1-1-2010

Permission to Speak: A Logic for Access Control and Conformance

Nikhil Dinesh
University of Pennsylvania, nikhild@seas.upenn.edu

Aravind Joshi
University of Pennsylvania, joshi@seas.upenn.edu

Insup Lee
University of Pennsylvania, lee@cis.upenn.edu

Oleg Sokolsky
University of Pennsylvania, sokolsky@seas.upenn.edu

Permission to Speak: A Logic for Access Control and Conformance

Abstract
Formal languages for policy have been developed for access control and conformance checking. In this paper, we describe a formalism that combines features that have been developed for each application. From access control, we adopt the use of a saying operator. From conformance checking, we adopt the use of operators for obligation and permission. The operators are combined using an axiom that permits a principal to speak on behalf of another. The combination yields benefits to both applications. For access control, we overcome the problematic interaction between hand-off and classical reasoning. For conformance, we obtain a characterization of legal power by nesting saying with obligation and permission.

The axioms result in a decidable logic. We integrate the axioms into a logic programming approach, which lets us use quantification in policies while preserving decidability of access control decisions. Conformance checking, in the presence of nested obligations and permissions, is shown to be decidable. Non-interference is characterized using reachability via permitted statements.

Keywords
Access control; Conformance; Modal logic; Deontic logic

Disciplines
Computer Engineering | Engineering

Permission to Speak: A Logic for Access Control and Conformance[☆]

Nikhil Dinesh[a], Aravind Joshi[a], Insup Lee[a], Oleg Sokolsky[a]

[a] *Department of Computer Science*
University of Pennsylvania
Philadelphia, PA 19104-6389, USA

Abstract

Formal languages for policy have been developed for access control and conformance checking. In this paper, we describe a formalism that combines features that have been developed for each application. From access control, we adopt the use of a saying operator. From conformance checking, we adopt the use of operators for obligation and permission. The operators are combined using an axiom that permits a principal to speak on behalf of another. The combination yields benefits to both applications. For access control, we overcome the problematic interaction between hand-off and classical reasoning. For conformance, we obtain a characterization of legal power by nesting saying with obligation and permission.

The axioms result in a decidable logic. We integrate the axioms into a logic programming approach, which lets us use quantification in policies while preserving decidability of access control decisions. Conformance checking, in the presence of nested obligations and permissions, is shown to be decidable. Non-interference is characterized using reachability via permitted statements.

1. Introduction

Access control is an important problem in trust management systems. Informally, a trust management system involves a set of actors or principals, and a set of controlled or regulated actions, e.g., accessing medical information, or downloading a song. The goal of such a system is to administrate requests to perform actions. Trust management systems are commonly decomposed into two (interacting) components [1]: (a) *authentication* - determining the source of a request, and (b) *access control* - determining whether a request is permitted

[☆]This research was supported in part by ONR MURI N00014-07-1-0907, NSF CCF-0429948, and ARO W911NF-05-1-0158.

Email addresses: nikhild@seas.upenn.edu (Nikhil Dinesh), joshi@seas.upenn.edu (Aravind Joshi), lee@seas.upenn.edu (Insup Lee), sokolsky@seas.upenn.edu (Oleg Sokolsky)

according to a policy. Abadi et al. [3] cast access control as a problem for logic. We assume as given an action (p), which is controlled by a principal (A), and a request to perform p from a principal (B). Access is granted if we can prove, using A's policy, that A *says* that B is *permitted* to perform p. In access control logics, such as [1–3, 16, 17], *says* is treated as a (modal) operator. However, the use of an operator for *permission* has not been explored.

The concept of *representation* is prevalent in access control policies, and it forms the central focus of this work. Representation arises in situations where a principal is held to declarations made on her behalf (cf. [18]). For example, consider a scenario where a software company authorizes project managers to permit their team members to access the production server. If a project manager says that a team member is permitted to access the server (on behalf of the company), we conclude that the company says that the team member is permitted to access the server. In such a scenario, project managers *represent* the company on permitting access to the server. All access control logics provide principals with the capability to let other principals represent them on statements. In the example above, the company would *say*, in its policy, that "Project managers represent the company on permitting team members to access the production server". The manner in which such a policy is formally expressed depends on the logic, and we will discuss a few choices in later sections.

In this paper, we argue for an explicit account of *permission* in a logic for access control. We motivate and develop a logic that combines *saying* and *permission*, using an axiom that permits a principal to speak on behalf of another. The combination leads us to a novel account of *representation*. In the logics of saying, where there is no notion of permission, representation is accommodated using variants of the hand-off axiom [1]. Abadi [1] pointed out some problematic interactions between the hand-off axiom and classical reasoning, which we will discuss in detail in Section 2.1. The use of permission provides a way to avoid these problems. An explicit account of permission leads naturally to an explicit account of *obligation*, which in turn leads us to examine *legal powers* and *conformance checking*. We now introduce these topics.

Legal Power: Representation is a special case of the broader concept of *legal power*. Hohfeld, in his seminal work, defined the concept of power as follows [31, Page 44]:

> A person (or persons) may be said to have the power to effect a change in legal relations, if the change in legal relations results from some superadded facts that are under his volitional control.

We decompose this definition into three components, to give the main intuitions for our approach:

1. The description of power - A principal (A) grants the power of representation to another principal (B) on certain statements, if A *says* that B is *permitted* to issue those statements (on her behalf).
2. The "superadded facts" by which a power is exercised - B exercises the power of representation by issuing statements on behalf of A.

3. The change in legal relations - If A grants the power of representation to B, and B exercises this power, then we will infer that the statement issued by B is issued by A as well.

The logical analysis of power has been of interest for several years [18, 34–36, 40]. Our approach is related to two lines of research. With regard to the description of power (Item 1 above), Lindahl [40, Part II] (see also [34]) suggested that various notions of power can be distinguished by nesting obligations and permissions with an action modality. *Saying* is our analog of the action modality. With regard to the change in legal relations (Item 3 above), Jones and Sergot [35] and Gelati et al. [18] describe general frameworks to reason about situations where an act by a principal *counts as* a means to create a state of affairs within an institution. We consider a restricted scenario where a statement by one principal *counts as* an identical statement made by another principal. However, in [18, 35], the concept of *counts as* is taken to be the description of power itself, and it does not arise via *saying*. The dependence of power on *saying*, in our approach, leads us to a novel analysis of recursive notions of power, e.g., "empowerment to empower". We discuss the differences in Section 2.2.

Conformance Checking: The problems of access control, representation, and power arose for us while extending our prior work [14] on conformance checking to privacy regulation. We introduce the problem of conformance here, and discuss how the ideas in [14] relate to this work in Section 2.3. In conformance, one is interested in checking whether the operations of organizations obey a policy. We are given a policy and a description of an organization's operations (as a state or trace). An organization (A) conforms to the policy if we can prove that for all p, if the policy *says* that A is *required* or *obligated* to do p, then A does p. The design of logics for conformance, notably deontic logic, has been of interest for several years, and we refer the reader to [33, 45] for a broad perspective. In recent years, the focus has been more on tailoring logics for the regulations at hand, and examples include business contracts [4, 19–21, 27, 38] and health-care regulations [10, 14]. Our focus in this work is on how *power* interacts with the question of *conformance*.

Contributions and Outline: In this paper, we motivate and design a formalism that combines *saying* and *permission*, with applications to access control and conformance. The combination yields benefits to both applications:

1. For access control, we propose a new decidable axiomatization which accommodates delegation [3, 39] and "speaking for" [2, 3, 16]. Our approach overcomes the problematic interactions with classical reasoning, pointed out by Abadi [1]. "Speaking for" and delegation are obtained as consequences of an axiom that permits a principal to speak on behalf of another.
2. For conformance, the proposed axiomatization is used to reason about declarative powers [18], by nesting saying with obligation and permission. We obtain a novel analysis of recursive notions of power, e.g., "empowerment to empower". Conformance, as the satisfaction of obligations, is shown to be decidable.

In Section 2, we give a detailed motivation and background for our approach in three parts. First, we consider representation in access control, under which we include delegation [3, 39] and "speaking for" [1, 3, 16]. Second, we discuss examples of powers, and compare our approach to the *counts as* frameworks for power [18, 35]. And, finally, we discuss how we integrate the work here with our prior work [14].

Section 3 develops a logic in the form of two interacting components. *The inference component* determines what has been said, and involves the choice of appropriate axioms [1, 3, 17]. We introduce two axioms to characterize the interaction between saying and permission. The decidability of the resulting logic is established. *The saying component* is used to create new utterances. For this component, we extend the formalism in [14], which is a generalized form of logic programming. The modularization allows us to use restricted forms of quantification while preserving decidability of access control and conformance. We also prove a non-interference property which is crucial for the distributed policies that arise in access control.

In Section 4, we discuss our formalism in the context of related work. We consider access control examples, and conformance in the presence of powers. We also identify some interesting lines for further research. Section 5 concludes.

2. Permission to Speak

In this section, we motivate the explicit use of saying and permission in a formal language for policy. Section 2.1 considers the problem of representation in access control, under which we include delegation [3, 39] and "speaking for" [1, 3, 16]. In Section 2.2, we discuss examples of powers conveyed by nested permissions. We compare and contrast our approach with the *counts as* approaches to power. Finally, we discuss how we integrate the work here with our prior work [14], to clarify some methodological decisions (Section 2.3).

2.1. Representation in Access Control

While there are a wide variety of access control logics, one commonality that stands out is a notion of *saying* [1]. We can express the fact that a principal makes a statement. We use $\text{says}_{l(A)}\varphi$ to denote that principal A says φ in the set of laws $l(A)$. Informally, a law is understood as a single statement in the policy of a principal, e.g., a hospital says "Alice is permitted to access her health information", in its policy. And, the interpretation of a set of laws is the conjunction of the individual laws. These intuitions are formalized in Section 3. Our approach differs from others in that we associate statements to a principal via a set of laws ($\text{says}_{l(A)}\,\varphi$) rather than directly with the principal ($\text{says}_A\,\varphi$). This indirection lets us use *saying* to reason about exceptions to laws, as in [14], and we will discuss an example in Section 3.3.

All access control logics give a principal the ability to let another principal make statements on her behalf. As an example (based on [16]), consider a file access scenario, where an administrator (A) has control the operation of deleting

files shared by groups of principals. When there are many shared files in the system, A cannot personally handle all requests. Suppose that the administrator authorizes the leader of a group (B) to decide when a particular file is to be deleted (del). In this scenario, we say that B *represents* A on del, and we wish to conclude that if $\text{says}_{l(B)}(\text{del})$, then $\text{says}_{l(A)}(\text{del})$.

How do we accommodate this inference? A naive approach is to introduce "$\text{says}_{l(B)}(\text{del}) \Rightarrow \text{says}_{l(A)}(\text{del})$" into A's policy (where $\Rightarrow$ is the implication connective of the underlying logic). However, such statements create an access control risk, because "$\text{says}_{l(B)}(\text{del}) \Rightarrow \text{says}_{l(A)}(\text{del})$" could be introduced by B, thereby giving B the ability to decide whether any file is to be deleted.

To address this security risk, a principal A is only allowed to introduce statements of the form $\text{says}_{l(A)}\,\psi$. Additional machinery (usually an axiom) is needed to accommodate representation. Abadi [1] discusses several alternatives, involving variants of the hand-off axiom:

- $\text{says}_{l(A)}(\phi \Rightarrow \text{says}_{l(A)}\,\psi) \Rightarrow (\phi \Rightarrow \text{says}_{l(A)}\,\psi)$

B represents A on del is expressed as:

- $\text{says}_{l(A)}(\text{says}_{l(B)}(\text{del}) \Rightarrow \text{says}_{l(A)}(\text{del}))$

The hand-off axiom lets us conclude that $\text{says}_{l(B)}(\text{del}) \Rightarrow \text{says}_{l(A)}(\text{del})$. However, the hand-off axiom has displeasing consequences in classical logics. For example, $\text{says}_{l(B)}\,\varphi \Rightarrow (\neg\varphi \Rightarrow \text{says}_{l(B)}\,\psi)$ (for all ψ) is provable [1], i.e., if a statement by B fails, then B gives access to all the actions that she controls. The solution to this problem has been to move to an intuitionistic setting, as in [2, 16, 17].

We suggest that the problem is not with classical reasoning, but with the hand-off axiom. The key idea is to reformulate the axiom using the interaction between *saying* and *permission*. We now introduce the reformulated version of the axiom, followed by a discussion of its benefits.

We say that B represents A on del, if A says that B is *permitted* to say del. More formally, the statement $\text{says}_{l(A)}(\mathcal{P}_B(\text{says}_{l(B)}\,\text{del}))$ is added to A's policy, where $\mathcal{P}_B(\text{says}_{l(B)}\,\text{del})$ is read as "B is permitted to say del". The following are equivalent versions of *the axiom of representation*:

- If A says that B is permitted to say φ, then if B says φ, A says φ
- $\text{says}_{l(A)}(\mathcal{P}_B(\text{says}_{l(B)}\varphi)) \Rightarrow (\text{says}_{l(B)}\,\varphi \Rightarrow \text{says}_{l(A)}\,\varphi)$

The axiom of representation is intended for a particular sense of speaking/saying, i.e., *speaking on someone's behalf*. This sense of saying is the usual one in access control. To simplify matters, we do not explicitly represent the principal on behalf of whom a statement is being made.

"Speaking for" [2, 3, 16] is a case of representation when one principal represents another on all statements. If B speaks for A, we wish to conclude $\text{says}_{l(B)}\,\varphi \Rightarrow \text{says}_{l(A)}\,\varphi$ for all φ. "Speaking for" has a compelling definition in

our approach. We say that B speaks for A if A permits B to say anything ($\bot$) on her behalf, i.e., $\text{says}_{l(A)}\, \mathcal{P}_B(\text{says}_{l(B)}\, \bot)$.

A novelty in our approach is that "speaking for" and hand-off are both obtained as a consequence of the axiom of representation. In [2, 3, 16], "speaking for" and hand-off are not related, i.e., the former involves an algebra over principals or second-order quantification, and the latter is obtained using an axiom (which implies hand-off). This suggests that the representation axiom is quite different from the hand-off axiom. It is tempting to relate the representation axiom to a restricted version of hand-off:

- $\text{says}_{l(A)}(\text{says}_{l(B)}\, \varphi \Rightarrow \text{says}_{l(A)}\, \varphi) \Rightarrow (\text{says}_{l(B)}\, \varphi \Rightarrow \text{says}_{l(A)}\, \varphi)$

However, even for this restricted case, we do not know of a complete semantics for hand-off, which makes it difficult to show that a statement is not provable (Abadi et al. [3] observe similar difficulties). We believe that the representation axiom is a persuasive alternative to hand-off, because it yields a decidable logic with a complete semantics, and more importantly, it has an intuitive interpretation.

A restricted version of the axiom of representation has been proposed by Becker et al. [7], in the context of the authorization language SECPal. In SECPal, representation is restricted to atomic predicates, and hence, "speaking for" cannot be accommodated. Moreover, the relationship between permission and obligation is not explored. Our formalism generalizes SECPal, to accommodate both "speaking for" and obligation. We now discuss further motivation for our approach.

2.2. Powers and Nested Constructions

In this section, we consider examples of powers that arise via nested obligations and permissions. We compare and contrast our approach to the *counts as* approaches to power [18, 35]. The comparison is intended to illustrate the interplay between *power* and *saying*. We then discuss an example where our approach offers only a limited analysis.

We begin by discussing our approach to nested permissions. Consider the following statement: "A hospital (H) permits a patient (A) to permit her mother (B) to access her information". We will rephrase the permission as follows: H says that A is *permitted to say* that B is permitted to access her information. Formally, this is expressed as: $\text{says}_{l(H)}(\mathcal{P}_A(\text{says}_{l(A)}(\mathcal{P}_B\text{access})))$. If A does indeed permit access to her mother ($\text{says}_{l(A)}(\mathcal{P}_B\text{access})$), we will conclude $\text{says}_{l(H)}(\mathcal{P}_B\text{access})$ using the axiom of representation, i.e., H permits access to B. As a result, nested permissions are related to representation, i.e., "H permits A to permit B to do φ" iff "A represents H in permitting B to do φ".

We now turn to the analysis by Gelati et al. [18]. To simplify presentation, we describe their approach using the notation that we have already introduced. In [18], *declarative power*, which includes representation, is defined formally in terms of a *counts as* operator/connective:

(P1) $\text{DP}_A^H(\varphi) = \text{CountsAs}(\text{says}_{l(A)}\, \varphi, \text{says}_{l(H)}\, \varphi)$

$DP_A^H(\varphi)$ is read as "H grants A the power to declare φ on its behalf". And, $\text{CountsAs}(\text{says}_{l(A)}\,\varphi, \text{says}_{l(H)}\,\varphi)$ is read as "A saying φ *counts as* B saying φ". The logic of *counts as* [18, 35] has broad applicability, and a detailed exposition is well beyond the scope of this paper. For present purposes, it suffices to note that a version of the following is provable:

(P2) $\vdash \text{CountsAs}(\text{says}_{l(A)}\,\varphi, \text{says}_{l(H)}\,\varphi) \Rightarrow (\text{says}_{l(A)}\,\varphi \Rightarrow \text{says}_{l(H)}\,\varphi)$

$\vdash \phi$ is read as "ϕ is *provable*", i.e., ϕ is a theorem of the language. The key observation here is that *power* (conveyed by *counts as*) can result in the creation of statements, using (P2). However, the converse is not true. Let us return to the example of nested permissions see why this is important. Using (P1) and (P2), we can show that:

(P3) $\vdash \text{DP}_A^H(\mathcal{P}_B\text{access}) \Rightarrow (\text{says}_{l(A)}(\mathcal{P}_B\text{access}) \Rightarrow \text{says}_{l(H)}(\mathcal{P}_B\text{access}))$

And, (P3) plays the role of the representation axiom. As a result, our approach is quite similar to that of Gelati et al. [18], when there is one level of nesting. However, differences arise when we consider one more level of nesting.

Suppose H says that A is *empowered* to *empower* B to permit C to access her information. Note that *empowerment* can be paraphrased as *permission to say* in our approach, and the analysis would proceed analogously to the previous case. Gelati et al. [18] express this empowerment to empower as: $\text{DP}_A^H(\text{DP}_B^H(\mathcal{P}_C\text{access}))$. Let $\varphi = \mathcal{P}_C\text{access}$. Using (P1) and (P2), we obtain:

(P4) $\vdash \text{DP}_A^H(\text{DP}_B^H(\varphi)) \Rightarrow (\text{says}_{l(A)}(\text{DP}_B^H(\varphi)) \Rightarrow \text{says}_{l(H)}(\text{DP}_B^H(\varphi)))$

Given $\text{DP}_A^H(\text{DP}_B^H(\varphi))$, if A exercises this power by empowering B to declare φ, we will conclude, using (P3), that $\text{says}_{l(H)}(\text{DP}_B^H(\varphi))$, i.e., H *says* that B is empowered to declare φ. However, the following is *not* provable:

(P5) $\nvdash \text{says}_{l(H)}(\text{DP}_B^H(\varphi)) \Rightarrow (\text{says}_{l(B)}\,\varphi \Rightarrow \text{says}_{l(H)}\,\varphi)$

Thus, B cannot exercise the power in the same way as A, due to the difference between (P3) and (P5). This assymetry, between primary and recursive powers, arises because the *counts as* operator is taken to be the description of power itself, and it does not arise via *saying* (or some other action).[1] The dependence of power on *saying*, in our approach, leads to an analogous treatment of primary and recursive powers.

Finally, we consider an example of nested obligations to illustrate a scenario where our approach gives only a limited analysis. We adopt the definition of obligation as the dual of permission, i.e., $\mathcal{P}_A\varphi = \neg\mathcal{O}_A\neg\varphi$ ($\mathcal{O}_A\varphi$ is read as φ is obligatory for A). Consider the following statement: "A says that B is required to forbid her child (C) from playing near the road (play)". As we did

[1] More precisely, power needs to be linked to an *institutional action*, which is *not* effective. An effective action modality ($\Box$) is one which accommodates $\vdash (\Box\varphi) \Rightarrow \varphi$, and would be unsuitable for access control.

with the nested permissions, we paraphrase it as "A says that B should *say* that C is forbidden from playing near the road". Formally, this is expressed as: $\text{says}_{l(A)}\, \mathcal{O}_{\mathcal{B}}\, \text{says}_{l(B)}\, \mathcal{O}_{\mathcal{C}} \neg \text{play}$. If B imposes this requirement by *saying* so $(\text{says}_{l(B)}\, \mathcal{O}_{\mathcal{C}} \neg \text{play})$, we will conclude that B has *fulfilled* her obligation toward A, i.e., B *conforms* w.r.t. A. Does this capture the intent of the statement? Consider an alternate paraphrase of the statement: "A says that B is required to *see to it that* C does not play". And, it may require a stronger action of B, e.g., physically preventing C from playing near the road. The analysis of such requirements is beyond the scope of this work. The action modalities in the logics of power [18, 34, 35, 40] offer a good solution.

2.3. Exceptions and Doesn't Say

The problem of exceptions to laws has been studied extensively for several years [5, 44, 51], and is related to the broader area of non-monotonic reasoning [41, 46, 47, 49]. In [14], building on Reiter's Default Logic [49] and Kripke's theory of truth [37], we expressed laws using labeled conditional statements of the form:

$$\text{(id)}\ \varphi \mapsto \psi$$

Our informal interpretation of such statements was "If φ is true, then the regulator *says* ψ via the law labeled (id)", where "id" is an identifier for the law. This interpretation of rules has the flavor of the *counts as* connective [18, 35], i.e., φ *counts as* a statement of ψ from the regulator. Now, we can consider statements of the form:

$$\text{(id1) The regulator does not say } \psi \text{ via the law labeled (id)} \mapsto \psi'$$

In other words, "If the regulator *does not say* ψ via the law labeled (id), then the regulator *says* ψ' via the law labeled (id1)". *Does not say* is useful in expressing exceptions to laws, and the law labeled (id) would serve as an exception to the law labeled (id1). We discuss examples in Section 3.3.

Exceptions make regulations non-monotonic, in the sense that adding a new exception would prevent certain conclusions that were drawn before. There are also well-established reinterpretations of non-monotonic logics as modal logics, and here, we refer the reader to some classic works on autoepistemic logic [28, 46]. Given these connections, an important question that arises is whether the underlying logic for *saying* should be *non-monotonic*. The approach we take in this work is to start with a *monotonic* logic with *saying*, *obligation*, and *permission* (Section 3.2), and then integrate it into a *non-monotonic* framework (Section 3.3). The idea is that the non-monotonic component resolves exceptions, giving us a consistent set of statements on which to base access control and conformance decisions. This aspect of our approach was motivated purely by methodological convenience, and sufficed for the regulations at hand. A proper non-monotonic treatment of nested modalities is a challenging problem (see [28]), and we leave an investigation to future work.

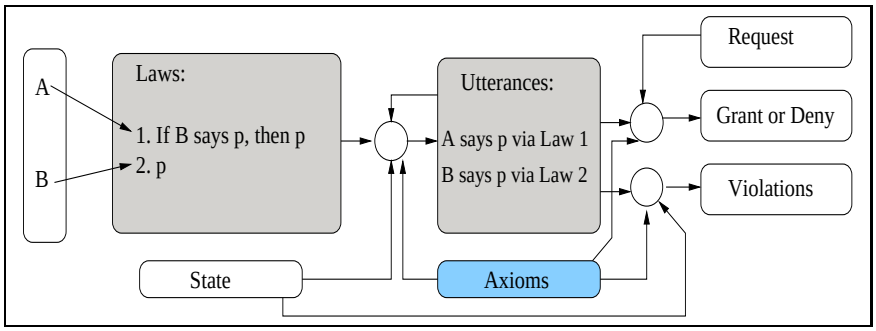

Figure 1: Interaction between the components of the logic

3. A Logic for Access Control and Conformance

In this section, we develop a logic in the form of two interacting components – (a) the inference component, which involves the choice of appropriate axioms, and (b) the saying component, which is used to represent policies. Figure 1 shows the interaction between the components of the access control system. There are two kinds of actions of interest – (1) operational acts, e.g., downloading a song, and (2) speech acts. The operational acts are described using a state, which contains the interpretation of predicates, and the speech acts are described using laws.

A principal speaks by introducing laws. In Figure 1, the principals A and B introduce the laws 1 and 2 respectively. The laws are evaluated using the axioms to produce a set of *utterances*, i.e., what the principals say via their laws. A set of laws can be thought of as a logic program, and utterances as the extensions that result from the program (via a fixed point computation). Once we have the utterances, there are several decision problems of interest. *The access control problem* is to decide whether a request is permitted by the set of utterances. *The conformance problem* is to decide whether operational and speech acts satisfy the obligations imposed by the utterances, and if they do not, violations are reported.

In Section 3.1, we introduce an example from privacy regulation, which we will use to illustrate the various definitions. Section 3.2 is an overview of the inference component. We describe (axiomatically) a logic with two modalities – *saying* and *obligation*. In Section 3.3, we adapt the formalism in [14] for the saying component. We extend [14] in two ways. First, we prove a non-interference property which is crucial for the distributed policies that arise in access control (Section 3.4). Second, we show that conformance, in the presence of nested obligations and permissions, is decidable (Section 3.5).

3.1. Example

We will use an example from the Health Insurance Portability and Accountability Act (HIPAA) (cf. [10]), to illustrate the various definitions. HIPAA regulates the uses and disclosures of patient health information, and provides a

natural test bed for investigating both access control and conformance. The following example is intended to illustrate several subtleties involved in reasoning about rights:

(1) A patient has the right to view his records that are maintained in a designated record set, except for:
   a. Psychotherapy notes.
   b. Records compiled for a legal proceeding.
   c. ...

There are three (types of) principals associated with this right:

- The regulatory authority behind HIPAA, which *enforces* the right.
- Patients, who can *exercise* the right, and
- Principals who maintain records of the patient, and have to *conform* to the right.

Section 3.2 is concerned with how to formally express the phrase "has the right". In Section 3.3, we deal with the exceptions. Sections 3.4 and Section 3.5 consider the access control and conformance aspects.

3.2. The Inference Component – Axioms

In this section, we develop a predicate logic with two modalities *saying* and *obligation*. We allow formulas with free variables, but no quantifier over objects. The quantification over objects is carried out in the process of saying (Section 3.3), which uses provability in the propositional subset of the language defined here. We begin by defining the syntax:

Definition 1 (Syntax)**.** *Given sets $\Phi_1, ..., \Phi_n$ (of predicate names), countable sets of object names O, principal names $O_P \subseteq O$, variables X, variables for principals $X_P \subseteq X$, identifiers ID, and a function $l : O_P \to 2^{ID}$, the language $L(\Phi_1, ..., \Phi_n, O, O_P, X, X_P, l, ID)$, abbreviated as L, is defined as follows:*

$$\begin{array}{l}\varphi_y ::= \alpha \mid \varphi_y \wedge \varphi_y \mid \neg\varphi_y \mid \text{says}_{Id_y}\, \psi \\ \psi_y ::= \varphi_y \mid \psi_y \wedge \psi_y \mid \neg\psi_y \mid \mathcal{O}_y\varphi_y \\ \varphi ::= \varphi_y \text{ (for all } y \in X_P \cup O_P) \mid \varphi \wedge \varphi \mid \neg\varphi \\ \psi ::= \psi_y \text{ (for all } y \in X_P \cup O_P) \mid \psi \wedge \psi \mid \neg\psi\end{array}$$

where, $y \in X_P \cup O_P$, and α generates atomic predicates of the form $p(z_1, ..., z_j)$ with $p \in \Phi_j$ and $(z_1, ..., z_j) \in (X \cup O)^j$. In addition, $\emptyset \subset Id_y \subseteq l(y)$ if $y \in O_P$ and $Id_y = l(y)$ otherwise ($y \in X_P$). We assume that for all distinct $A, B \in O_P$, $l(A) \cap l(B) = \emptyset$ and $l(A) \neq \emptyset$, i.e., the assigned identifiers are disjoint, and non-empty.

The set of formulas generated by each BNF rule are referred to as L_{φ_y}, L_φ, L_{ψ_y} and L_ψ respectively, and $L = L_\varphi \cup L_\psi$.

Disjunction $\varphi \vee \psi = \neg(\neg\varphi \wedge \neg\psi)$ and implication $\varphi \Rightarrow \psi = \neg\varphi \vee \psi$ are derived connectives.

There is a set of object names O with a distinguished set $O_P \subseteq O$ called *principals*. Principals include *individual persons*, such as patients and doctors, and *institutions*, such as hospitals and regulatory authorities. We use upper case letters for principals, e.g., A, B. Other named objects ($O - O_P$) include entities with no associated notion of agency, e.g., medical records and songs. We use lower case letters for these objects, except for the letters $\{x, y, z\}$ which are reserved for variables. It is useful to divide the objects in $O - O_P$ further into sorts, but we avoid it to simplify notation. Variables are divided into two sorts as well, i.e., all variables X, and variables for principals X_P. In a slight abuse of notation, we will use the symbols for variables, i.e., x, y and z, to stand for a generic element in $X \cup O$ or $X_P \cup O_P$.

$\mathcal{O}_y\varphi$ is read as "φ is obligatory for the principal y". Permission is defined as the dual of obligation, i.e., $\mathcal{P}_y\varphi = \neg\mathcal{O}_y\neg\varphi$.

The saying operator is understood as follows. Principals speak by introducing identified laws, as shown in Figure 1. The function l assigns non-empty and disjoint sets of identifiers to each principal, and for example, $l(A)$ denotes the set of identifiers for laws introduced by the principal $A \in O_P$. $\text{says}_{Id_y}\,\varphi$ is read as "y says ψ via the laws Id_y". In the case where $Id_y = l(y)$, $\text{says}_{l(y)}\,\psi$ is read as "y says ψ via her laws", or briefly "y says ψ".

We give some examples to clarify the notation for identifiers. Given $A \in O_P$, let $l(A) = \{1, 2\}$. The formulas $\text{says}_{l(A)}\,\varphi$ and $\text{says}_{\{1,2\}}\,\varphi$ are *identical*. In many examples, we will have need only for the notation $\text{says}_{l(A)}\,\varphi$.[2] Specific identifiers (e.g., $\text{says}_{\{1\}}\,\varphi$) will be used to accommodate exceptions to laws (Section 3.3). Exceptions are often conveyed by phrases such as "except as specified in Section 120 of HIPAA" [10, 14], and a subset of identifers would correspond to the laws in "Section 120 of HIPAA". Given a variable over principals $x \in X_P$, we will only use the notation $\text{says}_{l(x)}\,\varphi$. This is useful, for example, to grant powers to a class of principals, e.g., patients of a hospital.

We now mention a peculiarity of Definition 1. The BNF rules ensure the alternation of *obligation* and *saying* modalities, e.g., $\mathcal{O}_y\,\text{says}_{l(y)}\,\mathcal{O}_z\varphi \in L$, but $\mathcal{O}_y\mathcal{O}_z\varphi \notin L$. Following von Wright [53], we understand obligations as applying to actions and their consequences. The language L_{φ_y} (obtained from the first BNF rule) is used to describe actions of a principal y – (a) atomic actions, (b) combinations of actions (using connectives), or (c) *saying*, which is (a consequence of) a speech act. An obligation is an opinion, which is created via a speech act, but is not an act by itself. These restrictions are similar in spirit to the logics of power [18, 34, 35, 40].

The statements in L will be used in *the inference component* of access control,

[2]The assumptions about assignment of identifiers are purely (and hopefully) for clarity. We do not consider obligations, permissions, and statements associated with groups of individuals in this work, and shared identifiers may be useful here. We believe that these can be straightforwardly added to the present framework.

i.e., to determine what has been said. In other words, we will be given a set of utterances U and a question ψ, and we need to determine whether ψ is *provable* from U. We focus on provability for the propositional subset of L, i.e., without variables. The propositional subset of L has the modalities $\text{says}_{Id_A}\,\varphi$ and $\mathcal{O}_A(\varphi)$ (for all $A \in O$ and $Id_A \subseteq l(A)$).

A1 All substitution instances of propositional tautologies.

A2 $\mathcal{Q}(\varphi \Rightarrow \psi) \Rightarrow (\mathcal{Q}(\varphi) \Rightarrow \mathcal{Q}(\psi))$ (for all modalities $\mathcal{Q}$)

A3 $\text{says}_{Id_A}\,\varphi \Rightarrow \text{says}_{Id'_A}\,\varphi$ (for all $A \in O_P$ and $Id_A \subseteq Id'_A \subseteq l(A)$)

A4 $\mathcal{O}_A\varphi \Rightarrow \mathcal{P}_A\varphi$ (for all $A \in O_P$)

A5 $\text{says}_{Id_A}(\mathcal{P}_B\,\text{says}_{Id_B}\,\varphi) \Rightarrow (\text{says}_{Id_B}\,\varphi \Rightarrow \text{says}_{Id_A}\,\varphi)$ (for all $\{A, B\} \subseteq O_P$, $Id_A \subseteq l(A)$, and $Id_B \subseteq l(B)$)

A6 $\text{says}_{Id_A}(\mathcal{P}_A\,\text{says}_{Id_A}\,\varphi) \Rightarrow \text{says}_{Id_A}\,\varphi$ (for all $A \in O_P$, and $Id_A \subseteq l(A)$)

R1 From $\vdash \varphi \Rightarrow \psi$ and $\vdash \varphi$, infer $\vdash \psi$

R2 From $\vdash \varphi$, infer $\vdash \mathcal{Q}(\varphi)$ (for all modalities $\mathcal{Q}$)

Figure 2: Axiomatization of the propositional fragment of L. The set of modalities $\mathcal{Q}$ consists of $\text{says}_{Id_A}\,\varphi$ and $\mathcal{O}_A(\varphi)$ (for all $A \in O$ and $Id_A \subseteq l(A)$).

We adopt the axiomatization in Figure 2. **A1** and **R1** give us propositional reasoning. **A2** and **R2** are common to both saying and obligation. **A3** and **A4** are specific to saying and obligation respectively. Finally, **A5** and **A6** describe the interaction between the two modalities.

The notion of provability is of crucial interest. We say that φ is provable (denoted $\vdash \varphi$), if φ is an instance of the axioms **A1**-**A6** or follows from the axioms using the rules **R1** and **R2**. Given a finite set of formulas Δ, we say that φ is provable from Δ, denoted $\Delta \vdash \varphi$, if $\vdash (\bigwedge \Delta) \Rightarrow \varphi$.

We mention some provable statements that we will use in the example from HIPAA (Section 3.2.2):

Proposition 1. *The following are provable:*

1. $\vdash \text{says}_{l(A)}(\mathcal{O}_B\,\text{says}_{l(B)}\,\varphi) \Rightarrow (\text{says}_{l(B)}\,\varphi \Rightarrow \text{says}_{l(A)}\,\varphi)$
2. $\vdash \text{says}_{l(A)}(\mathcal{O}_A\,\text{says}_{l(A)}\,\varphi) \Rightarrow \text{says}_{l(A)}\,\varphi$
3. $\vdash \text{says}_{l(A)}(\mathcal{P}_B\,\text{says}_{l(B)}\,\bot) \Rightarrow (\text{says}_{l(B)}\,\varphi \Rightarrow \text{says}_{(A)}\,\varphi)$

The proofs are easy and we leave the details to the reader. Items 1 and 2 show that versions of axioms **A5** and **A6** hold for obligation. Item 3 gives us *speaking for*, i.e., B speaks for A, as we discussed in Section 2.1.

The rest of the section is organized as follows. We begin in Section 3.2.1 by discussing the various axioms in the context of related work. Section 3.2.2 considers the example from HIPAA, and the various subtleties involved in reasoning about rights. We then present a complete Kripke semantics for the axioms (Section 3.2.3), and use it to show that provability is decidable (Section 3.2.4).

3.2.1. Discussion of Axioms

We now discuss the axioms. The axioms **A1** and **A2**, together with the rules **R1** and **R2**, gives us the modal logic **K**. The **K** axiomatization was used by Abadi et al. [3] as a basis for all (classical) access control logics. From **A3**, it follows that if A says φ via the laws (Id_A), then φ also holds w.r.t. a larger set of laws issued by A (Id'_A). In Section 3.3, when evaluating regulations, we will obtain a statement of the form $\text{says}_{\{id_A\}}\,\varphi$ from each law of A. And, **A3** can be informally understood as a *monotonicity* condition, i.e., if φ is a consequence of what is said via an individual law, then φ is said via all sets of laws that include it. At a first glance, **A3** is at odds with the fact that regulations are *non-monotonic*. However, as shown in Figure 1, we are interested in reasoning about utterances, which are obtained from the laws *after* all exceptions have been resolved. If the law changes, then the utterances will have to be recomputed.

The **K** axiomatization, together with **A4**, gives us the the modal logic **KD**. This axiomatization is common to many systems, giving it the name Standard Deontic Logic (SDL) (c.f. [33]). We note that SDL is a very simplistic system of obligation, and several objections can be raised. The most serious objection is that SDL does not cope with contrary-to-duty (CTD) obligations (see, e.g., [22, 42, 48]). A CTD obligation is one that arises when another has been violated. This is useful, for example, in business contracts to describe mitigating actions [20, 21, 27, 38], e.g., "paying a fine", upon failure to deliver goods. We do not address CTD structures in this work, as they are not as prevalent in privacy regulation as they are in contracts. Governatori and Rotolo [22] propose that CTDs are not a problem with obligations per se, but can be understood as a special kind of exception. We agree entirely with their perspective. However, accommodating these kinds of exceptions involves the introduction of a preference operator, and we leave this to future work.

As we discussed in Section 2.1, **A5** is needed to accommodate notions of representation in access control. *The self-respecting axiom*, **A6**, is read as "If A permits herself to say φ, then A says φ". We discuss the use of **A6** in the example from HIPAA.

3.2.2. Example

We consider the example from HIPAA, introduced in Section 3.1. In this section, our focus is on the *utterances* obtained from the laws of the various principals.

Let H stand for (the regulator who wrote) HIPAA. And, let Alice (A) be a patient whose records (r) are maintained by an insurance company run by Bob (B). Let us assume further that A *has the right* to access her records. The utterance obtained from H's laws would be:

(u1) $\text{says}_{l(H)}\, \mathcal{P}_A\, \text{says}_{l(A)}\, \mathcal{O}_B\, \text{says}_{l(B)}\, \mathcal{P}_A \text{access}(A, r)$

The direct reading of (u1) in English is unwieldy, i.e., we get "H says that A is permitted to say that B is required to say that A is permitted to access her records". A better reading is obtained by eliding all occurrences of *says* that appear immediately above an obligation or permission, except for the outermost one. Applying this ellipsis to (u1), we get: $\text{says}_{l(H)}\, \mathcal{P}_A ... \mathcal{O}_B ... \mathcal{P}_A \text{access}(A, r)$, which is read as: "HIPAA says that Alice is permitted to require Bob to permit her to access her records". We will use such readings henceforth.

The word *right* does not have a unique translation into logic. Hohfeld [31] pointed out that the word *right* is used in different senses, and depending on the context, it can entail a *permission*, *claim*, or *power*.[3] The formulation in (u1) corresponds to the *power* interpretation. As we mentioned in Section 1, our descriptions of powers follows the suggestion of Lindahl [40, Part II] (see also [34]), in terms of nesting obligations and permissions with an action modality.

How does Alice exercise this right? In our approach, rights are exercised by the introduction of a law. The specific mechanism for introducing such laws is application dependent. For example, if Alice sends an email to Bob requiring him to grant her access, then this may *count as* Alice exercising her right (see [18, 35]). Alice's attempt to exercise her right would result in the following utterance:

(u2) $\text{says}_{l(A)}\, \mathcal{O}_B\, \text{says}_{l(B)}\, \mathcal{P}_A \text{access}(A, r)$

In other words, Alice says that Bob is required to permit her to access her records.

How does Bob comply with this right? In our approach, this happens via Bob's access control policy. Suppose Bob wants to permit a patient to access their records only if HIPAA requires it. Bob's policy is represented as follows:

(u3) $\text{says}_{l(B)}\, \mathcal{P}_H\, \text{says}_{l(H)}\, \mathcal{O}_B\, \text{says}_{l(B)}\, \mathcal{P}_A \text{access}(A, r)$

In other words, Bob permits HIPAA to require him to permit Alice to view her records. Note that Bob has no regard for Alice's requirement to see her records, but only what HIPAA says.

Let Δ consist of the utterances (u1), (u2), and (u3) above. Since Alice is attempting to view her records, the access control system tries to prove that $\text{says}_{l(B)}\, \mathcal{P}_A \text{access}(A, r)$ from Δ. The derivation proceeds as follows:

(d1) $\Delta \vdash \text{says}_{l(H)}\, \mathcal{O}_B\, \text{says}_{l(B)}\, \mathcal{P}_A \text{access}(A, r)$ (from (u1) and (u2) using **A5**).

(d2) $\Delta \vdash \text{says}_{l(B)}\, \mathcal{O}_B\, \text{says}_{l(B)}\, \mathcal{P}_A \text{access}(A, r)$ (from (u3) and (d1) using Proposition 1 item 1)

(d3) $\Delta \vdash \text{says}_{l(B)}\, \mathcal{P}_A \text{access}(A, r)$ (from (d2) using Proposition 1 item 2)

[3] Hohfeld [31] describes a *claim* as the correlative of obligation, i.e., when a claim is invaded an obligation is violated. For example, a patient has a claim that hospitals notify her of disclosures of her health information. And, the claim is equivalent to an obligation on the hospital to notify her.

Step (d1) is understood as HIPAA enforcing Alice's right, i.e., HIPAA requires Bob to permit Alice to view her records. In step (d2), Bob acknowledges HIPAA's authority by requiring himself to permit Alice to view her records. Step (d3) shows the utility of **A6**, i.e., by forcing Bob to say what he requires himself to say. Due to (d3), Alice is indeed permitted to view her records! In Section 3.5, we will show how blame can be assigned to Bob if he fails to introduce (d3) or something that implies it.

In summary, to reason about a right, we had to use utterances from the enforcer (HIPAA), the person exercising the right (Alice), and the person complying with it (Bob). The precise manner in which Alice's utterance is obtained is left unspecified. In assessing violations of rights, the issue in question is often whether the right was exercised. For example, Bob may claim that Alice did not request to see her records. We do not believe that this is a problem for logic, but it is a problem in implementing a system that allows principals to exercise their rights. However, we do believe that the logic provides a good intuition for the inferences involved, given the appropriate utterances.

The reasoning involved in this example is outside the scope of prior access control logics [1–3, 11, 16, 17, 39], because obligation is not accommodated. We believe that the reasoning can be accommodated by the *counts as* frameworks for power [18, 35], but as discussed in Section 2.2, some reformulation is needed.

3.2.3. Semantics, Soundness, and Completeness

In this section, we provide a Kripke semantics for which the axiomatization is sound and complete. Semantic completeness is used mainly as a tool, for example, to show that a statement is not provable. Identifying a compelling semantics for *says* is an important open problem in access control logics (see [1]), and we do not address it in this work.[4] We begin by defining models (Kripke structures):

Definition 2 (Models)**.** *Given countable sets of object names O, principal names $O_P \subseteq O$, $\Phi_1, ..., \Phi_n$ (where Φ_j is a set of predicate names of arity j), identifiers for rules ID, and $l : O_P \to 2^{ID}$, a model $M(O, O_P, \Phi_1, ..., \Phi_n, ID, l)$, abbreviated as M, is the tuple $(S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$ where:*

- *S is a set of states*
- *$I_{\Phi_j} : \Phi_j \times S \to 2^{O^j}$ is the interpretation of predicates of arity j. Given $p \in \Phi_j$, we will say that $p(o_1, ..., o_j)$ is true at state s iff $(o_1, ..., o_j) \in I_{\Phi_j}(p, s)$.*
- *$\delta_{\mathcal{L}} : S \times 2^{ID} \to 2^S$. $\delta_{\mathcal{L}}(s, \mathrm{Id})$ corresponds to a description of s according to the laws labeled with identifiers in Id (taken conjunctively).*

[4] We speculate that a good semantics for *says* has to come from an application other than access control and conformance. In these applications, *saying* arises via policies, which are expressed using formulas. There does not seem to be a corresponding computational interpretation.

- $\delta_{\mathcal{O}} : \mathcal{S} \times O_P \rightarrow 2^S$. *$\delta_{\mathcal{O}}(s, A)$ corresponds to an idealization of s, for which the principal A is held responsible.*

For the axioms **A3**-**A6** *we need the following constraints* **C3**-**C6** *(resply). For all $s \in \mathcal{S}$:*

C3 *$\delta_{\mathcal{L}}(s, \mathrm{Id}_A) \supseteq \delta_{\mathcal{L}}(s, \mathrm{Id'}_A)$ for all $A \in O_P$ and $Id_A \subseteq Id'_A \subseteq l(A)$*

C4 *$\delta_{\mathcal{O}}(s, A) \neq \emptyset$ for all $A \in O_P$*

C5 *For all $\{A, B\} \subseteq O_P$, $Id_A \subseteq l(A)$, $Id_B \subseteq l(B)$, and $s' \in \delta_{\mathcal{L}}(s, Id_A)$:*

1. *$s' \in \delta_{\mathcal{L}}(s, Id_B)$, or*
2. *There exists $s_1 \in \delta_{\mathcal{L}}(s, Id_A)$ such that for all $s_2 \in \delta_{\mathcal{O}}(s_1, B)$, $s' \in \delta_{\mathcal{L}}(s_2, Id_B)$*

C6 *For all $A \in O_P$, $Id_A \subseteq l(A)$, and $s' \in \delta_{\mathcal{L}}(s, Id_A)$:*
There exists $s_1 \in \delta_{\mathcal{L}}(s, Id_A)$ such that for all $s_2 \in \delta_{\mathcal{O}}(s_1, A)$, $s' \in \delta_{\mathcal{L}}(s_2, Id_A)$

C5 and **C6** can be understood in the context of soundness (Lemma 1). Given the object names O, $O_P \subseteq O$, predicate names $(\Phi_1, ..., \Phi_n)$, identifiers ID, and the function l, the space of models is denoted by $\mathcal{M}(O, O_P\ \Phi_1, ..., \Phi_n, ID, l)$, abbreviated as $\mathcal{M}$. We can now define satisfaction and validity, and we restrict attention to the propositional fragment of L:

Definition 3 (Semantics)**.** *Given a model $M = (S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$, $s \in S$ and a propositional $\varphi \in L$, the relation $(M, s) \models \varphi$ is defined inductively as follows:*

- *$(M, s) \models p(o_1, ..., o_j)$ iff $(o_1, ..., o_j) \in I_{\Phi_j}(p, s)$.*
- *The semantics of conjunction and negation is defined in the usual way.*
- *$(M, s) \models \mathrm{says}_{Id_A}\ \varphi$ iff $(M, s') \models \varphi$, for all $s' \in \delta_{\mathcal{L}}(s, Id_A)$.*
- *$(M, s \models \mathcal{O}_A \varphi$ iff $(M, s') \models \varphi$, for all $s' \in \delta_{\mathcal{O}}(s', A)$.*

We can now define validity:

- *φ is valid in a model M ($M \models \varphi$) iff for all $s \in S$, $(M, s) \models \varphi$*
- *φ is valid ($\models \varphi$) iff for all $M \in \mathcal{M}$, $M \models \varphi$*

Theorem 1 (Soundness and Completeness)**.** *Given a propositional $\varphi \in L$, $\vdash \varphi$ iff $\models \varphi$*

Lemma 1 (Soundness)**.** *Given a propositional $\varphi \in L$, if $\vdash \varphi$, then $\models \varphi$*

Proof. We need to show that the axioms are valid, and that the rules preserve validity. It is well-known that the axioms **A1** and **A2** are valid, and that **R1** and **R2** preserve validity in all Kripke structures. The validity of **A3** and **A4** can easily be shown using **C3** and **C4**. We discuss the case for **A5**.

Suppose **A5** is not valid. There exists M, s, φ, A, B, Id_A and Id_B such that:

- $(M, s) \models \text{says}_{Id_A}(\mathcal{P}_B \, \text{says}_{Id_B} \, \varphi)$
- $(M, s) \models \text{says}_{Id_B} \, \varphi$, and
- $(M, s) \not\models \text{says}_{Id_A} \, \varphi$

Since $(M, s) \not\models \text{says}_{Id_A} \, \varphi$, there exists $s' \in \delta_{\mathcal{L}}(s, Id_A)$ such that $(M, s') \not\models \varphi$. Since **C5** holds, there are two cases to consider:

1. If $s' \in \delta_{\mathcal{L}}(s, Id_B)$, then $(M, s) \not\models \text{says}_{Id_B} \, \varphi$ giving us a contradiction.
2. If there exists $s_1 \in \delta_{\mathcal{L}}(s, Id_A)$ such that for all $s_2 \in \delta_{\mathcal{O}}(s_1, B)$, $s' \in \delta_{\mathcal{L}}(s_2, Id_B)$, then:
   - $(M, s_1) \models \mathcal{O}_B \neg \, \text{says}_{Id_B} \, \varphi$
   - $(M, s) \not\models \text{says}_{Id_A}(\neg \mathcal{O}_B \neg \, \text{says}_{Id_B} \, \varphi)$

   Hence, $(M, s) \not\models \text{says}_{Id_A}(\mathcal{P}_B \, \text{says}_{Id_B} \, \varphi)$ (since $\mathcal{P}_B \varphi = \neg \mathcal{O}_B \neg \varphi$), giving us a contradiction.

Hence, **A5** is valid. The proof for **A6** is similar. □

Lemma 2 (Completeness)**.** *Given a propositional $\varphi \in L$, if $\models \varphi$, then $\vdash \varphi$*

The rest of this section gives the proof. We will use a canonical model argument (c.f. [29]). We show the contrapositive, i.e., if $\nvdash \varphi$, then $\not\models \varphi$. In other words, if $\nvdash \varphi$ then there exist M and s such that $(M, s) \models \neg\varphi$. We begin with some terminology.

We say that φ is *consistent* if $\neg\varphi$ is not provable ($\nvdash \neg\varphi$). A finite set of formulas $\{\varphi_1, ..., \varphi_n\}$ is consistent if $\varphi_1 \wedge ... \wedge \varphi_n$ is consistent. An infinite set of formulas is consistent if every finite subset is consistent. A set of formulas Δ is *maximal consistent* if for all $\varphi \in L - \Delta$, $\Delta \cup \{\varphi\}$ is inconsistent. The following are properties of maximal consistent sets:

Proposition 2. *Given a maximal consistent set Δ:*

1. *For all $\varphi \in L$, exactly one of $\varphi \in \Delta$ or $\neg\varphi \in \Delta$*
2. *If $\vdash \varphi \Rightarrow \psi$ and $\varphi \in \Delta$, then $\psi \in \Delta$*
3. *If $\vdash \varphi$, then $\varphi \in \Delta$ and $\mathcal{Q}\varphi \in \Delta$ (for all modalities $\mathcal{Q}$)*

The proof is straightforward. We now define *the canonical model*, in which every consistent formula is true at some state:

Definition 4 (Canonical Model)**.** *The canonical model $M = (S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$ is such that:*

- S *is the set of all maximal consistent sets*
- $(o_1, ..., o_j) \in I_{\Phi_j}(p, \Delta)$ *iff* $p(o_1, ..., o_j) \in \Delta$
- $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_A)$ *iff for all* φ*, if* $\text{says}_{Id_A}\, \varphi \in \Delta$*, then* $\varphi \in \Delta'$
- $\Delta' \in \delta_{\mathcal{O}}(\Delta, A)$ *iff for all* φ*, if* $\mathcal{O}_A\varphi \in \Delta$*, then* $\varphi \in \Delta'$

We now show that the canonical model satisfies the frame constraints:

Proposition 3. *The canonical model satisfies the frame constraints* **C3**-**C6**

Proof. The proof that **C3** and **C4** hold are left to the reader. We discuss the case for **C5**. Given the canonical model $M = (S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$, $\Delta \in S$, and suppose for the purpose of contradiction that there exists $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_A)$ such that:

- $\Delta' \notin \delta_{\mathcal{L}}(\Delta, Id_B)$. By construction, there exists $\text{says}_{Id_B}\, \psi \in \Delta$ such that $\neg\psi \in \Delta'$.
- For all $\Delta_1 \in \delta_{\mathcal{L}}(\Delta, Id_A)$, there exists $\Delta_2 \in \delta_{\mathcal{O}}(\Delta_1, B)$, $\Delta' \notin \delta_{\mathcal{L}}(\Delta_2, Id_B)$. By Proposition 4 (below), there exists $\text{says}_{Id_A}\, \mathcal{P}_B\, \text{says}_{Id_B}\, \varphi \in \Delta$ such that $\neg\varphi \in \Delta'$.

Using Proposition 2, $\text{says}_{Id_B}(\varphi \vee \psi) \in \Delta$ and $\text{says}_{Id_A}\, \mathcal{P}_B\, \text{says}_{Id_B}(\varphi \vee \psi) \in \Delta$. So, $\text{says}_{Id_A}(\varphi \vee \psi) \in \Delta$, and hence $\varphi \vee \psi \in \Delta'$. That is $\varphi \in \Delta'$ or $\psi \in \Delta'$, which contradicts the fact that $\neg\varphi \in \Delta'$ and $\neg\psi \in \Delta'$. The proof of **C6** is similar. □

Proposition 4. *Given the canonical model* $M = (S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$*, for all* $\Delta \in S$*,* $\{A, B\} \subseteq O_P$*,* $Id_A \subseteq l(A)$*,* $Id_B \subseteq l(B)$*, and* $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_A)$*:*

- *If for all* $\Delta_1 \in \delta_{\mathcal{L}}(\Delta, Id_A)$*, there exists* $\Delta_2 \in \delta_{\mathcal{O}}(\Delta_1, B)$*,* $\Delta' \notin \delta_{\mathcal{L}}(\Delta_2, Id_B)$*, then there exists* $\text{says}_{Id_A}\, \mathcal{P}_B\, \text{says}_{Id_B}\, \varphi \in \Delta$ *and* $\neg\varphi \in \Delta'$

Proof. Fix Δ, A, B, Id_A, Id_B and $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_A)$. We proceed by contradiction. Suppose for all $\varphi \in L$, if $\text{says}_{Id_A}\, \mathcal{P}_B\, \text{says}_{Id_B}\, \varphi \in \Delta$, then $\varphi \in \Delta'$. Let F be the smallest set such that:

- If $\text{says}_{Id_A}\, \varphi \in \Delta$, then $\varphi \in F$, and
- If $\neg\psi \in \Delta'$, then $\mathcal{O}_B \neg\, \text{says}_{Id_B}\, \psi \in F$.

We claim that F is consistent.[5] Suppose not:

[5]Note that if there exists φ such that $\Delta \vdash \text{says}_{Id_A}\, \varphi$ and $\Delta \vdash \text{says}_{Id_A}\, \neg\varphi$, then $\delta_{\mathcal{L}}(\Delta, Id_A) = \emptyset$, and **C5** and **C6** are vacuously satisfied. In Proposition 4 (and Proposition 6 in Section 3.2.4), the contradiction applies only to cases where there exists $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_A)$, and hence, no such φ exists.

(1) There exists $\{\varphi_1, ..., \varphi_n, \mathcal{O}_B \neg \text{says}_{Id_B} \psi_1, ..., \mathcal{O}_B \neg \text{says}_{Id_B} \psi_m\} \subseteq F$ such that:

$\vdash \neg(\varphi_1 \wedge ... \wedge \varphi_n \wedge \mathcal{O}_B \neg \text{says}_{Id_B} \psi_1 \wedge ... \wedge \mathcal{O}_B \neg \text{says}_{Id_B} \psi_m)$

(2) $\vdash \varphi_1 \wedge ... \wedge \varphi_n \Rightarrow \mathcal{P}_B \, \text{says}_{Id_B}(\psi_1 \vee ... \vee \psi_m)$ (from (1) using **A1**, **A4**, **R1** and **R2**)

(3) $\vdash \text{says}_{Id_A}(\varphi_1 \wedge ... \wedge \varphi_n \Rightarrow \mathcal{P}_B \, \text{says}_{Id_B}(\psi_1 \vee ... \vee \psi_m)) \in \Delta$ (from (2) using **R2**)

(4) By construction, $\text{says}_{Id_A} \varphi_i \in \Delta$ for all $1 \leq i \leq n$. So, using **A2** and (3), we can derive that $\text{says}_{Id_A} \mathcal{P}_B \, \text{says}_{Id_B}(\psi_1 \vee ... \vee \psi_m) \in \Delta$. As a result, $\psi_1 \vee ... \vee \psi_m \in \Delta'$, and there exists $\psi_i \in \Delta'$ where $1 \leq i \leq m$.

(5) By construction, $\neg\psi_i \in \Delta'$ for all $1 \leq i \leq m$, which together with (4) contradicts the consistency of Δ'.

We can extend F into a maximal consistent set Δ_1 such that $\Delta_1 \in \delta_{\mathcal{L}}(\Delta, Id_A)$. $\mathcal{P}_B \, \text{says}_{Id_B} \varphi \in \Delta_1$ iff $\varphi \in \Delta'$. So, for all $\Delta_2 \in \delta_{\mathcal{O}}(\Delta_1, B)$, if $\text{says}_{Id_B} \varphi \in \Delta_2$, then $\varphi \in \Delta'$. This suffices to conclude that $\Delta' \in \delta_{\mathcal{L}}(\Delta_2, Id_B)$ for all $\Delta_2 \in \delta_{\mathcal{O}}(\Delta_1, B)$, giving us a contradiction. □

The completeness proof is now finished in the usual way (see, for example, [29]). Given the canonical model M and a state Δ, it is easy to show that for all $\varphi \in L$, $(M, \Delta) \models \varphi$ iff $\varphi \in \Delta$. Furthermore, given a consistent φ, we can construct a maximal consistent set Δ such that $\varphi \in \Delta$. As a result, for every consistent φ, there exists a state Δ in the canonical model such that $(M, \Delta) \models \varphi$. Hence, if $\nvdash \varphi$, then $\not\models \varphi$. We observe that compactness follows as a corollary of the existence of the canonical model:

Corollary 1 (Compactness)**.** *An infinite set of formulas is satisfiable iff every finite subset is satisfiable.*

Given an infinite set of formulas Δ, if every finite subset is satisfiable, then by soundness, every finite subset of Δ is consistent. And, by definition, Δ is consistent. We can extend Δ into a maximal consistent set, corresponding to a state in the canonical model.

3.2.4. Decidability

In this section, we adapt the completeness proof to show *the bounded-model property*, i.e., if ϕ is satisfiable, then it is satisfiable in a model of bounded size (exponential in the size of ϕ). We begin by defining the set of subformulas:

Definition 5 (Subformulas)**.** *Given a propositional $\phi \in L$, the set of subformulas $sub(\phi)$ is the smallest set such that:*

1. $\phi \in sub(\phi)$
2. *If $\varphi \in sub(\phi)$, then $\neg\varphi \in sub(\phi)$ ($\neg\neg\varphi$ is identified with φ)*
3. *If $\varphi \wedge \psi \in sub(\phi)$, then $\varphi \in sub(\phi)$ and $\psi \in sub(\phi)$*

4. *If* $\mathcal{O}_A\psi \in sub(\phi)$ *or* $\text{says}_{Id_A}\,\psi \in sub(\phi)$*, then* $\psi \in sub(\phi)$
5. *If* $\text{says}_{Id_A}\,\psi_1 \in sub(\phi)$ *and* $\text{says}_{Id'_A}\,\psi_2 \in sub(\phi)$ *such that* $Id_A \subseteq l(A)$ *and* $Id'_A \subseteq l(A)$*, then* $\text{says}_{Id_A \cup Id'_A}\,\psi_1 \in sub(\phi)$
6. *If* $\text{says}_{Id_A}(\bigvee \Delta_1) \in sub(\phi)$ *and* $\text{says}_{Id_B}(\bigvee \Delta_2) \in sub(\phi)$*, then* $\text{says}_{Id_A}(\bigvee \Delta_2) \in sub(\phi)$ *and* $\text{says}_{Id_A}(\bigvee(\Delta_1 \cup \Delta_2)) \in sub(\phi)$
7. *If* $\text{says}_{Id_A}\,\psi_1 \in sub(\phi)$ *and* $Id_A \subseteq l(A)$*, then* $\mathcal{P}_A\,\text{says}_{Id_A}\,\psi_1 \in sub(\phi)$

The last three clauses in Definition 5 are used to ensure that **C5** and **C6** hold. Note that in Clause 5, we consider disjunction over sets of formulas Δ_1 and Δ_2. Formulas which are not disjunctions are understood as disjunctions over singleton sets, e.g., $\varphi \wedge \psi = \bigvee\{\varphi \wedge \psi\}$. To obtain the analog of Proposition 3, we need to ensure that formulas appearing within the scope of *says* are closed under disjunction. We use sets of formulas to ensure that only finitely many disjunctions are introduced, i.e., a disjunct need not be repeated. Due to Clauses 5 and 6, the number of subformulas is exponential in the size of ϕ. It is possible to eliminate both these clauses, by filtering the model that we construct here. But, this further filtration is not needed for the results proved in this work. Clause 7 is key to obtaining the analog of Proposition 4.

Given $\phi \in L$, we will consider maximal consistent sets w.r.t. $sub(\phi)$. A set $\Delta \subseteq sub(\phi)$ is said to be maximal consistent iff Δ is consistent and for all $\psi \in sub(\phi) - \Delta$, $\Delta \cup \{\psi\}$ is inconsistent. We write $\Delta \vdash \varphi$ to denote $\vdash \bigwedge \Delta \Rightarrow \varphi$. The definition of the canonical model needs a few changes:

Definition 6 (Canonical Model of ϕ)**.** *The canonical model of* ϕ*, denoted* $M_\phi = (S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$*, is such that:*

- S *is the set of all maximal consistent sets w.r.t.* $sub(\phi)$
- $(o_1, ..., o_j) \in I_{\Phi_j}(p, \Delta)$ *iff* $p(o_1, ..., o_j) \in \Delta$
- $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_A)$ *iff for all* $\psi \in sub(\phi)$ *and* $Id'_A \subseteq Id_A$*, if* $\text{says}_{Id'_A}\,\psi \in \Delta$*, then* $\psi \in \Delta'$
- $\Delta' \in \delta_{\mathcal{O}}(\Delta, A)$ *iff for all* $\psi \in sub(\phi)$*, if* $\mathcal{O}_A\psi \in \Delta$*, then* $\psi \in \Delta'$*.*

We will show that the canonical model of ϕ satisfies the frame constraints. We adapt Propositions 3 and 4 to obtain Propositions 5 and 6 resply.

Proposition 5. *The canonical model of* ϕ *satisfies the frame constraints* **C3**-**C6**

Proof. The proof that **C3** and **C4** hold are left to the reader. We discuss the case for **C5**. Given $M_\phi = (S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$, consider some $\Delta \in S$. If $\delta_{\mathcal{L}}(\Delta, Id_A) = \emptyset$, then **C5** is vacuously satisfied. Otherwise, let $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_A)$. There are two cases to consider.

First, we have the boundary case, where there is no subformula $\text{says}_{Id'_A}\,\varphi' \in sub(\phi)$ such that $Id'_A \subseteq Id_A$. By defintion, $\delta_{\mathcal{L}}(\Delta, Id_A) = S$. Consider the set

$F \subseteq sub(\phi)$ such that $\psi \in F$ iff ψ is of the form $\mathcal{O}_B \neg \text{says}_{Id_B} \varphi$ and $\neg\varphi \in \Delta'$. We claim that F is consistent. Since Δ' is consistent, we can construct a model M' with states S', and $s' \in S'$ such that $(M', s') \models \bigwedge \Delta'$. Without loss of generality, we can assume that there exists $s'' \in S'$ such that $\delta_{\mathcal{O}}(s'', A) = \{s''\}$ and $\delta_{\mathcal{L}}(s'', Id_B) = S'$ for all $B \in O_P$ and $Id_B \subseteq l(B)$. Note that states, such as s'', trivially satisfy the frame constraints, and can be added to any model. It is easy to see that $(M', s'') \models \bigwedge F$, and by soundness, F is consistent. We can extend F into a maximal consistent set Δ_1 such that for all $\Delta_2 \in \delta_{\mathcal{O}}(\Delta_1, B)$, we have $\Delta' \in \delta_{\mathcal{L}}(\Delta_2, Id_B)$. Since $\delta_{\mathcal{L}}(\Delta, Id_A) = S$, we have $\Delta_1 \in \delta_{\mathcal{L}}(\Delta, Id_A)$, and **C5** is satisfied.

For the second case we proceed as follows. Let Id^*_A be the largest subset of Id_A such that there is a subformula $\text{says}_{Id^*_A} \varphi' \in sub(\phi)$. The existence of a largest subset is guaranteed by Clause 5 in Definition 5. Fix $Id_B \subseteq l(B)$. If there is no subformula $\text{says}_{Id'_B} \psi' \in sub(\phi)$ with $Id'_B \subseteq Id_B$, then $\delta_{\mathcal{L}}(\Delta, Id_A) \subseteq \delta_{\mathcal{L}}(\Delta, Id_B) = S$, and **C5** is satisfied. Otherwise, let Id^*_B be the largest subset of Id_B such that there is a subformula $\text{says}_{Id^*_B} \psi' \in sub(\phi)$. We proceed by contradiction analogous to the completeness proof:

- $\Delta' \notin \delta_{\mathcal{L}}(\Delta, Id_B)$. By construction, there exists $\psi \in sub(\phi)$ such that $\text{says}_{Id'_B} \psi \in \Delta$ for some $Id'_B \subseteq l(B)$ and $\neg\psi \in \Delta'$. And, using **A3**, $\Delta \vdash \text{says}_{Id^*_B} \psi$.
- For all $\Delta_1 \in \delta_{\mathcal{L}}(\Delta, Id_A)$, there exists $\Delta_2 \in \delta_{\mathcal{O}}(\Delta_1, B)$, $\Delta' \notin \delta_{\mathcal{L}}(\Delta_2, Id_B)$. By Proposition 6 (below), there exists $\varphi \in sub(\phi)$ such that $\text{says}_{Id^*_B} \varphi \in sub(\phi)$, $\Delta \vdash \text{says}_{Id^*_A} \mathcal{P}_B \text{says}_{Id^*_B} \varphi$ and $\neg\varphi \in \Delta'$.

Since $\Delta \vdash \text{says}_{Id^*_B}(\varphi \vee \psi)$ and $\Delta \vdash \text{says}_{Id^*_A} \mathcal{P}_B \text{says}_{Id^*_B}(\varphi \vee \psi)$, we have $\Delta \vdash \text{says}_{Id^*_A}(\varphi \vee \psi)$. Using Clause 6 in Definition 5, there exists $\text{says}_{Id^*_A} \varphi_1 \in sub(\phi)$ such that $\vdash \varphi_1 \Leftrightarrow (\varphi \vee \psi)$. As a result, $\text{says}_{Id^*_A} \varphi_1 \in \Delta$, and hence $\varphi_1 \in \Delta'$. Since $\vdash \varphi_1 \Leftrightarrow (\varphi \vee \psi)$, we have $\varphi \in \Delta'$ or $\psi \in \Delta'$, which contradicts the fact that $\neg\varphi \in \Delta'$ and $\neg\psi \in \Delta'$. The proof of **C6** is similar. □

Proposition 6. *Given $\phi \in L$, $Id_A \subseteq l(A)$ and $Id_B \subseteq l(B)$ such that there are largest subsets $Id^*_A \subseteq Id_A$ and $Id^*_B \subseteq Id_B$ with formulas $\text{says}_{Id^*_A} \varphi' \in sub(\phi)$ and $\text{says}_{Id^*_B} \psi' \in sub(\phi)$, let $M_\phi = (S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$ be the canonical model of ϕ. Then, for all $\Delta \in S$ and $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_A)$:*

- *If for all $\Delta_1 \in \delta_{\mathcal{L}}(\Delta, Id_A)$, there exists $\Delta_2 \in \delta_{\mathcal{O}}(\Delta_1, B)$ such that $\Delta' \notin \delta_{\mathcal{L}}(\Delta_2, Id_B)$, then there exists $\varphi \in sub(\phi)$ such that $\text{says}_{Id^*_B} \varphi \in sub(\phi)$, $\Delta \vdash \text{says}_{Id^*_A} \mathcal{P}_B \text{says}_{Id^*_B} \varphi$ and $\neg\varphi \in \Delta'$*

Proof. Fix Δ and $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_A)$. We proceed by contradiction. Suppose for all $\varphi \in sub(\phi)$ with $\text{says}_{Id^*_B} \varphi \in sub(\phi)$, if $\Delta \vdash \text{says}_{Id_A} \mathcal{P}_B \text{says}_{Id_B} \varphi$, then $\neg\varphi \notin \Delta'$. Let F be the smallest set such that:

- If $\text{says}_{Id'_A} \varphi \in \Delta$ for some $Id'_A \subseteq Id_A$, then $\varphi \in F$

- If $\neg\psi \in \Delta'$ and $\mathcal{O}_B \neg \text{says}_{Id_B^*}\, \psi \in sub(\phi)$, then $\mathcal{O}_B \neg \text{says}_{Id_B^*}\, \psi \in F$.

We claim that F is consistent (see Footnote 5). Suppose not:

(1) There exists $\{\varphi_1, ..., \varphi_n, \mathcal{O}_B \neg \text{says}_{Id_B^*}\, \psi_1, ..., \mathcal{O}_B \neg \text{says}_{Id_B^*}\, \psi_m\} \subseteq F$ such that: $\vdash \neg(\varphi_1 \wedge ... \wedge \varphi_n \wedge \mathcal{O}_B \neg \text{says}_{Id_B^*}\, \psi_1 \wedge ... \wedge \mathcal{O}_B \neg \text{says}_{Id_B^*}\, \psi_m)$

(2) $\vdash \varphi_1 \wedge ... \wedge \varphi_n \Rightarrow \mathcal{P}_B\, \text{says}_{Id_B^*}(\psi_1 \vee ... \vee \psi_m)$ (from (1) using **A1**, **A4**, **R1** and **R2**)

(3) $\vdash \text{says}_{Id_A^*}(\varphi_1 \wedge ... \wedge \varphi_n \Rightarrow \mathcal{P}_B\, \text{says}_{Id_B^*}(\psi_1 \vee ... \vee \psi_m)) \in \Delta$ (from (2) using **R2**)

(4) By construction, $\Delta \vdash \text{says}_{Id_A^*}\, \varphi_i$ for all $1 \leq i \leq n$. So, using **A2** and (3), we can derive that $\Delta \vdash \text{says}_{Id_A^*}\, \mathcal{P}_B\, \text{says}_{Id_B^*}(\psi_1 \vee ... \vee \psi_m)$. Using Clause 6 in Definition 5, there exists $\psi' \in sub(\phi)$ such that $\text{says}_{Id_B^*}\, \psi' \in sub(\phi)$ and $\vdash \psi' \Leftrightarrow (\psi_1 \vee ... \vee \psi_m)$. It follows that $\Delta \vdash \text{says}_{Id_A^*}\, \mathcal{P}_B\, \text{says}_{Id_B^*}\, \psi'$, and by assumption, $\neg\psi' \notin \Delta'$, i.e., $\Delta' \vdash \psi'$. As a result, $\Delta' \vdash \psi_1 \vee ... \vee \psi_m$, and there exists $1 \leq i \leq m$ such that $\psi_i \in \Delta'$ (since $\psi_i \in sub(\phi)$).

(5) By construction, $\neg\psi_i \in \Delta'$ for all $1 \leq i \leq m$, which together with (4) gives us a contradiction.

We can extend F into a maximal consistent set Δ_1 such that $\Delta_1 \in \delta_{\mathcal{L}}(\Delta, Id_A)$. Consider $\Delta_2 \in \delta_{\mathcal{O}}(\Delta_1, B)$. We claim that for all $Id'_B \subseteq Id_B$, if $\text{says}_{Id'_B}\, \varphi \in \Delta_2$, then $\varphi \in \Delta'$. Suppose not. There exists $\text{says}_{Id'_B}\, \varphi \in \Delta_2$ such that $\neg\varphi \in \Delta'$. Using Clauses 5 and 6 in Definition 5, it follows that $\text{says}_{Id_B^*}\, \varphi \in sub(\phi)$, and using **A3**, $\text{says}_{Id_B^*}\, \varphi \in \Delta_2$. Since $\text{says}_{Id_B^*}\, \varphi \in sub(\phi)$, by Clause 7 in Definition 5, $\mathcal{O}_B \neg \text{says}_{Id_B^*}\, \varphi \in sub(\phi)$. By construction, $\mathcal{O}_B \neg \text{says}_{Id_B^*}\, \varphi \in \Delta_1$, and so, $\neg\, \text{says}_{Id_B^*}\, \varphi \in \Delta_2$, contradicting the consistency of Δ_2. This suffices to conclude that $\Delta' \in \delta_{\mathcal{L}}(\Delta_2, Id_B)$ for all $\Delta_2 \in \delta_{\mathcal{O}}(\Delta_1, B)$, giving us a contradiction. □

A standard argument (see, for example, [29]) can be used to show that for all $\varphi \in sub(\phi)$, $(M_\phi, \Delta) \models \varphi$ iff $\varphi \in \Delta$. We can now establish decidability:

Theorem 2 (Decidability)**.** *Given a propositional $\varphi \in L$, checking whether $\vdash \varphi$ is decidable*

Proof. Decidability is established via the bounded model property:

ϕ is satisfiable in M_ϕ iff ϕ is satisfiable

One direction is trivial, i.e., if ϕ is satisfiable in M_ϕ, then ϕ is satisfiable (by definition). For the other direction, we can use a standard filtration argument, to show that M_ϕ can be obtained from the canonical model (Definition 4). □

We set aside the issue of complexity, and more importantly, the identification of tractable fragments to future work. The techniques discussed here would be most relevant to such an investigation. In the following section, we will use provability (and its negation) to describe the process of saying.

3.3. The Saying Component - Policies

In this section, we describe the representation and evaluation of policies or regulations. The result of evaluating regulation is a set of utterances, which forms the basis for access control and conformance. The formalism developed here is an extension of [14], and is a generalized form of logic programming. Logic programs are popular in representing regulatory texts [23, 43, 52], and access control policies [9, 12, 39]. We begin by defining the syntax of regulations:

Definition 7 (Syntax of Regulation)**.** *Given countable sets of identifiers ID, principal names O_P, and a function $l : O_P \to 2^{ID}$, a law is a statement of the form* (id) $\varphi \mapsto \psi$*, where $\varphi \in L_\varphi$, $\psi \in L_\psi$, and there exists $A \in O_P$ such that $id \in l(A)$. The set of all possible laws is denoted by* $\text{Laws}(O_P, l, L)$*, abbreviated* Laws*.*

A body of regulation $\text{Reg} \subseteq \text{Laws}$ *is a finite set such that for all $id \in ID$, there exists at most one pair $(\varphi, \psi) \in L_\varphi \times L_\psi$ such that* (id) $\varphi \mapsto \psi \in \text{Reg}$

(id) $\varphi \mapsto \psi$ is read as: "If φ is true, then A *says* ψ via the law (id)", where $id \in l(A)$. To evaluate laws, we need a way to evaluate preconditions ($\varphi \in L_\varphi$). There are two kinds of atoms in L_φ – predicates and formulas of the form $\text{says}_{Id_y}\, \varphi$. The predicates are evaluated against a *state*, and formulas of the form $\text{says}_{Id_y}\, \varphi$ are evaluated *provability* (as defined in Section 3.2) from a *set of utterances*. We begin by defining states:

Definition 8 (States and Assignments)**.** *Given countable sets O of object names, principal names $O_P \subseteq O$, and predicate names $\Phi_1, \ldots, \Phi_n$, a state $s(O, O_P, \Phi_1, ..., \Phi_n)$, abbreviated s, is the tuple $(I_{\Phi_1}, \ldots, I_{\Phi_n})$ where $I_{\Phi_j} : \Phi_j \to 2^{O^j}$ is the interpretation of predicates of arity j. Given $p \in \Phi_j$, we will say that $p(o_1, ..., o_j)$ is true at state s iff $(o_1, ..., o_j) \in I_{\Phi_j}(p)$. The set of all states is denoted by $\mathcal{S}$.*

Given a set of variables X, and principal variables X_P, an assignment is a function $v : X \to O$, such that for all $x \in X_P$, we have $v(x) \in O_P$. The set of all assignments is denoted by $V(X, X_P, O, O_P)$, abbreviated V.

A state $s \in \mathcal{S}$ is a description of operations, and gives us information, for example, about the accesses to records that *actually* happened. The definition of utterances relies on propositionalizing formulas:

Definition 9 (Propositionalization)**.** *Given $\phi \in L$ and an assignment $v \in V$, the propositionalization of φ w.r.t. v, denoted $v(\phi)$, is defined inductively as follows:*

- $v(p(y_1, ..., y_n)) = p(o_1, ..., o_n)$*, where $o_i = v(y_i)$ if $y_i \in X$ and $o_i = y_i$ otherwise ($y_i \in O$).*
- $v(\varphi \wedge \psi) = v(\varphi) \wedge v(\psi)$*, and* $v(\neg\varphi) = \neg v(\varphi)$
- $v(\mathcal{O}_y\varphi) = \mathcal{O}_A(v(\varphi))$*, where $A = v(y)$ if $y \in X_P$ and $A = y$ otherwise.*

- $v(\text{says}_{Id_y}\,\varphi) = \text{says}_{Id_A}(v(\varphi))$, *where* $Id_A = l(v(y))$ *if* $y \in X_P$ *and* $Id_A = Id_y$ *otherwise.*

We can now define utterances:

Definition 10 (Utterances)**.** *Given a body of regulation* Reg*, and an assignment* $v \in V$*, an utterance is a statement* $v(\text{says}_{\{id\}}\,\psi)$ *such that* $id \in ID$ *and* $(\text{id})\ \varphi \mapsto \psi \in Reg$*. The set of all utterances is denoted by* $\mathcal{U}(\text{Reg}, V)$*.*

To build intuition for the definitions, we first present a simplified version, which can accommodate some (but not all) kinds of regulations. We will then identify the difficult cases, and generalize the definition. Let us assume as given a state s, a body of regulation Reg and an assignment $v \in V$. We wish to determine whether the precondition of a law ($\varphi \in L_\varphi$) is "true" w.r.t. s, Reg and v. Consider the relation $(s, \text{Reg}, v) \models_1 \varphi$ defined inductively as follows:

P1 $(s, \text{Reg}, v) \models_1 p(y_1, ..., y_j)$ iff $(o_1, ..., o_j) \in I_{\Phi_j}(p)$, where $o_i = v(y_i)$ if $y_i \in X$ and $o_i = y_i$ otherwise.

P2 Conjunction and negation are handled in the usual way

P3 $(s, \text{Reg}, v) \models_1 \text{says}_{Id_y}\,\psi'$ iff there exists a set $U \subseteq \mathcal{U}(\text{Reg}, V)$ such that:

- **P3.1** For all $\phi \in U$, there exists $(\text{id})\ \varphi \mapsto \psi \in \text{Reg}$ and $v' \in V$ such that $(s, \text{Reg}, v') \models_1 \varphi$ and $\phi = v'(\text{says}_{\{id\}}\,\psi)$, and
- **P3.2** $U \vdash v(\text{says}_{Id_y}\,\psi')$

P3 is understood as follows. $(s, \text{Reg}, v) \models_1 \text{says}_{Id_y}\,\psi'$ iff there is a set of utterances U such that all formulas in U come from laws with true preconditions (**P3.1**), and $v(\text{says}_{Id_y}\,\psi')$ is provable from U (**P3.2**). We remind the reader that provability is defined in the propositional subset of the language L (Section 3.2). We now give an example to illustrate this definition:

Proposition 7. *Given a principal* $A \in O_P$ *with* $l(A) = \{id1, id2\}$*, and a body regulation* Reg *consisting of only the following statements:*

$(\text{id1})\ p(x) \mapsto \neg q(x)$

$(\text{id2})\ \neg\,\text{says}_{\{id1\}}\,\neg q(y) \mapsto q(y)$

Then for all $s \in \mathcal{S}$ *and* $v \in V$*, we have:*

1. $(s, \text{Reg}, v) \models_1 (\neg p(x)) \Leftrightarrow \text{says}_{\{id2\}}\,q(x)$
2. $(s, \text{Reg}, v) \not\models_1 \text{says}_{l(A)}\,\bot$

Proof. The laws correspond to a standard pattern in default reasoning. The law (id1) is understood as an *exception*, and read as "If $p(x)$ holds, then A *says* $\neg q(x)$ via law (id1)". The law (id2) is the *default*, and read as "If A *does not* say $\neg q(x)$ via law (id1), then A *says* $q(x)$ via law (id2)".

The proofs of both items rely on a property of utterances that satisfy **P3.1**. Given a state s, let U_s be the set of utterances such that for all $v \in V$:

- If $(s, \mathrm{Reg}, v) \models p(x)$, then $v(\mathrm{says}_{\{id1\}}\, \neg q(x)) \in U_s$
- Otherwise, $v(\mathrm{says}_{\{id2\}}\, \neg q(x)) \in U_s$

It is easy to show that U_1 satisfies **P3.1** iff $U_1 \subseteq U_s$.
Item 1: Suppose $(s, \mathrm{Reg}, v) \models_1 \neg p(x)$ for some $v \in V$. Then, $v(\mathrm{says}_{\{id2\}}\, q(x)) \in U_s$. Hence, $U_s \vdash v(\mathrm{says}_{\{id2\}}\, q(x))$ (satisfying **P3.2**), and hence $(s, \mathrm{Reg}, v) \models_1 v(\mathrm{says}_{\{id2\}}\, q(x))$. The converse is similarly verified.
Item 2: Suppose $(s, \mathrm{Reg}, v) \models_1 \mathrm{says}_{l(A)}\, \bot$. It follows that there exists U_1 satisfying **P3.1** such that $U_1 \vdash \mathrm{says}_{l(A)}\, \bot$. However, it is easily seen that $U_s \nvdash \mathrm{says}_{l(A)}\, \bot$, and since $U_1 \subseteq U_s$, by propositional reasoning, $U_1 \nvdash \mathrm{says}_{l(A)}\, \bot$, giving us a contradiction. □

We note that there is nothing intrinsic about the formalism that prevents conflicts in a principal's laws. It is easy to construct a regulation Reg such that for all $s \in \mathcal{S}$ and $v \in V$, we have $(s, \mathrm{Reg}, v) \models_1 \mathrm{says}_{l(A)}\, \bot$. One has to explicitly prevent conflicts via the use of default rules, e.g., (id2) in the example above. It is also possible to modify **P3** so that utterances with conflicts are not used, which would be in line with the approach of defeasible logic [20, 26, 47]. But, we do not explore this option in this work.

Next, we illustrate the sense in which regulations are non-monotonic:

Proposition 8. *There exist regulations* Reg *and* Reg' *such that* $\mathrm{Reg} \subset \mathrm{Reg}'$*, and a state* $s \in \mathcal{S}$*, an assignment* $v \in V$ *and a formula* $\varphi \in L_\varphi$ *such that:*
$(s, \mathrm{Reg}, v) \models_1 \varphi$ *and* $(s, \mathrm{Reg}', v) \not\models_1 \varphi$

Proof. We will give an example similar to the default rules discussed above. The key idea is to make use of a formula $\mathrm{says}_{l(A)}\, \psi$, where $l(A)$ has some identifiers without corresponding laws in Reg. Let $l(A) = \{id1, id2, id3\}$, and Reg consists of only the following two laws:

(id1) $\neg\, \mathrm{says}_{l(A)}\, \neg q(x) \mapsto q(x)$

(id2) $p(x) \mapsto \neg q(x)$

We construct Reg' by adding a law, which conveys an *exception*, to Reg:

(id3) $r(x) \mapsto \neg q(x)$

Given $s \in \mathcal{S}$ and $v \in V$ such that $(s, \mathrm{Reg}, v) \models_1 r(x) \wedge \neg p(x)$ (the exception in (id3) applies, but the exception in (id2) does not), it follows that $(s, \mathrm{Reg}, v) \models_1 \mathrm{says}_{l(A)}\, q(x)$ and $(s, \mathrm{Reg}', v) \not\models_1 \mathrm{says}_{l(A)}\, q(x)$ □

As we mentioned, $\models_1$ is not well-defined for all kinds of regulations:

Proposition 9. *There is no relation* $\models_1$ *that satisfies the properties* **P1**-**P3**

Proof. Given $A \in O_P$ and $l(A) = \{id\}$, consider a regulation Reg that consists of only the following statement:

$$\text{(id)}\ \neg\, \mathrm{says}_{\{id\}}\, p(x) \mapsto p(x)$$

In other words, "If *A does not say* $p(x)$ via law (id), then *A says* $p(x)$ via law (id)". The self-referential nature of this sentence, together with negation, is the source of the problem. We begin by observing that for all $v \in V$ and $U \subseteq \mathcal{U}$, $U \vdash v(\text{says}_{\{id\}}\, p(x))$ iff $v(\text{says}_{\{id\}}\, p(x)) \in U$.

Suppose $(s, \text{Reg}, v) \models_1 \text{says}_{\{id\}}\, p(x)$. By **P3.2**, there exists a set U such that $U \vdash v(\text{says}_{\{id\}}\, p(x))$, and so, $v(\text{says}_{\{id\}}\, p(x)) \in U$. As a result, by **P3.1**, $(s, \text{Reg}, v) \models_1 \neg\, \text{says}_{\{id\}}\, p(x)$, giving us a contradiction.

Suppose $(s, \text{Reg}, v) \not\models_1 \text{says}_{\{id\}}\, p(x)$. Then, $(s, \text{Reg}, v) \models \neg\, \text{says}_{\{id\}}\, p(x)$. The set $U = \{v(\text{says}_{\{id\}}\, p(x))\}$ satisfies **P3.1** and **P3.2**. So, $(s, \text{Reg}, v) \models_1 \text{says}_{\{id\}}\, p(x)$, giving us a contradiction. □

To handle such circular statements, we use a technique from Kripke's theory of truth [37], which also forms the basis for the Kripke-Kleene-Fitting semantics of logic programs [15]. There are two pieces of machinery needed. First, we move to a three-valued logic, where the third (middle) value stands for ungrounded. The values are denoted by $\mathcal{B}^3 = \{\top, ?, \bot\}$. Second, we modify **P3** to use a pair of sets of utterances (U, U') such that $U \subseteq U'$. Informally, U will be the set of utterances obtained from laws with true preconditions, while U' will be set of utterances from laws with true or ungrounded preconditions (by modifying **P3.1**). The truth of $\text{says}_{Id_y}\, \varphi$ will determined using U, and falsity is determined using U' (by modifying **P3.2**). We note that it is not possible to define the three-valued interpretation in a manner isomorphic to **P1**-**P3**. This is because **P3** implicitly assumes the existence of a unique fixed point, and this assumption no longer holds in the three-valued setting. We move the choice of (U, U') and **P3.1** to a separate definition (Definition 12), and consider **P1**, **P2**, and a modified version of **P3.2** together for a fixed (U, U') (Definition 11).

We begin by defining a function **tv** which assigns truth values to preconditions:

Definition 11 (Evaluating Preconditions)**.** *Given a body of regulation* Reg *and a pair utterance sets* (U, U') *such that* $U \subseteq U' \subseteq \mathcal{U}(\text{Reg}, V)$*, the function* $\mathbf{tv}_{(U,U')} : L_\varphi \times \mathcal{S} \times V \to \mathcal{B}^3$ *is defined as follows:*

Predicates are evaluated to true or false. Conjunction and negation are handled using the Kleene semantics.

$$\mathbf{tv}_{(U,U')}(\text{says}_{Id_y}\, \psi, s, v) = \begin{cases} \top \text{ if } U \vdash v(\text{says}_{Id_y}\, \psi) \\ \bot \text{ if } U' \nvdash v(\text{says}_{Id_y}\, \psi) \\ ? \text{ otherwise} \end{cases}$$

Note that if $U = U'$, then $\mathbf{tv}_{(U,U')}(\text{says}_{Id_y}\, \psi, s, v) \in \{\top, \bot\}$, and is identical to **P3.2**. The third value arises only if $U \subsetneq U'$. We now define consistency for the pair of utterances (U, U'), used in Definition 11. We need to ensure that U (resply. U') corresponds to laws with true (resply. true or ungrounded) preconditions:

Definition 12 (Consistent Utterances)**.** *Given a regulation* Reg *and a state* $s \in \mathcal{S}$*, the utterance pair* (U, U') *is consistent iff for all* $\phi \in \mathcal{U}(\text{Reg}, V)$ *:*

- *If* $\phi \in U$*, then there exists* (id) $\varphi \mapsto \psi \in$ Reg *and* $v \in V$ *such that* $v(\text{says}_{\{id\}}\, \psi) = \phi$ *and* $\mathbf{tv}_{(U,U')}(\varphi, s, v) = \top$.
- *If* $\phi \notin U'$*, then for all* (id) $\varphi \mapsto \psi \in$ Reg *and* $v \in V$ *such that* $v(\text{says}_{\{id\}}\, \psi) = \phi$*, we have* $\mathbf{tv}_{(U,U')}(\varphi, s, v) = \bot$.

$C^s_{\mathcal{U}(\text{Reg},V)} = \{(U,U')|\ (U,U') \in 2^{\mathcal{U}(\text{Reg},V)} \times 2^{\mathcal{U}(\text{Reg},V)}$ *and* (U,U') *is consistent*$\}$ *is the set of all consistent utterance pairs w.r.t.* Reg*,* V *and* s.

It is immediate from the definition that for all $(U,U') \in C^S_{\mathcal{U}(\text{Reg},V)}$, we have $U \subseteq U'$. Consider the self-referential example (from Proposition 9) – (id) $\neg \text{says}_{\{id\}}\, p(x) \mapsto p(x)$. For this example, any pair (U,U') such that $v(\text{says}_{\{id\}}\, p(x)) \in U$ or $v(\text{says}_{\{id\}}\, p(x)) \notin U'$ is *not* consistent, as that would mean assigning a value from $\{\top, \bot\}$ to the $\text{says}_{\{id\}}\, p(x)$. Consistency ensures that $\mathbf{tv}_{(U,U')}(\text{says}_{\{id\}}\, p(x), s, v) = ?$ for all $s \in \mathcal{S}$, $v \in V$ and consistent pairs (U,U'). A partial order is defined over the space of consistent utterance pairs:

Definition 13 (Partial Order)**.** *Given the utterance pairs* (U_1,U_1') *and* (U_2,U_2')*, we say that* $(U_1,U_1') \leq (U_2,U_2')$ *iff* $U_1 \subseteq U_2$ *and* $U_1' \supseteq U_2'$.

The pair $(C^s_{\mathcal{U}(\text{Reg},V)}, \leq)$ *is a partially ordered set (poset).*

We are now ready to define the function whose fixed points we will be interested in.

Definition 14 (Inflationary function)**.** *Given a poset* $(C^s_{\mathcal{U}(\text{Reg},V)}, \leq)$*, the function* $\mathcal{I}^s_{\mathcal{U}(\text{Reg},V)} : C^s_{\mathcal{U}(\text{Reg},V)} \to C^s_{\mathcal{U}(\text{Reg},V)}$ *is defined as follows.* $\mathcal{I}^s_{\mathcal{U}(\text{Reg},V)}(U_1,U_1')$ *is the pair* $(U_2,U_2') \in C^s_{\mathcal{U}(\text{Reg},V)}$ *such that for all* $\phi \in \mathcal{U}(\text{Reg},V)$*:*

- $\phi \in U_2$ *iff there exists* (id) $\varphi \mapsto \psi \in$ *Reg and* $v \in V$ *such that* $v(\text{says}_{\{id\}}\, \psi) = \phi$ *and* $\mathbf{tv}_{(U_1,U_1')}(\varphi, s, v) = \top$.
- $\phi \notin U_2'$ *iff for all* (id) $\varphi \mapsto \psi \in$ *Reg and* $v \in V$ *such that* $v(\text{says}_{\{id\}}\, \psi) = \phi$*, we have* $\mathbf{tv}_{(U_1,U_1')}(\varphi, s, v) = \bot$.

The existence of fixed points relies on some properties of $\mathcal{I}^S_{\mathcal{U}(\text{Reg},V)}$, i.e., being inflationary and monotonic:

Proposition 10. *Given a poset* $(C^s_{\mathcal{U}(\text{Reg},V)}, \leq)$*, abbreviated* $(C_{\mathcal{U}}, \leq)$*, the function* $\mathcal{I}^s_{\mathcal{U}(\text{Reg},V)}$*, abbreviated* $\mathcal{I}_{\mathcal{U}}$*, is:*

1. *Inflationary - For all* $(U_1,U_1') \in C_{\mathcal{U}}$*,* $(U_1,U_1') \leq \mathcal{I}_{\mathcal{U}}(U_1,U_1')$
2. *Well-defined - For all* $(U_1,U_1') \in C_{\mathcal{U}}$*,* $\mathcal{I}_{\mathcal{U}}(U_1,U_1') \in C_{\mathcal{U}}$
3. *Monotonic - For all* $\{(U_1,U_1'),(U_2,U_2')\} \in C^S_{\mathcal{U}}$*, if* $(U_1,U_1') \leq (U_2,U_2')$*, then* $\mathcal{I}_{\mathcal{U}}(U_1,U_1') \leq \mathcal{I}_{\mathcal{U}}(U_2,U_2')$

Proof. **Item 1:** Let $(U_2,U_2') = \mathcal{I}_{\mathcal{U}}(U_1,U_1')$. We are given that (U_1,U_1') is consistent. Hence, for all $\phi \in \mathcal{U}$:

- If $\phi \in U_1$, then by Definition 12, there exists (id) $\varphi \mapsto \psi \in \text{Reg}$ and $v \in V$ such that $v(\text{says}_{\{id\}}\, \psi) = \phi$ and $\mathbf{tv}_{(U,U')}(\varphi, s, v) = \top$. Therefore, by Definition 14, $\phi \in U_2$. We can conclude that $U_1 \subseteq U_2$.

- If $\phi \notin U_2$, then by Definition 12, for all (id) $\varphi \mapsto \psi \in \mathit{Reg}$ and $v \in V$ such that $v(\text{says}_{\{id\}}\, \psi) = \phi$, we have $\mathbf{tv}_{(U,U')}(\varphi, s, v) = \bot$. Therefore, by Definition 14, $\phi \notin U_2$. We can conclude that $U_1 \supseteq U_2$.

Hence, by Definition 13, $(U_1, U_1') \leq (U_2, U_2')$
Interlude: For the second and third items, we need the following observations. Given $U_1 \subseteq U_1' \subseteq \mathcal{U}$ and $U_2 \subseteq U_2' \subseteq \mathcal{U}$, if $(U_1, U_1') \leq (U_2, U_2')$, then for all $\varphi \in L_\varphi$ and $v \in V(X, O)$:

(D1) If $\mathbf{tv}_{(U_1,U_1')}(\varphi, s, v) = \top$, then $\mathbf{tv}_{(U_2,U_2')}(\varphi, s, v) = \top$

(D2) If $\mathbf{tv}_{(U_1,U_1')}(\varphi, s, v) = \bot$, then $\mathbf{tv}_{(U_2,U_2')}(\varphi, s, v) = \bot$

These are established easily by induction over the structure of φ. Note that the claims are for *all* pairs of utterances, and not just the consistent ones.
Item 2: Let $(U_2, U_2') = \mathcal{I}_{\mathcal{U}}(U_1, U_1')$. From Item 1, it follows that $(U_1, U_1') \leq (U_2, U_2')$. Suppose, for the purpose of contradiction, that (U_2, U_2') is not consistent. Then, by Definition 12, there exists $\phi \in \mathcal{U}$ such that:

- $\phi \in U_2$ and for all (id) $\varphi \mapsto \psi \in \mathit{Reg}$ and $v \in V$ such that $v(\text{says}_{\{id\}}\, \psi) = \phi$, we have $\mathbf{tv}_{(U_2,U_2')}(\varphi, s, v) \neq \top$, and by (D1), $\mathbf{tv}_{(U_1,U_1')}(\varphi, s, v) \neq \top$. Therefore, by Definition 14, $\phi \notin U_2$, giving us a contradiction.

- The second case (where $\phi \notin U_2'$) is contradicted similarly using (D2).

The proof of Item 3 is along similar lines. □

The existence of fixed points is established using Zorn's lemma (cf. [50]), which applies to *chain-complete posets*. Given a poset $(C^s_{\mathcal{U}(\text{Reg},V)}, \leq)$, a set $\mathcal{C}' \subseteq C^s_{\mathcal{U}(\text{Reg},V)}$ is called a chain (totally ordered set) if for all $(U_1, U_1'), (U_2, U_2') \in \mathcal{C}'$, we have $(U_1, U_1') \leq (U_2, U_2')$ or $(U_2, U_2') \leq (U_1, U_1')$. A poset is *chain-complete* if every chain has a supremum. We now show that $(C^s_{\mathcal{U}(\text{Reg},V)}, \leq)$ is a chain-complete poset:

Proposition 11. *$(C^s_{\mathcal{U}(\text{Reg},V)}, \leq)$ is a chain-complete poset.*

Proof. Given a chain $\mathcal{C}' \subseteq C^s_{\mathcal{U}(\text{Reg},V)}$, consider the pair (U_s, U_s') defined as follows:

$$U_s = \bigcup_{(U,U') \in \mathcal{C}'} U \qquad U_s' = \bigcap_{(U,U') \in \mathcal{C}'} U'$$

It is immediate from the construction that $\forall (U, U') \in \mathcal{C}' : (U, U') \leq (U_s, U_s')$. It is also easy to see that if (U_s, U_s') is consistent, then it is the supremum of $\mathcal{C}'$. Thus, it suffices to show that (U_s, U_s') is consistent, and this can be established by an argument similar to the proof of Proposition 10. □

Theorem 3. *Given a poset of consistent utterance pairs* $(C^s_{\mathcal{U}(\mathrm{Reg},V)}, \leq)$ *and a function* $\mathcal{I}^s_{\mathcal{U}(\mathrm{Reg},V)} : C^s_{\mathcal{U}(\mathrm{Reg},V)} \to C^s_{\mathcal{U}(\mathrm{Reg},V)}$ *which is inflationary and monotonic,* $\mathcal{I}^s_{\mathcal{U}(\mathrm{Reg},V)}$ *has a least fixed point and a maximal fixed point.*

The proof is obtained as a corollary to Zorn's Lemma (cf. [50]), and we refer the reader to [14] for the argument.[6] To obtain the least fixed point, we consider the pair $(U_0, U'_0) \in C^s_{\mathcal{U}(\mathrm{Reg},V)}$, where $U_0 = \emptyset$ and $U'_0 = \mathcal{U}(\mathrm{Reg}, V)$. Let $(U_i, U'_i) = \mathcal{I}^s_{\mathcal{U}(\mathrm{Reg},V)}(U_{i-1}, U'_{i-1})$ for $i \geq 1$. It is easy to see that if $|\mathcal{U}(\mathrm{Reg}, V)|$ is finite (i.e., the number of objects O is finite), then there exists $n \in N$ such that (U_n, U'_n) is the least fixed point, i.e., $(U_n, U'_n) = (U_{n+i}, U'_{n+i})$ for all $i \in N$. We also note that in the case of finite domains, $|C^s_{\mathcal{U}(\mathrm{Reg},V)}|$ is finite, and so, the maximal fixed points can be enumerated (in theory). To make the approach practical, restrictions are needed. In [13], we explored an assumption that lets us compile out occurrences of *says* in the preconditions of laws, leading to efficient checking for states with a large number of objects. These methods need to be extended to accommodate reasoning that arises via representation (axioms **A5** and **A6**), and we leave an investigation to future work.

A state s together with a consistent utterance pair forms the basis for all decision problems. We define a notion of validity at a state, which we will use to formalize access control and conformance decisions (in Sections 3.4 and 3.5 resply.):

Definition 15 (Validity at a State)**.** *Given a state* s*, a body of regulation* Reg*, a consistent utterance pair* $(U, U') \in C^s_{\mathcal{U}(\mathrm{Reg},V)}$ *and a propositional* $\varphi \in L_\varphi$*, we say that* φ *is valid at* s *w.r.t.* Reg *and* (U, U')*, denoted* $(s, \mathrm{Reg}) \models_{(U,U')} \varphi$*, iff* $\mathbf{tv}_{(U,U')}(\varphi, s, v) = \top$ *for all* $v \in V$*.*

The choice of which utterance pair to use depends on the application. If there is a unique (least) fixed point, then it is the appropriate choice. However, matters are not so clear when there are multiple fixed points. We conclude this section with a discussion of examples to build intuition about the definitions of access control and conformance.

3.3.1. Examples

We discuss two examples. First, we consider the statements from HIPAA (presented in Section 3.1 and also discussed Section 3.2.2). A unique fixed point will be obtained for these statements. Second, we consider an example involving multiple fixed points.

Example 1: Consider the statements from HIPPA in Section 3.1. Let $H \in O_P$ stand for the regulatory authority behind HIPPA, and $l(H) = \{1, 1a, 1b\}$. As we discussed in Section 3.2.2, the phrase *has the right* is analysed as a power. We use the following abbreviation:

$$\mathrm{hasRight}(x, z, \varphi) = \mathcal{P}_x \,\mathrm{says}_{l(x)}\, \mathcal{O}_z \,\mathrm{says}_{l(z)}\, \mathcal{P}_x \varphi$$

[6] Kripke [37] describes Theorem 3 as being well-known to logicians. However, we have not found a standard reference for this proof.

The HIPAA rule is formalized as follows:

(1) $\text{pat}(x) \wedge \text{rec}(y, x, z) \wedge \neg \text{says}_{\{1a,1b\}}\, e(y) \mapsto \text{hasRight}(x, z, \text{access}(x, y))$
  (1a) $\text{psyNotes}(y_1) \mapsto e(y_1)$
  (1b) $\text{compForLegal}(y_2) \mapsto e(y_2)$

The law (1) is read as follows: "If x is a patient ($\text{pat}(x)$), and y is a record of x maintained by z ($\text{rec}(y, x, z)$), and HIPPA does not say that there is an exception applying to y ($\neg \text{says}_{\{1a,1b\}}\, e(y)$), then HIPAA says that x has the right to access her records via the law (1)".

The law (1a) is read as follows: "If y_1 is a record of psychotherapy notes ($\text{psyNotes}(y_1)$), then HIPAA says that an exception applies to y_1 via the law (1a)". And, the law (1b) is read as follows: "If y_2 is a record compiled for legal proceedings ($\text{compForLegal}(\text{y}_2)$), then HIPAA says that an exception applies to y_2 via the law (1b)".

We set aside the important problem of defining these predicates further, i.e., the definitional aspects of the law [33]. For example, HIPPA provides rules describing who counts as a patient, and the interpretation of $\text{pat}(x)$ is dependent on these rules. In addition, the predicate $e(y)$ could be interpreted as a permission to the maintainer of y not to grant access.[7] Such extensions are easily accommodated.

Suppose Alice (A) wants to view her records (r) which are maintained by Bob (B). Alice introduces the following rule:

(2) Bob must show me my records.

As discussed in Section 3.2.2, the manner in which this rule arises is left unspecified. For example, Alice may send an e-mail to Bob, requiring to see her records. Alice's law is formalized as follows. Let $A \in O_P$ stand for Alice, and $B \in O_P$ stand for Bob. In addition, $l(A) = \{2\}$. Law (2) is formally expressed as:

(2) $\top \mapsto \mathcal{O}_B\, \text{says}_{l(B)}\, \mathcal{P}_B \text{access}(A, r)$

Bob complies with this request via his access control policy. Suppose Bob's policy consists of the following rule:

(3) HIPAA is permitted to require me to permit a patient to access her records.

Let $l(B) = \{3\}$. Law (3) is formally expressed as:

(3) $\text{pat}(\text{x}_3) \wedge \text{rec}(y_3, x_3, B) \mapsto \mathcal{P}_H\, \text{says}_{l(H)}\, \mathcal{O}_B\, \text{says}_{l(B)}\, \mathcal{P}_{x_3} \text{access}(x_3, y_3)$

[7] Our understanding of the HIPAA rule is that (1a) and (1b) are only meant to cancel the right provided by (1), and do not entail any explicit permission to the maintainer of the records.

Table 1 shows a state together with the fixed point utterances obtained from $l(H)$, $l(A)$, and $l(B)$. Here, r is a record about Alice maintained by Bob ($\text{rec}(r, A, B)$), which has been compiled for legal proceedings ($\text{compForLegal}(r)$). The precondition of HIPAA's law (1b) is true, and we obtain the utterance $\text{says}_{\{1b\}}\, e(r)$. As a result, the precondition of (1) is false, and no right is granted to Alice. The preconditions of Alice's and Bob's laws ((2) and (3) resply.) are true, and the corresponding utterances are obtained.

Objs	Predicates	Fixed Point Utterances
H	$\text{pat}(A)$, $\text{rec}(r, A, B)$	$\text{says}_{\{1b\}}\, e(r)$
A, r	$\neg\text{psyNotes}(r)$	$\text{says}_{\{2\}}\, \mathcal{O}_B\, \text{says}_{l(B)}\, \mathcal{P}_B \text{access}(A, r)$
B	$\text{compForLegal}(r)$	$\text{says}_{\{3\}}\, \mathcal{P}_H\, \text{says}_{l(H)}\, \mathcal{O}_B\, \text{says}_{l(B)}\, \mathcal{P}_A \text{access}(A, r)$

Table 1: A state and fixed point utterances for the HIPAA example.

Let us consider the questions of access control and conformance informally, given the state and fixed point in Table 1. Is Alice permitted to access her record? No, because HIPAA does not require Bob to permit her to access it. Is Bob conformant? On one hand, HIPAA doesn't require anything of Bob, so *yes*. On the other hand, Alice says that Bob is required to permit her to access her records, and he does *not* comply with this request. Thus, conformance is better seen as a relation between two principals w.r.t. a set of laws. In Section 3.5, we will say that B conforms to A w.r.t. the laws $l(A)$ iff B satisfies the obligations imposed by those laws.

Example 2: We now discuss an example involving multiple fixed points, based on the well-known Nixon-diamond problem in Default Logic [49]. Consider the following laws:

(4) Except as otherwise specified, quakers must be pacifists.

(5) Except as otherwise specified, republicans must not be pacifists.

The laws are formally expressed as:

(4) $q(x) \wedge \neg\, \text{says}_{\{5\}}\, \neg\mathcal{O}_x p(x) \mapsto \mathcal{O}_x p(x)$

(5) $r(x) \wedge \neg\, \text{says}_{\{4\}}\, \neg\mathcal{O}_x \neg p(x) \mapsto \mathcal{O}_x \neg p(x)$

Law (4) is read as "If x is a quaker ($q(x)$), and the regulator does not say that x is not required to be a pacifist ($p(x)$) via law (5), then the regulator says that x must be a pacifist via law (4)". Law (5) is read similarly, and $r(x)$ is read as: "x is a republican".

Table 2 gives an example of a state, where a principal (N) for Nixon is a quaker ($q(N)$), a republican ($r(N)$), but not a pacifist ($\neg p(N)$). The least fixed point is given by (U, U'), where $U = \emptyset$ and U' consists of all utterances. The preconditions of both laws are *ungrounded*. This corresponds to skeptical reasoning in non-monotonic logic.

Two maximal fixed points are obtained. In the first fixed point in Table 2, denoted (U_1, U_1'), we have $U_1 = U_1' = \{\text{says}_{\{4\}}\, \mathcal{O}_N p(N)\}$. The precondition

of law (4) (resply., law (5)) is true (resply., false). In the second fixed point in Table 2, denoted (U_2, U_2'), we have $U_2 = U_2' = \{\text{says}_{\{5\}}\, \mathcal{O}_N \neg p(N)\}$. The precondition of law (5) (resply., law (4)) is true (resply., false). The maximal fixed points correspond to credulous reasoning in non-monotonic logic.

Objs	Predicates	Fixed Point 1	Fixed Point 2
N	$q(N)$, $r(N)$, $\neg p(N)$	$\text{says}_{\{4\}}\, \mathcal{O}_N p(N)$	$\text{says}_{\{5\}}\, \mathcal{O}_N \neg p(N)$

Table 2: A state and distinct maximal fixed points obtained from the Nixon-diamond example

Let us consider the question of conformance informally. Given the state in Table 2, does the principal N conform to the regulation? The answer depends on which fixed point we consider. If we consider the first fixed point, then the answer is *no*, because N is not a pacifist. If we consider the second fixed point, then the answer is *yes*, for the same reason. We note that if N wasn't both a quaker and a republican, a unique fixed point is obtained.

While the Nixon-diamond construction arises in the area of knowledge representation, the question of interest is whether there are regulations where multiple fixed points are needed. We have not encountered such examples.[8]

3.4. Non-interference in Access Control

An access control decision is made when a principal A requests the performance of action p which is controlled by B. Given a state s, regulation Reg and fixed point (U, U') resulting from the evaluation of policy, the decision problem is whether $(s, \text{Reg}) \models_{(U,U')} \text{says}_{l(B)}\, \mathcal{P}_A(p)$, i.e., does B say that A is permitted to perform p.

A problem with this definition is that the policies in access control are usually distributed. It is unreasonable to expect (U, U') to reside on a single system. Given that we wish to evaluate $\text{says}_{l(B)}\, \mathcal{P}_A(p)$, the question is whether a smaller set of utterances suffice to answer this question. In other words, the evaluation should be carried out locally by B or a designated evaluator for B, as in [6].

Non-interference properties are used to obtain such results, and to demonstrate that the logic protects the rights of each principal [2, 17]. In our case, the access control decision is of the form $(s, \text{Reg}) \models_{(U,U')} \text{says}_{Id_B}\, \psi$, and this holds iff $U \vdash \text{says}_{Id_B}\, \psi$. The goal is to identify a subset of utterances ($U^* \subseteq U$), such that $U \vdash \text{says}_{Id_B}\, \psi$ iff $U^* \vdash \text{says}_{Id_B}\, \psi$.

Let us consider an example to build some intuition. Suppose we have four principals A, B, C, and D, with $l(A) = \{id1\}$, $l(B) = \{id2\}$, $l(C) = \{id3\}$, and $l(D) = \{id4\}$. Suppose C is a patient, and A and B maintain records about C. A only permits C to access her records, while B permits C to permit her mother (D) to access her records. Let U consist of the following utterances:

[8] It is tempting to analyse conflicting obligations that can arise in contrary-to-duty (CTD) structures (c.f. [48]) using multiple fixed points. However, we do not believe that this is the right approach. The analysis of CTD structures is left to future work

(u1) $\text{says}_{l(A)}\, \mathcal{P}_C \text{access}(C, r)$

(u2) $\text{says}_{l(B)}\, \mathcal{P}_C\, \text{says}_{l(C)}\, \mathcal{P}_D \text{access}(D, r_1)$

(u3) $\text{says}_{l(C)}\, \mathcal{P}_D \text{access}(D, r_1)$

Now, suppose D wants to access C's records that are maintained by A. It is easy to see that $U \nvdash \text{says}_{l(A)}\, \mathcal{P}_D \text{access}(D, r)$. But, do we need all of the utterances in U to make this determination? Intuitively, *no*, because the only utterance from A is (u1) and there is no representation conveyed via (u1). So, (u1) alone should suffice to make this determination. In this case, we say that *(u2) and (u3) do not interfere with the access control decision.*

Next, suppose D wants to access C's records that are maintained by B. It follows that $U \vdash \text{says}_{l(B)}\, \mathcal{P}_D \text{access}(D, r_1)$, and so D is indeed granted access. Here, (u2) is certainly relevant, and since it gives the power of representation to C, (u3) is also relevant. However, no mention of A is made by (u2) or (u3), and so, *(u1) does not interfere with the access control decision.*

We begin by defining the subset of utterances that are relevant to an access control decision:

Definition 16 (Reachable Utterances)**.** *Given a set of utterances U and a formula $\text{says}_{Id_B}\, \psi$, $U^*_{Id_B}$ is the smallest set such that:*

- *If $id_B \in Id_B$ and $\text{says}_{\{id_B\}}\, \varphi \in U$, $\text{says}_{\{id_B\}}\, \varphi \in U^*_{Id_B}$*
- *If $\text{says}_{\{id_B\}}\, \varphi \in U^*_{Id_B}$ and $\text{says}_{Id_A}\, \psi'$ is a subformula of φ, then $U^*_{Id_A} \subseteq U^*_{Id_B}$*

If we think of formulas $\text{says}_{Id_B}\, \psi$ as *pointing* to utterances in U (labeled Id_B), then $U^*_{Id_B}$ is the set of utterances that are pointed to directly (the first clause), or pointed to by subformulas of utterances that are pointed to (the second clause). In these terms, the computation of the set $U^*_{Id_B}$ corresponds to a reachability computation on a graph, and hence, we call it the set of *reachable utterances.* We believe that it is reasonable to restrict to the reachable utterances, because given the question $\text{says}_{l(B)}\, \psi$, $U^*_{l(B)}$ is determined by B and the principals to whom she grants the power of representation. We can now show the following:

Theorem 4 (Non-interference)**.** *Given a set of utterances U, for all $\text{says}_{Id_B}\, \psi \in L$, we have $U \vdash \text{says}_{Id_B}\, \psi$ iff $U^*_{Id_B} \vdash \text{says}_{Id_B}\, \psi$*

Proof. One direction follows easily using propositional reasoning, i.e., if $U^*_{Id_B} \vdash \text{says}_{Id_B}\, \psi$, then $U \vdash \text{says}_{Id_B}\, \psi$, since $U^*_{Id_B} \subseteq U$.

For the other direction, we proceed by contradiction. Suppose $U \vdash \text{says}_{Id_B}\, \psi$, and $U^*_{Id_B} \nvdash \text{says}_{Id_B}\, \psi$. So, $\phi = U^*_{Id_B} \wedge \neg \text{says}_{Id_B}\, \psi$ is satisfiable. Let $M = (S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$ be a model of ϕ. Hence:

- There exists $s^\phi \in S$ such that $(M, s^\phi) \models \phi$ for some $s^\phi \in S$, and
- There exists $s^{\neg\psi} \in \delta_{\mathcal{L}}(s, Id_B)$ such that $(M, s^{\neg\psi}) \models \neg\psi$

We will construct a new model M' with a state s^* such that $(M', s^*) \models \bigwedge U$ and $(M', s^*) \models \neg \text{says}_{Id_B} \psi$. This would contradict the assumption that $U \vdash \text{says}_{Id_B} \psi$. The main difficulty with the construction is that ψ may have a subformula $\text{says}_{Id_C} \psi'$ such that there is a statement $\text{says}_{\{id_C\}} \varphi \in U - U^*_{Id}$ with $id_C \in Id_C$. Thus, changing the truth of $\text{says}_{\{id_C\}} \varphi$ could result in a change in the truth of $\text{says}_{Id_C} \psi'$. Handling this case makes the construction involved.

We construct a new model $M' = (S', I'_{\Phi_1}, ..., I'_{\Phi_n}, \delta'_{\mathcal{L}}, \delta'_{\mathcal{O}})$ as follows:
The states S': For each state $s \in S$, we assign a new state, denoted $c(s)$, which is to be understood as a copy of s. We assume that $c(s) \notin S$ and $c(s) = c(s')$ iff $s = s'$. Given $S_1 \subseteq S$, $c(S_1)$ denotes the set of states such that $c(s) \in c(S_1)$ iff $s \in S_1$. In addition, we add two special states s^* (at which the contradiction will be obtained) and s^W (which provides *witnesses* as needed for **C5** and **C6**). $S' = S \cup c(S) \cup \{s^*, s^W\}$
Interpretation of Predicates: $I'_{\Phi_1}, ..., I'_{\Phi_n}$ is the same as $I_{\Phi_1}, ..., I_{\Phi_n}$ with the copies of states having the same assignment as the states in S. No predications hold at s^* and s^W.
Accessibility Relation $\delta'_{\mathcal{O}}$: $\delta'_{\mathcal{O}}$ respects $\delta_{\mathcal{O}}$ for $s \in S$. $\delta'_{\mathcal{O}}(c(s), A) = c(\delta_{\mathcal{O}}(s, A))$, for all $A \in O_P$ and $c(s) \in c(S)$. In addition, $\delta_{\mathcal{O}}(s^*, A) = \{s^*\}$, and $\delta_{\mathcal{O}}(s^W, A) = \{s^W\}$, for all $A \in O_P$.
Accessibility Relation $\delta'_{\mathcal{L}}$: This is the main part of the construction. $\delta'_{\mathcal{L}}$ respects $\delta_{\mathcal{L}}$ for $s \in S$. For all $A \in O_P$ and $Id_A \subseteq l(A)$, $\delta'_{\mathcal{L}}(s^W, Id_A) = S'$. We now describe the construction for the other states, starting with some notation.

Given $Id_A \subseteq l(A)$, let Id^*_A be the set such that for all $id_A \in Id_A$, $id_A \in Id^*_A$ iff $id_A \in Id_B$ or there exists a subformula $\text{says}_{Id'_A} \varphi \in U^*_{Id_B}$ such that $id_A \in Id'_A$.
The state s^*- For all $A \in O_P$ and $Id_A \subseteq l(A)$, we have the following cases:

- If $Id^*_A \neq Id_A$, then $\delta'_{\mathcal{L}}(s^*, Id_A) = \emptyset$
- Otherwise, $\delta'_{\mathcal{L}}(s^*, Id_A) = \delta_{\mathcal{L}}(s^\phi, Id_A) \cup c(\delta_{\mathcal{L}}(s^\phi, Id_A))$.

The first clause ensures that $(M', s^*) \models \text{says}_{\{id_A\}} \varphi$ for all $\text{says}_{\{id_A\}} \varphi \in U - U^*_{Id_B}$, since $\{id_A\}^* = \emptyset$. The second clause adds both the states that are accessible from s^ϕ and their copies. The accessibility relations associated with a copy ($c(s^{\neg\psi})$) will be modified in order to preserve **C5**.
The copies – For all $C \in O_P$, $Id_C \subseteq l(C)$, and $c(s) \in c(S)$:

- If $c(s) \notin \delta'_{\mathcal{O}}(c(s^{\neg\psi}), C)$ or $Id^*_A = Id_A$, then $\delta'_{\mathcal{L}}(c(s), Id_C) = c(\delta_{\mathcal{L}}(s, Id_C))$.
- Otherwise, $\delta'_{\mathcal{L}}(c(s), Id_C) = \delta_{\mathcal{L}}(s^\phi, Id_C) \cup c(\delta_{\mathcal{L}}(s^\phi, Id_C)) \cup c(\delta_{\mathcal{L}}(s, Id_C)) \cup \{s^W\}$.

Note that the second clause does not affect the truth of any subformula in $U^*_{Id_B}$, and it ensures that there are *witnesses* as needed for **C5** for the cases where $\delta'_{\mathcal{L}}(c(s^\phi), Id_A) = \emptyset$.
Frame Constraints: We need to verify that the frame constraints hold in M'. The only difficulty is in showing that **C5** holds at the copies and s^*. Fix A, C, Id_A and Id_C. Given $c(s) \in c(S)$, there are two cases:

- $c(s) \notin \delta'_{\mathcal{O}}(c(s^{\neg\psi}), A)$ or $Id^*_A = Id_A$. By construction, $\delta'_{\mathcal{L}}(c(s), Id_A) = c(\delta_{\mathcal{L}}(s, Id_A))$. Consider $c(s') \in \delta'_{\mathcal{L}}(c(s), Id_A)$. Since **C5** holds at s in M:
  - $s' \in \delta_{\mathcal{L}}(s, Id_C)$, in which case $c(s') \in \delta'_{\mathcal{L}}(c(s), Id_C)$, or
  - There exists $s_1 \in \delta_{\mathcal{L}}(s, Id_A)$ such that for all $s_2 \in \delta_{\mathcal{O}}(s_1, C)$, we have $s' \in \delta_{\mathcal{L}}(s_2, Id_C)$. By construction, for all $c(s_2) \in \delta'_{\mathcal{O}}(c(s_1), C)$, we have $c(s') \in c(\delta_{\mathcal{L}}(s_2, Id_C)) \subseteq \delta'_{\mathcal{L}}(c(s_2), Id_C)$.
- Otherwise, by construction, $s^W \in \delta_{\mathcal{L}}(c(s), Id_A)$, and $\delta'_{\mathcal{O}}(s^W, C) = \{s^W\}$. Since $\delta'_{\mathcal{L}}(c(s), Id_A) \subseteq \delta'_{\mathcal{L}}(s^W, Id_C) = S'$, **C5** is trivially satisfied.

Next we consider the state s^* for which there are three cases:

1. $Id^*_A \neq Id_A$. $\delta'_{\mathcal{L}}(s^*, Id_A) = \emptyset$ and **C5** is vacuously satisfied.
2. $Id^*_A = Id_A$ and $Id^*_C \neq Id_C$. For each $c(s) \in \delta'_{\mathcal{O}}(c(s^{\neg\psi}), C)$, we have $\delta'_{\mathcal{L}}(s^*, Id_A) \subseteq \delta'_{\mathcal{L}}(c(s), Id_C)$, thereby satisfying **C5**.
3. $Id^*_A = Id_A$ and $Id^*_C = Id_C$. In this case, **C5** is satisfied because **C5** holds in M and the copies of states are isomorphic.

Establishing the contradiction: The following are established easily by induction:

(P1) For all $s \in S$ and $\varphi \in L$, $(M, s) \models \varphi$ iff $(M', s) \models \varphi$

(P2) For all $s \in S$ and $\varphi \in L$ such that for all subformulas $\text{says}_{Id_A} \varphi'$ of φ, $Id^*_A = Id_A$, $(M, s) \models \varphi$ iff $(M', c(s)) \models \varphi$.

We can now reason as follows:

1. $(M', s^*) \models \bigwedge U^*_{Id_B}$, since for all $\text{says}_{\{id_A\}} \varphi \in U^*_{Id_B}$ and $s' \in \delta'_{\mathcal{L}}(s^*, \{id_A\})$, the following condition holds. Either $s' \in \delta'_{\mathcal{L}}(s^\phi, \{id_A\})$, in which case $(M', s) \models \varphi$ (using (P1)), or $s' \in \delta'_{\mathcal{L}}(c(s^\phi), \{id_A\})$, in which case $(M', s') \models \varphi$ (using (P2)).
2. $(M', s^*) \models \text{says}_{\{id_A\}} \varphi$, for all $\text{says}_{\{id_A\}} \varphi \in U - U^*_{Id_B}$ (by construction, since $\{id_A\}^* \neq \{id_A\}$ and $\delta'_{\mathcal{L}}(s^*, \{id_A\}) = \emptyset$)
3. Hence, $(M', s^*) \models \bigwedge U$
4. $(M', s^*) \models \neg\, \text{says}_{Id_B} \psi$, since $s^{\neg\psi} \in \delta_{\mathcal{L}}(s^*, Id_B)$, and $(M', s^{\neg\psi}) \models \neg\psi$ (using (P1))

The last two items contradict the assumption that $U \vdash \text{says}_{Id_B} \psi$. □

We note that the distinction between the inference component and the saying component allows us to restrict attention to inferences of the form $U \vdash \text{says}_{Id_B} \varphi$, where U only has formulas of the form $\text{says}_{Id_A} \psi$. If the set U could contain arbitrary formulas, non-interference would have a more complex characterization, as in [17]. For example, if we allowed formulas of the form $\neg\, \text{says}_{Id_A} \psi$ in U, then any principal can render U inconsistent.

3.5. Conformance

We now turn to the definition of conformance. While the definition of conformance has some variation between the various formalisms [4, 14, 19–21, 27, 38], all of them require a principal to satisfy the obligations that are imposed on her. In the context of contracts, several works [20, 21, 27, 38] accommodate reasoning about mitigating actions such as "paying a fine" if an obligation is not satisfied. The analysis of such mitigating actions is left to future work.

We define conformance as a relation between a principal and another principal w.r.t. a set of laws:

Definition 17 (Conformance)**.** *Given a state s with a set of objects O, a body of regulation* Reg*, and $\{A, B\} \subseteq O_P$, we say that A conforms to B w.r.t. the laws $Id_B \subseteq l(B)$ and a fixed point (U, U') with $U = U'$ iff for all propositional $\varphi \in L_{\varphi_A}$:*

If $(s, \mathrm{Reg}) \models_{(U,U')} \mathrm{says}_{Id_B} \mathcal{O}_A\varphi$*, then* $(s, \mathrm{Reg}) \models_{(U,U')} \varphi$

In other words, conformance is the satisfaction of all obligations. The syntactic restrictions in Definition 1 justify the restriction to $\varphi \in L_{\varphi_A}$, as these are the only formulas that can appear within the scope of $\mathcal{O}_A$. The restriction to fixed points (U, U'), where $U = U'$, ensures that all formulas are either true or false. Definition 17 is not appropriate when $U \neq U'$, since classically provable formulas, e.g. $\varphi \vee \neg\varphi$, may be ungrounded. In such cases, the principal would be found (trivially) non-conformant. One way to accomodate these cases is to modify Definition 17 so that if $(s, \mathrm{Reg}) \models_{(U,U')} \mathrm{says}_{Id_B} \mathcal{O}_A\varphi$, we require only that φ be true or ungrounded. With this modification, our proof of decidability carries over to the case where $U \neq U'$.

Let us apply Definition 17 to our example from HIPAA in Table 1 (Section 3.3). As we discussed, we are interested in the conformance of Bob (B). Bob does not conform to Alice (A) w.r.t. the law $l(A) = \{2\}$, because $(s, \mathrm{Reg}) \models_{(U,U')} \mathrm{says}_{l(A)} \mathcal{O}_B \mathrm{says}_{l(B)} \mathcal{P}_A\mathrm{access}(A, r)$ and $(s, \mathrm{Reg}) \not\models_{(U,U')} \mathrm{says}_{l(B)} \mathcal{P}_A\mathrm{access}(A, r)$. However, it can be shown that Bob conforms to HIPAA (H), w.r.t. the laws $\{1, 1a, 1b\} = l(H)$. We will discuss additional examples in Section 4.2.

We now discuss the proof of decidability of conformance. Given a state s and a fixed point (U, U'), there are potentially infinitely many formulas $\varphi \in L_{\varphi_A}$ such that $(s, \mathrm{Reg}) \models_{(U,U')} \mathrm{says}_{Id_B} \mathcal{O}_A\varphi$. For example, if there is some φ such that $(s, \mathrm{Reg}) \models_{(U,U')} \mathrm{says}_{Id_B} \mathcal{O}_A\varphi$, then for all $\varphi' \in L_{\varphi_A}$, we have $(s, \mathrm{Reg}) \models_{(U,U')} \mathrm{says}_{Id_B} \mathcal{O}_A(\varphi \vee \varphi')$. We will prove that it suffices to restrict attention to a single formula, which may be understood as a prime implicant of all the obligations imposed on A via the laws Id_B.

The proof relies on properties of the canonical model of a formula (Definition 6). We begin with some notation. Given $\phi \in L$, let $M_\phi = (S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$ be the canonical model of ϕ. Recall that each state $\Delta \in S$ is a maximal consistent set of subformulas of ϕ, i.e., $\Delta \subseteq \mathrm{sub}(\phi)$. Given $\Delta \in S$ and $Id_B \subseteq l(B)$, Δ_{Id_B} is the set such that $\varphi \in \Delta_{Id_B}$ iff there exists $Id'_B \subseteq Id_B$ such that $\mathrm{says}_{Id'_B} \varphi \in \Delta$. Similarly, given $\Delta \in S$ and $A \in O_P$, Δ_A is the set such that $\varphi \in \Delta_A$ iff $\mathcal{O}_A\varphi \in \Delta$.

We now establish some properties of maximal consistent sets that are useful in the proof.

Proposition 12. *Given $\phi \in L$, let $M_\phi = (S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$ be the canonical model of ϕ. The following hold for all $\varphi \in L_{\varphi_A}$, $\psi \in L_\psi$ and $\Delta \in S$:*

1. *If for all $\Delta' \in \delta'_{\mathcal{L}}(\Delta, A)$, $\Delta' \vdash \varphi$, then $\Delta \vdash \mathcal{O}_A\varphi$*
2. *If for all $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_B)$, $\Delta' \vdash \psi$, then $\Delta \vdash \text{says}_{Id_B}\psi$*
3. *If $\Delta \vdash \text{says}_{Id_B}\mathcal{O}_A\varphi$, then for all $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_B)$, $\Delta'_A \vdash \varphi$*

Proof. For the first two items, we will need the following observation. Given $\Gamma \subseteq \text{sub}(\phi)$, let $S_\Gamma \subseteq S$ be the set such that $\Delta \in S_\Gamma$ iff $\Gamma \subseteq \Delta$. Then, for all $\varphi \in L$:

$(*)$ $\Gamma \vdash \varphi$ iff for all $\Delta \in S_\Gamma$, $\Delta \vdash \varphi$.

This follows using propositional reasoning, since S is the set of all maximal consistent sets w.r.t. $\text{sub}(\varphi)$, and S_Γ is the set of all maximal consistent sets containing Γ.
Item 1: Consider $\varphi \in L_{\varphi_A}$ such that for all $\Delta' \in \delta'_{\mathcal{L}}(\Delta, A)$, $\Delta' \vdash \varphi$. By construction, $\delta_{\mathcal{L}}(\Delta, A) = S_{\Delta_A}$, and by $(*)$, $\Delta_A \vdash \varphi$. Using **R2**, $\vdash \mathcal{O}_A(\bigwedge \Delta_A \Rightarrow \varphi)$. Since $\Delta \vdash \mathcal{O}_A(\bigwedge \Delta_A)$, using **A2**, $\Delta \vdash \mathcal{O}_A\varphi$. The proof of item 2 is similar.
Item 3: We proceed by contradiction. Suppose there exists $\varphi \in L_{\varphi_A}$ and $\Delta \in S$ such that $\Delta \vdash \text{says}_{Id_B}\mathcal{O}_A\varphi$, and $\Delta'_A \nvdash \varphi$ for some $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_B)$. So, there exists a model $M' = (S', I'_{\Phi_1}, ..., I'_{\Phi_n}, \delta'_{\mathcal{L}}, \delta'_{\mathcal{O}})$ and $s^{\neg\varphi} \in S'$ such that $(M', s^{\neg\varphi}) \models \bigwedge \Delta'_A$ and $(M', s^{\neg\varphi}) \models \neg\varphi$. We construct a new model $M'' = (S'', I''_{\Phi_1}, ..., I''_{\Phi_n}, \delta''_{\mathcal{L}}, \delta''_{\mathcal{O}})$ combining M_ϕ and M' as follows:

- $S'' = S \cup S'$. We assume that S and S' are disjoint.
- The interpretation of predicates respects those in M_ϕ and M'
- $\delta''_{\mathcal{L}}$ respects the accessibility relations $\delta_{\mathcal{L}}$ and $\delta'_{\mathcal{L}}$
- $\delta''_{\mathcal{O}}$ respects the accessibility relations $\delta_{\mathcal{O}}$ and $\delta'_{\mathcal{O}}$, except that: $\delta''_{\mathcal{O}}(\Delta', A) = \delta_{\mathcal{O}}(\Delta', A) \cup \{s^{\neg\varphi}\}$

The satisfaction of the constraints **C3**-**C6** is immediate from the construction, as the only modification is to $\delta''_{\mathcal{O}}(\Delta', A)$. The following are established easily by induction:

(1) For all $s \in S'$, $(M'', s) \models \psi$ iff $(M', s) \models \psi$

(2) For all $\Delta \in S$ and $\psi \in \text{sub}(\bigwedge U)$, $(M'', \Delta) \models \psi$ iff $\psi \in \Delta$

We can now reason as follows:

(3) $(M'', \Delta) \models \bigwedge \Delta$ (using (2))

(4) $(M'', s^{\neg\varphi}) \not\models \varphi$ (using (1))

(5) $(M'', \Delta') \not\models \mathcal{O}_A\varphi$ (from (4) since $s^{\neg\varphi} \in \delta''_{\mathcal{O}}(\Delta', A)$)

(6) $(M'', \Delta) \not\models \text{says}_{Id_B} \mathcal{O}_A\varphi$ (from (5) since $\Delta' \in \delta''_{\mathcal{L}}(\Delta, Id_B)$)

(7) $\Delta \not\vdash \text{says}_{Id_B} \mathcal{O}_A\varphi$ (from (3) and (6), by soundness)

Item (7) contradicts the assumption that $\Delta \vdash \text{says}_{Id_B} \mathcal{O}_A\varphi$. □

We are now ready to show that conformance checking is decidable:

Theorem 5 (Decidability of Conformance)**.** *Given a state S, a body of regulation* Reg*, a fixed point (U, U') where $U = U'$ and $|U|$ is finite, principals $\{A, B\} \subseteq O_P$, and identifiers $Id_B \subseteq l(B)$, there is a procedure to decide whether A conforms to B w.r.t. the laws Id_B.*

Proof. First, we observe that for all $\varphi \in L_{\varphi_A}$, $(s, \text{Reg}) \models_{(U,U')} \text{says}_{Id_B} \mathcal{O}_A\varphi$ iff $U \vdash \text{says}_{Id_B} \mathcal{O}_A\varphi$ (by definition). So, it suffices to check that for all $\varphi \in L_{\varphi_A}$, if $U \vdash \text{says}_{Id_B} \mathcal{O}_A\varphi$, then $(s, \text{Reg}) \models_{(U,U')} \varphi$

The key idea is to show that there is a formula $\varphi_U \in L_{\varphi_A}$ such that:

(P1) $U \vdash \text{says}_{Id_B} \mathcal{O}_A\varphi_U$, and

(P2) For all $\varphi \in L_{\varphi_A}$ such that $U \vdash \text{says}_{Id_B} \mathcal{O}_A\varphi$, we have $\vdash \varphi_U \Rightarrow \varphi$.

Assuming that such a φ_U exists, we can show the following:

- A conforms to B w.r.t. Id_B iff $(s, \text{Reg}) \models_{(U,U')} \varphi_U$

If A conforms to B w.r.t. Id_B, since $U \vdash \text{says}_{Id_B} \mathcal{O}_A\varphi_U$, we have $(s, \text{Reg}) \models_{(U,U')} \varphi_U$. For the other direction, we need the observation that for all $\phi \in L_{\varphi_A}$, if $\vdash \phi$, then $(s, \text{Reg}) \models_{(U,U')} \phi$. Note that this claim does not hold when $U \neq U'$. When $U = U'$, the claim is easily verified by showing that the axioms **A1**-**A3**, **A5**, and **A6** are valid at s w.r.t. (U, U'), and that the rules **R1** and **R2** preserved validity. Instances of axiom schema **A4** are not in L_{φ_A}. Now suppose that $(s, \text{Reg}) \models_{(U,U')} \varphi_U$. For all $\varphi \in L_{\varphi_A}$ such that $U \vdash \text{says}_{Id_B} \mathcal{O}_A\varphi$, we have $(s, \text{Reg}) \models_{(U,U')} \varphi_U \Rightarrow \varphi$ (using (P2)), and since $(s, \text{Reg}) \models_{(U,U')} \varphi_U$, $(s, \text{Reg}) \models_{(U,U')} \varphi$. Thus, if $(s, \text{Reg}) \models_{(U,U')} \varphi_U$, then A conforms to B w.r.t. Id_B. Since checking whether $(s, \text{Reg}) \models_{(U,U')} \varphi_U$ is decidable, conformance checking is decidable, provided that such a φ_U exists.

We now turn to the construction of φ_U. Let $M_U = (S, I_{\Phi_1}, ..., I_{\Phi_n}, \delta_{\mathcal{L}}, \delta_{\mathcal{O}})$ be the canonical model for $\bigwedge U$. Let $S_U = \{\Delta_1 | \Delta_1 \in S \text{ and } U \subseteq \Delta_1\}$. We will now define a formula φ_Δ for each $\Delta \in S_U$, and define φ_U as their disjunction:

$$\varphi_\Delta = \bigvee_{\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_B)} \bigwedge \Delta'_A \qquad \varphi_U = \bigvee_{\Delta \in S_U} \varphi_\Delta$$

We claim the following for all $\Delta \in S$:

(P3) $\Delta \vdash \text{says}_{Id_B} \mathcal{O}_A\varphi_\Delta$

(P4) For all $\varphi \in L$, if $\Delta \vdash \text{says}_{Id_B} \mathcal{O}_A\varphi$, then $\vdash \varphi_\Delta \Rightarrow \varphi$

Proof of (P3): Using propositional reasoning, for all $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_B)$ and $\Delta'' \in \delta_{\mathcal{O}}(\Delta', A)$, $\Delta'' \vdash \varphi_\Delta$. Hence, for all $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_B)$, by Proposition 12 Item 1, we have $\Delta' \vdash \mathcal{O}_A\varphi_\Delta$. And using, Proposition 12 Item 2, $\Delta \vdash \text{says}_{Id_B}\, \mathcal{O}_A\varphi_\Delta$.
Proof of (P4): Suppose $\Delta \vdash \text{says}_{Id_B}\, \mathcal{O}_A\varphi$. By Proposition 12 Item 3, for all $\Delta' \in \delta_{\mathcal{L}}(\Delta, Id_B)$, $\Delta'_A \vdash \varphi$. And, using propositional reasoning, $\vdash \varphi_\Delta \Rightarrow \varphi$
Proof of (P1): Using (P3), for all $\Delta \in S_U$, we have $\Delta \vdash \text{says}_{Id_B}\, \mathcal{O}_A\varphi_\Delta$. And, by propositional reasoning, $U \vdash \text{says}_{Id_B}\, \mathcal{O}_A\varphi_\Delta$ (since S_U is the set of all maximal consistent sets containing U).
Proof of (P2): Using (P4), for all $\varphi \in L$, if $U \vdash \text{says}_{Id_B}\, \mathcal{O}_A\varphi$, then $\Delta \vdash \text{says}_{Id_B}\, \mathcal{O}_A\varphi$ for all $\Delta \in S_U$. Hence, $\vdash \varphi_\Delta \Rightarrow \varphi$ for all $\Delta \in S_U$, and by propositional reasoning, $\vdash \varphi_U \Rightarrow \varphi$. □

4. Discussion

In this section, we discuss how various constructs from the literature are expressed in our framework. In Section 4.1, we discuss access control examples. Section 4.2 discusses conformance in the presence of nested obligations and permissions. We then discuss other relationships to prior work, in Section 4.3.

4.1. Access Control

We discuss two access control examples in this section. The first example highlights an important restriction of the policies in Section 3.3, i.e., a policy lets us conclude what has been said, but not what actually happens. The second example illustrates how the delegation operator of Li et al. [39] can be defined in our framework.
Example 1: We begin with an example from Garg and Abadi [16]. Consider a file-access scenario with an administrating principal (A), a user (B), a file (file1), and the following policy:

1. If A says that file1 should be deleted, then this must be the case.
2. A trusts B to decide whether file1 should be deleted.
3. B wants to delete file1.

We introduce a new principal F for the file system. The following are the utterances (U) obtained at the fixed point:

1. $\text{says}_{l(F)}\, \mathcal{P}_A\, \text{says}_{l(A)}\, \mathcal{O}_F(\text{delfile1})$
2. $\text{says}_{l(A)}\, \mathcal{P}_B\, \text{says}_{l(B)}\, \mathcal{O}_F(\text{delfile1})$
3. $\text{says}_{l(B)}\, \mathcal{O}_F(\text{delfile1})$

The first utterance is read as follows: The file system F says that A is permitted to require it (F) to delete file1. The second utterance is the delegation from A to B, and the third utterance is B's wish to delete file1. Using **A5**, we will conclude that $U \vdash \text{says}_{l(F)}\, \mathcal{O}_F(\text{delfile1})$. In other words, we conclude that the system requires itself to delete file1.

Our analysis differs in an important way from Garg and Abadi [16]. We do not conclude that file1 is actually deleted, i.e., $U \nvdash$ delfile1. In fact, we can show that there is no policy (as defined in Section 3.3) that lets us make this conclusion. delfile1 is true at a state where F conforms to $l(F)$, as per Definition 17. In some cases, it may be warranted to assume/axiomatize self-conformance, i.e., $(\text{says}_{l(F)}\, \mathcal{O}_F(\varphi)) \Rightarrow \varphi$. However, conflicting self-imposed requirements would make U inconsistent.

Example 2: The delegation operator of Li et al. [39] has a compelling definition in our framework. The syntax (in Li et al. [39]) for delegation is "x delegates $(\varphi)^d$ to y", where d is the depth of delegation. We define the schema $\text{ps}(\varphi, x, d)$, where x is used to generate variable names, and $d \in N$:

- $\text{ps}(\varphi, x, 1) = \mathcal{P}_{x_1}\, \text{says}_{l(x_1)}\, \varphi$
- $\text{ps}(\varphi, x, d) = \mathcal{P}_{x_d}\, \text{says}_{l(x_d)}(\varphi \wedge \text{ps}(\varphi, x, d-1))$, for $d > 1$

The statement "A delegates $(\text{delfile1})^2$ to B" is interpreted as follows: A says delfile1 if B says it or anyone that B trusts says it. Suppose, in addition, that B delegates $(\text{delfile1})^1$ to C, and C says delfile1. We express this with the following rules:

- (1) $(x_2 = B) \mapsto \text{ps}(\text{delfile1}, x, 2)$
- (2) $(y_1 = C) \mapsto \text{ps}(\text{delfile1}, y, 1)$
- (3) $\top \mapsto \text{delfile1}$

We assume that $1 \in l(A)$, $2 \in l(B)$ and $3 \in l(C)$. At the fixed point, we will have $U \vdash \text{says}_{l(A)}\, \text{delfile1}$, i.e., A says delfile1. Further re-delegations by C (by modifying statement 3) will not be attributed to A.

In the logic of Li et al. [39], a representation statement is used to grant permission to speak *without consuming delegation depth*. If C represents B on delfile1, then C should be permitted to at most one re-delegation. Statement 2 is modified as follows:

- (2) $(y_2 = C) \mapsto \text{ps}(\text{delfile1}, y, 2)$

With this modification, a delegation by C will be attributed to A. The reader may have noticed the similarity between statement 1 and the modified version of statement 2. In our approach, delegation is just a special kind of representation. A delegates $(\varphi)^d$ to B iff B represents A on "delegating $(\varphi)^{d-1}$ to anyone". If C represents B on "delegating $(\varphi)^{d-1}$ to anyone", then she represents A as well.

As Li et al. [39] point out, in the presence of representation, delegation depth does not have much meaning. For example, A may not wish to trust C to the same extent as B. There are a few options to address this issue by modifying the representation axiom. One way is to keep track of the delegation depth in the axiom, as in the SECPal language [7]. Yet another way is to keep track of the principal on behalf of whom a statement in made. We avoid these modifications, to simplify presentation.

4.2. Nested Obligations and Permissions

We discuss two examples of conformance in the presence of nested obligations and permissions. The first example illustrates how several fine-grained notions of conformance can be captured, and is intended to supplement the example from HIPAA in Section 3. The second example points out an important practical difficulty.

Example 1: Consider the following law:

(6) The owners of parking lots ought to forbid parking near the entrance.

What does it mean to conform to (6)? We analyze this sentence as follows: "The owners of parking lots ought to (introduce laws that) forbid parking near the entrance.". In other words, (6) is an obligation to introduce a prohibition. If the owner introduces such a law, then the person parking is viewed as non-conformant, but it is the owner that needs to conform to (6). We can represent (6) in logic as follows:

$$(6)\ \mathrm{own}(x) \wedge \mathrm{p}(y) \mapsto \mathcal{O}_x\, \mathrm{says}_{l(x)}\, \mathcal{O}_y \neg \mathrm{pk}(y, x)$$

Here $\mathrm{own}(x)$ is true iff x is the owner of a parking lot, $\mathrm{p}(y)$ is true iff y is a person, and $\mathrm{pk}(y, x)$ is true iff y parks near the entrance of the lot owned by x. $l(x)$ refers to the laws that are introduced by x.

Let us assume a state $s = (I_{\Phi_1}, ..., I_{\Phi_n})$ in which A is the owner of a parking lot, and B parks near the entrances of A's lot. The true predications are: $\{\mathrm{own}(A),\ \mathrm{p}(B),\ \mathrm{pk}(B, A)\}$. In addition, A is assigned the identifier 7, i.e., $l(A) = \{7\}$. We will now consider two scenarios – (a) A does not introduce any laws, and (b) A introduces a law forbidding parking near the entrance. We are interested in the conformance (Definition 17) of the owner A and the driver B.

Scenario 1: Suppose that A does not introduce any laws. The fixed point utterance pair is:

$$U = U' = \{\mathrm{says}_{\{6\}}\, \mathcal{O}_A\, \mathrm{says}_{\{7\}}\, \mathcal{O}_B \neg \mathrm{pk}(B, A)\}$$

In this case, A does not conform to $\{6\}$ because:

- $U \vdash \mathrm{says}_{\{6\}}\, \mathcal{O}_A\, \mathrm{says}_{\{7\}}\, \mathcal{O}_B \neg \mathrm{pk}(B, A)$, but
- $(s, \mathrm{Reg}) \not\models_{(U,U')} \mathrm{says}_{\{7\}}\, \mathcal{O}_B \neg \mathrm{pk}(B, A)$

However, it can be shown that B conforms to $\{6\}$.

Scenario 2: Now suppose that A introduces the law:

(7) $\mathrm{p}(y) \mapsto \mathcal{O}_y \neg \mathrm{pk}(y, A)$

The fixed point utterance pair is:

$$U = U' = \{\mathrm{says}_{\{6\}}\, \mathcal{O}_A\, \mathrm{says}_{\{7\}}\, \mathcal{O}_B \neg \mathrm{pk}(B, A), \mathrm{says}_{\{7\}}\, \mathcal{O}_B \neg \mathrm{pk}(B, A)\}$$

It can be shown that A conforms to $\{6\}$. What about B? It is clear that B does not conform to $\{7\}$, but what about $\{6\}$? Observe that $U \vdash \mathrm{says}_{\{6\}}\, \mathcal{O}_B \neg \mathrm{pk}(B, A)$

(using the representation axiom **A5**), but $(s, \mathrm{Reg}) \not\models_{(U,U')} \neg\mathrm{pk}(B, A)$. Hence, B does not conform to $\{6\}$. In other words, the statement (6) conveys an obligation to A and if A conforms, the embedded obligation is conveyed to B. As we noted in Section 2.1, we are formalizing the notion of *speaking on someone's behalf*, i.e., the obligation (7) issued by A is understood as being on behalf of the issuer of (6). Some applications may need a distinction between the different senses of saying.

Example 2: Consider the following example:

(8) You are required to allow a patient to see his records.

By our analysis, (8) is an obligation on the hospital to provide a permission. Suppose that a hospital introduces such a permission in its policy. Has it conformed to (8)? The problem arises in distinguishing between *claimed permission*, and *actual permission*. A hospital claims that it permits a patient to see his records, by making an appropriate rule. On the other hand, a hospital actually permits a patient to see his records, by taking an action, e.g., sending the records via mail.

We suggest that a formalization of actual permission needs notions of *bringing about* or *seeing to it that* (e.g., [8, 32]). If a principal A *says* that she permits an action p, we need to check if she prevents p either by some other action or non-action. We leave an investigation of this issue to future work.

4.3. Related Work

We have discussed several relationships to prior work, in Sections 2, 4.1, and 4.2. In this section, we discuss other relationships, to identify interesting lines for further research.

Logic programming has been popular in access control [9, 12, 39]. The formalism that we adopted (Section 3.3) provides a way to integrate the logic programming approaches with the logics of saying [1–3, 16, 17], i.e., by evaluating saying using provability. The negation of provability gives a good interpretation to *didn't say*, thereby establishing a connection between saying and non-monotonic reasoning. Non-monotonic reasoning plays a useful role in formalizing exceptions to laws [5, 44, 51]. With regard to the introduction of modalities in utterances, our approach follows in the spirit of [23, 26], where defeasible logic is extended to include modalities. It is of interest to obtain a formal characterization of the relationships.

The non-interference theorem (Section 3.4) can be used to obtain relevant utterances for access control in a distributed setting. The techniques for distributed proofs, developed by Bauer et al. [6], are directly applicable here. However, the provability tests $U \vdash \mathrm{says}_{l(A)}\, \psi$ can be expensive, if U is large. Logic programs restrict the heads of rules to be atomic (as in [9, 12, 39]). This restriction to atomic formulas lets one decide provability in polynomial time. An important question is whether similar restrictions can be applied here to get polytime fragments. Disjunction is the main culprit, and leads to exponential

worst-case complexities. Even if we exclude disjunction syntactically, the representation axiom can lead to constraints involving disjunction, as we saw in Propositions 3 and 5. This leads to the following question: *Is there a fragment of the logic that accommodates representation, and yields a polytime decision procedure?*

Due to the problematic interactions between hand-off and classical logic (Section 2.1), intuitionistic approaches have been developed [2, 16, 17]. While we have focused on the classical setting here, the representation axiom can be adapted to the intuitionistic setting. However, as Garg and Pfenning [17] point out, a notion of constructivism is also desirable in an intuitionistic logic. Constructivism requires the meaning of an operator to be independent of others, and as a result, axioms which describe interaction between operators (such as the representation axiom) are excluded. While constructivism is important in programming languages (see [2, 17]), interaction axioms have also proved useful. For example, Halpern et al. [30] discuss 48 systems of knowledge and time. This leads to our next question: *Is there a more constructive form of the representation axiom, that yields a useful programming language?*

Finally, notions of time have been used in conformance checking [13, 19, 38]. The policies are used to synthesize monitors that are used to detect violations at runtime [13, 38]. Since the saying component (Section 3.3) uses the formalism in [13, 14], notions of linear time can be easily added here, and the monitor synthesis in [13] can be used directly. Once notions of time are available, we can place constraints on how a policy changes [24, 25]. This leads to our final question: *Are there useful interactions between saying and time, to characterize how a policy is updated?*

5. Conclusions

We have motivated and described a logic for access control and conformance. The focus was on the interaction between *saying* and *permission*, as needed for these applications. We proposed two axioms to characterize their interaction (Section 3.2), and showed how these axioms could be incorporated into a logic programming approach (Section 3.3).

A combined analysis of saying and permission yielded benefits to both applications. For access control, we find a way to avoid the problematic interaction between hand-off and classical reasoning. Our axioms yield a decidable logic with a complete semantics (Section 3.2), and we hope that they have intuitive appeal to the reader. For conformance, we obtained a characterization of legal power by nesting saying with obligation and permission. We showed, in Section 3.5, that conformance checking remains decidable.

We believe that the joint study of access control and conformance is a rich area for research. In Section 4.3, we identified avenues for further inquiry.

Acknowledgements: We would like to thank the anonymous reviewers for their detailed and helpful comments on earlier versions of this paper.

References

[1] Abadi, M., 2003. Logic in access control. In: Proceedings of the Symposium on Logic in Computer Science. pp. 228–233.

[2] Abadi, M., 2007. Access control in a core calculus of dependency. Electronic notes in Theoretical Computer Science 172, 5–31.

[3] Abadi, M., Burrows, M., Lampson, B., Plotkin, G., 1993. A calculus for access control in distributed systems. ACM Transactions on Programming Languages and Systems 15 (4), 706–734.

[4] Abrahams, A., 2002. Developing and executing electronic commerce applications with occurrences. Ph.D. thesis, Univeristy of Cambridge.

[5] Alchourron, C., Makinson, D., 1981. Hierarchies of regulation and their logic. In: Hilpinen, R. (Ed.), New Studies in Deontic Logic. Reidel, pp. 125–148.

[6] Bauer, L., Garriss, S., Reiter, M. K., 2007. Distributed proving in access control systems. In: Proceedings of the IEEE Computer Security Foundation Symposium. pp. 81–95.

[7] Becker, M. Y., Fournet, C., Gordon, A. D., 2007. Design and semantics of a decentralized authorization language. In: Computer Security Foundations Symposium. pp. 3–15.

[8] Belnap, N. D., Bartha, P., 1995. Marcus and the Problem of Nested Deontic Modalities. In: Sinnot-Armstrong, W., Raffman, D., Asher, N. (Eds.), Morality and Belief: Festschrift in Honour of Ruth Barcan Marcus. Cambridge University Press, pp. 174–197.

[9] Bertino, E., Catania, B., Ferrari, E., Perlasca, P., 2003. A logical framework for reasoning about access control models. ACM Transactions on Information Systems Security 6 (1), 71–127.

[10] Breaux, T. D., Vail, M. W., Anton, A. I., 2006. Towards regulatory compliance: Extracting rights and obligations to align requirements with regulations. In: Proceedings of the 14th IEEE International Requirements Engineering Conference. pp. 46–55.

[11] Cirillo, A., Jagadeesan, R., Pitcher, C., Riely, J., 2007. Do as I SaY! Programmatic access control with explicit identities. In: Proceedings of the IEEE Computer Security Foundations Symposium. pp. 16–30.

[12] Crampton, J., Loizou, G., Shea, G. O., 2001. A logic of access control. The Computer Journal 44 (1), 137–149.

[13] Dinesh, N., Joshi, A., Lee, I., Sokolsky, O., 2008. Checking traces for regulatory conformance. In: Proceedings of the Workshop on Runtime Verification. pp. 86–103.

[14] Dinesh, N., Joshi, A., Lee, I., Sokolsky, O., 2008. Reasoning about conditions and exceptions to laws in regulatory conformance checking. In: Proceedings of the Conference on Deontic Logic in Computer Science. pp. 110–124.

[15] Fitting, M., 1985. A Kripke/Kleene Semantics for logic programs. Journal of Logic Programming 2 (4), 295–312.

[16] Garg, D., Abadi, M., 2008. A modal deconstruction of access control logics. In: Proceedings of the 11th International Conference on Foundations of Software Science and Computation Structures (FoSSaCS). pp. 216–230.

[17] Garg, D., Pfenning, F., 2006. Non-interference in constructive authorization logic. In: 19th IEEE Computer Security Foundations Workshop. pp. 283–296.

[18] Gelati, J., Governatori, G., Rotolo, A., Sartor, G., 2004. Normative autonomy and normative co-ordination: Declarative power, representation, and mandate. Artificial Intelligence and Law 12 (1-2), 53–81.

[19] Giblin, C., Liu, A., Muller, S., Pfitzmann, B., Zhou, X., 2005. Regulations Expressed as Logical Models (REALM). In: Legal Knowledge and Information Systems (JURIX). pp. 37–48.

[20] Governatori, G., 2005. Representing business contracts in rule ml. International Journal of Cooperative Information Systems 14 (2-3), 181–216.

[21] Governatori, G., Milosevic, Z., Sadiq, S., 2006. Compliance checking between business processes and business contracts. In: 10th International Enterprise Distributed Object Computing Conference (EDOC). pp. 221–232.

[22] Governatori, G., Rotolo, A., 2006. Logic of violations: A gentzen system for reasoning with contrary-to-duty obligations. Australasian Journal of Logic 4, 193–215.

[23] Governatori, G., Rotolo, A., 2008. Bio logical agents: Norms, beliefs, intentions in defeasible logic. Autonomous Agents and Multi-Agent Systems 17 (1), 36–69.

[24] Governatori, G., Rotolo, A., 2008. Changing Legal Systems: Abrogation and Annulment. Part I: Revision of defeasible theories. In: Deontic Logic in Computer Science. pp. 3–18.

[25] Governatori, G., Rotolo, A., 2008. Changing Legal Systems: Abrogation and Annulment. Part II: Temporalised defeasible logic. In: Normative Multi Agent Systems. pp. 112–127.

[26] Governatori, G., Rotolo, A., 2008. A computational framework for institutional agency. Artificial Intelligence and Law 16 (1), 25–52.

[27] Grosof, B., Labrou, Y., Chan, H. Y., 1999. A declarative approache to business rules in contracts: Courteous logic programs in xml. In: ACM Conference on Electronic Commerce. pp. 68–77.

[28] Halpern, J. Y., 2001. Multi-agent only knowing. Journal of Logic and Compuation 11 (1), 41–70.

[29] Halpern, J. Y., Moses, Y., 1992. A guide to the completeness and complexity for modal logics of knowledge and belief. Artificial Intelligence 54 (3), 319–379.

[30] Halpern, J. Y., van der Meyden, R., Vardi, M. Y., 2004. Complete axiomatizations for reasoning about knowledge and time. SIAM Journal of Computing 33 (3), 674–703.

[31] Hohfeld, W. N., 1913. Fundamental legal conceptions as applied in judicial reasoning. Yale Law Journal 23, 16–59.

[32] Horty, J. F., Belnap, N. D., 1995. The Deliberative Stit: A Study of Action, Omission, Ability, and Obligation. Journal of Philosophical Logic 29, 109–136.

[33] Jones, A. J. I., Sergot, M. J., 1992. Deontic logic in the representation of law: Towards a methodology. Artificial Intelligence and Law 1, 45–64.

[34] Jones, A. J. I., Sergot, M. J., 1992. Formal specification of security requirements using the theory of normative positions. In: European Symposium on Reasearch in Computer Security (ESORICS). pp. 103–121.

[35] Jones, A. J. I., Sergot, M. J., 1996. A formal characterization of institutionalized power. Journal of the IGPL 4 (3), 429–445.

[36] Kanger, S., 1972. Law and logic. Theoria 38, 105–132.

[37] Kripke, S., 1975. Outline of a theory of truth. Journal of Philosophy 72, 690–716.

[38] Kyas, M., Prisacariu, C., Schneider, G., 2008. Run-time monitoring of electronic contracts. In: 6th International Symposium on Automated Technology for Verification and Analysis (ATVA'08). pp. 397–407.

[39] Li, N., Grosof, B. N., Feigenbaum, J., 2003. Delegation logic: a logic-based approach to distributed authorization. ACM Transactions on Information and System Security 6 (1), 128–171.

[40] Lindahl, L., 1977. Position and Change: A Study in Law and Logic. Synthese Library 112, D. Reidel, Dordrecht.

[41] Makinson, D., 1994. General patterns in non-monotonic reasoning. In: Gabbay, D., Hogger, C., Robinson, J. (Eds.), Handbook of Logic in Artificial Intelligence and Logic Programming. Oxford University Press, pp. 35–110.

[42] Makinson, D., van der Torre, L., 2000. Input/output logics. Journal of Philosophical Logic 29, 383–408.

[43] McCarty, L. T., 1989. A Language for Legal Discourse - I. Basic Features. In: Proceedings of the International Conference on Artifical Intelligence and Law. pp. 180–189.

[44] McCarty, L. T., Cohen, W. W., 1990. The case for explicit exceptions. In: Proceedings of the Workshop on Logic Programming and Nonmonotonic Reasoning. pp. 81–94.

[45] Minsky, N. H., Rozenshtein, D., 1987. System = program + users + law. In: Proceedings of the International Conference on Artificial Intelligence and Law. pp. 170–180.

[46] Moore, R. C., 1985. Semantical considerations on non-monotonic logic. Artificial Intelligence 25, 272–279.

[47] Nute, D., 1987. Defeasible reasoning. In: Proceedings of the Hawaii International Conference on System Science. pp. 470–477.

[48] Prakken, H., Sergot, M., 1996. Contrary-to-duty obligations. Studia Logica 57 (1), 91–115.

[49] Reiter, R., 1980. A logic for default reasoning. Artificial Intelligence 13, 81–132.

[50] Rudin, W., 1987. Real and Complex Analysis. McGraw-Hill Book Company.

[51] Sartor, G., 1991. The structure of norm conditions and nonmonotonic reasoning in law. In: Proceedings of the 3rd international conference on Artificial intelligence and law. pp. 155–164.

[52] Sergot, M. J., Sadri, F., Kowalski, R. A., Kriwaczek, F., Hammond, P., Cory, H. T., 1986. The British Nationality Act as a logic program. Communications of the ACM 29 (5), 370–86.

[53] von Wright, G. H., 1951. Deontic logic. Mind 60, 1–15.