Ordered Epistemic Logic: Semantics, Complexity and Applications

Hanne Vlaeminck and **Joost Vennekens** and **Maurice Bruynooghe** and **Marc Denecker**
Department of Computer Science, Katholieke Universiteit Leuven, Belgium
{hanne.vlaeminck,joost.vennekens,maurice.bruynooghe,marc.denecker}@cs.kuleuven.be

Abstract

Many examples of epistemic reasoning in the literature exhibit a stratified structure: defaults are formulated on top of an incomplete knowledge base. These defaults derive extra information in case information is missing in the knowledge base. In autoepistemic logic, default logic and ASP this inherent stratification is not preserved as they may refer to their own knowledge or logical consequences. Defining the semantics of such logics requires a complex mathematical construction. As an alternative, this paper further develops *ordered epistemic logic*. This logic extends first order logic with a modal operator and stratification is maintained. This allows us to define an easy to understand semantics. Moreover, inference tasks have a lower complexity than in autoepistemic logic and the logic integrates seamlessly into classical logic and its extensions. In this paper we also propose a generalization of ordered epistemic logic, which we call distributed ordered epistemic logic. We argue that it can provide a semantic foundation for a number of distributed knowledge representation formalisms found in the literature.

Introduction

Ordered Epistemic Logic (OEL) was first defined by Konolige (1988a) under the name *Hierarchic Autoepistemic Theories*. He observed that in non-monotonic reasoning the notion of inference from a *specific* body of knowledge often plays an important role. Recently, OEL was independently reintroduced by (Denecker et al. 2010) in order to merge the contributions of ASP (Brewka, Eiter, and Truszczyński 2011; Baral 2003) on the level of Knowledge Representation into classical first order logic (FO). As many of the original motivating examples of ASP involve defaults and autoepistemic propositions (Gelfond and Lifschitz 1991), this was done by adding an epistemic operator to FO.

We observed that these motivating examples for ASP, as well as many examples of Default Logic (DL) (Reiter 1980), often have a simple stratified structure: the goal is to reason on an existing incomplete knowledge base, e.g. by adding default assumptions.More complex examples can have several levels of stratification. However, neither autoepistemic logic (AEL) (Moore 1984), default logic, nor ASP preserve this inherent stratification. As a result, a theory in each of these logics is a theory that refers to its own information content through a reflexive epistemic operator (see (Denecker, Marek, and Truszczynski 2011) for a recent account). This is a source of complexity that complicates both their semantics and their reasoning procedures.

By contrast, OEL maintains a stratified representation where each level extends the knowledge of the lower levels. This simplifies the logic considerably, while still being able to handle a lot of useful applications from AEL or DL, as we will show here. Contrary to AEL, DL or ASP, an OEL theory always defines a unique belief set, represented as a set of possible worlds. We will show that OEL solves some well-known problems of ASP in the context of epistemic applications. Syntactically, OEL extends FO; the only difference with FO is that OEL is a *closed domain* version of FO: all possible worlds share the same domain and interpretation of terms, like many first order modal logics. With exception of this feature, OEL is a conservative extension of FO; its epistemic operator stands orthogonal to many other extensions of FO (e.g., types, inductive definitions, aggregates,…), and hence seamlessly integrates with them. Our examples here will include such extensions. By combining them, a very rich KR language is obtained in which many of the motivating examples in DL, AEL and ASP, as well as other extensions of FO such as FO(ID) (Denecker and Ternovska 2008), have a natural expression.

We here extend the initial work in (Konolige 1988a; Denecker et al. 2010), in several ways. First, we prove that, in a given finite domain, the data complexity of model checking, satisfiability checking and query answering for OEL theories is in Δ_2^P, which is indeed lower then for AEL and DL, where some instances of satisfiability checking problems can be proven to be Σ_2^P-complete. We also show how a model generator for OEL can be implemented.

Second, we illustrate the use of OEL and of model generation in the context of a scheduling problem with an epistemic component. Third, we extend OEL to a logic for distributed epistemic agents, which we call *distributed ordered epistemic logic* (d-OEL). Knowledge bases are still hierarchically ordered, but now theories at one level no longer automatically possess all the knowledge of lower levels.

Distributed ordered epistemic logic can cope with distributed knowledge, which makes it relevant for a number of new application areas. One example is the specification of access control policies. Formal specification languages used for this purpose (e.g., (Barker 2009)) often include a construct allowing one policy manager to query the knowledge of another. Another potential application area is the Semantic Web[1], which can be seen as a huge network linking different sources of data and knowledge, often in the form of ontologies. Several proposals haven been made to combine data from such ontologies, for example, through the use of *bridge rules* (Bouquet et al. 2003). Others address the need for expressing defaults such as “if source *x* does not specify the color of the car, we assume its color is black”. In yet other approaches, the web is seen as an open environment consisting of purely positive knowledge bases that do not contain negative information. This is to avoid potential inconsistencies between different sources (Berners-Lee, Dan Connolly, and Scharf 2005). Here, a limited form of “negation-as-failure” called *scoped negation* (Polleres, Feier, and Harth 2006; Berners-Lee, Dan Connolly, and Scharf 2005) has been proposed: this is an epistemic operator to query whether a specific source (e.g., some ontology) does not know a proposition. Common to all such applications is the presence of a number of knowledge sources that can query each other through some form of epistemic operator. We will argue that the logic d-OEL provides a simple formalism with a precise semantics to tackle some of these applications in a natural way.

This paper is organized as follows. First we recall some preliminaries; next we define syntax and semantics of both OEL and d-OEL, and prove a number of properties about both languages. After that, we have a closer look at model generation for OEL and d-OEL and prove our complexity results. Finally, we go into more detail about the relationship between OEL and other knowledge representation languages and we conclude.

Preliminaries on FO and AEL

We briefly recall the basic concepts of first order logic (FO) and its extensions. A vocabulary Σ consists of a set Σ^F of constant and function symbols and a set Σ^P of predicate symbols. Formulas φ over vocabulary Σ are constructed using standard inductive rules for $\wedge, \neg, \vee, \Rightarrow, \Leftrightarrow, \exists, \forall$. A Σ-interpretation I (or Σ-structure) consists of a domain U_I, for each function symbol F/n an n-ary function $F^I : U_I^n \rightarrow U_I$, and, for each predicate symbol P/n an n-ary relation $P^I \subseteq U_I^n$. The interpretation $t^{I\theta}$ of a term t under an interpretation I and variable assignment θ is defined in the standard way, as is the satisfaction relation $I\theta \models \varphi$ of a formula φ. The interpretation of variable-free ground terms t and sentences φ does not depend on the variable assignment θ, in which case we simply write t^I and $I \models \varphi$. We call an interpretation of a vocabulary σ such that $\sigma \subseteq \Sigma^F$ a pre-interpretation (i.e., a pre-interpretation consists of a domain and an interpretation of a subset of the function symbols of the vocabulary).

The family of languages FO($\cdot$) extends standard FO with a number of useful features. In this paper, we will use the logic $FO(sort, ar, agg)$ which extends FO with *sorts*, *arithmetic*, and *aggregates* such as *cardinality*, *sum*, or *average*. In the rest of this paper, we will abuse notation and refer to this logic as FO($\cdot$). For a formal definition of its syntax and semantics, we refer to (Denecker and Ternovska 2008).

A *Herbrand interpretation* for a vocabulary Σ is an interpretation I such that U_I is the *Herbrand universe* (i.e., the set of all ground terms that can be constructed using the constants and function symbols of Σ) and I interprets each ground term t by itself.

Next, we recall the basics of autoepistemic logic (Moore 1984). Let $\mathcal{L}$ be the language of propositional logic based on a set of atoms Σ. Extending this language with a modal operator K gives a language $\mathcal{L}_K$ of modal propositional logic. An autoepistemic theory is a set of formulas in this language $\mathcal{L}_K$. An *interpretation* or *world* is a subset of the vocabulary Σ. The set of all interpretations of Σ is denoted by $\mathcal{I}_\Sigma$, i.e., $\mathcal{I}^\Sigma = 2^\Sigma$. A *possible world structure* is a set of interpretations, i.e., the set of all possible world structures $\mathcal{W}_\Sigma$ is defined as $2^{\mathcal{I}_\Sigma}$. A possible-world structure can be viewed as a universal Kripke model with a total accessibility relation (Chellas 1980; Hughes and Cresswell 1996). Possible-world structures were used by Moore (Moore 1984) and later by Levesque (Levesque 1990) in the investigations of autoepistemic logic. For a formula $\varphi \in \mathcal{L}_K$, the truth function $\mathcal{H}_{Q,I}$, where Q is a possible-world structure, and I an interpretation, is defined as follows.

- For an atom p, $\mathcal{H}_{Q,I}(p) = \mathbf{t}$ iff $p \in I$
- $\mathcal{H}_{Q,I}(\varphi_1 \wedge \varphi_2) = \mathbf{t}$ iff $\mathcal{H}_{Q,I}(\varphi_1) = \mathbf{t}$ and $\mathcal{H}_{Q,I}(\varphi_2) = \mathbf{t}$.
- $\mathcal{H}_{Q,I}(\varphi_1 \vee \varphi_2) = \mathbf{t}$ iff $\mathcal{H}_{Q,I}(\varphi_1) = \mathbf{t}$ or $\mathcal{H}_{Q,I}(\varphi_2) = \mathbf{t}$.
- $\mathcal{H}_{Q,I}(\neg\varphi) = \mathbf{t}$ iff $\mathcal{H}_{Q,I}(\varphi) = \mathbf{f}$.
- $\mathcal{H}_{Q,I}(K\varphi) = t$, iff for every interpretation $J \in Q$, $\mathcal{H}_{Q,J}(\varphi) = \mathbf{t}$.

For every modal theory T, Moore defined an operator D_T on $\mathcal{W}_\Sigma$: $D_T(Q) = \{I | \mathcal{H}_{Q,I}(\varphi) = \mathbf{t}, \text{ for every } \varphi \in T\}$. The intuition behind this definition is as follows. The possible-world structure $D_T(Q)$ is a revision of a possible-world structure Q. This revision consists of the worlds that make all the formulas in T true in the context of the current belief state Q. Fixpoints of the operator D_T represent “stable” belief sets, i.e., they cannot be revised any further. Though originally, Moore used the term *stable expansion* to denote all *formulas* φ such that $K\varphi$ is true in such a fixpoint of the operator D_T, we will use the term stable expansion here to refer to the fixpoints itself.

Syntax and semantics

The logic OEL was defined first in (Konolige 1988a) and later independently in (Denecker et al. 2010). This logic has epistemic operators $K_{T'}$ that allow one theory T to query another theory T'. While T' may in turn query a third theory T'', cycles are not allowed. To enforce this hierarchical structure, the theories must be partially ordered. Moreover, in those papers, it was assumed that these partially ordered theories are increasingly expert in the domain; i.e., every T also knows everything that the theories below it know. In this section, we first recall the original definition of OEL

[1]http://www.w3.org/standards/semantic web/

and prove some additional properties of it. Then we generalize OEL to *distributed ordered epistemic logic* (d-OEL). In this logic a theory does not have all the knowledge of the theories below it. This is a useful extension, for example, it allows us to model agents with contradicting beliefs.

Ordered Epistemic Logic

The syntax of OEL is defined as follows.

Definition 1. An ordered epistemic theory over vocabulary Σ is a pair $\langle \mathcal{T}, \leq \rangle$ of a finite set $\mathcal{T}$ of theories and a partial order $\leq$ on $\mathcal{T}$, such that each $T \in \mathcal{T}$ is a theory in the language $\mathcal{L}_T$, which is inductively defined as first order logic with one extra formula-building rule: If φ is a formula in $\mathcal{L}_{T'}$ and $T' < T$, then $K_{T'}(\varphi)$ is in $\mathcal{L}_T$.

It is straightforward to extend the syntax (and the semantics we define further down) for OEL over first order logic with other objective language constructs from FO($\cdot$) such as definitions (where the modal expressions $K_{T'}(\varphi)$ can be used in the bodies of the definition rules) and aggregates.

If the partial order $\leq$ is clear from context, we often identify an ordered epistemic theory with its set of theories $\mathcal{T}$. For each $T \in \mathcal{T}$, we denote by Σ_T the vocabulary that appears objectively (i.e., not in the scope of some $K_{T'}$) in T. Similarly, we also define $\Sigma_{\leq T}$ as the set of predicate symbols in $\mathcal{T}_{\leq T}$, where $\mathcal{T}_{\leq T}$ denotes the restriction of $\mathcal{T}$ to $\{T' \in \mathcal{T} \mid T' \leq T\}$.

Self-referential statements such as $T = \{\neg K_T(P) \Rightarrow Q\}$ cannot be expressed in OEL. The benefit is that the semantics can be defined as an extension of FO semantics. The definition performs induction over the partial order $\leq$.

Definition 2. Let $\langle \mathcal{T}, \leq \rangle$ be an OEL theory over Σ and I_σ a pre-interpretation for a subset σ of Σ^F. We call the symbols of σ *rigid*. For an interpretation I of Σ extending I_σ, the relation I satisfies T, denoted $I \models_{OEL} T$, is defined as follows:

- For every $T' < T$, it holds that $I \models_{OEL} T'$;
- For an atom $P(\bar{t})$, $I \models_{OEL} P(\bar{t})$ iff $\bar{t}^I \in P^I$;
- For a modal literal $K_{T'}\varphi$, $I \models_{OEL} K_{T'}\varphi$ iff $J \models_{OEL} \varphi$ for all Σ-interpretations J extending I_σ such that $J \models_{OEL} T'$;
- The inductive cases for $\wedge, \vee, \neg, \exists, \forall$ are defined as usual.

The unique *belief system* $Mod(\mathcal{T})$ of an OEL theory $\langle \mathcal{T}, \leq \rangle$ with $\mathcal{T} = \{T_1, \ldots, T_n\}$ is the set $\mathcal{M}$ of all interpretations I extending I_σ for which $I \models_{OEL} T_i$ for every $T_i \in \mathcal{T}$.

For a $\leq$-minimal theory T, the absence of modal operators means that $\models_{OEL}$ reduces to classical $\models$.

For simplicity, the rest of this paper considers only Herbrand interpretations of the vocabulary Σ. This effectively means that all function and constant symbols are *rigid*: they have the same interpretation in all possible worlds. Moreover, because our complexity results need a *finite* domain, we will also assume that there are no function symbols and only a finite number of constants. However, apart from the complexity results and transformations to FO, our definitions and results can be generalized to interpretations extending any given pre-interpretation. Also, in what follows, we assume without loss of generality that modal operators are only applied to FO formulas. Indeed, nestings of modal operators can be removed by a simple generalization of such transformations for S_5[2].

Example 1. Consider the following scenario from (Gelfond and Lifschitz 1991). Assume a database with complete knowledge about the grade point average (GPA) of students and partial knowledge about whether they belong to a minority group. It can be represented in FO as:

$$DB = \left\{ \begin{array}{l} \forall x (\mathit{FairGPA}(x) \Leftrightarrow x = \mathit{Bob} \vee x = \mathit{Ann}). \\ \forall x (\mathit{HighGPA}(x) \Leftrightarrow x = \mathit{John}). \\ \mathit{Minority}(\mathit{Ann}). \end{array} \right\}$$

Now suppose the department decides that they want to interview students who may be, but are not necessarily, eligible for a grant, where eligible students are those with either a high GPA or a fair GPA and belonging to a minority. We now build on top of the database DB a second layer $T_I > DB$:

$$T_I = \left\{ \begin{array}{l} \forall x\ \mathit{Interview}(x) \Leftrightarrow \\ \qquad \mathit{FairGPA}(x) \wedge \neg K_{DB}(\mathit{Minority}(x)) \\ \qquad \wedge \neg K_{DB}(\neg \mathit{Minority}(x)). \end{array} \right\}$$

The theories T_{DB} and T_I both have several models (in which *Minority* (*Bob*) and *Minority* (*John*) can both be true and false). However, in every interpretation of $Mod(\mathcal{T})$, *Interview* (*Bob*) is true.

We now illustrate how the *closed world assumption* (Reiter 1977) can be represented in OEL. As an alternative to the database DB, we can decide to store in the database DB' only the positive facts about the relations for which we have complete information. We can then add an extra layer of expertise, adding the extra knowledge *on top* of DB', that if the DB' *does not say* that a student *has a particular GPA*, we conclude that he/she does not have that GPA. This extra layer T_{cwa} contains an ASP-style CWA:

$$DB' = \left\{ \begin{array}{l} \mathit{FairGPA}(\mathit{Ann}) \wedge \mathit{FairGPA}(\mathit{Bob}) \\ \qquad \wedge \mathit{HighGPA}(\mathit{John}) \wedge \mathit{Minority}(\mathit{Ann}) \end{array} \right\}$$

$$T_{cwa} = \left\{ \begin{array}{l} \forall x (\neg \mathit{FairGPA}(x) \Leftrightarrow \neg K_{DB'} \mathit{FairGPA}(x)). \\ \forall x (\neg \mathit{HighGPA}(x) \Leftrightarrow \neg K_{DB'} \mathit{HighGPA}(x)). \end{array} \right\}$$

Now in all $M \in Mod(\mathcal{T})$, where $\mathcal{T}$ is the OEL theory $\mathcal{T} = DB' \leq T_{cwa}$ it holds that *FairGPA* (*John*) is false. We could now add another layer of expertise to the expertise of DB' and T_{cwa}, for example, again the theory T_I expressing when an interview should happen.

As the next lemma states, each OEL theory is equivalent to an FO theory. However, this FO theory depends crucially on the given Herbrand domain, because for each formula $K_T\varphi(\bar{x})$ *and* each tuple of constants $\bar{c}$, it will explicitly add either a literal $P_{T,\varphi}(\bar{c})$ or a literal $\neg P_{T,\varphi}(\bar{c})$ to the theory (depending on whether T knows φ or not). As a consequence, the transformation leads to a polynomial increase in the size of the theory and, as we will show below, assuming a finite domain, computing it is an NP-hard problem.

[2]To see this, note that $K_{T_1}((\neg)K_{T_2}(\varphi))$ is equivalent with $(\neg)K_{T_2}(\varphi)$, $K_{T_1}(\varphi_1 \vee K_{T_2}(\varphi_2))$ with $K_{T_1}(\varphi_1) \vee K_{T_2}(\varphi_2)$ and $K_{T_1}(\varphi_1 \wedge K_{T_2}(\varphi_2))$ with $K_{T_1}(\varphi_1) \wedge K_{T_2}(\varphi_2)$.

Lemma 3. *For a given vocabulary Σ (and thus also a given Herbrand universe) and an OEL theory $\mathcal{T}$, there exists an FO theory $FO(\mathcal{T})$ over an extended vocabulary $\Sigma' \supseteq \Sigma$, such that for every Σ'-Herbrand interpretation M it holds that $M \models FO(\mathcal{T})$ iff $M|_\Sigma \in Mod(\mathcal{T})$.*

Proof. This follows immediately from Proposition 10 and Lemma 11 further in this paper, where we prove this lemma in a more general setting. $\square$

While a number of auto-epistemic features can be represented in OEL, OEL is in general less expressive than AEL, for example, one cannot express self-referential statements. The advantage this gives OEL is that the semantics are simpler, since we always have exactly one belief system (In contrast, a theory in AEL might have many or no stable expansions), and, as we show further on, OEL has a lower complexity than AEL. However, despite these differences, in some useful cases we can still view an ordered epistemic theory as an autoepistemic theory.

For a given OEL theory $\mathcal{T}$, define $AEL(\mathcal{T})$ as the modal theory obtained by taking the union of all $T \in \mathcal{T}$ and turning each occurrence of a modal operator K_T into the same operator K. Intuitively, an operator K_T expresses the knowledge of the source T and all sources lower in the hierarchy, while the operator K can be seen as a *distributed knowledge* operator, expressing the combined knowledge of *all sources*. This operation does not always preserve equivalence. E.g., let $\mathcal{T} = \{T_1, T_2\}$, $T_1 < T_2$, and $T_1 = \{\}, T_2 = \{Q, \neg K_{T_1}(Q) \Rightarrow R\}$. In this case, $\{Q, R\}$ is the only possible world of $\mathcal{T}$, while the unique stable expansion of $AEL(\mathcal{T})$ contains a second possible world $\{Q\}$. This difference is caused by the fact that T_2 itself possesses additional knowledge about the atom Q that occurs in its query to T_1. Therefore, while the system as a whole knows that Q is true, the theory T_1 in isolation does not. When all knowledge about a particular atom is contained within a single theory T_i, this cannot be the case and we would therefore expect $\mathcal{T}$ and $AEL(\mathcal{T})$ to be equivalent. However, in the translation to AEL, there is a subtle case where one theory T_i might still influence the interpretation of atoms that were described in a different theory T_j, and that is when T_i introduces an inconsistency that eliminates all models of T_j. To rule out such cases, we use the following concept.

Definition 4. An autoepistemic theory T is *permaconsistent* if every theory T' that can be constructed from T by independently replacing all non-nested occurrences of modal literals by $\mathbf{t}$ or $\mathbf{f}$ is consistent.

Proposition 5. *Given an OEL theory $\mathcal{T}$. If (1) each predicate symbol $P \in \Sigma$ has objective occurrences in at most one sub-theory $T \in \mathcal{T}$, (2) P appears only in modal literals $K_{T'}\varphi$ such that $T < T'$ and (3) $AEL(\mathcal{T})$ is a permaconsistent theory, then $AEL(\mathcal{T})$ has a unique autoepistemic expansion B equal to the unique belief system $Mod(\mathcal{T})$ of $\mathcal{T}$.*

Proof. It is straightforward to prove this proposition using the stratification results for AEL from (Vennekens, Gilis, and Denecker 2006). $\square$

While permaconsistency may seem like a strong criterion, it is often met in practice. For instance, the entire subset of AEL that corresponds to ASP satisfies it.

While the theory (DB, T_I) from Example 1 also leads to a permaconsistent AEL theory, it fails to satisfy the other condition of Prop. 5, since both DB and T_I contain an objective occurrence of $FairGPA$. However, only DB actually provides information about $FairGPA$, because T_I does not pose any constraints on the interpretation of this predicate. Indeed, each model of the database DB can be extended with an interpretation of $Interview$ that satisfies the theory T_I. As the next proposition shows, this conditions suffices to ensure that the models of the OEL theory (DB, T_I) are also models of $AEL(DB, T_I)$.

Proposition 6. *Given an OEL theory $\mathcal{T}$. If for each $T \in \mathcal{T}$, every $\Sigma_{\leq T}$ model of $\mathcal{T}_{\leq T}$ can be extended to a model of $\mathcal{T}$ and for every literal $K_T\varphi$ in $\mathcal{T}$, φ contains only symbols of Σ_T, then the belief system $Mod(\mathcal{T})$ is an expansion of the theory $AEL(\mathcal{T})$.*

Proof. For simplicity, we assume that $\mathcal{T}$ is a propositional theory, that each $T_i \in \mathcal{T}$ has at most one occurrence of a modal operator K_{T_j}, and if so, that $j = i - 1$. The proof easily extends to the general case. If T is a theory containing $K_{T'}\varphi$ and $\mathcal{M}$ a set of interpretations, we denote by $T\langle\mathcal{M}\rangle$ the theory derived from T by evaluating $K_{T'}\varphi$ in $\mathcal{M}$, i.e., replacing each modal literal $K_{T'}\varphi$ by $\mathbf{t}$ if φ holds in all interpretations $I \in \mathcal{M}$ and by $\mathbf{f}$ otherwise. Also, for a FO theory T, we denote with $mod(T)$ the set of all models M of T.

We denote with $\mathcal{M}_i$ a set of $\Sigma_{\leq T_i}$ interpretations. It follows directly from the semantics of OEL that the modal operators $K_{T_{i-1}}$ that appear in a theory T_i are evaluated in the belief system $\mathcal{M}_{i-1}$ of the theory $\mathcal{T}_{\leq i-1}$. Therefore it holds that a set $\mathcal{M}_i$ is an OEL belief system of $\mathcal{T}_{\leq T_i}$ iff

$$\mathcal{M}_i = mod(T_0 \cup \bigcup_{1 \leq j \leq i} (T_j\langle\mathcal{M}_{j-1}\rangle)),$$

where for every $j < i$, $\mathcal{M}_j$ is the belief system of $\mathcal{T}_{\leq T_j}$. It now holds for every j such that $1 \leq j \leq i$ that $T_j\langle\mathcal{M}_{j-1}\rangle = T_j\langle\mathcal{M}_i\rangle$. Indeed, it holds that $\mathcal{M}_i|_{\leq\Sigma_{j-1}} \subseteq \mathcal{M}_{j-1}$, since each model of T_i is by definition also a model of each $T_j \leq T_i$. But also $\mathcal{M}_{j-1} \subseteq \mathcal{M}_i|_{\leq\Sigma_{j-1}}$, because every model $m \in \mathcal{M}_{j-1}$ is extendible to a model $m' \in \mathcal{M}_i$. Therefore,

$$\mathcal{M}_i = mod(T_0 \cup \bigcup_{1 \leq j \leq i} (T_j\langle\mathcal{M}_i\rangle)),$$

and thus that $\mathcal{M}_i$ is a fixpoint of the $D_{AEL(\mathcal{T}_{\leq T_i})}$ operator. We can conclude that if $\mathcal{M}$ is the belief system of a OEL theory $\mathcal{T}$, then $\mathcal{M}$ is indeed a stable expansion of $AEL(\mathcal{T})$. $\square$

The other direction of the proof does not hold because AEL theories can have self-supporting expansions, as is illustrated by the following example.

$$T_1 : \{K_{T_0}(R) \Rightarrow R.\}$$
$$T_0 : \{R \vee \neg R.\}$$

The OEL theory $\mathcal{T} = T_0, T_1$ has as its unique belief set $Mod(\mathcal{T})$ the set $\{\{\}, \{R\}\}$. This set is also an expansion. However, the autoepistemic theory $AEL(\mathcal{T})$ has a second expansion, that is, the self-supporting expansion $\{\{R\}\}$. It has been argued that these self-supporting expansions are in fact anomalies, and several attempts have been made to define autoepistemic expansions that have a stronger notion of 'groundedness'. Konolige defined *moderately grounded expansions* (which are in fact maximal expansions) that eliminate some, but not all of these unwanted self-supporting expansions (Konolige 1988b). In the same paper he also defined *strongly grounded* expansions[3], which are more restrictive then moderately grounded expansions, but do have the undesirable property of depending on the syntactic representation of the theory (in the sense that two equivalent theories can give rise to different expansions). With the similar goal of avoiding such unsupported models, (Denecker, Marek, and Truszczynski 2003) introduced a stable and well-founded semantics for AEL.

Distributed Ordered Epistemic Logic

In this section we define a generalization of OEL, called *distributed ordered epistemic* logic (d-OEL), where higher levels no longer possess the knowledge of lower levels, as is the case in the autoepistemic setting of OEL. This generalization has a number of interesting theoretical applications, that is, the class of problems where there are distributed epistemic agents that can query each other. We again define the semantics by induction over the partial order $\leq$.

Definition 7. Let $\langle \mathcal{T}, \leq \rangle$ be an distributed ordered epistemic theory. For a Σ_T-interpretation I, the relation $I \models_{dist} T$ is defined as follows:

- For an atom $P(\bar{t})$, $I \models_{dist} P(\bar{t})$ iff $\bar{t}^I \in P^I$;
- For a modal literal $K_{T'}\varphi$, $I \models_{dist} K_{T'}\varphi$ iff $J \models \varphi$ for all $\Sigma_{T'}$-interpretations J such that $J \models_{dist} T'$;
- The inductive cases for $\wedge, \vee, \neg, \exists, \forall$ are defined as usual.

This semantics differs from that of OEL (Def. 2) by no longer requiring that a model of a theory $T \in \mathcal{T}$ has to satisfy every lower theory $T' < T$. As a consequence, this changes the way a modal operator K_T is evaluated. K_T now looks at all models of T, instead of only those models of T that are also models of the subtheories of T.

It follows directly from Def. 7 that, for each theory $T \in \mathcal{T}$, we can construct the set $\mathcal{M}_T$ of all d-OEL models of T (i.e., the set of interpretations M such that $M \models_{dist}$ T) by means of a bottum-up construction along the partial order $\leq$. That is, for the minimal theories $T_1^{min}, \ldots, T_n^{min} \in \mathcal{T}$, each $\mathcal{M}_{T_i^{min}}$ is just the set of classical models of T_i^{min}; and once we have constructed the sets of models $\mathcal{M}_{T'}$ for all $T' < T$, we can construct the models of T itself, by interpreting all modal formulas $K_{T'}\varphi$ according to $\mathcal{M}_{T'}$ and then constructing the classical models of the remaining objective theory. It follows from this construction process that each $T \in \mathcal{T}$ has a unique set of models. Note that this construction remains valid in case the formula φ in $K_{T'}\varphi$ contains another modal operator as long as the nested one refers to a theory lower than T' in the partial order.

Definition 8. The unique *belief system* $Mod(\mathcal{T})$ of a d-OEL theory $\langle \mathcal{T}, \leq \rangle$ with $\mathcal{T} = \{T_1, \ldots, T_n\}$ is the tuple $(\mathcal{M}_{T_1}, \ldots, \mathcal{M}_{T_n})$ such that for each i, $\mathcal{M}_{T_i}$ is the set of all Σ_{T_i}-Herbrand interpretations I for which $I \models_{dist} T_i$.

The next definition defines a syntactic transformation from an OEL theory to an d-OEL theory.

Definition 9. Given an OEL theory $\langle \mathcal{T}, \leq \rangle$ over the vocabulary Σ, we define the d-OEL theory $\langle \mathcal{T}^{\text{d-OEL}}, \leq \rangle$ as:

- For every $T \in \mathcal{T}$, $\mathcal{T}^{\text{d-OEL}}$ contains a corresponding element $T^{\text{d-OEL}} = T \cup \bigcup_{T' < T} T'$.
- $\mathcal{T}^{\text{d-OEL}}$ contains one maximal element $T_{max}^{\text{d-OEL}} = (T_1^{max})^{\text{d-OEL}} \cup \ldots \cup (T_n^{max})^{\text{d-OEL}}$, where $T_1^{max}, \ldots, T_n^{max}$ are the maximal elements of $\langle \mathcal{T}, \leq \rangle$.

The following proposition now states that an OEL theory $\langle \mathcal{T}, \leq \rangle$ is equivalent with the corresponding d-OEL theory $\langle \mathcal{T}^{\text{d-OEL}}, \leq \rangle$.

Proposition 10. *Given an OEL theory $\langle \mathcal{T}, \leq \rangle$ over the vocabulary Σ, for any Σ-interpretation I it holds that $I \models_{dist} T_{max}^{\text{d-OEL}}$ iff $I \models_{OEL} \mathcal{T}$.*

The proof of this proposition is straightforward. Indeed, it is obviously the case that when the higher-up theories syntactically include the lower ones, information is guaranteed to increase along the partial order $\leq$, that is, the semantics of OEL are simulated. As a consequence of this proposition, we can see OEL as a sublogic of d-OEL.

Example 2. To illustrate how d-OEL can be used to model the beliefs of different agents, we reconsider Example 1, but now assume we also have a second database, talking about which students are minorities:

$$DB_2 = \left\{ \begin{array}{l} \neg \mathit{Minority}(\mathit{Ann}). \\ \mathit{Minority}(\mathit{John}) \end{array} \right\}.$$

Clearly, both databases have contradictory beliefs about which students are minorities. We can now build another layer T_+ on top of these two databases that expresses how such contradictory data should be handled; for example, T_+ could express that when the two databases do not agree, the student should be interviewed:

$$T_+ = \left\{ \begin{array}{l} \forall x \; \mathit{Interview}(x) \Leftrightarrow \\ \qquad K_{DB}(\mathit{Minority}(x)) \wedge K_{DB_2}(\neg \mathit{Minority}(x)) \\ \quad \vee \; K_{DB}(\neg \mathit{Minority}(x)) \wedge K_{DB_2}(\mathit{Minority}(x)) \end{array} \right\}$$

The theory of interest here, T_+, has exactly one Herbrand model I, in which $Interview^I = \{Ann\}$.

Like OEL theories, a d-OEL theory $\mathcal{T}$ is equivalent to an FO theory as well. In the next lemma we assume without loss of generality that no predicate symbol occurs in two different vocabularies Σ_{T_i}[4].

[3]The original definition later proved incorrect and a correct version was independently given by (Konolige 1989) and (Marek and Truszczynski 1989).

[4]Indeed, all theories in a d-OEL theory $\mathcal{T}$ are independent modules that can only access each others knowledge through the modal operator. This means we can replace all occurrences of a predicate symbol P in T_i by a new predicate symbol P', as long as we also replace P by P' in every modal atom in $\mathcal{T}$.

Lemma 11. *For a given set of vocabularies $\Sigma_{T_1}, \ldots \Sigma_{T_n}$ and a d-OEL theory $\mathcal{T}$, there exists an FO theory $FO(\mathcal{T})$ over $\Sigma' \supseteq \Sigma_{T_1} \cup \ldots \cup \Sigma_{T_n}$ such that for every Σ'-Herbrand interpretation M it holds that $M \models FO(\mathcal{T})$ iff for every T_i, $M|_{\Sigma_{T_i}} \in \mathcal{M}_{T_i}$, where $(\mathcal{M}_{T_1}, \ldots, \mathcal{M}_{T_n})$ is $Mod(\mathcal{T})$.*

Proof. For each modal expression $K_T(\varphi[\bar{x}])$ occurring in $\mathcal{T}$, we introduce a predicate $P_{T,\varphi}$ with arity $\#(\bar{x})$ and for each P/n that occurs objectively in T, we introduce a predicate P_T/n. For each formula φ, let $FO_T(\varphi)$ denote the formula obtained from φ by substituting atoms $P_{T',\varphi}(\bar{x})$ for modal literals $K_{T'}(\varphi[\bar{x}])$ and $P_T(\bar{t})$ for objective occurrences of atoms $P(\bar{t})$. Our theory $FO(\mathcal{T})$ consists of:

- all formulas $FO_T(\varphi)$ for which $T \in \mathcal{T}$ and $\varphi \in T$;
- for each modal literal $K_T\varphi[\bar{x}]$ in $\mathcal{T}$ and each tuple $\bar{t}$ of terms (i.e., elements of the Herbrand universe) either the literal $P_{T,\varphi}(\bar{t})$ if $T \models \varphi[\bar{t}]$, or $\neg P_{T,\varphi}(\bar{t})$ otherwise.

Clearly, since a minimal theory T_i^{min} contains no modal literals, we have that for any Herbrand interpretation M, it holds that $M \in mod(FO(T_i^{min}))$ iff $M|_{\Sigma_{T_i^{min}}} \models_{dist} T_i^{min}$. It is easy to show by induction on the order of theories that this holds for any $T \in \mathcal{T}$. Note that this proof stays valid if not all Σ_{T_i} are disjunct. □

The method we use here to transform a d-OEL theory into an equivalent FO theory is similar to the RES procedure of (Levesque and Lakemeyer 2000), which transforms a modal query w.r.t. a first order knowledge base into a standard first order query.

Problem Solving

This section studies the inference tasks we can do for OEL and d-OEL. First, we investigate data complexity of model checking, satisfiability checking and query answering. To measure data complexity, we consider inference in a class of theories of the form $\mathcal{T} = \{DB, T_1, \ldots, T_n\}$ over finite vocabularies Σ without function symbols in which $T_1, \ldots, T_n$ are fixed and only DB and the set of constants in Σ vary. Moreover, we assume that DB is a database theory, i.e., a set of ground literals, and that it is minimal in the theory order $\leq$. We denote the class of such theories as $\mathcal{DB}$.

Proposition 12. *The decision problems of model checking, satisfiability checking and query answering in the class of d-OEL theories $\mathcal{DB}$ are in Δ_2^P. There are theories for which these tasks are both NP-hard and co-NP-hard. The same holds for the class of OEL theories $\mathcal{DB}$.*

Proof. Assume $\mathcal{T}$ is a d-OEL-theory. Consider $FO(\mathcal{T})$. Since DB contains no modal literals, $FO(\mathcal{T})$ consists of DB, the set of literals describing the predicates $P_{T,\varphi}$ and the fixed theory $FO_{T_1}(T_1) \cup \cdots \cup FO_{T_n}(T_n)$.

First, we show that this theory can be constructed using a polynomial number of calls to an NP-oracle. The construction proceeds along the order $\leq$ on the theories, starting from $FO(\mathcal{T}_{\leq DB}) = DB$ as a base case. The size of $FO(\mathcal{T}_{\leq DB})$ is trivially linear in the size of DB.Assume we have already constructed $FO(\mathcal{T}_{\leq T_i})$ and that it is polynomial in the size of DB. We can then construct $FO(\mathcal{T}_{T_{i+1}})$ by first adding all formulas $FO_{T_{i+1}}(\varphi)$ to $FO(\mathcal{T}_{\leq T_i})$, and then adding for every modal literal $K_{T_j}\varphi$ of T_i and tuple $\bar{t}$ of terms, either $P_{T_j,\varphi}(\bar{t})$ or $\neg P_{T_j,\varphi}(\bar{t})$. To decide which of these two to add, we need to know whether $T_j \models \varphi(\bar{t})$. Lemma 11 states that this is the case iff the theory $FO(\mathcal{T}_{\leq T_j}) \wedge \neg FO_{T_j}(\varphi(\bar{t}))$ does not have a Herbrand model. Because the size of $FO(\mathcal{T}_{\leq T_j})$ is polynomial in DB (and the size of $\varphi(\bar{t})$ is constant), checking whether a given Herbrand interpretation is a model of this theory can be done in polynomial time. Therefore, an NP oracle will be able to tell us which of the two literals to add. Because we add one literal for each tuple $\bar{t}$ of terms, the size of the result is still polynomial in the size of DB.

Each of the three tasks can be easily implemented using the theory $FO(\mathcal{T})$. Because, in Herbrand interpretations, model checking for FO is polynomial in the size of the domain, satisfiability checking is NP and query answering is co-NP, we obtain that all three tasks are at most Δ_2^P for d-OEL. It follows from the proof of Prop. 10 that constructing a d-OEL theory equivalent with a given OEL theory $\mathcal{T}$ can be done in linear time (in the size of the Herbrand universe), therefore the same complexity results hold for OEL theories.

As for the second statement of the proposition, it suffices to focus on the least complex of the three problems, i.e., model checking. We prove the hardness properties by encoding the NP-complete graph kernel problem as a model checking problem of d-OEL. Consider the parameterized theory $\mathcal{T}$ consisting of $DB < T$ where DB encodes a graph G and consists of, for each pair A, B of domain elements, either the atom $G(A, B)$ or the negative literal $\neg G(A, B)$. This is a complete description of the graph G on the Herbrand universe. Σ^P contains G and two other predicates $Kernel/1$ and P which do not appear in DB.

Next we define T. T specifies that the new propositional symbol P is true iff DB knows that the set represented by $Kernel$ is not a graph kernel of G. A graph kernel is a set of vertices such that no kernel elements are directly connected and each non-kernel element is connected to a kernel element. I.e., T is singleton containing the formula

$$P \Leftrightarrow K_{DB}\neg\varphi$$

where φ expresses that *Kernel* is a kernel of G:

$$\forall x \forall y(\mathit{Kernel}(x) \wedge \mathit{Kernel}(y) \Rightarrow \neg G(x, y)) \wedge \\ \forall x(\neg \mathit{Kernel}(x) \Rightarrow \exists y(\mathit{Kernel}(y) \wedge G(y, x)))$$

It is easy to see that since DB contains no information about *Kernel*, the only way in which DB can know that *Kernel* is not a kernel of G is if it knows that G has no kernels. Hence, deciding whether $K_{DB}\neg\varphi$ holds, boils down to solving a co-NP hard problem. Thus, checking whether an interpretation in which P is true is a model is a co-NP-complete problem. Checking whether an interpretation in which P is false is a model is an NP-hard problem. This concludes the proof. □

As shown by Niemelä 1992 and Gottlob 1992, the problem of deciding whether an autoepistemic theory has a stable expansion is Σ_2^P-complete. The result in Proposition 12 thus asserts our earlier claim that OEL is of a lower complexity then AEL (the inclusion $\Delta_2^P \subseteq \Sigma_2^P$ is believed to be strict).

The proof of this proposition also shows that inference tasks in d-OEL can be implemented using FO tools. For instance, a d-OEL model generator could be implemented on top of any FO model generator, such as (Wittocx, Mariën, and Denecker 2008; Answer ; Claessen and Sörensson 2003; Torlak and Jackson 2007), by using the layered approach described in the proof above: first we use a polynomial number of calls to such a model generator to construct $FO(\mathcal{T})$ and then we call the model generator again to find out whether FO($\mathcal{T}$) has a Herbrand model, from which we then can extract the model of $\mathcal{T}$. Such a model generator could be used to solve the following practical problem.

Example 3. Imagine that every week a hospital needs to distribute patients over its available rooms. For this, it uses a knowledge base that expresses what a valid assignment is. Possible constraints might be that assigned rooms have to be available, that the maximum capacity of the rooms cannot be exceeded, that patients with a specific treatment need rooms with appropriate facilities, etc. In FO(·), this would become:

$$T_s = \left\{ \begin{array}{l} (\mathit{Used}\,(r,d) \Leftrightarrow \exists p\, \mathit{RoomOfPat}\,(r,p,d)). \\ \forall p\, r\, d\, (\mathit{Used}\,(r,d) \Rightarrow \mathit{Avail}\,(r,d)). \\ \forall d\, r\, (\#\{p : \mathit{RoomOfPat}\,(r,p,d)\} \leq \mathit{Capac}\,(r)). \\ \forall d\, p\, r\, (\mathit{RoomOfPat}\,(r,p,d) \wedge \mathit{ReqFacil}\,(p,f,d) \\ \qquad\qquad \Rightarrow \mathit{HasFacil}\,(r,f)). \end{array} \right\}$$

The first sentence of this theory, for example, defines that a room is used on a particular day, if on that day, the room is assigned to a certain patient. The other sentences express constraints on how the rooms are used, e.g., the second sentence says that if a room is used on particular day, it has to be available. Assume that a room is unavailable if it is being cleaned, and the cleaning company provides a complete database T_{cl} about when which room is available. A model generator can then be used on $T_s \cup T_{cl}$, together with another complete data base DB about the capacity, etc., to generate a valid schedule.

Now, assume that the hospital is refurbishing its rooms over an extended period of time, and regularly some rooms might not be available. The company that executes the refurbishments has its own knowledge base T_{ref} expressing when rooms will be available. However, unlike the cleaning company, this company can no longer provide *complete* information about the availability. Instead, it might say:

$$T_{ref} = \left\{ \begin{array}{l} \mathit{Avail}\,(\mathit{R1}, \mathit{Mo}) \wedge \neg \mathit{Avail}\,(\mathit{R3}, \mathit{Tue}). \\ \mathit{Avail}\,(\mathit{R2}, \mathit{Fri}) \wedge \forall r\, \neg \mathit{Avail}\,(r, \mathit{Thu}). \end{array} \right\}$$

The knowledge base T_{ref} indeed contains no information about, e.g., the availability of R_1 on Tuesday. Again, the hospital would like to get a valid schedule. However, this time, we cannot simply give $T_s \cup T_{ref} \cup DB$ to a model generator, since that would try to fill in the missing information about the availability, leading to many schedules: one where R_1 is available on Tuesday, one where R_1 is not available on Tuesday, and so on. The problem here is that we want to use T_s and T_{ref} in different ways: we want to do model generation on T_s, but not on T_{ref}.

We can achieve this by means of OEL's epistemic operator $K_{T_{ref}}$. Let T'_s be T_s together with the sentence:

$$\forall r \forall d (\mathit{Avail}\,(r,d) \Leftrightarrow K_{T_{ref}}(\mathit{Avail}\,(r,d))).$$

The interpretation of predicate *RoomOfPat* in the models of the OEL theory $\mathcal{T} = \{T'_s, T_{ref}\}$ now represents a *safe* schedule, that is, only rooms of which it is *certain* that they are available (according to T_{ref}) are used in the schedule. Alternatively, the hospital might decide that it is allowed to use a maximum of two 'uncertain' rooms. Such a schedule is represented by the interpretation of *RoomOfPat* in the models of $\mathcal{T}' = \{T''_s, T_{ref}\}$, where T''_s is the theory T_s with the rule $\forall p\, r\, d\, (\mathit{Used}\,(r,d) \Rightarrow \mathit{Avail}\,(r,d))$ replaced by the following rule:

$$\begin{array}{l} \forall d\, (2 \geq \#\{r : \mathit{Used}\,(r,d) \wedge \neg K_{T_{ref}} \neg \mathit{Avail}\,(r,d) \wedge \\ \qquad \neg K_{T_{ref}} \mathit{Avail}\,(r,d)\}). \end{array}$$

Note that for the theories T and T' the OEL and d-OEL semantics coincide (in the sense of Proposition 10). However, suppose that the hospital wants to use both the theory T_{cl} from the cleaning company and the theory T_{ref} from the refurbishing company. Since these two theories come from different companies that know nothing about each other, they are most likely inconsistent with each other. This means that any OEL theory that contains both of them will also be inconsistent. In d-OEL, this is not a problem and the hospital can just add a formula:

$$\begin{array}{r} \forall r \forall d (\mathit{Avail}\,(r,d) \Leftrightarrow K_{T_{ref}}(\mathit{Avail}\,(r,d))) \\ \wedge K_{T_{cl}}(\mathit{Avail}\,(r,d)). \end{array}$$

to T_s (we denote this theory by T'''_s). The interpretation of *RoomOfPatient* in the models of T'''_s of the d-OEL theory $\mathcal{T}'' = \{T'''_s, T_{cl}, T_{ref}\}$ again represents such a schedule.

In summary, this section showed that model checking, model generation and query answering for both d-OEL and OEL are in Δ_2^P. We explained how a prototype model generator can be implemented on top of an FO model generator, and illustrated, by means of the above example, that a class of interesting constraint problems (in particular, model generation problems that need to query a possibly incomplete knowledge base), can be formulated as model generation problems for d-OEL or OEL.

Related work

Answer Set Programming

This section takes a closer look at the relation between OEL and Answer Set Programming, more specifically its core language called Extended Logic Programming (Gelfond and Lifschitz 1991). We first briefly recall the basics. An answer set program P over vocabulary Σ is a collection of rules:

$$\mathtt{l}'_0 \mathtt{v} \ldots \mathtt{v} \mathtt{l}'_\mathtt{k} \leftarrow \mathtt{l}_1, \ldots, \mathtt{l}_\mathtt{m}, \mathtt{not}\ \mathtt{l}_{\mathtt{m}+1}, \ldots, \mathtt{not}\ \mathtt{l}_\mathtt{n}.$$

where each $\mathtt{l}'_\mathtt{j}, \mathtt{l}_\mathtt{i}$ is an atom $\mathtt{P}(\mathtt{t}_1, \ldots, \mathtt{t}_\mathtt{k})$ or a strong negation literal $\neg\mathtt{P}(\mathtt{t}_1, \ldots, \mathtt{t}_\mathtt{k})$. A program with variables represents the set of all ground rules obtained by substituting ground terms of Σ for all variables. We call an answer set program *disjunctive* if $k > 1$ for at least one rule.

An answer set A of P is a $\subseteq$-minimal element in the collection of all sets S of ground atoms and strong negation literals, such that, for each rule

$$\mathtt{l}'_1 \mathtt{v} \ldots \mathtt{v}\, \mathtt{l}'_\mathtt{k} \leftarrow \mathtt{l}_1, \ldots, \mathtt{l}_\mathtt{m}, \mathtt{not}\ \mathtt{l}_{\mathtt{m}+1}, \ldots, \mathtt{not}\ \mathtt{l}_\mathtt{n}.,$$

if $l_1, \dots, l_m \in S$ and $l_{m+1}, \dots l_n \notin S$, then at least one $l'_i \in S$. Intuitively, an answer set program represents the knowledge of an introspective epistemic agent: a rule of the above form states that the agent must believe one of the $\mathtt{l}'_\mathtt{k}$ if he also believes $\mathtt{l}_1, \dots, \mathtt{l}_\mathtt{m}$ and does not know any of the literals $\mathtt{l}_{\mathtt{m}+1}, \dots, \mathtt{l}_\mathtt{n}$. Moreover, the agent does not believe anything he is not forced to believe. Each answer set represents a possible state of belief of the agent.

Let us now return to Example 2. The incomplete database DB' can be represented in ASP as follows:

$$DB' = \left\{ \begin{array}{ll} \mathtt{FairGPA(Ann).} & \mathtt{Minority(Ann).} \\ \mathtt{HighGPA(John).} & \mathtt{FairGPA(Bob).} \end{array} \right\}$$

The partial completeness of this database can be expressed by two CWA rules:

$$CWA = \left\{ \begin{array}{l} \neg\mathtt{FairGPA(x)} \leftarrow \mathtt{not\ FairGPA(x)}. \\ \neg\mathtt{HighGPA(x)} \leftarrow \mathtt{not\ HighGPA(x)}. \end{array} \right\}$$

The next rule expresses who is to be interviewed:

$$IV = \{\mathtt{Interview(x)} \leftarrow \mathtt{FairGPA(x)}, \mathtt{not\ Minority(x)}, \\ \mathtt{not}\ \neg\mathtt{Minority(x)}.\}$$

The program $DB' \cup CWA \cup IV$ has one answer set, in which only Bob is to be interviewed. Both CWA and IV exploit the epistemic nature of negation-as-failure: the former to assert that unknown facts are false, and the latter to check whether the minority status of x is unknown. This solution is not entirely equivalent to ours in Example 1. Indeed, this answer set does not contain literals $\neg Interview(\dots)$. Hence, whereas in our solution, Bob and $John$ are known not to be interviewed, in the belief state corresponding to this answer set, this is unknown. To derive these conclusions, rules need to be added that correspond to the only-if part of the interview rule:

$$\neg\mathtt{Interview(x)} \leftarrow \neg\mathtt{FairGPA(x)}.$$
$$\neg\mathtt{Interview(x)} \leftarrow \mathtt{Minority(x)}.$$
$$\neg\mathtt{Interview(x)} \leftarrow \neg\mathtt{Minority(x)}.$$

Gelfond (1991) pointed to the following problem in the context of disjunctive answer set programs. He considered a database with the disjunctive information that either Ann or Bob belongs to a minority:

$$DB_\vee = \left\{ \begin{array}{c} \mathtt{FairGPA(Ann).\,FairGPA(Bob).} \\ \mathtt{Minority(Ann)\ v\ Minority(Bob).} \end{array} \right\}$$

Again, we would like to infer that both Ann and Bob must be interviewed, but $DB_\vee \cup IV$ will give two answer sets: one with Interview(Ann) and one with Interview(Bob). If we add the only-if rules, the first answer set will contain ¬Interview(Bob) and the second ¬Interview(Ann). To solve this problem, Gelfond proposed the logic ASP^K, which adds an epistemic operator K, called the *strong introspection operator*, to ASP. This operator queries whether a literal is present in *all* answer sets. It can correctly express the definition of Interview as:

$$\mathtt{Interview(x)} \leftarrow \mathtt{FairGPA(x)}, \\ \mathtt{not\ K\ not\ Minority(x)}, \mathtt{not\ K\ Minority(x)}.$$

This rule gives the right result for both $DB_\vee$ and DB'.

In OEL, the disjunctive database can be represented by:

$$DB_\vee^{OEL} = \left\{ \begin{array}{l} FairGPA(Ann).\ FairGPA(Bob.) \\ Minority(Ann) \vee Minority(Bob). \end{array} \right\}$$

This $DB_\vee^{OEL}$ can be queried by the same theory T_I as in Example 1 and this will result in the correct conclusion that both Ann and Bob have to be interviewed.

Unlike OEL, theories in ASP^K may have more than one "world view" ($\sim$ belief system), and as recently shown by Gelfond (2011), some of these world views are unintended. While that paper also suggests a change to the semantics of ASP^K, it is currently not known whether this solves all issues.

In comparison to the OEL solution, the ASP solution without strong introspection is less *elaboration tolerant* in the sense that adding a disjunction results in error and the ASP solution with strong introspection requires a combination of two epistemic operators which seems to be overly complex for the situation at hand. Thus, for this sort of epistemic applications, OEL seem to offer a good solution.

Default Logic

We already showed that OEL is able to represent a certain kind of default information, namely the Closed World Assumption. Other defaults are of course also possible. For instance, the following theory $T_{bird} < T_{def}$ models one of the standard examples:

$$T_{bird} = \left\{ \begin{array}{c} Bird(Tweety) \wedge Bird(Clyde). \\ Penguin(Clyde). \\ \forall x (Penguin(x) \Rightarrow \neg Flies(x)). \end{array} \right\}$$

$$T_{def} = \left\{ \begin{array}{l} \forall x (Bird(x) \wedge \neg K_{T_{bird}}(\neg Flies(x)) \\ \hfill \Rightarrow Flies(x)). \end{array} \right\}$$

However, the question is what happens in cases with multiple default extensions. For instance, let us apply the above method to model Nixon's diamond:

$$T_{base} = \left\{ \begin{array}{c} Republican(Nixon) \wedge Quaker(Nixon) \\ \forall x (\neg Dove(x) \vee \neg Hawk(x)) \end{array} \right\}$$

$$T_{def} = \left\{ \begin{array}{l} \forall x (Republican(x) \wedge \neg K_{T_{base}}(\neg Hawk(x)) \\ \hfill \Rightarrow Hawk(x)) \\ \forall x (Quaker(x) \wedge \neg K_{T_{base}}(\neg Dove(x)) \\ \hfill \Rightarrow Dove(x)) \end{array} \right\}$$

This OEL theory entails that Nixon is dove and hawk and hence it is inconsistent. In contrast, the solution in default logic has two extensions. In some cases the latter may be the desired behaviour. However, as argued in (Denecker, Marek, and Truszczynski 2011), multiple extensions of a default theory are caused by conflicting defaults that offer insufficient guidance how to solve the conflict, e.g., what happens for republican quakers? Such conflicts may arise inadvertently for the programmer in which case it is often better to resolve these conflicts by refining the condition of one or both constraints, e.g., by expressing that in case of republican quakers, being a dove is the default. In this case the inconsistency reported by the OEL representation is a useful sign that user intervention is required.

When multiple extension are really desired, it is here too the case, as for AEL, that OEL's hierarchical structure is unsuitable. For such cases, a solution might be searched in the direction of a non-hierarchical (or, alternatively, a self-referential) theory:

$$T_{base} = \left\{ \begin{array}{c} Republican(Nixon) \wedge Quaker(Nixon) \\ \forall x(\neg Dove(x) \vee \neg Hawk(x)) \end{array} \right\}$$

$$T_{hawk} = \left\{ \begin{array}{l} \forall x(Republican(x) \\ \qquad \wedge \neg K_{T_{dove}}(\neg Hawk(x)) \\ \qquad \Rightarrow Hawk(x)) \end{array} \right\}$$

$$T_{dove} = \left\{ \begin{array}{l} \forall x(Quaker(x) \wedge \neg K_{T_{hawk}}(\neg Dove(x)) \\ \hfill \Rightarrow Dove(x)) \end{array} \right\}$$

While it would in principle be possible to define an equilibrium semantics for OEL that would allow cyclic dependencies of this kind (and produce the same two extensions as Default Logic does for this example), this would go against our goal of simplicity.

Other Hierarchical Approaches

Deciding wether an autoepistemic theory has expansions is Σ_2^P-complete in general. However, for certain subclasses, reasoning can be done more efficiently. A popular family of such subclasses are *stratified* theories, i.e., theories that can be divided into a number of levels, such that the expansions of the entire theory can be computed by combined the expansions of each level. This approach was widely studied in both AEL and Default Logic (Gelfond 1987; Marek and Truszczyński 1991; Cholewinski 1994; Vennekens, Gilis, and Denecker 2006). In many of these works, stratified theories are guaranteed to have a unique expansion. Though this might seem similar to our approach here, there are some important differences. In these stratified approaches the idea is to give *syntactical constraints*, such that theories that satisfy these constraints can be *automatically* split up in smaller theories that do not circularly depend on each other. In OEL on the other hand, the goal is not to automatically split up a theory, but it is the *person writing down the theory* himself that is responsible for dividing the theory into suitable sub-theories. Because of this, our theories do not need to satisfy restrictive syntactic conditions which allows us to do more than is possible in these stratified approaches, e.g., theories that contain *closed world assumptions* will typically not be stratified while, as we have illustrated with some examples, closed world assumptions can be expressed in OEL.

Other somewhat related work is that of *stratified belief bases*[5] (Brewka 1989), which actually is an approach to inconsistency handling. Information is represented by stratified belief bases, i.e., finite sets of (possibly inconsistent) formulas equipped with a total pre-order which represents the available preferences over the given beliefs. The idea is then to select a preferred consistent subbase (where theories lower in the order have preference over theories higher in the order) of the given stratified belief base. While this framework also uses hierarchical knowledge bases, its purpose is clearly very different. First of all, we cannot see a stratified belief base as an OEL theory as such, since clearly an inconsistent OEL theory has no models. However, we can simulate inference for *certain* stratified belief bases with OEL (and similarly with d-OEL), that is, for belief bases where the conflicts arise *between* sub-theories, and not within. This would work as follows. Given a stratified belief base $\{\Delta_1, \ldots, \Delta_n\}$, we construct a OEL theory $\mathcal{T} = \{T_1, \ldots, T_n\}$ as follows. Take $T_1 = \Delta_1$, and for each formula $\varphi \in \Delta_i$ with $i > 1$, we add the formula $\neg K_{T_{i-1}}(\neg\varphi) \Rightarrow \varphi$ to T_i. A formula φ is now *strongly provable* (Brewka 1989) in the stratified belief base if it holds that $M \models \varphi$ for *all* models M of $\mathcal{T}$ and *weakly provable* if $M \models \varphi$ for some model M of $\mathcal{T}$.

[5] sometimes also referred to as prioritized knowledge/belief bases.

More Related Work

This section considers some knowledge representation formalisms that are related to d-OEL, that is, formalisms for handling distributed (possibly contradictory) knowledge.

Polleres, Feier, and Harth (2006) attempt to incorporate *contextual queries* and *non-monotonic* reasoning techniques into current Semantic Web technologies, in particular, extensions of RDF or OWL with rule languages. Because of the open-world setting in the Semantic Web, only a limited form of negation-as-failure is possible, in which negation has an explicitly defined, finite scope. As the authors claim, clear definitions and formal semantics for such scoped negation are missing, and thus they propose a *logic programming* based framework for the combination of *rule bases* on the web. One of the key features in this work is that every fact or rule is associated to a *context* (the URI of the rule base). Literals can then be tagged with these URIs to form *scoped literals*. This allows queries such as "*Which movies are rated "Bad" by http://moviereviews.com?*" to be represented as:

$$\begin{array}{l} \mathtt{Answer(x)} \leftarrow \mathtt{Movie(X)}, \\ \qquad \mathtt{Rated(X, Bad)}@\text{“}\mathtt{http://moviereviews.com}\text{''}. \end{array}$$

Negation is only allowed to appear in these scoped literals.

Each scoped literal corresponds naturally to a modal literal $K_{<uri>}P(\bar{t})$ or $K_{<uri>}\neg P(\bar{t})$ of d-OEL. As shown in (Denecker et al. 2010), the principle for extending FO to OEL can be easily applied also to FO(ID), the extension of FO with a rule-based representation for inductive definitions (Denecker and Ternovska 2008). Applying this to d-OEL leads to a logic that generalizes the scoped literal approach by allowing arbitrary formulas both in the bodies of rules and in the scope of the modal operators. In this way, d-OEL can also handle queries for, e.g., disjunctive knowledge.

Scoped negation was also introduced to FLORA-2 (Kifer 2005), which is a query answering system that integrates F-Logic with a number of other formalisms. In FLORA-2, a module is a container or a concrete knowledge base. Negative queries can then be posed to a certain module. Another formalism that has a form of scoped negation is N3 (Berners-Lee, Dan Connolly, and Scharf 2005), which is a language for representing RDF rules on the Semantic Web.

The need for a semantically well-defined construct to query other sources of knowledge is not limited to the Semantic Web context, but also exists in other knowledge representation domains. For example, Barker (2009) defines a meta-model for access control policies, using a logic programming based rule language. Literals occurring in the body of rules may be defined in the local program, but also in a remotely located program identified by v. Such a literal is then denoted by "$L \hookleftarrow v$". Typically, this construct is used for querying the knowledge of another access control manager. The semantics of the rule language is based on Clark's completion (1978), where, for remotely defined literals $L \hookleftarrow v$, the union of v and the local program is considered. Again, OEL generalizes this language.

Our work is also related to attempts to combine Logic Programs with Descriptions Logics. A representative approach is the hybrid combination proposed by Pührer, Heymans, and Eiter (2010). Here, an ASP program is allowed to query an OWL knowledge base to find out what it does and does not know. Like OEL, this is also a layered approach, with the obvious difference that DL-programs allow only two components, where the top component must be written in ASP and the bottom component in OWL. An interesting feature of DL-programs is that they allow additional atoms to be temporarily added to the OWL knowledge base for the evaluation of a query. In OEL, this could be simulated by exploiting the ability to have more than two components: a single minimal theory would then consist of the OWL knowledge base; in the next layer, we would find a number of theories that each add different additional facts to the knowledge base; finally, a single maximal theory would then correspond to the ASP program, posing queries to the different theories in the middle layer.

In addition to such hybrid integrations, there also exist a number of formalisms that offer a full integration of Logic Programming and Description Logics. These typically abandon the idea of a layered structure altogether and resort to languages that are very similar to AEL. In particular, de Bruijn et al. (2007) present a formalism that is embedded into first-order AEL, while Motik and Rosati (2010) uses an embedding into the logic MKNF, which has been shown to be equivalent to AEL (Rosati 1997). Our approach can be seen as claiming a middle ground between these approaches and the hybrid ones: we do retain a layered structure, but use a single language in the different components.

Conclusions

In this paper we further developed *ordered epistemic logic*, recently introduced by Denecker et al. to handle a range of interesting epistemic examples, without the high complexity of autoepistemic logic. In this paper we proved this claim by showing that model generation for OEL is in Δ_2^P, and gave a number of examples that show that the closed world assumption, defaults, … can indeed be naturally represented in OEL. While OEL does not allow the self-references of AEL, we proved that in some cases, an OEL theory can be seen as an AEL theory. Next, we defined a generalization of OEL, which we called *distributed ordered epistemic logic*, in which a higher level theory in the hierarchy no longer includes the knowledge of lower level theories. This logic is well-suited to model distributed (and potentially inconsistent) knowledge of agents. Applications for such a logic exists in the Semantic Web, where ontologies are distributed over the web, and one ontology might need the knowledge of another. Several approaches have been proposed to do this, and we argued that d-OEL is a natural generalization of several of them.

A limitation of our formalism is that it assumes the same fixed domain for all theories. Allowing different theories to have different *finite* domains is not problematic, since this can already be simulated in our current logic[6]. Allowing each theory to have its own infinite domain on the other hand, would be a real extension of the logic, and it is not clear which of the results presented in this paper would carry over. Another interesting topic for future work would be to implement a model generator for OEL and d-OEL, along the lines sketched earlier in this paper. Given the relatively high complexity of this task, it might, however, also be interesting to investigate how and when approximation techniques for FO (e.g., those developed by Wittocx, Denecker, and Bruynooghe (2010)) can be used to approximately solve OEL and d-OEL queries in polynomial time. Again, the layered structure of OEL and d-OEL seems to make it ideally suited for such an approach.

Acknowledgements

This work is partially supported by IWT-Vlaanderen.

References

Answer set programming with propositional schemata. http://www.cs.uky.edu/ai/aspps/.

Baral, C. 2003. *Knowledge Representation, Reasoning and Declarative Problem Solving*. Cambridge University Press.

Barker, S. 2009. The next 700 access control models or a unifying meta-model? In *Proceedings of the 14th ACM symposium on Access control models and technologies*, SACMAT '09, 187–196. ACM.

Berners-Lee, T.; Dan Connolly, E. P.; and Scharf, Y. 2005. Experience with N3 rules. In *W3C Workshop on Rule Languages for Interoperability*.

Bouquet, P.; Giunchiglia, F.; Harmelen, F. V.; Serafini, L.; and Stuckenschmidt, H. 2003. C-OWL: Contextualizing ontologies. In *Journal Of Web Semantics*, 164–179. Springer Verlag.

[6]If we want each theory $T \in \mathcal{T}$ to have its own specific domain D_T, we can simulate this by taking the union $\cup_{T \in \mathcal{T}} D_T$ as a fixed domain throughout $\mathcal{T}$ and defining in each theory T a predicate $Domain_T$ that explicitly enumerates all the elements of D_T:

$$\forall x (Domain_T(x) \Leftrightarrow \bigvee_{c \in D_T} x = c)$$

Brewka, G.; Eiter, T.; and Truszczyński, M. 2011. Answer set programming at a glance. *CACM* 54(12):92–103.

Brewka, G. 1989. Preferred subtheories: An extended logical framework for default reasoning. In *IJCAI*, 1043–1048.

Chellas, B. 1980. *Modal Logic: An Introduction*. Cambridge University Press.

Cholewinski, P. 1994. Stratified default theories. In *CSL*, 456–470.

Claessen, K., and Sörensson, N. 2003. New techniques that improve MACE-style model finding. In *MODEL*.

Clark, K. L. 1978. Negation as failure. In *Logic and Data Bases*, 293–322. Plenum Press.

de Bruijn, J.; Pearce, D.; Polleres, A.; and AugustínValverde. 2007. Quantified equilibrium logic and hybrid rules. In *International Conference on Web Reasoning and Rule Systems (RR)*, 58–72.

Denecker, M., and Ternovska, E. 2008. A logic of nonmonotone inductive definitions. *ACM Transactions on Computational Logic (TOCL)* 9(2):Article 14.

Denecker, M.; Vennekens, J.; Vlaeminck, H.; Wittocx, J.; and Bruynooghe, M. 2010. Answer set programming's contributions to classical logic. An analysis of ASP methodology. In Balduccini, M., and Tran, S., eds., *MG-65: Symposium on Constructive Mathematics in Computer Science*.

Denecker, M.; Marek, V. W.; and Truszczynski, M. 2003. Uniform semantic treatment of default and autoepistemic logics. *Artif. Intell.* 143(1):79–122.

Denecker, M.; Marek, V.; and Truszczynski, M. 2011. Reiter's default logic is a logic of autoepistemic reasoning and a good one, too. In Brewka, G.; Marek, V.; and Truszczynski, M., eds., *Nonmonotonic Reasoning – Essays Celebrating Its 30th Anniversary*. College Publications. 111–144.

Gelfond, M., and Lifschitz, V. 1991. Classical negation in logic programs and disjunctive databases. *New Generation Computing* 9(3/4):365–386.

Gelfond, M. 1987. On Stratified Autoepistemic Theories. In *National Conference on Artificial Intelligence (AAAI'87)*, 207–211. Morgan Kaufman.

Gelfond, M. 1991. Strong introspection. In *AAAI*, 386–391.

Gelfond, M. 2011. New semantics for epistemic specifications. In Delgrande, J. P., and Faber, W., eds., *LPNMR*, volume 6645 of *Lecture Notes in Computer Science*, 260–265. Springer.

Gottlob, G. 1992. Complexity results for nonmonotonic logics. *J. Log. Comput.* 2(3):397–425.

Hughes, G. E., and Cresswell, M. J. 1996. *A New Introduction To Modal Logic*. Routledge.

Kifer, M. 2005. Nonmonotonic reasoning in FLORA-2. In Baral, C.; Greco, G.; Leone, N.; and Terracina, G., eds., *LPNMR*, volume 3662 of *LNCS*, 1–12. Springer.

Konolige, K. 1988a. Hierarchic autoepistemic theories for nonmonotonic reasoning. In *AAAI*, 439–443.

Konolige, K. 1988b. On the relation between default and autoepistemic logic. *Artificial Intelligence* 35:343–382.

Konolige, K. 1989. On the relation between autoepistem ic logic and circumscription. In *IJCAI*, 1213–1218.

Levesque, H. J., and Lakemeyer, G. 2000. *The logic of knowledge bases*. MIT Press.

Levesque, H. J. 1990. All I know: A study in autoepistemic logic. *Artif. Intell.* 42(2-3):263–309.

Marek, V. W., and Truszczynski, M. 1989. Relating autoepistemic and default logics. In *KR*, 276–288.

Marek, V. W., and Truszczyński, M. 1991. Autoepistemic logic. *Journal of the ACM* 38(3):588–619.

Moore, R. C. 1984. Possible-world semantics for autoepistemic logic. In *Proceedings of the Workshop on Non-Monotonic Reasoning*, 344–354. Reprinted in: M. Ginsberg, ed., *Readings on Nonmonotonic Reasoning*, pages 137–142, Morgan Kaufmann, 1990.

Motik, B., and Rosati, R. 2010. Reconciling description logics and rules. *Journal of the ACM (JACM)* 57.

Niemelä, I. 1992. On the decidability and complexity of autoepistemic reasoning. *Fundam. Inform.* 17(1-2):117–155.

Polleres, A.; Feier, C.; and Harth, A. 2006. Rules with contextually scoped negation. In *Proc. 3 rd European Semantic Web Conf. (ESWC2006*, 332–347. Springer.

Pührer, J.; Heymans, S.; and Eiter, T. 2010. Dealing with inconsistency when combining ontologies and rules using dl-programs. In Aroyo, L.; Antoniou, G.; Hyvönen, E.; ten Teije, A.; Stuckenschmidt, H.; Cabral, L.; and Tudorache, T., eds., *ESWC (1)*, volume 6088 of *Lecture Notes in Computer Science*, 183–197. Springer.

Reiter, R. 1977. On closed world data bases. In Gallaire, H., and Minker, J., eds., *Logic and Data Bases*, 55–76. Plenum Press.

Reiter, R. 1980. A logic for default reasoning. *Artif. Intell.* 13(1-2):81–132.

Rosati, R. 1997. Embedding minimal knowledge into autoepistemic logic. In *Proceedings of the Fifth Conference of the Italian Association for Artificial Intelligence (AI*IA'97)*, Lecture Notes in Artificial Intelligence, 231–241. Springer.

Torlak, E., and Jackson, D. 2007. Kodkod: A relational model finder. In Grumberg, O., and Huth, M., eds., *TACAS*, volume 4424 of *LNCS*, 632–647. Springer.

Vennekens, J.; Gilis, D.; and Denecker, M. 2006. Splitting an operator: Algebraic modularity results for logics with fixpoint semantics. *ACM Transactions on Computational Logic* 7(4):765–797.

Wittocx, J.; Denecker, M.; and Bruynooghe, M. 2010. Constraint propagation for extended first-order logic. *CoRR* abs/1008.2121.

Wittocx, J.; Mariën, M.; and Denecker, M. 2008. The IDP system: a model expansion system for an extension of classical logic. In Denecker, M., ed., *LaSh*, 153–165.