An Extended Description Logics Approach to Agent Communication Language Semantics

Riza Cenk Erdur and Inanç Seylan
Ege University, Department of Computer Engineering,
35100 Bornova, Izmir, Turkey
{cenk.erdur, inanc.seylan}@ege.edu.tr

Abstract. The recent efforts by agent developers try to integrate the standard ontology languages of the Semantic Web into their agents towards better interoperability. Similarly, the field of ACL semantics helps to build more flexible agents that can “understand” the acts exchanged. To better integrate such an ontology language with agent communication semantics, this paper proposes a hybrid description logic language with modal extensions for belief and intention. It then describes a practical approach to put the theory into practice.

1 Introduction

According to the vision of Tim Berners-Lee et. al. from a highly quoted article, the Semantic Web will bring structure to the meaningful content of Web pages, creating an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users [1]. This structure, namely ontologies, will help build “a common understanding of the knowledge exchanged” which is listed as a fundamental component for agents to interact and interoperate in [2]. An ontology is defined as a formal explicit description of concepts in a domain of discourse, properties of each concept describing various features and attributes of the concept, and restrictions of slots [3]. The W3C’s ontology language, OWL\(^2\), is a semantic markup language for publishing and sharing ontologies on the World Wide Web.

The Semantic Web standards have drawn the attention of many agent researchers and they’ve integrated some of these standards (especially OWL) into their systems. ITtalks [4] system offers access to information about activities such as talks and seminars related with information technology. ITtalks uses DAML+OIL – a major influence on OWL – for knowledge representation and allows agents to retrieve and manipulate information stored in the ITtalks knowledge base. The smart meeting room system [5] is a distributed system that consists of agents, services, devices and sensors that provide relevant services and information to the meeting participants based on their contexts. This system uses semantic web languages for representing context ontologies. TAGA (Travel

---

1 http://www.w3.org
2 http://www.w3.org/TR/owl-ref/
Agent Game in Agentcities) [6] also utilizes semantic web languages (RDF and OWL) to specify the underlying ontologies. The agents exchange knowledge using OWL as the content language within ACL messages and use FOWL[7] to reason about content represented in OWL. All of these systems extend a multi-agent middleware in their underlying infrastructure. For example, ITtalks uses Jackal [8], smart meeting room system uses Jade [9] and TAGA both Jade and AAP (April Agent Platform) to demonstrate agent interoperability. SEAGENT on the other hand was developed from scratch to provide native support for transferring OWL ontologies in ACL messages and handling of semantic knowledge in agent’s internal architecture [10]. Other features of SEAGENT are match-making of agents in the directory facilitator using a semantic service-matching engine and managing and translation of ontologies in a platform instance through ontology management service. The trend towards agent interoperability shows us that agent researchers will look for more ways to integrate semantic web standards into their agents.

All of the middleware (platform, etc.) listed above use either FIPA ACL or KQML which are agent communication languages based on speech acts. Both of these languages have formal semantics (FIPA Communicative Act Library Specification and [11]). The semantics allows one to resolve ambiguities and helps to build more flexible multi-agent systems. As far as we know, the work by Louis and Martinez [12] is the first framework to facilitate the programming of agents that attempt to conform to the semantics of FIPA ACL. This is a noble approach in that it allows agents to interpret messages by its semantics rather than hardcoding the agents to conform to a limited set of interaction protocols.

Based on the discussion about multi-agent systems that use OWL, it is obvious that they use it as a means to have more flexible and interoperable agents. Similarly, the field of ACL semantics serves to the same purpose. However, the systems we mentioned concentrate on the ontological aspect more than the other although it looks promising with the initial implementations coming out.

In this paper, we focus on using OWL as a content language and discuss why it isn’t sufficient by itself to capture the ACL semantics of BDI agents. As a solution, by taking advantage of OWL’s description logic foundation, we extend it with the modalities belief and intention. The paper is organized as follows. In Section 2, we give an introductory formal background on Description Logics. In Section 3, we discuss OWL as a content language for agent communication. In Section 4, we present a DL based language with modal operators belief and intention. Section 5 concludes the work.

2 Description Logics

2.1 Introduction

Description Logics are a family of class-based (concept-based) knowledge representation formalisms [13]. With a description logic language, one can define
a domain of discourse using concepts and then listing the known facts as individuals. These languages are characterised by the use of various constructors to build complex classes from simpler ones.

Using concept names and roles, it is possible to define complex concepts with the help of operators provided by the language. For example, $\text{Man} \equiv \text{Person} \sqcap \text{Male}$ is the concept of people that are male which is given the symbolic name $\text{Man}$. These concept definitions form the $TBox$ component of a DL KB where the terminological axioms (general knowledge about the problem domain) are kept. The other component, $ABox$ contains the assertional axioms which are about the individuals in the domain of discourse. For example, $\text{Male(John)} \land \text{loves(John, Mary)}$ asserts that John is a male and he loves Mary.

The language $\mathcal{AL}$ (attributive language) is a minimal language. The other languages of this family are extensions to $\mathcal{AL}$. More expressive languages can be obtained by adding other constructors to $\mathcal{AL}$. $\mathcal{ALC}$ is an extension of $\mathcal{AL}$ with the negation of arbitrary concepts. A concept $\neg C$ is interpreted as $(\neg C)^I = \Delta^I \setminus C^I$.

Description logics is a subset of first order predicate logic, therefore any terminological and assertional axiom has an equivalent FOPL formula.

2.2 Description Logics with Modal Operators

Traditional description logic languages represent the knowledge about an application domain by introducing the concepts of that domain and then listing the facts that hold in that particular instance of the domain. Although this is sufficient in many cases, sometimes there arises the need to represent more dynamic knowledge such as beliefs, intentions, actions, time-dependence, etc. This is particularly true in systems that model the intellectual aspects of intelligent agents. Those systems usually use modal logics to represent such notions. Indeed, description logics have a strong connection to modal logics. As proved by [14], the description logic $\mathcal{ALC}$ is in fact a notational variant of the propositional modal logic $K_{(m)}$. In addition to the correspondence between two logics, there have been efforts in the literature to integrate modal operators into description logic languages (see for example [15], [16] and for a survey [17]). [18] lists the properties that determine the design of such a language.

3 OWL as an ACL Content Language

In this section, we evaluate OWL as a content language. In doing so, we reference the work that has also evaluated it and used it.

Botelho et. al. has reviewed content languages that are suitable for agent to agent communication[19]. They came up with three requirements that a content language must satisfy in order to be used with FIPA ACL. These are:

1. It must be capable of representing propositions,
2. It must be capable of representing actions,
3. It must be capable of representing objects, including identifying referential expressions to describe objects.
DAML+OIL, which OWL is based on, is among the languages that were reviewed. There are some several points that are worth to mention of the conclusion about DAML+OIL:

1. DAML+OIL is good for expressing class and individual declarations. Since it is an ontology definition language, it is a good way to define languages. Despite this advantage, a language defined in DAML+OIL needs to have its own semantics for the new terms and operators defined. In case of the agent domain, the new language must provide semantics for actions, beliefs, goals, etc.

2. It is not clear how defined logical operators could be conveniently composed.

3. It does not allow referential expressions. With the introduction of SWRL[20] however, it is now possible to define rules in OWL. However it is not widely adopted in reasoners yet.

Later, the work of Zou et. al.[6] reviewed the list of test cases presented in [19] for OWL. The reasons why they used OWL as a content language can be summarized as follows. First of all, OWL’s expressive power as a knowledge representation language seems to be adequate for many agent-based systems. It is designed to be compatible with the architecture of the World Wide Web in general, and the Semantic Web in particular. Consequently, it has a distributed nature and direct support for URI’s, so terms from multiple ontologies can easily be used. It is a standard of the W3C, hence it has the potential to be widely accepted and used (the last statement can be considered moderate because it seems that OWL is already popular). The acceptance of OWL will enhance the interoperability among many systems. As a final point, returning back to the first sentence of the paper, it is in the vision of the guiding figures of the industry.

The SEAGENT[10] framework first used FIPA RDF0 as the content language of FIPA ACL messages. Then FIPA RDF0 was extended to make use of the OWL’s semantic capabilities over RDF (intersections, unions, restrictions and etc. to name a few)[21]. Carrying domain specific information with an OWL-based content language is easier compared to FIPA SL, KIF or Prolog. Nevertheless, the underlying semantic model of FIPA ACL messages which is given in terms of the mental attitudes (belief, uncertainty, desire, goal, intention) of BDI agents can not be used. For example, the hierarchical task network [22] planner in SEAGENT supports predefined interaction protocols. However, conforming to a limited set of interaction protocols usually results in rigid agents. In case that an agent receives a message outside the scope of a protocol, it matches plans to execute based on the communicative act and other parameters of the ACL message along with the content.

Using OWL (and with it a RDF query language like RDQL or SPARQL) to define the preconditions of plans, one can not take advantage of the precise meaning for communication primitives in FIPA ACL. Even if the mental attitudes can be modelled as OWL axioms, this is verbose and insufficient. It requires additional semantics as stated by the first conclusion of Botelho et. al.’s work.

http://www.w3.org/2004/OWL/
1 is an excerpt from [6] showing how a belief is expressed using OWL. It means that “Steve believes X”. The Belief itself is a concept and such propositions are given inside person (agent) individuals. This usage is rather cumbersome and it raises questions about how multiple levels of modal operators can be applied to propositions.

<Person rdf:ID="steve">
  <hasProposition>
    <Belief rdf:ID="stevebelief1">
      <believe>true</believe>
      <Statement>X</Statement>
    </Belief>
  </hasProposition>
</Person>

Fig. 1. An example of belief expressed using OWL

In the same fashion, the FIPA RDF Content Language\(^6\) assumes that every statement (triple) in a RDF graph indicates the agent’s belief in that statement. To express logical disbelief, it extends rdf:Statement from the reification vocabulary of RDF Schema to include a boolean valued property such as belief. This is again problematic if we want to express greater modal depths.

Finally, one might want to take advantage of description logic languages being fragments of first-order predicate logic and try to translate for instance a concept \(C\) into an equivalent predicate formula \(F_C(x)\) with one free variable \(x\). Unfortunately, no legal well-formed expression in FIPA SL contains a free variable\(^7\), thus ruling out this alternative.

Another aspect of a content language is to allow access to objects matching certain criteria. Identifying referential expressions are used to represent the open questions what, which, who and etc. To support identifying referential expressions and querying the knowledge base, SEAGENT uses an inference engine with a matching algorithm that was a slight modification of a service matching algorithm proposed by Sycara et. al.\(^{23,24}\) This algorithm considers the following relations on a model where \(\text{DoM}(C_1, C_2)\) is the function which determines semantic match degree between concepts, \(C_1\) and \(C_2\):

\[
\begin{align*}
\text{DoM}(C_1, C_2) &= \text{EXACT} \text{ if } C_1 \text{ is a direct subclass of } C_2 \text{ or } C_1 = C_2 \\
\text{DoM}(C_1, C_2) &= \text{PLUG-IN} \text{ if } C_1 \text{ is a distant subclass of } C_2 \\
\text{DoM}(C_1, C_2) &= \text{SUBSUMES} \text{ if } C_2 \text{ is a direct or distant subclass of } C_1 \\
\text{DoM}(C_1, C_2) &= \text{FAIL} \text{ otherwise}
\end{align*}
\]

It was first used in DF to find an agent giving a particular service in \(\text{DoM}\) closeness to the requested service \(^{25,26}\). This is a practical approach and it

---

\(^6\) \url{http://www.fipa.org/specs/fipa00011/}\\
\(^7\) \url{http://www.fipa.org/specs/fipa00008/}
allows querying the KB by choosing exact as the degree of match. However, with this kind of an inference, the user is only allowed to use the predefined four rules.

4 The $\mathcal{ALC}_{BI}$ Language

As we have discussed in Section 3, OWL is not a sufficient content language to be used in popular ACLs (FIPA ACL and KQML) where the semantics of agent communication is explained in terms of the mental attitudes of participants. We gave some examples from the literature of its usage as a content language but these approaches don’t take semantics in the sense of mental attitudes into account. Additionally, trying to express such semantics with OWL is cumbersome and it provides technical problems. These difficulties are inherent from the expressivity limitations of the formalism of OWL, namely description logics. Therefore it must be extended with modal operators that can express agents’ mental attitudes. It is then possible to define agent interaction semantics by listing the axioms of communication. Here, one can adopt an existing framework of communication such as FIPA. However only description logic formulas are allowed in modal expressions.

The language that we’ll define is an extension of $\mathcal{ALC}$ that includes modal operators for belief and intention. We name it $\mathcal{ALC}_{BI}$. OWL is based on $\mathcal{SH}$ family of description logics. Our approach can be expanded to it as well. The $\mathcal{SH}$ family is basically $\mathcal{ALC}$ extended with transitive properties and a property hierarchy[27].

$\mathcal{ALC}_{BI}$ is used for defining the formal semantics of communicative acts (in terms of feasibility precondition and rational effect). Similar to FIPA SL, it is also designed as a content language for use in conjunction with FIPA ACL. Note that the logic under consideration has neither dynamic nor temporal characteristics (see [28] for a survey of temporal extensions of description logics and [14,29] for the correspondance between description logics and propositional dynamic logics). Being such, this might raise e.g. the question of how actions and accordingly requests can be modelled. We think of actions as entities in the domain of an executor (for instance a HTN planner), therefore instances of an action concept are exchanged between agents in order to request for acts to be done. Accordingly, it is not a full agent programming language in the sense that it allows agent plans to be defined. Writing plans is still done with a lower level programming language.

Ideally, to interpret the formulas of the language, the tableaux algorithms for standard DLs must be expanded to include modal descriptions. Nevertheless, this is not the way we prefer to implement the interpretation of formulas of the language. Our approach is similar to [12] in which we try to approximate the communication semantics of a framework we chose. The modal operators are used to define the rules of introspection of an agent and cooperation between agents. Inside modal operators are definitions and assertions in OWL, thus satisfying our goal of representing domain-dependent knowledge with it. The rationale for
such a design is that OWL is a web standard and there are many tools (being) developed for it. Therefore we want to make use these tools in order to reduce the development time.

In the rest of this section, in accordance with the order we gave in the first paragraph, we will first give the syntax and semantics of our hybrid language so that it’s formally defined. Then we will discuss the framework of communication by listing the set of axioms for relations between mental attitudes and for cooperation of agents. Finally we will explain the implementation.

4.1 Syntax and Semantics

First we give the syntax of this multi-dimensional DL with modal operators. The alphabet of the language is defined as follows.

Definition 1. As for standard DL, we assume a set of concept names \( \{C_0, C_1, \ldots \} \), role names \( \{R_0, R_1, \ldots \} \), and a set of object names \( \{x_0, x_1, \ldots \} \) to be given. \( \top \) and \( \perp \) denote top and bottom concepts respectively. \( \land \) and \( \neg \) represent standard logical connectives. \( C \rightarrow D \) is an abbreviation for \( \neg (C \land \neg D) \). Let \( AG = \{i, j, k, \ldots \} \) be the set of agents. Then to every \( i \in AG \), the modal operators of belief \( B_i \) and intention \( I_i \) are associated.

Concept descriptions are defined as follows.

Definition 2. All concept names, as well as \( \top \) and \( \perp \) are concepts. If \( C \) and \( D \) are concepts, and \( R \) is a role name then a) \( \neg C \) (concept negation), \( C \land D \) (concept conjunction), and \( C \lor D \) (concept disjunction) b) \( \forall R.C \) (value restriction) and \( \exists R.C \) (exists restriction) are concepts.

New formulas are introduced according to the definition below.

Definition 3. Let \( C \) and \( D \) be concepts, \( R \) a role name, \( x \) and \( y \) object names and \( m \) a (possibly empty) sequence of modal operators from \( \{B, I\} \times AG \). Then axioms of the form \( C = D \) (terminological), \( R(x, y) \) and \( C(x) \) (assertional) are atomic formulas. If \( \varphi \) and \( \psi \) are formulas then so are \( m \varphi \), \( \neg \varphi \), and \( \varphi \land \psi \).

A formula \( B_i \varphi \) is read “agent \( i \) believes \( \varphi \)”. The formula \( I_i \varphi \) is read “agent \( i \) intends \( \varphi \)”. The modal operators will be interpreted by “possible worlds” semantics using an extended multi-relational Kripke model.

Definition 4. A Kripke model \( M = \langle W, \Gamma, K_i \rangle \) consists of a set \( W \) of possible worlds, a set of accessibility relations on the worlds in \( W \) (in order to model beliefs and intentions), and a \( K \)-interpretation \( K_i \) over \( W \).

\( \Gamma \) contains for every \( i \in AG \) a) an accessibility relation \( \gamma_{B_i} \), which is a function \( \gamma_{B_i} : W \rightarrow 2^W \) and b) a function \( \eta_i : W \rightarrow 2^{2^W} \).

The \( K \)-interpretation \( K_i \) consists of a domain \( \Delta^{K_i} \) and an interpretation function \( \cdot^{K_i} \). \( \Delta^{K_i} \) is the union of non-empty domains \( \Delta^{K_i(w)} \) for all worlds \( w \in W \). The interpretation function \( \cdot^{K_i} \) associates with each \( w \) a structure...
where $\Delta^{K_i(w)}$ is the domain of $w$, $R^{K_i(w)}$ are binary relations on $\Delta^{K_i(w)}$, $C^{K_i(w)}$ subsets of $\Delta^{K_i(w)}$, and $x^{K_i(w)}$ are objects in $\Delta^{K_i(w)}$.

The relation $\gamma_B$, is transitive, serial and Euclidean. The belief operator of FIPA SL has the same logical model\(^8\). In contrast to the formal model of FIPA where intention is explained in terms of the goals of an agent, here it is a primitive mental attitude. $\eta_i$ is a neighborhood function of classical modal logics with the only rule:

$$RE : \frac{\varphi \leftrightarrow \psi}{I_i \varphi \leftrightarrow I_i \psi}$$

The operator $I_i$ can be translated into normal modal formula $\neg I_{i,1}\neg(I_{i,2}\varphi \land I_{i,3}\neg \varphi)$ where $I_{i,1}$, $I_{i,2}$ and $I_{i,3}$ are normal modal operators\([30]\). Hence $I$ contains the three accessibility relations for intention $\gamma_{I,1}, \gamma_{I,2}, \gamma_{I,3}$ along with $\gamma_B$, for each $i \in \mathcal{A}$. The reason that we provide a neighborhood semantic for intentions is because of the formal framework of communication we chose that will be explained in the following two subsections.

The interpretation function is extended to concept descriptions by the following inductive definitions:

$$\top^{K_i(w)} = \Delta^{K_i(w)}$$
$$\bot^{K_i(w)} = \emptyset$$
$$(C \sqcap D)^{K_i(w)} = C^{K_i(w)} \cap D^{K_i(w)}$$
$$(C \sqcup D)^{K_i(w)} = C^{K_i(w)} \cup D^{K_i(w)}$$
$$(-C)^{K_i(w)} = \Delta^{K_i(w)} \setminus C^{K_i(w)}$$
$$(\forall R.C)^{K_i(w)} = \{x \in \Delta^{K_i(w)} | \forall y.(x, y) \in R^{K_i(w)} \rightarrow y \in C^{K_i(w)} \}$$
$$(\exists R.C)^{K_i(w)} = \{x \in \Delta^{K_i(w)} | \exists y.(x, y) \in R^{K_i(w)} \land y \in C^{K_i(w)} \}$$

**Definition 5.** For a model $M = \langle W, \gamma_B, \gamma_{I,1}, \gamma_{I,2}, \gamma_{I,3}, K_I \rangle$ and a world $w \in W$, a formula $\varphi$ is satisfied (written as $(M, w) \models \varphi$) in the following way:

$$(M, w) \models C = D \text{ iff } C^{K_i(w)} = D^{K_i(w)}$$
$$(M, w) \models C(x) \text{ iff } x^{K_i(w)} \in C^{K_i(w)}$$
$$(M, w) \models R(x, y) \text{ iff } (x^{K_i(w)}, y^{K_i(w)}) \in R^{K_i(w)}$$
$$(M, w) \models B \varphi \text{ iff } (M, v) \models \varphi \forall u.(w, v) \in \gamma_B$$
$$(M, w) \models I \varphi \text{ iff } (M, v) \models \varphi \forall u.(w, v) \in \gamma_I$$
$$(M, w) \models \neg \varphi \text{ iff } (M, w) \not\models \varphi$$

---

8 http://www.fipa.org/specs/fipa00037/
4.2 Axiomatization

In this section, we define the axioms holding in the language. We adopt the work of Herzig et al. in which they try to obtain a minimal logic of intention that can be mechanized in a simple way [31]. The aim of the authors is to define the simplest dynamic doxastic logic. For this reason, their semantics is easier to implement than FIPA ACL’s.

The FIPA ACL semantics has some severe problems which limit its practical applicability. [32] is a good discussion of these problems wherein the authors share their experience on trying to implement such semantics and examine how the semantics for some standard communicative acts can be improved. For the interactions between agents in a multi-agent system though, the paper advises that building agents that implement the formal ACL semantics (using agents’ mental states) is not a good way to start. This idea has been discussed by Singh [33]. He emphasizes social agency where agent communication depends on the agent’s social context over mental agency. While we don’t disagree with this, we focus on already implemented systems e.g. using FIPA ACL and their improvement using the Semantic Web ontology languages.

Also in their work, Herzig et al. use assertive speech acts which serve to make an assertion that, in the speaker’s belief, some proposition is true. They show that the request communicative act and yes-no questions can be inferred from literal communicative acts indirectly. In our opinion, assertive speech acts are helpful because by using them agents only inform each other about their mental states. They do not need to know the explicit semantic specification of all the communicative acts. Agents deduce these acts by the communication axioms.

The relations between mental attitudes is given below.

\[
\begin{align*}
B_i(\varphi \rightarrow \psi) & \rightarrow B_i\varphi \rightarrow B_i\psi \quad (K) \\
B_i\varphi & \rightarrow \neg B_i\neg\varphi \quad (D) \\
B_i\varphi & \rightarrow B_iB_i\varphi \quad (4) \\
\neg B_i\varphi & \rightarrow B_i\neg B_i\varphi \quad (5) \\
I_i\varphi & \rightarrow B_i\neg\varphi \quad (A1) \\
B_i\varphi & \rightarrow \neg I_i\varphi \quad (T1) \\
\neg B_i\varphi & \rightarrow \neg I_i\neg B_i\varphi \quad (T2) \\
(I_iB_i\varphi \land B_i\neg\varphi) & \leftrightarrow I_i\varphi \quad (A2) \\
I_i\varphi & \rightarrow I_iB_i\varphi \quad (A3) \\
B_iI_i\varphi & \leftrightarrow I_i\varphi \quad (A4) \\
B_i\neg I_i\varphi & \leftrightarrow \neg I_i\varphi \quad (A5)
\end{align*}
\]

Axiom (A1) defines the simplest relation between intention and belief. Thus, if agent \( i \) intends \( \varphi \), then it believes that \( \neg \varphi \) holds. In other words, it drops its intention to achieve \( \varphi \) as soon as it believes that \( \varphi \) holds.

Axiom (A2) expresses that if agent \( i \) intends to believe \( \varphi \) but it believes \( \neg \varphi \) at the same time, then it should act in order to change the world, thus intending \( \varphi \).

Axiom (A3) means that an agent can not intend \( \varphi \) without intending to believe \( \varphi \). The reverse implication does not hold. An agent can intend to believe
without intending \( \varphi \). This can also be seen in axiom (A2) in which both believing \( \neg \varphi \) and intending to believe \( \varphi \) leads to intending \( \varphi \).

Axioms (A4) and (A5) are analogous to positive and negative introspection for belief respectively. Hence an agent is aware of its intentions and non-intentions.

Theorem 1 (T1) means that an agent does not intend \( \varphi \) if it believes \( \varphi \) already holds.

Theorem 2 (T2) states that an agent does not intend something that it does not believe.

4.3 Cooperation Principles

Consistent with the aim of a simpler logic, the cooperation among agents is explained in terms of two principles: belief adoption and intention generation.

Belief adoption In order to constrain the adoption of any belief that has been uttered by another agent, Herzig et. al. use the notion of competency. Basically, an agent adopts \( i \)'s belief if it believes that \( i \) is competent at that belief. \( i \leadsto \varphi \) means that \( i \) is competent at \( \varphi \). Using this relation, they formulate the following axiom:

\[
B_i \varphi \rightarrow \varphi \text{ if } i \leadsto \varphi \land \text{md}(\varphi) = 0
\]

where \( \text{md}(\varphi) \), modal depth of a formula \( \varphi \), means the maximal depth of nested modal operators in \( \varphi \). If \( \text{md}(\varphi) = 0 \), then the formula \( \varphi \) is said to be objective.

Remark 1. If \( \langle i, j, B_i \varphi \rangle \) is the speech act that has been performed and \( \varphi \) is a formula in an OWL ontology \( O \) and \( j \) understands \( O \) (e.g. can process individuals of a class from \( O \) ), then \( B_i \varphi \rightarrow \varphi \) holds without further inspection on \( i \)'s competence.

Intention generation These principles explain how an agent e.g. \( i \) can adopt the intention of another agent e.g. \( j \) in order to satisfy its (\( j \)'s) goals. They are related to the basic axiom, (A1), in that \( i \) should only generate the intention that \( \varphi \) if it believes \( \neg \varphi \).

\[
\begin{align*}
(B_i I_j \varphi \land \neg B_i \varphi \land \neg I_i B_i \neg \varphi) & \rightarrow I_i B_i \varphi \quad (G11) \\
(B_i I_j \varphi \land B_i \neg \varphi) & \rightarrow I_i \varphi \quad (T3) \\
B_i I_j \varphi & \rightarrow I_i B_j \varphi \quad (T4)
\end{align*}
\]

Axiom \( (G11) \) is the main principle of intention generation. It expresses that when \( i \) knows that \( j \) intends something \( i \) has no idea about, \( i \) should first intend to believe it.

It is then according to the result of this intention (\( B_i \neg \varphi \) or \( B_i \varphi \) ) that \( i \) can choose to intend \( \varphi \) (T3) or intend to make believe \( j \) about it (T4).
4.4 An Operational Model

Although we’ve formalized the semantics of $\text{ALC}_{BI}$, we don’t use it in an explicit way. Our goal being to use ontologies in the ACL semantics, we wanted to come up with an operational model that would allow us to handle $\text{ALC}_{BI}$ formulas of reasonable complexity. This would in turn help process the semantics of communicative acts. Acts are defined in terms of two semantic features: feasibility precondition (FP) and rational effect (RE). The former characterizes the conditions that have to be satisfied for the act to be planned and the latter the reasons for which the act is selected\(^9\). When an agent intends to achieve the RE of an act, it will then elect to perform that act. Whenever the agent elects to perform an act, it intends to satisfy the act’s FP. In other words, it generates $I_iB_i\varphi$ for the formula $\varphi$ given in the FP. This intention seeks the satisfiability of the formula $\varphi$ in the agent’s knowledge base. If a formula $\varphi$ is not satisfied, then by axiom (A2), the agent will have the intention that $\varphi$. This will then lead to the planning of the act whose RE is $I_i\varphi$. If an intention of the kind $I_i\varphi$ fails, then the action also fails.

When an agent observes a communicative act however, it has to take the cooperation principles into account. It first generates the sincerity precondition (corresponds to the ability precondition in FIPA terminology) of the speech act. Because our speech acts are of assertive type, the sincerity precondition describes the hearer’s belief that the utterer believes what it has asserted. For example, assume that the speech act that has just been performed is $\langle i,j,\varphi \rangle$, $\varphi$ being an objective formula. The sincerity condition of this act is $B_jB_i\varphi$. If $i \sim \varphi$, then agent $j$ will adopt $i$’s belief, hence we get the formula $B_j\varphi$. As we mentioned before, the request communicative act and yes-no questions can be inferred from literal communicative acts indirectly. In those cases, the hearer generates an intention to satisfy the uttered intention. Let $\langle i,j,I_i\varphi \rangle$ the speech act that has been performed. Then the sincerity condition takes the form of formula $B_jB_iI_i\varphi$. This simplifies to $B_jI_i\varphi$ by axiom (A4). Hereupon, agent $j$ should decide which intention to generate (using intention generation axioms) by considering its belief about $\varphi$. If the agent never remembers about $\varphi$, this will always result in a research of $i$ in its knowledge base according to axiom ($G_{I1}$). Later, the result of $I_jB_j\varphi$ will determine which axiom to choose (T3 or T4).

To summarize, we came up with the following five types behavior that the interpretation process exerts. Figure 2 depicts how communicative acts and formulas are processed.

1. Belief checking: It corresponds to $B_i\varphi$ and means that the agent $i$ consults its knowledge base whether $B_i\varphi$ or $B_i\neg\varphi$.
2. Rational behavior: When the agent intends the RE of an act, then it performs the act.
3. Sincerity condition generation: The agent $i$ asserts $B_iB_j\varphi$ after it receives the speech act $\langle j,i,\varphi \rangle$.

\(^9\) [http://www.fipa.org/specs/fipa00037/](http://www.fipa.org/specs/fipa00037/)
4. Belief adoption: For agent i to adopt the belief of agent j at a formula, i first decides whether j is competent at that formula.
5. Intention generation: It allows an agent to decide how to adopt the intention of another agent by considering intention generation axioms.

![Fig. 2. An operational model for agent interaction semantics](image)

### 4.5 An Example

In this scenario, John is Mary’s boyfriend and he wants to take her out on Saturday night to a concert. First we define the acts that John is capable of performing.

<table>
<thead>
<tr>
<th>Act:</th>
<th>( (j, m, I_j(\text{Action}(\text{arrangeDate}) \land \text{done}(\text{arrangeDate}, \text{true})) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Desc.:</td>
<td>John’s intention to arrange a date with Mary results in this speech act which is indirectly interpreted as a request by Mary.</td>
</tr>
<tr>
<td>Input:</td>
<td>Appointment(?apMary)\land appointment(arrangeDate, ?apMary)</td>
</tr>
<tr>
<td>Input:</td>
<td>DateTimeDuration(?d)</td>
</tr>
<tr>
<td>FP:</td>
<td>( \varphi_1 = \text{actor}(\text{apMary}, m) \land \text{duration}(\text{apMary}, ?d) \land \text{state}(\text{apMary}, \text{idle}) )</td>
</tr>
<tr>
<td>RE:</td>
<td>done(arrangeDate, true)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Act:</th>
<th>( (j, m, I_j \text{Bif}_j \varphi_1) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Desc.:</td>
<td>When John wants to know whether Mary is idle during the concert time, he asserts so to Mary.</td>
</tr>
<tr>
<td>FP:</td>
<td>( \top )</td>
</tr>
<tr>
<td>RE:</td>
<td>\text{Bif}_j \varphi_1</td>
</tr>
</tbody>
</table>


The response by Mary to the second speech act of John will be Mary’s belief about her status during the concert time. Hence John knows that Mary is competent about her own status during a period of time.

<table>
<thead>
<tr>
<th>Comp. Formula</th>
<th>$m \rightsquigarrow \varphi_1$</th>
</tr>
</thead>
</table>

The following is an action of Mary:

<table>
<thead>
<tr>
<th>Act:</th>
<th>$\langle m, j, \varphi_1 \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Desc:</td>
<td>This act allows Mary to inform John about her status only if he’s her boyfriend.</td>
</tr>
<tr>
<td>FP:</td>
<td>$Contact(j) \land hasSpouse(j,m)$</td>
</tr>
<tr>
<td>RE:</td>
<td>$B_j \varphi_1$</td>
</tr>
</tbody>
</table>

John’s intention to arrange a date with Mary matches with the RE of a communicative act in his action list. Then it intends to believe the feasibility condition of the act, hence $I_j Bi_f j \varphi_1$. This intention is a standard action, which results in the satisfiability checking of the formula $\varphi_1$. If $\varphi_1$ was satisfied, then the agent would execute the action. However, it matches with the RE of another act. Therefore, John directly performs it because its precondition is always satisfied.

After Mary receives the speech act $\langle j, m, I_j Bi_f j \varphi_1 \rangle$, it assumes that John is sincere by asserting $B_m B_j I_j Bi_f j \varphi_1$. By the agent’s introspection about its intentions, it simplifies to $B_m I_j Bi_f j \varphi_1$. Then by (G1) (also assuming that an agent has an assertion of the form $\neg B_m \varphi_1$), we have $I_m Bi_f m \varphi_1$. Again this results in the satisfiability checking of the formula $\varphi_1$. It is satisfied so, according to axiom (T4), Mary now has the intention to make John believe $\varphi_1$. This intention matches with the RE of a speech act, and the FP of it is satisfied so Mary performs the speech act.

Upon receiving it, John generates the formula $B_j B_m \varphi_1$ by trusting Mary’s sincerity. Mary is known to be competent at formula $\varphi_1$ so he adopts her belief which satisfies the FP of the initial plan that the action arrangeDate to be done.

5 Conclusion

The Jade Semantic Add-On [12] is the first implementation attempt towards an agent communication oriented middleware. In this paper, we wanted to give a similar perspective by starting from the idea “what if our agents used ontologies?” We argued that OWL by itself is not enough to model the mental attitudes of agents. This led us to define a new description logic language with modal extensions.

The next step for us is to implement the operational model we proposed using our agent middleware SEAGENT. This will provide us with an evaluation of this work. Currently agent plans are written according to an HTN planner. Based on our experience with students using it in their projects, it is not a trivial task to define provisions and outcomes, and then connect these plans to others. We believe the new approach will abstract this process in the favor of better communicating agents.
References

67608 Kaiserslautern
Germany (1994)


