcal{R}$ (called the* universe *of the ring) with distinct elements 0 and 1 that is equipped with two binary operations $+$ and $\cdot$ such that $(\mathcal{R}, +, 0)$ is an Abelian group, $(\mathcal{R}, \cdot, 1)$ is a monoid, and the following identities hold for all $\alpha, \beta, \gamma \in \mathcal{R}$:*

– $(\alpha + \beta) \cdot \gamma = \alpha \cdot \gamma + \beta \cdot \gamma$ *(right distributivity)*
– $\alpha \cdot (\beta + \gamma) = \alpha \cdot \beta + \alpha \cdot \gamma$ *(left distributivity)*

We call the binary operations $+$ and $\cdot$ respectively the *addition* and the *multiplication* of the ring. The elements 0 and 1 are called respectively *zero* and *unit*. The (additive) inverse of an element $a \in \mathcal{R}$ is denoted $-a$. A ring is *commutative* if its multiplication is commutative.

It has been shown in [20] that for any group theory E there exists a corresponding ring $\mathcal{R}_{\mathsf{E}}$. We can rephrase the definition of $\mathcal{R}_{\mathsf{E}}$ as follows. Let a be a name ($a \in \mathcal{N}$), the universe of $\mathcal{R}_{\mathsf{E}}$ is $\mathcal{T}(\Sigma, \{a\})/\mathsf{E}$, that is the set of equivalence classes of terms built over Σ and a under equivalence by the equational axioms E. The constant 0, the sum $+$ and the additive inverse $-$ of the ring are defined as in the algebra $\mathcal{T}(\Sigma, \{a\})/\mathsf{E}$.

Given an element of the ring $\mathcal{R}_{\mathsf{E}}$, and a term v, multiplication in the ring is defined by $u \cdot v := u[a \mapsto v]$ where $u[a \mapsto v]$ denotes the term u where any occurrence of a has been replaced by v. It can be shown [20] that $\mathcal{R}_{\mathsf{E}}$ is commutative if, and only if, E has commuting homomorphisms, *i.e.*, $\mathsf{h}_1(\mathsf{h}_2(x)) =_{\mathsf{E}} \mathsf{h}_2(\mathsf{h}_1(x))$ for any two homomorphisms h_1 and h_2. For instance, we have that:

– The ring $\mathcal{R}_{\mathsf{ACUN}}$ consists of the two elements 0 and 1 and we have $0 + 1 = 1 + 0 = 1$, $0 + 0 = 1 + 1 = 0$, $0 \cdot 0 = 1 \cdot 0 = 0 \cdot 1 = 0$, and $1 \cdot 1 = 1$. Hence, $\mathcal{R}_{\mathsf{ACUN}}$ is isomorphic to the commutative ring (field) $\mathbb{Z}/2\mathbb{Z}$.
– The ring $\mathcal{R}_{\mathsf{AGh}}$ is isomorphic to $\mathbb{Z}[\mathsf{h}]$ which is a commutative ring. Note that there are two homomorphisms in the theory AGh, namely $-$ and h and these two homomorphisms commute: $\mathsf{h}(-x) = -(\mathsf{h}(x))$.

By abuse of notation, we often omit the $\cdot$ and we mix up the elements of isomorphomic rings. Thus, we will write $2v$ instead of $(a + a) \cdot v$, and $(\mathsf{h} + \mathsf{h}^2)v$ instead of $(\mathsf{h}(a) + \mathsf{h}^2(a)) \cdot v$.

3 Constraint systems

As mentioned in the introduction, constraint systems are quite common (see *e.g.* [19, 11, 7]) to model the possible executions of a protocol once an interleaving has been fixed. We recall here their formalism.

3.1 Definitions

Following the notations of [7], we consider a new set $\mathcal{W}$ of variables, called *parameters* $w_1, w_2, \ldots$ and a new set $\mathcal{X}^2$ of variables called *second-order variables* $X, Y, \ldots$, each variable with an arity, denoted $ar(X)$. We call $\mathcal{T}(\Sigma, \mathcal{W} \cup \mathcal{X}^2)$ the set of second-order terms and $\mathcal{T}(\Sigma, \mathcal{N} \cup \mathcal{X})$ the set of first-order terms, or simply terms. Given a term t we denote by $\mathsf{var}^1(t)$ (resp. $\mathsf{var}^2(t)$) the first-order (resp. second-order) variables of t, *i.e.* $\mathsf{var}^1(t) = v(t) \cap \mathcal{X}$ (resp. $\mathsf{var}^2(t) = v(t) \cap \mathcal{X}^2$). We lift these notations to sets and sequences of terms as expected.

Definition 3. *A* constraint system *is a triple* $(\Phi; \mathcal{D}; \mathcal{E})$ *where:*

- Φ *is a sequence of the form* $\{w_1 \rhd t_1, \ldots, w_\ell \rhd t_\ell\}$ *where* t_i *are terms and* w_i *are parameters;*
- $\mathcal{D}$ *is a set of deducibility constraints of the form* $X \rhd^? x$ *with* $ar(X) < \ell$*;*
- $\mathcal{E}$ *is a set of equalities of the form* $s =^?_{\mathsf{E}} s'$ *where* s*,* s' *are first-order terms.*

In the following we will, by abuse of notation, confuse sequences $\{w_1 \rhd t_1, \ldots, w_\ell \rhd t_\ell\}$ with corresponding substitutions $\{w_1 \to t_1, \ldots, w_\ell \to t_\ell\}$. We will not formally introduce a language for describing protocols and we only informally describe how a constraint system is associated to an interleaving of a protocol. We refer to [19] for a more detailed description. We simply suppose that protocols may perform three kinds of action:

- A protocol may output terms. These terms correspond to the t_i in Φ and represent the adversary's knowledge after having executed part of the protocol. We call the sequence of terms Φ the *frame*.
- A protocol may input terms which can be computed by the adversary. Each input corresponds to a deducibility constraint $X \rhd^? x \in \mathcal{D}$. The second-order variable X of arity k has to be instantiated by a context over the terms $t_1, \ldots, t_k$. This models the computation, used by the adversary to deduce the first-order term that will instantiate x.
- A protocol may perform tests on inputs to check that the terms match some expected values. These tests are modelled by the equality constraints in $\mathcal{E}$ and may as such contain the variables x which correspond to previously received inputs.

Example 2. Consider the group theory AG and an interleaving of a protocol described by the following sequence:

$$\mathsf{out}(a).\mathsf{out}(b).\mathsf{in}(x_1).\mathsf{out}(c + 2x_1).\mathsf{in}(x_2).[x_1 + x_2 = c]$$

where a, b, and c are names in $\mathcal{N}$, $\mathsf{out}(t)$ models the output of term t, $\mathsf{in}(x)$ the input of a term that will be bound to x and $[t_1 = t_2]$ models the conditional which tests that t_1 and t_2 are equal modulo AG, after having instantiated previous input variables. This protocol yields the constraint system $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ where:

- $\Phi = \{w_1 \rhd a, w_2 \rhd b, w_3 \rhd c + 2x_1\}$,
- $\mathcal{D} = \{X_1 \rhd^? x_1, X_2 \rhd^? x_2\}$ with $ar(X_1) = 2$ and $ar(X_2) = 3$, and
- $\mathcal{E} = \{x_1 + x_2 =^?_{\mathsf{E}} c\}$.

Indeed the three elements of the sequence Φ correspond to the three outputs of the protocol. The two deduction constraints in $\mathcal{D}$ model that the adversary needs to provide the inputs. Note that the first input occurs after two outputs. Hence the adversary may refer to w_1 and w_2, but not w_3. This is modelled by setting $ar(X_1) = 2$. As the second input occurs after three outputs we have that $ar(X_2) = 3$. Finally, $\mathcal{E}$ simply consists in the test performed by the protocol.

The *size* of a frame $\Phi = \{w_1 \rhd t_1, \ldots, w_\ell \rhd t_\ell\}$, denoted $|\Phi|$, is its length ℓ. We also assume the following conditions are satisfied on a constraint system:

1. for every $x \in \mathsf{var}^1(\mathcal{C})$, there exists a unique X such that $(X \rhd^? x) \in \mathcal{D}$, and each variable X occurs at most once in $\mathcal{D}$;
2. for every $1 \leq k \leq \ell$, for every $x \in \mathsf{var}^1(t_k)$, there exists $(X \rhd^? x) \in \mathcal{D}$ such that $ar(X) < k$.

These constraints are natural whenever the constraint system models an interleaving of a protocol. Condition 1 simply states that each variable defines a unique input. Condition 2 ensures a form of causality: whenever a term t_k is output it may only use variables that have been input before; the condition $ar(X) < k$ ensures that the adversary when computing the input to be used for x only refers to terms in the frame that have been output before. This second condition is often called *origination property*.

Given a frame $\Phi = \{w_1 \rhd t_1, \ldots, w_n \rhd t_n\}$, and a second-order term T with parameters in $\{w_1, \ldots, w_n\}$ and without second-order variable $T\Phi$ denotes the first-order term obtained from T by replacing each w_i by t_i. We define the *structure* of a constraint system $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ to be $|\Phi|$ and $\mathsf{var}^2(\mathcal{D})$ with their arity.

Example 3. Note that the two additional conditions are fulfilled by the constraint system $\mathcal{C}$ given in Example 2. In particular, we have that the variable x_1 that occurs in t_3 has been introduced by the deducibility constraint $X_1 \rhd^? x_1$ and $ar(X_1) = 2 < 3$. Let $\mathcal{C}' = (\Phi'; \mathcal{D}; \mathcal{E}')$ where $\Phi' = \{w_1 \rhd a', w_2 \rhd b', w_3 \rhd c' + x_1\}$, and $\mathcal{E}' = \{x_2 + 2x_1 =^?_{\mathsf{E}} c'\}$. We have that $\mathcal{C}'$ is a constraint system that has the same structure as $\mathcal{C}$. Note that $|\Phi| = 3 = |\Phi'|$ and $\mathsf{var}^2(\mathcal{D}) = \{X_1, X_2\}$.

3.2 Satisfiability and equivalence problems

First, we have to define the notion of solution of a constraint system.

Definition 4. *A* pre-solution *of a constraint system $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ is a substitution θ such that:*

- *$dom(\theta) = \mathsf{var}^2(\mathcal{C})$, and*
- *$X\theta \in \mathcal{T}(\Sigma, \{w_1, \ldots, w_k\})$ for any $X \in dom(\theta)$ with $ar(X) = k$.*

The substitution λ with $dom(\lambda) = \mathsf{var}^1(\mathcal{C})$ and such that $x\lambda = (X\theta)(\Phi\lambda)$ for any $X \rhd^? x$ in $\mathcal{D}$ is called the first-order extension of θ for $\mathcal{C}$.

Intuitively, in the preceding definition the substitution θ stores the computation done by the adversary in order to compute the messages he sends (stored in λ) during the execution. Note that, because of the definition of a constraint system, once θ is fixed, its first-order extension is uniquely defined.

To obtain a solution we need to additionally ensure that the first-order extension λ of a pre-solution θ verifies the equality constraints in $\mathcal{E}$.

Definition 5. *Let $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ be a constraint system. A* solution *of $\mathcal{C}$ is a pre-solution θ of $\mathcal{C}$ whose first-order extension λ satisfies the equalities,* i.e. *for every $(s =^?_{\mathsf{E}} s') \in \mathcal{E}$, we have that $s\lambda =_{\mathsf{E}} s'\lambda$. In such a case, the substitution λ is called the* first-order solution *of $\mathcal{C}$ associated to θ. The set of solutions of a constraint system $\mathcal{C}$ is denoted $\mathsf{Sol}_{\mathsf{E}}(\mathcal{C})$.*

We now define the two problems we are interested in.

Definition 6. *A constraint system $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ is* satisfiable *if $\mathsf{Sol}_{\mathsf{E}}(\mathcal{C}) \neq \emptyset$.*

In the context of security protocols satisfiability of a constraint system corresponds to the adversary's ability to execute an interleaving of the protocol. This generally corresponds to an attack. For instance confidentiality of some secret term s can be encoded by adding an additional deducibility constraint $X_s \rhd^? x_s$ together with an equality constraint $x_s =^?_{\mathsf{E}} s$ (or equivalently adding a final input $\mathsf{in}(x_s)$ to the protocol and testing that the adversary is able to send the term s by adding $[x_s = s]$).

Definition 7. *Let $\mathcal{C}_1 = (\Phi_1; \mathcal{D}_1; \mathcal{E}_1)$ and $\mathcal{C}_2 = (\Phi_2; \mathcal{D}_2; \mathcal{E}_2)$ be two constraint systems having the same structure. We say that $\mathcal{C}_1$ is included in $\mathcal{C}_2$, denoted by $\mathcal{C}_1 \sqsubseteq \mathcal{C}_2$, if $\mathsf{Sol}_{\mathsf{E}}(\mathcal{C}_1) \subseteq \mathsf{Sol}_{\mathsf{E}}(\mathcal{C}_2)$. They are* equivalent *if $\mathcal{C}_1 \sqsubseteq \mathcal{C}_2$ and $\mathcal{C}_2 \sqsubseteq \mathcal{C}_1$,* i.e. *$\mathsf{Sol}_{\mathsf{E}}(\mathcal{C}_1) = \mathsf{Sol}_{\mathsf{E}}(\mathcal{C}_2)$.*

Again, in the context of security protocols this problem corresponds to the adversary's inability to distinguish whether the protocol participants are executing the interleaving modelled by $\mathcal{C}_1$ or $\mathcal{C}_2$. For the exact encoding we refer the reader to [7].

Example 4. Consider the constraint systems $\mathcal{C}$ and $\mathcal{C}'$ described in Example 2 and Example 3. The substitution $\theta = \{X_1 \mapsto w_1 + w_2, X_2 \mapsto -3w_1 - 3w_2 + w_3\}$ is a pre-solution of both $\mathcal{C}$ and $\mathcal{C}'$. The first-order extension of θ for $\mathcal{C}$ is the substitution $\lambda = \{x_1 \mapsto a + b, x_2 \mapsto -a - b + c\}$ whereas the first-order extension of θ for $\mathcal{C}'$ is the substitution $\lambda' = \{x_1 \mapsto a' + b', x_2 \mapsto -2a' - 2b' + c'\}$. It is easy to check that θ is actually a solution of both $\mathcal{C}$ and $\mathcal{C}'$, and thus both constraint systems are satisfiable. Actually, we have that $\mathcal{C}$ and $\mathcal{C}'$ are equivalent, *i.e.* $\mathsf{Sol}_{\mathsf{AG}}(\mathcal{C}) = \mathsf{Sol}_{\mathsf{AG}}(\mathcal{C}')$.

In what follows, we consider decidability and complexity issues for the satisfiability and equivalence problems for group theories. In particular, we proceed in three main steps:

1. we reduce both problems to the case of simple constraint systems (where the terms t_i that occurs in the frame Φ are ground terms);
2. we show how to encode solutions of a (simple) constraint system in a system of (linear) equations;
3. we conclude by showing how to solve such a system of equations.

4 Towards simple constraint systems

The aim of this section is to show how we can transform constraint systems in order to obtain *simple* constraint systems while preserving satisfiability and inclusion. This transformation has been first introduced in [9] to simplify the satisfiability problem for the exclusive or and Abelian group theories. We reuse it in a more general setting. From now on, we consider a group equational theory (see Definition 1).

Let $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ where $\Phi = \{w_1 \rhd t_1, \ldots, w_\ell \rhd t_\ell\}$. Let $\tau = \{w_1 \to w_1 - M_1, \cdots, w_\ell \to w_\ell - M_\ell\}$ be a substitution with $dom(\tau) = \{w_1, \cdots, w_\ell\}$. We say that the substitution τ is *compatible* with $\mathcal{C}$ iff $M_1, \cdots, M_\ell$ are second-order terms that do not contain parameters and such that $\mathsf{var}^2(M_i) \subseteq \{X \in \mathsf{var}^2(\mathcal{C}) \mid ar(X) < i\}$. We define the constraint system $\mathcal{C}_\tau$ as $(\Phi_\tau; \mathcal{D}; \mathcal{E})$ where:

- $\Phi_\tau = \{w_1 \rhd t'_1, \ldots, w_\ell \rhd t'_\ell\}$, and
- $t'_i = t_i + M_i\{X \to x \mid X \rhd^? x \in \mathcal{D}\}$ for all $1 \leq i \leq \ell$.

Notice that, if τ is compatible with $\mathcal{C}$, the origination property is satisfied for $\mathcal{C}_\tau$, thus $\mathcal{C}_\tau$ is a constraint system. Let θ be a pre-solution of $\mathcal{C}$ (or equivalently of $\mathcal{C}_\tau$). We denote by θ_τ the substitution $(\theta \circ \tau)^m$ where $m = \#\mathsf{var}^2(\mathcal{C})$.

Example 5. Consider again the constraint systems $\mathcal{C}$ and $\mathcal{C}'$ described in Example 2 and Example 3. Let $\tau = \{w_1 \to w_1, w_2 \to w_2, w_3 \to w_3 - (-2X_1)\}$. We have that τ is a substitution compatible with $\mathcal{C}$ and $\mathcal{C}'$. Then, following the definition, we have that $\mathcal{C}_\tau$ is $(\Phi_\tau; \mathcal{D}; \mathcal{E})$ where $\Phi_\tau = \{w_1 \rhd a, w_2 \rhd b, w_3 \rhd c\}$ whereas $\mathcal{C}'_\tau$ is $(\Phi'_\tau; \mathcal{D}; \mathcal{E}')$ where $\Phi'_\tau = \{w_1 \rhd a', w_2 \rhd b', w_3 \rhd c' - x_1\}$

Consider the substitution $\theta = \{X_1 \to w_1 + w_2, X_2 \to -3w_1 - 3w_2 + w_3\}$ as defined in Example 4. We have that $(\theta \circ \tau) = \{X_1 \to w_1 + w_2, X_2 \to -3w_1 - 3w_2 + w_3 + 2X_1\}$, thus $\theta_\tau = (\theta \circ \tau)^2 = \{X_1 \to w_1 + w_2, X_2 \to -w_1 - w_2 + w_3\}$.

It follows that $\lambda = \{x_1 \to a + b, x_2 \to -a - b + c\}$ (as defined in Example 4) is also the first-order extension of θ_τ for $\mathcal{C}_\tau$, and thus $\theta_\tau \in \mathsf{Sol}_{\mathsf{AG}}(\mathcal{C}_\tau)$. Similarly, we have that $\lambda' = \{x_1 \to a' + b', x_2 \to -2a' - 2b' + c'\}$ (as defined in Example 4) is also the first-order extension of θ_τ for $\mathcal{C}'_\tau$, and thus $\theta_\tau \in \mathsf{Sol}_{\mathsf{AG}}(\mathcal{C}'_\tau)$.

The fact that the messages computed by the attacker in both cases are the same can be formally shown. This is the purpose of the following lemma that

shows that the first-order extensions of θ for $\mathcal{C}$ and of θ_τ for $\mathcal{C}_\tau$ coincide. Actually, the changes made in the frame (Φ is transformed into Φ_τ) are compensated by the computations that are performed by the attacker (θ is transformed into θ_τ). This will be used later on for simplifying the two problems we are interested in.

Lemma 1. *Let $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ be a constraint system defined as above and $\tau = \{w_1 \to w_1 - M_1, \cdots, w_\ell \to w_\ell - M_\ell\}$ be a substitution compatible with $\mathcal{C}$. Let θ be a pre-solution of $\mathcal{C}$. Then, the first-order extension of θ for $\mathcal{C}$ is equal to the first-order extension of θ_τ for $\mathcal{C}_\tau$.*

Thanks to this lemma, we are able to establish the following proposition.

Proposition 1. *Let $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ and $\mathcal{C}' = (\Phi'; \mathcal{D}'; \mathcal{E}')$ be two constraint systems having the same structure and such that $|\Phi| = |\Phi'| = \ell$. Let $\tau = \{w_1 \to w_1 - M_1, \ldots, w_\ell \to w_\ell - M_\ell\}$ be a substitution compatible with $\mathcal{C}$ (and $\mathcal{C}'$). We have that:*

1. *$\mathcal{C}$ satisfiable if, and only if, $\mathcal{C}_\tau$ satisfiable;*
2. *$\mathcal{C} \sqsubseteq \mathcal{C}'$ if, and only if, $\mathcal{C}_\tau \sqsubseteq \mathcal{C}'_\tau$.*

Let $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ with $\Phi = \{w_1 \rhd t_1, \ldots, w_\ell \rhd t_\ell\}$. We say that the constraint system $\mathcal{C}$ is *simple* if the terms $t_1, \ldots, t_\ell$ are ground. We observe that for any constraint system there exists a substitution yielding a simple constraint system. Indeed, for any frame $\Phi = \{w_1 \rhd t_1, \ldots, w_\ell \rhd t_\ell\}$ we have that for all $1 \leq i \leq \ell$ there exist t_i^n and t_i^v such that $t_i =_{\mathsf{E}} t_i^n + t_i^v$, $v(t_i^n) = \emptyset$ and $n(t_i^v) = \emptyset$. Now let $\tau_{\mathcal{C}} = \{w_1 \to w_1 - M_1, \cdots, w_\ell \to w_\ell - M_\ell\}$, where $M_i = -t_i^v\{x \to X \mid X \rhd^? x \in \mathcal{D}\}$ for all $1 \leq i \leq \ell$. By construction the system $\mathcal{C}_{\tau_{\mathcal{C}}}$ is simple.

Moreover, we say that constraint systems $\mathcal{C}$ and $\mathcal{C}'$ are *simplifiable* if there exists τ such that both $\mathcal{C}_\tau$ and $\mathcal{C}'_\tau$ are simple. This class of constraint systems is motivated by the fact that when checking *real-or-random* secrecy properties as those studied in [7] we obtain systems that have this property. More precisely when encoding real-or-random properties we obtain systems $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ and $\mathcal{C}' = (\Phi'; \mathcal{D}'; \mathcal{E}')$ such that $\Phi = \{w_1 \rhd t_1, \ldots, w_\ell \rhd t_\ell\}$, $\Phi' = \{w_1 \rhd t'_1, \ldots, w_\ell \rhd t'_\ell\}$ and for some $1 \leq k \leq \ell$ we have that $t_i = t'_i$ for $i \leq k$ and t_i, t'_i are ground when $i > k$. It immediately follows that $\tau_{\mathcal{C}} = \tau_{\mathcal{C}'}$ and hence $\tau_{\mathcal{C}}$ simplifies both systems.
Using Proposition 1, we can reduce:

1. the satisfiability problem of a general constraint systems to the satisfiability problem of a simple constraint system; and
2. the inclusion problem between solutions of general constraint systems to the inclusion problem between solutions of a simple constraint system and a general one, respectively to the inclusion between solutions of simple constraint systems in the case these constraint systems are simplifiable.

Below, we illustrate how Proposition 1 can be applied.

Example 6. Let $\mathcal{C}$ and $\mathcal{C}'$ be the constraint systems defined in Example 2 and in Example 3. We have that $\tau_{\mathcal{C}} = \tau$ where $\tau = \{w_1 \to w_1, w_2 \to w_2, w_3 \to w_3 - (-2X_1)\}$ is the substitution as defined in Example 5. Relying on Proposition 1, it follows that $\mathcal{C} \sqsubseteq \mathcal{C}'$ if, and only if, $\mathcal{C}_\tau \sqsubseteq \mathcal{C}'_\tau$ where $\mathcal{C}_\tau$ and $\mathcal{C}'_\tau$ are defined in Example 5. Thus, the equivalence problem between constraint systems $\mathcal{C}$ and $\mathcal{C}'$ is reduced to the equivalence problem between a simple constraint system $\mathcal{C}_\tau$ and a general constraint system $\mathcal{C}'_\tau$.

The purpose of the next section is to show how to decide this simplified problem in a systematic way.

5 Encoding solutions into systems of equations

The purpose of this section is to show how to construct systems of equations that encode solutions of constraint systems.

5.1 General constraint systems

Consider a constraint system $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ where $\Phi = \{w_1 \rhd t_1, \ldots, w_\ell \rhd t_\ell\}$, $\mathcal{D} = \{X_1 \rhd^? x_1, \ldots, X_m \rhd^? x_m\}$, and $\mathcal{E} = \{s_1 =^?_{\mathsf{E}} s'_1, \ldots, s_n =^?_{\mathsf{E}} s'_n\}$.

Step 1. First, we encode second-order variables as sums of terms containing unknown variables over $\mathcal{R}_{\mathsf{E}}$. Actually, for all $1 \leq i \leq m$, each second-order variable X_i can be seen as a sum $y^i_1 t_1 + \cdots + y^i_{ar(X_i)} t_{ar(X_i)}$, where $y^i_1, \cdots, y^i_{ar(X_i)}$ are unknowns over $\mathcal{R}_{\mathsf{E}}$. Therefore every constraint system $\mathcal{C} = (\Phi; \mathcal{D}; \mathcal{E})$ (as described above) can be brought in the following form:

$$\begin{cases} y^1_1 t_1 + \cdots + y^1_{ar(X_1)} t_{ar(X_1)} = x_1 & s_1 = s'_1 \\ \qquad\qquad \cdots & \cdots \\ y^m_1 t_1 + \cdots + y^m_{ar(X_m)} t_{ar(X_m)} = x_m & s_n = s'_n \end{cases}$$

where for all $1 \leq i \leq m$, $1 \leq j \leq ar(X_i)$, y^i_j are unknowns over $\mathcal{R}_{\mathsf{E}}$, the terms $s_1, s'_1, \ldots, s_n, s'_n$ are first-order terms that contain only variables $x_1, \cdots, x_m$ and for all $1 \leq i \leq m$, the terms $t_1, \ldots, t_{ar(X_i)}$ are first-order terms that contain only variables $x_1, \ldots, x_{i-1}$.

Step 2. Our next goal is to remove variables $x_1, \cdots, x_m$ from the first-order terms $t_1, \cdots, t_{ar(X_m)}$. For each variable x_i, we inductively construct a term $E(x_i)$:

$$E(x_1) = y^1_1 t_1 + \cdots + y^1_{ar(X_1)} t_{ar(X_1)}$$
$$E(x_i) = (y^i_1 t_1 + \cdots + y^i_{ar(X_i)} t_{ar(X_i)})[E(x_1)/x_1, \cdots, E(x_{i-1})/x_{i-1}] \text{ where } i > 1.$$

Clearly, we have that, for all $1 \leq i \leq m$, the term $E(x_i)$ is a term that does not contain variables $x_1, \ldots, x_m$.

Step 3. Finally, we will show how a system of equations can be obtained. Given the constraint system $\mathcal{C}$, let $\mathcal{S}(\mathcal{C})$ denote its associated system of equations that we construct. The variables of $\mathcal{S}(\mathcal{C})$ are $\{y_j^i \mid 1 \leq i \leq m, 1 \leq j \leq ar(X_i)\}$ and each solution σ to $\mathcal{S}(\mathcal{C})$ encodes a second-order substitution $\{X_i \mapsto y_1^i w_1 + \cdots + y_{ar(X_i)}^i w_{ar(X_i)} \mid 1 \leq i \leq n\}$ which is a solution of $\mathcal{C}$.

We take each equation $s_i = s_i'$ in $\mathcal{E}$ and we add a set of equations into the system $\mathcal{S}(\mathcal{C})$. We assume that the equation $s_i = s_i'$ has the form $a_1 x_1 + \cdots + a_m x_m = p_i$, where $a_i \in \mathcal{R}_{\mathsf{E}}$ and p_i is a ground first-order term. Notice that any equation can be brought to this form by bringing factors that contain variables to the left-hand side, and the other factors to the right-hand side. Next, we remove the variables from the left-hand side by replacing them with the terms $E(x_i)$, for all $1 \leq i \leq m$. Thus, we now have the equation $a_1 E(x_1) + \cdots + a_m E(x_m) = p_i$. We obtain an equation for each constant, by taking the corresponding coefficients from the left-hand side and equalizing with the coefficients from the right-hand side. Finally, we add this equation to $\mathcal{S}(\mathcal{C})$. We give below an example to illustrate the construction.

Example 7. Consider the constraint system $\mathcal{C}_\tau'$ defined in Example 5. We have that $\mathcal{C}_\tau' = (\Phi_\tau'; \mathcal{D}; \mathcal{E}')$ where: $\Phi_\tau' = \{w_1 \rhd a', w_2 \rhd b', w_3 \rhd c' - x_1\}$, and $\mathcal{E}' = \{x_2 + 2x_1 =_{\mathsf{E}}^? c'\}$.

Step 1. We rewrite the constraint system $\mathcal{C}_\tau'$ as

$$\begin{cases} y_1^1 a' + y_2^1 b' = x_1 & x_2 + 2x_1 = c' \\ y_1^2 a' + y_2^2 b' + y_3^2 (c' - x_1) = x_2 \end{cases}$$

Step 2. We construct the terms $E(x_1)$ and $E(x_2)$:

$$E(x_1) = y_1^1 a' + y_2^1 b' \qquad \begin{aligned} E(x_2) &= (y_1^2 a' + y_2^2 b' + y_3^2 (c' - x_1))[E(x_1)/x_1] \\ &= y_1^2 a' + y_2^2 b' + y_3^2 c' - y_3^2 y_1^1 a' - y_3^2 y_2^1 b' \end{aligned}$$

Step 3. We take the equation $x_2 + 2x_1 =_{\mathsf{E}}^? c$ and, by replacing x_2 with $E(x_2)$ and x_1 with $E(x_1)$, we obtain: $a(y_1^2 - y_3^2 y_1^1 + 2y_1^1) + b(y_2^2 - y_3^2 y_2^1 + 2y_2^1) + c y_3^2 = c$. Thus, we obtain the following system of equations $\mathcal{S}(\mathcal{C}_\tau')$:

$$\mathcal{S}(\mathcal{C}_\tau') = \left\{ y_1^2 - y_3^2 y_1^1 + 2y_1^1 = 0; \quad y_2^2 - y_3^2 y_2^1 + 2y_2^1 = 0; \quad y_3^2 = 1 \right\}$$

Note that any integer solution over $\mathcal{R}_{\mathsf{E}}$ of the system of equations encodes a solution of the constraint system. For instance, take $y_1^1 = 1, y_2^1 = 1, y_1^2 = -1, y_2^2 = -1, y_3^2 = 1$ which is a solution of $\mathcal{S}(\mathcal{C}_\tau')$. This encodes the substitution $\theta = \{X_1 \to w_1 + w_2, X_2 \to -w_1 - w_2 + w_3\}$, which is a solution of $\mathcal{C}_\tau'$.

Proposition 2. *Let $\mathcal{C}$ and $\mathcal{C}'$ be two constraint systems having the same structure. Let $\mathcal{S}(\mathcal{C})$ and $\mathcal{S}(\mathcal{C}')$ be the systems of equations obtained from $\mathcal{C}$ and $\mathcal{C}'$ using the construction described above. We have that:*

1. *$\mathcal{C}$ is satisfiable if, and only if, $\mathcal{S}(\mathcal{C})$ has a solution;*
2. *$\mathcal{C} \sqsubseteq \mathcal{C}'$ if, and only if, the solutions of $\mathcal{S}(\mathcal{C})$ are also solutions of $\mathcal{S}(\mathcal{C}')$.*

5.2 Simple constraint systems

When the constraint system $\mathcal{C}$ is simple, then $\mathcal{S}(\mathcal{C})$ is a system of *linear* equations.

Lemma 2. *Let $\mathcal{C}$ be a simple constraint system. The system $\mathcal{S}(\mathcal{C})$ is a system of linear equations.*

Indeed, in Step 2, substitutions are no longer needed since the terms $t_1, \ldots, t_\ell$ do not contain variables, *i.e.* we simply define $E(x_i) = (y_1^i t_1 + \cdots + y_{ar(X_i)}^i t_{ar(X_i)})$ for $1 \leq i \leq m$. The following example illustrates this fact.

Example 8. Consider the constraint system $\mathcal{C}_\tau$ defined in Example 5. We have that $\mathcal{C}_\tau = (\Phi_\tau; \mathcal{D}; \mathcal{E})$ where $\Phi_\tau = \{w_1 \rhd a, w_2 \rhd b, w_3 \rhd c\}$, $\mathcal{D} = \{X_1 \rhd^? x_1, X_2 \rhd^? x_2\}$, and $\mathcal{E} = \{x_1 + x_2 =_{\mathsf{E}}^? c\}$.

Step 1. Then, we bring this constraint system into the following form:

$$\begin{cases} y_1^1 a + y_2^1 b = x_1 & x_1 + x_2 = c \\ y_1^2 a + y_2^2 b + y_3^2 c = x_2 \end{cases}$$

Step 2. It follows that: $E(x_1) = y_1^1 a + y_2^1 b$ and $E(x_2) = y_1^2 a + y_2^2 b + y_3^2 c$.

Step 3. Thus, taking equation $x_1 + x_2 = c$ and replacing x_1 with $E(x_1)$ and x_2 with $E(x_2)$ we obtain $a(y_1^1 + y_1^2) + b(y_2^1 + y_2^2) + cy_3^2 = c$. Therefore the obtained system of linear equations $\mathcal{S}(\mathcal{C}_\tau)$ is:

$$\mathcal{S}(\mathcal{C}_\tau) = \left\{ y_1^1 + y_1^2 = 0; \quad y_2^1 + y_2^2 = 0; \quad y_3^2 = 1 \right\}$$

6 Applications and discussion

In this section we will show how to use the previous results to decide satisfiability and equivalence of constraint systems for several equational theories of interest. Relying on Propositions 1 and 2, as well as Lemma 2, we have that:

Theorem 1. *Let E be a group theory and $\mathcal{R}_{\mathsf{E}}$ its associated ring.*

- *The satisfiability problem of a constraint system is reducible in polynomial time to the problem of deciding whether a system of linear equation admits a solution;*
- *The equivalence problem between constraint systems is reducible in polynomial time to the problem of deciding whether the solutions of a system of linear equations are included in the set of solutions of a system of equation. Moreover, if the constraint systems are simplifiable, the latter system can also be assumed to be linear.*

Actually, several interesting group theories induce a ring for which those problems are decidable in PTIME. To prove this, we have shown that:

Proposition 3. *Let $\mathcal{S}_{\mathsf{linear}}$ be a system of linear equations over $\mathbb{Z}/2\mathbb{Z}$ (resp. $\mathbb{Z}$, $\mathbb{Z}[\mathsf{h}]$, $\mathbb{Z}/2\mathbb{Z}[\mathsf{h}]$) and $\mathcal{S}$ be a system of equations over $\mathbb{Z}/2\mathbb{Z}$ (resp. $\mathbb{Z}$, $\mathbb{Z}[\mathsf{h}]$, $\mathbb{Z}/2\mathbb{Z}[\mathsf{h}]$) such that both systems are built on the same set of variables. The problem of deciding whether each solution of $\mathcal{S}_{\mathsf{linear}}$ is a solution of $\mathcal{S}$ is decidable in PTIME.*

Proof. (sketch) Roughly, in case $\mathcal{S}_{\mathsf{linear}}$ is satisfiable (note that otherwise, the inlusion problem is trivial), we first put it in solved form $x_1 = t_1/d_1, \ldots, x_n = t_n/d_n$ where t_i are terms that may contain some additional variables y_j, and d_i are elements in the ring under study. Then, we multiply each equation in $\mathcal{S}$ with a factor that is computed from d_i and $\mathcal{S}$ and we replace in the resulting system each x_i with t_i. Lastly, we check whether these equations are valid or not. If the answer is yes, then this means that the solutions of $\mathcal{S}_{\mathsf{linear}}$ are indeed solutions of $\mathcal{S}$. Otherwise, we can show that the inclusion does not hold. □

Example 9. Consider the system of linear equations $\mathcal{S}(\mathcal{C}_\tau)$ given in Example 8

$$\mathcal{S}(\mathcal{C}_\tau) = \left\{ y_1^1 + y_1^2 = 0; \quad y_2^1 + y_2^2 = 0; \quad y_3^2 = 1 \right\}$$

This system can be rewritten into a solved form as:

$$\mathcal{S}(\mathcal{C}_\tau) = \left\{ y_1^1 = -y_1^2; \quad y_2^1 = -y_2^2; \quad y_3^2 = 1 \right\}$$

Consider also the system of equations $\mathcal{S}(\mathcal{C}'_\tau)$ given in Example 7:

$$\mathcal{S}(\mathcal{C}'_\tau) = \left\{ y_1^2 - y_3^2 y_1^1 + 2y_1^1 = 0; \quad y_2^2 - y_3^2 y_2^1 + 2y_2^1 = 0; \quad y_3^2 = 1 \right\}$$

It can be seen that the solutions of $\mathcal{S}(\mathcal{C}_\tau)$ are also solutions of the equations of $\mathcal{S}(\mathcal{C}'_\tau)$. Indeed, all the terms reduce when replacing y_1^1 with $-y_1^2$, y_2^1 with $-y_2^2$ and y_3^2 with 1, as indicated in the solved form of $\mathcal{S}(\mathcal{C}_\tau)$. Thus, we can finally conclude that $\mathcal{C}_\tau \sqsubseteq \mathcal{C}'_\tau$, and thus $\mathcal{C} \sqsubseteq \mathcal{C}'$ where $\mathcal{C}_\tau, \mathcal{C}'_\tau$ are defined in Example 5 and $\mathcal{C}$ (resp. $\mathcal{C}'$) are defined in Example 2 (resp. Example 3).

Decidability and complexity results are summarized in the table. A brief discussion on each equational theory can be found below.

Theory E	$\mathcal{R}_\mathsf{E}$	Satisfiability	Equivalence
ACUN	$\mathbb{Z}/2\mathbb{Z}$	PTIME [9]	PTIME (*new*)
AG	$\mathbb{Z}$	PTIME [9]	PTIME (*new*)
ACUNh	$\mathbb{Z}/2\mathbb{Z}[\mathsf{h}]$	PTIME	PTIME (*new*)
AGh	$\mathbb{Z}[\mathsf{h}]$	PTIME	PTIME (*new*)

Theory ACUN (exclusive or). The ring corresponding to this equational theory is the finite field $\mathbb{Z}/2\mathbb{Z}$. The satisfiability problem for the theory ACUN has already been studied and shown to be decidable in PTIME [9].

However, the equivalence problem has only been studied in a very particular case, the so-called static equivalence problem [1]. Static equivalence models indistinguishability of two frames, *i.e.* the adversary cannot interact with the protocol. In our setting the problem of static equivalence of frames Φ and Φ' can be rephrased as the equivalence between two particular constraint systems $(\Phi; \mathcal{D}; \{x_1 =^?_\mathsf{E} x_2\})$ and $(\Phi'; \mathcal{D}; \{x_1 =^?_\mathsf{E} x_2\})$ where:

– Φ and Φ' are arbitrary frames of same size that only contain *ground* terms;
– $\mathcal{D} = \{X_1 \rhd^? x_1; X_2 \rhd^? x_2\}$ where $ar(X_1) = ar(X_2) = |\Phi|$.

The static equivalence problem has been shown to be decidable in PTIME [12]. Here, relying on our reduction result (Theorem 1), we show that we can decide the problem of equivalence of general constraint systems in PTIME as well.

Theory AG (Abelian groups). The ring associated to this equational theory is the ring $\mathbb{Z}$ of all integers. There exist several algorithms to compute solutions of linear equations over $\mathbb{Z}$ and to compute a base of the set of solutions (see for instance [21]). Hence, we easily deduce that the satisfiability problem is decidable in PTIME. This was already observed in [9]. Deciding inclusion of solutions of a system of linear equations in solutions of a system of non-linear equations is more tricky but we have shown that it can be done in PTIME (see Proposition 3).

Theories ACUNh and AGh. For the theory ACUNh (resp. AGh) the associated ring is $\mathbb{Z}/2\mathbb{Z}[\mathsf{h}]$ (resp. $\mathbb{Z}[\mathsf{h}]$), *i.e.* the ring of polynomials in one indeterminate over $\mathbb{Z}/2\mathbb{Z}$ (resp. $\mathbb{Z}$). The satisfiability problem for these equational theories has already been studied in [18], but in a slightly different setting. The intruder deduction problem for these theories has been studied in [14] and shown to be decidable in PTIME. Similar to static equivalence the intruder deduction problem considers a passive attacker which simply asks whether a term can be deduced by an adversary from a frame. In our setting we rephrase this problem whether a ground term t can be deduced from Φ as the satisfiability of the particular constraint system $(\Phi; \mathcal{D}; \{x =^?_{\mathsf{E}} t\})$ where:

– Φ is an arbitrary frame that only contains *ground* terms,
– $\mathcal{D} = \{X \rhd^? x\}$ where $ar(X) = |\Phi|$.

In [14] this problem is reduced to the problem of satisfiability of a system of linear equations. Hence, the techniques for the problem of deciding secrecy for a passive adversary are the same as for an active adversary and we immediately obtain the same PTIME complexity as in [14]. However, results obtained on the equivalence problem are new. We are able to use the same technique as for AG to obtain decidability in PTIME. This generalizes and refines the decidability result (without known complexity) for ACUNh and AGh in the particular case of static equivalence [12].

References

1. M. Abadi and V. Cortier. Deciding knowledge in security protocols under equational theories. *Theoretical Computer Science*, 387(1-2):2–32, November 2006.
2. M. Abadi and C. Fournet. Mobile values, new names, and secure communication. In *Proc. 28th ACM Symposium on Principles of Programming Languages (POPL'01)*, pages 104–115. ACM Press, 2001.
3. M. Abadi and A. D. Gordon. A calculus for cryptographic protocols: The spi calculus. *Inf. Comput.*, 148(1):1–70, 1999.

4. M. Arapinis, T. Chothia, E. Ritter, and M. D. Ryan. Analysing unlinkability and anonymity using the applied pi calculus. In *Proc. 23rd Computer Security Foundations Symposium (CSF'10)*, pages 107–121. IEEE Comp. Soc. Press, 2010.
5. A. Armando, R. Carbone, L. Compagna, J. Cuéllar, and M. L. Tobarra. Formal analysis of SAML 2.0 web browser single sign-on: breaking the SAML-based single sign-on for google apps. In *Proc. 6th ACM Workshop on Formal Methods in Security Engineering (FMSE 2008)*, pages 1–10. ACM Press, 2008.
6. F. Baader. Unification in commutative theories. *Journal of Symbolic Computation*, 8(5):479–497, 1989.
7. M. Baudet. Deciding security of protocols against off-line guessing attacks. In *Proc. 12th Conference on Computer and Communications Security (CCS'05)*, pages 16–25. ACM Press, 2005.
8. V. Cheval, H. Comon-Lundh, and S. Delaune. Automating security analysis: symbolic equivalence of constraint systems. In *Proc. 5th International Joint Conference on Automated Reasoning (IJCAR'10)*, LNAI, pages 412–426. Springer, 2010.
9. Y. Chevalier and M. Rusinowitch. Symbolic protocol analysis in the union of disjoint intruder theories: Combining decision procedures. *Theoretical Computer Science*, 411(10):1261–1282, 2010.
10. Y. Chevalier and M. Rusinowitch. Decidability of equivalence of symbolic derivations. *J. Autom. Reasoning*, 48(2):263–292, 2012.
11. H. Comon-Lundh, V. Cortier, and E. Zalinescu. Deciding security properties of cryptographic protocols. application to key cycles. *Transaction on Computational Logic*, 11(2), 2010.
12. V. Cortier and S. Delaune. Decidability and combination results for two notions of knowledge in security protocols. *J. of Autom. Reasoning*, 48(4):441–487, 2012.
13. V. Cortier, S. Delaune, and P. Lafourcade. A survey of algebraic properties used in cryptographic protocols. *Journal of Computer Security*, 14(1):1–43, 2006.
14. S. Delaune. Easy intruder deduction problems with homomorphisms. *Information Processing Letters*, 97(6):213–218, Mar. 2006.
15. S. Delaune, S. Kremer, and D. Pasaila. Security protocols, constraint systems, and group theories. Research Report LSV-12-06, Laboratoire Spécification et Vérification, ENS Cachan, France, 2012. 23 pages.
16. S. Delaune, S. Kremer, and M. D. Ryan. Verifying privacy-type properties of electronic voting protocols. *Journal of Computer Security*, 17(4):435–487, 2009.
17. S. Delaune, P. Lafourcade, D. Lugiez, and R. Treinen. Symbolic protocol analysis for monoidal equational theories. *Inf. Comp.*, 206(2-4):312–351, 2008.
18. P. Lafourcade, D. Lugiez, and R. Treinen. Intruder deduction for AC-like equational theories with homomorphisms. In *Proc. 16th International Conference on Rewriting Techniques and Applications (RTA'05)*, volume 3467 of *LNCS*, pages 308–322, Nara (Japan), 2005. Springer.
19. J. Millen and V. Shmatikov. Constraint solving for bounded-process cryptographic protocol analysis. In *Proc. 8th ACM Conference on Computer and Communications Security (CCS'01)*. ACM Press, 2001.
20. W. Nutt. Unification in monoidal theories. In *Proc. 10th International Conference on Automated Deduction, (CADE'90)*, volume 449 of *LNCS*, pages 618–632, Kaiserslautern (Germany), 1990. Springer.
21. A. Schrijver. *Theory of Linear and Integer Programming*. Wiley, 1986.
22. A. Tiu and J. Dawson. Automating open bisimulation checking for the spi-calculus. In *Proc. 23rd Computer Security Foundations Symposium (CSF'10)*, pages 307–321. IEEE Comp. Soc. Press, 2010.