Knowing Minimum/Maximum n Formulae

Thomas Ågotnes [1] and **Natasha Alechina** [2]

Abstract. We introduce a logical language with nullary operators $min(n)$, for each non-negative integer n, which mean 'the reasoner has at least n different beliefs'. The resulting language allows us to express interesting properties of non-monotonic and resource-bounded reasoners. Other operators, such as 'the reasoner has at most n different beliefs' and the operator introduced in [1, 4]: 'the reasoner knows at most the formulae $\phi_1, \ldots, \phi_n$', are definable using $min(n)$. We introduce several syntactic epistemic logics with $min(n)$ operators, and prove completeness and decidability results for those logics.

1 Introduction

In this paper we propose a logical language which allows us to say that a reasoner has at least n different beliefs (or knows at least n different formulae). We take a syntactic approach to epistemic and doxastic logics, which allows an agent's beliefs to be, e.g., not closed under logical consequence, finite, and/or inconsistent. We extend the traditional language of belief logics, propostional logic with an additional unary operator B where $B\phi$ stands for 'the agent believes ϕ', with nullary operators $min(n)$, for every non-negative integer n. Such operators allow us to formulate interesting properties of reasoners, and the resulting logics have nice formal properties. We introduce several doxastic logics with $min(n)$ operators, and show that they have (weakly) complete axiomatisations, and are decidable.

In the language with $min(n)$, we can express dual operators $max(n)$ (meaning: the reasoner has at most n different beliefs), and those operators, in turn, enable us to completely axiomatically describe reasoners with a bound n on the maximal size of their belief set. The bound on the number of distinct beliefs an agent can have, naturally corresponds to a bound on the size of the agent's memory (assuming that each formula is a word of fixed size). Logics for agents with bounded memory were studied, for example, in [9], and more recently in [5]. However, in the language of standard epistemic logic it is impossible to express properties such as 'the agent can apply the rule of modus ponens to its beliefs unless its memory is full'. We show later in the paper some examples of properties of bounded memory reasoners which become expressible in the language with $min(n)$ operators.

Somewhat surprisingly, we can also define the $\triangledown$ operator introduced in [1, 4], where $\triangledown\{\phi_1, \ldots, \phi_n\}$ stands for 'the agent believes at most the formulae $\phi_1, \ldots, \phi_n$'. The latter operator makes other useful properties easy to express. For example, formalising non-monotonic arguments becomes easier, because we can say in the language 'the agent knows a set of formulae X *and nothing else*'.

The rest of the paper is organised as follows. In section 2, we introduce the language and models of syntactic epistemic logics with only the B operator, based on syntactic structures introduced in [12]. We show that formulae in this language are preserved under Σ-isomorphism (truth assignments agreeing on atomic and epistemic formulae in the set Σ) for certain sets Σ. In section 3 we introduce the language and interpretation of the logic with operators $min(n)$ and show that $min(n)$ is not definable in the basic language, since $min(n)$ formulae are not preserved under Σ-isomorphism. We also show how to define $max(n)$, $\triangledown$ and other operators in the extended language. In section 4, we give a sound and (weakly) complete axiomatisation of the logic with $min(n)$, and prove that its satisfiability problem is NP-complete. We also show that adding $max(n)$ as an axiom captures exactly the set of models where the agent's belief set has cardinality of at most n. In section 5 we define a modal logic with $min(n)$ and 'next state' modality $\Diamond$, in which we can express properties relating to dynamics of agent's beliefs, such as bounded monotonicity: until the agent's state is 'full', it carries all its current beliefs to the next state. We prove soundness and weak completeness for this logic and show that its satisfiability problem is in PSPACE.

2 Syntactic Structures

In this section we introduce the language and models of basic syntactic epistemic logic. Syntactic epistemic logic considers beliefs as syntactic objects rather than propositions (sets of possible worlds). This is done to, e.g., avoid closure under logical consequence and identifying logically equivalent beliefs. We base our account of the interpretations of the language of epistemic logic on *syntactic structures* introduced in [12].

The language $\mathcal{L}$ of basic syntactic epistemic logic is parameterised by a set $\mathcal{P}$ of primitive propositions. $\mathcal{L}$ is defined as follows.

$$\phi ::= p \in \mathcal{P} \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid B\phi$$

We use the usual derived propositional connectives. The intended meaning of $B\phi$ is that ϕ is believed by the agent. We denote the set of *epistemic atoms* $\{B\phi : \phi \in \mathcal{L}\}$ by $\mathcal{B}$. The elements of the set $\mathcal{P} \cup \mathcal{B}$ of primitive propositions and epistemic atoms, are called *atoms*.

A *syntactic structure* [12], henceforth sometimes called just "a model", is a pair $M = (S, \sigma)$, where S is a set of *states* and

$$\sigma : S \rightarrow 2^{\mathcal{P} \cup \mathcal{B}}$$

Thus a syntactic structure identifies a set of formulae believed by the agent in a state $s \in S$. We call this set the agent's *epistemic state* in s, and denote it $\overline{\sigma}(s)$: $\overline{\sigma}(s) = \{\phi : B\phi \in \sigma(s)\}$. A pair M, s, where s is a state of M, is called a *pointed model*. The definition of satisfaction of a formula ϕ in a pointed model M, s, written $M, s \models$

[1] University of Bergen, Norway, agotnes@ii.uib.no
[2] University of Nottingham, UK, nza@cs.nott.ac.uk

ϕ, is straightforward:

$$\begin{array}{lcl} M,s \models p & \Leftrightarrow & p \in \sigma(s) \\ M,s \models B\phi & \Leftrightarrow & \phi \in \overline{\sigma}(s) \\ M,s \models \neg\phi & \Leftrightarrow & M,s \not\models \phi \\ M,s \models \phi_1 \wedge \phi_2 & \Leftrightarrow & M,s \models \phi_1 \text{ and } M,s \models \phi_2 \end{array}$$

We remark that in this definition of syntactic knowledge by [12] in a possible worlds framework, the states themselves do not play any important part: the interpretation of a formula in a state s does not depend on any state different from s. Thus, the interpretation in s of any formula can be defined solely in terms of the set $\sigma(s)$. We will, however, make use of the full set of states S when we introduce syntactic *relational* structures in Section 5.

We use $\mathcal{M}$ to denote the class of all syntactic structures. $\mathcal{M}$ can be seen as models of agents with unbounded memory. In the following, we will also be interested in other classes of syntactic structures. For a given natural number n, $\mathcal{M}^n$ is the class of syntactic structures where at most n formulae are believed at the same time, i.e., where $|\overline{\sigma}(s)| \leq n$ for every $s \in S$. $\mathcal{M}^n$ can be seen as models of agents with a fixed memory size. $\mathcal{M}_{fin}$ denotes the class of syntactic structures with *finite* epistemic states, i.e., $\mathcal{M}_{fin} = \bigcup_{n \in \mathbb{N}} \mathcal{M}^n$. $\mathcal{M}_{fin}$ can be seen as models of agents with finite memory.

2.1 Preservation

As we will soon extend the logical language, we need to be able to compare the expressiveness of different languages. To do that precisely, we introduce the notion of Σ-*isomorphism* between two pointed models, where Σ is a set of atoms. Two pointed models are Σ-isomorphic, if they agree on all formulae in Σ. Formally:

Definition 1 *Let $M = (S, \sigma)$ and $M = (S', \sigma')$ be models, $s \in S$ and $s' \in S'$, and $\Sigma \subseteq \mathcal{P} \cup \mathcal{B}$. $(M, s) \sim_\Sigma (M', s')$,* (M, s) and (M', s') are Σ-isomorphic*, is defined as follows:*

$$(M, s) \sim_\Sigma (M', s') \iff \sigma(s) \cap \Sigma = \sigma'(s') \cap \Sigma$$

Given a $\phi \in \mathcal{L}$, the set $Subf(\phi)$ is the set of all subformulae of ϕ (with subformulae of the form $B\psi$ treated as atomic formulae), and $At(\phi)$ is the set of atomic subformulae of ϕ, including epistemic atoms: $At(\phi) = Subf(\phi) \cap (\mathcal{P} \cup \mathcal{B})$.

The following lemma states that the truth value of a formula depends only on the truth value of its atomic subformulae, or, in other words, satisfaction of $\mathcal{L}$ formulae ϕ is invariant under $At(\phi)$-isomorphism.

Lemma 1 *For any two pointed models M, s and M', s', $\Sigma \subseteq \mathcal{P} \cup \mathcal{B}$ and $\phi \in \mathcal{L}$ such that $At(\phi) \subseteq \Sigma$:*

$$(M, s) \sim_\Sigma (M', s') \Rightarrow (M, s \models \phi \Leftrightarrow M', s' \models \phi)$$

The proof is straightforward.

3 Upper/Lower Bounds on Belief Sets

We now extend the language $\mathcal{L}$ with operators for expressing properties of syntactic structures such as "at most n different formulae are believed". Furthermore, we show that an extension of the language indeed was necessary to express such properties.

The language $\mathcal{L}_{min}$ is defined by adding a nullary operator $min(n)$, for each natural number n, to the language $\mathcal{L}$. Formally $\mathcal{L}_{min}$ is defined as follows:

$$\phi ::= p \in \mathcal{P} \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid B\alpha : \alpha \in \mathcal{L} \mid min(n) : n \in \mathbb{N}$$

Let $M = (S, \sigma)$ be a model, and $s \in S$. The satisfaction of a $\mathcal{L}_{min}$ formula ϕ in (M, s) is defined by adding the following clause to the definition of satisfaction of $\mathcal{L}$ formulae:

$$M, s \models min(n) \Leftrightarrow |\overline{\sigma}(s)| \geq n$$

A formula $min(n)$ captures the notion that *at least* n formulae are believed. As mentioned above, we are also interested in a dual expressing that *at most* n formulae are believed. The reason that a dual operator to $min(n)$ is not included in $\mathcal{L}_{min}$, is that it is in fact derivable. We define it as the following derived operator:

$$max(n) \equiv \neg min(n+1)$$

It is easy to see that

$$M, s \models max(n) \Leftrightarrow |\overline{\sigma}(s)| \leq n$$

3.1 Expressive Power

We compare the expressive power of the languages $\mathcal{L}$ and $\mathcal{L}_{min}$: are, for example, $min(n)$ and/or $max(n)$ expressible in $\mathcal{L}$?

It was shown in the previous section that $\mathcal{L}$ formulae are invariant under Σ-isomorphism. On the other hand, as the following lemma shows, in the case of $\mathcal{L}_{min}$ formulae, satisfaction is *not* invariant under Σ-isomorphism[3].

Lemma 2 *There are pointed models M, s, M', s' and $\Sigma \subseteq \mathcal{P} \cup \mathcal{B}$ such that $(M, s) \sim_\Sigma (M', s')$, but for some $\phi \in \mathcal{L}_{min}$ with $At(\phi) \subseteq \Sigma$*

$$M, s \models \phi \text{ and } M', s' \not\models \phi$$

Proof. Take $\phi = min(1)$, $M = (S, \sigma)$, $M' = (S', \sigma')$, $\overline{\sigma}(s) = \{\alpha\}$ for some $\alpha \in \mathcal{L}$, $\overline{\sigma}(s') = \emptyset$ and $\Sigma = \emptyset$. Trivially, $At(\phi) \subseteq \Sigma$. $|\overline{\sigma}(s)| \geq 1$, so $M, s \models \phi$. $|\overline{\sigma}'(s')| \not\geq 1$, so $M', s' \not\models \phi$. □

The following theorem follows immediately from the fact that $\mathcal{L}_{min}$ can discern between Σ-isomorphic models, while $\mathcal{L}$ cannot (Lemmata 1 and 2).

Theorem 1 *$\mathcal{L}_{min}$ is strictly more expressive than $\mathcal{L}$.*

3.1.1 Knowing At Least and At Most

In [1, 4] two dual operators were introduced to express properties of syntactic knowledge. Both operators are unary, and take a *finite set* of object formulae as argument (it is thus assumed that the language has symbols for finite sets of formulae). Let $X \subseteq \mathcal{L}$ be finite. First, $\triangle X$ is intended to mean that the agent knows *at least* the formulae in the set X: the agent knows every formula in X, but might in addition also know other formulae not in X. Second, $\triangledown X$ is intended to mean that the agent knows *at most* the formulae in the set X: every formula the agent knows is in X, but it might not know every formula in X. The $\triangledown$ operator can thus be seen as a syntactic version, without the assumption that belief is closed under logical consequence, of an "only knowing" operator [15]. $\triangledown X$ is not definable by B [2]. The

[3] Most logics satisfy the principle of locality: the truth value of a formula does not depend on the assignment to variables other than the formula's free variables. This is such an obvious property that it usually goes unremarked; however some logics do violate it. This phenomenon was investigated for predicate logics in e.g. [16]; for propositional logics, the only example we know of in addition to the logics of $\triangledown$ and $min(n)$ is the logic of *only knowing* [15].

conjunction of knowing at least X and knowing at most X, knowing *exactly* X, is written $\boxtimes X$. The $\triangledown$ and $\boxtimes$ operators can be used to express compactly that the agent knows the given formulas *and nothing else*. For example, from the fact that $\boxtimes\{\texttt{Bird(Tweety)}\}$ we can derive $\neg B\neg\texttt{Flies(Tweety)}$.

Formally, satisfaction is defined as follows:

$$\begin{array}{lcl} M, s \models \triangle X & \Leftrightarrow & X \subseteq \overline{\sigma}(s) \\ M, s \models \triangledown X & \Leftrightarrow & \overline{\sigma}(s) \subseteq X \\ M, s \models \boxtimes X & \Leftrightarrow & \overline{\sigma}(s) = X \end{array}$$

It turns out that both notions of knowing at most and knowing at least a finite set of formulae are definable in $\mathcal{L}_{min}$. We leave it to the reader to check that

$$\begin{array}{l} \triangle X \equiv \bigwedge_{\alpha \in X} B\alpha \\ \boxtimes X \equiv \triangle X \wedge max(|X|) \\ \triangledown X \equiv \bigvee_{Y \subseteq X} \boxtimes Y \end{array}$$

4 Completeness and Complexity

In this section, we give a complete and sound axiomatisation of the logic of $min(n)$. Let $\mathcal{S}$ be the logic defined by the following axiom schemata and rules over the language $\mathcal{L}_{min}$:

Prop	all substitution instances of propositional logic	
MIN0	$min(0)$	
MIN1	$min(n) \rightarrow min(m)$	$m < n$
MIN2	$(B\phi_1 \wedge \cdots \wedge B\phi_n) \rightarrow min(n)$	$\forall_{i \neq j \in [1,n]} \phi_i \neq \phi_j$
MP	If $\phi, \phi \rightarrow \psi$ then ψ	

It is easy to see that all axioms are valid on all syntactic structures, and that the following holds.

Lemma 3 *$\mathcal{S}$ is sound with respect to $\mathcal{M}$.*

Observe that the logic of $\mathcal{M}$ (or of $\mathcal{M}_{fin}$) is not compact. For example, consider a set of formulae which says that the agent has at least one belief, but it does not believe any formula: $\{min(1)\} \cup \{\neg B\phi : \phi \in \mathcal{L}\}$. Every finite subset of this set is satisfiable, but the set itself is not. This means that we can at most prove weak completeness.

The remainder of this section consists of constructions and intermediate results leading up to the main completeness results in Theorems 2, 3 and 4. First, some definitions. Given sets of formulae Δ, Ξ, we say that Ξ is Δ-maximal if either $\phi \in \Xi$ or $\neg\phi \in \Xi$ for each $\phi \in \Delta$. Let $Cl(\phi)$ be the closure of $Subf(\phi)$ with respect to single negations and $min(..)$, namely:

- if $\psi \in Subf(\phi)$, then $\psi \in Cl(\phi)$
- $min(0) \in Cl(\phi)$
- $min(|\{B\alpha : B\alpha \in Subf(\phi)\}|) \in Cl(\phi)$
- if $min(n) \in Cl(\phi)$, then $min(m) \in Cl(\phi)$, for all m with $0 < m < n$
- if $\psi \in Cl(\phi)$, then $\neg\psi \in Cl(\phi)$ unless $\psi = \neg\chi$ for some χ

Clearly, $Cl(\phi)$ is finite.

To prove completeness, given an $\mathcal{S}$-consistent formula ϕ we now construct a finite model M_ϕ and show that it satisfies ϕ. When ϕ is an $\mathcal{S}$-consistent formula, let Γ_ϕ be some $Cl(\phi)$-maximal $\mathcal{S}$-consistent subset of $Cl(\phi)$ which contains ϕ (it is easy to prove that such a set exists if ϕ is $\mathcal{S}$-consistent, just pick one of them). Let $Bel(\Gamma_\phi) = \{\psi : B\psi \in \Gamma_\phi\}$. Let $m_\phi = max(m : min(m) \in \Gamma_\phi)$. Since Γ_ϕ is finite and contains $min(0)$ (by **MIN0** and the fact that $min(0) \in Cl(\phi)$), such an m_ϕ exists. By **MIN2** and the fact that $min(|\{B\alpha : B\alpha \in Subf(\phi)\}|) \in Cl(\phi)$, the cardinality of $Bel(\Gamma_\phi)$ is less or equal to m_ϕ.

To build the model M_ϕ, in the case that $|Bel(\Gamma_\phi)| = m_\phi$ we can just let the epistemic state be identical to $Bel(\Gamma_\phi)$; as we show below, it is easy to prove a truth lemma in that case. However, when $|Bel(\Gamma_\phi)| < m_\phi$ (for example, if $\phi = min(10)$, then $|Bel(\Gamma_\phi)| = 0$ and $m_\phi = 10$), we must pad the epistemic state with $m_\phi - |Bel(\Gamma_\phi)|$ extra formulae. These formulae should not come from $\{\psi : B\psi \in Subf(\phi)\}$, but we have an infinite supply of formulae in $\mathcal{L}$. So let $k_\phi = m_\phi - |Bel(\Gamma_\phi)|$ and for all $i \in \{1, \ldots, k_\phi\}$, choose some (unique) $B\alpha^i_\phi \notin Subf(\phi)$. We are now ready to define M_ϕ. Let $M_\phi = (\{s_\phi\}, \sigma_\phi)$ where σ_ϕ is such that

$$p \in \sigma_\phi(s_\phi) \Leftrightarrow p \in \Gamma_\phi \text{ when } p \in \mathcal{P}$$

$$\overline{\sigma}_\phi(s_\phi) = \begin{cases} Bel(\Gamma_\phi) & |Bel(\Gamma_\phi)| = m_\phi \quad \text{(A)} \\ Bel(\Gamma_\phi) \cup \{\alpha^1_\phi, \ldots, \alpha^{k_\phi}_\phi\} & \text{otherwise} \quad \text{(B)} \end{cases}$$

Note that in both case (A) and (B), the size of the epistemic state is exactly m_ϕ: $|\overline{\sigma}_\phi(s_\phi)| = m_\phi$.

Lemma 4 *(Truth Lemma) For each $\psi \in Subf(\phi)$,*

$$M_\phi, s_\phi \models \psi \;\Leftrightarrow\; \psi \in \Gamma_\phi$$

Proof. The proof is by induction over the structure of ψ. The case when p is a propositional variable is immediate. When $\psi = B\alpha$, $\alpha \neq \alpha^i_\phi$ for all $i \in [1, k_\phi]$ since $B\alpha \in Subf(\phi)$, so that case is also immediate. Let $\psi = min(n)$. $M_\phi, s_\phi \models \psi$ iff $m_\phi \geq n$. For the direction to the right, $min(m_\phi) \in \Gamma_\phi$, so $min(n) \in \Gamma_\phi$ for any n such that $m_\phi \geq n$ by **MIN1** (and the fact that $min(n) \in Cl(\phi)$). For the direction to the left, if $min(n) \in \Gamma_\phi$, then $n \leq m_\phi$ immediately by definition of m_ϕ. The inductive step (negation and conjunction) is straightforward. $\Box$

Define max_ϕ as the maximum of $|\{B\alpha : B\alpha \in Subf(\phi)\}|$ and $max(m : min(m) \in Subf(\phi))$. The following Lemma, showing that every satisfiable ϕ is satisfied in a model of bounded size – particularly in one where the size of the epistemic state is no greater than max_ϕ, follows immediately (it is easy to see that $m_\phi \leq max_\phi$):

Lemma 5 *Any $\mathcal{S}$-consistent formula ϕ is satisfied in a state in a model $M = (\{s\}, \sigma)$ where $|\overline{\sigma}(s)| \leq \max_\phi$ and $|\sigma(s) \cap \mathcal{P}| \leq |At(\phi) \cap \mathcal{P}|$.*

The following theorem follows immediately from Lemmata 3 and 5.

Theorem 2 *$\mathcal{S}$ is sound and weakly complete with respect to $\mathcal{M}$.*

Furthermore, since Lemma 5 shows satisfiability in a model with a *finite* epistemic state, the following also holds.

Theorem 3 *$\mathcal{S}$ is sound and weakly complete with respect to $\mathcal{M}_{fin}$.*

We now discuss axiomatisation of the class $\mathcal{M}^n$. Let $n \in \mathbb{N}$ be fixed. Define $\mathcal{S}^n$ to be $\mathcal{S}$ extended with the axiom $max(n)$.

Theorem 4 *$\mathcal{S}^n$ is sound and weakly complete with respect to $\mathcal{M}^n$.*

Proof. Soundness follows immediately from Lemma 3 and the definition of $\mathcal{S}^n$. For completeness, let ϕ be a $\mathcal{S}^n$-consistent formula, and let $\phi' = \phi \wedge max(n)$. Since ϕ is $\mathcal{S}^n$-consistent, ϕ' is $\mathcal{S}$-consistent, so it is satisfied in a state with epistemic state of size no greater than $max_{\phi'}$. It must be the case that $\neg min(n+1) \in \Gamma_{\phi'}$

(otherwise $min(n+1) \in \Gamma_{\phi'}$, since $min(n+1) \in Cl(\phi')$, and thus $\Gamma_{\phi'}$ would be inconsistent). If $max_{\phi'} \geq n+1$, $min(n+1) \in \Gamma_{\phi'}$ by **MIN1** and the fact that $min(max_{\phi'}) \in \Gamma_{\phi'}$ which is a contradiction, so $max_{\phi'} < n+1$. Thus, ϕ', and therefore also ϕ, are satisfied in a state where the epistemic state is no greater than n. □

4.1 Complexity

The *satisfiability problem* for $\mathcal{S}$ is the problem of determining, given a formula ϕ, whether there exists a structure M with a state s such that $M, s \models \phi$ (we here abuse the terminology somewhat and use $\mathcal{S}$ to denote not only the the axiomatic system but also the logic of $\mathcal{M}$).

To determine the complexity of the satisfiability problem precisely, we need to decide on the encoding of formulas and models. Consider some standard encoding of propositional formulas as strings, for example where propositional variables are encoded by a single symbol p followed by an index of the variable in binary (see, e.g., [7]). We can encode $min(n)$ in a similar way, as a single symbol m followed by the representation of n in binary. Pointed models, which are essentially just assignments to atoms, can be encoded either as a concatenation of encodings of the atoms which are true under the assignment, or, if it is an assignment to a finite ordered set of m atoms, as a binary string of length m. Note that in either case the length of the encoding of a satisfying model for a formula ϕ guaranteed by Lemma 5 may be exponential in $|\phi|$. For example, if $\phi = min(n)$, then $|\phi| = 1 + log_2(n)$ and the length of the encoding of the satisfying assignment is n (under the bit string approach) or $nlog(n)$ (under the list of atoms approach). This is bad news, because from the existence of an exponential satisfying model we can only infer a NEXPTIME upper bound for the complexity of satisfiability. However, we can encode the satisfying model more efficiently, so that the encoding is not exponential in size, but still can be recognised as a model and used to evaluate a formula. Namely, when we are guessing a model M_ϕ for ϕ, with a single state s_ϕ, we will only explicitly represent the assignment to $At(\phi)$ in s_ϕ, which is polynomial in $|\phi|$. Instead of explicitly representing the assignment to 'padding formulae', we will guess the total number of epistemic atoms in $\overline{\sigma}(s)$ and write it down in binary. So the representation of s_ϕ will look as follows: $(\alpha_1, \ldots, \alpha_m, n)$, where $\alpha_1, \ldots, \alpha_m \subseteq At(\phi)$ are the atoms true in s_ϕ, and $n \geq m$ is a binary representation of the size of $\overline{\sigma}(s)$. Clearly, this is polynomial in $|\phi|$, and this information is sufficient to evaluate ϕ, which can be done in polynomial time.

Theorem 5 *The satisfaction problem for $\mathcal{S}$ is NP-complete.*

Proof. We know that ϕ is satisfiable iff it is satisfiable in a model of size at most max_ϕ, by Lemma 5. We guess this model M_ϕ, and represent it as a string linear in the size of $|\phi|$, by explicitly representing truth values for atoms in $At(\phi)$, and guessing the total number of true epistemic atoms, written in binary. We can check whether ϕ is satisfied by the model in polynomial time. NP-hardness follows from the fact that the logic extends propositional logic. □

5 Adding State Transitions

In this section we consider the dynamics of the agent's beliefs. We extend syntactic structures to include state transitions, and the languages $\mathcal{L}$ and $\mathcal{L}_{min}$ to include a 'next state' modality $\diamond$, to obtain the languages $\mathcal{L}^\diamond$ and $\mathcal{L}^\diamond_{min}$, respectively; a formal definition is given below. Extending the logic this way allows us to describe how the agent's beliefs change. For example, if the agent knows an inference rule, then it can apply it to beliefs in its current state to derive new beliefs which are added to the next epistemic state. To use an example from [12], suppose the agent knows $a = b$ and $b = c$, and is capable of reasoning about equality. Unlike in [12], we do not assume that in this case the agent can derive $a = c$ instantaneously. Rather, we interpret 'being able to reason about equality' as being able to derive new statements about equality which follow from the current beliefs, in some future state. In particular, the agent should be able to reach a state where it believes $a = c$. This property is expressible by the following $\mathcal{L}^\diamond$ formula, in which $\diamond$ stands for 'there exists a successor state where…':

$$B(a = b) \wedge B(b = c) \rightarrow \diamond B(a = c)$$

We may also want to express that the agent's knowledge grows monotonically, which can be done by adding an axiom schema

$$B\phi \rightarrow \Box B\phi$$

(where $\Box$ is the dual of $\diamond$, meaning 'in every successor state…').

The expressive power we get in $\mathcal{L}^\diamond$ is similar to, for example, step logics [10]: we can describe how the beliefs of an agent capable of applying certain inference rules increase over time. Imposing some simple conditions would guarantee that the set of beliefs remains finite. One of such possible conditions is requiring that each state transition corresponds to deriving exactly one new formula; this is expressible by a schema:

$$\diamond(B\phi \wedge B\psi) \rightarrow (B\phi \vee B\psi)$$

(if in the next state the agent believes two formulae, then at least one of those formulae is already believed in the current state). Note that the condition that each successor state has at most one extra formula can be more elegantly expressed in $\mathcal{L}^\diamond_{min}$:

$$max(n) \rightarrow \Box max(n+1)$$

Another useful property, namely that each state transition adds some new belief, is not expressible in $\mathcal{L}^\diamond$ at all, unless we introduce existential quantification over formulae or infinite disjunctions; however in $\mathcal{L}^\diamond_{min}$ we can say

$$min(n) \rightarrow \Box min(n+1)$$

If we do take the size bound on the agent's epistemic state seriously, however, the combination of monotonicity and ability to derive new formulae becomes problematic. A natural restriction on monotonicity in this case would be to say: if the set of beliefs is less than the maximal size, then monotonicity holds; otherwise, the agent can still derive a new formula, but at the expense of 'overwriting' one of the old beliefs. This assumption is made, e.g., in [5], which studies logics for bounded memory reasoners. The property of bounded monotonicity (if the cardinality of the set of beliefs is less than n, then all beliefs persist into the next state) can be expressed in $\mathcal{L}^\diamond_{min}$ as

$$max(n-1) \wedge B\phi \rightarrow \Box B\phi$$

Hopefully, the examples above have given the reader a flavour of the kind of properties we would like to express in $\mathcal{L}^\diamond_{min}$. Now we proceed to give formal definitions of the language and the structures.

Let the language $\mathcal{L}^\diamond_{min}$ is defined by the following grammar:

$$\phi ::= p \in \mathcal{P} \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid B\phi : \phi \in \mathcal{L} \mid min(n) : n \in \mathbb{N} \mid \diamond\phi$$

Define $\Box\phi$ as $\neg\Diamond\neg\phi$. A *syntactic relational structure* is a triple $M = (S, \sigma, R)$ where (S, σ) is a syntactic structure and $R \subseteq S \times S$ a relation over the states. The class of all syntactic relational structures is denoted $\mathcal{M}^\Diamond$. The satisfaction relation $M, s \models \phi$ is defined as before; the extra clause for $\Diamond\phi$ is standard in modal logic:

$$M, s \models \Diamond\phi \quad \Leftrightarrow \quad \exists s'(R(s, s') \text{ and } M, s' \models \phi)$$

Let $\mathcal{K}_{min}$ be the logical system obtained by adding to $\mathcal{S}$ the axiom schema **K**: $\Box(\phi \to \psi) \to (\Box\phi \to \Box\psi)$ and the necessitation rule **N**: $\vdash \phi \Rightarrow \vdash \Box\phi$.

Theorem 6 *$\mathcal{K}_{min}$ is sound and weakly complete wrt. $\mathcal{M}^\Diamond$.*

Proof. The proof consists of minor modifications of the constructions and proof in Section 4. The construction of a satisfying model $M \in \mathcal{M}^\Diamond$ for a $\mathcal{K}_{min}$-consistent formula ϕ uses a method described in [7].

Define $Cl(\phi)$ as in Section 4. Construct $M = (S, \sigma, R)$ as follows. Let S be the set of ($Cl(\phi)$-) maximal consistent subsets of $Cl(\phi)$. In Section 4 it was enough to define a satisfying assignment in a single state s_ϕ corresponding to the maximal consistent set Γ_ϕ containing ϕ; here we must repeat that exercise for *any* such set. For each $s \in S$, the assignment $\sigma(s)$ is defined from s in exactly the same way as $\sigma_\phi(s_\phi)$ was defined from Γ_ϕ in Section 4. Note that we do not include the 'padding' formulae $B\alpha_1, \ldots, B\alpha_k$ in the state s; they are just a technical device we use to define the assignment. In particular, the statement of the Truth Lemma is restricted to the subformulas of ϕ.

Finally, let $R(s, t)$ hold if, and only if, $\phi_s \wedge \Diamond\phi_t$ is $\mathcal{K}_{min}$-consistent, where ϕ_s (ϕ_t) is the conjunction of formulae in $\sigma(s)$ ($\sigma(t)$). The proof of the Truth lemma is standard, see e.g. [7]. □

Since the model we have constructed is exponential in length of the encoding of a formula ϕ, the theorem above implies that $\mathcal{K}_{min}$ is decidable in NEXPTIME. However, it is possible to give a tighter upper bound, namely PSPACE:

Theorem 7 *The problem of whether a formula $\phi \in \mathcal{L}^\Diamond_{min}$ is satisfied in a model in $\mathcal{M}^\Diamond$ is PSPACE complete.*

Proof. It is easy to show that every satisfiable formula ϕ has a tree model of polynomial depth (the proof is the same as for basic modal logic K, see for example [7]). Each state in the model has polynomial size, as we have shown in Lemma 5. Hence, each branch of the satisfying model can be encoded as a string whose length is polynomial in the length of the encoding of ϕ. The *Witness* algorithm given in [7] which essentially builds a tableaux for ϕ one branch at a time, can be easily adapted to check for satisfiability in $\mathcal{M}^\Diamond$. For PSPACE-hardness, observe that the satisfiability problem for modal logic K can be reduced to the satisfiability problem for $\mathcal{K}_{min}$, and the former is PSPACE-complete [7]. □

6 Conclusions

We have modeled the beliefs of an agent with the help of syntactic assignments, an approach also used by several others [14, 11, 6, 4] in order to model properties of reasoners which are difficult or impossible to model with traditional modal epistemic logics. The same model of belief we have used in this paper, was recently used [3] to give a semantics to Ho Ngoc Duc's [8] logic of rational, but not logically omniscient, agents. While many approaches have been suggested to alleviate the logical omniscience problem [13] the syntactic approach can be seen as the most general one. The results in this paper should be readily applicable to other logics with a syntactic component. Of particular interest for future work would be to apply them to *the logic of general awareness* [11].

In the language of syntactic epistemic logic with $min(n)$ operators, we can express many interesting properties of agents, such as bounded memory, and knowing exactly or at most the given set of formulae and nothing else. We introduce several natural logics in this language, and show that they have sound and complete axiomatisations, and that their decidability problem is in the same class as their non-epistemic counterparts.

Acknowledgements Thomas Ågotnes' work was supported by NFR grant 166525/V30. The initial phases of this work were carried out while Thomas Agotnes was visiting the intelligent agents lab in the School of Computer Science at the University of Nottingham. Natasha Alechina would like to thank the Royal Society for sponsoring related work on logics for resource-bounded agents, Isaac Newton Institute for hospitality during work on this paper, and Andrei Krokhin for discussing complexity.

REFERENCES

[1] T. Ågotnes, *A logic of Finite Syntactic Epistemic States*, Ph.D. thesis, Department of Informatics, University of Bergen, Norway, April 2004.

[2] T. Ågotnes and N. Alechina, 'The dynamics of syntactic knowledge', Technical Report 304, Dept. of Informatics, Univ. of Bergen, Norway, (2005).

[3] T. Ågotnes and N. Alechina, 'Semantics for dynamic syntactic epistemic logics', in *Principles of Knowledge Representation and Reasoning: Proceedings of the Tenth International Conference (KR'06)*. To appear.

[4] T. Ågotnes and M. Walicki, 'Strongly complete axiomatizations of "knowing at most" in standard syntactic assignments', in *Sixth International Workshop, CLIMA VI, City University London, UK, June 27-29, 2005. Revised Selected and Invited Papers.*, eds., F. Toni and P. Torroni, volume 3900 of *Lecture Notes in Artificial Intelligence (LNAI)*, pp. 57–76. Heidelberg: Springer-Verlag, (2006).

[5] A. Albore, N. Alechina, P. Bertoli, C. Ghidini, B. Logan, and L. Serafini, 'Model-checking memory requirements of resource-bounded reasoners', in *Proceedings of the Twenty First National Conference on Artificial Intelligence (AAAI)*. AAAI Press, (2006). To appear.

[6] N. Alechina, B. Logan, and M. Whitsey, 'A complete and decidable logic for resource-bounded agents', in *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2004)*, pp. 606–613. ACM Press (2004).

[7] P. Blackburn, M. de Rijke, and Y. Venema, *Modal Logic*, volume 53 of *Cambridge Tracts in Theoretical Computer Science*, Cambridge University Press, Cambridge University Press, 2001.

[8] Ho Ngoc Duc, 'Reasoning about rational, but not logically omniscient, agents', *Journal of Logic and Computation*, **7**(5), 633–648, (1997).

[9] J. Elgot-Drapkin, M. Miller, and D. Perlis, 'Memory, reason and time: the Step-Logic approach', in *Philosophy and AI: Essays at the Interface*, 79–103, MIT Press, Cambridge, Mass., (1991).

[10] J. Elgot-Drapkin and D. Perlis, 'Reasoning situated in time I: Basic concepts', *Journal of Experimental and Theoretical Artificial Intelligence*, **2**, 75–98, (1990).

[11] R. Fagin and J. Y. Halpern, 'Belief, awareness and limited reasoning', *Artificial Intelligence*, **34**, 39–76, (1988).

[12] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi, *Reasoning About Knowledge*, The MIT Press, Cambridge, Massachusetts, 1995.

[13] J. Hintikka, 'Impossible possible worlds vindicated', *Journal of Philosophical Logic*, **4**, 475–484, (1975).

[14] K. Konolige, *A Deduction Model of Belief*, Morgan Kaufmann Publishers, Los Altos, California, 1986.

[15] H. J. Levesque, 'All I know: a study in autoepistemic logic', *Artificial Intelligence*, **42**, 263–309, (1990).

[16] I. Németi, 'Fine-structure analysis of first-order logic', in *Arrow logic and multi-modal logic*, eds., M. Masuch and L. Pólos, 221 – 247, CSLI Publications, Stanford, (1996).