Easy Intruder Deductions

Hubert Comon-Lundh and Ralf Treinen

Laboratoire Spécification et Vérification
CNRS UMR 8643, INRIA Futurs projet SECSI, and ENS de Cachan
61, Avenue du Président Wilson
F-94235 Cachan cedex, France

http://www.lsv.ens-cachan.fr/{~comon,~treinen}

Abstract. We investigate extensions of the Dolev-Yao model of a passive intruder into a cryptographic protocol by some algebraic properties of cryptographic primitives. We provide sufficient conditions under which the intruder deduction problem is decidable in polynomial time. We apply this result to the equational theory of homomorphism, and show that in this case the intruder deduction problem is linear, provided that the messages are in normal form.

1 Introduction

The automatic verification of cryptographic protocols has obtained a lot of attention these last years since several attacks were found on established protocols [3, 14]. There are, however, strong algorithmic limitations to protocol verification: The verification problem is undecidable, even with several restrictions [8, 1, 4] and it is actually co-recursively enumerable. Hence we are left either to search for attacks, in which case there are complete methods (e.g. bounding the number of sessions as in [12]), or to use abstractions. In the latter case, when the protocol is proved to be secure at the abstract level, then it is secure. However, a failure of proof search does not necessarily indicate that there is an attack.

A cornerstone of both approaches is the *intruder deduction problem*: Given a finite (or regular) set of messages T, and a given message m, is it possible to retrieve m from T? This corresponds to the security decision problem in presence of a passive eavesdropper.

Most of the approaches rely on the so-called *Dolev-Yao model* of an intruder [7]. This model describes in particular the capabilities of an intruder to execute a passive attack against the security of a cryptographic protocol, that is her capabilities to obtain a secret knowledge by analyzing the messages that she has observed. In the classical model as proposed by Dolev and Yao, these capabilities are just to deduce new messages from observed ones by building or decomposing pairs, encryption, and decryption of a cipher under the condition that she knows the key. This model is justified by the *perfect cryptography assumption*, that is the assumption that there is no way to obtain knowledge about the plaintext pertaining to a ciphertext without knowing the key.

The perfect cryptographic assumption conveniently separates the problem of analyzing a concrete cryptographic protocol into a cryptography part, which belongs in general to the realm of mathematics, and a protocol part, which can be seen as a distributed algorithm and hence falls into the realm of computer science. This is a useful abstraction since it helps us concentrate on the algorithmic aspects of a cryptographic protocol. In fact, many attacks to cryptographic protocols have been found using the perfect cryptography assumption, in some cases even attacks against protocols in industrial use for years, see for instance [14].

It is, however, not always possible to separate completely the algorithmic aspects of a cryptographic protocol from the properties of the encryption method used. The reason is that components of commonly used encryption methods have algebraic properties that may interfere with algorithmic properties of a cryptographic protocol. Examples are the properties of the *exclusive or* and homomorphism properties of the RSA encryption algorithm. The attack described by Simmons [13] against the original TMN key distribution algorithm [15], for instance, exploits in a clever way both some of the laws of *exclusive or* and the homomorphism properties of RSA.

This is why there are some recent attempts to include some of these properties into the deduction engine [5, 2]. However, these papers only consider two particular algebraic properties: the properties of *exclusive or*, and the properties of Abelian groups.

In this paper we investigate for which classes of equational axioms the intruder model of Dolev and Yao can be extended such that the problem, whether an intruder can deduce a given secret from a given set of messages, is decidable. As an application, we show in Sect. 5 that the intruder deduction problem in presence of the homomorphism property

$$\{\langle x, y\rangle\}_z = \langle \{x\}_z, \{y\}_z\rangle$$

is decidable in polynomial time.

2 Preliminaries

We summarize some basic notations used in this paper, see [6] for an overview of rewriting.

We assume a fixed signature Σ, containing in particular two distinguished binary function symbols denoting *pairing* and *encryption*. Applications of these function symbols are written in mixfix notation, that is $\langle u, v\rangle$ for the pair consisting of u and v, and $\{u\}_v$ for the cipher built by encryption of the message u with the key v. We denote by Σ^- the signature Σ without the symbols for encryption $\{\cdot\}_\cdot$ and pairing $\langle\cdot,\cdot\rangle$.

$T(\Sigma, X)$ is the set of terms over the signature Σ and the set of variables X, that is the smallest set such that

1. $X \subseteq T(\Sigma, X)$;

2. if $t_1, \ldots, t_n \in T(\Sigma, X)$, and $f \in \Sigma$ has arity $n \geq 0$, then $f(t_1, \ldots, t_n) \in T(\Sigma, X)$.

We abbreviate $T(\Sigma, \emptyset)$ as $T(\Sigma)$; elements of $T(\Sigma)$ are called *Σ-ground terms.* The set of variables occurring in a term t is denoted by $\mathcal{V}(t)$.

The *set of occurrences* of a term t is defined recursively as $\mathcal{O}(f(t_1, \ldots, t_n)) = \{\epsilon\} \cup \bigcup_{i=1\ldots n} i \cdot \mathcal{O}(t_i)$. For instance, $\mathcal{O}(f(a, g(b, x))) = \{\epsilon, 1, 2, 21, 22\}$. The *size* $|t|$ of a term t is defined as its number of occurrences, that is $|t| = cardinality(\mathcal{O}(t))$. If $o \in \mathcal{O}(t)$ then the *subterm of t at position o* is defined recursively by

- $t \mid_\epsilon = t$
- $f(t_1, \ldots, t_n) \mid_{j \cdot o} = t_j \mid_o$

$St(t)$ is the set of (not necessarily strict) *subterms* of the term t, that is $St(t) = \{t \mid_o \mid o \in \mathcal{O}(t)\}$. If t and s are terms and $o \in \mathcal{O}(t)$ then the *grafting* of s onto t at position o is defined recursively as

- $t[\epsilon \leftarrow s] = s$
- $f(t_1, \ldots, t_n)[j \cdot o \leftarrow s] = f(t_1, \ldots, t_{j-1}, t_j[o \leftarrow s], t_{j+1}, \ldots, t_n)$

For instance, $f(a, g(b, x))[22 \leftarrow h(c)] = f(a, g(b, h(c)))$.

A *Σ-equation* is a pair $(l, r) \in T(\Sigma, X)$, commonly written as $l = r$. The relation $=_E$ generated by a set of Σ equations E is the smallest congruence on $T(\Sigma)$ that contains all ground instances of all equations in E.

Example 1. As a running example, we will use the equational theory E_h defined by the single axiom:

$$\{\langle x, y\rangle\}_z = \langle \{x\}_z, \{y\}_z \rangle$$

A *Σ-rewriting system* R is a finite set of so-called *rewriting rules* $l \rightarrow r$ where $l \in T(\Sigma, X)$ and $r \in T(\Sigma, \mathcal{V}(l))$. A term $t \in T(\Sigma, X)$ *rewrites* to s in one step by R if there is a rewriting rule $l \rightarrow r$ in R, an occurrence o and a substitution σ such that $t \mid_o = l\sigma$ and $s = t[o \leftarrow r\sigma]$. We write $\rightarrow^*$ for the reflexive and transitive closure of $\rightarrow$. A term t is in *normal form* if there is no term s with $t \rightarrow s$. If $t \rightarrow^* s$ and s is a normal form then we say that s is a *normal form of* t, and write $s = t\downarrow$, or $t \rightarrow^! s$.

A term rewriting system is called *canonical* if it is

- *strongly terminating*, that is if there is no infinite sequence of the form $t_1 \rightarrow t_2 \rightarrow t_3 \rightarrow \cdots$.
- *locally confluent*, that is if $t \rightarrow s_1$ and $t \rightarrow s_2$ then there exists a term r with $s_1 \rightarrow^* r$ and $s_2 \rightarrow^* r$.

By a well known result (see, e.g., [6]), every canonical rewrite system is *confluent*, that is if $t \rightarrow^* s_1$ and $t \rightarrow^* s_2$ then there exists a term r with $s_1 \rightarrow^* r$ and $s_2 \rightarrow^* r$. As a consequence, in a canonical rewrite system every term has a unique normal form

A *Σ-context* is a λ-term $\lambda x_1, \ldots, x_n.t$ with $t \in T(\Sigma, \{x_1, \ldots, x_n\})$, also written as $t[x_1, \ldots, x_n]$. Application of a context $t[x_1, \ldots, x_n]$ to arguments $s_1, \ldots, s_n$

is written as $t[s_1, \ldots, s_n]$. For instance if $t = f(x, g(y, a))$, then $t[x, y]$ is a context. Its application to the terms b and c is written as $t[b, c]$, which is equal to $f(b, g(c, a))$. Note that it is not required that all of the variables $x_1, \ldots, x_n$ actually occur in t. For instance, $\lambda xy.x$ is a context, and its application to the terms b and c yields the term b.

3 The Intruder Model of Dolev and Yao Modulo an Equational Theory

The deduction system of Dolev and Yao, extended by equational reasoning modulo a set of equational axioms, is given in Fig. 1 for symmetric key encryption (it is not difficult to design a similar deduction system for asymmetric encryption). This proof system lets the intruder choose when, and in which way, she wants to

$$\text{(A)}\quad \frac{u \in T}{T \vdash_1 u} \qquad\qquad \text{(P)}\quad \frac{T \vdash_1 u \qquad T \vdash_1 v}{T \vdash_1 \langle u, v\rangle}$$

$$\text{(UL)}\quad \frac{T \vdash_1 \langle u, v\rangle}{T \vdash_1 u} \qquad\qquad \text{(UR)}\quad \frac{T \vdash_1 \langle u, v\rangle}{T \vdash_1 v}$$

$$\text{(C)}\quad \frac{T \vdash_1 u \qquad T \vdash_1 v}{T \vdash_1 \{u\}_v} \qquad\qquad \text{(D)}\quad \frac{T \vdash_1 \{u\}_v \qquad T \vdash_1 v}{T \vdash_1 u}$$

$$\text{(F)}\quad \frac{T \vdash_1 u_1 \quad \cdots \quad T \vdash_1 u_n}{T \vdash_1 f(u_1, \ldots, u_n)}\ f \in \Sigma^- \qquad\qquad \text{(E)}\quad \frac{T \vdash_1 u \qquad u =_E v}{T \vdash_1 v}$$

Fig. 1. The deduction system of Dolev and Yao extended by equational reasoning.

use the algebraic properties, i.e. the equations of E. While this system defines the expected extension of the Dolev-Yao model by equational reasoning, it is not effective since, because of the E-rule, equivalence classes can be infinite.

In order to remedy this problem we want to exploit the fact that, for a given set of equational axioms E, we have for any term u a unique representative of its equivalence class modulo E written $u{\downarrow_R}$, or more shortly as $u{\downarrow}$ when the set E of equational axioms is known. For instance, this representative may be obtained by a terminating and confluent term rewrite system. The properties that we require of the function $\cdot{\downarrow}$ are:

1. for all ground terms u and v: if $u =_e v$ then $u{\downarrow} = v{\downarrow}$;
2. for all ground terms u: $u =_E u{\downarrow}$.

As a consequence, if $t[\cdot]$ is a context and u a ground term then $t[u{\downarrow}]{\downarrow} = t[u]{\downarrow}$.

Example 2. We continue Example 1. The equation can be turned into a rewrite rule which forms by itself a canonical rewrite system; both orientations are possible. We choose the following orientation:

$$\{\langle x, y\rangle\}_z \to \langle\{x\}_z, \{y\}_z\rangle$$

Our goal in this section is to define a new deduction relation $\vdash$ such that for all sets of terms T and terms u:

1. if $T \vdash u{\downarrow}$ then $T \vdash_1 u{\downarrow}$
2. if $T \vdash_1 u$ then $T \vdash u{\downarrow}$
3. if $T \vdash u$ then u is a normal form, that is $u = u{\downarrow}$

The first two conditions express respectively that the new deduction relation $\vdash$ is sound and complete with respect to the original deduction relation $\vdash_1$, while the last condition restricts the set of provable terms to those in normal form. This will be necessary for the proof normalization in Sect. 4.

A first attempt to define $\vdash$ consists in a set of rules as in Fig. 1 where $\vdash_1$ is replaced by $\vdash_2$, but with every conclusion of a rule $T \vdash_2 t$ replaced by $T \vdash_2 t{\downarrow}$, and the rule (E) dropped. The resulting system is given in Fig. 2. This rule

(A) $\dfrac{u \in T}{T \vdash_2 u{\downarrow}}$ (P) $\dfrac{T \vdash_2 u \qquad T \vdash_2 v}{T \vdash_2 \langle u, v\rangle{\downarrow}}$

(UL) $\dfrac{T \vdash_2 \langle u, v\rangle}{T \vdash_2 u{\downarrow}}$ (UR) $\dfrac{T \vdash_2 \langle u, v\rangle}{T \vdash_2 v{\downarrow}}$

(C) $\dfrac{T \vdash_2 u \qquad T \vdash_2 v}{T \vdash_2 \{u\}_v{\downarrow}}$ (D) $\dfrac{T \vdash_2 \{u\}_v \qquad T \vdash_2 v}{T \vdash_2 u{\downarrow}}$

(F) $\dfrac{T \vdash_2 u_1 \quad \cdots \quad T \vdash_2 u_n}{T \vdash_2 f(u_1, \ldots, u_n){\downarrow}} \quad f \in \Sigma^-$

Fig. 2. An attempt to incorporate term rewriting into the deduction system of Dolev and Yao.

system is obviously sound (condition 1) w.r.t. $\vdash_1$ (since every deduction step in $\vdash_2$ can be mimicked in $\vdash_1$ by applying an additional (E) step) and can only derive normalized terms (condition 3). The naïve idea for showing completeness of this rule system w.r.t. $\vdash_1$ is to show that any proof tree of $\vdash_1$ can be transformed into a proof tree of $\vdash_2$ by just normalizing all terms and dropping applications of the (E) rule. However, this doesn't always work, as can be seen in the following:

1. The rules (UL), (UR), and (D) of $\vdash_1$ require a term to be in a specific syntactic form, like being a pair or being a cipher. However, normalization

of a term t may yield a normal form $t\downarrow$ which is not in the required form even though some term in the equivalence class of t has the required form. As an example consider the equational axiom $\{\langle x, y\rangle\}_z = \langle\{x\}_z, \{y\}_z\rangle$. Depending on how this equation is oriented we can apply to the normal form of the term $\{\langle u, v\rangle\}_r$ either the rules (UL) and (UR), or (D). In the calculus $\vdash_1$, however, we can freely switch between $\{\langle u, v\rangle\}_r$ and $\langle\{u\}_r, \{v\}_r\rangle$.
2. The rule (D) is non-linear in its hypothesis in that there are two occurrences of the variable v. However, if we normalize all nodes in the proof tree then the non-linear applicability condition of rule (D) may no longer be satisfied, since even if $\{u\}_v\downarrow$ is a cipher, it may be different from $\{u\downarrow\}_{v\downarrow}$.

One possible solution to these problems is to impose conditions on the rewrite system which calculates the representatives of the equivalence classes modulo E, like

$$\{u\}_v\downarrow = \{u\downarrow\}_{v\downarrow} \quad \text{for all terms } u \text{ and } v \tag{1}$$
$$\langle u, v\rangle\downarrow = \langle u\downarrow, v\downarrow\rangle \quad \text{for all terms } u \text{ and } v \tag{2}$$

Lemma 1. *If the rewrite system satisfies the conditions 1 and 2 then the deduction system $\vdash_2$ is complete w.r.t. $\vdash_1$.*

Proof. By a simple induction on the proof tree.

However, these conditions are too strong to be useful in case of the equational axioms that we are interested in. For the equational theory $\{\langle x, y\rangle\}_z = \langle\{x\}_z, \{y\}_z\rangle$, for instance, there is no canonical rewrite system satisfying both of the conditions (1) and (2). Hence, we use a different proof system which uses matching modulo E and the existence of a canonical rewrite system for E to determine applicability of a rule. This rule system is given in Fig. 3. Again, it

$$\text{(A)}\quad \frac{u \in T}{T \vdash_3 u\downarrow} \qquad\qquad \text{(UL)}\quad \frac{T \vdash_3 r}{T \vdash_3 u\downarrow} \quad \text{if } \exists v\ \langle u, v\rangle \to^! r$$

$$\text{(P)}\quad \frac{T \vdash_3 u \qquad T \vdash_3 v}{T \vdash_3 \langle u, v\rangle\downarrow} \qquad\qquad \text{(UR)}\quad \frac{T \vdash_3 r}{T \vdash_3 v\downarrow} \quad \text{if } \exists u\ \langle u, v\rangle \to^! r$$

$$\text{(C)}\quad \frac{T \vdash_3 u \qquad T \vdash_3 v}{T \vdash_3 \{u\}_v\downarrow} \qquad\qquad \text{(D)}\quad \frac{T \vdash_3 r \qquad T \vdash_3 v}{T \vdash_3 u\downarrow} \quad \text{if } \{u\}_v \to^! r$$

$$\text{(F)}\quad \frac{T \vdash_3 u_1 \quad \cdots \quad T \vdash_3 u_n}{T \vdash_3 f(u_1, \ldots, u_n)\downarrow} \quad f \in \Sigma^-$$

Fig. 3. Incorporating equational reasoning into the deduction system of Dolev and Yao by using matching modulo an equational theory.

is obvious that $\vdash_3$ is sound w.r.t. to $\vdash_1$, and that only normalized terms can be derived.

Lemma 2. *The deduction system $\vdash_3$ is complete w.r.t. $\vdash_1$.*

Proof. Given a proof tree τ_1 for $T \vdash_1 t$ we transform it into a proof tree τ_3 for $T \vdash_3 t$ by changing all $\vdash_1$ into $\vdash_3$, normalizing all terms to the right of $\vdash_3$ by $\downarrow$, and dropping all applications of E (see Fig. 4). The resulting tree τ_3 has the

$$\frac{\dfrac{\psi_1}{T \vdash_1 u} \qquad u =_E v}{T \vdash_1 v} \quad\Longrightarrow\quad \frac{\psi_1'}{T \vdash_3 v\downarrow}$$

$$\frac{\dfrac{\psi_1}{T \vdash_1 u_1} \qquad \cdots \qquad \dfrac{\psi_n}{T \vdash_1 u_n}}{T \vdash_1 v} \quad\Longrightarrow\quad \frac{\dfrac{\psi_1'}{T \vdash_3 u_1\downarrow} \qquad \cdots \qquad \dfrac{\psi_n'}{T \vdash_3 u_n\downarrow}}{T \vdash_1 v\downarrow}$$

Fig. 4. The transformation of a $\vdash_1$ proof tree ψ into a $\vdash_3$ proof tree ψ' used in the proof of Lemma 2. The first rule applies to (E) steps, while the second rule applies to all steps besides (E).

same leaves than τ_1 and has root $T \vdash_3 t\downarrow$. We show by induction on τ_1 that τ_3 is indeed a proof tree for $\vdash_3$.

- If the last rule applied in τ_1 is (A) the claim follows immediately.
- If the last rule applied in τ_1 is (E) then the claim follows by induction hypothesis since $u\downarrow = v\downarrow$ if $u =_E v$.
- If the last rule applied in τ_1 is one of (P), (C), or (F) then the claim follows by induction hypothesis using the fact that $f(u_1, \ldots, u_n)\downarrow = f(u_1\downarrow, \ldots, u_n\downarrow)\downarrow$.
- If the last rule applied in τ_1 is (D), with hypothesis $\{u\}_v$ and v and conclusion u, then the root of τ_3 is labeled with $u\downarrow$, and its sons by $\{u\}_v\downarrow$ and $v\downarrow$ respectively. By induction hypothesis, the first subtree of τ_3 is a $\vdash_3$-proof tree of $\{u\}_v\downarrow$, and the second subtree of τ_3 is a $\vdash_3$-proof tree of $v\downarrow$. The last step in τ_3 is licensed by the (D) rule of $\vdash_3$ since $\{u\}_{v\downarrow} \to^! \{u\}_{v\downarrow}\downarrow$.
- The case where the last rule application in τ_1 is (UL) or (UR) is proven analogously to the previous case.

4 Normal Proofs

In the classical Dolev-Yao model, as well as in the case of the algebraic theory of *exclusive or*, it has been shown (see e.g. [5]) that, if there is a proof of $T \vdash_1 u$, then there is a proof in which all nodes are subterms of $T \cup \{u\}$. Using the terminology of [11] this means that the intruder theories are *local*. In order to extend this result to other equational theories we have to generalize the notion of a subterm. This leads to the following definition:

Definition 1. *Let $S: 2^{T(\Sigma)} \to 2^{T(\Sigma)}$. A $\vdash_3$-proof tree τ for $T \vdash_3 u$ is S-*normal *if for every node $T \vdash_3 v$ of τ we have that $v \in S(T \cup \{u\})$.*

Hence, *locality* in the sense of [11] is exactly St-normality (remember that $St(t)$ is the set of subterms of t).

In fact only those functions S are useful for us that for a finite argument yield a finite result (see Sect. 5). In the present section we are just interested in the question under which conditions to S we can always transform a proof into a S-normal proof. The considerations in this section are independent of the question whether $S(T)$ is finite for finite T.

Definition 2. *The rewrite system R is* consistent *if*

1. *there are no ground terms t_1, t_2 such that $\langle x, t_1 \rangle \to^! t_2$ or $\langle t_1, x \rangle \to^! t_2$.*
2. *there are no ground terms t_1, t_2 such that $\{x\}_{t_1} \to^! t_2$ or $\{t_1\}_x \to^! t_2$.*

These are reasonable conditions since their violation would make the calculus $\vdash_3$ virtually inconsistent. Violation of the first condition would allow to derive *any* term from the term t_2 (using the rule (UL), resp. (UR)). Violation of the second rule would allow to derive *any* term from the terms t_1 and t_2 (using the rule (D)) and would indicate a serious flaw in the encryption algorithm, respectively would allow to decode the cipher t_2 using *any* key.

For the rest of the paper we will always assume that the rewrite system is consistent.

4.1 Generalizing the Notion of Subterm

Definition 3. *For any term t in normal form let $S(t)$ be the smallest set of terms in normal form such that*

1. *$t \in S(t)$*
2. *For all $f \in \Sigma$: if $f(t_1, \ldots, t_n) \to^! t$ where $t, t_1, \ldots, t_n$ are ground and in normal form, then there are terms r_i, $1 \leq i \leq n$, such that*
   (a) for every i: r_i is either t_i or a variable,
   (b) and $f(r_1, \ldots, r_n) \to^! t$,
   (c) and $r_i \in S(t)$ for every i such that $r_i = t_i$.

The function S extends in a natural way to sets of terms in normal form by $S(T) := \bigcup_{t \in T} S(t)$. We write $S(T, t)$ as abbreviation for $S(T \cup \{t\})$.

Of course we have to assume that in the last case for all ground terms $t, t_1, \ldots, t_n$ there is one unique choice of a set of "dont-care positions", otherwise the existence of a minimal set satisfying these properties is not guaranteed.

Also note that, since t_1 and t_2 are normal terms, any rewriting of $f(t_1, t_2)$ must begin with a rewrite step at the topmost position of this term.

Proposition 1. *If t is in normal form then $St(t) \subseteq S(t)$.*

Proof. By induction on the subterm of t.

Example 3. When $=_E$ is the syntactic equality (perfect cryptography assumption), then we claim that $S = St$. It is easy to verify that St verifies the conditions of Def. 3:

1. By definition.
2. If $f(t_1, \ldots, t_n) \rightarrow^! t$ then $f(t_1, \ldots, t_n) = t$, and hence $t_i \in St(t)$.

Hence and by Prop. 1, $St(t)$ is the smallest set satisfying the conditions of Def. 3, that is $S = St$ in this case.

Proposition 2. *For any term t in normal form and any set T of terms in normal form:*

1. *S is monotone: If $T \subseteq T'$ then $S(T) \subseteq S(T')$.*
2. *S is idempotent: For all T we have that $S(S(T)) = S(T)$.*
3. *S is closed under* subterm *: $St(S(t)) \subseteq S(t)$.*

Proof. 1. Obvious from the way how the definition of S is extended to sets.
2. This is a consequence of the fact that $S(T)$ is defined to be the smallest extension of T satisfying certain closure properties.
3. We have that $St(S(t)) \subseteq S(S(t)) = S(t)$ by Prop. 1 and by idempotency of S.

Example 4. We continue Example 2. We claim that for this rewrite rule, the set $S(t)$ is the smallest set which

1. contains t;
2. is closed under subterm;
3. if $\langle \{u\}_z, \{v\}_z \rangle \in S(t)$ then $\langle u, v \rangle \in S(t)$.

First it is easy to see that any set satisfying the conditions of Def. 3 has to meet the above conditions:

1. $t \in S(t)$ is required by clause 1 of Def. 3.
2. Closure under subterm holds by clause 2 of Def. 3.
3. Since $\{\langle u, v \rangle\}_z \rightarrow^! \langle \{u\}_z, \{v\}_z \rangle$, clause 2 of Def. 3 (where f is the encryption function, $r_1 = \langle u, v \rangle$, and $r_2 = z$) requires that $\langle u, v \rangle \in S(\langle \{u\}_z, \{v\}_z \rangle)$.

Second, we show that the set $S(t)$ as defined above satisfies the conditions of Def. 3.

1. $t \in S(t)$ holds by definition.
2. Suppose that $f(t_1, t_2) \rightarrow^! t$ where t_1, t_2, t are ground and normal, and $f(t_1, t_2) \neq t$. This is only possible when $t_1 = \langle u, v \rangle$, $t_2 = z$ for some terms u, v, and z, and f is the encoding function, and $t = \langle \{u\}_z \downarrow, \{v\}_z \downarrow \rangle$. $S(z) \subseteq S(t)$ is obvious since z is a subterm of t, and it easy to see that $\langle u, v \rangle \in S(t)$.

Furthermore, this set is obviously finite.

Proposition 3. *The operator S has the following closure properties: For all ground terms u, v, r in normal form*

1. *If* $\langle t_1, t_2\rangle \to^! r$ *then* $t_1, t_2 \in S(r)$
2. *If* $\{t_1\}_{t_2} \to^! r$ *then* $t_1, t_2 \in S(r)$

Proof. For the first case (the second case is proven analogously): By condition 2 of Def. 3 there are terms r_1, r_2 such that $\langle r_1, r_2\rangle \to^! r$, and for every i either r_i is a variable or t_i. By the consistency hypothesis, however, it is not possible that one of the r_i is a variable. Hence, the last clause of condition 2 of Def. 3 yields that $t_1, t_2 \in S(r)$.

4.2 Admissibility of the Rewrite System

In the Dolev-Yao model, proving that there are St-normal proofs can be performed, roughly, as follows: the inference rules are divided into *composition rules* (F,C,P) and *decomposition rules* (D,UL,UR,A). For every composition rule, the premises are subterms of the conclusion. For every decomposition rule, it is proved that, if the proof is minimal, then every sub-proof ending with a decomposition rule has a branch which only consists in decomposition rules. Hence the conclusion of a decomposition rule must be a subterm of some term in T.

We follow a similar approach. However, there is a complication, since, for decomposition rules, the properties of S are not always sufficient to ensure that every minimal proof contains a branch consisting of decomposition rules only. This is illustrated by the following example.

Example 5. We continue Examples 2 and 4 and consider the following proof:

$$\text{(UL)}\ \dfrac{\text{(C)}\ \dfrac{\text{(A)}\ \dfrac{\langle u, v\rangle \in T}{T \vdash_3 \langle u, v\rangle} \qquad \text{(A)}\ \dfrac{k \in T}{T \vdash_3 k}}{T \vdash_3 \langle \{u\}_k, \{v\}_k\rangle}}{T \vdash_3 \{u\}_k}$$

The intermediate term $\langle \{u\}_k, \{v\}_k\rangle$ does not belong to $S(\langle u, v\rangle, k, \{u\}_k)$.

There is however an alternative proof with the same hypotheses and the same conclusion:

$$\text{(C)}\ \dfrac{\text{(UL)}\ \dfrac{\text{(A)}\ \dfrac{\langle u, v\rangle \in T}{T \vdash_3 \langle u, v\rangle}}{T \vdash_3 u} \qquad \text{(A)}\ \dfrac{k \in T}{T \vdash_3 k}}{T \vdash_3 \{u\}_k}$$

All nodes of this proof are in $S(\{u\}_k, \langle u, v\rangle, k)$.

That is why we consider proofs which are in normal form w.r.t. the possible transformations pushing decompositions towards the leaves. Otherwise said, we attempt to do decomposition as early as possible. A proof tree of the form

$$
\dfrac{\dfrac{T \vdash_3 t_1 \;\ldots\; T \vdash_3 \langle u, v\rangle \;\ldots\; T \vdash_3 t_n}{T \vdash_3 \langle f(t_1, \ldots, u, \ldots, t_n), f(t_1, \ldots, v, \ldots, t_n)\rangle}\,(\mathrm{F})}{T \vdash_3 f(t_1, \ldots, u, \ldots, t_n)}\,(\mathrm{UL})
$$

will be rewritten into

$$
\dfrac{T \vdash_3 t_1 \;\cdots\; \dfrac{T \vdash_3 \langle u, v\rangle}{T \vdash_3 u}\,(\mathrm{UL}) \;\cdots\; T \vdash_3 t_n}{T \vdash_3 f(t_1, \ldots, u, \ldots, t_n)}\,(\mathrm{F})
$$

Proofs in normal from with respect to this transformation are called *reduced* (see Def. 6 below). A rewrite system will be called *admissible* if it allows this kind of transformation of a proof tree:

Definition 4. *The rewrite system $\rightarrow$ is* admissible *if*

1. *if $\langle u, v\rangle\!\downarrow\, = f(r_1, \ldots, r_n)\!\downarrow$ where $u, v, r_1, \ldots, r_n$ are ground terms in normal form then there are terms in normal form $s_1, \ldots, s_n, t_1, \ldots, t_n$ and Σ-contexts C and D such that $u = C[s_1, \ldots, s_n]\!\downarrow$ and $v = D[t_1, \ldots, t_n]\!\downarrow$, and for all i either $r_i = s_i = t_i$ or $r_i = \langle s_i, t_i\rangle\!\downarrow$.*
2. *if $\{u\}_v\!\downarrow = f(r_1, \ldots, r_n)\!\downarrow$ where $u, v, r_1, \ldots, r_n$ are ground terms in normal form then there are terms in normal form $s_1, \ldots, s_n$ and a Σ-context C such that $u = C[s_1, \ldots, s_n]\!\downarrow$, and for all i either $r_i = s_i$ or $r_i = \{s_i\}_v\!\downarrow$.*

Admissibility expresses that application of a constructor $f \in \Sigma$ commutes in some sense with a succeeding application of a decomposition rule (see Lemma 3).

Example 6. We continue the example $E = \emptyset$ that is $=_E$ is syntactic equality (perfect cryptography assumption).

1. If $\langle u, v\rangle = f(r_1, \ldots, r_n)$ then f is the pairing function, and $u = r_1$, $v = r_2$. Condition 1 is hence satisfied by choosing $r_i = s_i = t_i$, $C = \lambda xy.x$, and $D = \lambda xy.y$.
2. If $\{u\}_v = f(r_1, \ldots, r_n)$ then f is encryption, and $u = r_1$, $v = r_2$. Condition 2 is hence satisfied by choosing $r_i = s_i = t_i$ and $C = \lambda xy.x$.

Example 7. We continue Example 2.

1. Note that $\langle u, v\rangle\!\downarrow = \langle u, v\rangle$ since u and v are in normal form. There are three cases according to f:

(a) If f is pairing then $\langle r_1, r_2\rangle\downarrow = \langle r_1, r_2\rangle$, and it follows that $u = r_1$ and $v = r_2$. Hence, this condition is satisfied by choosing $s_i = r_i = t_i$, $C = \lambda xy.x$ and $D = \lambda xy.y$.
(b) If $f \in \Sigma^-$ then $f(r_1, \ldots, r_n)\downarrow = f(r_1, \ldots, r_n)$, hence this case is not possible.
(c) If f is encryption and r_1 is not a pair then $\{r_1\}_{r_2}\downarrow = \{r_1\}_{r_2}$ and, again, this case is not possible.
(d) If f is encryption and $r_1 = \langle s_1, t_1\rangle$ then $\{\langle s_1, t_1\rangle\}_{r_2}\downarrow = \langle\{s_1\}_{r_2}\downarrow, \{t_1\}_{r_2}\downarrow\rangle$. Hence, $u = \{s_1\}_{r_2}\downarrow$ and $v = \{t_1\}_{r_2}\downarrow$, and we choose $C = D = \lambda xy.\{x\}_y$.

2. As in the previous case, f must be either pairing or encryption.
(a) For the case that f is encryption note that the rewrite system of this example has the property that, if u, v, r_1, r_2 are normal forms and if $\{u\}_v\downarrow = \{r_1\}_{r_2}\downarrow$, then $u = r_1$ and $v = r_2$. We hence conclude in this case with $r_i = s_i = t_i$, and $C = \lambda xy.x$.
(b) If f is pairing then the equation can only hold if $u = \langle u_1, u_2\rangle$. In this case, we have $\{\langle u_1, u_2\rangle\}_v\downarrow = \langle\{u_1\}_v\downarrow, \{u_2\}_v\downarrow\rangle$, and $\langle r_1, r_2\rangle\downarrow = \langle r_1, r_2\rangle$. Hence, $r_1 = \{u_1\}_v\downarrow$ and $r_2 = \{u_2\}_v\downarrow$. We conclude by choosing $r_1 = \{s_1\}_v\downarrow$, $r_2 = \{s_2\}_v\downarrow$ (and hence, $u_1 = s_1$ and $u_2 = s_2$), and the context $\lambda xy.\langle x, y\rangle$.

4.3 Proof Normalization

Definition 5. *A proof tree ψ of $T \vdash_3 t$ is* a proof by decomposition *if either it is a proof by (A), or its last rule application is of one of the forms*

$$\text{(X)}\ \dfrac{\dfrac{\psi'}{T \vdash_3 r}}{T \vdash_3 u} \qquad\qquad \text{(D)}\ \dfrac{\dfrac{\psi'}{T \vdash_3 r} \quad \dfrac{\psi''}{T \vdash_3 u}}{T \vdash_3 u}$$

where (X) is one of (UL), (UR), and where in both cases ψ' is by decomposition.

In other words, a proof by decomposition uses only decomposition rules except when constructing a key (since the subtree ψ'' is not constrained).

Definition 6. *A proof tree ψ of $T \vdash_3 t$ is* reduced *if whenever ψ contains a subtree of one of the forms*

$$\text{(X)}\ \dfrac{\dfrac{\psi'}{T \vdash_3 r}}{T \vdash_3 u} \qquad\qquad \text{(D)}\ \dfrac{\dfrac{\psi'}{T \vdash_3 r} \quad \dfrac{\psi''}{T \vdash_3 u}}{T \vdash_3 u}$$

where (X) is one of (UL), (UR), then ψ' is a proof by decomposition.

Lemma 3. *If there is a proof tree of $T \vdash_3 t$ then there is a reduced proof tree of $T \vdash_3 t$ provided that the rewrite system is admissible.*

Proof. We show that any application of one of the rules (F), (P), or (C), commutes with a succeeding application of any of the rules (UL), (UR), or (D). For example, if we have an application of (F) followed immediately by an application of (UL) (the other cases are shown analogously)

$$\dfrac{\dfrac{T \vdash_3 r_1 \quad \cdots \quad T \vdash_3 r_n}{T \vdash_3 f(r_1, \ldots, r_n)} \quad \to^! r \leftarrow^! \langle u, v\rangle}{T \vdash_3 u}$$

then we can rewrite it into

$$\dfrac{\dfrac{T \vdash_3 \langle s_1, t_1\rangle \to^! r_1}{T \vdash_3 s_1} \quad \cdots \quad \dfrac{T \vdash_3 \langle s_i, t_i\rangle \to^! r_i}{T \vdash_3 s_i} \quad T \vdash_3 r_{i+1} \quad \cdots \quad T \vdash_3 r_n}{T \vdash_3 C[s_1, \ldots, s_i, r_{i+1}, \ldots, r_n] \to^! u}$$

where we assume, to simplify notation, that $r_j = \langle s_j, t_j\rangle\!\downarrow$ for $j \leq i$, and $r_j = s_j = t_j$ for $j > i$. The construction of the context C is in fact a whole proof tree consisting of applications of rules (P), (C), and (F).

A simple strategy which guarantees termination of this procedure is to always rewrite those redexes that are innermost (closest to a leaf).

Lemma 4. *If ψ is a reduced proof of $T \vdash_3 u$, where the last rule application is one of*

$$(X)\ \dfrac{\dfrac{\psi'}{T \vdash_3 r}}{T \vdash_3 u} \qquad\qquad (D)\ \dfrac{\dfrac{\psi'}{T \vdash_3 r} \quad \dfrac{\psi''}{T \vdash_3 u}}{T \vdash_3 u}$$

and (X) is one of (UL), (UR), then $r \in S(T)$.

Proof. By induction on ψ, and case distinction on the last rule application of ψ:

- The claim holds of course in the base case where the last rule application of ψ is by (A).
- Assume that the last rule application of ψ is one of (UL) or (UR), that is $\frac{T \vdash_3 k}{T \vdash_3 k_i}$ with $\langle k_1, k_2\rangle \to^! k$ and $i \in \{1, 2\}$.
  By Prop. 3, we have that $k_1, k_2 \in S(k)$, and the claim follows by induction hypothesis and idempotency of S (Prop. 2).
- Assume that the last rule application of τ is (D), that is $\frac{T \vdash_3 k, T \vdash_3 l}{T \vdash_3 r}$ with $\{r\}_l \to^! k$.
  By Prop. 3, we have that $r \in S(k)$, and the claim follows by induction hypothesis and idempotency of S (Prop. 2).

Lemma 5. *For every proof of $T \vdash_3 u$ there is a S-normal proof of $T \vdash_3 u$ provided that the rewrite system is admissible.*

Proof. By induction on the proof tree ψ for $T \vdash_3 u$. By Lemma 3, we may assume that ψ is reduced.

- If the last rule application is (A) then the claim obviously holds.
- If the last rule application is (UL) (the case of (UR) is analogous), then there is a smaller proof tree ψ' for $T \vdash_3 r$ where $\langle u, v\rangle \to^! r$ for some v, and hence by induction hypothesis there is a S-normal proof τ' of $T \vdash_3 r$ where all nodes of τ' are in $S(T, r)$.
  Since ψ is reduced, τ' is reduced as well and we obtain by Lemma 4 that $r \in S(T)$. Hence and by idempotency of S, all nodes of τ' are in $S(T)$. By consequence, we have a proof of $T \vdash_3 u$ where all nodes are in $S(T) \cup \{u\} \subseteq S(T, u)$, that is a S-normal proof of $T \vdash_3 u$.
- If the last rule application is (D) then there is are smaller proof tree ψ_1 of $T \vdash_3 r$ and ψ_2 of $T \vdash_3 v$, where $\{u\}_v \to^! r$. Hence, by induction hypothesis, there are S-normal proofs τ_1 of $T \vdash_3 r$ and τ_2 of $T \vdash_3 v$.
  Since ψ is reduced, Lemma 4 applies, and we obtain that $r \in S(T)$. By Prop. 3 we obtain that $v \in S(T)$ as well. Hence all nodes of τ_1 and τ_2 are in $S(T)$. By consequence, we obtain a proof of $T \vdash_3 u$ where all nodes are in $S(T) \cup \{u\} \subseteq S(T, u)$, that is a S-normal proof of $T \vdash_3 u$.
- If the last rule application is one (P), (C), or (F), then there are smaller proof trees ψ_i of $T \vdash_3 u_i$, and $f(u_1, \ldots, u_n) \to^! u$. For every i there is by induction hypothesis a S-normal proof of $T \vdash_3 u_i$, and since by condition (2) of Def. 3, $u_i \in S(u)$, we obtain that all nodes of these τ_i are in $S(T, u)$. Hence we obtain a S-normal proof of $T \vdash_3 f(t_1, \ldots, t_n)\downarrow$.

5 Deciding Provability

Now, as in the case of local theories [11], we derive from Lemma 5 decision and complexity results for provability. First, we consider the extension of the notion of order of an inference rule in [11]:

Definition 7. *For each inference rule* $\frac{T\vdash_3 t_1 \ldots T\vdash_3 t_n}{T\vdash_3 t_{n+1}}$*, the* rank *of the rule is the minimal cardinal of a subset of indices* $I \subseteq \{1, \ldots, n+1\}$ *such that, given a ground term (in normal form)* $t_i\theta$ *for every* $i \in I$*, there is at most one instance of the rule* $\frac{T\vdash_3 t_1\theta \ldots T\vdash_3 t_n\theta}{T\vdash_3 t_{n+1}\theta}$.

For instance, in the case of perfect cryptography, the rank of every rule is 1 since there is always a term (either a premise or the conclusion of the inference rule) which contains all meta-variables of the inference rule. We will see that in the example of the homomorphism theory ($E = E_h$), every rule has also rank 1. Note that the rank is always bounded by $n + 1$ if n is the number of premises. (When the number of premises is unbounded, as in the `xor` case, the rank is still bounded by $|S(T, t)|$ for S-normal proofs.)

Finally, we also have to consider the complexity of checking (or computing) an instance of an inference rule. In what follows, we will assume that, for the equational theory E, if $r(E)$ is the maximal rank of the inference rules of Fig. 3, checking that there is an instance of the inference rule corresponding to $r(E)$ terms $t_1, \ldots, t_{r(E)}$ and computing it can be performed in time $F_1(N)$ if N is the maximal size of a premise or conclusion. For instance, in case of perfect cryptography, F_1 is linear. We will also see (Lemma 7) that there is a linear F_1 in the case of the homomorphism theory. (In case of the theory of `xor`, it is a polynomial in $|St(T)|$, as shown in [2].)

Theorem 1. *If for any term t in normal form the set $S(t)$ is finite and can be computed in time $O(F_2(|t|))$ then, given a finite set of terms T and a term t, we can decide in time $O((F_2(|T|+|t|))^{r(E)} \times F_1(s))$ whether $T \vdash_3 t$, where s is the maximal size of a term in $S(T,t)$.*

Proof. The proof is similar to the case of local theories [11]. By Lemma 5, it is sufficient to consider S-normal proofs. We first compute $S(T,t)$ in time $O(F_2(|T|+|t|))$.

Then, for each of the inference rules of Fig. 3, we submit candidates for instances of the rules, both premises and conclusion: this requires considering $O((F_2(|t|+|T|))^{r(E)})$ terms. Next, for each of these instances, we check (and compute the missing terms) whether we submitted a valid instance of the rule: this requires at most time $O(F_1(s))$ if s is the maximal size of a term in $S(T,t)$.

Finally, given all possible valid instances of the inference rules, we are left to decide a HornSat problem, which can be performed in linear time.

Summing up the time requirements, we get $O((F_2(|t|+|T|))^{r(E)} * F_1(s))$.

Note that, in case of perfect encryption, $F_1(x) = x$, $F_2(x) = x$ and $r(E) = 1$ hence the theorem states that deciding provability can be decided in time $O(|T|+|t|)$.

Now, coming back to our running example of homomorphism, we get similar complexity bounds, following the two lemmas:

Lemma 6. *If $E = E_h$ then $S(t)$ can be computed in time $O(|t|)$, that is we have for the equational theory E_h that $F_2(x) = x$.*

Proof. Following Example 4, the following deduction system is complete for $S(t)$:

$$\text{(Base)} \quad \frac{}{t \in S(t)} \qquad\qquad \text{(Subterm)} \quad \frac{u \in S(t_i)}{u \in S(f(t_1, \ldots, t_n))}$$

$$\text{(Pair)} \quad \frac{t \in S(\langle u, v \rangle)}{t \in S(\langle \{u\}_k, \{v\}_k \rangle)}$$

In other words, $S(t)$ is the smallest set which contains t, is closed under subterm and such that $\langle u, v \rangle \in S(t)$ whenever $\langle \{u\}_z, \{v\}_z \rangle \in S(t)$. Then we can compute $S(t)$ as follows:

$$\begin{aligned} S(\langle \{u\}_k, \{v\}_k \rangle) &= \{\langle \{u\}_k, \{v\}_k \rangle, \{u\}_k, \{v\}_k\} \cup S(k) \cup S(\langle u, v \rangle) \\ S(f(u_1, \ldots, u_n)) &= \{f(u_1, \ldots, u_n)\} \cup S(u_1) \cup \ldots \cup S(u_n) \qquad \text{otherwise} \end{aligned}$$

Now, the sum of sizes of arguments of the recursive calls is strictly smaller than the size of the parameter, hence the computation is linear. [1]

Lemma 7. *When $E = E_h$, $r(E) = 1$ and we can choose $F_1(x) = x$ if T is a set of terms in normal form.*

Proof. Let us inspect the inference rules of Fig. 3.

(A) Since T is assumed to contain terms in normal form, we only have to check membership to T.

(P) Given $\langle u, v\rangle \downarrow = \langle u, v\rangle$, we can compute u, v in linear time.

(UL), (UR) Given r in normal form, if u, v are in normal form, then $\langle u, v\rangle \rightarrow^! r$ implies $r = \langle u, v\rangle$, and u and v can be computed in linear time.

(C) Given t, we can check in linear time whether $t = \{u\}_v \downarrow$ for some u, v and, if this is the case, retrieve u, v in linear time. For, t must be of the form $C[\{w_1\}_v, \ldots, \{w_m\}_v]$ where C is a context consisting of pairs only. And then $u = C[w_1, \ldots, w_m]$.

(D) Similar to (C): given r, we can check whether $r = C[\{w_1\}_v, \ldots, \{w_m\}_v]$ and, if so, get $u = C[w_1, \ldots, w_m]$ and v in linear time.

(F) Similar to the previous cases.

In all cases, given only one term either in the premises or the conclusion, we are able to check whether it corresponds to a possible instance of the rule and, if so, compute the instance of the rule: $r(E) = 1$. Moreover, we can choose $F_1(x) = x$ since the computation is linear.

As a consequence of the theorem and previous lemmas, we get:

Theorem 2. *If $E = E_h$, and T, t consist of terms in normal form, then we can decide $T \vdash_3 t$ in time $O(|T| + |t|)$.*

Note that we assume here that T consists of a collection of terms in normal form. However, with our notion of size, normalization may require an exponential time. If we want to consider sets of terms which are not in normal form, we must compact the terms by sharing identical subterms. Then the size of a term would be defined as the number of its distinct subterms. Normalization can be performed in polynomial time and all above results still hold, replacing $|t|$ with $|t| \log |t|$.

[1] We assume here that union is performed in constant time. In other words, we compute a list $S(t)$ rather than a set; $S(t)$ may contain repetitions.

We also could consider that the term t is given in compact form, i.e. that its size is the number of *distinct* subterms of t. In that case, we compute a set $S(t)$, which requires $O(|t| \log |t|)$. The complexity is again the same as in the perfect encryption case.

6 Conclusion

We extended the Dolev-Yao model as well as the decision results concerning intruder deductions to a class of equational theories. As an example, we have shown that intruder deductions are as easy in presence of a homomorphism property (which can be seen as a model for block cipher) as in the perfect cryptography case.

However, the class of equational theories which are in the scope of our results does not extend much beyond this homomorphism example. For instance, our result does not apply to several simple and relevant theories such as:

- the theory of exclusive or
- the theory of Abelian groups
- the theory consisting of the single axiom

$$\langle\langle u, v\rangle, w\rangle = \langle u, \langle v, w\rangle\rangle$$

  (associativity of pairing)
- the theory of inverses $I(I(x)) = x$

nor to any of their combinations with homomorphism. However, the results in [5, 2] and the results of the present paper share many similar patterns suggesting that there is a more general result, encompassing both equational theories.

We tried here to extend the concept of locality introduced in [11]: local proofs only involve subterms of the hypotheses and the conclusion. Local theories are those for which any provable sentence has a local proof. Our extension replaces "subterms" above with an arbitrary function S which applies to the premises and the conclusion.

As future work we will try to extend the results of this paper, exploring two directions:

- design more powerful (larger, but still computable) sets $S(t)$, thus widening the set of S-normal proofs;
- design more powerful proof normalization rules, thus restricting the proofs that have to be S-normal proofs.

For instance, in the *exclusive or* case, $S(t)$ is simply the set of subterms of t. However, the proof normalization rules are stronger. On the opposite side, for associativity of pairing, the set $S(t)$ seems to be too small.

Finally, to conclude this conclusion, let us emphasize that we have to be modest: paraphrasing Z. Manna and A. Pnueli [10], we only prove properties of our mathematical model, which reflects more or less adequately the actual intruders. Exploiting some algebraic properties of cryptographic primitives only provides a (hopefully) better approximation of the intruder, but it is far from modeling her faithfully.

Acknowledgments

The first author thanks Z. Manna for his help when he was visiting Stanford for a sabbatical year and starting this work on cryptographic protocol verification. This work has been partly supported by the ACI VERNAM and RNTL project EVA.

References

1. R. Amadio and W. Charatonik. On name generation and set-based analysis in the Dolev-Yao model. In *Proc. CONCUR '02*, volume 2421 of *LNCS*, pages 499–514, 2002.
2. Y. Chevalier, R. Kuester, M. Rusinowitch, and M. Turuani. An NP decision procedure for protocol insecurity with XOR. In Kolaitis [9], pages 261–270.
3. J. Clark and J. Jacob. A survey of authentication protocol literature, 1997. Available via http://www.cs.york.ac.uk/~jac/papers/drareview.ps.gz.
4. H. Comon and V. Cortier. Tree automata with one memory, set constraints and cryptographic protocols. *Theoretical Comput. Sci.*, 2003. To appear.
5. H. Comon-Lundh and V. Shmatikov. Constraint solving, exclusive or and the decision of confidentiality for security protocols assuming a bounded number of sessions. In Kolaitis [9], pages 271–280.
6. N. Dershowitz and J.-P. Jouannaud. Rewrite systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B - Formal Models and Semantics, chapter 6, pages 243–320. Elsevier Science Publishers and The MIT Press, 1990.
7. D. Dolev and A. Yao. On the security of public key protocols. *IEEE Transactions on Information Theory*, 29(2):198–208, 1983.
8. N. Durgin, P. Lincoln, J. Mitchell, and A. Scedrov. Undecidability of bounded security protocols. In *Proc. Workshop on formal methods in security protocols*, Trento, Italy, 1999.
9. P. Kolaitis, editor. *Eighteenth Annual IEEE Symposium on Logic in Computer Science*, Ottawa, Canada, June 2003. IEEE Computer Society.
10. Z. Manna and A. Pnueli. On the faithfulness of formal models. In *Mathematical Foundations of Computer Science*, volume 520 of *Lecture Notes in Computer Science*, pages 28–42. Springer-Verlag, 1991.
11. D. McAllester. Automatic recognition of tractability in inference relations. *Journal of the ACM*, 40(2):284–303, 1993.
12. M. Rusinowitch and M. Turuani. Protocol insecurity with finite number of sessions, composed keys is NP-complete. *Theoretical Comput. Sci.*, 299(1–3):451–475, 2003.
13. G. J. Simmons. Cryptoanalysis and protocol failure. *Commun. ACM*, 37(11):56–65, Nov. 1994.
14. SPORE: The Security Protocol Open Repository. Available at http://www.lsv.ens-cachan.fr/spore/.
15. M. Tatebayashi, N. Matsuzaki, and D. B. Newmann. Key distribution protocol for digital mobile communication systems. In G. Brassard, editor, *CRYPTO'89*, volume 435 of *Lecture Notes in Computer Science*, pages 324–334. Springer-Verlag, 1989.