Multi-Agent Logics of Dynamic Belief and Knowledge*

Renate A. Schmidt[1,2] and Dmitry Tishkovsky[1]

[1] Department of Computer Science, University of Manchester
{schmidt,dmitry}@cs.man.ac.uk
[2] Max-Planck-Institut für Informatik, Saarbrücken, Germany
schmidt@mpi-sb.mpg.de

Abstract. This paper proposes a family of logics for reasoning about the dynamic activities and informational attitudes, i.e. the beliefs and knowledge, of agents. The logics are based on a new formalisations and semantics of the test operator of propositional dynamic logic and a representation of actions which distinguishes abstract actions from concrete actions. The new test operator, called informational test, can be used to formalise the beliefs and knowledge of particular agents as dynamic modalities. This approach is consistent with the formalisation of the agents' beliefs and knowledge as *K(D)45* and *S5* modalities. Properties concerning the preservation of informativeness, truthfulness and belief are proved for a derivative of the informational test operator. It is shown that common belief and common knowledge can be expressed in these logics. As a consequence, these logics are more expressive than propositional dynamic logic with an extra modality for belief or knowledge. However, the logics are still decidable and in 2EXPTIME. Versions of the considered logics express natural additional properties of beliefs or knowledge and interaction of beliefs or knowledge with actions. A simulation of *PDL* is constructed in one of these extensions.

1 Introduction

It is accepted that multi-agent systems are important to many areas in computer science, starting from specifying and verifying multi-threading systems and multi-processor computers to modelling populations of robots which have animal or human-like behaviour (more applications of multi-agent systems can be found in e.g. [23]). In order to reason about agents appropriate mathematical tools must be developed. Logical methods are widely used for studying and formalising multi-agent systems. Modal logics, in particular, are often used for this purpose and offer a number of advantages [20, 21]. In the landscape of agent systems among the more well-known agent formalisms with a modal flavour are the seminal combination of dynamic logic and epistemic logic by [16], the *BDI* model [17, 18], the *KARO* framework [15, 21], and temporal logics of knowledge

* We thank C. Dixon, M Fisher and U. Hustadt for valuable discussions. This research is supported by EPSRC Research Grants GR/M88761 and GR/R92035.

and belief [3, 6, 7]. Examples of more recent work is the proposal of a modal logic framework of belief and change in [12], and an epistemic dynamic logic in [10].

This paper is an investigation of actions and the informational attitudes, i.e. beliefs and knowledge, of agents. For reasoning about actions, a single agent framework is usually considered and then multiple copies of the framework are taken to formalise a multi-agent system. It is accepted that this approach has limitations and is generally not sufficient. For example, the approach does not allow the formalisation of the statement that 'if Michael does something then his brother Jerry does the same'. The key in this example is the implicit quantifier over the action 'does something'. It is easy to see that 'does something' and 'does the same' both stand for the same action. In a single agent framework the actions an agent can perform are fixed and, hence, disjoint copies of this framework will give us an independent set of actions for the different agents. This is not natural, because, for instance, both Michael and Jerry are able to eat, to sleep, etc. The action expressed by the verb 'to eat' is a general action which belongs to both agents Michael and Jerry. In the logic proposed in this paper the aim is to eliminate this independence. This is done by distinguishing between abstract and concrete actions. In our example, the verb 'to eat' is considered an abstract action, while 'Michael eats' is a concrete action.

Another idea proposed in this paper is an action of making the implicit informational attitudes (belief or knowledge) of an agent explicit, that is, we introduce an action of confirming belief or knowledge. The action of confirming belief (or knowledge) is modelled by a modified test operator, called *informational test*. The informational test operator was introduced in [19] as replacement for the standard test operator of propositional dynamic logic (*PDL*). Thus instead of having the rather strong capability of confirming all true properties in the current world, agents only have the capability of confirming formulae in their belief or knowledge states. It is not normal that agents have full awareness of the environment in which they operate, because usually, some parts of the environment are hidden from an agent. For example, due to some defect or an obstacle an agent may not be able to test whether a door is open or closed. However, if the agent's sensor has successfully tested the door, the agent's belief or knowledge state will be updated and now the agent can bring her belief/knowledge to the fore with the informational test operator. There are also logical reasons for not admitting the standard test operator of *PDL*. In [19] it was shown that the test operator interacts negatively with the epistemic *S5* modality in the axiomatic product of *PDL* and *S5*. In particular, the interaction of the dynamic modalities and the epistemic modality the commuting axioms of no learning and perfect recall (these axioms will be defined below) has the unwanted effect of eliminating the epistemic operator.

The structure of the paper is as follows. In Section 2 we define a logic, called *BDL*, as a basic semantical formalism. In particular, we define the language and a formal semantics. A finite axiomatisation of *BDL* is given in Section 3 and we describe a family of important extensions of *BDL* for which we prove the completeness, the small model property and decidability. We also construct a simulation of *PDL* in one of the extensions of *BDL*. In Section 4 we consider some useful properties of the informational test operator which it has in contrast to the standard test operator of *PDL*. Section 5 discusses the contributions of the paper and contains some remarks about further directions of research.

2 Language, semantics and expressiveness

Before we define the language and formal semantics we discuss from an informal point of view what we want to express in the language and which properties of real examples of multi-agent systems must be incorporated in the semantics.

First of all, we assume that a typical agent is active and intelligent. An example of an agent is a user of the Internet. Such a user is usually very active (at least, during her Internet session) and her requests are reasonable. (We do not speak about the IQ level or professional skills of an agent and determine her intelligence by her ability to make some logical reasoning and decisions.) Second, we work under the thesis that agents interact with each other, acting in some common environment. If there is no interaction between the agents then, obviously, a multi-agent system can be reduced to a single agent system. Continuing our example, all users of the web can interact with each other via protocols such as PPP, FTP, HTTP, SMTP, etc. Also, the common environment changes due to the agents' activities. The state of the web is changed simply by a file transfer using FTP and because of the quantity of such, or other, information transfers being done by large numbers of generally unpredictable users the web is changing unpredictably every fraction of a second. This requires the formal semantics to be a semantics of *states* and it must take account of the unpredictability of the agents behaviour, i.e. it must reflect a kind of *non-determinism* in the system.

As agents are active, they can perform some basic or *atomic actions* allowed in the system. For example, our Internet user can upload and download documents. Of course, these actions can be disassembled into more primitive actions like the instructions of a transfer protocol but we are not interested in such detail and will reason about actions *in general* in some abstract way.

Further, an agent has intelligence and is allowed to combine simple actions into complex ones, for example the action 'download a file from a site and then send it to an address' is the composition of two obvious actions. The actions which are allowed to be performed by *any* agent will be called *abstract actions*. We already know that they can be atomic or complex.

An abstract action is just a name for a real, *concrete action* which an agent can perform. Any concrete action is always associated with some agent because different agents can perform an (abstract) action in different ways. For example, one of the users of the web might use a standard web-browser to download a file, while another user might use telnet and her knowledge of a transfer protocol to get access to the same file. But from an abstract point of view users do the same action, only in different ways.

We want the agent herself to choose how to perform an action. This again is motivation for *non-deterministic* actions or alternative path execution. Non-determinism actually means that we do not have complete information about the system behaviour or it is difficult to obtain or store such information. The web example is a case in point. Indeed, we cannot know a priori which user will connect to a server and what kind of requests she will make.

Another property which we aim to preserve is that concrete actions even if they are associated with different agents can be composed in complex blocks. This property is very helpful for describing the joint work of some group of agents, i.e. teamwork. For example, we want to be able to express the (concrete) action: 'agent 1 designs a database and then agent 2 implements the database'.

Finally, because agents are intelligent in some sense, they must have some *knowledge* or, at least, *beliefs* about the world. One can think about an agent's beliefs (knowledge) as some set of formulae in which the agent believes (or which the agent knows). Because agents act and 'learn' during their activity there must be some kind of interdependence between beliefs (knowledge) and actions. The paper introduces and investigates the idea that obtaining information about the agent's beliefs or knowledge is nothing other than a certain action inside the agent's mind. The agent's beliefs and knowledge are incorporated through a dynamic operator, called *informational test*. This operator confirms the belief (knowledge) of an agent as opposed to confirming the properties of the actual state as the classical test operator does in *PDL*.

The language of BDL. The language of *BDL* has four disjoint sorts: the sort Ag for agents, the sort AA for abstract actions, the sort CA for concrete actions, and the sort For for formulae. Abstract actions can be atomic or complex. Concrete actions are abstract actions associated with some concrete agent. They can also be composed to form complex concrete actions, and the semantics of these actions must incorporate the non-deterministic nature of agents. As usual, formulae describe the properties of multi-agent systems.

The language $\mathcal{L}$ of the logic *BDL* is defined over a countable set $\mathrm{Var} = \{p, q, r, \ldots\}$ of propositional variables, a countable set $\mathrm{AtAA} = \{a, b, c, \ldots\}$ of *abstract atomic action* variables, and a countable set $\mathrm{Ag} = \{i, j, k, \ldots\}$ of agent variables. The logical connectives include the classical connectives, $\rightarrow$ and $\perp$, the *informational test* operator ⁇, the standard *PDL* connectives on action formulae, $\cup$, $;$, and $*$, and the dynamic modal operator $[\_]$ (box). Abstract actions, concrete actions and formulae are defined by simultaneous induction so that AA, CA and For are the smallest sets that satisfy the following conditions.

- $\mathrm{AtAA} \subseteq \mathrm{AA}$, $\mathrm{Var} \cup \{\perp\} \subseteq \mathrm{For}$.
- If $\phi \in \mathrm{For}$ and $\alpha, \beta \in \mathrm{AA}$ then $\phi⁇, \alpha^*, \alpha \cup \beta, \alpha;\beta \in \mathrm{AA}$.
- If $\alpha \in \mathrm{AA}$ and $i \in \mathrm{Ag}$ then $\alpha_i \in \mathrm{CA}$.
- If $\alpha, \beta \in \mathrm{CA}$ then $\alpha^*, \alpha \cup \beta, \alpha;\beta \in \mathrm{CA}$.
- If $\phi, \psi \in \mathrm{For}$ and $\alpha \in \mathrm{CA}$ then $\phi \rightarrow \psi, [\alpha]\phi \in \mathrm{For}$.

The following are assumed to be defined formula connectives: $\neg$, $\top$, $\vee$, $\wedge$, $\leftrightarrow$. As usual $\langle\_\rangle$ is the dual of $[\_]$, defined by $\langle\alpha\rangle\phi \leftrightarrow \neg[\alpha]\neg\phi$. We define the *belief* operator $\mathbf{B}_i$ as the modal operator $[(\top ?)_i]$. The formula $\mathbf{B}_i p$ is read as 'agent i believes p'.

By a *logic* we understand any subset of the set For closed under the inference rules of modus ponens, generalisation and the substitution rule.

(Modus ponens) $\qquad \phi, \phi \to \psi \vdash \psi$

(Generalisation) $\qquad \phi \vdash [\alpha]\phi$

(Here $\phi, \psi \in$ For and $\alpha \in$ CA.) If a formula belongs to a logic then the substitution rule specifies that all substitution instances belong to the logic. The substitution rule allows the uniform substitution of formulae for propositional variables, abstract actions for atomic abstract actions and agent variables for agent variables in the following sense.

Definition 1. *Assume x denotes an agent variable, a propositional variable, or an atomic abstract action in any formula ϕ. Let s denote any agent variable, respectively any formula or abstract action. A formula ψ is a* substitution instance *of ϕ under the substitution of s for x, written $\psi = \phi\{s/x\}$, if all occurrences of the variable x in ϕ are replaced by the term s in ψ.*

Definition 2. *Let a be an atomic abstract action. Suppose a does not occur in any complex abstract actions in the formula ϕ, and $a_{i_1}, \ldots, a_{i_n}$ are all the occurrences of a in ϕ. Suppose the agent variables $i_1, \ldots, i_n$ do not occur anywhere else in ϕ except in $a_{i_1}, \ldots, a_{i_n}$. Let j be a distinguished agent variable in a concrete action α. A formula ψ is a* substitution instance *of ϕ under the substitution α for a_j, written $\psi = \phi\{\alpha/a_j\}$, if all occurrences of $a_{i_1}, \ldots, a_{i_n}$ of the term a in ϕ are respectively replaced by the occurrences of $\alpha\{i_1/j\}, \ldots, \alpha\{i_n/j\}$ in ψ.*

Examples of legal substitutions for an atomic action variable are:

$$([a_i]\mathbf{B}_i p \to \mathbf{B}_i[a_i]p)\{(b;c)/a\} = [(b;c)_i]\mathbf{B}_i p \to \mathbf{B}_i[(b;c)_i]p$$
$$([(a_i)^*]p \to [a_i][(a_i)^*]p)\{(b_j;c_k)/a_j\} = [(b_i;c_k)^*]p \to [b_i;c_k][(b_i;c_k)^*]p$$

In the first example, an abstract action is substituted for an atomic abstract action. The second example shows, we can substitute for a_j if occurrences of a_i do not bind any other actions via agent i. The following is illegal, because a_i is bounded through i with $\mathbf{B}_i$.

$$([a_i]\mathbf{B}_i p \to \mathbf{B}_i[a_i]p)\{(b_j;c_k)/a_j\} = [b_i;c_k]\mathbf{B}_i p \to \mathbf{B}_i[b_i;c_k]p$$

Remark 1. According to the substitution rule each formula of the logic is true independently from the values of propositional variables, atomic actions and agents. This means, the substitution rule permits reasoning about *the most general* properties of agent systems in a uniform way. More specifically, any formula of a logic is true if and only if the universal closure of the formula over all propositional variables, abstract atomic actions and agents is true in the logic.

For any two sets of formulae Γ and Δ, $\Gamma \oplus \Delta$ denotes the least logic which contains both Γ and Δ.

By definition, the logic *BDL* is the set of formulae valid in all *BDL* models, which are defined next.

Semantics. A *BDL model* is a tuple $M = \langle S, Q, R, \models \rangle$ of a non-empty set S of states, a mapping Q which associates with each concrete action α a binary relation $Q(\alpha)$, a mapping R from the set of agents into the set of transitive and Euclidean relations over S, and $\models$ is the truth relation. Any model must satisfy the following ten conditions. The usual conditions on Q for any *concrete* actions α and β:

(M1) $Q(\alpha \cup \beta) = Q(\alpha) \cup Q(\beta)$
(M2) $Q(\alpha;\beta) = Q(\alpha) \circ Q(\beta)$
(M3) $Q((\alpha)^*) = (Q(\alpha))^*$

Here $\circ$ denotes relational composition and $*$ is the reflexive transitive closure operator. The conditions which link the concrete actions with abstract actions, for any *abstract* actions α, β and agent i:

(M4) $Q((\alpha \cup \beta)_i) = Q(\alpha_i \cup \beta_i)$
(M5) $Q((\alpha;\beta)_i) = Q(\alpha_i;\beta_i)$
(M6) $Q((\alpha^*)_i) = Q((\alpha_i)^*)$

The standard conditions on the truth relation must be true, where s denotes any state, ϕ and ψ are arbitrary formulae and α is any *concrete* action:

(M7) $M, s \not\models \bot$
(M8) $M, s \models \phi \to \psi \iff M, s \models \phi \Rightarrow M, s \models \psi$
(M9) $M, s \models [\alpha]\phi \iff \forall t\, ((s,t) \in Q(\alpha) \Rightarrow M, t \models \phi)$

And the semantics of the informational test operator is given by:

(M10) $Q((\phi\text{?`})_i) = \{(s,t) \in R_i \mid \forall u\, ((t,u) \in R_i \Rightarrow M, u \models \phi)\}$.

(In this paper, we write R_i instead of $R(i)$.) Intuitively $p\text{?`}$ is an action which can be successfully accomplished only if agent has $p$ as one of her *beliefs* in the current state. Thus, $p\text{?`}$ is the action of confirming the agent's own beliefs. This is in contrast to the interpretation of the classical test operator in *PDL*. The classical test operator gives the agent the (rather strong) capability of confirming the universal truth of any formulae, rather than just confirming own beliefs.

A formula ϕ is said to be *valid* in a state s of a model M if $M, s \models \phi$, and ϕ is *valid* in M if it is valid in all states of the model.

Informally, a model is a set of states of some multi-agent system. Each state is characterised by the set of properties of the multi-agent system which hold in the state and is represented by a set of formulae. The truth relation of a model assigns such a set of properties to any state. Each action is interpreted as a set of transitions between the states, roughly speaking, a set of arrows which connect the states. Every arrow describes one possible way of executing the action. Thus, any Kripke model is a multi-agent system represented on some level of abstraction. Of course, one can propose other levels of abstraction and detailisations of multi-agent systems, and each of them will have it's own notion of a model, for instance, interpreted systems [7], Aumann structures [3], hypercubes [14], general Kripke models and algebras [24].

What is the expressiveness of the logic just defined? Trivially, *PDL* without the (classical) test operator is a sublogic of *BDL*. Consequently, it is possible to reason about simple programs inside *BDL*. To give us the capability to reason about any programs a minimal reinforcement is used in the next section (Theorem 5).

It is possible to show that *BDL* is more expressive than formalisms given by the combination of multiple copies of *PDL* with belief operators. For example, in such logics the notion of common belief cannot be expressed. In *BDL* common belief can be defined as follows. Let I be a finite set of agents, i.e. $I = \{i_0, \ldots, i_m\} \subseteq \mathrm{Ag}$. The operator 'every agent in I believes that ...', denoted by $\mathbf{E}_I$, is defined by the formula $\mathbf{E}_I p \leftrightarrow [(\top ?)_{i_0} \cup \cdots \cup (\top ?)_{i_m}]p$. The operator $\mathbf{C}_I$ of common belief among the agents in I can then be defined by: $\mathbf{C}_I p \leftrightarrow [((\top ?)_{i_0} \cup \cdots \cup (\top ?)_{i_m})^*]\mathbf{E}_I p$. That is, common belief is actually the transitive closure of everyone believes.

3 Axiomatisation, decidability and complexity

A Hilbert-style axiomatisation of *BDL* is given by the following set of axioms and the previously defined inference rules of modus ponens, generalisation and substitution.

(A1) Axioms of classical propositional logic
(A2) *PDL*-like axioms for test-free actions:
  (A2.1) $[a_i](p \to q) \to ([a_i]p \to [a_i]q)$
  (A2.2) $[a_i \cup b_j]p \leftrightarrow [a_i]p \wedge [b_j]p$
  (A2.3) $[a_i;b_j]p \leftrightarrow [a_i][b_j]p$
  (A2.4) $[(a_i)^*]p \to p \wedge [a_i]p$
  (A2.5) $[(a_i)^*]p \to [a_i][(a_i)^*]p$
  (A2.6) $p \wedge [(a_i)^*](p \to [a_i]p) \to [(a_i)^*]p$
(A3) *K45* axioms for the belief operators:
  (A3.1) $\mathbf{B}_i p \to \mathbf{B}_i \mathbf{B}_i p$
  (A3.2) $\neg \mathbf{B}_i p \to \mathbf{B}_i \neg \mathbf{B}_i p$
(A4) Axioms of correspondence between abstract and concrete actions:
  (A4.1) $[(a \cup b)_i]p \leftrightarrow [a_i \cup b_i]p$
  (A4.2) $[(a;b)_i]p \leftrightarrow [a_i;b_i]p$
  (A4.3) $[(a^*)_i]p \leftrightarrow [(a_i)^*]p$
(A5) An axiom for the informational test operator:
  (A5.1) $[(p?)_i]q \leftrightarrow \mathbf{B}_i(\mathbf{B}_i p \to q)$

Remark 2. Because the substitution rule is one of the inference rules it follows in this axiomatic system that, for example, $[(\alpha)^*]\phi \to \phi \wedge [\alpha]\phi$ is true for all concrete actions α and any formula ϕ, i.e. it is the generalisation the axiom (A2.4). Similarly, for all other axioms.

It is easy to check that the proposed axiomatic system is sound with respect to all *BDL* models. The proof of completeness is omitted due to lack of space.

Theorem 1 (Completeness). *The above axiomatic system is complete with respect to the class of all BDL models.*

Useful extensions of BDL. In some examples of the multi-agent systems, the agents have more intelligence than we have put into *BDL*. Fagin, Halpern and Moses [3] give numerous examples of reasonable extra properties of the agents' beliefs. We consider here two important properties.

Very often a property is required that the agent's beliefs adequately reflect the properties of the system, that is, the agent's beliefs actually become knowledge in this case. In order to accommodate this extra intelligence of agents we need to add the following axiom to *BDL*:

$$(T) \qquad \mathbf{B}_i p \to p.$$

Then $\mathbf{B}_i$ is an *S5* modality and represents the knowledge operator.

Another natural requirement is the consistency of the agents' beliefs. For instance, local copies of the database from a server must be consistent. This property can be formulated by an additional axiom for *BDL* [3].

$$(D) \qquad \mathbf{B}_i p \to \neg \mathbf{B}_i \neg p$$

In modelling intelligent agents a key concern is the connection between informational attitudes and action. The most well-known and natural connection between informational attitudes and action are the properties of no learning and prefect recall (see e.g. [10, 20]). No learning is commonly formulated by the axiom schema

$$(NL) \qquad [a_i]\mathbf{B}_i p \to \mathbf{B}_i [a_i] p,$$

where $\mathbf{B}_i$ normally represents an epistemic operator. The axiom schema says that agent i knows the result of her action in advance; in other words, there is no learning. Perfect recall, also called accordance, is commonly formulated by the axiom schema

$$(PR) \qquad \mathbf{B}_i [a_i] p \to [a_i]\mathbf{B}_i p.$$

It expresses the persistence of the agent's knowledge (sometimes belief) after the execution of an action.

Note 1. It should be noted that the axiom schema *PR* together with *D* has a counterintuitive meaning when p in *PR* is substituted with falsum [9]. In this case the reading of *PR* is: 'If the agent believes she cannot perform an action then she cannot perform this action at all'. This shows that *PR* combined with *D* in *BDL* is quite a strong logic. However, weaker versions of *PR* can be considered to avoid this unwanted property [9].

We consider the following properties on models.

$$(eq) \qquad \forall i \; R_i \text{ is an equivalence relation}$$

$$(br) \qquad \forall i \forall s \exists t \, (s, t) \in R_i$$

$$(com^l) \qquad \forall i \forall \alpha \, (R_i \circ Q(\alpha_i) \subseteq Q(\alpha_i) \circ R_i)$$

$$(com^r) \qquad \forall i \forall \alpha \, (Q(\alpha_i) \circ R_i \subseteq R_i \circ Q(\alpha_i))$$

Main results. The main results in this paper are the following. The proofs use well-known methods of modal and dynamic logic [7, 3, 5, 4], but require non-trivial modifications due to the presence of the new informational test operator.

Theorem 2 (Completeness). *Let L be an extension of BDL by any combination of the axioms T, D, NL, and PR. Then L is complete with respect to the class of all models restricted by the corresponding combinations of the properties* (eq), (br), (com^r), *and* (com^l).

Theorem 3 (Small Model Theorem). *Let ϕ be an arbitrary formula. Suppose n is the number of symbols in ϕ, and m is the number of different agents associated with some test actions in ϕ. If ϕ is satisfiable in a model with (or without) the properties from the list* (eq), (br), (com^r), *and* (com^l), *then ϕ is satisfiable in a (finite) model satisfying the same properties but which has no more than $2^n \cdot (2^{2^n})^m$ states.*

Theorem 4 (Decidability and complexity). *The satisfiability problem in all extensions of BDL by all possible combinations of axioms T, D, NL, and PR, and in BDL itself, is EXPTIME-hard and decidable in 2EXPTIME.*

Note 2. If we do not consider *NL* and *PR* then it is possible to prove the above theorems even without the requirement that the relations R_i are transitive and Euclidean. In other words, the theorems are still true for the logic without axioms (A3). More precisely, let ϕ is an axiom which (i) contains $\mathbf{B}_i$ as the only modal operators, and (ii) has a corresponding first-order frame property P for the relations R_i and P is persistent under the least filtration. Then the axiomatic system, which consists of the axioms (A1), (A2), (A4), (A5) and ϕ as an extra axiom, is (sound and) complete with respect to the class of models with the property P, has the small model property and is decidable in the sense of Theorems 3 and 4.

The axiom schemas *NL* and *PR* should be used with care, since decidability is easily lost. If the condition on actions in Definition 2 that binds actions to the same agent is omitted, then schemas of the form $\mathbf{B}_i\mathbf{B}_j p \to \mathbf{B}_j\mathbf{B}_i p$ are valid. As a consequence, the logic has the axiomatic product of several copies of *S5* embedded in it. It then follows from results in [13] that the logic with *T*, *NL* and *PR* becomes undecidable if there are more than two agents.

Now we define a simulation of (full-test) *PDL* in *BDL* $\oplus$ T. This proves that despite the absence of the standard test operator it is possible to reason about while programs in the logic *BDL*. Select an agent, agent i say, and define a mapping σ from formulae of *PDL* to For by the following.

$$\begin{array}{lll} \sigma p = \mathbf{B}_i p & \sigma\bot = \bot & \sigma a = a_i \\ \sigma(\psi?) = ((\sigma\psi)⁇)_i & \sigma(\alpha \cup \beta) = \sigma\alpha \cup \sigma\beta & \sigma(\alpha;\beta) = \sigma\alpha;\sigma\beta \\ \sigma(\alpha^*) = (\sigma\alpha)^* & \sigma(\phi \to \psi) = \mathbf{B}_i(\sigma\phi \to \sigma\psi) & \sigma([\alpha]\psi) = \mathbf{B}_i[\sigma\alpha]\sigma\psi \end{array}$$

The following theorem shows that the translation σ preserves truth of formulae in both directions.

Theorem 5. *For any PDL-formula ϕ, $\phi \in PDL$ iff $\sigma\phi \in BDL \oplus T$.*

4 Properties of informational test

In this section we show the informational test operator has properties which are generally regarded as good [22].

One of the most important features of a test operation is the fact that it confirms the agent's beliefs. For example, a student has been preparing for an examination and done this thoroughly. But she is not certain yet whether she is prepared really well. As a good student she tests herself on some mock examination exercises. Does this testing add something to her belief? The answer is no, because her beliefs were formed during her preparation. She already believed that she has prepared well and the action of the pre-examination just confirms her beliefs. Thus, the action of examination (testing) can be regarded as making the agent's implicit beliefs explicit. In this respect, we prove that the action of confirming a formula together with the action of confirming the negation of the formula can provide belief (knowledge) for an agent, i.e. following [22], this combined action is an *informative* action. Like [11] we call any action of confirming a formula and its negation an *alternative test.*

Another desirable feature of alternative test is the fact that the truth or falsity of the formula that is tested should not be affected by the actual act of testing, i.e. the outcome of the alternative test should correspond to the state in which the test is performed. This characteristic of alternative tests is called *truthfulness* [22]. In our example, it means that if the student is prepared well and knows a lot about the subject of the examination then she remains well-prepared after the examination.

We also consider the property of *minimal change* for alternative test [22]. This property states that an alternative test must not add or remove formulae from the belief set of an agent which are not relevant to the action of testing. It means that, if the student believed before the examination on a topic of mathematics that Sydney is a city in Australia then after the examination she believes that Sydney is still a city in Australia.

Definition 3. *(i) An abstract action α is* informative *with respect to a formula ϕ in a logic L, if the formula $[\alpha_i](\mathbf{B}_i\phi \vee \mathbf{B}_i\neg\phi)$ belongs to L. (ii) An action α is* strongly informative *with respect to ϕ in L, if α is informative with respect to ϕ in L and the formula $\langle\alpha_i\rangle(\mathbf{B}_i\phi \vee \mathbf{B}_i\neg\phi)$ belongs to L.*

Definition 4. *(i) An abstract action α is* truthful *with respect to a formula ϕ in a logic L, if the formula $(\phi \to [\alpha_i]\phi) \wedge (\neg\phi \to [\alpha_i]\neg\phi)$ belongs to L. (ii) An action α is* strongly truthful *with respect to ϕ in L, if α is truthful with respect to ϕ in L and the formula $(\phi \to \langle\alpha_i\rangle\phi) \wedge (\neg\phi \to \langle\alpha_i\rangle\neg\phi)$ belongs to L.*

Definition 5. *An abstract action α* preserves beliefs *in logic L, if the formula $\mathbf{B}_i\phi \to [\alpha_i]\mathbf{B}_i\phi$ belongs to L for any formula ϕ.*

Proposition 1. *(i) $[(p⁇)_i]\mathbf{B}_ip, [(p⁇ \cup \neg p⁇)_i](\mathbf{B}_ip \vee \mathbf{B}_i\neg p) \in BDL$. (ii) For any formula ϕ, $\langle(p⁇ \cup \neg p⁇)_i\rangle\phi \notin BDL$. (iii) $p \to [(p⁇ \cup \neg p⁇)_i]p \in BDL \oplus T$.*

Theorem 6. *(i) The action $p? \cup \neg p?$ is informative with respect to p but not strongly informative with respect to any formula in BDL. (ii) The action $p? \cup \neg p?$ is truthful but not strongly truthful with respect to p in the logic $BDL \oplus T$ of knowledge. (iii) The action $p? \cup \neg p?$ preserves beliefs in BDL.*

5 Conclusion

The paper proposed a multi-agent logic *BDL* and extensions for reasoning about the actions agents can perform and their informational attitudes. In this paper we focused on belief given by *K45* or *KD45* modalities and knowledge given by *S5* modalities. *BDL* logics provide an abstract way of reasoning about multi-agent systems and provide a basis for the reasoning about multi-agent systems. Many other agent logics which pretend to be real multi-agent logics (for reasoning about actions and beliefs or knowledge) can be represented as extensions of *BDL*. Some axiomatic extensions frequently considered were studied in this paper and results of soundness, completeness, decidability and the finite model property were proved. It turned out that Sahlqvist correspondence [1] does not completely work for *BDL* but nevertheless we could prove the expected model properties for the axioms of permutability of the informational modalities and action modalities (perfect recall and no learning). These extensions of *BDL* are therefore closely related to the products of logics [5]. It was shown in [5] that, usually, results on decidability is hard to obtain for the products and some advanced techniques must be developed for each particular instance. Nevertheless, the decidability for such extensions of *BDL* was shown in this paper. For further work, it would be interesting to find in the framework of *BDL* an appropriate axiomatisation of logics of interpreted systems, hypercubes and full systems as it was done in [14] for reasoning about knowledge and also investigate the decidability of these logics.

The language of the logic *BDL* is very expressive. It is more expressive than *PDL*, and $BDL \oplus T$ is more expressive than known logics of common knowledge. It is therefore surprising that the complexity of the logic does not increase too much. The expressiveness is mainly achieved by introducing dynamics into agent beliefs which is incorporated via the new informational test operator. It was proved that this operator has many advantages in comparison to the usual test operator of *PDL*. The distinction between abstract and concrete actions is also crucial. This property makes it possible to reason about groups of agents rather than just about single agents, i.e. is suitable to express cooperation between agents and teamwork [2]. In this respect, some operators on the set of agents, for instance, an operator of grouping of agents and a pipeline operator (i.e. an operator which forms a sequence of agents), can also be added to represent some kinds of cooperation inside groups of agents. It seems that such operators will not considerably increase the expressivity of the language but can represent some properties of multi-agent systems in more compact notation (very similar to what quantifiers do in quantified Boolean logic with respect to Boolean logic). The decidability of such extended logic needs careful further investigation. Another way to enhance the logic is to add quantifiers over agents. This could lead to undecidability, though it may be hard to prove (the decidability or undecidability of this logic).

References

1. P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge Univ. Press, 2001.
2. P. R. Cohen and H. J. Levesque. Teamwork. *Noûs*, 25(4):487–512, 1991.
3. R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about knowledge*. MIT Press, 1995.
4. M. J. Fischer and R. E. Ladner. Propositional dynamic logic of regular programs. *J. Comput. Syst. Sci.*, 18(2):194–211, 1979.
5. D. Gabbay and V. Shehtman. Products of modal logics, part 1. *Logic J. IGPL*, 6(1):73–146, 1998.
6. J. Y. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54:319–379, 1992.
7. J. Y. Halpern and M. Y. Vardi. The complexity of reasoning about knowledge and time. I Lower bounds. *J. Computer and System Sci.*, 38:195–237, 1989.
8. D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. MIT Press, 2000.
9. A. Herzig, 2002. Personal communication.
10. A. Herzig, J. Lang, D. Longin, and T. Polacsek. A logic for planning under partial observability. In *Proc. AAAI'2000*, pp. 768–773. AAAI Press/MIT Press, 2000.
11. A. Herzig, J. Lang, and T. Polacsek. A modal logic for epistemic tests. In *Proc. ECAI'2000*, Berlin, 2000. IOS Press.
12. A. Herzig and D. Longin. Belief dynamics in cooperative dialogues. *J. Semantics*, 17(2), 2000.
13. M. Kracht. Highway to the danger zone. *J. Logic Computat.*, 5(1):93–109, 1995.
14. A. R. Lomiscio, R. van der Meyden, and M. Ryan. Knowledge in multi-agent systems: Initial configurations and broadcast. *ACM Trans. Computational Logic*, 1(2):247–284, 2000.
15. J.-J. C. Meyer, W. van der Hoek, and B. van Linder. A logical approach to the dynamics of commitments. *Artificial Intelligence*, 113(1–2):1–40, 1999.
16. R. C. Moore. A formal theory of knowledge and action. In J. R. Hobbs and R. C. Moore, editors, *Formal Theories of the Commonsense World*. Ablex, 1985.
17. A. S. Rao. Decision procedures for propositional linear-time belief-desire-intention logics. In *Proc. ATAL'95*, vol. 1037 of *LNAI*, pp. 102–118. Springer, 1996.
18. A. S. Rao and M. P. Georgeff. Modeling rational agents within a BDI-architecture. In *Proc. KR'91*, pp. 473–484. Morgan Kaufmann, 1991.
19. R. A. Schmidt and D. Tishkovsky. On Axiomatic Products of PDL and S5: Substitution, Tests and Knowledge. *Bull. the Section of Logic*, 31(1):27–36, 2002.
20. W. van der Hoek. Logical foundations of agent-based computing. In *Multi-Agent Systems and Applications*, vol. 2086 of *LNAI*, pp. 50–73. Springer, 2001.
21. B. van Linder, W. van der Hoek, and J.-J. C. Meyer. Formalizing abilities and opportunities of agents. *Fundamenta Informaticae*, 34(1, 2):53–101, 1998.
22. B. van Linder, W. van der Hoek, and J.-J.Ch. Meyer. Tests as epistemic updates. In *Proc. ECAI'94*, pp. 331–335. John Wiley & Sons, 1994.
23. M. Wooldridge. Agent-based computing. *Interoperable Comm. Networks*, 1(1):71–97, 1998.
24. M. Zakharyaschev, F. Wolter, and A. Chagrov. *Advanced Modal Logic*. Kluwer, 1998.