Algebraic Intruder Deductions

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Technical Report No. 485, Rev. 2.0

Algebraic Intruder Deductions (Extended Version)*

David Basin, Sebastian Mödersheim, and Luca Viganò

Information Security Group, Dep. of Computer Science, ETH Zurich, Switzerland
www.infsec.ethz.ch/~{basin,moedersheim,vigano}

Abstract. Many security protocols fundamentally depend on the algebraic properties of cryptographic operators. It is however difficult to handle these properties when formally analyzing protocols, since basic problems like the equality of terms that represent cryptographic messages are undecidable, even for relatively simple algebraic theories. We present a framework for security protocol analysis that can handle algebraic properties of cryptographic operators in a uniform and modular way. Our framework is based on two ideas: the use of modular rewriting to formalize a generalized equational deduction problem for the Dolev-Yao intruder, and the introduction of two parameters that control the complexity of the equational unification problems that arise during protocol analysis by bounding the depth of message terms and the operations that the intruder can perform when analyzing messages. We motivate the different restrictions made in our model by highlighting different ways in which undecidability arises when incorporating algebraic properties of cryptographic operators into formal protocol analysis.

1 Introduction

Motivation. Many security protocols fundamentally depend on the algebraic properties of cryptographic operators [16]. For example, protocols based on the Diffie-Hellman key-exchange, such as the Station-to-Station, IKE, and JFK protocols, exploit the property of modular exponentiation that $(g^x)^y \bmod p = (g^y)^x \bmod p$. Without this property, these protocols could not even be executed.

A number of approaches have been proposed for formally analyzing security protocols in the presence of an active intruder. Independent of which formalism is adopted, one of the core problems is the *intruder deduction problem*: given a state

* This work was partially supported by the FET Open Project IST-2001-39252 and by the BBW Project 02.0431, "AVISPA: Automated Validation of Internet Security Protocols and Applications", as well as by the Zurich Information Security Center. This work represents the views of the authors.

of the protocol execution, can the intruder derive a given message M? Derivation here is relative to the terms the intruder currently knows, i.e. relative to the closure under a set of deduction rules of his initial knowledge augmented with the messages that he has observed. The intruder deduction problem provides the basis for solving a number of practically relevant protocol analysis problems. We can, for instance, use it to determine whether the intruder is able to construct a message of the form that some honest agent is expecting to receive, or whether he is able to obtain a message that is intended to be a secret, e.g. a key shared by two honest agents.

In this paper, we focus on the intruder deduction problem in the presence of algebraic equations that express properties of cryptographic operators. The underlying intruder model we employ is that of Dolev and Yao [18], in which the intruder observes all network traffic and can generate new messages, impersonating other agents, but cannot break cryptography. Although the Dolev-Yao intruder model is very commonly used, most analysis approaches based on this model are also based on the *free algebra assumption*. Under this assumption, two terms are equal if and only if they are syntactically equal. But, as we noted above, this is inappropriate for protocols that rely on algebraic properties.

Relaxing the free algebra assumption is however nontrivial: even for relatively simple sets of equations, the most basic problem, the *unifiability problem* (i.e. the equality of terms under substitutions for their variables), is only semi-decidable [4, 6, 22]. Moreover, even for those theories where unification is decidable, the intruder deduction problem may still be undecidable [1, 2].

Solutions for the intruder deduction problem have been given for individual algebraic theories of cryptographic operators, such as those formalizing different properties of modular exponentiation or bitwise exclusive or [11, 12, 26]. However, even though these approaches are specialized to particular algebraic properties, the algorithms and correctness proofs are quite complex and usually must be revised or completely re-designed when new properties are added. More general approaches have been recently proposed [13, 23, 25] and we compare our work with them in the concluding section §5.

Contributions. Our principal contribution in this paper is a framework for protocol analysis that is general and can handle algebraic properties of cryptographic operators in a uniform and modular way. In doing so, we pave the way for implementing analysis tools that are not specialized to particular algebraic theories and thereby allow users to declare new operators and properties as part of the protocol specifications. Of course, given the undecidability of the relevant problems, this goal cannot be achieved in full, without any restrictions. We now briefly describe the main ideas and restrictions of our proposed approach.

Our framework is based on two ideas. The first idea is to use *modular rewriting* to formalize a generalized equational deduction problem for the Dolev-Yao intruder. In doing so, we exploit the fact that we can distinguish two kinds of equational theories associated with security protocols: *cancellation theories* (where equations express that certain operations cancel each other out, such as encryption and decryption with the same symmetric key) and *finite equivalence*

class theories (which are theories that induce finite equivalence classes for all terms). We show how our use of modular rewriting leads to efficient solutions to the intruder deduction problem.

The second idea is to introduce two "depth parameters" that bound the depth of message terms and the operations that the intruder can use to analyze messages (i.e. decompose messages based on his current knowledge, under perfect cryptography). These bounds control the complexity of the equational unification problems that arise, transforming undecidable problems into decidable ones. Moreover, these bounds effectively serve as search parameters that can be used to control the search over the space of messages.

Our framework is thus parameterized by algebraic theories of the two kinds above and provides a general algorithm for the algebraic intruder deduction problem when the depth of message terms and the analysis operations of the intruder are bounded. Our framework allows us to identify several sub-problems of the intruder deduction problem (e.g. the reduction of terms to their normal forms) and provide general algorithms for them. Along the way, we also show that the problems considered become undecidable when any of the restrictions made in our framework are removed.

Two remarks are in order to help put into context our use of depth parameters. First, rather than considering specialized theories of algebraic properties of cryptographic operators, the focus of our work is to provide a general and flexible framework that supports a large class of such theories. However, in this generality, many problems are undecidable unless we introduce some restrictions. Our work shows that bounding the term depth and the message analysis by the intruder simplifies many of the problems that arise and turns undecidable problems into decidable ones. Moreover, many protocol analysis methods require bounds on messages in the first place, e.g. methods based on typed models.

Second, our algorithms are less efficient than those algorithms, when they exist, that are specialized to particular algebraic theories, e.g. [11, 12, 26], which usually work without bounds. Our framework is open to the integration of such specialized algorithms, albeit under the restriction of bounded message depth. In this way, we can benefit from research advances for specialized theories, while being able to fall back on general algorithms when specialized ones are not available.

Finally, we note that our framework is not biased towards a particular protocol analysis method. It can be used as a basis for handling algebraic equations when employing different types of formalisms (such as strand spaces, process calculi, or rewriting) or techniques (such as abstractions or the symbolic *lazy intruder* technique employed in our protocol model-checker OFMC [8, 9]).

Organization. We proceed as follows. In §2 we provide background for our approach. In §3 we introduce a concrete equational theory as a running example and give an overview of our framework, presenting the central definitions and theorems. In §4 we focus on how the intruder can analyze messages. In §5 we compare with related work and draw conclusions.

2 Background

Messages and cryptography. As is standard, we represent protocol messages as terms built over a finite signature Σ. We write Σ^n, for $n \geq 0$, to denote function symbols of arity n. Terms in Σ^0 are *constants* (i.e. nullary function symbols) and represent *atomic messages* like agent names or nonces. We define the *depth* of a term t as the number of nodes in the longest path from the root to a leaf in its tree representation, and the *size* of t as the number of nodes (both inner nodes and leaves). We write $\mathcal{T}(\Sigma, V)$ to denote the set of terms that can be generated using symbols of Σ and variables from a set V, and we write $\mathcal{T}(\Sigma)$ for the set of *ground* terms.

Algebraic properties of cryptographic operators. Most approaches to protocol analysis follow the *free algebra assumption*, under which two ground terms are equal iff they are syntactically equal. Many protocols, however, do actually depend on algebraic properties of cryptographic operators, in the sense that the properties are required for the agents to carry out the steps prescribed by their protocol roles. Hence, unlike the practice of abstracting from the concrete behavior of cryptography, we cannot ignore the algebraic properties on which the protocol to be analyzed is based. For example, as we noted above, protocols based on the Diffie-Hellman key-exchange, such as the Station-to-Station, IKE, and JFK protocols (see the web-page of the IETF [20]), exploit the property of modular exponentiation that $(g^x)^y \bmod p = (g^y)^x \bmod p$. As another example, note that many protocols combine two secrets into one using *associative* and *commutative (AC) operators* like *bitwise exclusive or (xor)* $\cdot \oplus \cdot$. Given such a composed secret, every agent who knows one of the two secrets can also find out the other one, but no other agent can. For instance, if an agent knows $x \oplus y$ and x, then he can exploit the properties of $\oplus$ to compute y as $(x \oplus y) \oplus x$.

Equational Theories. The formal analysis of protocols like those above requires explicitly reasoning about the relevant properties of the cryptographic operators employed. We address in this paper those properties that are formalizable by finite sets of equations of the form $t \approx s$, where $t, s \in \mathcal{T}(\Sigma, V)$. For example, the property required for the Diffie-Hellman key-exchange is that $exp(exp(g, x), y) \bmod p \approx exp(exp(g, y), x) \bmod p$.

We assume that notions like *substitution*, *matching*, *unification*, and *unifiability* are defined as standard, e.g. as in [4, 6]. *Term positions* are represented as sequences of natural numbers, which are partially ordered by the prefix ordering. We define the *equational theory* $\approx_E$ *induced by a set* E *of equations* to be the least congruence on the term algebra that is closed under substitution and contains E. We define the *equivalence class* $[t]_{\approx_E}$ of a term t as $\{s \mid t \approx_E s\}$. Given a set E of equations, we interpret terms of $\mathcal{T}(\Sigma, V)$ in the *quotient algebra* of the term algebra with the congruence on terms, written $\mathcal{T}(\Sigma, V)/_{\approx_E}$. In this algebra, two terms are equal iff they are equivalent due to $\approx_E$. The *ground word problem* for a theory E is the problem of deciding $s \approx_E t$ for arbitrary

$s, t \in \mathcal{T}(\Sigma)$. Note that, for brevity, we often refer to a set E of equations as a "theory", meaning the equational theory $\approx_E$ induced by E.

We say that a substitution σ is an *instance* of a substitution θ modulo E, and write $\sigma \succsim_E \theta$, iff there is a substitution λ such that $x\sigma \approx_E x\theta\lambda$ for all $x \in \mathit{domain}(\theta)$. Given a set $\mathcal{S}$ of substitutions, $\mathcal{S}_0$ is a *complete set of substitutions of* $\mathcal{S}$ *under* E iff for all $\sigma \in \mathcal{S}$ there is a $\theta \in \mathcal{S}_0$ with $\sigma \succsim_E \theta$.

Definition 1. *Let* vars(t) *denote the variables of a term* t*. A* rewrite rule *is an equation* $l \approx r$*, where* l *is not a variable and* vars$(l) \supseteq$ vars(r)*. In this case, we may write* $l \to r$ *instead of* $l \approx r$*. A* term-rewriting system (TRS) *is a set of rewrite rules. A TRS* C *and an equational theory* E *induce a* modular rewriting relation *on* E*-equivalence classes of terms as follows:* $[t]_{\approx_E} \to_{C/E} [s]_{\approx_E}$ *iff there are terms* t' *and* s' *such that* $t \approx_E t'$*,* $t' \to_C s'$*, and* $s' \approx_E s$*.*

Let $\to^+$ *and* $\to^*$ *denote the transitive and the transitive-reflexive closure of a binary relation* $\to$*. Given* $\to$*, we say that* t *is* reducible *(and we call* t *a* redex*) iff* $t \to s$ *for some* s*.* t_1 *and* t_2 *are* joinable*, denoted by* $t_1 \downarrow t_2$*, iff there is some* s *such that* $t_1 \to^* s$ *and* $t_2 \to^* s$*.* t *is a* normal form *iff it is not reducible, and* s *is* a normal form of t *iff* $t \to^* s$ *and* s *is a normal form. We denote the normal form of* t *by* $t{\downarrow}$*, when it is unique. We say that* $\to$ *is* confluent *iff* $t \to^* t_1$ *and* $t \to^* t_2$ *implies that* $t_1 \downarrow t_2$*. Finally,* $\to$ *is* convergent *iff it is confluent and terminating.*

Although $\to_{C/E}$ is defined on equivalence classes of terms, for notational simplicity we will also write $t \to_{C/E} s$, for terms s and t, rather than $[t]_{\approx_E} \to_{C/E} [s]_{\approx_E}$. Employing the same convention, we will also write $t{\downarrow_{C/E}}$ for $[t]_{\approx_E}{\downarrow_{C/E}}$. Note that for a convergent relation $\to$, every term has a unique normal form, and hence $t{\downarrow_{C/E}}$ is always defined.

The definition of modular rewriting works directly on E-equivalence classes, rather than defining a special notion of convergence modulo E. However, while theoretically appealing, this definition is algorithmically difficult to work with. Therefore many approaches to modular rewriting employ a weaker but more tractable variant $\to_{C,E}$ of the relation $\to_{C/E}$, namely $s \to_{C,E} t$ iff $\exists (u \to v) \in C.\ \exists \sigma.\ s \approx_E u\sigma \wedge t = v\sigma$. For $\to_{C,E}$, there is a completion method [7, 21], and it is not necessary to explore the entire E-equivalence class of a term t in order to determine if t is a redex. While we consider here the relation $\to_{C/E}$, we remark that all constructions and algorithms in this paper can be adapted to $\to_{C,E}$ as well.

A standard result tells us that we can solve the ground word problem for terms in the theory $C \cup E$ by normalizing the terms under C and checking the results for equality modulo E. Formally, if $\to_{C/E}$ is convergent and t_1 and t_2 are ground terms, then $t_1 \approx_{C \cup E} t_2$ iff $[t_1]_{\approx_E}{\downarrow_{C/E}} = [t_2]_{\approx_E}{\downarrow_{C/E}}$.

The Dolev-Yao intruder. The standard Dolev-Yao model [18] formalizes the abilities of an intruder who controls the communication network. The intruder can analyze messages, decomposing them into submessages, and synthesize new messages from their subparts. In our formalization of this, we assume we are

given a set of function symbols $\mathcal{O} \subset \Sigma$ that describe the ways of constructing messages (e.g. pairing or cryptographic operations like encryption or hashing). We also call the set $\mathcal{O}$ the set of *intruder-accessible operators*. For readability, we will however avoid displaying the set $\mathcal{O}$ as an explicit parameter of the intruder deduction problem.

Definition 2. *Given a finite set of ground terms IK (for "intruder knowledge") and an equational theory E, we define $\mathcal{DY}_E(IK)$ (for "Dolev-Yao") as the least set that is closed under the rules*

$$\frac{}{t \in \mathcal{DY}_E(IK)}\ \text{AX}\ (t \in IK)\,, \qquad \frac{t_1 \in \mathcal{DY}_E(IK)}{t_2 \in \mathcal{DY}_E(IK)}\ \text{EQ}\ (t_1 \approx_E t_2)\,,$$

$$\frac{t_1 \in \mathcal{DY}_E(IK) \quad \cdots \quad t_n \in \mathcal{DY}_E(IK)}{op(t_1, \ldots, t_n) \in \mathcal{DY}_E(IK)}\ \text{OP}\ (op \in \mathcal{O})\,.$$

The (Dolev-Yao) intruder deduction problem with respect to the equational theory *E is the problem of deciding whether $t \in \mathcal{DY}_E(IK)$ for ground terms t and finite sets of ground terms IK.*

Note that in this formalization we do not have analysis rules for decomposing terms. For example, the decryption rule for symmetric encryption

$$\frac{\{\!|m|\!\}_k \in \mathcal{DY}_E(IK) \quad k \in \mathcal{DY}_E(IK)}{m \in \mathcal{DY}_E(IK)}$$

is subsumed by the equation $\{\!|\{\!|m|\!\}_k|\!\}_k \approx m$: whenever the intruder has $\{\!|m|\!\}_k$ and k, he can compose them to construct $\{\!|\{\!|m|\!\}_k|\!\}_k$, which is equal under $\approx_E$ to m.

The intruder deduction problem is the core deduction problem in protocol analysis. Consider a trace of messages exchanged between honest agents and an intruder. For each message m that is sent by the intruder in this trace, the intruder must be able to derive m, i.e. $m \in \mathcal{DY}_E(IK)$, where E is the equational theory considered and IK is the intruder knowledge consisting of the initial intruder knowledge and all messages the intruder has observed so far. Note that in many state-of-the-art approaches to protocol analysis (see [14] for an overview), the term m may contain variables and the resulting symbolic trace represents the set of traces that are obtained by substituting for the variables arbitrary terms from $\mathcal{DY}_E(IK)$. The use of symbolic terms avoids the naïve enumeration of all terms that the intruder can generate from his knowledge.

3 A framework for algebraic properties

While equational reasoning is a general paradigm, our focus in this paper is on its application to security protocol analysis. Let us begin with a concrete example: an algebraic theory formalizing relevant properties used in many protocols, including those based on the Diffie-Hellman key-exchange.

Example 1. Let $\Sigma_{ex} = (\Sigma^0_{ex}, \Sigma^1_{ex}, \Sigma^2_{ex})$, where Σ^0_{ex} is a countable set of constants; $\Sigma^1_{ex} = \{inv(\cdot), \cdot^{-1}\}$, where $inv(t)$ and t^{-1} are the inverses of a message term t for asymmetric encryption and exponentiation, respectively, and the symbols in $\Sigma^2_{ex} = \{\{\cdot\}., \{|\cdot|\}., \langle\cdot,\cdot\rangle, exp(\cdot,\cdot), \cdot \oplus \cdot\}$ denote *asymmetric encryption* $\{t_2\}_{t_1}$ and *symmetric encryption* $\{|t_2|\}_{t_1}$ of a message t_2 with a message t_1, *concatenation* $\langle t_1,t_2\rangle$ of two messages t_1 and t_2, *modular exponentiation* $exp(t_1, t_2)$ of a message t_1 with a message t_2, and *bitwise xor* $t_1 \oplus t_2$ of a message t_1 with a message t_2 (with identity element e). Our example theory E_{ex} is induced by the following equations over Σ_{ex} (where the x_i are variables from a set disjoint from Σ_{ex}):

$$x_1 \oplus x_2 \approx x_2 \oplus x_1 \quad (1)$$
$$(x_1 \oplus x_2) \oplus x_3) \approx x_1 \oplus (x_2 \oplus x_3) \quad (2)$$
$$exp(exp(x_1, x_2), x_3) \approx exp(exp(x_1, x_3), x_2) \quad (3)$$
$$exp(exp(x_1, x_2), {x_2}^{-1}) \approx x_1 \quad (4)$$
$$inv(inv(x_1)) \approx x_1 \quad (5)$$
$$({x_1}^{-1})^{-1} \approx x_1 \quad (6)$$
$$\{\{x_2\}_{x_1}\}_{inv(x_1)} \approx x_2 \quad (7)$$
$$\{\{x_2\}_{inv(x_1)}\}_{x_1} \approx x_2 \quad (8)$$
$$\{|\{|x_2|\}_{x_1}|\}_{x_1} \approx x_2 \quad (9)$$
$$x_1 \oplus x_1 \approx \mathsf{e} \quad (10)$$
$$x_1 \oplus \mathsf{e} \approx x_1 \quad (11)$$

We split E_{ex} into two subtheories: F_{ex} is induced by the equations (1)–(3), and C_{ex} is induced by the equations (4)–(11). □

Note that, as is often done, we leave implicit the modulus of exponentiation in E_{ex}: instead of $g^x \bmod p$ (i.e. $exp(g, x) \bmod p$) we write simply g^x (i.e. $exp(g, x)$), assuming that exponentiation is always performed using the same (publicly known) modulus. Note also that E_{ex} does not contain redundant equations (which are entailed by the given equations) such as $\mathsf{e} \oplus x_1 \approx x_1$.

There is a subtlety about modeling exponentiation that is worth remarking on here. The modulus plays an important role in cryptographic algorithms like RSA, which are based on modular exponentiation. In the RSA algorithm, given a message x, the intruder in general should not be able to compute x^{-1} with $exp(exp(b, x), x^{-1}) = b$ modulo m, unless he knows the prime factors of m. Otherwise, if we gave the intruder the ability to compute x^{-1} without knowing the prime factors, then he can derive the private key for each public key he knows (which is clearly not what one wants to model). On the other hand, in the Diffie-Hellman key-exchange, we should assume that the intruder is always able to build x^{-1}, since the modulus m is a publicly known prime number in this case.

3.1 Two kinds of theories

Our framework is based on *modular rewriting* and exploits the fact that we can distinguish two kinds of equational theories associated with security protocols: cancellation theories and modulo theories. C_{ex} is an example of a *cancellation theory*, which is a theory whose equations express that certain operations (such as encryption followed by decryption with the same key) cancel each other out. Such equations can usually be described by a convergent TRS and we can thus

apply these equations to rewrite all terms into normal form. The advantage of separating out a convergent subtheory is that we can then neglect its equations during subsequent equality reasoning when all terms are normalized.

Definition 3. *A* cancellation theory *is a theory induced by* cancellation rules *of the form* $op(t_1, \ldots, t_n) \approx s$*, with* s *a constant or a subterm of one of the* t_i*.*

F_{ex} is an example of a *modulo theory*, which is a theory that comprises equations that cannot be oriented into terminating rewrite rules; the standard examples from rewriting are the equations for properties like associativity and/or commutativity. It is common for these equations to form a "background theory" used when applying other rewrite rules (such as the cancellation equations); that is, one performs rewriting modulo the equations of a modulo theory.

Here we will not restrict ourselves to a particular modulo theory, like AC, but rather work with a class of theories, namely *finite equivalence class theories*.

Definition 4. *An equational theory* E *is a* finite equivalence class (FEC) theory *if the equivalence class* $[t]_{\approx_E} = \{t' \mid t' \approx_E t\}$ *is finite for all terms* $t \in \mathcal{T}(\Sigma, V)$*.*

Note that for a cancellation theory, there are always terms with an infinite equivalence class (e.g. the terms on the right-hand side of equations). Thus, FEC and cancellation theories are disjoint theory classes.

In the following, we will use C and F to denote cancellation and FEC theories, respectively. We can prove that:

Lemma 1. *F_{ex} is an FEC theory and C_{ex} is a cancellation theory.*

Proof. Every rule in C_{ex} is a cancellation rule by definition. F_{ex} is an FEC theory since in a ground term, there are only finitely many ways to commute and associate subterms composed with $\oplus$ or *exp*. □

As is standard, the *equational matching problem* for a theory E is the question of whether, given a ground term t and a term s with variables, there is a substitution σ such that $t \approx_E s\sigma$. From the definition of FEC theories, we have:

Theorem 1. *The equational matching problem for an FEC theory F is decidable. In particular, there is a terminating algorithm that returns a complete set of matches modulo F for a given instance of the problem.*

Proof. It is quite straightforward to show that the following problem is decidable for every FEC theory F: Given two terms t_1 and t_2, where t_1 is a ground term, is there a substitution σ such that $t_1 \approx_F t_2\sigma$? This is the case if there is a term $t_1' \in [t_1]_{\approx_F}$ such that $t_1' = t_2\sigma$ for some substitution σ. Since F is an FEC theory, the equivalence class $[t_1]_{\approx_F}$ is finite by definition, and thus we have reduced the problem to (finitely many instances of) standard syntactic matching. □

A special case of equational matching is the ground word problem (when s is also ground), and hence this problem is also decidable for FEC theories.

As we will see below, our framework relies on the decidability of matching for FEC theories. In contrast, the unification problem (where both terms may

contain variables) for FEC theories is undecidable. Consider the theory of distributivity and associativity $D_{\star+}A_+ = \{x\star(y+z) \approx (x\star y)+(x\star z),\ x+(y+z) \approx (x+y)+z\}$. Unifiability in this theory is undecidable as shown in [27]. As equivalence classes in $D_{\star+}A_+$ are finite, we thus have that unifiability modulo an FEC theory is in general undecidable.

In §4 we will use the following important property of FEC theories, namely that they cannot contain equations that introduce new variables:

Lemma 2. *If $l \approx r$ is an equation of an FEC theory, then* $\mathrm{vars}(l) = \mathrm{vars}(r)$.

Hence, $l \in \mathcal{V}$ implies $l = r$, so that such trivial equations can be safely omitted.

Proof. Suppose, without loss of generality, that an FEC theory F contains an equation $(l \approx r)$ with a variable $v \in \mathrm{vars}(l) \setminus \mathrm{vars}(r)$, and let σ be a ground substitution for r. Then the ground term $r\sigma$ matches (by σ) the right-hand side r, and thus is equal, under F, to $l\sigma$ for any substitution of the remaining variables. As there is at least one variable v, and we can instantiate it with any term, the equivalence class of $r\sigma$ is infinite and hence F cannot be an FEC theory. □

We conclude this subsection by observing the relevance of these two kinds of theories to security protocol analysis. As we will see, cancellation rules are closely related to the analysis (e.g. decryption) of terms by the intruder and honest agents, and therefore have a distinguished role in deductions. We will namely define a normal form of the intruder knowledge as a state where the applications of cancellation rules do not give him any "new" terms (in a sense to be precisely defined later).

3.2 Restriction to a bounded variable depth model

As unifiability modulo an FEC theory is undecidable, we must introduce a restriction under which unification becomes decidable. We achieve this by introducing bounds on messages. There are several ways to do this, e.g. by bounding the number of operations that the intruder can perform to synthesize new messages from his knowledge, or by limiting the depth of terms that may be substituted for variables in the rules formalizing the steps of a protocol execution. We take the second approach here and bound the depth of message terms. To this end, we first define a subset of the variable symbols with an associated depth bound, and we then define which substitutions are permissible for these variables.

Definition 5. *We call a* bounded variable *a variable for which only terms with bounded depth can be substituted. Let $\mathcal{VB} \subseteq \mathcal{V}$ be the* set of bounded variables *such that every variable v has an associated depth bound $\mathit{depth}(v) \in \mathbb{N}$. We extend the function $\mathit{depth}(\cdot)$ to arbitrary terms as follows:*

$$\begin{aligned} \mathit{depth}(v) &= \infty && \text{for } v \in \mathcal{V} \setminus \mathcal{VB}\,, \\ \mathit{depth}(c) &= 0 && \text{for } c \in \Sigma^0, \\ \mathit{depth}(op(t_1, \ldots, t_n)) &= 1 + \mathit{max}_{i=1}^{n}\ \mathit{depth}(t_i) && \text{for } op \in \Sigma^n,\ \text{with } n > 0\,. \end{aligned}$$

We say that a substitution σ respects the depth restrictions of the variables in a term t*, and write* $respect\_depth(\sigma, t)$*, iff* $depth(v\sigma) \leq depth(v)$ *for all* $v \in \mathrm{vars}(t)$.

We call the *bounded variable depth model (BVDM)* the restricted protocol analysis model in which only substitutions are allowed that respect the depth of variables. For example, we may formulate the problem whether, given terms s and t with $\mathrm{vars}(s) \cup \mathrm{vars}(t) \subseteq \mathcal{VB}$, it holds that

$$\exists \sigma.\, respect\_depth(\langle s,t \rangle, \sigma) \wedge s\sigma \approx_E t\sigma \ .$$

The following lemma tells us that any computable function on ground terms can be extended to a computable function on terms with bounded variables. This will allow us, in the rest of this paper, to restrict ourselves to the ground case while all results can be carried over to terms with bounded variables.

Lemma 3. *Let f be a computable function that takes as input n terms that may contain variables and m ground terms, and which returns a finite set of terms. Then the following function f' is also computable. f' takes as input n terms that may contain (arbitrary) variables and m terms that may contain only bounded variables, and returns a finite set of terms and substitutions such that:*

$$\begin{aligned}
&\forall s_1, \ldots, s_n \in \mathcal{T}(\Sigma, V).\ \forall t_1, \ldots, t_m \in \mathcal{T}(\Sigma, \mathcal{VB}).\ \forall \sigma. \\
&\ [ground(t_1\sigma) \wedge \ldots \wedge\ ground(t_m\sigma) \wedge\ domain(\sigma) \subseteq \mathcal{VB} \wedge \\
&\qquad respect_depth(\langle s_1, \ldots, s_n, t_1, \ldots, t_m \rangle, \sigma)] \implies \\
&\ [(r, \sigma) \in f'(s_1, \ldots, s_n, t_1, \ldots, t_m) \iff r\sigma \in f(s_1\sigma, \ldots, s_n\sigma, t_1\sigma, \ldots, t_m\sigma)] \, .
\end{aligned}$$

Proof. Observe that, since we have assumed a finite signature, there are only finitely many ground terms of a given depth. Thus, the set of admissible substitutions for a bounded term is finite. Now, it is possible to construct f' given f as in the assumptions of this theorem:

- Given $t_1, \ldots, t_m$ with only bounded variables, compute the finite set Θ of admissible substitutions for the t_i.
- For every $\sigma \in \Theta$, compute $f(s_1\sigma, \ldots, s_n\sigma, t_1\sigma, \ldots, t_m\sigma)$, which yields a finite set R. For each $r \in R$, the result of $f'(s_1, \ldots, s_n, t_1, \ldots, t_m)$ contains the tuple (r, σ). □

Lemma 3 allows us, for instance, to easily lift the matching algorithm for FEC theories F to a unification algorithm where one of the two input terms contains only bounded variables.

Lemma 4. *For an FEC theory F, the* one-side-bounded unifiability problem *is decidable, i.e. it is decidable whether, given terms s and t with* $\mathrm{vars}(t) \subseteq \mathcal{VB}$*, there is a substitution σ such that $respect\_depth(\langle s, t \rangle, \sigma)$ and $s\sigma \approx_F t\sigma$. Moreover, there is a complete one-side-bounded F-unification algorithm, i.e. given terms s and t with* $\mathrm{vars}(t) \subseteq \mathcal{VB}$*, the algorithm returns a set S of substitutions such that for every substitution σ with $s\sigma \approx_F t\sigma$ and $respect\_depth(\langle s, t \rangle, \sigma)$, there is a $\tau \in S$ with $\sigma \gtrsim_F \tau$.* □

Note that the depth of messages is often bounded in protocol analysis. For instance, many model-checking approaches bound terms to obtain a finite-state system, e.g. [3, 24]. Moreover, when other parameters of the model are unbounded, like the number of sessions, then restricting the message depth is essential for decidability [19]. Note also that [10] presents an approach that similarly bounds the depth of message terms in order to tackle the problem of algebraic properties in intruder deductions; the approach of [10] is however specialized to a particular algebraic theory.

3.3 Matching and unification in FEC theories in the BVDM

We have shown that for every FEC theory F, we can decide the matching problem. By Lemma 3, when the variables are bounded on one side, we can reduce an F-unification problem to a finite number of F-matching problems, which we can solve by Theorem 1. The algorithms that we can obtain from the constructive proof of Theorem 1 however have poor complexity. Moreover, there exist more efficient, specialized algorithms for some of the theories that are relevant for the analysis of security protocols, e.g. [11, 12, 26].

We give a solution to handle F-unification efficiently in the bounded case and which allows for the straightforward integration of existing unification algorithms for disjoint subtheories of F. Before we present the idea, however, we first want to discuss why standard results for combining unification procedures for disjoint theories such as [5] cannot be applied in our setting. In a nutshell, the idea of [5], based on the classical Nelson-Oppen combination method, is to transform a given unification problem into a *pure* one, i.e. where each equation to unify contains only symbols from *one* of the disjoint subtheories. For instance, in our example theory E_{ex}, we may transform the given unification problem

$$exp(g, x_1 \oplus x_2) \approx exp(x_3, x_4)$$

into the problems

$$exp(g, x_5) \approx exp(x_3, x_4) \quad \text{and} \quad x_5 \approx x_1 \oplus x_2 \ .$$

Then, each equation can be unified using the unification algorithm for the respective subtheory. The problem in our setting is that, in general, we can only support unification where the variables on one side of each equation are bounded. Suppose that in this example the variables x_1 and x_2 are unbounded, while x_3 and x_4 are bounded. Then the newly introduced variable x_5 cannot be bounded, and thus we have a unification problem with unbounded variables. (Note that a similar problem would occur if we considered only matching, e.g. if we had ground terms instead of x_3 and x_4.) It is thus not possible to integrate such an idea into our framework without full F-unification (which would lead to undecidable problems).[1]

[1] As we later want to reduce ground terms (or terms with bounded variables) according to $\rightarrow_{C/F}$ for a cancellation theory C and FEC theory F, we must at least be able to decide F-matchability (or F-unifiability where variables on one side are bounded).

The idea we present now is based on a subproblem of matching/unification that gives us a different way to address the problem of integrating algorithms for disjoint subtheories.

The subproblem that we consider is one that has previously been considered in unification algorithms for AC-theories: given a term t, an n-ary operator symbol op, and fresh variables $x_1, \ldots, x_n$, determine the unifiers σ such that $t\sigma \approx_E op(x_1, \ldots, x_n)\sigma$. We call such a unifier a *toplevel decomposition* of t for the operator op with respect to fresh variables $x_1, \ldots, x_n$. In the following, we will consider, for a given toplevel decomposition problem, a subset of the substitutions that represent the possible decompositions and is complete (in the sense defined in §2). We will write CSTD for the *complete set of toplevel decompositions*.

As a simple example, in the free term algebra every term has a singleton CSTD, namely its syntactic decomposition. A more interesting example is the term $x \oplus c$ (for a variable x and a constant c) in our theory F_{ex}. It has the following CSTD for the operator $\oplus$ and fresh variables x_1 and x_2: first, we have the "syntactic" solution $[x_1 \mapsto x, x_2 \mapsto c]$; second, unless the variable x is bounded to depth 0, we have the solution $[x \mapsto x_3 \oplus x_4, x_1 \mapsto x_3, x_2 \mapsto x_4 \oplus c]$ for fresh variables x_3 and x_4; and finally, for each solution we also have its commutation (swapping x_1 and x_2). All other toplevel decompositions are instances of one of these solutions.

The CSTD gives us a notion of all the ways to decompose a term into an operator and its arguments, modulo equivalent representations of the argument terms. Observe that this is a subproblem of the intruder deduction problem, as we must check how the intruder can compose a term from known subterms. Moreover, CSTD is obviously a subproblem of the E-unification problem. We will now show, conversely, that we can obtain a CSTD-based algorithm for the F-unification problem for terms with bounded variables on one side. Note that for every term with only bounded variables, there is a finite CSTD and it can be computed by a terminating algorithm.

Consider an F-unification problem $\{s \approx_F t\} \cup Eq$ where one side of each equation contains only bounded variables, say s for the first equation. We consider the following two cases: First, if t is a variable. If t occurs in s, but $t \neq s$, then the unification fails.[2] Otherwise, we have the partial unifier $[t \mapsto s]$ and continue with the problem $Eq[t \mapsto s]$. The second case is $t \notin \mathcal{V}$, i.e. $t = op(t_1, \ldots, t_n)$. In this case, we compute the CSTD of s for operator op and fresh variables $z_1, \ldots, z_n$. If the CSTD is $\{\tau_1, \ldots, \tau_m\}$, then we continue recursively with the following problems (for $1 \leq i \leq m$):

$$\{z_1\tau_i \approx t_1\tau_i, \ldots, z_n\tau_i \approx t_n\tau_i\} \cup Eq\tau_i \quad .$$

Observe that in every recursion step, the $z_j\tau_i$ (for $1 \leq j \leq n$) are terms with only bounded variables since the term s is bounded. The termination now follows

[2] Assume there is a unifier σ, then $s\sigma \approx_F t\sigma$ where $t\sigma$ is a proper subterm of $s\sigma$. Thus the F-equivalence class of $t\sigma$ is infinite, although F is FEC.

from the assumption that all variables on one side are bounded (in our notation, the variables on the left-hand sides of the equations).

Let us illustrate with an example that in the BVDM this algorithm for the F-unification problem based on CSTDs is in fact an improvement over the naïve algorithm for F-unification based on exploring the F-equivalence class of one of the terms. Consider the unification problem $\{s \approx t\}$ in F_{ex} where $t = t_1 \oplus \ldots \oplus t_n$ and $s = s_1 \oplus \ldots \oplus s_n$. The naïve algorithm explores the entire equivalence class of either s or t, say s, without loss of generality. The equivalence class of s contains, for instance, all associations of permutations of the s_i. Then, the algorithm will syntactically unify each $s' \in [s]_{F_{ex}}$ with t. The algorithm based on CSTDs, however, will explore only the ways to decompose one of the terms, say s, into an operator and subterms. In this case, this can only be $s'_1 \oplus s'_2$ where s'_1 and s'_2 are a partition of the s_i into two parts. This algorithm is still exponential, but it saves us from also exploring the equivalence classes of the subterms s'_1 and s'_2.

With the notion of CSTD as a subproblem of unification in the BVDM, we not only have a more efficient algorithm for the general case, but we also have a basis for combining existing unification algorithms, e.g. combining a specialized algorithm for a particular subtheory (like exponentiation) with the general algorithm sketched just above.

As is standard, let us say that two equational theories E_1 and E_2 are *disjoint* if they are induced by equations over disjoint signatures Σ_1 and Σ_2.

We observe that for disjoint FEC theories we have the following property:

Lemma 5. *Let F_1 and F_2 be two disjoint FEC theories over signatures Σ_1 and Σ_2. If $op(t_1, \ldots, t_n) \approx_{F_1 \cup F_2} op'(s_1, \ldots, s_m)$ then either $op, op' \in \Sigma_1$ or $op, op' \in \Sigma_2$.*

Proof. Suppose that $op \in \Sigma_1$ (the proof for the case $op \in \Sigma_2$ is analogous). By Lemma 2, both sides of equations in FEC theories must have an operator symbol (not a variable symbol) as root. Therefore, any application of a rule of F_1 and F_2 is either to a subterm (thus leaving the top symbol op) or it can only be from F_1 and therefore the new top symbol is again from Σ_1. Therefore, after any number of rule applications, the toplevel symbol is still from Σ_1, thus $op' \in \Sigma_1$. □

The key property of disjoint FEC theories F_1 and F_2 is the following:

Theorem 2. *Let F_1 and F_2 be two disjoint FEC theories over signatures Σ_1 and Σ_2, and let $op \in \Sigma_1$. Then*

$$op(t_1, \ldots, t_n) \approx_{F_1 \cup F_2} op'(s_1, \ldots, s_m)$$

iff there are terms $s'_1, \ldots, s'_m$ such that

$$op(t_1, \ldots, t_n) \approx_{F_1} op'(s'_1, \ldots, s'_m)$$

and $s_i \approx_{F_1 \cup F_2} s'_i$ for all i with $1 \leq i \leq m$.

Proof. The direction "left to right" of the "iff" is straightforward.

For the converse direction, let us define the notion of the Σ_1-*threshold* of a term t, $\mathrm{thr}_{\Sigma_1}(t)$, which is the set of positions in the term such that up to that position only symbols of Σ_1 (and variable symbols) occur:

$$\mathrm{thr}_{\Sigma_1}(t) = \begin{cases} \{\epsilon\} \cup \bigcup_{i=1}^{n} \bigcup_{p \in \mathrm{thr}_{\Sigma_1}(t_i)} i \cdot p & \text{if } t = op(t_1, \ldots, t_n) \wedge op \in \Sigma_1 \ , \\ \emptyset & \text{if } t = op(t_1, \ldots, t_n) \wedge op \notin \Sigma_1 \ , \\ \{\epsilon\} & \text{if } t \in \mathcal{V} \ . \end{cases}$$

Moreover let

$$\mathrm{below}_{\Sigma_1}(t) = \{t_{|p} \mid p \in pos(t) \wedge p \notin \mathrm{thr}_{\Sigma_1}(t)\}$$

be the set of all subterms of t at positions below the Σ_1-threshold.

Now let $P = \mathrm{thr}_{\Sigma_1}(op(t_1, \ldots, t_n))$ and let a proof for $op(t_1, \ldots, t_n) \approx_{F_1 \cup F_2} op'(s_1, \ldots, s_n)$ be given by a sequence of terms $u_1, \ldots, u_k$, where each term can be obtained from the previous term by one application of a rule in F_1 or in F_2, and $u_1 = op(t_1, \ldots, t_n)$ and $u_k = op(s_1, \ldots, s_n)$. Also, let p_i $(1 \leq i < k)$ be the position in the term u_i at which the rule is applied to produce u_{i+1}. We note that we have here a *universal word-problem*, i.e. the terms may contain variables, and we have to show their equivalence independent of the variables; in particular, we cannot instantiate the variables during the proof of $\approx_{F_1 \cup F_2}$, since the equivalence must hold under any instantiation.

The idea is now to split the proof into two parts: in the first part, rules are applied only above the Σ_1-threshold of the respective terms (and we will show that these rules must be from F_1), and in the second part, rules are applied only below the threshold (but may be from both F_1 and F_2).

First, we show that for every prefix $u_1, \ldots, u_l$, with $l < k$, of the proof sequence such that $p_i \in \mathrm{thr}_{\Sigma_1}(u_i)$, with $1 \leq i \leq l$, it holds that the applied rule is from F_1 and that $\mathrm{below}_{\Sigma_1}(u_i) = \mathrm{below}_{\Sigma_1}(u_{i+1})$. The latter equality means that the set of subterms below the threshold does not change (although the number of occurrences of identical subterms might change).

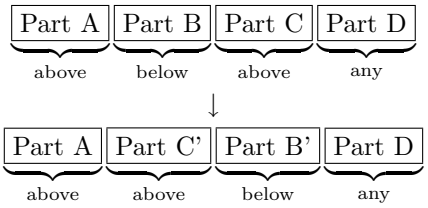
We show this by induction over l, i.e. assuming that the conditions hold already for every prefix up to $l - 1$, and $p_l \in \mathrm{thr}_{\Sigma_1}(u_l)$.

By Lemma 2, the applied rule must have a top-level symbol from the respective Σ_i. The case $u_{l|_{p_l}} \in \mathcal{V}$ is excluded, as in this case the rule can only be applied when instantiating a variable of u_l (which is impossible as noted above). Therefore $u_{l|_{p_l}}$ is a term that is not a variable; the root symbol must be from Σ_1 as otherwise p_l does not belong to the Σ_1-threshold of u_l. Therefore the top-level symbol of the applied rule must be from Σ_1, i.e. the step is from theory F_1. It remains to show that the new term u_{l+1} has the same set $\mathrm{below}_{\Sigma_1}(u_{l+1})$ as u_l. Let $lhs \approx rhs \in F_1$ be the rule that transforms u_l into u_{l+1}, i.e. there is a substitution σ such that $lhs\sigma = u_{l|_{p_l}}$ and $rhs\sigma = u_{l+1|_{p_l}}$ and $u_l[p_l]rhs\sigma = u_{l+1}$. Suppose there is a $w \in \mathrm{below}_{\Sigma_1}(u_l) \setminus \mathrm{below}_{\Sigma_1}(u_{l+1})$ (the the case $w \in \mathrm{below}_{\Sigma_1}(u_{l+1}) \setminus \mathrm{below}_{\Sigma_1}(u_l)$ is analogous). Since the only change is below position p_l, it must hold that $w \in \mathrm{below}_{\Sigma_1}(u_{l|_{p_l}})$. Let p be the position of w in $u_{l|_{p_l}}$. It holds that w is below the threshold, and that lhs cannot contain

symbols from Σ_2, and that $lhs\sigma$ matches $u_{l|_{p_l}}$; therefore it follows that there is a prefix p' of p (not necessarily a proper one) such that $lhs_{|p'}$ is a variable. By Lemma 5, this variable also occurs in rhs, thus $rhs\sigma$ has w as a subterm, which contradicts that w does not occur in u_{l+1}.

Up to here we have established that an initial part of the equivalence proof that applies rules above the threshold can only apply F_1 rules and does not change the set of subterms below the threshold.

Next we show that if the equivalence proof contains an intermediate sequence of rule applications below the threshold, we can permute the sequence so that all rule applications below the threshold are performed at the end of the proof (after all above-threshold applications). This can be illustrated as follows:



Note that there might be changes in the part B in this transformation (thus we wrote B'), as the number of occurrences of some identical subterms may change and therefore the number of repetitions (at different positions) of certain subparts of the equivalence proof may change. Moreover, the subterms of part C may be different and thus we wrote C'.

Repeated application of this transformation results in a proof that contains first a sequence of applications above the threshold (that can only contain F_1 rule applications due to what we have shown so far) and a part that contains only rules below the threshold.

First, we show that the steps of part C can precede the steps of part B. The reason is that part B can change only subterms below the threshold: by induction we can show that each rule of part C is still applicable since the changes are at positions that match the variables of the rule. Since we have shown that the above threshold application of rules doesn't change the set of subterms below the threshold, each subterm affected by part B still exists (though possibly a different number of times) and thus the respective transformations can still be done.

With this transformation, we can conclude the proof, as we have now a sequence of steps from u_1 to some term u_{ll} that contains only above-threshold applications, and therefore only rule applications from only F_1, i.e. we have shown $u_1 \approx_{F_1} u_{ll}$ and u_{ll} thus has the form $op'(s'_1, \ldots, s'_m)$ for some $op' \in \Sigma_1$ and suitable s'_i. Moreover, we have a proof with only below threshold applications of $u_{ll} \approx_{F_1 \cup F_2} u_k$. Since all applications are below the threshold, they cannot be toplevel applications (as $op' \in \Sigma_1$) and therefore we have proofs that $s'_i \approx_{F_1 \cup F_2} s_i$ for any $1 \leq i \leq m$. □

This theorem is the key for integrating specialized matching or unification algorithms into the general CSTD-based unification algorithm we have sketched

above. Namely, according to this theorem, to compute CSTD of a term t for decomposition with the operator op, we can use the unification algorithm for the subtheory to which op belongs, for the following reason. In the case that $t = op'(\ldots)$ for an operator op' that belongs to a different subtheory, there cannot be a unifier. In the case that op' belongs to the same subtheory, Theorem 2 tells us that we can decompose t modulo the subtheory of op and op' and then to F-unify the subterms.

3.4 Intruder deduction modulo F

So far we have considered the problem of unification and matching modulo an FEC theory F. We now turn to the intruder deduction problem modulo F, i.e. whether $t \in \mathcal{DY}_F(IK)$ holds for a ground term t and a set of ground terms IK.

Lemma 6. *If F is an FEC theory, then the problem $t \in \mathcal{DY}_F(IK)$ is decidable for a term t and a set of terms IK.*

To prove this, we first show the following lemma which decouples consequences of F (or of a subtheory of F) from derivations of the intruder:

Lemma 7. *For any theory $E = E_1 \cup E_2$, application of equivalences due to E_1 can be split from the rest of the derivation: $t \in \mathcal{DY}_E(IK)$ iff $t \in [\mathcal{DY}_{E_1}(IK)]_{\approx_E}$.*

Proof. In a $\mathcal{DY}_E$-proof, we can move all E_1-equivalence steps down over all OP steps:

$$\dfrac{\dfrac{t_0 \approx_{E_1} t_1 \quad t_0}{t_1}\, E_1}{f(t_1)}\, OP \quad \rightarrow \quad \dfrac{f(t_0) \approx_{E_1} f(t_1) \quad \dfrac{t_0}{f(t_0)}\, OP}{f(t_1)}\, E_1$$

This is because $\approx_E$ is a congruence relation. As a consequence, every proof can be transformed into one where the OP rule is only applied to terms/subproofs that are free from equivalences.

Proof of Lemma 6. By Lemma 7, $t \in \mathcal{DY}_F(IK)$ iff there exists $t' \approx_F t$, and $t' \in \mathcal{DY}_\emptyset(IK)$. The latter problem (i.e. derivation in the free algebra) is easy for ground terms: $t' \in \mathcal{DY}_\emptyset(IK)$ iff $t' \in IK$ or $t' = op(t_1, \ldots, t_n)$ and $t_i \in \mathcal{DY}_\emptyset(IK)$ for all $1 \leq i \leq n$. Since $[t]_{\approx_F}$ is finite, there are finitely many terms t' for which this needs to be checked. □

In the following, we will consider the generalization of the problem $t \in \mathcal{DY}_F(IK)$, where the term t may contain variables. This is an important question even for a model with only ground terms, since we will later consider intruder derivations modulo $F \cup C$. In particular, given a set IK of ground terms, we must decide whether there is some ground instance $t\sigma$ of the left-hand-side t of a cancellation rule of C such that $t\sigma$ can be derived modulo F from IK (note that t is here a term with unbounded variables).

Lemma 8. *There is an FEC theory F such that it is undecidable for a term t and a set of ground terms IK, whether there exists a substitution σ such that $t\sigma$ is ground and $t\sigma \in \mathcal{DY}_F(IK)$.*

Proof. We define an FEC theory F_{PCP} and show that the PCP problem can be reduced to derivability modulo F_{PCP} of terms with variables; the undecidability of this problem follows then from the undecidability of the PCP problem. In F_{PCP} occur the function symbols $\langle \cdot, \cdot \rangle$ (concatenation with associativity), $\langle\!| \cdot, \cdot |\!\rangle$ (concatenation without associativity), as well as $res(\cdot,\cdot,\cdot)$ and $initres(\cdot)$, which will be used to keep track of partial solutions of the PCP problem:

$$\begin{gathered}
\langle A,\langle B,C\rangle\rangle \approx \langle\langle A,B\rangle,C\rangle \\
A.B \approx B.A \\
A.(B.C) \approx (A.B).C \\
\langle\!| Ident, res(X,Y,\langle\!| Ident,\langle\!| X_i,Y_i|\!\rangle|\!\rangle.S)|\!\rangle \approx res(\langle X_i,X\rangle,\langle Y_i,Y\rangle,\langle\!| Ident,\langle\!| X_i,Y_i|\!\rangle|\!\rangle.S) \\
\langle\!| Ident, initres(\langle\!| Ident,\langle\!| X_i,Y_i|\!\rangle|\!\rangle.S)|\!\rangle \approx res(X_i,Y_i,\langle\!| Ident,\langle\!| X_i,Y_i|\!\rangle|\!\rangle.S)
\end{gathered}$$

For the encoding of identifiers without bounds, we additionally need the constant 0 and the function symbol s (for successor). For simplicity, we will in the following refer to identifiers as $1,\ldots,n$ rather than their actually encoding as terms, namely $s(0),\ldots,s^n(0)$. Similarly, if the underlying alphabet of the PCP is $\{c_1,\ldots,c_n\}$, then the term encoding is also $s(0),\ldots,s^n(0)$, i.e. we don't use disjoint subsets of $\mathcal{T}(\Sigma)$ for identifiers and characters of the PCP problem. We define the intruder-accessible operators as $\mathcal{O} = \{s(\cdot), \langle\cdot,\cdot\rangle, \langle\!|\cdot,\cdot|\!\rangle\}$.

For a given instance $P = \{(x_1,y_1),\ldots,(x_n,y_n)\}$ of the PCP problem, we define the function $\mathsf{symbtab}_P$ as:

$$\begin{gathered}
\mathsf{symbtab}_P = \langle\!| 1,\langle\!| < x_1 > , < y_1 > |\!\rangle|\!\rangle. \ \ \ldots \ \ .\langle\!| n,\langle\!| < x_n > , < y_n > |\!\rangle|\!\rangle \\
< s > = \langle c_1,\langle \ldots,c_n\rangle\rangle \text{ for a string } s \text{ of characters } c_1,\ldots,c_n.
\end{gathered}$$

We first observe that $\mathsf{symbtab}_P \approx_{Fpcp} \langle\!| i,\langle\!| x,y|\!\rangle|\!\rangle.s$ iff $x = x_i$ and $y = y_i$, i.e. we can use the symbol-table to check whether a certain pair of strings belongs to the PCP problem, and which index it has.

Let now $IK_P = \{initres(symbtab_P), 1,\ldots,m\}$ for a given PCP P where $\{1,\ldots,m\}$ is the set of constants that identify each pair of strings of P.

An important property of our construction is the following:

$$res(X,Y,symbtab_P) \in \mathcal{DY}_{Fpcp}(IK_P)$$

holds for suitable terms X and Y iff there exist indices $i_1,\ldots,i_k$, $k > 0$ such that

$$X \approx_{Fpcp} \langle x_{i_1},\langle \ldots,x_{i_k}\rangle\rangle \text{ and } Y \approx_{Fpcp} \langle y_{i_1},\langle \ldots,y_{i_k}\rangle\rangle \ .$$

In other words, the intruder can derive $res(X,Y,symtab_P)$ iff X and Y are the corresponding strings of a certain sequence of indices. It is immediate that for any sequence $i_1,\ldots,i_k$, $k > 0$, the intruder can obtain

$$res(\langle x_{i_1},\langle \ldots,x_{i_k}\rangle\rangle,\langle y_{i_1},\langle \ldots,y_{i_k}\rangle\rangle,symtab_P) \ .$$

To show the converse direction of the property, first observe that the intruder does not initially know any term $res(\cdot,\cdot,\cdot)$ and this symbol is also not accessible to him. Thus, the only way to generate a *res* term is via equivalence in F. The statement can be proved as an invariant of the equations that contain the symbol *res*.

Finally, let $t = res(Z, Z, S)$, then with the property that we have just proved we have that $t \in \mathcal{DY}_{F_{PCP}}(IK_P)$ iff there is a sequence of indices $i_1, \ldots, i_k$ (for some $k > 0$), such that both $Z \approx_{Fpcp} \langle x_{i_1}, \langle \ldots, x_{i_k} \rangle \rangle$ and $Z \approx_{Fpcp} \langle y_{i_1}, \langle \ldots, y_{i_k} \rangle \rangle$, which is the case iff the PCP P has a solution. □

Hence, to decide the intruder deduction problem for terms with variables, we must make further restrictions. By Lemma 3, the problem is decidable if t contains only bounded variables.

4 Cancellation equations

We now turn to the cancellation equations such as $\{|\{|x_2|\}_{x_1}|\}_{x_1} \approx x_2$. Such an equation cannot be formalized as part of an FEC theory like F_{ex} since all equivalence classes are infinite. As introduced in §2, we will now consider rewriting for cancellation theories C modulo an FEC theory F. Note that every cancellation theory is a rewrite theory as every cancellation equation $l \approx r$ has the property that $\text{vars}(l) \supseteq \text{vars}(r)$.

The principal property that we require is that the modular rewriting relation $\rightarrow_{C/F}$ is convergent. The following subsection addresses the question of convergence.

4.1 Checking Convergence

In this subsection, we will sketch ways to check that $\rightarrow_{C/F}$ is indeed convergent. This task is particularly difficult in our case, as the standard method to show (local) confluence, the *critical pairs method*, cannot be applied for modular rewriting.

Note that it is not part of our framework to check whether two given theories F and C have the properties we require, i.e. whether F is an FEC theory, C is a cancellation theory, and $\rightarrow_{C/F}$ is convergent. When using our framework on theories that do not satisfy these requirements, the properties established in the lemmata and theorems of this paper do not necessarily hold anymore. In particular, the unification algorithm for $F \cup C$ may be non-terminating and miss unifiers.

Termination. Given an FEC theory F and a cancellation theory C, then $\rightarrow_{C/F}$ is not necessarily terminating as the following example shows:

$$F = \{1 \approx s(0)\} \text{ and } C = \{s(0) \rightarrow 1\} .$$

(We write the equation of C as a rule, because it is handled as a rewrite rule, i.e. with a "direction".)

The general method to prove termination of a (modular) rewrite relation $\to$ is to define a metric $\mathcal{M}$ on terms (or on E-equivalence classes of terms in the case of rewriting modulo E) and prove that $s \to t$ implies $\mathcal{M}(s) > \mathcal{M}(t)$. This excludes infinite chains of rewrite steps

$$s_1 \to s_2 \to s_3 \to \dots$$

as the metric is always positive and strictly decreases in every step.

When considering an FEC theory F and a cancellation theory C, then usually an appropriate metric for proving the termination of $\to_{C/F}$ is based on the size of terms (i.e. the number of nodes in their tree representation).[3] The reason why this metric is often appropriate in our framework is that typically F-equivalent terms have the same size, and typically applying $\to_C$ strictly reduces the size of terms.

Consider the example theories F_{ex} and C_{ex}. It is straightforward that for every ground instance of equations in F_{ex}, the left-hand side and right-hand side have the same size, and therefore, by induction, F_{ex}-equivalent terms have the same size. Similarly, for every ground substitution of the variables of the equations in C_{ex}, the left-hand side has a strictly larger size than the right-hand side, thus $\to_{C_{ex}}$ strictly reduces the size of every term it is applied to.

Assume now that $[t_1]_{\approx_F} \to_{C/F} [t_2]_{\approx_F}$ and F-equivalent terms have the same size and $\to_C$ reduces the size of terms. By definition, there exist terms $t_1' \approx_F t_1$ and $t_2' \approx_F t_2$ such that $t_1' \to_C t_2'$. Then we have $\mathsf{size}(t_1) = \mathsf{size}(t_1') > \mathsf{size}(t_2') = \mathsf{size}(t_2)$, and therefore $\mathsf{size}([t_1]_{\approx_F}) > \mathsf{size}([t_2]_{\approx_F})$.

For our example theories, it thus follows that $\to_{C_{ex}/F_{ex}}$ is terminating.

Local Confluence. It is standard to first show a weaker property than confluence, namely *local confluence*. Then, one can use the fact that for a terminating relation $\to$, local confluence implies confluence and thus convergence:

Definition 6. *A relation $\to$ is called* locally confluent *if from $t \to t_1$ and $t \to t_2$ it follows that $t_1 \downarrow t_2$.*

Lemma 9 (Newman's Lemma). *A terminating relation is confluent if it is locally confluent.*

Proof. See, for instance, [4]. □

For non-modular rewriting, the standard method to establish local confluence is to check its *critical pairs*:

Definition 7. *Let C be a rewrite theory. Let $(l_1 \to r_1), (l_2 \to r_2) \in C$ be two (not necessarily different) rewrite rules that are renamed so that their variables are disjoint. Let p be a non-variable position in l_1, and let θ be a substitution with $l_{1|p}\theta = l_2\theta$. Then the pair of terms $(r_1\theta, l_1\theta[r_2\theta]p)$ is called a* critical pair of C.

[3] Recall that $\to_{C/F}$ is defined on F-equivalence classes of ground terms; the size-metric is hence defined on F-equivalence classes of ground terms to be the size of the largest term in the class.

Lemma 10. *If C is a rewrite theory such that all critical pairs (t_1, t_2) of C are joinable, $t_1 \downarrow_C t_2$, then $\rightarrow_C$ is locally confluent.*

Proof. See, for instance, [4]. □

Observe that we can split our example theory E_{ex} into disjoint theories as follows:

Subtheory	Equations	Occurring symbols
E_{xor}	(1), (2), (10), (11)	$\oplus$, $\mathtt{e}$
E_{exp}	(3), (4), (6)	exp, $\cdot^{-1}$
E_{crypt}	(5), (7), (8)	$\{\cdot\}_\cdot$, $inv(\cdot)$
E_{scrypt}	(9)	$\{\!\lvert \cdot \rvert\!\}_\cdot$

Note that the theories E_{crypt} and E_{scrypt} consist of only cancellation equations, i.e. $E_{crypt} \cap F_{ex} = \emptyset$ and $E_{scrypt} \cap F_{ex} = \emptyset$. Thus, each of them alone forms a non-modular rewriting theory, and we can prove the local confluence (and therefore also the convergence) for both subtheories with the standard critical pairs method:

Lemma 11. *$\rightarrow_{E_{crypt}}$ and $\rightarrow_{E_{scrypt}}$ are convergent.*

Proof. Build all (most general) critical pairs for these theories and check that they are joinable; then Lemma 9 and termination of $\rightarrow_{C_{ex}/F_{ex}}$ (which is a superset of $\rightarrow_{E_{crypt}}$ and $\rightarrow_{E_{scrypt}}$) imply the convergence of these two relations. □

We will now proceed to prove the convergence of the modular rewrite relations that the other subtheories induce. Then we can obtain the convergence of $\rightarrow_{C_{ex}/F_{ex}}$ by showing that convergence of disjoint subtheories implies convergence of their union.

Lemma 12. *$\rightarrow_{C_{xor}/F_{xor}}$ and $\rightarrow_{C_{exp}/F_{exp}}$ are convergent.*

Note that this lemma cannot be established by using the critical pairs method as this method is built on the assumption that for every "conflict" between two possible rule applications, it holds that one of the positions where the rules are applied is a sub-position of the other. For modular rewriting this is not true. Consider for example the term $t = (a \oplus a) \oplus e$. We can apply to t both the rules (10) and (11), and the affected subterms partially overlap. Therefore the critical pairs method may miss conflicts between rules in modular rewriting.

Proof. We first conjecture the unique normal form for both theories and then show that every rule application gets the term closer to this conjectured normal form.

First, we conjecture that $\rightarrow_{C_{xor}/F_{xor}}$ leads to the following normal form: let $t_1 \oplus \ldots \oplus t_n$ be a representation of the term to normalize such that the subterms t_i are already normalized and $t_i \not\approx_{F_{xor}} s_1 \oplus s_2$ for any terms s_1 and s_2. Consider the multiset of the t_i where equality is modulo F_{xor}. The normal form is the term

$s_1 \oplus \ldots \oplus s_n$ where the s_i are exactly those terms that occur in the multiset with an odd number of occurrences, if there is at least one such s_i. Otherwise, the normal form is simply e. Observe that every application of $\rightarrow_{C_{xor}/F_{xor}}$ reduces the size of the term to which it is applied and brings the term closer to the normal-form, as either two xor-ed occurrences of F_{xor}-equivalent subterms are replaced by e, or an xor-ed e is removed.

The conjectured normal form for $\rightarrow_{C_{exp}/F_{exp}}$ is as follows. Let

$$exp(exp(\ldots exp(s, t_1), t_2), \ldots t_n)$$

be a term with normalized subterms t_i, and such that $s \not\approx_{F_{exp}} exp(s_0, s_1)$ for any terms s_i. Consider the multiset M of the t_i where equality is modulo F_{exp}. We build a multiset M' as follows: for every term t that is contained n times in M and such that t^{-1} is contained k times in M, let t be contained in M' $n - k$ times, if $n \geq k$, and $k - n$ times otherwise. If M' is empty, then the normal form is simply s, otherwise it is $exp(...exp(s, ...)...)$ with the exponents being the elements of M'.

Observe that every application of $\rightarrow_{C_{exp}/F_{exp}}$ reduces the size of the term to which it is applied and brings the term closer to the normal form, as two inverse exponents are removed in every step.

We thus have shown the confluence of these relations; termination follows from the fact that the entire $\rightarrow_{C_{ex}/F_{ex}}$ is terminating. Thus, we have convergence of these subtheories. □

It now remains to show that the union of disjoint convergent theories (of the form we consider) is also convergent.

Lemma 13. *Let F be an FEC theory and let C be a cancellation theory. Let $E = F \cup C$, and let E_1 and E_2 be a partition of E into disjoint subtheories. Let further $F_i = F \cap E_i$ and $C_i = C \cap E_i$. Finally, let $\rightarrow_{C_1/F_1}$ and $\rightarrow_{C_2/F_2}$ be convergent. Then also $\rightarrow_{C/F}$ is convergent.*

Proof. Suppose the opposite holds, i.e. that there are terms t_1 and t_2 such that $t_1 \approx_F t_2$, $t_1 \rightarrow_C t_1'$, and $t_2 \rightarrow_C t_2'$ where t_1' and t_2' are not joinable. Then there are rules $(l_1 \rightarrow r_1) \in C_1$ and $(l_2 \rightarrow r_2) \in C_2$, positions p_1 in t_1 and p_2 in t_2, and substitutions σ_1 and σ_2 such that $t_{1|p_1} \approx_F l_1\sigma$ and $t_{2|p_2} \approx_F l_2$. We can exclude the case that the two rules belong to the same subtheory C_i since otherwise the $\rightarrow_{C_i/F_i}$ would not be convergent.

Now we show that we can find a term $t \approx_F t_1 \approx_F t_2$ such that both rules $(l_1 \rightarrow r_1)$ and $(l_2 \rightarrow r_2)$ are applicable to t and the results of these applications are joinable with t_1' and t_2', respectively. After proving this statement, we can prove the lemma as follows: since C_1 and C_2 are disjoint, the positions of t at which the two rules are applicable are either non-overlapping subterms of t (thus both rules are independently applicable and there cannot be a conflict) or one is a proper subterm of the other. In this case, there also cannot be a conflict since performing the outer reduction first is equivalent to first applying the inner reduction and then the outer one (as the respective subterm is removed by the outer reduction).

So it remains to show that there is a term $t \approx_F t_1 \approx_F t_2$ and terms t_1'' and t_2'' such that $t \to_{C_1} t_1''$, $t \to_{C_2} t_2''$, $t_1' \downarrow_{C/F} t_1''$, and $t_2' \downarrow_{C/F} t_2''$.

For a term s and theory E, let $subterm_E(s)$ be the set of subterms s' of any term in $[s]_{\approx_E}$ and such that the toplevel symbol of s' is in Σ_E. Let F_1 and F_2 be disjoint FEC theories and let s_1 and s_2 be terms with $s_1 \approx_{F_1} s_2$. Then it holds that $[subterms_{F_2}(s_1)]_{\approx_{F_1}} = [subterms_{F_2}(s_2)]_{\approx_{F_1}}$. Thus, after transformation with one subtheory F_1, all subterms with a root symbol from the other subtheory F_2 are still present (modulo equivalence in F_1).

So given $t_1 \approx_F t_2$, one can obtain a term $t \approx_F t_1$ such that both rules are applicable to t in other non-overlapping positions or subpositions. Also the resulting terms must be joinable with t_1' and t_2', respectively, since otherwise we can construct a counter-example for the convergence of $\to_{C_1/F_1}$ or $\to_{C_2/F_2}$. □

Putting lemmata 11-13 together, we can thus conclude:

Lemma 14. *$\to_{C_{ex}/F_{ex}}$ is convergent.*

4.2 Equality modulo $F \cup C$

The following theorem tells us that the ground word problem modulo $F \cup C$ is decidable in our framework. This is a direct consequence of our assumption that $\to_{C/F}$ is convergent and that we can decide matchability modulo an FEC theory F as a consequence of Theorem 1.

Theorem 3. *Let F be an FEC theory and C a cancellation theory, and let $\to_{C/F}$ be convergent. Then the ground word problem for $F \cup C$ is decidable.*

Proof. Recall that, since $\to_{C/F}$ is convergent, $t_1 \approx_{F \cup C} t_2$ implies $t_1 \downarrow_{C/F} = t_2 \downarrow_{C/F}$. The normal form of the terms can be computed for every term since C is convergent modulo F and F is an FEC theory. □

By Lemma 3, it follows that we can construct a unification algorithm modulo $F \cup C$ for terms with bounded variables. In particular, this implies that the unifiability problem modulo $F \cup C$ for terms with bounded variables is decidable.

4.3 Cancellation as analysis

The results that we have presented so far allow us to decide, for ground terms or terms with bounded variables, the equality of terms modulo an FEC theory F and a cancellation theory C, as well as the intruder deduction problem in the theory F. We now consider how to solve the intruder deduction problem in the theory $F \cup C$. In §4.4, we will see that this problem is in general undecidable, so to obtain a decidable problem we must further restrict our model: we bound the number of operations that the intruder can perform.

The idea that we put forth here to solve the intruder deduction problem with respect to $F \cup C$ is to distinguish synthesis (or composition) and analysis (or decomposition) of messages by the intruder. Observe that these two aspects of

intruder deduction are not completely independent; for instance, if the intruder knows the messages $\{\!|m|\!\}_{\langle k_1,k_2\rangle}$ and k_1 and k_2, then he can analyze the encrypted message, but only after synthesizing the key $\langle k_1,k_2\rangle$. We now define a general notion of analysis based on an arbitrary cancellation theory C.

Intuitively, we speak of *synthesis* when the intruder applies the OP rule to compose terms, excluding the case when the resulting composed term is a redex according to the cancellation theory C (as we can then reduce it to a simpler term). We speak of *analysis* when the intruder applies the OP rule to obtain a redex whose normal form cannot be composed from his current knowledge. We can then formalize the notion of the intruder knowledge being completely analyzed based on the notion of cancellation rules present in our framework: we say that the intruder has analyzed his knowledge as far as possible if, by applying the cancellation rules, the intruder can only derive messages (except redices in C) that he can also derive without cancellation rules. Formally:

Definition 8. *Let C be a cancellation theory convergent modulo an FEC theory F. We say that a finite set of ground terms IK is* analyzed with respect to C modulo F *if $t{\downarrow_{C/F}} \subseteq \mathcal{DY}_F(IK)$ for each $t \in \mathcal{DY}_F(IK)$.*

As an example, consider again F_{ex} and C_{ex}. The set $IK = \{\{\!|m|\!\}_k, k\}$ is not analyzed with respect to C_{ex} modulo F_{ex} as the intruder can generate $t = \{\!|\{\!|m|\!\}_k|\!\}_k \in \mathcal{DY}_{F_{ex}}(IK)$, and $t{\downarrow_{C_{ex}/F_{ex}}} = [m]_{\approx_{F_{ex}}}$, but $m \notin \mathcal{DY}_{F_{ex}}(IK)$. However, $IK' = IK \cup \{m\}$ is analyzed since all messages that can be obtained only by normalizing terms in $\mathcal{DY}_{F_{ex}}(IK')$ are already contained in $\mathcal{DY}_{F_{ex}}(IK')$.

We thus have a characterization of analyzed intruder knowledge as a set that contains all messages that can be derived under $\mathcal{DY}_{F\cup C}(\cdot)$ and but not under $\mathcal{DY}_F(\cdot)$. The idea is that when the set of messages known by the intruder is analyzed, then there is no need to consider the cancellation theory in the derivations of the intruder. Hence we can decide the intruder deduction problem $\mathcal{DY}_{F\cup C}(\cdot)$ when the intruder knowledge is analyzed:

Theorem 4. *Let F be an FEC theory and C a cancellation theory, and let $\rightarrow_{C/F}$ be convergent. Further, let t be a ground term and IK be a finite set of ground terms analyzed with respect to C modulo F. Then it is decidable whether $t \in \mathcal{DY}_{F\cup C}(IK)$.*

Proof. Since IK is analyzed, this problem is equivalent to the problem $t{\downarrow_{C/F}} \subseteq \mathcal{DY}_F(IK)$. Since we can effectively compute $t{\downarrow_{C/F}}$, by Lemma 6, this problem is decidable. □

By Lemma 3, it follows that the intruder deduction problem is decidable for terms with bounded variables when the intruder knowledge is analyzed.

4.4 Undecidability of analysis

The previous method for solving the intruder deduction problem is restricted to the case where the intruder knowledge is analyzed. The central question thus is how to transform an arbitrary intruder knowledge into an analyzed one.

Theorem 5. *There is an FEC theory F and a cancellation theory C, where $\rightarrow_{C/F}$ is convergent, such that it is undecidable whether a finite set of ground terms IK is analyzed with respect to C modulo F. Moreover, the intruder deduction problem $t \in \mathcal{DY}_{F \cup C}(IK)$ is also undecidable.*

Note that [1, 2] have shown that the intruder deduction problem in a theory E can be undecidable even if unification in E is decidable. Our theorem is incomparable to this result as it does not require E to be decidable.

Proof. We consider again the FEC theory F_{PCP} defined in the proof of Lemma 8. Let $C_{PCP} = \{f(res(Z, Z, S)) \rightarrow secret\}$ for some new constant *secret* (that the intruder does not know) and a new function symbol f that we need for convergence. Note that we extend the set $\mathcal{O}$ of intruder-accessible operators by the function f.

We first show convergence. Since f is a new function symbol, there cannot be a conflict between two instances of the single rule of C_{PCP} in a non-variable position, and similarly there cannot be a conflict with F_{PCP} in a non-variable position. As obviously $\rightarrow_{C_{PCP}/F_{PCP}}$ terminates, we conclude that it is convergent.

Now, by the proof of Lemma 8, we know the intruder can generate a ground instance of the term $res(Z, Z, S)$ from IK_P if and only if the given PCP P has a solution. Thus the intruder can derive the constant *secret* from IK_P iff the PCP P has a solution. Also, the intruder knowledge IK_P is analyzed with respect to C_{PCP} modulo F_{PCP} iff the intruder cannot derive *secret*. Thus, by the undecidability of the PCP problem, also the problem whether the intruder knowledge is analyzed with respect to C_{PCP} modulo F_{PCP} is undecidable. □

We thus need to make further restrictions to obtain a general procedure for analyzing the intruder knowledge. We proceed by limiting the operations that the intruder can perform when analyzing a single message (i.e. the number of steps before he obtains a new redex). We define a bounded derivation of the intruder as follows:

Definition 9. *Given a finite set IK of ground terms and an algebraic theory E, we define the* k-bounded intruder model *as the least set $\mathcal{DY}^k_E(IK)$ that is closed under the rules*

$$\frac{}{t \in \mathcal{DY}^k_E(IK)}\ \mathrm{AX}^k\ (t \in IK\,,\, k \geq 0)\,, \qquad \frac{t_1 \in \mathcal{DY}^k_E(IK)}{t_2 \in \mathcal{DY}^k_E(IK)}\ \mathrm{EQ}^k\ (t_1 \approx_E t_2)\,,$$

$$\frac{t_1 \in \mathcal{DY}^k_E(IK) \cdots\ t_n \in \mathcal{DY}^k_E(IK)}{op(t_1, \ldots, t_n) \in \mathcal{DY}^{k+1}_E(IK)}\ \mathrm{OP}^k\ (op \in \Sigma^n)\,.$$

Note that, under the EQ^k rule, the use of an equivalence from E does not count as a step, i.e. it does not increase the counter k.

Definition 10. *Let F be an FEC theory and C a cancellation theory, and let $\rightarrow_{C/F}$ be convergent. Given a constant $k \in \mathbb{N}$, we say that the intruder knowledge IK, which is a finite set of ground terms, is* k-analyzed *(with respect to C modulo F) iff $t{\downarrow_{C/F}} \subseteq \mathcal{DY}^k_F(IK)$ for each $t \in \mathcal{DY}^k_F(IK)$.*

Theorem 6. *Let F be an FEC theory and C a cancellation theory, let $\rightarrow_{C/F}$ be convergent, and let $k \in \mathbb{N}$. Then it is decidable if a finite set of ground terms IK is k-analyzed (with respect to C modulo F).*

Proof. The set $IK' = \mathcal{DY}^k_F(IK)$ is finite, since F is an FEC theory and all compositions are bounded by k. We can then check if any message of $t \in IK'$ has a normal form $t{\downarrow_{C/F}}$ that is not contained in IK'. □

Note, however, that given a finite set of ground terms IK, there does not always exist a finite superset IK' of ground terms that is (k-)analyzed. Consider, for example, the theories $F = \{f(x) = g(h(x))\}$ and $C = \{g(X) = X\}$. Clearly, F is a FEC theory, C is a cancellation theory, and $\rightarrow_{C/F}$ is convergent. Furthermore, let $\mathcal{O} = \{f\}$ be the set of functions that the intruder can access, and let IK be a finite set of ground terms that contains a constant c. We then, for instance, have that $h(c), h(h(c)), \ldots \in \mathcal{DY}_{F\cup C}(IK)$. Thus, there is no finite set $IK' \supseteq IK$ such that IK' is analyzed. For the bounded case, observe that $g(t) \in \mathcal{DY}^k_{F\cup C}(IK \cup t)$ for any ground term t, $k \geq 1$, and $n \in \mathbb{N}$. Thus, any k-analyzed superset of IK must also contain $g^n(c)$ for any $n \in \mathbb{N}$, so it must be infinite. Hence, to complete our framework, we must be able to check bounded derivability without first computing an analyzed intruder knowledge. The following theorem tells us that this is possible:

Theorem 7. *Let F be an FEC theory and C a cancellation theory, let $\rightarrow_{C/F}$ be convergent, and let $k \in \mathbb{N}$. Then it is decidable if a ground term t can be derived from a finite set of ground terms IK, i.e. whether $t \in \mathcal{DY}^k_{F\cup C}(IK)$.*

Proof. Lemma 7 can be extended to bounded derivations, i.e. $t \in \mathcal{DY}^k_{E_1\cup E_2}(IK)$ iff $t \in [\mathcal{DY}^k_{E_1}(IK)]_{\approx_{E_1\cup E_2}}$. This is because the transformation employed in the proof of Lemma 7 does not change the number of applications of any derivation rule.

With this extension of Lemma 7, we can reduce the question whether $t \in \mathcal{DY}^k_{F\cup C}(IK)$ to $t \in [\mathcal{DY}^k_F(IK)]_{\approx_{F\cup C}}$. Since $\mathcal{DY}^k_F(IK)$ is finite, we can effectively check if t is $F \cup C$-equivalent to a term in $\mathcal{DY}^k_F(IK)$. Therefore we can decide whether $t \in \mathcal{DY}^k_{F\cup C}(IK)$. □

Together with the fact that, by Lemma 3, all problems over terms with bounded variables can be reduced to problems over ground terms, we have now the basis for protocol analysis modulo algebraic theories. Namely, we can check whether a term with bounded variables — representing the set of messages that some agent in its current state can receive as a valid protocol message — can be derived from a ground intruder knowledge under the bounds that we have introduced.

5 Related work and concluding remarks

We have presented a framework for security protocol analysis that can handle algebraic properties in a uniform and modular way. It is not specialized to any

particular algebraic theory and thereby allows users to declare new operators and properties as part of the protocol specification. Our framework is based on the use of modular rewriting to formalize a generalized equational deduction problem for the Dolev-Yao intruder, and on bounding the depth of message terms and the analysis operations of the intruder to control the complexity of the equational unification problems that arise. These bounds allow us to give general algorithms for the equational unification and intruder deduction problems. Moreover, under these bounds, our framework is also open to the integration of more efficient algorithms that are specialized to particular algebraic theories (and which usually work without such bounds), e.g. [11, 12, 26].

The idea of providing a general approach for integrating equational properties into security protocol analysis has recently attracted considerable attention. [17] presents an approach based on standard rewriting that supports the specification of properties like the cancellation theories of our framework. However it does not allow for properties like AC, which are handled by our FEC theories. The approach of [13] has aims similar to ours: to provide a general framework that is open to the integration of existing algorithms. This approach, however, is based on a different idea, namely ordered rewriting, and is therefore applicable to classes of theories that are incomparable to the ones that are supported by our framework. The approaches of [2, 15, 23, 25] are the most closely related to ours as they also employ modular rewriting. They differ from our work in that they are more restrictive in terms of the kinds of modulo theories that can be considered; namely they consider a fixed modulo theory (or, similarly, assume given a unification procedure for the modulo theory), or they require that the unification problems are finitary. These restrictions, however, allow them to work without the bounds required by our approach.

Our framework is not biased towards a particular analysis method, and thus can be used as a basis for handling algebraic equations when employing different types of formalisms or techniques for protocol analysis. As a concrete example, we have begun integrating our framework into our protocol model-checker OFMC [8, 9]. In this integration, the message and analysis bounds become parameters of the protocol analysis problem, along with other parameters like the number of sessions. We can then use different search techniques (like iterative deepening) to effectively search the resulting multi-dimensional search space.

The equational reasoning problems that we considered in this paper are in general undecidable and hence one must introduce restrictions to regain decidability. The restrictions that we have introduced are motivated by the practical problems in security protocol analysis and we have begun investigating whether and how they can be applied to other equational reasoning problems.

References

1. M. Abadi and V. Cortier. Deciding knowledge in security protocols under equational theories. In *Proceedings of ICALP'2004*, LNCS 3142, pages 46–58. Springer, 2004.

2. M. Abadi and V. Cortier. Deciding knowledge in security protocols under (many more) equational theories. In *Proceedings of CSFW'05*, pages 62–76. IEEE Computer Society Press, 2005.
3. A. Armando and L. Compagna. Automatic SAT-Compilation of Protocol Insecurity Problems via Reduction to Planning. In *Proceedings of FORTE 2002*, LNCS 2529, pages 210–225. Springer, 2002.
4. F. Baader and T. Nipkow. *Term Rewriting and All That*. Cambridge University Press, 1998.
5. F. Baader and K. U. Schulz. Unification in the union of disjoint equational theories: Combining decision procedures. *Journal of Symbolic Computation*, 21:211–243, 1996.
6. F. Baader and W. Snyder. Unification theory. In *Handbook of Automated Reasoning*, volume I, pages 445–532. Elsevier Science, 2001.
7. L. Bachmair and N. Dershowitz. Completion for rewriting modulo a congruence. *Theoretical Computer Science*, 67:173–201, 1989.
8. D. Basin, S. Mödersheim, and L. Viganò. Constraint Differentiation: A New Reduction Technique for Constraint-Based Analysis of Security Protocols. In *Proceedings of CCS'03*, pages 335–344. ACM Press, 2003.
9. D. Basin, S. Mödersheim, and L. Viganò. OFMC: A symbolic model checker for security protocols. *International Journal of Information Security*, 4(3):181–208, 2005.
10. M. Boreale and M. G. Buscemi. A framework for the analysis of security protocols. In *Proceedings of CONCUR 2002*, LNCS 2421, pages 483–498. Springer, 2002.
11. Y. Chevalier, R. Küsters, M. Rusinowitch, and M. Turuani. An NP Decision Procedure for Protocol Insecurity with XOR. In *Proceedings of LICS'03*, pages 261–270. IEEE Computer Society Press, 2003.
12. Y. Chevalier, R. Küsters, M. Rusinowitch, and M. Turuani. Deciding the Security of Protocols with Diffie-Hellman Exponentiation and Products in Exponents. In *Proceedings of FST TCS'03*, LNCS 2914, pages 124–135. Springer, 2003.
13. Y. Chevalier and M. Rusinowitch. Combining Intruder Theories. In *Proceedings of ICALP 2005*, LNCS 3580, pages 639–651, 2005.
14. H. Comon and V. Shmatikov. Is It Possible to Decide Whether a Cryptographic Protocol Is Secure Or Not? *Journal of Telecommunications and Information Technology*, 4:5–15, 2002.
15. H. Comon-Lundh and S. Delaune. The finite variant property: How to get rid of some algebraic properties. In *Proceedings of RTA'05*, LNCS 3467, pages 294–307. Springer, 2005.
16. V. Cortier, S. Delaune, and P. Lafourcade. A survey of algebraic properties used in cryptographic protocols. *Journal of Computer Security*, 2005. To appear.
17. S. Delaune and F. Jacquemard. A decision procedure for the verification of security protocols with explicit destructors. In *Proceedings of CCS'04*, pages 278–287. ACM Press, 2004.
18. D. Dolev and A. Yao. On the Security of Public-Key Protocols. *IEEE Transactions on Information Theory*, 2(29), 1983.
19. N. Durgin, P. D. Lincoln, J. C. Mitchell, and A. Scedrov. Undecidability of Bounded Security Protocols. In *Proceedings of the FLOC'99 Workshop on Formal Methods and Security Protocols (FMSP'99)*, 1999.
20. IETF: The Internet Engineering Task Force. `http://www.ietf.org`.
21. J.-P. Jouannaud and H. Kirchner. Completion of a set of rules modulo a set of equations. *SIAM Journal of Computing*, 15(4):1155–1194, 1986.

22. D. Kapur, P. Narendran, and L. Wang. An E-unification algorithm for analyzing protocols that use modular exponentiation. In *Proceedings of RTA 2003*, LNCS 2706, pages 165–179. Springer, 2003.
23. P. Lafourcade, D. Lugiez, and R. Treinen. Intruder deduction for AC-like equational theories with homomorphisms. In *Proceedings of RTA'05*, LNCS 3467, pages 308–322. Springer, 2005.
24. G. Lowe. Casper: a Compiler for the Analysis of Security Protocols. *Journal of Computer Security*, 6(1):53–84, 1998.
25. J. Meseguer and P. Thati. Symbolic reachability analysis using narrowing and its application to verification of cryptographic protocols. *Journal of Higher-Order and Symbolic Computation*, to appear.
26. J. K. Millen and V. Shmatikov. Symbolic protocol analysis with products and Diffie-Hellman exponentiation. In *Proceedings of CSFW'03*, pages 47–61. IEEE Computer Society Press, 2003.
27. J. Siekmann and P. Szabó. The undecidability of the D_A unification problem. *Journal of Symbolic Computation*, 54(2):402–414, 1989.