A Logic for Reasoning about Agents with Finite Explicit Knowledge

Thomas Ågotnes Michal Walicki
Department of Informatics, University of Bergen
PB. 7800, N-5020 Bergen, Norway
agotnes@ii.uib.no michal@ii.uib.no

Abstract. It is widely accepted that there is a need for theories describing what agents explicitly know and can act upon, as opposed to what they know only implicitly. Most approaches to modeling explicit knowledge describe rules governing closure conditions on explicit knowledge. Our approach is different; we model *static* knowledge at a point in time and assume that knowledge is represented *syntactically* rather than as propositions. Another concept which has been suggested is called *only knowing*, and describes all that an agent knows. We argue that a proper logic for reasoning about knowledge must combine this concept with the concept of explicit knowledge, and present such a logic. The logic is weakly complete with respect to a simple and natural semantics; we present a characterization of sets of formulae, called *finitary theories*, for which we also have completeness when used as premises and thus can be used to extend the logic with e.g. epistemic properties. Incorporating axioms which describe purely epistemic properties corresponds to simply removing states from the set of legal epistemic states for each agent. An interesting property of our semantic model, if we require that knowledge must be true, is an incompleteness of knowledge: it is impossible to know "all I know".

1 Finite Explicit Knowledge

It is widely accepted that there is a need for theories describing what agents *actually* or *explicitly* know and can act upon, as opposed to what they *implicitly* know, i.e. what follows logically from their explicit knowledge. This distinction is closely related to the logical omniscience problem [9]; in classical modal epistemic logics agents know all the logical consequences of their knowledge – a description of implicit knowledge. Levesque has proposed to formalize the rules governing explicit knowledge [13], and his approach has been extended by or inspired others [10, 4]. Our model of explicit knowledge in this paper takes a different approach. First, instead of describing closure conditions on explicit knowledge, we assume that explicit knowledge *has* been obtained, and we construct a logic for reasoning about *static* explicit knowledge in a group of agents. A framework for reasoning about static knowledge is useful for analyzing the knowledge in a group of agents at an instant in time (or a time span when no epistemic changes are made), for example between computing deductions. Second, we consider ascribing knowledge of propositions to agents in multiagent systems to be unrealistic. Instead, we assume that the agents posses and process syntactical objects. (Of course, neither of these ideas are new; we briefly discuss related work in the last section).

The need for reasoning about explicit knowledge is illustrated by the following example. Consider an agent a sending an encrypted version m_e of a secret message m to agent c through

a public channel, that it is possible to decipher the message using two (large) prime numbers n_1 and n_2, and that the product $n = n_1 n_2$ is publicly known. Particularly, m_e and n are known by agent b. If we use the "implicit" knowledge concept, the sentence

$$\text{"agent } b \text{ knows } m\text{"} \tag{1}$$

could be derived from the sentences

$$\text{"agent } b \text{ knows } m_e\text{"} \tag{2}$$
$$\text{"agent } b \text{ knows } n\text{"} \tag{3}$$

assuming agent b knows the rules of arithmetic, since the values of n_1 and n_2 follows logically from the value of n.

However, even if we use the "explicit" knowledge concept, the sentence

$$\text{"agent } b \text{ does not know } m\text{"} \tag{4}$$

does *not* follow logically from sentences (2) and (3). Information about what the agent explicitly knows, does not make us able to deduce what he (explicitly) does *not* know. For example, agent b could have gotten to know m even before the message was sent. But if we add the sentence

$$\text{"sentences (2) and (3) describe all that is known by } b\text{"} \tag{5}$$

then sentence 4 follows from 2, 3 and 5. The concept of *only knowing* has been suggested to capture "all an agent knows", but most approaches are in the context of *implicit* knowledge. As illustrated by our example, using the "implicit" knowledge concept does not allow us to deduce sentence 4 from 2, 3 and 5, because m is implicitly included in what is "only known" since it follows logically from other known facts. Thus, a proper logic for reasoning about knowledge combines reasoning about explicit knowledge with the concept of only knowing.

We construct a logic for explicit knowledge. We are not concerned with how the agents obtain their explicit knowledge, nor in the relation of this knowledge to reality (an agent can e.g. know false facts or contradictions). Particularly, we do not in general assume anything about the completeness or consistency of the deliberation mechanism, i.e. we do not assume any closure condition of explicit knowledge.

We want to capture the concept "all an agent explicitly knows". Everything an agent *implicitly* knows may be possible to describe by one single formula α; the agent implicitly knows everything that is logically entailed by the formula. If the operator K_I denotes the concept of implicit knowledge, we can then express the agent's knowledge by $K_I\alpha$. Everything an agent *explicitly* knows, however, cannot be described by a single formula (if it knows more than one formula), because there is no closure condition for explicit knowledge. If the operator K_E denotes explicit knowledge, the formulae $K_E\alpha$ and $K_E\beta$ say that the agent knows α and β, but does not say anything about e.g. whether the agent knows the formula $\gamma = (\alpha \wedge \beta)$. We therefore need to express knowledge about *sets* of formulae. We describe the fact that the formulae $\alpha_1, \ldots, \alpha_k$ are all that is explicitly known by agent i by the formula

$$\diamondsuit_i\{\alpha_1, \ldots, \alpha_k\} \tag{6}$$

When we consider static sets of explicit knowledge, there are a number of aspects we can reason about, for example "the agent knows more than X" or "the agent knows less than X". We formalize the fact that *at least* the formulae $\alpha_1, \ldots, \alpha_k$ are known by i by the formula

$$\triangle_i\{\alpha_1, \ldots, \alpha_k\} \tag{7}$$

This formula says that agent i knows each α_i, but it may know more. We use the formula

$$\triangledown_i\{\alpha_1, \ldots, \alpha_k\} \tag{8}$$

to express the fact that agent i knows *at most* $\alpha_1, \ldots, \alpha_k$, i.e. that all he knows is included in the set but he may know less. Evidently

$$\diamondsuit_i X \Leftrightarrow \triangle_i X \wedge \triangledown_i X$$

1.1 Semantic Assumptions

We make the following semantic assumptions: **a)** Each agent represents knowledge explicitly and syntactically, in a propositional language, parameterized by a countable set of primitive propositions Θ, expressing facts about the world and about agents' knowledge. **b)** An agent can know only a *finite number* of facts. Like logical omniscience, knowing (having computed) an infinite number of explicit facts at the same time is an impossibility in real agents. **c)** An agent can represent propositions about knowledge of finite *sets* of facts. This is necessary to allow the agents to reason about their own and others' knowledge in the same way as *we* reason about the agents' knowledge. Formally, OL is the set of all facts which can be known by the agents:

Definition 1 (OL). OL is the least set such that:

- $OL_0 = \Theta$
- If $X \in \wp^{fin}(OL_k)$ then $\triangle_i X, \triangledown_i X \in OL_{k+1}$
- If $\alpha, \beta \in OL_k$ then $\neg\alpha, (\alpha \wedge \beta) \in OL_{k+1}$
- $OL = \bigcup_{k=0}^{\infty} OL_k$

2 Language

A language for reasoning about explicit knowledge over OL needs a representation of the elements in OL – which again require a representation of sets of such elements. In the next definition, the *agent language* and the *term language* are defined by mutual recursion. The latter is simply a notation for sets of the former, while the former includes such sets in its definition.

Definition 2 (TL, AL). TL and AL are the least sets such that

- $\Theta \subseteq AL$
- If $T \in TL$ then $\triangle_i T, \triangledown_i T \in AL$

- If $\alpha, \beta \in AL$ then $\neg\alpha, (\alpha \wedge \beta) \in AL$
- If $\alpha_1, \ldots, \alpha_k \in AL$ then $\underline{\{}\alpha_1\underline{,}\ldots\underline{,}\alpha_k\underline{\}} \in TL$
- If $T, U \in TL$ then $(T \sqcup U), (T \sqcap U) \in TL$

Each element in OL is represented by infinitely many formulae in AL; the AL-formulae $\triangle_i\underline{\{}\ \bigtriangledown_j\ \underline{\{}p\underline{,}r\underline{\}}\underline{,}q\underline{\}}$ and $\triangle_i\underline{\{}q\underline{,}\ \bigtriangledown_j\ \underline{\{}p\underline{,}r\underline{\}\}}$ and $\triangle_i\underline{\{}\ \bigtriangledown_j\ (\underline{\{}p\underline{,}q\underline{,}r\underline{\}}\sqcap\underline{\{}p\underline{,}s\underline{,}r\underline{\}}\underline{,}q)\underline{\}}$ all represent the element$\triangle_i\{\bigtriangledown_j\{p, r\}, q\} \in OL$. We will henceforth drop the underlined notation for AL formulae to increase readability. Terms without the connectives $\sqcup, \sqcap$ are called *basic* terms.

Finally, we introduce the *epistemic language* EL, the logical language[1] we will use to reason about explicit knowledge. Since this is exactly what we have assumed the agents themselves do, this meta-language is identical to the agent language – only extended by formulae to express relations between sets.

Definition 3 (EL). Given a set of primitive formulae Θ, the epistemic language EL is the least set such that:

- $AL \subseteq EL$
- If $T, U \in TL$ then $(T \doteq U) \in EL$
- If $\phi, \psi \in EL$ then $\neg\alpha, (\alpha \wedge \beta) \in EL$

We use the propositional connectives $\vee, \rightarrow, \leftrightarrow$ as abbreviations with the usual meaning, in addition to $\diamondsuit_i T$ for $(\triangle_i T \wedge \bigtriangledown_i T)$ and $T \preceq U$ for $T \sqcup U \doteq U$. $p, q, \ldots$ denote members in Θ, $\alpha, \beta, \ldots$ AL-formulae, $T, U, \ldots$ terms in TL and $\phi, \psi, \ldots$ EL-formulae. $\triangle_i, \bigtriangledown_i$ are called epistemic (i-) operators.

3 Semantics

The main semantic idea is to represent each agent as a point in the lattice of finite subsets of OL.

Definition 4 (Knowledge Set Structure). A Knowledge Set Structure (KSS) is an $n+1$-tuple

$$M = (s_1, \ldots, s_n, \pi)$$

where $s_i \in \wp^{fin}(OL)$ and $\pi : \Theta \rightarrow \{\mathbf{true}, \mathbf{false}\}$ is a truth assignment. s_i is called the *epistemic state* of agent i. The set of all knowledge set structures is denoted $\mathcal{M}$.

In order to define satisfaction of an EL formula in a KSS, we must first define the interpretation $[T] \in \wp^{fin}(OL)$ of a term $T \in TL$.

Definition 5 (Interpretation of Terms). The interpretations $[\alpha] \in OL$ and $[T] \in \wp^{fin}(OL)$ of a formula $\alpha \in AL$ and of a term $T \in TL$, respectively, are defined by mutual recursion:

[1] Formally the languages AL, TL and EL are defined over the alphabet $\Theta \cup \{\triangle_i, \bigtriangledown_i, \neg, \wedge, (,), \doteq\ \underline{\{}, \underline{\}}, \underline{,}\}$, and EL is defined over AL and TL which again are defined by mutual recursion. Ågotnes and Walicki [1] show that these sets are well defined, by constructing a least fixed point of the recursive definition.

- $[p] = p$ for $p \in \Theta$
- $[\neg\alpha] = \neg[\alpha]$, for $\alpha \in AL$
- $[\alpha_1 \wedge \alpha_2] = [\alpha_1] \wedge [\alpha_1]$, for $\alpha_1, \alpha_2 \in AL$
- $[\triangle_i T] = \triangle_i[T]$, for $T \in TL$
- $[\bigtriangledown_i T] = \bigtriangledown_i[T]$, for $T \in TL$
- $[\underline{\{}\alpha_1\underline{,}\ldots\underline{,}\alpha_n\underline{\}}] = \{[\alpha_1], \ldots, [\alpha_n]\}$
- $[S \sqcup T] = [S] \cup [T]$
- $[S \sqcap T] = [S] \cap [T]$

Definition 6 (Satisfaction). Satisfaction of an *EL*-formula ϕ in a KSS $M = (s_1, \ldots, s_n, \pi) \in \mathcal{M}$, written $M \models \phi$, is defined as follows:

$$\begin{aligned}
M \models p &\Leftrightarrow \pi(p) = \mathbf{true} \\
M \models \neg\phi &\Leftrightarrow M \not\models \phi \\
M \models (\phi \wedge \psi) &\Leftrightarrow M \models \phi \text{ and } M \models \psi \\
M \models \triangle_i T &\Leftrightarrow [T] \subseteq s_i \\
M \models \bigtriangledown_i T &\Leftrightarrow s_i \subseteq [T] \\
M \models T \doteq U &\Leftrightarrow [T] = [U]
\end{aligned}$$

The truth conditions for the derived operators are easily seen to be:

$$\begin{aligned}
M \models (\phi \vee \psi) &\Leftrightarrow M \models \phi \text{ or } M \models \psi \\
M \models (\phi \rightarrow \psi) &\Leftrightarrow M \not\models \phi \text{ or } M \models \psi \\
M \models \Diamond_i T &\Leftrightarrow [T] = s_i \\
M \models T \preceq U &\Leftrightarrow [T] \subseteq [U]
\end{aligned}$$

As an example, consider again the encrypted message case. If we assume that m_e, m, n are primitive propositions, sentences 2, 3, 5 and 4 can be expressed as

$$\triangle_b\{m_e\} \tag{9}$$
$$\triangle_b\{n\} \tag{10}$$
$$\bigtriangledown_b\{m_e, n\} \tag{11}$$
$$\neg \triangle_b \{m\} \tag{12}$$

It is easy to see that (12) is a logical consequence of (9) – (11). If the case were that we did not know whether agent b has more information than the encrypted message and the product, we would remove formula (11). Now, (12) does not follow from (9) and (10).

4 A Logical System

Let the *term formulae* be the subset of EL consisting of propositional combinations of term equalities $T \doteq U$. In order to define a logical system for *EL,* we first define a *term calculus* – a logical system for the term formulae.

4.1 Term Calculus

Definition 7 (TC). The term calculus TC is the logical system for the language of term formulae consisting of the following axiom schemata and rule:

All substitution instances of tautologies of prop. calculus		(**Prop**)
$T \doteq T$	eq. (refl.)	(T1)
$T \doteq U \rightarrow U \doteq T$	eq. (symm.)	(T2)
$T \doteq U \wedge U \doteq V \rightarrow T \doteq V$	eq. (trans.)	(T3)
$T \doteq U \wedge S \doteq V \rightarrow S \sqcup T \doteq V \sqcup U$	congruence	(T4)
$T \doteq U \wedge S \doteq V \rightarrow S \sqcap T \doteq V \sqcap U$		(T5)
$T \sqcup U \doteq U \sqcup T$	commutativity	(T6)
$T \sqcap U \doteq U \sqcap T$		(T7)
$(T \sqcup U) \sqcup V \doteq T \sqcup (U \sqcup V)$	associativity	(T8)
$(T \sqcap U) \sqcap V \doteq T \sqcap (U \sqcap V)$		(T9)
$T \sqcup (T \sqcap U) \doteq T$	absorbtion	(T10)
$T \sqcap (T \sqcup U) \doteq T$		(T11)
$T \sqcap (U \sqcup V) \doteq (T \sqcap U) \sqcup (T \sqcap V)$	distributivity	(T12)
$\{\alpha_1, \ldots, \alpha_n\} \doteq \{\alpha_1\} \sqcup \cdots \sqcup \{\alpha_n\}$		(L1)
$\{\alpha\} \doteq \{\beta\} \rightarrow \{\alpha\} \sqcap \{\beta\} \doteq \{\alpha\}$		(L2)
$\neg(\{\alpha\} \doteq \{\beta\}) \rightarrow \{\alpha\} \sqcap \{\beta\} \doteq \emptyset$		(L3)
$\{\alpha_1, \ldots \alpha_k\} \preceq \{\alpha'_1, \ldots, \alpha'_m\} \rightarrow$ $(\{\alpha_1\} \doteq \{\alpha'_1\} \vee \cdots \vee \{\alpha_1\} \doteq \{\alpha'_m\}) \wedge$ $\vdots$ $(\{\alpha_k\} \doteq \{\alpha'_1\} \vee \cdots \vee \{\alpha_k\} \doteq \{\alpha'_m\}))$		N1
$\neg(\{p\} \doteq \{\alpha\})$ if $\alpha \neq p$		N2
$T \doteq S \leftrightarrow \{\triangle_i T\} \doteq \{\triangle_i S\}$		N3
$\neg(\{\triangle_i T\} \doteq \{\alpha\})$ if $\alpha \neq \triangle_i S$		N4
$T \doteq S \leftrightarrow \{\bigtriangledown_i T\} \doteq \{\bigtriangledown_i S\}$		N5
$\neg(\{\bigtriangledown_i T\} \doteq \{\alpha\})$ if $\alpha \neq \bigtriangledown_i S$		N6
$\{\alpha\} \doteq \{\beta\} \leftrightarrow \{\neg\alpha\} \doteq \{\neg\beta\}$		N7
$\neg(\{\neg\alpha\} \doteq \{\beta\})$ if $\beta \neq \neg\gamma$		N8
$\{\alpha_1\} \doteq \{\beta_1\} \wedge \{\alpha_2\} \doteq \{\beta_2\} \leftrightarrow \{\alpha_1 \wedge \alpha_2\} \doteq \{\beta_1 \wedge \beta_2\}$		N9
$\neg(\{(\alpha_1 \wedge \alpha_2)\} \doteq \{(\beta)\})$ if $\beta \neq (\beta_1 \wedge \beta_2)$		N10
$\dfrac{\vdash \phi, \vdash \phi \rightarrow \psi}{\vdash \psi}$		**MP**

The term calculus has two components. The first is the lattice calculus T1–L3, which is an axiomatization of a power-set lattice. In addition to axiomatizing equivalence in a distributive lattice (T1–T12), we also need to "connect" non-basic terms to the basic terms they denote. For example, we must have that $\{\alpha\} \sqcup \{\beta\} \doteq \{\alpha, \beta\}$. The axioms L1–L3 ensure this.

The second component of the term calculus is an axiomatization of equality and inequality of basic terms, i.e. of agent language formulae, N1–N10. In short, equal formulae differ at most by the the occurrence of (syntactically) different terms which can be proved equal. N2–N10 axiomatize equality and inequality for all possible combinations of singular basic terms. N1 characterizes inequality of non-singular basic terms, in terms of inequality of singular basic terms (note that the other direction of N1 also holds).

Theorem 1. *If ϕ is a term formula, then $\models \phi \Leftrightarrow \vdash_{TC} \phi$*

4.2 Epistemic Calculus

We define a sound and, for finite and a certain class of infinite theories, complete logical system EC for EL.

Definition 8 (EC). The epistemic calculus EC is the logical system for EL extending the term calculus TC with the following axiom schemata and rule:

All substitution instances of tautologies of propositional calculus	**Prop**
$\triangle_i \emptyset$	E1
$(\triangle_i T \wedge \triangle_i U) \rightarrow \triangle_i (T \sqcup U)$	E2
$(\triangledown_i T \wedge \triangledown_i U) \rightarrow \triangledown_i (T \sqcap U)$	E3
$(\triangle_i T \wedge \triangledown_i U) \rightarrow T \preceq U$	E4
$(\triangledown_i (U \sqcup \{\alpha\}) \wedge \neg \triangle_i \{\alpha\}) \rightarrow \triangledown_i U$	E5
$\triangle_i T \wedge U \preceq T \rightarrow \triangle_i U$	**KS**
$\triangledown_i T \wedge T \preceq U \rightarrow \triangledown_i U$	**KG**
$\dfrac{\Gamma \vdash \phi, \Gamma \vdash \phi \rightarrow \psi}{\Gamma \vdash \psi}$	**MP**

Theorem 2 (Soundness). *EC is sound: $\Gamma \vdash \phi \Rightarrow \Gamma \models \phi$*

Theorem 3 (Weak Completeness). *EC is weakly complete: $\models \phi \Rightarrow \vdash \phi$*

We investigate the question of strong completeness in the next section.

5 Modeling Epistemic Properties

In addition to the fundamental properties of our knowledge operators, one may want to model additional epistemic properties of the agents. An example of such a property is that it is impossible to know both a formula and its negation (a contradiction) at the same time. *Semantically*, additional epistemic properties can be modeled by restricting the set of structures. *Syntactically*, they can be modeled by adopting additional axioms. In modal epistemic logic, the correspondence between axioms and semantic constraints has been extensively studied (see e.g. [15]). Particularly, several of the most commonly suggested epistemic axioms correspond to "natural" classes of Kripke structures (such as e.g. "all reflexive Kripke structures").

We now show that axioms in a class which describe purely epistemic properties can be modeled by "natural" semantic constraints. In Sec. 5.2 we investigate at the problem of completeness when we add axioms to EC. In Sec. 5.3 we look at some common epistemic axioms.

5.1 Constructing Model Classes for Epistemic Axioms

Not all formulae in EL should be considered as candidates for describing epistemic properties. One example is $p \rightarrow \triangle_i\{p\}$. This formula does not solely describe the *agent* – it describes a relationship between the agent and the world, i.e. about an agent in a situation. Another example is $\diamondsuit_i\{p\} \rightarrow \diamondsuit_j\{q\}$, which describes a constraint on one agent's belief set contingent on another agent's belief set. Neither of these two formulae describes purely *epistemic* properties of an agent.

Candidates for *epistemic axioms* should therefore a) only refer to epistemic facts and not to external facts and b) only describe one particular agent. In the following definition, EF is the set of epistemic formulae and Ax is the set of candidate epistemic axioms.

Definition 9 (EF, EF^i, Ax).

- $EF \subseteq EL$ is the least set such that

$$\begin{aligned} \text{If } T \in TL \text{ then} \quad & \triangle_i T, \triangledown_i T \in EF \\ \text{If } \phi, \psi \in EF \text{ then} \quad & \neg\phi, (\phi \wedge \psi) \in EF \end{aligned}$$

- $EF^i = \{\phi \in EF : \text{Every epistemic operator in } \phi \text{ is an } i\text{-operator}\}$
- $Ax = \bigcup_{1 \leq i \leq n} EF^i$

We show that extending EC with a set of epistemic axioms $\Phi \subseteq Ax$ corresponds to restricting the set of legal structures by *locally restricting the set of legal epistemic states of each agent*. In other words, we show that we can construct a set S_i^Φ of legal epistemic states for each agent such that

$$mod(\Phi) = \{(s_1^\Phi, \ldots, s_n^\Phi, \pi) : s_i^\Phi \in S_i^\Phi\} \tag{13}$$

where $mod(\Phi)$ is the set of models of Φ. First, we construct a set of legal epistemic states S_i^ϕ for each agent, corresponding to a single formula ϕ:

Definition 10 (S_i^ϕ). For each epistemic axiom $\phi \in EF^i$, S_i^ϕ is constructed by structural induction over ϕ as follows:

$$\begin{aligned} \phi = \triangle_i T: \quad & S_i^\phi = \{X \in \wp^{fin}(OL) : [T] \subseteq X\} \\ \phi = \triangledown_i T: \quad & S_i^\phi = \{X \in \wp^{fin}(OL) : X \subseteq [T]\} \\ \phi = \neg\psi: \quad & S_i^\phi = \wp^{fin}(OL) \backslash S_i^\psi \\ \phi = \psi_1 \wedge \psi_2: \quad & S_i^\phi = S_i^{\psi_1} \cap S_i^{\psi_2} \end{aligned}$$

The epistemic states which are not removed in the construction of S_i^ϕ are the possible states — an agent can be placed in any of these states and will satisfy the epistemic axiom ϕ. It is easy to see that $mod(\phi) = \{(s_1^\phi, \ldots, s_n^\phi, \pi^\phi) \in \mathcal{M} : s_i \in S_i^\phi\}$.

The set of legal epistemic states corresponding to a set $\Phi \subseteq Ax$ is given by: $S_i^\Phi = \bigcap_{\phi \in (\Phi \cap EF^i)} S_i^\phi$

Lemma 1. $mod(\Phi) = \{(s_1^\Phi, \ldots, s_n^\Phi, \pi) : s_i^\Phi \in S_i^\Phi\}$

A constructive definition of the model class of a *general* set $\Phi \subseteq EL$ (possibly including non-epistemic axioms) can be made, but will not necessarily correspond to the removal of illegal epistemic states. Lemma 1 shows that we can model epistemic axioms semantically by requiring that for each epistemic state s_i in a structure, $s_i \in S_i^{\Phi} \subseteq \wp^{fin}(OL)$ – independent of the rest of the structure. Below, we show examples of how the adoption of common epistemic axioms makes certain epistemic states illegal.

5.2 Finitary Theories and Completeness

What happens with completeness when we add a set of axioms to EC? Ideally, we would have strong completeness – EC would be complete with respect to the corresponding model class. Unfortunately, this is inherently impossible for certain sets of axioms. Consider the following set:

$$\mathbf{inf} = \{\triangle_i\{p\} : p \in \Theta\}$$

Clearly, $\mathbf{inf}$ is unsatisfiable since it describes an infinite agent and there are no infinite points. Completeness, i.e. $\mathbf{inf} \models \phi \Rightarrow \mathbf{inf} \vdash \phi$ for all ϕ, requires that $\mathbf{inf}$ must then be inconsistent. However, a proof of inconsistency of $\mathbf{inf}$ would have to use every single formula in $\mathbf{inf}$ and hence be infinitely long. Moreover, it is not only unsatisfiable theories that cause problems with completeness. Consider the following set of premises:

$$\mathbf{4i} = \{\triangle_i\{\alpha\} \rightarrow \triangle_i\{\triangle_i\{\alpha\}\} : \alpha \in AL\}$$

If agent i knows *any* fact, $\mathbf{4i}$ "forces" it to know infinitely many facts — which is impossible. Thus, the fact that agent i does not know anything is a logical consequence of $\mathbf{4i}$: $\mathbf{4i} \models \bigtriangledown_i\emptyset$. However, $\bigtriangledown_i\emptyset$ is not deducible from $\mathbf{4i}$ in EC: $\mathbf{4i} \nvdash \bigtriangledown_i\emptyset$, because such a deduction would involve an infinite number of premises from $\mathbf{4i}$. Moreover, this is a fundamental incompleteness; is is not possible to extend EC with axiom schemata and/or rules with a finite number of antecedents to obtain full completeness. We would need a rule with an infinite number of antecedents, such as:

$$R_f = \frac{\text{for all } T \in TL\text{: } \Gamma \cup \{\bigtriangledown_i T\} \vdash \bot}{\Gamma \vdash \bot}$$

The result of extending EC with R_f is strongly complete for all Γ. Since we obviously do not want to use such infinite rules, we go on to characterize theories for which EC *is* complete.

Definition 11 (Finitary Theory). A theory Γ is *finitary* iff it is consistent and for all ϕ, [2]

$$\begin{gathered}\Gamma \vdash (\bigtriangledown_1 T_1 \wedge \cdots \wedge \bigtriangledown_n T_n) \rightarrow \phi \text{ for all terms } T_1, \ldots, T_n \\ \Downarrow \\ \Gamma \vdash \phi\end{gathered}$$

Theorem 4. *All finite theories are finitary.*

Theorem 5. *If Γ is a finitary theory, then: $\Gamma \models \phi \Rightarrow \Gamma \vdash \phi$.*

[2] In higher order notation: Γ is fi nitary iff it is consistent and $\forall\phi((\forall T_1 \cdots \forall T_n \Gamma \vdash (\bigtriangledown_1 T_1 \wedge \cdots \wedge \bigtriangledown_n T_n) \rightarrow \phi) \Rightarrow \Gamma \vdash \phi)$.

5.3 Incorporating Common Axioms

The most commonly suggested additional axioms in modal epistemic logic, written in our notation ($\triangle_i\{\alpha\}$ "corresponds" to $K_i\alpha$ in the traditional notation) are the following:

$\triangle_i\{(\alpha \to \beta)\} \to (\triangle_i\{\alpha\} \to \triangle_i\{\beta\})$	Distrib.	$\mathbf{K_i}$
$\triangle_i\{\alpha\} \to \neg\triangle_i\{\neg\alpha\}$	Consistency	$\mathbf{D_i}$
$\triangle_i\{\alpha\} \to \alpha$	Knowledge	$\mathbf{T_i}$
$\triangle_i\{\alpha\} \to \triangle_i\{\triangle_i\{\alpha\}\}$	Pos. Introsp.	$\mathbf{4_i}$
$\neg\triangle_i\{\alpha\} \to \triangle_i\{\neg\triangle_i\{\alpha\}\}$	Neg. Introsp.	$\mathbf{5_i}$

Theorem 6. $\mathbf{K_i}$ *and* $\mathbf{T_i}$ *are finitary,* $\mathbf{4_i}$ *and* $\mathbf{5_i}$ *are not.*

This result shows that $\mathbf{K_i}$ and $\mathbf{D_i}$ are "compatible" with the semantic assumptions laid out in the beginning of this abstract, and that $\mathbf{4_i}$ and $\mathbf{5_i}$ are not. Taking $\mathbf{K_i}$ or $\mathbf{D_i}$ as premises corresponds to excluding certain legal epistemic states of each agent, as described in Sec. 5.1. For example, extending EC with $\mathbf{D_i}$ corresponds to removing epistemic states s where there both $\alpha \in s$ and $\neg\alpha \in s$ for any α, individually for each agent, in the sense that the resulting system is sound and complete with respect to the model class just described.

We suspect that $\mathbf{T_i}$ is finitary, but do not have a proof (note that unlike the other four axioms, $\mathbf{4i}$ is not an epistemic axiom).

5.4 An Incompleteness of Knowledge

We show a property of our notion of knowledge. In the following theorem, we adopt the $\mathbf{T_i}$ axioms (knowledge implies truth).

Theorem 7. *For any term* S*:*

$$mod(\mathbf{T_i}) \models \neg\triangle_i\{\bigtriangledown_i S\}$$

Proof. It is easy to see that $\mathbf{T_i}$ is satisfiable in $\mathcal{M}^3$. Let $M \models \mathbf{T_i}$, and assume that $M \models \triangle_i\{\bigtriangledown_i S\}$ for some S, i.e. that $\bigtriangledown_i[S] \in s_i$. By $\mathbf{T_i}$, $M \models \bigtriangledown_i S$ and $s_i \subseteq [S]$. Then $\bigtriangledown_i[S] \in [S]$, which is impossible. □

Theorem 7 can be seen as an epistemic analogue of Gödels incompleteness theorem: there exist true formulae which are impossible to know — more specifically, it is impossible to know that "this is all I know" (if knowledge is truth)[4].

As an example, consider an agent i who knows that it knows at most $\{p\}$, i.e. $\triangle_i\{\bigtriangledown_i\{p\}\}$. Under $\mathbf{T_i}$, this would imply that $\bigtriangledown_i\{p\}$ were true, which is not the case since i also knows $\bigtriangledown_i\{p\}$. Trying to add this latter formula to the set, we get $\triangle_i\{\bigtriangledown_i\{p, \bigtriangledown_i\{p\}\}\}$, which under $\mathbf{T_i}$ implies that $\bigtriangledown_i\{p, \bigtriangledown_i\{p\}\}$ is true — which it clearly is not since i also knows $\bigtriangledown_i\{p, \bigtriangledown_i\{p\}\}$. We see that trying to solve the problem by adding new formulae leads to an infinite regression.

Note that the property (Th. 7) does not arise because we require the agents to be finite. It is a consequence of an impossibility of self-reference of terms: a (finite) term cannot be its own proper subterm.

[3] Consider for example a structure with $s_i = \{p\}$ and $\pi(p) = true$.

[4] A formula like $\triangle_i\{\bigtriangledown_j S\}$, $i \neq j$, is of course fully consistent with $\mathbf{T_i}$; an agent can potentially know all *another* agent knows.

6 Conclusions and Related and Future Work

A proper logic for reasoning about knowledge must combine the concepts of explicit knowledge and only knowing. We have presented a logic which does this. The main semantic idea is to represent each agent as a point in the lattice of finite subsets of all explicit facts — a largely syntactical approach. One of the most interesting results is a characterization of theories which describe axiomatazible finite agents. One property induced by our language and semantics, is that under the axiom that knowledge must be true, it is impossible to know "this is all I know".

There are several approaches to explicit knowledge, only knowing and syntactical models of knowledge in the literature. Levesque [13] proposed to describe the rules governing explicit knowledge. Several [7, 5, 12, 6] have suggested a concept of only knowing, and Levesque [12] formalized it in a logical language. Lakemeyer [11] describes a first-order logic with only knowing, where the agents does not have perfect knowledge about the world (but do have full introspection regarding their own knowledge). None of these approaches take a purely syntactic view like ours. Several *general* approaches to reasoning about knowledge treat knowledge syntactically [2, 14, 8]. Our approach introduces a new concept compared to other syntactical models of knowledge: the $\bigtriangledown_i$-operator. This operator allows us to express in one formula facts which in a language with only a traditional K_i-operator would require infinitely many formulae on the form $\neg K_i\phi$. Furthermore, we can express facts such as $\triangle_i\{\bigtriangledown_j X\}$ which would be impossible with only a K_i-operator because it would require a infinitely long term (in a purely syntactic model of knowledge).

In future work we will present a semantical characterization of finitaryness, as algebraic conditions on the sets of legal epistemic states. We are also working on a model of reasoning in which the agents can reason by moving around in the lattice of finite sets, extending the static model of knowledge with a dynamic dimension.

References

[1] Thomas Ågotnes and Michal Walicki. Only explicitly knowing. Technical Report 224, Dept. of Informatics, Univ. of Bergen, Norway, 2002.

[2] R. A. Eberle. A logic of believing, knowing and inferring. *Synthese*, 26:356–382, 1974.

[3] Ronald Fagin and Joseph Y. Halpern. Belief, awareness and limited reasoning. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 491–501, Los Angeles, CA, 1985.

[4] Ronald Fagin and Joseph Y. Halpern. Belief, awareness and limited reasoning. *Artificial Intelligence*, 34:39–76, 1988. A preliminary version appeared in [3].

[5] Joseph Y. Halpern. A theory of knowledge and ignorance for many agents. *Journal of Logic and Computation*, 7(1):79–108, February 1997.

[6] Joseph Y. Halpern and Gerhard Lakemeyer. Multi-agent only knowing. In Yoav Shoham, editor, *Theoretical Aspects of Rationality and Knowledge: Proceedings of the Sixth Conference (TARK 1996)*, pages 251–265. Morgan Kaufmann, San Francisco, 1996.

[7] Joseph Y. Halpern and Yoram Moses. Towards a theory of knowledge and ignorance. In Krzysztof R. Apt, editor, *Logics and Models of Concurrent Systems*, pages 459–476. Springer-Verlag, Berlin, 1985.

[8] Joseph Y. Halpern and Yoram Moses. Knowledge and common knowledge in a distributed environment. *Journal of the ACM*, 37(3):549–587, July 1990.

[9] J. Hintikka. Impossible possible worlds vindicated. *Journal of Philosophical Logic*, 4:475–484, 1975.

[10] Gerhard Lakemeyer. Tractable meta-reasoning in propositional logics of belief. In John McDermott, editor, *Proceedings of the 10th International Joint Conference on Artificial Intelligence (IJCAI '87)*, pages 401–408, Milan, Italy, August 1987. Morgan Kaufmann.

[11] Gerhard Lakemeyer. Decidable reasoning in first-order knowledge bases with perfect introspection. In William Dietterich, Tom; Swartout, editor, *Proceedings of the 8th National Conference on Artificial Intelligence*, pages 531–537. MIT Press, July 29–August 3 1990.

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

[13] Hector J. Levesque. A logic of implicit and explicit belief. Technical Report 32, Fairchild Laboratory of Artificial Intelligence, Palo Alto, US, 1984.

[14] R. C. Moore and G. Hendrix. Computational models of beliefs and the semantics of belief sentences. Technical Note 187, SRI International, Menlo Park, CA, 1979.

[15] Johan van Benthem. Correspondence theory. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic, Volume II: Extensions of Classical Logic*, volume 165 of *Synthese Library*, chapter II.4, pages 167–247. D. Reidel Publishing Co., Dordrecht, 1984.