In Didier Galmiche ed., *Proceedings of the International Conference on Analytic Tableaux and Related Methods (TABLEAUX-97)*. To appear in Lecture Notes in Artificial Intelligence, © Springer Verlag, 1997

Tableaux Methods for Access Control in Distributed Systems

Fabio Massacci*

Computer Laboratory
University of Cambridge, England (UK)
e-mail: Fabio.Massacci@cl.cam.ac.uk

Abstract. The aim of access control is to limit what users of distributed systems can do directly or through their programs. As the size of the systems and the sensitivity of data increase formal methods of analysis are often required.
This paper presents a prefixed tableaux method for the calculus of access control in distributed system developed at DEC-SRC by Abadi, Lampson et. al. Beside the applicative interest, the calculus poses interesting technical challenges, since it has not the tree-model property, introduces relations between modalities which cannot be compiled into axiom schemas, and has some features of the universal modality.
As a side-effect we show a tableaux calculus for the universal modality which distinguishes it from $S5$ (via satisfiability on non tree-models).

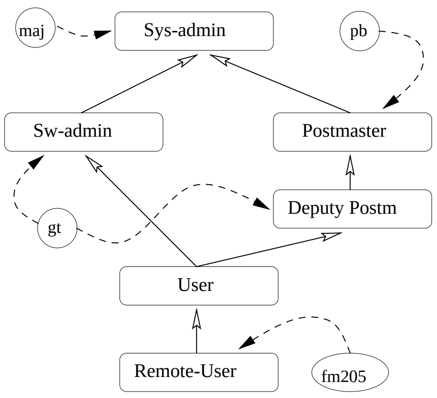
1 Introduction

Access control is a key issue for the security of computer systems (see [25] for an introduction). Its main purpose is to restrain the actions which legitimate (or malicious) users may perform, either directly or indirectly (through their programs). Its need arises in any system with multiple users and sensitive information or shared resources such as the military [4], banking and commerce [7] or health care services [2].

Distributed systems face additional challenges (e.g. large scale, insecure communications, delegation of management etc.) which require new modelling and reasoning techniques. For instance, access control must be combined with authentication [31], and policies must be refined at the various levels of delegation [22]. The definition of the jurisdiction capabilities of communicating agents plays also a key role in the analysis of security protocols [5, 28]. Indeed access control is just a problem of jurisdiction in complex and distributed systems.

As systems become more complex, human (and informal) verification becomes infeasible. Hence formal methods, logics and automated reasoning techniques can be useful tools for the verification of security policies and access control procedures (see e.g. [5, 20, 19, 22, 31]). Our target is the development of (tableaux based) automated reasoning techniques for access control.

* Current address: Dip. Informatica e Sistemistica, Universitá di Roma "La Sapienza", via Salaria 113, I-00198 Roma (I), e-mail massacci@dis.uniroma1.it.



maj: Titan Room Manager, knows about UNIX, DEC stations, Alphas, Ethernet, ATM, printer systems, backup systems and TeX.
gt: Knows about UNIX, Suns, HP bobcats and snakes, GNU emacs, X, Ethernet, Lisp and similar languages. Deputy Postmaster.
pb: Postmaster, knows about UNIX, Suns, X, mail, news, and wide area communications.

[...] If you anticipate a need to login from outside Cambridge, you should consult maj or pb.

Fig. 1. From "Computing Facilities at the Computer Laboratory"

The principles of access control can be described with few abstractions: *subjects* (humans, programs etc.), *objects* (data, other programs etc.) and *privileges* which subjects detain on objects (e.g. read, write and execute in UNIX). The use of these abstractions is the basis of most formal models proposed in the literature, starting from the classical access matrix [18, 24] to more advanced systems [10, 19, 22, 23]. A key feature of the new approaches is the attempt to model more closely the (hierarchical) relationships between the various subjects, where some privileges can be inherited along the chains (e.g. Fig. 1).

The use of these abstractions leads naturally towards a formalisation of the problem with multi modal logics: one subject, one modality. There has a been a number of works on modelling security and obligations in a multiagents setting, e.g. [8, 17, 30], and in particular we focus on the expressive calculus developed at DEC-SRC [1]. This calculus is interesting for a number of reasons:

- it provides a uniform framework for reasoning about access control in presence of delegation and has a simple semantics [1];
- it constitutes the basis of a real system [19, 31];
- its features pose interesting technical challenges for deduction.

One of the characteristics, which challenge "standard" tableaux calculi, is the presence of formulae used for modelling delegation certificates and hierarchical relationships between subjects (i.e. modalities). Those relations have the same force of axiom schemas and are close to role-value-map constructs of AI languages [26]. The key difficulty is that we cannot "compile" them into tableaux rules (nor axiom schemas) since their presence depends on the particular non logical axioms *and* the particular theorem we want to prove. Two different theorems (i.e. access requests with different delegation certificates) may impose totally different relations between subjects.

Another feature of the logic is the absence of the tree model property which also hinders a straightforward extension of tableaux calculi. Finally it shares some properties of the universal modality [12, 13] which cannot be axiomatised[2].

In a nutshell, some global properties of the underlying Kripke models can only be discovered *on-line* during the deduction process.

The tableaux method proposed here is based on Simple Step Tableaux [9, 14, 21] given their flexibility to adapt to various logics, the possibility of a simple implementation along the line of lean theorem proving [3] and its potential use as a target calculus for translations from the matrix methods [27]. The use of tableaux is not new for security and dates back to the `VERUS` system [20].

The proposed solution uses an extension of prefixed tableaux where prefixes constitute a forest rather than a tree and the set of global axioms is modified at run time. This requires to change the definition of tableau branches and makes possible a simple iterative construction for the universal modality.

We derive a sound and complete decision method for a large fragment of the calculus and a correct one for the whole language (the calculus is undecidable in general). This method extends the deduction capabilities of [1] as we can prove important properties which must be added as non-logical axioms in [1, 19].

An important side-effect of these techniques, shown in the appendix, is an EXPTIME tableau for multi-modal logics with the universal modality [12, 13].

In what follows, we present the DEC-SRC calculus (§2) and analyse its semantical features (§3). We discuss the tableaux calculus (§4) with some examples (§5) and sketch its soundness and completeness proofs (§6). Conclusions (§7) are followed by an appendix on the universal modality.

2 The DEC-SRC Calculus for Access Control

To make the paper self-contained we sketch the intuitions behind the calculus and refer to [1] for a formal treatment and to [19, 31] for its applications.

Users, roles, groups and cryptographic keys are represented by atomic subjects, or *principals*, and denoted by A, B, K_A etc. Complex principals (P or Q) are built by conjunction "&" and quoting "|". The intuition is that $P \,\&\, Q$ is a principal with the privileges of both P and Q, whereas $P \,|\, Q$ corresponds to the principal P claiming to quote a request from Q. Notice that P may claim to quote Q even when Q never said anything.

Other operators are possible i.e. $A \;\mathtt{for}\; B$ and $A \;\mathtt{as}\; R$ [1]: the former is used when A is claiming to act as a delegate for B; the latter when A speaks using a role[3] R. Since they can be encoded using | and &, we do not use them. For instance $A \;\mathtt{for}\; B \doteq (A \,\&\, D) \,|\, B$ where D is a delegation server [1].

Operations over objects are represented by *statements* denoted by s. Atomic statements (r) are uninterpreted operations or requests [1, 23] i.e. propositional letters which can be true in a particular state of the system (request granted)

[2] This may also explain why only a sound axiomatisation has been devised in [1].

[3] For a distinction between the security concepts of role and group see [10, 23].

$SysAdm \Rightarrow SwAdm \wedge$	$maj \Rightarrow SysAdm \wedge$
$SysAdm \Rightarrow PostMaster \wedge$	$gt \Rightarrow SwAdm \wedge$
$SwAdm \Rightarrow Usr \wedge$	$gt \Rightarrow DepPostM \wedge$
$PostMaster \Rightarrow Usr \wedge$	$pb \Rightarrow PostMaster \wedge$
$Usr \Rightarrow RemUsr$	$fm205 \Rightarrow RemUsr$
$\wedge$	$\wedge$
$SysAdm\ \texttt{controls}\ (fm205 \Rightarrow Usr) \wedge$	$maj\ \texttt{says}\ (fm205 \Rightarrow Usr) \wedge$
$Usr\ \texttt{controls}\ login(telnet) \wedge$	$fm205\ \texttt{says}\ login(telnet) \wedge$
$RemUsr\ \texttt{controls}\ login(ftp) \wedge$	$fm205\ \texttt{says}\ read(mail)$
$(fm205 \,\vert\, Usr)\ \texttt{controls}\ read(mail)$	

Fig. 2. A Logical Formalisation of Fig. 1

or false (not granted). Complex statements are built with boolean connectives $\wedge, \neg, \supset$ etc.: for instance *login_telnet* $\supset$ *login_ftp* whose intuitive (and formal) interpretation is "if telnet has been granted so has ftp".

To represent *user requests* we use the modal statement "$P\ \texttt{says}\ s$": principal P requests s to be granted. If P is a group then we follow [1, 19] and interpret it as "somebody in group P says s".

Hierarchical relations between principals are constructed with the *speaks-for* statement $P \Rightarrow Q$. The intuition is that P has at least all the privileges of Q i.e. P can speak for Q. If P says s this would be as Q itself said s. It is also used for group membership: $P \Rightarrow G$ means that P has at least all privileges of group G.

Principals and statements are linked by *privileges attributions* [23]: the statement "$P\ \texttt{controls}\ s$" captures the intuition that principal P has access control over s. In the literature on authentication this is called *jurisdiction* of a principal [5, 28] and axiomatised as $A\ \texttt{says}\ s \wedge A\ \texttt{controls}\ s \ \supset\ s$.

The aim is to to replace it by a more complex but more realistic axiom, where the relation between two principals A and B is expressed with the $\Rightarrow$ operator:

$$A\ \texttt{says}\ s \wedge B\ \texttt{controls}\ s \wedge \text{ "some relation between } A \text{ and } B\text{"} \supset s.$$

A particular $P \Rightarrow Q$ may depends on the statements of other principals. For instance $fm205 \Rightarrow Usr$ in Fig. 2 depends on maj's statements.

3 Formal Syntax and Semantics

The language (described informally in §2) is the following, where A is an atomic principal and r an atomic propositional request:

$$P, Q\ ::=\ A \mid P \,\&\, Q \mid (P \,\vert\, Q) \qquad s, s'\ ::=\ r \mid \neg s \mid s \wedge s' \mid P\ \texttt{says}\ s \mid (P \Rightarrow Q)$$

Other connectives are abbreviations, e.g. $s \supset s' \equiv \neg(s \wedge \neg s')$. Also $P\ \texttt{controls}\ s$ is a shortcut for $(P\ \texttt{says}\ s) \supset s$.

In the sequel we assume that in $P \Rightarrow Q$ either P or Q is an atomic principal, w.l.o.g. since $P \Rightarrow Q$ is equivalent to $P \Rightarrow A \wedge A \Rightarrow Q$ for a new atomic A.

A statement is *left (right) restricted* when speaks-for subformulae have the form $A \Rightarrow Q$ (respectively $P \Rightarrow A$) i.e. the left (right) principal is atomic. It is *weakly left (right) restricted* when statements $P \Rightarrow Q$ are admitted if they occur under the scope of an odd number of negations[4]. It is *request restricted* when in each statement of the form $P\ \mathtt{says}\ s$, the statement s is either an atomic request or a group membership (both possibly negated). For instance the formalisation in Fig. 2 is left, right and request restricted.

In practice statements are right and request restricted. If $\Rightarrow$ is used for hierarchies and group and role membership, as in Fig. 1, the rightmost principal is atomic. Moreover, in almost all systems [25], privileges attributions are represented by ACL (Access Control Lists). In the DEC-SRC language, an ACL for a request r is simply the conjunctions of statements $\bigwedge_i P_i\ \mathtt{controls}\ r$, where r is uninterpreted [19, 23]. If we add, among the possible privileges, the possibility to hand over delegation to other principals such as $P_i\ \mathtt{controls}\ (Q_j \Rightarrow A_k)$, then we still have a right restricted language.

The semantics is based on Kripke models [1, 11, 16]: a relation models the compatibility of a state with the requests made by a principal in the real world.

A model is a pair $\langle W, \mathcal{I} \rangle$, where W is a non empty set of states and $\mathcal{I}$ an interpretation such that for every atomic principal A it is $A^{\mathcal{I}} \subseteq W \times W$ and for every propositional letter r it is $r^{\mathcal{I}} \subseteq W$. Then $\mathcal{I}$ is extended as follows:

$$
\begin{array}{ll}
(\neg s)^{\mathcal{I}} & = W - s^{\mathcal{I}} \\
(s \wedge s')^{\mathcal{I}} & = s^{\mathcal{I}} \cap s'^{\mathcal{I}} \\
(P \Rightarrow Q)^{\mathcal{I}} & = \text{ if } Q^{\mathcal{I}} \subseteq P^{\mathcal{I}} \text{ then } W \text{ else } \emptyset \\
(P\ \mathtt{says}\ s)^{\mathcal{I}} & = \left\{ w \mid \forall w^* \in W \text{ if } \langle w, w^* \rangle \in P^{\mathcal{I}} \text{ then } w^* \in s^{\mathcal{I}} \right\} \\
(P \,\&\, Q)^{\mathcal{I}} & = P^{\mathcal{I}} \cup Q^{\mathcal{I}} \\
(P \mid Q)^{\mathcal{I}} & = \left\{ \langle w, w^* \rangle \mid \exists w^{**} \langle w, w^{**} \rangle \in P^{\mathcal{I}} \text{ and } \langle w^{**}, w^* \rangle \in Q^{\mathcal{I}} \right\}
\end{array}
$$

Definition 1. A statement s is *satisfiable* iff there is a model $\langle W, \mathcal{I} \rangle$ where $(s)^{\mathcal{I}}$ is not empty. A statement s is *valid* iff for every model $\langle W, \mathcal{I} \rangle$ it is $(s)^{\mathcal{I}} = W$.

For simplicity, we write $w \Vdash s$ for $w \in s^{\mathcal{I}}$ and interchange a set of statements with their conjunction. Next we introduce the set of *global axioms* G which holds in every possible world [11, 21]. They are non-logical axioms describing the access control system: groups membership, privileges attributions etc.

Definition 2. A statement s is a *logical consequence* of G, i.e. $G \models s$, iff for every $\langle W, \mathcal{I} \rangle$ if $\forall w \in W, w \Vdash G$, then $\forall w \in W, w \Vdash s$.

Global axioms can be incorporated in the axiomatization of [1] with the modal deduction theorem [16, 11], but their explicit representation is more effective because the modal deduction theorem leads to an exponential blow up [15].

[4] For instance the formula $\neg A\ \mathtt{says}\ ((B \,\&\, C) \Rightarrow D)$ is weakly left restricted since the group membership $(B \,\&\, C) \Rightarrow D$ is under the scope of one negation.

Remark. The semantics of $P \Rightarrow Q$ reflects global properties of the model, is close to the universal modality [12, 13], and can introduce axiom schemas on the fly.

We can represent explicity the relation with the universal modality $[u]$ as:

$$P \Rightarrow Q \equiv [u](P \Rightarrow_{loc} Q)$$

where we have the following two conditions:

$$\begin{aligned}(P \Rightarrow_{loc} Q)^{\mathcal{I}} &= \left\{ w \mid \forall w^* \in W \text{ if } \langle w, w^* \rangle \in Q^{\mathcal{I}} \text{ then } \langle w, w^* \rangle \in P^{\mathcal{I}} \right\} \\ ([u]s)^{\mathcal{I}} &= \text{if } \forall v \in W v \Vdash s \text{ then } W \text{ else } \emptyset\end{aligned}$$

A key property is the possibility of *introducing axiom schemas "on the fly"*. For instance $P \Rightarrow P \mid P$ forces the transitivity of relation $P^{\mathcal{I}}$, where P may be a complex principal. Yet, these global properties may or may not be present. As an example, suppose we have $A\,\texttt{says}\,(B \Rightarrow B \mid B)$. Transitivity of B will follow only if $\neg A\,\texttt{says}\,\bot$ is the case. So B's properties depend on the particular global axioms and theorems we are trying to prove.

Another feature is the *absence of the tree-model property* [29, 16]:

$$A \Rightarrow A \mid A \wedge \neg(A \Rightarrow B) \wedge B\,\texttt{says}\,\bot \wedge \neg(A\,\texttt{says}\,\bot) \wedge A\,\texttt{says}\,(B\,\texttt{says}\,\bot \wedge \neg A\,\texttt{says}\,\bot)$$

has no tree model at all, although it is satisfiable in the world 1 of the model with $W = \{1, 11, 2\}$ and $A^{\mathcal{I}} = \{\langle 1, 11\rangle, \langle 11, 11\rangle\}$ and $B^{\mathcal{I}} = \{\langle 2, 2\rangle\}$. The key point is that this model has *two* clusters (connected components) so that 1 satisfies the (local) **says** statements and 2 satisfies the global $\neg(A \Rightarrow B)$.

In model theoretic terms, (un)satisfiability is not preserved under disjoint union as in "traditional" modal logics [29]. This is due to the "hidden" presence of the universal modality which makes impossible the complete characterisation of the logic with an Hilbert system [12, 13, 29].

The logic is also *undecidable* and a reduction to pushdown automata (without details) is pointed out in [1].

A simpler proof uses the techniques of [26] and reduces validity to the world problem of (semi)groups: map elements to atomic principals, composition "$\circ$" to quoting "$\mid$" and equations between words $p \cong q$ to statements $P \Rightarrow Q \wedge Q \Rightarrow P$ (for short $P = Q$) for the corresponding P and Q. For groups introduce a principal I for identity, one A^c for the converse of each atomic A, and the relative equations. Next use a new principal Gr with global axioms $Gr \mid A = Gr$ for every atomic A and the statement $Gr\,\texttt{says}\,(P = Q)$ for every $p \cong q$ characterising the group. Then one can prove that $Gr\,\texttt{says}\,(P' = Q')$ is valid with those assumptions iff the equation $p' \cong q'$ holds for the group [26].

4 A Tableaux Calculus

Prefixed tableaux use *prefixed statements*, i.e.pairs $\langle \sigma : s \rangle$ where s is a statement and σ is an alternating sequence of integers n and atomic principals A called *prefix* and defined as $\sigma ::= n \mid \sigma.A.n$. A key difference from "standard" prefixed

$$\langle quote\rangle : \frac{\sigma : \neg(P \mid Q\,\mathsf{says}\,s)}{\sigma : \neg P\,\mathsf{says}\,(Q\,\mathsf{says}\,s)} \qquad [quote] : \frac{\sigma : P \mid Q\,\mathsf{says}\,s}{\sigma : P\,\mathsf{says}\,(Q\,\mathsf{says}\,s)}$$

$$\langle and\rangle : \frac{\sigma : \neg(P \,\&\, Q\,\mathsf{says}\,s)}{\sigma : \neg P\,\mathsf{says}\,s \mid \sigma : \neg Q\,\mathsf{says}\,s} \qquad [and] : \frac{\sigma : P \,\&\, Q\,\mathsf{says}\,s}{\begin{array}{c}\sigma : P\,\mathsf{says}\,s\\ \sigma : Q\,\mathsf{says}\,s\end{array}}$$

$$\text{Glob} : \frac{\vdots}{\sigma : s} \text{ if } \sigma \text{ is present in } \mathcal{B} \text{ and } s \in G_{\mathcal{B}}$$

$$\langle A\rangle : \frac{\sigma : \neg(A\,\mathsf{says}\,s)}{\sigma.A.m : \neg s}\ \sigma.A.m \text{ new} \qquad [A] : \frac{\sigma : A\,\mathsf{says}\,s}{\sigma.A.n : s}\ \sigma.A.n \text{ present}$$

$$D(A) : \frac{\sigma : A\,\mathsf{says}\,s}{\sigma : \neg A\,\mathsf{says}\,\neg s} \text{ with some } \sigma.A.n \text{ already present in the branch}$$

Fig. 3. PDL-like Rules for Modal Connectives

tableaux [11, 14] is that a set of prefixes now describes a *forest of trees*, where arcs are labelled with atomic principals and node with integers. With k different initial prefixes we have, in graph-theoretic terminology, k connected components or clusters [16]. With global axioms and the operator $\Rightarrow$ we can impose an euclidean or transitive closures on a cluster but we cannot collapse two clusters.

Still the definition of tableau is similar to prefixed tableaux for modal logics [11, 14, 21]: a *tableau* $\mathcal{T}$ is a rooted (binary) tree where nodes are labelled with prefixed statements in the usual fashion.

Definition 3. A *branch* of a tableau $\mathcal{T}$ is a pair $\langle\mathcal{B}, G_{\mathcal{B}}\rangle$ where $\mathcal{B}$ is a path from the root to a leaf of $\mathcal{T}$ and $G_{\mathcal{B}}$ is a set of global axioms.

Thus, each time we branch the tree we should also duplicate (in theory) the set of global axioms. This definition is essential because we need to modify the set of global axioms during the deduction process and therefore different branches may end up with different global axioms.

A prefix is *present* in a branch $\langle\mathcal{B}, G_{\mathcal{B}}\rangle$, if there is a prefixed statement with that prefix already in $\mathcal{B}$, and it is *new* if it is not already present.

The rules for propositional connectives are standard [11, 14, 21] and omitted. The rules for conjunction, quoting, the use of global axioms and the transitional rules for atomic principals are in Fig. 3.

To cope with $\Rightarrow$ we introduce a new set of propositional atoms x_i (distinct from r) to mark unsaid statements as in Fig. 4. Since $P \Rightarrow Q$ implies that if $P\,\mathsf{says}\,s$ then $Q\,\mathsf{says}\,s$ for all s, its negation means that there is "something" (an unknown x_i) which P said but Q didn't. The first two rules correspond to

$$\langle R_{gr}\rangle : \frac{\sigma : P{\Rightarrow}A \quad \sigma : \neg(A\,\mathsf{says}\,s)}{\sigma : \neg(P\,\mathsf{says}\,s)} \qquad [L_{gr}] : \frac{\sigma : A\,\mathsf{says}\,s \quad \sigma : A{\Rightarrow}Q}{\sigma : Q\,\mathsf{says}\,s}$$

$$\langle U_{gr}\rangle : \frac{\sigma : \neg(P{\Rightarrow}Q)}{\begin{array}{c} n : P\,\mathsf{says}\,x_i \\ n : \neg(Q\,\mathsf{says}\,x_i)\end{array}}\ x_i \text{ and } n \text{ new} \qquad [U_{gr}] : \frac{\sigma : P{\Rightarrow}Q}{G_{\mathcal{B}} := G_{\mathcal{B}} \cup \{P{\Rightarrow}Q\}}$$

Fig. 4. Rules for the speaks-for operator

the local features of the $\Rightarrow$ operator, whereas the last is due to its "universal" flavour. The $\langle U_{gr}\rangle$-rule combines both aspects.

Remark. Weakly left restricted statements do *not* need rule $\langle R_{gr}\rangle$ and $D(A)$ while right restricted languages do not need rule $[L_{gr}]$.

Further simplifications are possible: $P{\Rightarrow}(A \,\&\, B) \,|\, Q$ is logically equivalent to $P{\Rightarrow}A \,|\, Q \wedge P{\Rightarrow}B \,|\, Q$. This rule can be added when Q is empty since it may lead to right-restricted formulae. Rule $\langle U_{gr}\rangle$ must be applied only once for each subformula $\neg(P{\Rightarrow}Q)$, no matter its prefix. In a similar way rule $\langle A\rangle$ can be skipped if a prefixed formula $\sigma.A.n : \neg s$ is already present etc.

Definition 4. A branch $\langle \mathcal{B}, G_{\mathcal{B}}\rangle$ is *closed* if $\mathcal{B}$ contains both $\sigma : s$ and $\sigma : \neg s$, for some s and σ. It is *open* if all possible rules have been applied and it is not closed. A tableau is *closed* if all branches are closed; it is *open* if at least one branch is open.

Definition 5. A *validity tableau proof* for statement s with global axioms G is a closed tableau starting with the branch $\langle\{1 : \neg s\}, G\rangle$.

In a dual way a satisfiability witness is any open branch of the tableau starting with $\langle\{1 : s\}, G\rangle$, when the calculus is complete for the fragment at hand.

Theorem 6 (Strong Soundness). *If s has a tableau proof with global axioms G then s is a logical consequence of G.*

We give a completeness result only for two main fragments (as noted in §3 the second is the most important from an applicative point of view).

Theorem 7 (Strong WL-Completeness). *If s is a logical consequence of G and $G \cup \{\neg s\}$ are weakly left restricted then s has a proof.*

Theorem 8 (Strong WRR-Completeness). *If s is a logical consequence of G and $G \cup \{\neg s\}$ are weakly right and request restricted then s has a proof.*

Remark. A decision method (rather than a semidecidable procedure which could be based on first order translations) is important for security analysis because satisfiability gives information on security weaknesses.

(a)	$1 : \neg(\neg(A\ \mathtt{says}\ \bot)\ \supset\ (A\ \mathtt{controls}\ P\Rightarrow A))$	
(b)	$1 : \neg(A\ \mathtt{says}\ \bot)$	by α rules from (a)
(c)	$1 : \neg A\ \mathtt{controls}\ (P\Rightarrow A)$	
(d)	$1 : A\ \mathtt{says}\ (P\Rightarrow A)$	by reducing `controls` from (c)
(e)	$1 : \neg(P\Rightarrow A)$	
(f)	$2 : P\ \mathtt{says}\ x_1$	by $\langle U_{gr}\rangle$ from (e)
(g)	$2 : \neg(A\ \mathtt{says}\ x_1)$	
(h)	$1.A.3 : \neg\bot$	by $\langle A\rangle$ from (b)
(i)	$1.A.3 : P\Rightarrow A$	by $[A]$ from (d)
(l)	$G_1 := \{P\Rightarrow A\}$	by $[U_{gr}]$ from (i)
(m)	$2 : P\Rightarrow A$	by $Glob$ from G_1
	$\times$	contradiction between (e, m)

Fig. 5. Tableaux Proof of an Hand-off Axiom

For a *decision method* a simple condition, checkable in polynomial time, can be imposed on the global axioms and the consequence s. Associate a graph to the global axioms and the negation of the formula to be proved: each atomic principal is represented by a node and for every $P\Rightarrow Q$, under the scope of an even number of negation, draw an arc from the atomic principals in P to those in Q. If this graph is *acyclic* then the tableau construction terminates by using loop checking with an extended notion of the Fisher-Ladner closure. Notice that, in the embedding of the word problem (§3), the principal Gr creates cycles.

In access control, acyclicity is not a restriction but rather a requirement [10, 23]: if $\Rightarrow$ is used for hierarchies of groups/roles then cycles are not allowed.

5 Examples

For sake of simplicity, we assume that we have a direct (obvious) rule for `controls` rather than translating it back to $\wedge$ and $\neg$. A first example is the derivation of an *hand-off axiom* [1, 19].

$$\neg(A\ \mathtt{says}\ \bot)\ \supset\ (A\ \mathtt{controls}\ (P\Rightarrow A)).$$

Such axioms are used by principals to hand-over their privileges in [1, 19]. Notice that, although valid, they cannot be proved within the Hilbert system developed by [1] and are added as axioms. The tableau derivation is shown in Fig. 5. The key step is rule $[U_{gr}]$ which cannot be axiomatised.

To check that global axioms must be associated to a branch, try the (satisfiable) formula below (or in §3) with only one set of global axioms as in [11, 21].

$$\neg(A \,\&\, B\ \mathtt{says}\ \bot) \wedge A\ \mathtt{says}\ (B\Rightarrow A) \wedge A\ \mathtt{says}\ r \wedge B\ \mathtt{says}\ \neg r$$

For a "real-life" deduction (Fig. 6) we take delegation without certificates from [1, page 719] where a careful (and non trivial) Hilbert proof is given.

$$
\begin{array}{lll}
(a)\ 1:\neg\,(K_B\,\texttt{says}\,(Sc\,\texttt{says}\,r)\wedge K_S\,\texttt{says}\,(K_B\Rightarrow B)\wedge(B\,|\,A)\,\texttt{controls}\,r\supset r) & & \\
(b)\ 1:K_B\,\texttt{says}\,(Sc\,\texttt{says}\,r) & \alpha \text{ rules from } (a) & \\
(c)\ 1:K_S\,\texttt{says}\,(K_B\Rightarrow B) & & \\
(d)\ 1:B\,|\,A\,\texttt{controls}\,r & & \\
(e)\ 1:\neg r & & \\
(f)\ 1:K_S\Rightarrow S & \text{by } \mathit{Glob} & \\
(g)\ 1:S\,\texttt{says}\,(K_B\Rightarrow B) & \text{by } [L_{gr}] \text{ from } (c),(f) & \\
(h)\ 1:S\,\texttt{controls}\,(K_B\Rightarrow B) & \text{by } \mathit{Glob} & \\
\qquad\qquad\qquad\swarrow\ \searrow & & \\
(h)\ 1:\neg S\,\texttt{says}\,(K_B\Rightarrow B) & (i)\ 1:K_B\Rightarrow B & \text{red } \texttt{controls} \text{ from } (h) \\
\quad\times & (l)\ 1:B\,\texttt{says}\,(Sc\,\texttt{says}\,r) & \text{by } [L_{gr}] \text{ from } (b),(i) \\
\qquad\qquad\qquad\swarrow\ \searrow & & \\
\qquad\qquad (m)\ 1:r & (n)\ 1:\neg(B\,|\,A\,\texttt{says}\,r) & \text{red } \texttt{controls} \text{ from } (d) \\
\qquad\qquad\quad\times & (o)\ 1:\neg B\,\texttt{says}\,(A\,\texttt{says}\,r) & \text{by } \langle \mathit{quote}\rangle \text{ from } (n) \\
 & (p)\ 1.B.2:\neg A\,\texttt{says}\,r & \text{by } \langle B\rangle \text{ from } (o) \\
 & (q)\ 1.B.2:Sc\,\texttt{says}\,r & \text{by } [B] \text{ from } (l) \\
 & (r)\ 1.B.2:Sc\Rightarrow A & \text{by } \mathit{Glob} \\
 & (s)\ 1.B.2:A\,\texttt{says}\,r & \text{by } [L_{gr}] \text{ from } (q),(r) \\
 & \qquad\times &
\end{array}
$$

Fig. 6. Tableau Proof of Delegation without Certificates

Example. "A delegates to B who makes requests to C. For instance A may be a user with a sufficiently powerful smartcard, B a workstation and C a file server. [...] When B wishes to make a request r on A's behalf, B sends the signed requests along with A's name...in the format $K_B\,\texttt{says}\,(A\,\texttt{says}\,r)$...When C receives the request r he has evidence that B has said A has requested r but not that A has delegated to B; then C consults the ACL [Access Control List] for request r and determines whether the request should be granted. [...] A certification authority provides the certificates for the principals' public keys as needed. The necessary certificates are $K_S\,\texttt{says}\,(K_A\Rightarrow A)$ and $K_S\,\texttt{says}\,(K_B\Rightarrow B)$, where K_S is S's public key."

We add a level of indirection to the original problem (by modelling explicitly the smartcard Sc) and use the logic for the reasoning of the server C. The set of global axioms and the statement to be proven are:

$$G \doteq \{K_S\Rightarrow S, S\,\texttt{controls}\,(K_B\Rightarrow B), Sc\Rightarrow A\}$$
$$s \doteq (K_B\,\texttt{says}\,(Sc\,\texttt{says}\,r)\wedge K_S\,\texttt{says}\,(K_B\Rightarrow B)\wedge(B\,|\,A)\,\texttt{controls}\,r)\ \supset\ r$$

In Fig. 6 only $[L_{gr}]$-rule is used. A derivation with only $\langle R_{gr}\rangle$-rule is possible.

This is an example of the "incompilability" of $P\Rightarrow Q$ into axiom schemas (or rules). Indeed $K_B\Rightarrow B$ corresponds to $B^{\mathcal{I}}\subseteq K_B^{\mathcal{I}}$, i.e. $K_B\,\texttt{says}\,s\supset B\,\texttt{says}\,s$ but it is not always valid! It depends on the server's statements i.e. $S\,\texttt{says}\,(K_B\Rightarrow B)$.

Without the server's certificate i.e. with a different theorem, it doesn't hold. The possibility of adding "on-line" properties is critical here since delegations and groups membership depend on the security policy and the current certificates.

6 Soundness and (Partial) Completeness

To prove soundness we map prefixes to states and show that satisfiability is preserved by tableaux rules. After this key lemma, the rest is standard [11].

Definition 9. Let $\mathcal{B}$ be a set of prefixed formulae and $\langle W, \mathcal{I}\rangle$ a model, a *mapping* is a function $\imath()$ from prefixes to states s.t. for all σ and $\sigma.A.n$ present in $\mathcal{B}$ it is $\langle \imath(\sigma), \imath(\sigma.A.n)\rangle \in A^{\mathcal{I}}$.

Definition 10. A tableau branch $\langle \mathcal{B}, G_{\mathcal{B}}\rangle$ is *satisfiable* (SAT for short) in the model $\langle W, \mathcal{I}\rangle$ if for every $s_g \in G_{\mathcal{B}}$ and every $w \in W$ it is $w|\!|\!\!-s_g$ and there is a mapping $\imath()$ such that for every $\langle \sigma : s_b\rangle$ present in $\mathcal{B}$ it is $\imath(\sigma)|\!|\!\!-s_b$. A tableau is SAT if one branch is such.

Theorem 11 (Safe Extension). *If $\mathcal{T}$ is a SAT tableau, then the tableau $\mathcal{T}'$ obtained by an application of a tableau rule is also SAT.*

Proof. By induction on the rules applied as in [11, Chapter 8] or [9, 14, 21]. The $\Rightarrow$ operator is the only new case.

Suppose that $\langle \sigma : P{\Rightarrow}Q\rangle$ occurs in some SAT branch. Then there must be a model $\langle W, \mathcal{I}\rangle$ and a mapping $\imath()$ such that $\imath(\sigma)|\!|\!\!-P{\Rightarrow}Q$. Hence $(P{\Rightarrow}Q)^{\mathcal{I}} = W$ for the global property of $\Rightarrow$(it is not empty as it contains $\imath(\sigma)$). Therefore adding it to $G_{\mathcal{B}}$ as done by the $[U_{gr}]$ does not change the satisfiability of the branch wrt the $G_{\mathcal{B}}$ condition (and future applications of the *Glob*-rule).

If $\langle \sigma : \neg(P{\Rightarrow}Q)\rangle$ is present then by hp $\imath(\sigma)|\!|\!\!-\neg(P{\Rightarrow}Q)$. Hence there is a $\langle w, w^*\rangle \in Q^{\mathcal{I}}$ with $\langle w, w^*\rangle \notin P^{\mathcal{I}}$. Set $\imath(n) = w$ for the new prefix n and $(x_i)^{\mathcal{I}} = W - \{w^*\}$ for the new x_i. Clearly $\imath(n)|\!|\!\!-P\,\mathtt{says}\,x_i$ but $\imath(n)|\!|\!\!\not-(Q\,\mathtt{says}\,x_i)$. □

For completeness we apply a systematic and fair procedure (see [11, 14] for a definition) and use an open branch to construct a model. A key property the procedure must guarantee is *downward saturation* [11, 21]: all applicable rules must have been applied. The proof is given for weakly left restricted statements and its extension is discussed.

Theorem 12 (Model Existence). *If $\langle \mathcal{B}, G_{\mathcal{B}}\rangle$ is an open branch with weakly left restricted statements only, then there is a model $\langle W, \mathcal{I}\rangle$ on which it is SAT.*

Proof. Construct a pre-model $\langle W, \mathcal{I}_0\rangle$ as follows:

$$\begin{aligned} W &\doteq \{\sigma \mid \sigma \text{ is present in } \mathcal{B}\} \\ A^{\mathcal{I}_0} &\doteq \{\langle \sigma, \sigma.A.n\rangle \mid \sigma \text{ and } \sigma.A.n \text{ are present in } \mathcal{B}\} \\ r^{\mathcal{I}_0} &\doteq \{\sigma \mid \sigma : r \in \mathcal{B}\} \end{aligned}$$

Incorporate the constraints due to $\Rightarrow$ and build $\mathcal{I}$ from $\mathcal{I}_0$ as follows:

– for every formula $\sigma : A \Rightarrow P$ occurring in $\mathcal{B}$
  • compute $P^{\mathcal{I}}$;
  • if $\langle \sigma, \sigma^* \rangle \in P^{\mathcal{I}}$ then add $\langle \sigma, \sigma^* \rangle$ to $A^{\mathcal{I}}$;
– repeat until a fix-point is reached.

After this *closure phase* we must prove that if $\langle \sigma : s \rangle \in \mathcal{B}$ then $\imath(\sigma) \Vdash s$ by induction on the construction of s, where $\imath(\sigma) = \sigma$. The proof is similar to those used for PDL [9] or modal logics [11, 14, 21].

The difficult case is $\langle \sigma : A \,\mathsf{says}\, s \rangle$ since we must prove that for all prefixes $\langle \sigma, \sigma^* \rangle \in A^{\mathcal{I}}$ it is $\langle \sigma^* : s \rangle$ in $\mathcal{B}$ so that we can apply induction to $\langle \sigma^* : s \rangle$, get $\sigma^* \Vdash s$ and then the claim.

The difference with "traditional" proofs [11, 21] is that some prefixes σ^* are introduced in $A^{\mathcal{I}}$ during the closure phase. Hence we can have (i) $\sigma^* = \sigma.A.n$ for some n or (ii) $\langle \sigma, \sigma^* \rangle \in P^{\mathcal{I}}$ before the closure phase and $P^{\mathcal{I}} \subseteq A^{\mathcal{I}}$ afterwards.

The first case is standard [11, 21] whereas for the second case we use the saturation of the branch and a double induction: on the formula size and on the iterations of the closure phase needed to enter $\langle \sigma, \sigma^* \rangle$ into $A^{\mathcal{I}}$. For the base case we use the following result, proven by induction on P and s as in [9]:

Proposition 13. *Before each iteration step, if $\langle \sigma : P \,\mathsf{says}\, s \rangle$ is present in $\mathcal{B}$ and $\langle \sigma, \sigma^* \rangle \in P^{\mathcal{I}}$ then $\langle \sigma^* : s \rangle$ is present in $\mathcal{B}$*

For the induction step observe that whenever $\langle \sigma : A \Rightarrow P \rangle$ is present then by saturation $\langle \sigma : A \,\mathsf{says}\, s \rangle$ implies $\langle \sigma : P \,\mathsf{says}\, s \rangle$. So apply Prop. 13 to get $\langle \sigma^* : s \rangle$ and therefore when σ^* was added in $A^{\mathcal{I}}$ in the closure phase also $\langle \sigma^* : s \rangle$ was present. Now apply the induction hypothesis.

The local condition for $\Rightarrow$ is satisfied by construction. For the global condition we use mutual saturation between *Glob* and $[U_{gr}]$ rules. If $\sigma : A \Rightarrow P$ occurs in $\mathcal{B}$ then $[U_{gr}]$ implies that $A \Rightarrow P \in G_{\mathcal{B}}$. By *Glob* we have that for all $s \in G_{\mathcal{B}}$ and all prefixes σ in $\mathcal{B}$ it is $\langle \sigma : s \rangle$. Hence every σ satisfies the local condition i.e. W satisfies the global condition and $P^{\mathcal{I}} \subseteq A^{\mathcal{I}}$. □

For the right and request restricted fragment of the language the key point is that we only have literals or statements of the form $P \Rightarrow A$ under the scope of says. The operator $\Rightarrow$ does not create problems given its global nature and the only difficult part is due to literals l (r or $\neg r$).

The previous proof for $\langle \sigma : A \,\mathsf{says}\, s \rangle$ does not work since $\langle \sigma : P \,\mathsf{says}\, s \rangle$ for non atomic P cannot propagate over $\langle \sigma : P \Rightarrow A \rangle$. However we can prove the dual of the induction step: if $\langle \sigma : \neg(A \,\mathsf{says}\, l) \rangle$ and $P \Rightarrow A$ is also present then there is is a $\langle \sigma, \sigma^* \rangle \in P^{\mathcal{I}}$ such that $\sigma^* : \neg l$. This means that all $\langle \sigma : P \,\mathsf{says}\, l' \rangle$ are consistent with each $\langle \sigma : \neg(A \,\mathsf{says}\, l) \rangle$. At this stage we need to use the $D(A)$-rule, to prove that those P statements are consistent also with each $\langle \sigma : A \,\mathsf{says}\, l'' \rangle$. By $D(A)$ we obtain $\langle \sigma : \neg(A \,\mathsf{says}\, \neg l'') \rangle$ and then apply the dual property.

Since all l, l', l'' are literals this is enough: all $A \,\mathsf{says}\, l$ are consistent among themselves, and each of them with all $P \,\mathsf{says}\, l'$. This means that when we add a $\langle \sigma, \sigma^* \rangle$ from $A^{\mathcal{I}}$ to $P^{\mathcal{I}}$ in the closure phase we can always extend the valuation of the unspecified l' or l'' in σ^* so that the result is still a model. Again, this only works for *request restricted* statements.

7 Conclusion

The major contribution of this paper is the development of a tableaux method for the calculus of access control of Abadi, Lampson et al. [1, 19, 31]. We have also clarified some model theoretic features of the calculus that makes difficult its axiomatic characterisation. The completeness results presented here extend those in [1] and provide the basis for a full fledged automatisation.

This tableaux method requires novel techniques such as passing from a tree-like tableau to a forest of prefixes, a run time update of global axioms and the corresponding modification of the notion of branch. Future research is in the direction of providing a fully automated verifier, possibly using the results of [3].

As an aside, these tableaux techniques can be used for multi modal logics with the universal modality [13] and a sound and complete calculus is presented in the appendix. Therefore we can distinguish, in proof theoretic terms, between $S5$ and the universal modality (on non tree models).

A claim that we do *not* make is that logic and semantic tableaux should be used for run-time decisions on access control. Although possible, this may lead to unacceptable slow-downs. Logic and tableaux (or similar logic-based methods) should be used for *verification* and prototyping, for checking that access protocols respect security policies. This work is a step in this direction.

Acknowledgements

I would like to thank L. Paulson and the Computer Laboratory for their hospitality in Cambridge, M. Abadi, the Computer Security group (Cambridge), the Applied Logic group (IRIT) and the anonymous referees for many suggestions which helped to improve this paper. This research has been partly supported by ASI, CNR and MURST 40% and 60% grants and by EPSRC grant GR/K77051 "Authentication Logics".

References

1. M. Abadi, M. Burrows, B. Lampson, and G. Plotkin. A calculus for access control in distributed systems. *ACM Trans. on Prog. Lang. and Sys.*, 15(4):706–734, 1993.
2. R. Anderson. A security policy model for clinical information systems. In *Proc. of the 15th IEEE Symp. on Security and Privacy*. IEEE Comp. Society Press, 1996.
3. B. Beckert and R. Goré. Free variable tableaux for propositional modal logics. In THESE PROCEEDINGS, 1997.
4. D. Bell and L. La Padula. Secure computer systems: unified exposition and MULTICS. Report ESD-TR-75-306, The MITRE Corporation, March 1976.
5. M. Burrows, M. Abadi, and R. Needham. A logic for authentication. *ACM Trans. on Comp. Sys.*, 8(1):18–36, 1990. Also available as Res. Rep. SRC-39, DEC - System Research Center, 1989.
6. M. Castilho and A. Herzig. An alternative to the iteration operator of propositional dynamic logic. Tech. Rep. 96-05-R, IRIT (Toulouse), Univ. Paul Sabatier, jan 1996.

7. D. Clark and D. Wilson. A comparison of commercial and military computer security policies. In *Proc. of the 6th IEEE Symp. on Security and Privacy*, pp. 184–194. 1987.
8. F. Cuppens and R. Demolombe. A deontic logic for reasoning about confidentiality. In *3rd Int. Workshop on Deontic Logic in Computer Science*, Sesimbra, Portugal, 1996.
9. G. De Giacomo and F. Massacci. Tableaux and algorithms for propositional dynamic logic with converse. In *Proc. of the 13th Int. Conf. on Automated Deduction (CADE-96)*, LNAI 1104, pp. 613–628, 1996.
10. D. Ferraiolo, J. Cugini, and K. Richard. Role-based access control (rbac): Features and motivations. In *Proc. of the Annual (Computer Security Applications Conf.*, 1995.
11. M. Fitting. *Proof Methods for Modal and Intuitionistic Logics*. Reidel, 1983.
12. V. Goranko. Modal definability in enriched languages. *Notre Dame J. of Formal Logic*, 31(1), 1990.
13. V. Goranko and S. Passy. Using the universal modality: Gains and questions. *J. of Logic and Computation*, 2(1):5–30, 1992.
14. R. Goré. Tableaux method for modal and temporal logics. Tech. Rep. TR-ARP-15-5, Australian National Univ., 1995.
15. J. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54:319–379, 1992.
16. G. Hughes and M. Cresswell. *a Companion to Modal Logic*. Methuen, 1984.
17. C. Krogh. Obligations in multiagent systems. In *5th Scandinavian Conference on Artificial Intelligence (SCAI-95)*, pp. 29–31. ISO Press, 1995.
18. B. Lampson. Protection. *ACM Operating Sys. Reviews*, 8(1):18–24, 1974.
19. B. Lampson, M. Abadi, M. Burrows, and E. P. Wobber. Authentication in distributed systems: Theory and practice. *ACM Trans. on Comp. Sys.*, 10(4):265–310, 1992.
20. B. Marick. The `verus` design verification system. In *Proc. of the 2nd IEEE Symp. on Security and Privacy*, pp. 150–157, 1983.
21. F. Massacci. Strongly analytic tableaux for normal modal logics. In *Proc. of the 12th Int. Conf. on Automated Deduction (CADE-94)*, LNAI 814, pp. 723–737, 1994.
22. J. Moffet and M. Sloman. Policy hierarchies for distributed systems management. *IEEE J. on Selected Areas in Communications*, 11(9), 1993.
23. R. Sandhu, E. Coyne, H. Feinstein, and C. Youman. Role-based access controls models. *IEEE Computer*, 29(2), February 1996.
24. R. Sandhu. The typed access matrix model. In *Proc. of the 11th IEEE Symp. on Security and Privacy*, pp. 122–136, 1992.
25. R. Sandhu and P. Samarati. Access control: Principles and practice. *IEEE Communications Magazine*, pp. 40–48, September 1994.
26. M. Schmidt-Schauss. Subsumption in KL-ONE is undecidable. In *Proc. of the 1st Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR-89)*, pp. 421–431, 1989.
27. S. Schmitt and C. Kreitz. Converting non-classical matrix proofs into sequent-style systems. In *Proc. of the 13th Int. Conf. on Automated Deduction (CADE-96)*, LNAI 1104, pp. 418–432, 1996.
28. P. F. Syverson and P. C. van Oorschot. On unifying some cryptographic protocols logics. In *Proc. of the 13th IEEE Symp. on Security and Privacy*. IEEE Comp. Society Press, 1994.

29. J. van Benthem. Correspondence theory. In *Handbook of Philosophical Logic*, volume II. Reidel, 1986.
30. R. van der Meyden. The dynamic logic of permission. *J. of Logic and Computation*, 6(3):465–479, 1996.
31. E. Wobber, M. Abadi, and M. Burrows. Authentication in the Taos operating system. *ACM Trans. on Comp. Sys.*, 12(1):3–32, 1994.

A Tableaux for the Universal Modality

We can easily define a tableaux calculus for multi modal logics with the universal modality: use A **says** s as $[A]s$ to get the logic K_n or more precisely an almost PDL variant (close to [6] although they have a weaker common knowledge modality); add the Single Step Tableaux (SST) rules for the others modal logics of knowledge and belief between K_n and $S5_n$ [14, 21]; finally use a modified version of rule $\langle U_{gr}\rangle$ and $[U_{gr}]$ described below.

$$\langle u\rangle : \frac{\sigma : \neg[u]s}{n : \neg s}\ n \text{ new} \qquad [u] : \frac{\sigma : [u]s}{G_{\mathcal{B}} := G_{\mathcal{B}} \cup \{s\}}$$

Theorem 14 (Universal Modality). *If $\mathcal{R}_1, \ldots, \mathcal{R}_n$ are sound and complete SST rules for the (multi) modal logics $\mathcal{L}_1, \ldots, \mathcal{L}_n$ [14, 21] then the tableaux calculus enhanced with rules $[u]$ and $\langle u\rangle$ is sound and complete for multi modal logic $\mathcal{L}_1 \ldots \mathcal{L}_n$ with the universal modality.*

For the soundness part we replace $\sigma : P \Rightarrow Q$ with $\sigma : [u]s$ and $P \Rightarrow Q$ with s in the corresponding argument of Thm. 11. For completeness, the mutual induction between the application of the *Glob*-rule and the $[u]$-rule is identical to Thm. 12.

It is possible to "distinguish the (axiomatically) indistinguishable" i.e. $S5$ from the universal modality in proof theoretic terms (closed vs open branches for some formulae). Of course this distinction can only be done as satisfiability on k-clusters models for $k \geq 2$, since $S5$ and $[u]$ cannot be distinguished by validity [12, 13, 29] and thus by traditional tableaux (and 1-cluster models).

For instance, with the $S5$-rules for $[A]$ and $[B]$ given in [14, 21]:

$$\Phi_{SAT} \doteq \langle B\rangle[A]r \wedge \langle B\rangle[A]\neg r \qquad \Phi_{UNSAT} \doteq \langle B\rangle[A]r \wedge \langle B\rangle[u]\neg r$$

have different tableaux: one open, and the second closed. So replacing an occurrence of $[A]$ with $[u]$ changes the satisfiability of a formula. Equally one can combine $[A]$ only with $[u]$ and obtain different tableaux.

We can also derive *a small model theorem* by adapting loop checking techniques from [9, 14] and find a model (if any) of size at most $O(|G \cup \{s\}| \times 2^{|G \cup \{s\}|})$.

This article was processed using the LaTeX macro package with LLNCS style