Efficient Reasoning in Multiagent Epistemic Logics

Gerhard Lakemeyer[1] and **Yves Lespérance**[2]

Abstract. In many applications, agents must reason about what other agents know, whether to coordinate with them or to come out on top in a competitive situation. However in general, reasoning in a multiagent epistemic logic such as K_n has high complexity. In this paper, we look at a restricted class of knowledge bases that are sets of modal literals. We call these proper epistemic knowledge bases (PEKBs). We show that after a PEKB has been put in prime implicate normal form (PINF), an efficient database-like query evaluation procedure can be used to check whether an arbitrary query is entailed by the PEKB. The evaluation procedure is always sound and sometimes complete. We also develop a procedure to convert a PEKB into PINF. As well, we extend our approach to deal with introspection.

1 Introduction

In many applications, agents must reason about what other agents know, whether to coordinate with them or to do well in a competitive situation. For example, if Bob has forgotten which client he is supposed to call, but knows that his secretary Alice knows, then it makes sense for him to ask her; it makes no sense for him to ask someone whom he knows does not know. A popular approach is to use multiagent modal logic [4] to model such scenarios. However, reasoning in such logics is intractable in general. For example, in the case of K_n it is PSPACE-complete [11].

As the intractability problem already arises in the single-agent case and even without explicit modalities, a number of ways of dealing with this issue have been investigated. One approach for gaining efficiency is to consider weaker logics where beliefs are no longer closed under modus ponens, e.g. [17, 9, 14, 15, 8]. Removing modus ponens altogether from the inference mechanism turns out to be rather drastic. For this reason, Lakemeyer and Levesque [16] and later Liu et al. [19] considered a tractable form of inference which allowed unit propagation and a bounded number of applications of modus ponens. Unfortunately, the underlying semantics is rather involved and cumbersome, making it hard to analyze what actually follows from a knowledge base.

An interesting alternative is to restrict the form of the knowledge base in addition to limiting the inference mechanism. In particular, Levesque [18] considered what he calls *proper knowledge bases*, which correspond to (possibly infinite) sets of ground literals. This enables the definition of a very simple and efficient query evaluation mechanism, which is always sound, and is even complete for an interesting class of queries.

Note that while some of the work mentioned above is first-order, none deals with the multiagent case. In fact with few exceptions [14, 8], none considers nested beliefs. In this paper, we will remain propositional, but consider multiagent modal logic with arbitrary nesting of epistemic operators. Our work is inspired by Levesque's investigation of proper knowledge bases. Instead of having just literals in the KB, we consider *belief literals*, which are sequences of $\Box_i$ and $\Diamond_j$ modalities followed by a literal such as $\Box_1\Diamond_2\neg p$, i.e. agent 1 believes that agent 2 thinks that p may not hold. We call knowledge bases consisting of belief literals *proper epistemic knowledge bases (*PEKB*s)*. We will then propose a simple query evaluation mechanism for PEKBs, which reduces to the one by Levesque in the non-modal case. Besides soundness we also establish completeness for queries in a certain normal form, which again reduces to the normal form introduced by Levesque in the non-modal case.

As it turns out, belief literals introduce a non-trivial complication not present in the original proper knowledge bases. There it is trivial to decide whether a KB is satisfiable by simply checking whether the KB contains complementary literals. For PEKBs this is not so easy. For example, consider the following:

$$\{\ \Box\Diamond p,\quad \Diamond\Box p,\quad \Box\Box\neg p\ \}$$

This KB is unsatisfiable in logic K, yet it takes some effort to discover this as one needs to realize that $\Diamond\Diamond p$ follows from the first two belief literals. While the example is simple, the problem turns out to be NP-complete in general. This was proved in [6] in the case of a fragment of the description logic $\mathcal{ALC}$, which is known to be a notational variant of K_n [21]. In order to cope with this complexity, we will employ results by Bienvenu [2, 3], who considers prime implicates for modal logics. The idea is to transform a PEKB into prime implicate normal form (PINF), for which the satisfiability problem is again trivial, and define the query evaluation mechanism for such KBs. While the transformation into PINF leads to a double-exponential blowup for arbitrary theories, we will see that this is not the case for sets of belief literals.

There are many applications where the kind of epistemic KBs and reasoning procedure that we propose are useful. Consider a collaborative filtering [20] application where agents make movie recommendations to users based on the evaluation of other users with similar tastes. We assume a peer to peer setting where agents have incomplete information, and may only communicate directly (and share some information) with their immediate acquaintances. They store this information as nested belief literals in their KB, for example $\Box_2\Diamond_3 Likes(M1,4)$, i.e., agent 2 believes that agent 3 considers it possible that agent 4 likes movie $M1$. In such a setting, it is easy to provide examples of epistemic queries that agents may want to ask:

- If there is some movie m such that $Favorite(m,1) \land \Diamond_2\Diamond_3 Likes(m,4)$, i.e., m is one of agent 1's favorites and agent 2 considers it possible that agent 3 considers it possible that agent

[1] Department of Computer Science, RWTH Aachen University, Aachen, Germany, email: gerhard@informatik.rwth-aachen.de
[2] Department of Computer Science and Engineering, York University, Toronto, Canada, email: lesperan@cse.yorku.ca

4 likes m, then agent 4 may be considered as a source of information/recommendations for agent 1; agent 1 may want to evaluate such a query; note that an agent may want to consider similar more deeply nested queries.

- If for all movies m, $Favorite(m,1) \supset \Box_2\Box_3\Box_4 Likes(m,5)$ and there exists $M1$ such that $\neg Seen(1,M1) \wedge \Box_2\Box_3\Box_4 Likes(m,5)$, then $M1$ is a good recommendation for agent 1.
- If agent 1 shares tastes with agent 3, and $\Diamond_1\Box_2 Likes(M1,3)$, i.e., agent 1 considers it possible that agent 2 believes that agent 3 likes a movie $M1$, then agent 1 may want to ask agent 2 whether it actually believes that agent 3 likes $M1$. Similarly, if $\Diamond_1\Diamond_2\Diamond_4 Likes(M1,3)$, then agent 1 may want to ask agent 2 whether agent 4 believes that agent 3 likes $M1$.

There are many other applications where this kind of reasoning is useful, e.g., games, social networks, etc.

The rest of the paper goes as follows. In the next section, we review the basics of logic K_n and Bienvenu's work on using prime implicates to reason in this logic. In Section 3, we define proper epistemic knowledge bases, present our algorithm for transforming them into prime implicate normal form, and show its correctness and complexity. In the following section, we present our query evaluation algorithm, show its soundness, show completeness for queries in a particular normal form, and discuss complexity. In Section 5, we extend our approach to deal with introspection. In the conclusion, we review our results and discuss future work.

2 Background

We first briefly review the basics of modal logic K_n. We assume that there is a finite set of agents $\mathcal{A} = \{1,\dots,n\}$; we use i and j, possibly with decorations, to range over them. Formulas in K_n are built from a set of propositional variables V (we use p, possibly with decorations, to range over these), the standard logical connectives ($\neg$, $\wedge$, and $\vee$), and the modal operators $\Box_i$ and $\Diamond_i$, for $i \in \mathcal{A}$. We use ϕ, possibly with decorations, to range over formulas of K_n. $\Box_i\phi$ means that agent i believes that ϕ and $\Diamond_i\phi$ means that agent i thinks that ϕ is possibly true. $\Diamond_i$ is the dual of $\Box_i$ and $\Diamond_i\phi \equiv \neg\Box_i\neg\phi$ is valid in the logic. The language is defined as follows using BNF notation ($\top$ stands for *True* and $\bot$ for *False*):

$$\phi ::= \top \mid \bot \mid p \mid \neg\phi \mid \phi\wedge\phi \mid \phi\vee\phi \mid \Box_i\phi \mid \Diamond_i\phi$$

By the *depth* of a formula ϕ we mean the maximum number of nestings of modal operators within ϕ.

The semantics of the logic is formally defined in terms of Kripke structures [13, 12]. A Kripke structure for n agents is a tuple $M = (W, \pi, R_1,\dots,R_n)$, where W is a set of states or possible worlds, π is a mapping from V to 2^W, and $R_i \subseteq W \times W$ for $i \in \mathcal{A}$. R_i is the belief accessibility relation for agent i and $R_i(w,w')$ means that world w' is compatible with agent i's beliefs in world w. We define satisfaction/truth of a formula ϕ in a structure M and world w, written $M,w \models \phi$, as follows:

$M,w \models \top$
$M,w \not\models \bot$
$M,w \models p$ iff $w \in \pi(p)$
$M,w \models \neg\phi$ iff $M,w \not\models \phi$
$M,w \models \phi\wedge\phi'$ iff $M,w \models \phi$ and $M,w \models \phi'$
$M,w \models \phi\vee\phi'$ iff $M,w \models \phi$ or $M,w \models \phi'$
$M,w \models \Box_i\phi$ iff $M,w' \models \phi$ for all $w' \in R_i(w,w')$
$M,w \models \Diamond_i\phi$ iff $M,w' \models \phi$ for some $w' \in R_i(w,w')$

ϕ is valid iff for all structures M and worlds $w \in W$, $M,w \models \phi$. ϕ entails ϕ', i.e., $\phi \models \phi'$, iff for all structures M and worlds $w \in W$, if $M,w \models \phi$, then $M,w \models \phi'$. As mentioned earlier, reasoning in K_n has high complexity: checking satisfiability/validity/entailment is PSPACE-complete [11].

A *non-modal literal* is a propositional variable or the negation of a propositional variable. We say that a formula is in *negation normal form (NNF)* if in it, negation only appears in front of propositional variables. Note that there is a linear-time algorithm to transform any formula into an equivalent formula in NNF (see [3]).

Following Bienvenu [3], we define the notions of *modal literal* l, *clause* c, and *term* t as follows:

$$l ::= \top \mid \bot \mid p \mid \neg p \mid \Box_i\psi \mid \Diamond_i\psi$$
$$c ::= l \mid c \vee c$$
$$t ::= l \mid t \wedge t$$

(in the above ψ ranges over arbitrary K_n formulas in NNF). Note that according to this definition, a clause (term) may contain conjunctions (disjunctions resp.) as long as they are in the scope of a modal operator. Bienvenu has shown that this definition (unlike some alternatives) has many nice properties and can be used to define well behaved notions of (prime) implicates. A clause c is an *implicate* of a formula ϕ iff $\phi \models c$ (throughout $\models$ denotes entailment in K_n, unless stated otherwise). A clause c is a *prime implicate* of a formula ϕ iff c is an implicate of ϕ and for all c', if c' is an implicate of ϕ and $c' \models c$, then $c \models c'$.

As defined in [2], a formula ϕ is in *prime implicate normal form (PINF)* if and only if it satisfies one of the following conditions:

1. $\phi = \bot$
2. $\phi = \top$
3. ϕ is satisfiable and falsifiable and $\phi = c_1 \wedge \ldots \wedge c_p$ where
   (a) $c_i \not\models c_j$ for $i \neq j$
   (b) each prime implicate of ϕ is equivalent to some c_i
   (c) every c_i is a prime implicate of ϕ such that (i) if d is a disjunct of c_i, then $\not\models c_i \equiv c_i \setminus \{d\}$, (ii) for every agent j, there is at most one disjunct of c_i of the form $\Diamond_j\phi'$, (iii) for every disjunct d of c_i of the form $\Box_j\phi'$ or $\Diamond_j\phi'$ for some j, ϕ' is in PINF, and (iv) if there are disjuncts $\Diamond_j\phi'$ and $\Box_j\phi''$ of c_i for some j, then $\phi' \models \phi''$.

Bienvenu [2] defines a procedure *PINF*(ϕ) that takes any formula ϕ and returns a formula in PINF that is equivalent to ϕ. She also shows that in general, the smallest formula in PINF that is equivalent to ϕ may have a length double-exponential in the length of ϕ, $|\phi|$.

Bienvenu [2] also defines a procedure Π-$Subsume(\phi_1,\phi_2)$ to decide whether a PINF formula ϕ_1 entails another PINF formula ϕ_2. The procedure runs efficiently in quadratic time in $|\phi_1| + |\phi_2|$. Thus if one is willing to pay the price of compiling both the knowledge base and query into PINF, one can check whether a query is entailed by the KB very efficiently.

3 Proper Epistemic Knowledge Bases

We will now look at a class of K_n theories/knowledge bases that support a limited form of incomplete knowledge about the epistemic state of agents. A *restricted modal literal* r is a modal literal that does not contain any $\vee$, $\wedge$, $\top$, or $\bot$, and where negation only appears in front of propositional variables, i.e.:

$$r ::= p \mid \neg p \mid \Box_i r \mid \Diamond_i r$$

A *proper epistemic knowledge base (PEKB)* is a conjunction (or equivalently a set) of restricted modal literals. Note that PEKBs are always in NNF.

First, we note that:

Proposition 1 *Any non-empty PEKB ϕ is falsifiable.*

Proof: ϕ is falsifiable iff $\neg\phi$ is satisfiable. Take the first disjunct of $\neg\phi$. This is a restricted modal literal which does not contain $\top$ or $\bot$. So it is straightforward to construct a structure that satisfies it, and this structure satisfies $\neg\phi$. ∎

Below, we will use the following abbreviations (where t stands for any term):

- $B_i(t) = \{\phi \mid \Box_i\phi \in t\}$.
- $D_i(t) = \{\phi \mid \Diamond_i\phi \in t\}$.
- $Prop(t) = \{l \mid l \text{ is a non-modal literal and } l \in t\}$.

Our algorithm to convert a PEKB into an equivalent simple formula in prime implicate normal form is as follows:

Function *PEKB2PINF*(ϕ) : takes a non-empty PEKB ϕ, i.e. a conjunction of restricted modal literals, and returns a simple formula (see below) in PINF that is equivalent to ϕ

1. Let $F_i(\phi) = \bigwedge_{\psi \in B_i(\phi)} \psi$.
   Let $\Delta(\phi) = Prop(\phi) \cup$
   $\{\Diamond_i(\psi \wedge F_i(\phi)) \mid \psi \in D_i(\phi) \text{ and } B_i(\phi) \neq \emptyset\} \cup$
   $\{\Diamond_i(\psi) \mid \psi \in D_i(\phi) \text{ and } B_i(\phi) = \emptyset\} \cup$
   $\{\Box_i(F_i(\phi)) \mid B_i(\phi) \neq \emptyset\}$.
   Finally, let $\Sigma = \Delta(\phi)$.
2. For each $l \in \Sigma$, if l is of the form $\Diamond_i(\psi)$, replace it by $\Diamond_i(\textit{PEKB2PINF}(\psi))$ and if l is of the form $\Box_i(\psi)$, replace it by $\Box_i(\textit{PEKB2PINF}(\psi))$.
3. If either $\Diamond_i\bot$ is in Σ or if both p and $\neg p$ are in Σ, return $\bot$, otherwise return $\bigwedge_{\psi\in\Sigma} \psi$.

This algorithm is similar to Bienvenu's, but somewhat simpler. We do not explicitly check whether ϕ is unsatisfiable or valid. Instead, we detect unsatisfiability in step 3 (a non-empty PEKB cannot be valid). Moreover, we do not filter out implicates of ϕ in Σ that are not prime implicates, as they must all be prime implicates. Here are some simple examples: $\textit{PEKB2PINF}(\Box_i p \wedge \Diamond_i q) = \Box_i p \wedge \Diamond_i(q \wedge p)$; $\textit{PEKB2PINF}(\Box_j(\Box_i p \wedge \Diamond_i q) \wedge \Diamond_j q') =$
$\Box_j(\Box_i p \wedge \Diamond_i(q \wedge p)) \wedge \Diamond_j(q' \wedge \Box_i p \wedge \Diamond_i(q \wedge p))$.

Let us show that this algorithm is correct. First, we have:

Lemma 1 *For any term t, every prime implicate of t is equivalent to some element of $\Delta(t)$.*

Proof: See [3] Lemma 16.2. ∎

Definition 1 *The set of* simple formulas *is the smallest set which*

1. *includes every non-modal literal and*
2. *if $\phi_1, \ldots, \phi_n$ are simple formulas, then $\phi_1 \wedge \ldots \wedge \phi_n$, $\Box_i(\phi_1 \wedge \ldots \wedge \phi_n)$ and $\Diamond_i(\phi_1 \wedge \ldots \wedge \phi_n)$ are simple formulas.*

We can now show that:

Theorem 1 *For any non-empty PEKB ϕ, PEKB2PINF(ϕ) terminates and the formula it returns is a simple formula in prime implicate normal form that is equivalent to ϕ, and has depth at most that of ϕ.*

Proof: By induction on the depth of ϕ. Base case, where ϕ has depth 0: Then ϕ is a conjunction of non-modal literals. By Proposition 1, ϕ cannot be a tautology. If ϕ is unsatisfiable, then it must contain complementary literals; then, $\bot$ is correctly returned at step 3. Finally, suppose that ϕ is both satisfiable and falsifiable. Then, the algorithm eliminates duplicate literals and returns the conjunction of the remaining ones, which is clearly in PINF and is a simple formula. Inductive case: Assume that the result holds whenever ϕ has depth at most k. We show it must hold if ϕ has depth $k+1$. ϕ is a conjunction of restricted modal literals. By Proposition 1, it cannot be a tautology. If ϕ is unsatisfiable, then it must either contain complementary non-modal literals or the set of its modal literals is unsatisfiable. In the former case, $\bot$ is returned at step 3. For the latter case, a set of modal literals is inconsistent iff it entails $\Diamond_i\bot$ for some i. In step 1, we set Σ to $\Delta(\phi)$, which is equivalent to ϕ. By the above and Lemma 1, some element of Σ is equivalent to $\Diamond_i\bot$ for some i. As well, for every modal literal $\Diamond_i\psi(\Box_i\psi) \in \Sigma$, ψ is a non-empty PEKB. In step 2, we apply the *PEKB2PINF* function to these ψ, which are of depth at most k. By the induction hypothesis the result is equivalent to ϕ, is a simple formula, and the arguments of modal operators in it are in PINF. It follows that it contains $\Diamond_i\bot$. Given this, $\bot$ is returned at step 3 as required.
Finally, consider the case where ϕ is both satisfiable and falsifiable. In step 1, we set Σ to $\Delta(\phi)$, which is equivalent to ϕ. In step 2, we apply the *PEKB2PINF* function to subformulas of depth at most k which are PEKBs. By the induction hypothesis, the result is equivalent to ϕ, is a simple formula, and has depth at most $k+1$. Step 3 just conjoins the literals in Σ. Thus *PEKB2PINF*(ϕ) is equivalent to ϕ, is a simple formula, and has depth at most that of ϕ. It remains to show that the result is in PINF. Clearly, *PEKB2PINF*(ϕ) is a conjunction of literals. We have already shown that Σ is equivalent to ϕ. By Lemma 1, all prime implicates of ϕ are equivalent to some literal in Σ. In step 2, we apply *PEKB2PINF* to the arguments of the modal literals in Σ, which are PEKBs of depth at most k. By the induction hypothesis, the result is equivalent to ϕ, is a simple formula, and every modal literal in it is in PINF. Thus every prime implicate of ϕ is equivalent to some literal in *PEKB2PINF*(ϕ). No literal in *PEKB2PINF*(ϕ) implies another as there is at most one $\Box_i$ for each i and all the $\Diamond_i$ are obtained from distinct restricted modal literals. It is easy to check that condition 3c holds. Thus *PEKB2PINF*(ϕ) is in PINF. ∎

Next, we examine the spatial complexity of the prime implicate normal form for PEKBs, i.e. conjunctions of restricted modal literals. First, we show that:

Theorem 2 *Every conjunction of restricted modal literals ϕ is equivalent to a formula in prime implicate normal form whose length is at most exponential in the depth of ϕ.*

Proof: Clearly, *PEKB2PINF*(ϕ) makes at most $|\phi|$ recursive calls to *PEKB2PINF* with an argment of length at most $|\phi|$ which has a depth one less than that of ϕ. Apart from the recursive calls, *PEKB2PINF*(ϕ) runs in $O(|\phi|^2)$. Thus the running time of *PEKB2PINF*(ϕ) is $O(|\phi|^{d+2})$, where d is the modal depth of ϕ. ∎

As mentioned earlier, Bienvenu has shown that for an arbitrary K_n formula ϕ, the smallest formula in PINF that is equivalent to ϕ may have a length double-exponential in the length of ϕ. So the spatial complexity of PINF for PEKBs is much less.

We can also show that this upper bound is optimal in that there are PEKBs for which the transformation to PINF does involve an exponential blowup in length:

Theorem 3 *There exists a conjunction of restricted modal literals ϕ such that the smallest equivalent formula in prime implicate normal form has a length which is exponential in the depth of ϕ.*

Proof: Let $\phi_0 = p_0$ and $\phi_{i+1} = \Box(\phi_i) \wedge \Diamond p_{i+1}$. The depth of ϕ_n is n and $|\phi_n| = 4n + 1$ (where $|\phi|$ is the total number of propositional variables, connectives, and modal operators in ϕ). Note that strictly speaking, ϕ_n is not a PEKB, but it can easily be transformed into one whose length is $O(n^2)$ without affecting the depth. Then *PINF*(ϕ_0) $= \phi_0$ and *PINF*(ϕ_{i+1}) $= \Box($*PINF*$(\phi_i)) \wedge \Diamond(p_{i+1} \wedge$ *PINF*$(\phi_i))$. It is easy to show that $|$*PINF*$(\phi_n)|$ is $O(2^n)$. ∎

4 Query Evaluation

Let us now consider a query evaluation mechanism V for arbitrary queries in NNF and knowledge bases which consist of simple formulas. Let us call such knowledge bases *simple epistemic knowledge bases (SEKBs)*. From now on we use the term *query* to mean an arbitrary formula in NNF. V is inspired by the method introduced by Levesque for (non-modal) proper knowledge bases [18]. V is a mapping from SEKBs Σ and queries ϕ into $\{0, 1\}$. Calling $V[\Sigma, \phi]$ amounts to asking whether Σ entails ϕ, and a value 1 should be interpreted as *yes* and 0 as *don't know*.[3] The inference V performs is similar to database retrieval. For instance, when the query is a disjunction, V returns 1 just in case it returns 1 for one of the disjuncts. Evaluating queries in this manner is very efficient and always sound, but incomplete in general. Later we will see that we also have completeness for interesting classes of queries, provided the SEKB is also in PINF.

In the following we use set notation to access conjuncts of simple formulas (or SEKBs), i.e. $\phi \in \Sigma$ means that ϕ is one of the conjuncts of Σ. Since the arguments of $\Box_i$ and $\Diamond_i$ within a simple formula are themselves simple formulas, we do this recursively. V is then defined as follows for any SEKB Σ and query ϕ:

Definition 2 *1.* $V[\Sigma, \phi] = 1$ *if* $\Sigma = \bot$*; otherwise:*

2. $V[\Sigma, \top] = 1$*;*

3. $V[\Sigma, \bot] = 0$*;*

4. If l is a non-modal literal, then
$$V[\Sigma, l] = \begin{cases} 1 & \textit{if } l \in \Sigma \\ 0 & \textit{otherwise} \end{cases}$$

5. $V[\Sigma, \phi \vee \psi] = \max(V[\Sigma, \phi], V[\Sigma, \psi])$*;*

6. $V[\Sigma, \phi \wedge \psi] = \min(V[\Sigma, \phi], V[\Sigma, \psi])$*;*

7. $$V[\Sigma, \Box_i\phi] = \begin{cases} 1 & \textit{if for some } \Box_i\Sigma' \in \Sigma, \\ & V[\Sigma', \phi] = 1 \\ 0 & \textit{otherwise} \end{cases}$$

8. $$V[\Sigma, \Diamond_i\phi] = \begin{cases} 1 & \textit{if for some } \Diamond_i\Sigma' \in \Sigma, \\ & V[\Sigma', \phi] = 1 \\ 0 & \textit{otherwise} \end{cases}.$$

Note that when restricted to the non-modal case, where a SEKB simply consists of propositional literals, V behaves essentially like the version proposed by Levesque. To see what properties we have in the presence of modalities, the following well-known property of K_n will be useful:

Lemma 2 $\models \phi \supset \psi$ *iff* $\models \Box_i\phi \supset \Box_i\psi$ *iff* $\models \Diamond_i\phi \supset \Diamond_i\psi$.

Theorem 4 (Soundness of V) *Let Σ be a SEKB and ϕ a query. If $V[\Sigma, \phi] = 1$ then $\Sigma \models \phi$.*

Proof: If $\Sigma = \bot$ or $\phi = \top$ then V returns 1, which is the correct answer. For the other cases we proceed by induction on ϕ. If $\phi = l$ for some non-modal literal l and $V[\Sigma, l] = 1$, then $l \in \Sigma$ and, hence, $\Sigma \models l$. If $V[\Sigma, \phi \vee \psi] = 1$ then $V[\Sigma, \phi] = 1$ or $V[\Sigma, \psi] = 1$. Therefore, by induction, $\Sigma \models \phi$ or $\Sigma \models \psi$, from which $\Sigma \models \phi \vee \psi$ follows. The case for $\phi \wedge \psi$ is similar. If $V[\Sigma, \Box_i\phi] = 1$ then $V[\Sigma', \phi] = 1$ for some $\Box_i\Sigma' \in \Sigma$. Therefore, by induction, $\Sigma' \models \phi$ and, by Lemma 2, $\Box_i\Sigma' \models \Box_i\phi$. Since $\Box_i\Sigma' \in \Sigma$, $\Sigma \models \Box_i\phi$ follows. The case for $\Diamond_i\phi$ is completely symmetric. ∎

From the definition of V, it is straightforward to derive an algorithm to compute V in polynomial time. Indeed, a naive implementation would run in time $O(n^2)$, where n is the size of the knowledge base.[4]

Since V is polynomial, it clearly must be incomplete. Here is an example: Let Σ be the empty set and consider $\Box_i p \supset \Box_i(p \vee q)$, whose NNF is $\phi = \Diamond_i\neg p \vee \Box_i(p \vee q)$. Clearly, ϕ is valid in K_n, that is, $\Sigma \models \phi$ yet $V[\Sigma, \phi] = 0$.

When both the query and the SEKB are in PINF, then V is complete.

Theorem 5 *Let Σ and ϕ be in* PINF.
Then $V[\Sigma, \phi] = 1$ iff $\Sigma \models \phi$.

Proof: Follows from the completeness of the subsumption algorithm in [2, 1] ∎

While this theorem says that we can always get completeness by first converting both the SEKB and ϕ into PINF, there is no free lunch because, as we saw, this conversion can lead to a double-exponential blowup of ϕ. As we argued earlier, having the SEKB in PINF does not seem so bad as the exponential blowup is restricted to the depth of belief literals in the original PEKB. So from now on, let us assume that the SEKB is in PINF. Under this assumption, it turns out that there are many other cases for which V is complete, even if the query is not in PINF. A simple example is $\Box_i p \wedge \Box_i q$, which is not in PINF but for which V will give the correct answer. To characterize a class of formulas for which V is complete we again follow Levesque and define a kind of normal form $\mathcal{NF}$, which covers cases like the above example.

[3] Levesque used a three-valued V where 1 had the same meaning as here, 0 meant that Σ entails $\neg\phi$ and $1/2$ stood for *don't know*. The only difference is that we would need to explicitly call V with both ϕ and $\neg\phi$ to get the same effect.

[4] With some indexing on the contents of the knowledge base this can be easily reduced to $O(n \log n)$.

Definition 3 (Logical Separability) *A set of formulas Γ is logically separable iff for every satisfiable set of simple formulas L, if $L \cup \Gamma$ is unsatisfiable, then $L \cup \{\phi\}$ is unsatisfiable for some $\phi \in \Gamma$.*

Intuitively, logical separability tries to capture the property that there are no logical puzzles hidden within the formulas of Γ. For example, $\{\Box_i p, \Box_i q\}$ is logically separable, while $\Gamma = \{\Box_i p, \Box_i(p \supset q)\}$ is not, since $\Gamma \cup \{\Diamond_i \neg q\}$ is inconsistent yet satisfiable when only one of the two sentences in Γ is considered.

Definition 4 *$\mathcal{NF}$ is the least set such that*

1. *if ϕ is a propositional variable, then $\phi \in \mathcal{NF}$;*
2. *if $\phi \in \mathcal{NF}$, then $\neg\phi \in \mathcal{NF}$;*
3. *if $\phi \in \mathcal{NF}$, then $\Box_i \phi \in \mathcal{NF}$;*
4. *if $\Gamma \subseteq \mathcal{NF}$, Γ is logically separable, and Γ is finite, then $\bigwedge \Gamma \in \mathcal{NF}$.*

Note that our definition of $\mathcal{NF}$ differs from that of Levesque only in two places: in the definition of logical separability Levesque uses sets of propositional literals L and, of course, his $\mathcal{NF}$ does not consider $\Box_i$ formulas. Note also that formulas in $\mathcal{NF}$ do not mention $\top$ and $\bot$.

The main theorem of this section is that V is correct for queries in $\mathcal{NF}$. To prove it we need two lemmas.

Lemma 3 *Let $\Sigma = \{\Box_i \Sigma'_1, \Diamond_i \Sigma'_2, \ldots, \Diamond_i \Sigma'_n, l_1, \ldots, l_m\}$ be a consistent set of formulas, where the l_i are non-modal literals. Then $\Sigma \models \Box_i \phi$ iff $\Box_i \Sigma'_1 \models \Box_i \phi$.*

Proof: The if direction is immediate. Conversely, suppose $\Sigma \models \Box_i \phi$ yet $\Box_i \Sigma'_1 \not\models \Box_i \phi$, that is, $\{\Box_i \Sigma'_1, \Diamond_i \neg\phi\}$ is satisfiable. Let $M = (W, \pi, R_j)$ be a model[5] such that $M, w \models \Sigma$ for some $w \in W$. Let $M' = (W', \pi', R'_j)$ be a model such that $W \cap W' = \{\}$ and $M', w' \models \Box_i \Sigma'_1 \wedge \Diamond_i \neg\phi$. Then there is a w'' such that $w' R'_i w''$ and $M', w'' \models \Sigma'_1 \wedge \neg\phi$. Now construct a new model $M^* = (w^*, \pi^*, R^*_i)$, where $W^* = W \cup W'$, $\pi^*(p) = \pi(p) \cup \pi'(p)$, and R^*_j is the union of R_j and R'_j together with $w R^*_j w''$. Since $M^*, w'' \models \Sigma'_1 \wedge \neg\phi$, it is easy to see that $M^*, w \models \Sigma \wedge \Diamond_i \neg\phi$, contradicting the assumption that $\Sigma \models \Box_i \phi$. ∎

Lemma 4 *Let $\Sigma = \{\Box_i \Sigma'_1, \Diamond_i \Sigma'_2, \ldots, \Diamond_i \Sigma'_n, l_1, \ldots, l_m\}$ be a consistent set of simple formulas in* PINF*. Then $\Sigma \models \Diamond_i \phi$ iff for some $\Diamond_i \Sigma'_j \in \Sigma$, $\Diamond_i \Sigma'_j \models \Diamond_i \phi$.*

Proof: The if direction is immediate. Conversely, suppose $\Sigma \models \Diamond_i \phi$ yet $\Diamond_i \Sigma'_k \not\models \Diamond_i \phi$ for all k, that is, $\{\Sigma'_k, \neg\phi\}$ is satisfiable for all k. Let $M^k = (W^k, \pi^k, R^k_j)$ be models with disjoint W^k and $w^k \in W^k$ such that $M^k, w^k \models \Sigma'_k \wedge \neg\phi$ for each k. Construct a new model $M = (W, \pi, R_j)$, where $W = \bigcup W^k \cup \{w\}$ for some new world w, $\pi(p) = \bigcup \pi^k(p)$ if $p \notin \Sigma$ and $\pi(p) = \bigcup \pi^k(p) \cup \{w\}$ otherwise, and $R_j = \bigcup R^k_i \cup \{w R_j w^k\}$. By the construction of π, $M, w \models l$ for all literals $l \in \Sigma$. Since Σ is in PINF, we have $\Sigma'_k \models \Sigma'_1$ for all k and hence $M, w \models \Box_i \Sigma'_1$. Also, since $w R_j w^k$, we have $M, w \models \Diamond_i \Sigma'_k$ and since the w^k are the only accessible worlds from w, we have $M, w \models \Box_i \neg\phi$. In sum, $M, w \models \Sigma \wedge \Box_i \neg\phi$, contradicting the assumption that $\Sigma \models \Diamond_i \phi$. ∎

[5] We use the notation $M = (W, \pi, R_j)$ with j ranging over the set of all agents $\mathcal{A}$ as a shorthand for $M = (W, \pi, R_1, \ldots, R_n)$.

Theorem 6 *Let Σ be a SEKB in* PINF *and ϕ a formula in $\mathcal{NF}$ and in negation normal form. Then $V[\Sigma, \phi] = 1$ iff $\Sigma \models \phi$.*

Proof: If Σ is inconsistent, then $\Sigma = \bot$ because of the definition of PINF. In that case, the lemma holds trivially since $V[\bot, \phi] = 1$ and $\bot \models \phi$.

If Σ is consistent, the proof is by induction on the length of ϕ. If ϕ is a literal l, then $V[\Sigma, l] = 1$ iff $l \in \Sigma$ iff $\Sigma \models l$.

Suppose the lemma holds for formulas of length less than n and let ϕ be of length n. $V[\Sigma, \phi \wedge \psi] = 1$ iff $V[\Sigma, \phi] = 1$ and $V[\Sigma, \psi] = 1$ iff (by induction) $\Sigma \models \phi$ and $\Sigma \models \psi$ iff $\Sigma \models \phi \wedge \psi$.

Let $V[\Sigma, \phi \vee \psi] = 1$. Then $V[\Sigma, \phi] = 1$ or $V[\Sigma, \psi] = 1$ and hence (by induction) $\Sigma \models \phi$ or $\Sigma \models \psi$, from which $\Sigma \models \phi \vee \psi$ follows. Conversely, let $\Sigma \models \phi \vee \psi$. Then $\Sigma \cup \{\neg\phi, \neg\psi\}$ is inconsistent. Since $\phi \vee \psi$ is in $\mathcal{NF}$, $\{\neg\phi, \neg\psi\}$ are logically separable. Since Σ is equivalent to a set of simple formulas and, by the definition of logical separability, $\Sigma \cup \{\neg\phi\}$ or $\Sigma \cup \{\neg\psi\}$ is inconsistent, that is, $\Sigma \models \phi$ or $\Sigma \models \psi$ and hence, by induction, $V[\Sigma, \phi] = 1$ or $V[\Sigma, \psi] = 1$, which implies $V[\Sigma, \phi \vee \psi] = 1$.

Let $V[\Sigma, \Box_i \phi] = 1$. Then, by definition, $V[\Sigma', \phi] = 1$ for $\Box_i \Sigma' \in \Sigma$. If Σ' is inconsistent, then $\Sigma' = \bot$ by PINF and $\Box_i \Sigma' \models \Box_i \phi$ and thus $\Sigma \models \Box_i \phi$. If Σ' is consistent then $V[\Sigma', \phi] = 1$ iff $\Sigma' \models \phi$ by induction. By Lemma 2, $\Box_i \Sigma' \models \Box_i \phi$ and then, by Lemma 3, $\Sigma \models \Box_i \phi$. To prove the converse, let $\Sigma \models \Box_i \phi$. Again by Lemma 3, $\Box_i \Sigma' \models \Box_i \phi$ for $\Box_i \Sigma' \in \Sigma$. By Lemma 2, $\Sigma' \models \phi$. Thus $V[\Sigma', \phi] = 1$ by induction, from which $V[\Sigma, \Box_i \phi] = 1$ follows by the definition of V. (Note that Σ' is itself a SEKB in PINF so that the induction hypothesis applies.)

$V[\Sigma, \Diamond_i \phi] = 1$ iff $V[\Sigma', \phi] = 1$ for some $\Diamond_i \Sigma' \in \Sigma$ iff $\Sigma' \models \phi$ by induction iff $\Diamond_i \Sigma' \models \Diamond_i \phi$ by Lemma 2 iff $\Sigma \models \Diamond_i \phi$ by Lemma 4. Note that Σ' in this case must be consistent since otherwise the PINF would have reduced $\Diamond_i \Sigma'$ to $\bot$, making Σ inconsistent. ∎

5 Extension to $K45_n$

In many applications, one wants to assume that agents can correctly introspect on their own beliefs. One gets this by using the multiagent modal logic $K45_n$, where the following are valid:

$$\mathbf{4}: \quad \Box_i \phi \supset \Box_i \Box_i \phi$$
$$\mathbf{5}: \quad \neg\Box_i \phi \supset \Box_i \neg\Box_i \phi$$

That is, if an agent believes ϕ, then it believes that it believes ϕ, and if it does not believe ϕ, then it believes that it does not believe ϕ.

Let us see how our approach can be adapted to compute $K45_n$ entailment. First note that in $K45_n$, the following are valid:

$$\Box_i \Box_i \phi \equiv \Box_i \phi \qquad \Box_i \Diamond_i \phi \equiv \Diamond_i \phi \vee \Box_i \bot$$
$$\Diamond_i \Diamond_i \phi \equiv \Diamond_i \phi \qquad \Diamond_i \Box_i \phi \equiv \Box_i \phi \wedge \Diamond_i \phi$$

Given this, we can already eliminate any string of consecutive modal operators with the same agent in any PEKB. In general, any formula in $K45_n$ can be converted into a so-called reduced form so that the $K45_n$-validity problem for such formulas reduces to K_n-validity, and thus our existing method V can be applied. More precisely, we call a formula ϕ *i-objective* if all the modal operators which do not occur in the scope of another modal operator belong to agents other than i. In other words, i-objective formulas are about what is true in the world and about other agents' beliefs. For example, $p \vee \Box_j \neg\Box_i p$ is i-objective, but $p \vee \Box_i \neg\Box_i p$ is not. Let us call a formula ϕ *reduced* if for all subformulas $\Box_i \psi$ and $\Diamond_i \psi$, ψ is i-objective.

Lemma 5 *For any Σ and α there are Σ^* and α^* in reduced form such that $\Sigma \models_{K45_n} \alpha$ iff $\Sigma^* \models_{K_n} \alpha^*$.*

Proof: A proof can be found in [10]. ∎

We remark that transforming a formula into reduced form preserves NNF. However, when given a PEKB or SEKB, its reduced form may no longer be a PEKB or SEKB. This is because of the top right reduction rule above, which introduces a disjunction. For that reason we need to assume that any restricted modal literal, which is added to a knowledge base, is already in reduced form. Queries, on the other hand, can be arbitrary to start with and then pre-processed to turn them into reduced form. Then we obtain:

Theorem 7 *Let Σ be a reduced SEKB in* PINF *and ϕ a formula in $\mathcal{NF}$ and in reduced negation normal form. Then $V[\Sigma, \phi] = 1$ iff $\Sigma \models_{K45_n} \phi$.*

The proof is similar to the case of K_n and makes use of the fact that the algorithm to convert a reduced PEKB into PINF is the same for K_n and $K45_n$, and the corresponding Lemmas 2, 3, and 4 hold in $K45_n$ when restricted to reduced formulas.

6 Conclusion and Future Work

In this paper, we have proposed an approach to efficient reasoning in multiagent epistemic logic. Our approach focuses on a restricted class of knowledge bases (PEKBs) that are sets of modal literals. We have shown that if we put a PEKB in a particular normal form (PINF) proposed by Bienvenu, we can use an efficient database-like query evaluation procedure (V) to check entailment. We showed that the procedure is always sound. Moreover for queries in a certain normal form ($\mathcal{NF}$), the evaluation procedure is complete. We also proposed a procedure for putting PEKBs in PINF and showed its correctness. The procedure is efficient if the modal depth of the PEKB is small relative to its size.

We saw that complexity of reasoning in our approach depends on modal depth. In the psychological literature, it has been claimed that humans can only handle a limited amount of nesting in reasoning about the mental attitudes of others [22]. If this is the case, our approach might work quite well in many practical applications. On the other hand, work in philosophy and on distributed systems suggests that many interesting types of deception can be characterized by differences in beliefs at various levels of nesting [7], e.g. X believes that p but also believes that Y believes that X believes that not p (in this case, the difference is quite shallow, but in others it is not).

The kind of KBs considered in this paper have a very restricted form. While it may be sufficiently expressive for some applications, for many others it will not be. One important epistemic notion is *knowing whether* something holds, which can be expressed as $\Box_i\phi \vee \Box_i\neg\phi$. It would be very useful to allow this kind of information in our KBs, at least in the case where ϕ is a restricted modal literal. Another notion that is very important in coordination is common knowledge/mutual belief. It would be very useful to support reasoning about it. It would also be good to extend our approach to logics such as KT_n, where beliefs must be true, and KD_n, where beliefs must be consistent.

Another area for future work is first-order epistemic logic. As mentioned in the introduction, our work was partly inspired by the "proper KBs" approach to efficient reasoning in a first-order context. It would be interesting to examine whether the approach we developed in this paper can be combined with the techniques proposed for the first order case. An important consideration would be handling "quantifying in", e.g. X knows that Y knows who killed Z.

Also note that in general, to allow agents to reason about others so as to coordinate or compete with them, decide whether to communicate and what to say, we need much more than reasoning about beliefs. Reasoning about other agents' goals and intentions as well as reasoning about action is also required, for instance to model how requesting something from another agent can further one's goals. For this, one would need efficient reasoning techniques for a full logic of rational agency, albeit one with limited expressiveness [5]. Such techniques could then be used to develop agents that can work effectively with others, without having to prescript their interactions.

REFERENCES

[1] M. Bienvenu, 'Prime implicate normal form for $\mathcal{ALC}$ concepts: Long version', Research report IRIT/RR–2008-6–FR, IRIT, Université Paul Sabatier, (April 2008).

[2] Meghyn Bienvenu, 'Prime implicate normal form for $\mathcal{ALC}$ concepts', in *AAAI*, pp. 412–417, (2008).

[3] Meghyn Bienvenu, 'Prime implicates and prime implicants: From propositional to modal logic', *Journal of Artificial Intelligence Research*, **36**, 71–128, (2009).

[4] Brian Chellas, *Modal Logic: An Introduction*, Cambridge University Press, 1980.

[5] Maria del Pilar Pozos Parra, Abhaya C. Nayak, and Robert Demolombe, 'Theories of intentions in the framework of situation calculus', in *DALT*, pp. 19–34. Springer LNCS 3476, (2004).

[6] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Bernhard Hollunder, Werner Nutt, and Alberto Marchetti-Spaccamela, 'The complexity of existential quantification in concept languages', *Artif. Intell.*, **53**(2-3), 309–327, (1992).

[7] R. Fagin, J.Y. Halpern, Y. Moses, and M.Y. Vardi, *Reasoning about Knowledge*, MIT Press, 1995.

[8] Ronald Fagin, Joseph Y. Halpern, and Moshe Y. Vardi, 'A nonstandard approach to the logical omniscience problem', *Artif. Intell.*, **79**(2), 203–240, (1995).

[9] Alan M. Frisch, 'Inference without chaining', in *IJCAI*, pp. 515–519, (1987).

[10] Joseph Y. Halpern and Gerhard Lakemeyer, 'Multi-agent only knowing', in *Journal of Logic and Computation*, pp. 251–265. Oxford University Press, (2001).

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

[12] Jaakko Hintikka, *Knowledge and Belief*, Cornell University Press, Ithaca, NY, 1962.

[13] Saul A. Kripke, 'Semantical considerations on modal logic', *Acta Philosophica Fennica*, **16**, 83–94, (1963).

[14] Gerhard Lakemeyer, 'Tractable meta-reasoning in propositional logics of belief', in *IJCAI*, pp. 401–408, (1987).

[15] Gerhard Lakemeyer, 'Limited reasoning in first-order knowledge bases with full introspection', *Artif. Intell.*, **84**(1-2), 209–255, (1996).

[16] Gerhard Lakemeyer and Hector J. Levesque, 'Evaluation-based reasoning with disjunctive information in first-order knowledge bases', in *KR*, pp. 73–81, (2002).

[17] Hector J. Levesque, 'A logic of implicit and explicit belief', in *AAAI*, pp. 198–202, (1984).

[18] Hector J. Levesque, 'A completeness result for reasoning with incomplete first-order knowledge bases', in *KR*, pp. 14–23, (1998).

[19] Yongmei Liu, Gerhard Lakemeyer, and Hector J. Levesque, 'A logic of limited belief for reasoning with disjunctive information', in *KR*, pp. 587–597, (2004).

[20] Paul Resnick and Hal R. Varian, 'Recommender systems - introduction to the special section', *Commun. ACM*, **40**(3), 56–58, (1997).

[21] Klaus Schild, 'A correspondence theory for terminological logics: Preliminary report', in *IJCAI*, pp. 466–471, (1991).

[22] Rineke Verbrugge and Lisette Mol, 'Learning to apply theory of mind', *Journal of Logic, Language and Information*, **17**(4), 489–511, (2008).