Taming the Complexity of Linear Time BDI Logics

Nils Bulling
Clausthal University of Technology
bulling@in.tu-clausthal.de

Koen V. Hindriks
Delft University of Technology
k.v.hindriks@tudelft.nl

ABSTRACT

Reasoning about the mental states of agents is important in various settings, and has been recognized as vital for teamwork. But the complexity of some of the more well-known agent logics that facilitate reasoning about mental states prohibits the use of these logics in practice. An alternative is to investigate fragments of these logics that have a lower complexity but are still expressive enough for reasoning about the mental states of (other) agents. We explore this alternative and take as our starting point the linear time variant of BDI logic ($\mathbf{BDI_{LTL}}$). We summarize some of the relevant known complexity results for e.g. $\mathbf{LTL}$, $\mathbf{KD45}_n$, and $\mathbf{BDI_{LTL}}$ itself. We present a tableau-based method for establishing complexity bounds, and provide a map of the complexity of (various fragments of) $\mathbf{BDI_{LTL}}$. Finally, we identify a few fragments that may be usefully applied for reasoning about mental states.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent Systems*; I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods—*Modal logic*

General Terms

Theory, Verification

Keywords

reasoning about mental states, linear time BDI logic, satisfiability, complexity

1. INTRODUCTION

In a social context, and more specifically for teamwork, the ability of an agent to reason about other agents has been recognized as vital [7]. In particular, reasoning about ones own and the mental states of other agents is important to be successful in such contexts. Reasoning about the mental states of others is needed to establish joint commitments and joint intentions [17, 4], collaboration and cooperation [17], teamwork [4, 7, 20], and coordination more generally [8].

Cite as: Taming the Complexity of Linear Time BDI Logics, Nils Bulling and Koen V. Hindriks, *Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2011)*, Tumer, Yolum, Sonenberg and Stone (eds.), May, 2–6, 2011, Taipei, Taiwan, pp. 275-282.

One application area is socio-cognitive robotics and human-robot teamwork [15], where modeling of social behavior is needed and it has been recognized that "consideration of the knowledge, abilities, goals, and even feelings of others" is required.

In practice, however, little use is made of logical approaches such as [4, 18, 9] that support reasoning about mental states due to the inherent complexity of the logics. For example, the satisfiability problem for *agent dynamic logic*, a logic closely related to the KARO framework, is in **2EXPTIME** [19] and for the TeamLog framework including group attitudes as well as propositional dynamic logic it is **EXPTIME**-complete [9]. Practical approaches typically do not maintain many of the formal properties of mental attitudes, and, as noted in [11], "the need for high-level logic-based languages capturing the key components of the BDI model remains." This problem has also motivated the work reported in [9]. More generally, the issue is related to "the gap between theory and practice" [21] since complexity bounds at least theoretically determine what can be done in principle to bridge this gap. In practice, agent platforms typically have been restricted to reason *with* the beliefs and goals of a single agent [2, 3] and do not allow reasoning *about* the beliefs and goals of other agents, e.g. having a belief about the goal of another agent. Identifying such extended fragments therefore is important as it may allow the use of logical fragments in practice for *reasoning about other agents*.

One approach to deal with this problem is to study and identify fragments of agent logics that have a lower complexity but still support reasoning about other agents' mental states. There are general techniques for identifying such fragments and reducing the complexity. For example, it is known that restricting the number of propositional atoms used or the depth of modal nesting may reduce the complexity of a modal logic [12]. It is not cognitively plausible either that humans use unlimited depth of reasoning [7].

The problem we explore in this paper is the satisfiability problem of fragments of $\mathbf{BDI_{LTL}}$. We present a tableau-based proof method for a family of $\mathbf{BDI_{LTL}}$. The tableau-method can also be used to analyze the complexity of this family and we show $\mathbf{BDI_{LTL}}$ is in **PSPACE**. We then explore which fragments have a significantly lower complexity than full linear time BDI logic. More specifically, the aim is to identify fragments for which the satisfiability problem is in **NP**. Identifying such fragments is a first step towards establishing reasonable computational performance as typical problem instances may be easier to solve [13], and, as even satisfiability for propositional logic is an **NP**-complete problem we cannot do better without restricting this underlying logic. Our main motivation for doing so is that agents need a tool for reasoning about other agents' mental states in order to coordinate their actions and a logic-based approach seems most suitable. We also briefly informally consider the expressivity of these fragments.

The paper is organized as follows. Section 2 reviews relevant related work. Section 3 briefly introduces **BDI$_{\mathsf{LTL}}$** and discusses some fragments that are promising from a computational point of view. In Section 4 a tableau-based method for proving satisfiability is introduced. This method provides the basis for some of the complexity results for **BDI$_{\mathsf{LTL}}$**. Section 5 then presents the main complexity results for **BDI$_{\mathsf{LTL}}$** and for fragments of the logic. Section 6 informally discusses whether some minimal requirements are met for these fragments to be useful for reasoning about other agents. Finally, Section 7 concludes the paper.

2. RELATED WORK AND RESULTS

Although significant work has been done in isolating fragments of various modal logics including **LTL** [5], logics of knowledge and belief [13], and combinations thereof [6] for which the satisfiability problem is in **NP**, as far as we know, no existing work has identified fragments with similar complexity that allow the combination of *informational* and *motivational* attitude operators with *time*. However, if we want our agents to reason about both the informational as well as the motivational states of other agents we need exactly this combination. We believe that incorporating time is essential to be able to differentiate various types of goals such as achievement and maintenance goals (cf. also Section 6).[1]

Here we build on the work of [18] which introduces (a family of) linear time BDI logic(s) **BDI$_{\mathsf{LTL}}$**. **BDI$_{\mathsf{LTL}}$** provides a logical framework that allows an agent to distinguish between different mental attitudes, i.e. beliefs versus desires/intentions, and between different types of desires/intentions by means of temporal operators. This sets our work apart from [9] where complexity issues of a multi-agent logic called TeamLog are investigated in a setting without time. Before we explore fragments of **BDI$_{\mathsf{LTL}}$** itself, we first review relevant complexity results for linear temporal logic and logics of mental attitudes available in the literature. This will be useful for identifying fragments of **BDI$_{\mathsf{LTL}}$** for which the satisfiability problem is in **NP**. In the remainder we assume that Φ denotes a *set of propositions*.

Normal modal logics. A starting point for our search for a computational logic for reasoning about the mental states of (other) agents is provided by the extensive work on logics of knowledge and belief reported [13]. [13] presents results that show that the complexity of the satisfiability problem for single agent logics of knowledge **S5** and belief **KD45** are **NP**-complete, for multi-agent logics of knowledge **S5$_n$** and belief **KD45$_n$** are **PSPACE**-complete, and extensions with a common knowledge operator are **EXPTIME**-complete.

For conative logics that are used for modeling the motivational attitudes of agents typically the modal logic **KD** is used [21]. The logic **KD** is in between **K** and **S4**, i.e. **K** $\subseteq$ **KD** $\subseteq$ **S4** $=$ **KT4**. According to Ladner's Theorem [1] this means that the satisfiability problem for **KD** is **PSPACE**-complete.

[1]We will sometimes also talk about *goals* if there is no need to differentiate between desires and intentions.

In [9] it is shown that combining the multi-agent logic of belief and the multi-agent conative logic does not increase the complexity of the satisfiability problem which remains **PSPACE**-complete.

Restricted settings for the logics **K**,**K45**,**KD45**,**S5**,**S4**, and their multi-agent versions are considered in [12] and [9]. The main results are that satisfiability checking can be done in *linear-time* for many standard multi-agent extensions of **K** by *bounding the number of propositional atoms and the depth of modal operators*. Here, we will just present the result for **KD45$_n$** which are of interest for our purposes. It is shown that satisfiability checking can be done in *linear time* if the number of propositions and nestings is fixed. The problem remains in this class for the single-agent setting ($n = 1$) even if nestings are not bounded. The problem gets harder if there is no restriction on the number of propositions and only a bound on the number of nestings. In this case satisfiability checking is **NP**-complete. [9] shows that by bounding the modal depth by a constant the satisfiability problem of combinations of multi-agent belief and multi-agent conative logic is also **NP**-complete. Only bounding the number of propositional atoms does not lower complexity. Bounding *both* the number of propositional atoms *and* the depth of modal operators reduces complexity to linear time, however. This is true even when group attitudes such as common belief and collective intentions are added [9]. We use $depth^m(\varphi)$ to denote the *modal depth* of φ (i.e. the number of nested modal operators); e.g. $K_i p \wedge K_i K_j q$ as modal depth 2. (Cf. Section 3.2 for a formal definition.) In the following table we use $md \leq c$, $c \in \mathbb{N}_0$, to denote the restriction to formulae φ with $depth^m(\varphi) \leq c$. We use $|\Phi|$ to denote the number of propositions. The first cell of a row is understood as a constraint, e.g. $|\Phi| \leq c$, $md \leq c'$ characterises the case in which the number of propositions and the modal depth is bounded by some natural numbers c and c', respectively. In the table below we summarize some complexity results of the satisfiability problem which are relevant for our study:

	K45,**KD45**,**S5**	**K**,**KD$_n$**,**K45$_n$**,**KD45$_n$**,**S5$_n$**
no constraints	**NP**-compl.	**PSPACE**-compl.
$\lvert\Phi\rvert \leq c$	linear time	**PSPACE**-compl.
$md \leq c$	**NP**-compl.	**NP**-compl.
$\lvert\Phi\rvert \leq c$, $md \leq c'$	linear time	linear time

Temporal Logics. It is well-known that the complexity of the satisfiability problem for **LTL** is **PSPACE**-complete and for **CTL** **EXPTIME**-complete. In [5] a large number of propositional fragments of **LTL** is considered and complexity results for both model checking and satisfiability are established. Restrictions on the temporal operators that may be used, the number of propositional atoms, and the temporal depth are considered. Satisfiability for a limited number of fragments turns out to be in the class **NP**. This is shown for the following fragments of **LTL**: (1) The "future-only" fragment; (2) The "next-time" fragment; and (3) the fragment of **LTL** that allows no nesting of temporal operators. Satisfiability for the fragment with a fixed number of propositional atoms *and* limited temporal depth (but no restrictions on the temporal operators) can be solved in deterministic logarithmic space **L**. The table below summarises the relevant results. **LTL**($\mathcal{U}$) is used to denote the fragment with $\mathcal{U}$ being the only temporal operator. Similarly to the modal depth, we define the *temporal depth*, $depth^t(\varphi)$, of φ. In this case the number of nested *temporal* operators is considered. $td \leq c$ is also defined and used analogously to md.

	LTL	LTL($\mathcal{U}$)
$\lvert\Phi\rvert \leq c$, $td \leq c'$	**L**	**L**
td=0	**NP**-compl.	**NP**-compl.
$\lvert\Phi\rvert = 1$	**PSPACE**-compl.	**P**
$td = 2$	**PSPACE**-compl.	**PSPACE**-compl.

Logics of Knowledge and Time. It is natural to combine linear time with combinations of other modal operators (for representing mental attitudes) but only work on combining belief/knowledge and time is known to us. In [14] results related to numerous logics of knowledge and time are considered. A result of this work is that agents that do not forget or do not learn greatly increasing the complexity of reasoning about knowledge and time. It is shown that combining the logic of knowledge and linear time results in a logic for which the satisfiability problem is **PSPACE**-complete and the satisfiability problem for the combination of knowledge and branching time is **EXPTIME**-complete.

3. LINEAR TIME BDI LOGIC

In the literature surveyed, the complexity of fragments that combine time, and multi-agent belief and motivational attitudes is not discussed. Of course, closely related work does exist, and some of the more important work has been discussed above. We take the work reported in [18] as our starting point, which introduces a linear time BDI logic and presents decision methods for satisfiability using a tableau-based approach. Below, we first define the language and its semantics and then continue to discuss potentially interesting fragments from a complexity point of view.

3.1 Language and Semantics

The logic $\mathsf{BDI_{LTL}}$ introduced here is based more or less on [18]. We use the same language but extend it with multiple modal operators to be able to represent the mental attitudes of multiple agents. The semantics has also been set up slightly differently, and we use runs and time points as the basis for accessibility relations instead of worlds that are related through states in [18]; our setup is similar to that in [10].

The language of $\mathsf{BDI_{LTL}}$ includes temporal operators, and multiple belief, desire, and intention operators, one for each agent out of a finite set of agents. The temporal operators are the usual linear time temporal operators for next time and the until operator.

Definition 1 (The language of $\mathsf{BDI_{LTL}}$). *Let Φ be a set of propositional atoms with typical element p and $\mathbb{A}\mathrm{gt}$ be a finite set of agents with typical element i:*

$$\varphi \in \mathcal{L} \quad ::= \quad p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi\mathcal{U}\varphi \mid \bigcirc\varphi \mid B_i\varphi \mid D_i\varphi \mid I_i\varphi$$

Models for the language $\mathcal{L}$ of $\mathsf{BDI_{LTL}}$ consist of runs and of indexed relations on these runs used to define the semantics of mental attitudes of agents. Runs are derived from a given set of states and a transition relation on those states and are used to interpret temporal operators.

A *run* r over a set of states Q is an infinite sequence from Q^ω. We use $r[i]$ to denote the ith state on run r, starting from $i = 0$. $r[i, \infty]$ is used to denote the sub-run of r that starts at i. That is, $r[i, \infty] = r[i]r[i + 1] \ldots$. Given a set of states Q and a transition relation $\rightarrow$, the set of all runs induced by $(Q, \rightarrow)$ is denoted by $\mathcal{R}_{(Q,\rightarrow)}$. $\mathcal{R}_{(Q,\rightarrow)}$ thus consists of all runs r for which we have that: $\forall i \in \mathbb{N}_0 : r[i] \rightarrow r[i + 1]$. Finally, the set of *time points* (r, m) given a set $\mathcal{R}$ of runs is defined by: $Points_{\mathcal{R}} = \{(r, m) \mid r \in \mathcal{R}, m \in \mathbb{N}_0\}$.

Definition 2 (Models for $\mathsf{BDI_{LTL}}$). *A model for the language of $\mathsf{BDI_{LTL}}$ is a tuple: $\mathcal{M} = (Q, \rightarrow, \mathcal{R}, \{\mathcal{B}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \{\mathcal{D}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \{\mathcal{I}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \pi)$ where*

- *Q is a non-empty set of states,*
- *$\rightarrow \subseteq Q \times Q$ a serial (temporal) accessibility relation,*
- *$\mathcal{R} \subseteq \mathcal{R}_{(Q,\rightarrow)}$ is a non-empty set of runs,*
- *$\mathcal{B}_i \subseteq Points_{\mathcal{R}} \times Points_{\mathcal{R}}$ is a transitive, serial, and Euclidean belief accessibility relation,*
- *$\mathcal{D}_i \subseteq Points_{\mathcal{R}} \times Points_{\mathcal{R}}$ is a desire accessibility relation,*
- *$\mathcal{I}_i \subseteq Points_{\mathcal{R}} \times Points_{\mathcal{R}}$ is called an intention accessibility relation, and*
- *$\pi : Q \rightarrow \mathcal{P}(\Phi)$ a labelling or valuation function.*

Models for $\mathsf{BDI_{LTL}}$ include the usual restrictions on the type of accessibility relations in the definition of a model. We are also interested in some additional restrictions that may be imposed on models. For example, $\mathcal{B}$ may also be reflexive, to obtain the usual semantics for *knowledge*, and we consider the additional constraint where $\mathcal{D}$ is serial to exclude inconsistent desires. It is well-known that these constraints correspond with particular axiom schema labeled **K**, **T**, **D**, **4**, **5**. This gives rise to a family of BDI logics and we introduce some notation to refer to different variants. We use $\mathsf{BDI_{LTL}}$ to refer to $\mathsf{BDI_{LTL}^{KD45,K,KD}}$, i.e. the logic with **KD45** belief operators, **K** desire operators, and **KD** intention operators. Alternatively, by varying the restrictions on the accessibility relations, we obtain, for example, the logic $\mathsf{BDI_{LTL}^{S5,KD,KD}}$ which combines knowledge, consistent desires, and intentions. In the following we define $L := \{$**K**, **KD**, **KD45**, **S4**, **S5** $\}$ and often write $\mathsf{BDI_{LTL}^{X,Y,Z}}$ where we implicitly assume that $\mathbf{X}, \mathbf{Y}, \mathbf{Z} \in L$.

It is usual to consider a range of additional constraints on models, which give, for example, rise to interaction axioms [9, 18]. For reasons of space, we consider only two of the more interesting interaction axioms. That is, we consider the constraint $\mathcal{D}_i \subseteq \mathcal{B}_i$ giving rise to the axiom $B_i\varphi \rightarrow D_i\varphi$ that is called *realism* and the constraint

$$\forall t \in Points_{\mathcal{R}} \exists t' \in Points_{\mathcal{R}} : t\mathcal{D}_i t' \text{ and } t\mathcal{B}_i t'$$

giving rise to the axiom $D_i\varphi \rightarrow \neg B_i\neg\varphi$ that is called *weak realism* [18].

Definition 3 (Semantics).
Let $\mathcal{M}$ be a model with associated set $Points_{\mathcal{R}}$ and $(r, m) \in Points_{\mathcal{R}}$. Then the relation $\models$ is defined by:

- $\mathcal{M}, r, m \models p$ *iff* $p \in \pi(r[m])$
- $\mathcal{M}, r, m \models \neg\varphi$ *iff* $\mathcal{M}, r, m \not\models \varphi$
- $\mathcal{M}, r, m \models \varphi \wedge \psi$ *iff* $\mathcal{M}, r, m \models \varphi$ *and* $\mathcal{M}, r, m \models \psi$
- $\mathcal{M}, r, m \models \bigcirc\varphi$ *iff* $\mathcal{M}, r, m + 1 \models \varphi$
- $\mathcal{M}, r, m \models \varphi\mathcal{U}\psi$ *iff* $\exists k \geq m$ *with* $\mathcal{M}, r, k \models \psi$ *and* $\forall l$ *with* $m \leq l < k$ *we have that* $\mathcal{M}, r, l \models \varphi$
- $\mathcal{M}, r, m \models B_i\varphi$ *iff* $\mathcal{M}, r', m' \models \varphi$ *for all* (r', m') *with* $(r, m)\mathcal{B}_i(r', m')$
- $\mathcal{M}, r, m \models D_i\varphi$ *iff* $\mathcal{M}, r', m' \models \varphi$ *for all* (r', m') *with* $(r, m)\mathcal{D}_i(r', m')$
- $\mathcal{M}, r, m \models I_i\varphi$ *iff* $\mathcal{M}, r', m' \models \varphi$ *for all* (r', m') *with* $(r, m)\mathcal{I}_i(r', m')$

We say a formula φ is satisfiable *in $\mathcal{M}$ if there is a run $r \in \mathcal{R}$ and $m \in \mathbb{N}_0$ such that $\mathcal{M}, r, m \models \varphi$, and simply* satisfiable *if it is satisfiable in some model $\mathcal{M}$. φ is said to be* valid *if $\mathcal{M}, r, m \models \varphi$ for all runs $r \in \mathcal{R}_{\mathcal{M}}$ and $m \in \mathbb{N}_0$ in all models $\mathcal{M}$. We define the logic $\mathsf{BDI_{LTL}}$ as the set of formulas that are valid on the class of $\mathsf{BDI_{LTL}}$-models.*

3.2 Fragments

We are interested in establishing lower complexity bounds for fragments of the logic $\mathsf{BDI}_{\mathsf{LTL}}$ that are still expressive enough to allow reasoning about mental states of other agents. We have reviewed relevant results in Section 2. Based on the summary overview provided there, there are just a few fragments of $\mathsf{BDI}_{\mathsf{LTL}}$ that may be in **NP** (or even have a lower complexity). A number of options to lower complexity are not that interesting since we want to be able to reason about the mental states of agents. Therefore, single agent fragments, or single proposition fragments are not interesting. Fragments that may be interesting clearly need to restrict the temporal depth (i.e. nesting of temporal operators), modal depth (i.e. depth of nesting of mental attitude operators), and may need to restrict the number of propositional atoms used.

By inspecting the tables for linear time logic and multi-agent knowledge/belief logics, to obtain fragments that possibly are in **NP** the following is clear:

- modal depth, i.e. nesting of B, D, and I operators needs to be bounded, and
- no nesting of temporal operators is allowed at all if unbounded number of propositional atoms are allowed, or temporal depth needs to be bounded.

This means that there are only two fragments of the *language* that we need to consider in the remainder when we are looking for fragments for which satisfiability is in **NP**: (1) the fragment with finitely many propositional atoms, limited temporal height, and bounded nesting of other modal operators representing mental attitudes, and (2) the fragment with no nesting of temporal operators and bounded nesting of other modal operators representing mental attitudes. Apart from the language, we also consider slight modifications of the semantics as discussed above as well as the interaction axioms below.

Formally, we use $depth^m(\varphi)$ to denote the *modal depth* of φ with respect to modal operators B_i, I_i, and D_i. We define $depth^m(p) = 0$, $depth^m(\neg\varphi) = depth^m(\varphi)$, $depth^m(\bigcirc\varphi) = depth^m(\varphi)$, $depth^m(\varphi \circ \psi) = \max\{depth^m(\varphi), depth^m(\psi)\}$ for $\circ \in \{\wedge, \mathcal{U}\}$, and $depth^m(O_i\varphi) = depth^m(\varphi) + 1$. Similarly, we define the *temporal depth* $depth^t(\varphi)$ of φ. In this case the number of nested temporal operators are counted.

4. TABLEAU METHOD FOR BDI LOGIC

We will present a tableau-based method for the satisfiability problem of $\mathsf{BDI}^{\mathsf{X,Y,Z}}_{\mathsf{LTL}}$. We also consider settings where mental attitudes interact such as the conditions of realism and weak realism introduced above. The method has been used by others as well and our approach builds upon the work of [23] which discusses the tableau method for temporal logic extended with either a belief or a knowledge operator. The extension we propose also uses ideas presented in [9, 13, 18]. Although a tableau algorithm for linear time and BDI-operators has been provided in [18] as well, the main concern of our paper, i.e. the complexity of $\mathsf{BDI}^{\mathsf{X,Y,Z}}_{\mathsf{LTL}}$ and associated bounded fragments is not discussed in [18].

4.1 Basic Definitions

We use $sub(\varphi)$ to denote the *set of subformulas* of φ. Formulas are classified as either α- or β-*formula* (or none of these). Figure 1 shows which formulas are α- and which formulas are β-formulas. We also refer to α^O-formulas (see Figure 1) as α-formulas; the reason for introducing this separate class of rules is that they are needed when the accessibility relation associated with O_i is reflexive, a case that is treated differently from other properties. (Note, that some formulas match non of these cases, e.g. $\neg O_i\varphi$.) We note that for each α-formula (resp. β-formula) we have that $\alpha \leftrightarrow \alpha_1 \wedge \alpha_2$ (resp. $\beta \leftrightarrow \beta_1 \vee \beta_2$). A set Σ of formulas is said to be α-*closed* (resp. β-*closed*, α^O-closed), if for each α-formula (resp. β-formula, α^O-formula) we have $\{\alpha_1, \alpha_2\} \subseteq \Sigma$ (resp. $\{\beta_1, \beta_2\} \cap \Sigma \neq \emptyset$, $\{\alpha^O_1, \alpha^O_2\} \subseteq \Sigma$). A set Σ of formulas is said to be *fully expanded* if for each $\varphi \in \Sigma$ and for all subformulas $\psi \in sub(\varphi)$, $\psi \in \Sigma$ or $\neg\psi \in \Sigma$. A set of formulas is called *blatantly inconsistent* (*b-inconsistent* for short) if it contains $\perp$, $\neg\top$, or two complementary pairs of formulas $\varphi, \neg\varphi$. If a set is not b-inconsistent it is *b-consistent*.

α	α_1	α_2
$\neg\neg\varphi$	φ	φ
$\neg\bigcirc\varphi$	$\bigcirc\neg\varphi$	$\bigcirc\neg\varphi$
$\varphi \wedge \psi$	φ	ψ
$\neg(\varphi\mathcal{U}\psi)$	$\neg\psi$	$\neg\varphi \vee \neg\bigcirc(\varphi\mathcal{U}\psi)$

β	β_1	β_2
$\neg(\varphi \wedge \psi)$	$\neg\varphi$	$\neg\psi$
$\varphi\mathcal{U}\psi$	ψ	$\varphi \wedge \bigcirc(\varphi\mathcal{U}\psi)$

α^O	α^O_1	α^O_2
$O_i\varphi$	φ	$O_i\varphi$

Figure 1: α, β, α^O-rules with O some modal operator

In the remainder of this paper, we use O to refer to arbitrary BDI operators, i.e. $O \in \{B, D, I\}$, and $\mathcal{O}$ to refer to the corresponding accessibility relation. We say that an operator O is a T-*operator* if the associated accessibility relation $\mathcal{O}$ is reflexive. Similarly, we say that O is a KD-*operator* if the relation is serial, etc. We also write $\mathsf{X} \in schema(O)$ if O is an X-*operator*, $\mathsf{X} \in L$ and so on.

Definition 4 (PC-tableau). *We call a set Σ of formulas a* PC-tableau *if Σ is b-consistent, α- and β-closed, and fully expanded. Moreover, we assume that it is α^O-closed if O is a T-operator.*

A PC-tableau derived from Σ' is a PC-tableau Σ such that $\Sigma' \subseteq \Sigma$. The set of all PC-tableaux derived from Σ' is denoted $PC(\Sigma')$. (Note that if Σ' is b-inconsistent then $PC(\Sigma') = \emptyset$.)

As noted, the use of the rule α^O (see Figure 1) differs from others in that it is applied to all formulas $O\varphi$ iff O is a T-operator; it is therefore easier to incorporate this rule into the definition of a PC-tableau. Note that for reflexive relations $\mathcal{O}$ we actually have $\alpha^O \leftrightarrow \alpha^O_1 \wedge \alpha^O_2$.

4.2 Tableau Construction

We now discuss the tableau-based method for showing satisfiability of the family of logics $\mathsf{BDI}^{\mathsf{X,Y,Z}}_{\mathsf{LTL}}$. Each variant within this family of logics requires that some modifications are applied to the general and generic algorithm below. We begin with explaining the idea for the particular logic $\mathsf{BDI}^{\mathsf{KD,KD,KD}}_{\mathsf{LTL}}$, and thereafter discuss the modifications that are needed for the other members of our family.

$\mathsf{BDI}^{\mathsf{KD,KD,KD}}_{\mathsf{LTL}}$*-pseudo-structure algorithm.*

The tableau approach is based on the idea that tableaux can be used to construct models. Pseudo-structures have the important property that they can be extended to a model if

The algorithm specifies $S = (\hat{Q}, \hat{\rightarrow}, \{\hat{\mathcal{B}}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \{\hat{\mathcal{D}}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \{\hat{\mathcal{I}}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \hat{\pi})$ on input φ_0.

1. (Initialisation) For each $\Delta \in PC(\{\varphi_0\})$ add a state q with $\hat{\pi}(q) = \Delta$ to $\hat{Q}$.

2. Repeat until convergence:

(a) (Modal transitions) For any $q \in \hat{Q}$, if $\neg O_i\psi \in \hat{\pi}(q)$ set

$$(\star) \qquad \Sigma = \{\neg\psi\} \cup \{\chi \mid O_i\chi \in \hat{\pi}(q)\} \cup create\_label^i_{schema(O)}(\hat{\pi}(q)).$$

If such a formula does not exist then define

$$\Sigma = \{\chi \mid O_i\chi \in \hat{\pi}(q)\} \cup create\_label^i_{schema(O)}(\hat{\pi}(q)).$$

For each $\Delta \in PC(\Sigma)$ if there is a state $q' \in \hat{Q}$ with $\hat{\pi}(q') = \Delta$ add the relation $q\hat{\mathcal{O}}_i q'$. Otherwise,

$(\star\star)$ create a node q' with $\hat{\pi}(q') = \Delta$ and add relation $q\hat{\mathcal{O}}_i q'$.

(b) (Temporal transitions) For any $q \in \hat{Q}$, if $\bigcirc\psi \in \hat{\pi}(q)$ then for each $\Delta \in PC(\hat{\pi}(q)/\bigcirc)$ if there is a state $q' \in \hat{Q}$ with $\hat{\pi}(q') = \Delta$ add the transition $q\hat{\rightarrow}q'$ else add a new state q' with $\hat{\pi}(q') = \Delta$ and add the transition $q\hat{\rightarrow}q'$.

3. (Deletion) Delete a state $q \in \hat{Q}$ if one of the following conditions applies:
(a) $\exists\psi \in \hat{\pi}(q)$ such that ψ is (S, q)-temporally inconsistent.
(b) $\exists\psi \in \hat{\pi}(q)$ such that $\psi = \bigcirc\chi$ and there is no q' with $q\hat{\rightarrow}q'$.
(c) $\exists\psi \in \hat{\pi}(q)$ such that $\psi = \neg O_i\chi$ and there is no q' with $q\hat{\mathcal{O}}_i q'$ and $\neg\chi \in \hat{\pi}(q')$.
(d) $\exists\psi \in \hat{\pi}(q)$ such that $\psi = O_i\chi$ and there is no q' with $q\hat{\mathcal{O}}_i q'$ and $\chi \in \hat{\pi}(q')$ (only if O is a $\mathbf{D}$-operator).

Figure 2: $\mathbf{BDI}^{\mathbf{X},\mathbf{Y},\mathbf{Z}}_{\mathbf{LTL}}$-pseudo-structure algorithm for serial accessibility relations and input φ_0. We assume that $O \in \{B_i, D_i, I_i\}$ and that $\hat{\mathcal{O}}$ denotes the corresponding accessibility relation.

The procedure returns the following set:
1. $\emptyset$ if $\mathbf{X} \in \{\mathbf{K}, \mathbf{KD}\}$.
2. $\{O_i\psi \mid O_i\psi \in \Sigma\} \cup \{\neg O_i\psi \mid \neg O_i\psi \in \Sigma\}$ if $X \in \{\mathbf{KD45}, \mathbf{S5}\}$.
3. $\{O_i\psi \mid O_i\psi \in \Sigma\}$ if $X = \mathbf{S4}$.

Figure 3: $create\_label^i_X(\Sigma)$ **procedure.**

and only if the input formula is satisfiable. Accordingly, we present an algorithm that generates pseudo-structures.

Definition 5. *A* pseudo-structure *is a tuple $S = (\hat{Q}, \hat{\rightarrow}, \{\hat{\mathcal{B}}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \{\hat{\mathcal{D}}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \{\hat{\mathcal{I}}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \hat{\pi})$ where $\hat{Q}$ is a (possibly empty) set of states, and $\hat{\rightarrow}$, $\hat{\mathcal{B}}_i$, $\hat{\mathcal{D}}_i$, and $\hat{\mathcal{I}}_i$ are binary relations between states, and $\hat{\pi} : \hat{Q} \rightarrow \mathcal{P}(\mathcal{L})$ assigns sets of formulae to states.*

The basic algorithm is called $\mathbf{BDI}^{\mathbf{KD},\mathbf{KD},\mathbf{KD}}_{\mathbf{LTL}}$-*pseudo-structure algorithm* and is presented in Figure 2. If the algorithm returns a pseudo-structure and the input formula is satisfiable, a $\mathbf{BDI}_{\mathbf{LTL}}$-model can be extracted from the structure that witnesses the truth of the formula. In Theorem 2 it is shown that the pseudo-structure contains a state q whose label contains φ if, and only if, the pseudo-structure can be extended to a $\mathbf{BDI}^{\mathbf{KD},\mathbf{KD},\mathbf{KD}}_{\mathbf{LTL}}$-model satisfying φ.

The first step in the algorithm generates nodes each labeled with a PC-tableau derived from $\{\varphi\}$. Then, steps (2a) and (2b) are performed until none of these cases can be applied anymore. Step (2a) generates $\hat{\mathcal{O}}_i$ transitions. Depending on the properties of $\hat{\mathcal{O}}_i$ (referred to by procedure $schema(O)$) the procedure $create\_label^i_{schema(O)}(\Sigma)$ which is shown in Figure 3 creates the label of a (possibly new) node. Different logics require a different procedure. $\hat{\pi}(q)/\bigcirc$ is defined as $\{\psi \mid \bigcirc\psi \in \hat{\pi}(q)\}$ and $\hat{\pi}(q)/O_i$ analogously.

Condition $(\star\star)$ corresponds to the seriality condition of $\hat{\mathcal{O}}$. In (2b) temporal successors are created. The idea is that if the current state contains a formula $\bigcirc\varphi$ there has to be a $\rightarrow$-related state satisfying φ. Finally, in step (3) the algorithm deletes states which are not consistent – one way or another. Of particular interest is step (3a) which removes nodes in which an eventuality formula cannot be satisfied anymore. This step involves the notion of temporally consistent formulas, which is defined next.

Definition 6 (Temporally consistent). *Given a pseudo-structure S and a state $q \in \hat{Q}_S$ a formula φ is said to be (S, q)-temporally consistent iff if $\varphi = \psi_1\mathcal{U}\psi_2$ then there is a state q' reachable from q via the transitive closure of $\hat{\rightarrow}$ such that $\psi_2 \in \hat{\pi}(q')$. If φ is not (S, q)-temporally consistent it is said to be (S, q)-temporally inconsistent.*

In step 3(d) states are deleted that are not consistent with the fact that an operator O is a $\mathbf{KD}$-operator, which requires a successor state in the corresponding models.

Modifications: $\mathbf{BDI}^{\mathbf{X},\mathbf{Y},\mathbf{Z}}_{\mathbf{LTL}}$-pseudo-structure algorithms.

Before we state our general result, we briefly consider what needs to be modified to cater for other logics than $\mathbf{BDI}^{\mathbf{KD},\mathbf{KD},\mathbf{KD}}_{\mathbf{LTL}}$. We consider the general case and define a generic $\mathbf{BDI}^{\mathbf{X},\mathbf{Y},\mathbf{Z}}_{\mathbf{LTL}}$-pseudo-structure algorithm as follows. First, for operators and associated relations $\mathcal{O}$ that have other properties than seriality the labeling of nodes in $(\star)$ needs to be modified. The required modifications are listed in Figure 3 for the different cases that we consider here.

Finally, for operators that are not $\mathbf{D}$-operators, the cases that relate to the seriality of the accessibility relation in the algorithm need to be disregarded. This applies in particular to condition $(\star\star)$ which should be ignored when dealing with such operators. Finally, also the deletion of states needs to be modified and only the cases 3(a)-3(c) should be executed while case 3(d) in Figure 2 needs to be ignored.

4.3 Soundness and Completeness

We now consider the soundness and completeness of the algorithm. The proofs of the results are fairly standard (cf. [23, 9, 13, 18]) and we focus on some of the basic ideas.

A pseudo-structure $S = (\hat{Q}, \hat{\rightarrow}, \{\hat{\mathcal{B}}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \{\hat{\mathcal{D}}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \{\hat{\mathcal{I}}_i\}_{i\in\mathbb{A}\mathrm{gt}}, \hat{\pi})$ is said to be a $\mathbf{BDI}^{\mathbf{X},\mathbf{Y},\mathbf{Z}}_{\mathbf{LTL}}$-*tableau for* φ if the following 8 conditions are satisfied: (1) There is a state $q \in \hat{Q}$ such that $\varphi \in \hat{\pi}(q)$. (2) For each $q \in \hat{Q}$, $\hat{\pi}(q) \in PC(\hat{\pi}(q))$. (3) If $\mathbf{T} \in schema(O)$ then if $O_i\psi \in \hat{\pi}(q)$ then $\psi \in \hat{\pi}(q)$. (4) If $\mathbf{D} \in schema(O)$ then if $O_i\psi \in \hat{\pi}(q)$ then there is a state q' with $q\hat{\mathcal{O}}_i q'$. (5) If $\mathbf{4} \in schema(O)$ then if $O_i\psi \in \hat{\pi}(q)$ then for all q' with $q\hat{\mathcal{O}}_i q'$, $O_i\psi \in \hat{\pi}(q')$. (6) If $\mathbf{5} \in schema(O)$ then if $q\hat{\mathcal{O}}_i q'$ and $q\hat{\mathcal{O}}_i q''$ and $O_i\psi \in \hat{\pi}(q')$ then $\{O_i\psi, \psi\} \subseteq \hat{\pi}(q'')$. (7) If $\bigcirc\psi \in \hat{\pi}(q)$ then there is a $q' \in \hat{Q}$ such that $q\hat{\rightarrow}q'$. (8) If $\bigcirc\psi \in \hat{\pi}(q)$ then for all $q' \in \hat{Q}$ with $q\hat{\rightarrow}q'$ it holds that $\psi \in \hat{\pi}(q')$.

Conditions 3, 4, 5, and 6 correspond to reflexivity, seriality, transitivity, and Euclideanity, respectively. The following theorems are more or less standard.

Theorem 1 (Tableau satisfiability). *A formula φ is $\mathbf{BDI}^{\mathbf{X},\mathbf{Y},\mathbf{Z}}_{\mathbf{LTL}}$-satisfiable iff there is a $\mathbf{BDI}^{\mathbf{X},\mathbf{Y},\mathbf{Z}}_{\mathbf{LTL}}$-tableau for φ.*

Proof sketch. $\Rightarrow$: It is easily seen that one can define a tableau from a satisfying model. $\Leftarrow$: The set of timelines $\mathcal{R}$ is obtained by unravelling the relation $\hat{\rightarrow}$, enforcing the satisfaction of eventuality formulae (cf. [23] for details). The definition of the state-based $\hat{\mathcal{O}}$-relations to point-based $\mathcal{O}$ accessibility relations is also done according to [23]: For example we set $(r, m)\mathcal{O}_i(r', m')$ if $r(m)\hat{\mathcal{O}}_i r'(m')$. Minor modification are necessary to ensure that $\mathcal{O}$ is a *schema*(O)-operator (e.g. taking the reflexive transitive closure, etc.). □

Theorem 2 (Sound-, Completeness). *The algorithm terminates on all inputs and φ is $\mathsf{BDI}^{\mathsf{X},\mathsf{Y},\mathsf{Z}}_{\mathsf{LTL}}$-satisfiable iff the $\mathsf{BDI}^{\mathsf{X},\mathsf{Y},\mathsf{Z}}_{\mathsf{LTL}}$-algorithm on input φ returns a $\mathsf{BDI}^{\mathsf{X},\mathsf{Y},\mathsf{Z}}_{\mathsf{LTL}}$-tableau for φ iff the structure returned by the algorithm contains a state q containing φ.*

Proof sketch. Termination of the algorithm is guaranteed as there are only finitely many different PC-tableaux and the algorithm does not create nodes twice. Soundness is proved following similar steps as in [23]: It is shown that if the algorithm returns a pseudo-structure for φ then it is actually a tableaux for φ (this is achieved by verifying the 8 conditions of a tableau). For the completeness one shows that if there is no state q in the returned structure, then φ is not satisfiable (cf.[23, Th.4]). The modal operators are treated independently. □

4.4 Interaction Axioms

Finally, we consider the interaction axioms. The realism axiom is given by $B_i\varphi \rightarrow D_i\varphi$. In order to extend our tableau method an additional alpha rule is introduced: $\alpha = B_i\varphi$ and $\alpha_1 = B_i\varphi$ and $\alpha_2 = D_i\varphi$. The condition that we need to add to cover for this axiom is the requirement that a PC-tableau is also closed under this new rule (see Df. 4).

Similarly, for the weak realism axiom $D_i\varphi \rightarrow \neg B_i\neg\varphi$, a second new α-rule is introduced: $\alpha = D_i\varphi$ and $\alpha_1 = D_i\varphi$ and $\alpha_2 = \neg B_i\neg\varphi$, and the definition of a PC-tableau needs to be modified accordingly.

It is not difficult to see that the sound- and completeness results from Section 4.3 also hold with these extensions. The additional α-rules for realism and weak realism give rise to the corresponding rules

9. If $B_i\varphi \in \hat{\pi}(q)$ then $D_i\varphi \in \hat{\pi}(q)$; and

10. if $D_i\varphi \in \hat{\pi}(q)$ then $\neg B_i\neg\varphi \in \hat{\pi}(q)$, respectively,

in the $\mathsf{BDI}^{\mathsf{X},\mathsf{Y},\mathsf{Z}}_{\mathsf{LTL}}$-tableau (see also [18, 23]). The corresponding model is constructed from such a table in the same way as in the cases without interaction (cf. Theorem 1) and the completeness proof of Theorem 2 is done analogously (see e.g. [18, 9]).

5. COMPLEXITY OF SATISFIABILITY

In this section we consider the complexity of the $\mathsf{BDI}^{\mathsf{X},\mathsf{Y},\mathsf{Z}}_{\mathsf{LTL}}$-satisfiability problem. In [13] a tableau-based decision procedure has been used to prove **PSPACE** membership of the multi-agent logics **K**, **T**, **S4**, **S5**, and **KD45**. In [9] a **PSPACE** tableau algorithm for a **BDI** logic combining **S4**, **K**, and **KD** operators has been presented and in [18, 23] tableau-based algorithms for linear-time and combinations of other modalities were given. However, the complexity of the latter algorithms has not been analysed.

"Standard" **LTL**-tableau constructions have been shown implementable in **PSPACE** [22]. The algorithm presented here when executed with purely temporal formulae is essentially equivalent to that of [22]. The next result shows that the addition of other modal operators does not increase the complexity.

Theorem 3. *The $\mathsf{BDI}^{\mathsf{X},\mathsf{Y},\mathsf{Z}}_{\mathsf{LTL}}$-satisfiability problem is* **PSPACE**-*complete for all $\mathsf{X}, \mathsf{Y}, \mathsf{Z} \in L$.*

Proof sketch. The lower bound follows from **LTL**-satisfiability, cf. e.g. [5]. We sketch the upper bound. In the following we take $d = depth^m(\varphi)^{\mathcal{O}(1)}$.

(I) Purely modal part (step 2(a) in our algorithm): In [13] it was shown that the length of sequences of subsequent states generated by the tableau algorithm is bounded by d. (Note, that we abstracted from the calculation of PC-tableaux. The treatment is standard.) The "tree" consisting of all these polynomial length sequences is searched in a depth first search manner using only polynomial space. This procedure generalizes to multiple modal operators.

(II) Purely temporal part (step 2(b) in our algorithm): In [22] a **PSPACE**-tableau algorithm for **LTL** is presented. The main observation is that if an **LTL** formula is satisfiable then it is satisfiable on a path $q_0q_1 \dots q_m$ such that for some q_j, $j \leq m$ with $q_m \hat{\rightarrow} q_j$ and $m \leq 2^{\mathcal{O}(|\varphi|)}$. The following polynomial space algorithm implements this idea: Guess state q_j and guess valid subsequent successor states q (for at most $2^{\mathcal{O}(|\varphi|)}$ steps) until $q = q_j$ for some state in which all eventualities are fulfilled (φ is satisfiable). In memory, only the counter, the current state and q_j are kept.

(III) We consider $\mathsf{BDI}_{\mathsf{LTL}}$. As noted in [10] time and epistemic operators are independent from each other which allows for a combination of (I) and (II). From (I) we know that the number of consecutive epistemic steps is bounded by d. Now, each time we are in step 2(a) we apply the depth first search strategy from (I) and each time we execute step 2(b) we try to build the infinite trace as in (II). The number of "temporal traces" is bounded by $|\varphi| \cdot d$. Hence, one has to store at most $|\varphi| \cdot d$ counters, current states, and the states guessed to indicate the entry point of a loop for the temporal part and $|\varphi| \cdot d$ states constituting the current "epistemic path" which is kept in memory (possibly interrupted by a temporal path). (In addition to some other book keeping operations needed for the depth first search.) □

In a non-temporal setting, it was shown that adding various interaction axioms does not increase the complexity [9]. This is also the case in the temporal setting considered here. The result follows immediately since the interaction axioms do only require the application of some additional α-rules as explained above.

Corollary 4. *The $\mathsf{BDI}^{\mathsf{X},\mathsf{Y},\mathsf{Z}}_{\mathsf{LTL}}$-satisfiability problem assuming realsim or weak realism is* **PSPACE**-*complete.*

5.1 Bounded Temporal and Modal Depth

Here we consider fragments of $\mathsf{BDI}^{\mathsf{X},\mathsf{Y},\mathsf{Z}}_{\mathsf{LTL}}$ with bounds on the temporal depth $depth^t(\varphi)$ and modal depth $depth^m(\varphi)$ of a formula φ.

To be precise, we define the fragments of $\mathsf{BDI}^{\mathsf{X},\mathsf{Y},\mathsf{Z}}_{\mathsf{LTL}}$ considered here using the notions of temporal and modal depth. We define $\mathcal{L}^{i,j}$ as the set of formulas $\varphi \in \mathcal{L}$ with $depth^t(\varphi) \leq i$ and $depth^m(\varphi) \leq j$.

The upper bound of the next result follows from Theorem 3 and the lower bound from [5] where the satisfiability problem for **LTL** with a temporal bound of ≥ 2 is shown to be **PSPACE**-complete.

Proposition 5 ([5]). *Let $i, j \in \mathbb{N}$ and $i \geq 2$. The $\mathbf{BDI}_{\mathbf{LTL}}^{\mathbf{X,Y,Z}}$-satisfiability problem over the language $\mathcal{L}^{i,j}$ is* **PSPACE**-*complete. The same result holds over the class of models that satisfy (weak) realism.*

The previous result is negative. However, the complexity improves if the temporal depth is at most 1.

Theorem 6. *Let $i \in \mathbb{N}$. The $\mathbf{BDI}_{\mathbf{LTL}}^{\mathbf{X,Y,Z}}$-satisfiability problem over the language $\mathcal{L}^{1,i}$ is* **NP**-*complete. The same result holds over the class of models that satisfy (weak) realism.*

Proof sketch. In [5] it is shown that if an **LTL** formula with $depth^t(\varphi) \leq 1$ is satisfiable then satisfiability can be witnessed by an initial polynomial length prefix of a path. This prefix can be non-deterministically guessed and verified in polynomial time. Moreover, as shown in [13, 9] the purely modal tableaux have $|\varphi|^{\mathcal{O}(i)}$ many nodes. Combining both results and inspecting the proof sketch of Theorem 3 shows that for the full logic the number of nodes in the tableau is also bounded by $|\varphi|^{\mathcal{O}(i)}$. (Here, it is important to note that there can be at most one alternation between epistemic and modal transitions along each path in the tableau.) Thus, we can guess a tableau of polynomial size and check whether it satisfies φ. $\square$

Finally, we consider the case in which the temporal and modal depth are bounded and additionally only finitely many propositional symbols are available. We note that a finite set of propositions is often of less practical interest (e.g. settings requiring natural numbers usually require an unbounded number of propositional symbols). For the purely temporal fragment it has been proven that the problem can be checked in logarithmic deterministic space [5]. In [12], on the other hand, satisfiability of the purely modal fragment has been shown to be solvable in linear deterministic time. The proof of [5], however, can directly be used to show that $\mathbf{BDI}_{\mathbf{LTL}}^{\mathbf{X,Y,Z}}$-satisfiability for formulae of $\mathcal{L}^{i,j}$, i, j fixed, can be checked in logarithmic deterministic space. The basic idea relies on the observation that there are only finitely many inequivalent formulae over $\mathcal{L}^{i,j}$. For each class of equivalent formulae we identify a "canonical formula". Then, given a formula φ of which the satisfiability should be checked simple rewrite rules can be applied to determine the canonical representation ψ of φ. If ψ does not correspond to the canonical representation of $\perp$ it is satisfiable.

Theorem 7. *Let $i, j \in \mathbb{N}$. The $\mathbf{BDI}_{\mathbf{LTL}}^{\mathbf{X,Y,Z}}$-satisfiability problem over $\mathcal{L}^{i,j}$ over a finite set of propositional atoms can be solved in deterministic logarithmic space. This is also true for the class of models that satisfy (weak) realism.*

6. REASONING ABOUT MENTAL STATES

We have studied various fragments of the linear time BDI logic $\mathbf{BDI}_{\mathbf{LTL}}$ and obtained results on the complexity of the corresponding satisfiability problems. Our motivation for examining these fragments has been that agents need to be able to reason about other agents' mental states and logic seems one of the most suitable tools to do so. However, an agent also needs to be able to do so within reasonable amounts of space and time. We have shown that fragments exist of which the complexity can be reduced to **NP**. As argued above, as a first step towards a logic that can actually be used, these results are promising. Here we briefly consider informally whether these fragments are also satisfactory for the main task we had in mind, i.e. for the representation and reasoning with mental states of other agents.

In order to evaluate this, we briefly introduce and discuss some *minimal criteria*. In order to support reasoning about other agents' mental states, we argue that the minimal requirement a logic needs is that the logic (i) is able to discriminate between *informational attitudes* such as beliefs and knowledge and *motivational attitudes* such as desires, goals, and intentions; (ii) facilitates reasoning about a *finite number of nestings* of mental attitudes; (iii) facilitates reasoning about *any finite* number of agents; and (iv) is able to discriminate between types of motivational states, and, ideally would support reasoning about the class of *deadline goals* which subsumes *achievement* and *maintenance* goals.

Criterium (i) has motivated us to study fragments of $\mathbf{BDI}_{\mathbf{LTL}}$. This is one of the more well-known types of agent logics and clearly distinguishes between informational and motivational states as parts of an agent's state.

Criterium (ii) is in part motivated by results from cognitive science which inform us that a limited depth of mental operators seems sufficient for reasoning about mental states as humans are not able to nest such operators to a depth of more than 3 [7]. On the other hand, in the context of agent communication and reasoning about speech acts, one finds that one quickly needs at least three levels of nesting (see e.g. the example of a decision rule to inform another agent below). Given these basic results about depth of nesting of mental attitude operators, it seems reasonable to bound the depth of such operators. The $\mathbf{BDI}_{\mathbf{LTL}}$ fragments that we showed to be in **NP** clearly support such limited nesting.

Criterium (iii) is an obvious criterium given that we are motivated by logics that allow agents to reason about other agents' mental state. This motivated us to introduce the multi-agent variant of BDI logic. From a semantic and a complexity perspective this requirement poses no problems.

Criterium (iv) most clearly distinguishes our work from that of others, in particular [9], which studies a multi-agent logic called TeamLog that does not incorporate time. Being able to support some form of reasoning about time, however, greatly increases the expressivity and in particular allows to distinguish between various kinds of goals. For example, using a single $\mathcal{U}$ operator we are already able to represent *deadline goals*. Moreover, it is clear that achievement goals can be represented by $\Diamond\phi$ and maintenance goals by $\Box\phi$. The fragments we showed to be in **NP** support making these distinctions. However, only the fragment with finitely many atoms supports nesting of temporal operators (Th. 7). The fragment that allows for infinitely many atoms (Th. 6) thus does not allow reasoning about e.g. persistence goals $\Diamond\Box\varphi$. It would be interesting to investigate if that fragment could be extended with such specific combinations (while not allowing nesting of arbitrary temporal operators).

Clearly, group attitudes such as common knowledge and common or joint intentions cannot be defined using the bounded fragments we discussed. Introducing these concepts immediately blows up complexity (e.g. [9, 13]). Of course, we can define more basic notions such as "everybody in a group believes φ", i.e. $E\text{-}B_G(\varphi) \leftrightarrow \bigwedge_{i \in G} B_i\varphi$. Even without such stronger notions, however, agents can coordinate their behavior. The idea would be that agents can incorporate reasoning based on a $\mathsf{BDI_{LTL}}$ fragment into their decision making algorithms. This would already allow for action choices based on decision rules of the form: (i) **if** $B_iI_j\varphi$ **then do**(a), (ii) **if** $B_iB_j\varphi$ **then do**(a), and (iii) **if** $B_iD_j(K_j\varphi \vee K_j\neg\varphi) \wedge B_i\varphi$ **then inform** (φ).

7. CONCLUSION AND FUTURE WORK

We have discussed the issue of reasoning about mental states of other agents and argued that the BDI logic $\mathsf{BDI_{LTL}}$ offers a suitable tool to do so. We have studied several fragments of $\mathsf{BDI_{LTL}}$ and showed complexity of the satisfiability problem for these fragments to be in **NP**. We also introduced a very generic tableau method to do so.

From a complexity point of view there are many ways to continue the search for practical and useful fragments of $\mathsf{BDI_{LTL}}$. For example, in [6] it is shown that the satisfiability problem for a special semantics for temporal logic with knowledge called $XL5$ is **NP**-hard and it is interesting to study whether similar techniques can be applied to $\mathsf{BDI_{LTL}}$. [16] discusses the complexity of the satisfiability problem for a range of multi-modal logics *restricted to the Horn fragment* including e.g. KD_n, $\mathsf{KD45}_n$, etc. The restriction to the Horn fragment is relevant because it may provide practical extensions to e.g. Prolog. [16] also investigates bounded modal depth but does not discuss fragments that include both informational and motivational operators and does not discuss temporal operators either. Another line of research would involve looking at CTL instead of LTL, and identify complexity classes for fragments of $\mathsf{BDI_{CTL}}$.

In the future, we would also like to consider agents that have *perfect recall* or do not *learn* and extend our analysis in this respect. Both properties are of practical importance. However, usually reasoning becomes computationally much harder in the presence of these properties.

Finally, it is interesting to experiment in practice with implemented reasoners to determine what is feasible. Moreover, if we want to incorporate a restricted BDI logic into the decision making component of a software agent, we not only need a reasoner, but we will also need ways to efficiently update sets of $\mathsf{BDI_{LTL}}$ formulas when the agent receives new information from its environment or through communication with other agents.

8. REFERENCES

[1] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2001.

[2] R. H. Bordini, M. M. Dastani, J. Dix, and A. E. F. Seghrouchni. *Multi-Agent Programming: Languages, Platforms and Applications*. Springer, 2005.

[3] R. H. Bordini, M. M. Dastani, J. Dix, and A. E. F. Seghrouchni. *Multi-Agent Programming: Languages, Tools and Applications*. Springer, 2009.

[4] P. R. Cohen, H. R. Levesque, and I. Smith. On team formation. In J. Hintikka and R. Tuomela, editors, *Contemporary Action Theory. Synthese*. 1997.

[5] S. Demri and P. Schnoebelen. The complexity of propositional linear temporal logics in simple cases. Technical report, Lab. Spécification et Vérification, CNRS and ENS de Cachan, France, 1997.

[6] C. Dixon, M. Fisher, and B. Konev. Taming the complexity of temporal epistemic reasoning. In *Proceedings of the Seventh International Symposium on Frontiers of Combining Systems (FroCoS)*, 2009.

[7] B. Dunin-Keplicz and R. Verbrugge. Awareness as a vital ingredient of teamwork. In *Proceedings of the 5th international joint conference on Autonomous agents and multiagent systems*, pages 1017–1024, 2006.

[8] E. H. Durfee. Scaling up agent coordination strategies. *Computer*, 34(7):39–46, 2001.

[9] M. Dziubinski, R. Verbrugge, and B. Dunin-Keplicz. Complexity issues in multiagent logics. *Fundamenta Informaticae*, 75(1-4):239–262, 2007.

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

[11] M. Fisher. Implementing BDI-like Systems by Direct Execution. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence*, pages 316–321, 1997.

[12] J. Halpern. The effect of bounding the number of primitive propositions and the depth of nesting on the complexity of modal logic. *Artificial Intelligence*, 75(2):361–372, 1995.

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

[14] J. Halpern and M. Vardi. The complexity of reasoning about knowledge and time. i. lower bounds. *Journal of Computer System Sciences*, 38(1):195–237, 1989.

[15] W. G. Kennedy, M. D. Bugajska, A. M. Harrison, and J. G. Trafton. "like-me" simulation as an effective and cognitively plausible basis for social robotics. *International Journal of Social Robotics*, 1(2):181–194, 2009.

[16] L. A. Nguyen. On the complexity of fragments of modal logics. In *Advances in Modal Logic*, pages 249–268, 2005.

[17] J. Nick R. Commitments and conventions: The foundation of coordination in multi-agent systems. *The knowledge engineering review*, 8:223–250, 1993.

[18] A. S. Rao. Decision procedures for propositional linear-time belief-desire-intention logics. In *Intelligent Agents II: Agent Theories, Architectures, and Languages*, pages 33–48, 1996.

[19] R. A. Schmidt and D. Tishkovsky. A Decidable Dynamic Logic for Agents with Motivational Attitudes. In *Proceedings of the Workshop on Methods for Modalities-2 (M4M'01)*, 2001.

[20] M. Tambe. Agent architectures for flexible, practical teamwork. In *Proceedings of the National Conference on Artificial Intelligence*, pages 22–28, 1997.

[21] W. van der Hoek and M. Wooldridge. Towards a logic of rational agency. *Logic Journal of IGPL*, 11(2):135–159, 2003.

[22] P. Wolper. Temporal logic can be more expressive. *Information and Control*, 56:72–99, 1983.

[23] M. Wooldridge, C. Dixon, and M. Fisher. A tableau-based proof method for temporal logics of knowledge and belief. *Journal of Applied Non-Classical Logics*, 8:225–258, 1998.