Integrating Ontology Negotiation and Agent Communication

Marlo Souza[1], Alvaro Moreira[1], Renata Vieira[2], and John-Jules Ch. Meyer[3]

[1] Federal University of Rio Grande do Sul - UFRGS
Porto Alegre, Brasil
{marlo.souza, alvaro.moreira}@inf.ufrgs.br
[2] Pontifical Catholic University of Rio Grande do Sul - PUCRS
Porto Alegre, Brasil
renata.vieira@pucrs.br
[3] Utrecht University
Utrecht, The Netherlands
j.j.c.meyer@uu.nl

Abstract. Ontologies are considered a necessary ingredient for communication among heterogeneous agents in the Web. With the multiplication of ontologies for the same domains, semantic interoperability has become a challenge. In this work, we study the use of ontology negotiation in a agent communication mechanism for agents with ontological reasoning. The resulting communication mechanism allows agents to exchange not only factual but also terminological knowledge about an individual domain and is closely related to available mechanisms in the literature such as KQML and FIPA-ACL.

Keywords: Ontologies for agents, Agent Communication, Ontology negotiation

1 Introduction

It is commonly accepted that two essential ingredients for the construction of the Semantic Web are the use of ontologies and autonomous agents. Despite that fact, the integration of ontology-based reasoning in the semantics of communication mechanisms for multiagent systems has just recently became the focus of attention.

As different ontologies arise to describe the same domain, achieving *semantic interoperability* became essential to allow communication among agents in the Web. Various methods have been proposed to solve this problem in the area commonly known as Ontology Mediation [1].

The most popular approach to ontology mediation is ontology matching [2]. While it is a prolific area with mature methods, matching methods are static, i.e. the alignments are established before the agents' interactions. A dynamic alternative is to centralize ontology mediation in ontology agents [3]. The centralization required by this approach, however, is not easily scalable.

Seeking dynamic and decentralized ways to ontology mediation, [4] proposed the notion of ontology negotiation. In this approach, the agents resolve their communication problems by negotiating a conversation vocabulary among themselves. We believe this approach is particularly interesting since it includes agent communication as an integral part of ontology mediation.

The main contribution of this work is a proposal to embed ontology negotiation into a speech act-based communication mechanism for multiagent systems. We introduce a mechanism to allow communication between agents with different private vocabularies that can exchange terminological information, based on the work of [5]. We will, however, drop the requirement of classifiers and thus generalize the conditions for meaningful communication between heterogeneous agents.

This work is structured as follows: in Section 2 we discuss the related work; in Section 3 we define the central ideas of the work of approximate translation and information loss and Section4 establish formal conditions for proper communication between agents and algorithms for computing translations; in Section 5 we present a mechanism for ontology-based communication that allows ontology negotiation. We conclude the paper with some considerations about the problems that arise from integrating negotiation and communication in the language and the limitations of our technique.

2 Related Work

In [6] a framework for heterogeneous multiagent systems supporting ontology-based communication was defined. The framework allows agents to have different ontologies, and they implement an ontology service in which all the ontologies must be registered. The technique only allows assertional exchange and is centralized, by the use of ontology agents.

From the ontology negotiation point of view, [4] presents a protocol that allows agents to exchange parts of their terminologies and to interpret received messages. The communication mechanism, however, is very restricted.A similar approach but focused mainly on inductive methods used to locate similar concepts between the agents' terminologies is given in [7].

A method for semantic interoperability between taxonomies in peer-to-peer systems is presented in [8]. While their approach is very similar to ours, their work is limited to taxonomies, and they require a complete knowledge of the extension of the concepts.

The work of [5] presents different communication protocols for two agents to establish a communication vocabulary, by exchanging parts of their ontologies. They focus on how to preserve the extensional meaning of the concepts between different but jointly consistent ontologies. The main limitation of in [5] is that they assume that every agent knows the complete extension of their concepts - by the use of classification functions. In this way, for each individual a in the system and for each concept C and agent has in her terminology, the agent is able to decide whether a is in the extension of C, i.e. $C(a)$ holds, or if it doesn't. Complete knowledge, however, is not a realistic property for most applications.

We generalized the notions of lossless communication of [5] for agents with incomplete information about their domain. Also, we integrate their terminological negotiation protocol within a communication mechanism for multiagent systems, using the language AgentSpeak-DL to do so.

3 Approximated Translation and Information Loss

Our mechanism relies on two central notions: *approximate translation* and *information loss*. Informally, approximate translations are functions that transform a formula φ_s written using concepts of the speaker's ontology into a formula φ_h using concepts of the hearer's ontology in a way that preserves meaning. These translation functions are dynamically constructed by the agents when they need to communicate. All these actions - sending a message, explaining the meaning of a concept, receiving a translation - will be incorporated in a mechanism in Section 5. For now, we will present the core ideas of translations and approximate translation, and of information loss.

From here on, we assume familiarity with Description Logics (DL). Capital letters such as A, B, C, C', etc. represent concept names while lowercase letters as a, b represent individuals. The uppercase Greek letter Φ_C will usually denote the set of concept names of an ontology, Similarly, Φ_R the set role names in an ontology. We will call $\mathscr{B}_{\Phi_i}$ the set of atomic concept in the ontology $\mathscr{O}_i$ - which is dependent on the DL chosen. For DL-Lite [9] the foundational language of the OWL 2-QL profile, for example, it includes the concepts $\exists R$ for every role name $R \in \Phi_R$, besides the concept names.

We will not fixate a particular Description Logic, but only require that the negation of atomic concepts $\neg B$ to be representable in it. We are aware that some Description Logics do not allow atomic negation, such as the important *EL* family [10] used as the foundation for the OWL2 - EL profile. Notice, however, that several important Description Logics, such as the DL-Lite family, SHOIN and SROIC, used to defined the semantic of parts of the Ontology Web Language, allow these constructions. We leave for future developments to extend our methods for DLs without atomic negation.

We require that the ontologies used by the agents are expressible in the same Description Logic. A more general setting would require the formal machinery of translations between logics [11], which is outside the scope of this work. We also require that their sets of concept names are disjoint. Concepts with the same name in different ontologies can be differentiated by the use of namespaces.

In our mechanism, each agent i has a concept translation function T_{ij}, for each agent j in the system. A concept translation function $T_{ij} : \mathscr{B}_{\Phi_i} \rightarrow \mathscr{B}_{\Phi_j}$ is a partial function that maps a concept $C_i \in \mathscr{B}_{\Phi_i}$ to a concept $C_j \in \mathscr{B}_{\Phi_j}$. A similar function may be defined for role names $Tr_{ij} : \Phi_{R_i} \rightarrow \Phi_{R_j}$. For sake of space we will not include translation of roles, but we point out that for expressible DLs, such as *SROIC* [12] that bases the OWL 2 language, deciding translation for roles may be achieved using the same techniques as proposed below for concepts. We represent that the translation of C is undefined by $T_{ij}(C) = \bot$.

When a translation for a concept C in ontology $\mathscr{O}_i$ is undefined, the agent i will navigate in the hierarchy of concepts $\mathscr{O}_i$ searching for a concept $C' \neq C$ which is both "closer" to C in the hierarchy and is also translatable according to T_{ij} (i.e. $T_{ij}(C') \neq \bot$). We call such a concept C' a most specific super concept of C, translatable according to a concept translation function T_{ij}.

Definition 1 *A most specific super concept of a concept $C \in \mathscr{O}_i$ translatable according to a concept translation function T_{ij} is an atomic concept $C' \in \mathscr{O}_i$ satisfying the following properties:*

i) $C' \neq C$, $\mathscr{O}_i \models C \sqsubseteq C'$, $T_{ij}(C') \neq \bot$, *and*
ii) for all $C'' \in \mathscr{O}_i$, *if* $C'' \neq C$, $\mathscr{O}_i \models C \sqsubseteq C''$, *and* $T_{ij}(C'') \neq \bot$ *then* $\mathscr{O}_i \not\models C'' \sqsubset C'$

We call $mstsc(\mathscr{O}_i, T_{ij}, C)$ *the set of most specific super concepts of* C *translatable according to a concept translation function* T_{ij}

By (i) in the definition above, C' is a superconcept of C, translatable w.r.t T_{ij}, while (ii) states that C' is most specific than any other concept C'' that is also a superconcept of C and translatable w.r.t T_{ij}.

Note that a most specific super concept always exists since axiom $C \sqsubseteq \top$ always holds, and we require that the concepts $\top$ and $\bot$ are always translated to themselves. Also, it might not be unique. An algorithm for computing the set of most specific super concepts of a concept C translatable according to a concept translation function T_{ij}, denoted by , is presented in Figure 1.

```
Algorithm  $mstsc(\mathscr{O}_i, T_{ij}, C)$
Input :
    ontology $\mathscr{O}_i$ ,
    translation function $T_{ij}$ ,
    atomic concept $C$
Output :
    Set $S$ of mstsc of the concept $C$
  [1]  $S := \{C' \in \mathscr{B}_{\Phi_i} \mid C' \neq C, \mathscr{O}_i \models C \sqsubseteq C'$ and $T_{ij}(C') \neq \bot\}$
  [2]  $S' := S$
  [3]  for each $C' \in S'$
  [4]      for each $C'' \in S'$
  [5]          if $\mathscr{O}_i \models C'' \sqsubset C'$ then
  [6]              $S := S \setminus \{C'\}$
  [7]  return $S$
```

Fig. 1. Algorithm for computing the most specific translatable superconcepts of a concept

In the algorithm depicted in Figure 1, line 1 computes all atomic concepts that satisfy condition i) in the Definition 1. In the loop (lines 3 to 6), the concepts that are not most specific, (i.e. that violates condition ii in Defintion 1) are discarded. We can now define the notion of approximated concept translation.

Definition 2 *Let* $T_{ij} : \mathscr{B}_{\Phi_i} \to \mathscr{L}(\mathscr{B}_{\Phi_j})$ *be a concept translation function, and* $\mathscr{O}_i$ *the agent's ontology. We define the approximated concept translation function* $\overline{T_{ij}} : \mathscr{B}_{\Phi_i} \to \mathscr{B}_{\Phi_j}$ *as:*

$$\overline{T_{ij}}(C) = \begin{cases} T_{ij}(C) & \textit{if } T_{ij}(C) = \bot \\ \overline{T_{ij}}(C') & \textit{otherwise, w/ } C' = \mathsf{sel}(mstsc(\mathscr{O}_i, T_{ij}, C)) \end{cases}$$

The approximated translation of an atomic concept C is the concept translation of it, if it is defined, or it is the approximated concept translation of the most specific superconcept of C.

In the definition above, sel is a selection function on the set of most specific translatable superconcepts. A priori, unless by a specificity of the domain, every candidate in the set $mstsc(\mathscr{O}_i, T_{ij}, C)$ is equally desirable to be selected. In this case, we can use a lexicographic order, for instance. In the cases the domain or system engineer provides reasons to prefer one superconcept over another one, these reasons may be used.

Since most agent communication mechanisms in the literature [13–15] rely on communication of ground atomic facts, we will not deal with complex formulas. Our mechanism will be based only on instantiations of roles and concept literals, i.e. atomic concepts and their negations. Notice however that, given that the ontologies of the agents are defined over a same Description Logic, it is easy to lift the translation of atomic concepts to complex concept formulas by preserving the syntactic structure. The definition below may be, thus, generalized to be applicable in more expressive DLs.

Definition 3 *Let $C(a)$ be a concept instantiation, where C is concept literal formula in the vocabulary of the agent i. The translation of formula $C(a)$ to the terminology of agent j, represented by $\overline{T_{ij}}(C(a))$, is given by the following approximate formula translation function:*

$$\overline{T_{ij}}(C(a)) = \begin{cases} T_{ij}(C)(a)[Ex] & \textit{if } T_{ij}(C) \textit{ is defined} \\ \overline{T_{ij}}(C)(a) & \textit{otherwise} \end{cases}$$

In the above definition, we use an annotated formula $T_{ij}(C)(a)[Ex]$ to point out explicitly that this is the product of an exact translation - this will be important later when we define information loss. From a logical point of view, these annotations have no interpretation.

When an agent i needs to communicate an information $C(a)$ to an agent j she will first translate $C(a)$ into j's terminology, using the function in Definition 3, and then send the message with the resulting formula.

Notice that an agent j can identify if an information $C_j(a)$ received from agent i is the result of approximated translation or not by the use of the $[Ex]$ annotation in Definition 3. Thus, when receiving an approximated translation $C_j(a)$, she can reason whether the loss of information inherent in the approximation process is relevant in her terminology. This case happens when there is a subconcept C' of C_j for which no translation has been established, and thus is a candidate for a better translation of the original information.

We say that an agent i with ontology $\mathscr{O}_j$ and translation function T_{ij} satisfies $\textit{InfLoss}(at)$, i.e. $\langle \mathscr{O}_j, T_{ji} \rangle \vDash \textit{InfLoss}(at)$, when j detects a (possible) loss of information on receiving information at from agent j.

Definition 4 *We say the agent possessing ontology $\mathscr{O}_i$ and translation function T_{ij} detects an information loss when receiving an atomic formula at from agent j iff*

$$\langle \mathscr{O}_j, T_{ji} \rangle \vDash \textit{InfLoss}(at) \;\; \textit{iff} \;\; at \neq C(a)[Ex] \; \underline{\textit{and for some term } t} \\ \exists B \in \mathscr{B}_{\Phi_j} \; \textit{s.t.} \; \overline{T_{ji}}(B(t)) = \overline{T_{ji}}(at)$$

When an agent detects an information loss, she can ask the speaker to clarify the information. This action will result in the speaker introducing new concepts in the communication, i.e. explaining to the hearer the meaning of the concept the speaker intend to use in the communication. Knowing the meaning of the concept the speaker wants to use, she can inform to the speaker an appropriate translation to her terminology.

4 Formal Properties of Translations Between Ontologies

The main problem for establishing a translation of a message is to decide the semantic relations between the concepts of the two ontologies. In this section, we introduce the formal properties required of the translation functions used in the mechanism described later. We will adopt the requirements of [5] of maximal preservation of extensional meaning as the guideline to decide translations. In this section, we aim to provide a function *TransCand* to compute candidates for the translation of a concept literal C, as used in rule *AddConcept* explained later.

As discussed earlier, we believe the limitations imposed in [5] are very restrictive, in the sense that they may not be applied to a wide range of MAS. Giving up these assumptions, however, implies we can no longer have certainty on which translation is the correct. We can identify only those which cannot be good translations since the computation may only be performed over an incomplete set of information about the world.

We assume a fixed set Δ of individuals for all ontologies, i.e. there are no private names for individuals. This assumption is equivalent to require individuals to be referenced by URIs. In the following, we define a series of properties that have to be satisfied by our translation between ontologies. These properties are based on those given in [5] posed in the context of incomplete information about the domain.

Definition 5 (Sound Translation) *Let $\mathcal{O}_1$ and $\mathcal{O}_2$ be two ontologies. We say $T_{ij} : \mathcal{B}_{\Phi_i} \to \mathcal{B}_{\Phi_j}$ is a sound translation from $\mathcal{O}_i$ to $\mathcal{O}_j$ iff*

$$\forall C \in \mathcal{B}_{\Phi_i}, \forall a \in \Delta \ (\mathcal{O}_i \models C(a) \Rightarrow \mathcal{O}_j \not\models \neg T_{ij}(C)(a)).$$

Additionally, we say of any $C' \in \mathcal{B}_{\Phi_j}$ to be a sound translation for $C \in \mathcal{B}_{\Phi_i}$ if there is a sound translation T_{ij} from $\mathcal{O}_i$ to $\mathcal{O}_j$ such that $T_{ij}(C) = C'$.

The soundness condition means that the original meaning of the message is coherent with the translated message, i.e. the translation of an atomic concept encompass all the positive cases of the original concept. Since it cannot be guaranteed that there is no atomic concept in the target terminology with the same extension as the original concept, we consider a translation is sound if it is a superconcept of the original one, i.e. if it encompass all the positive information the original concept does.

Notice that we do not require, as [5], that the ontologies have complete information on all individuals. Consequently, by Definition 5 (and 6 below), the computation of appropriate translations relies mainly on the shared individuals, i.e. the individuals that appear in both ontologies.

The other property required is that of lossless communication. To define that more elegantly, we will use the notion of extension of a DL formula φ in an ontology $\mathcal{O}$, meaning all the individuals that are inferred to be an instance of φ in $\mathcal{O}$, symbolically $ext(\mathcal{O}, \varphi) = \{a \in \Delta \mid \mathcal{O} \models \varphi(a)\}$.

Definition 6 (Lossless Translation) *Let $\mathcal{O}_i$ and $\mathcal{O}_j$ be two ontologies. We say $T_{ij} : \mathcal{B}_{\Phi_i} \to \mathcal{B}_{\Phi_j}$ is a lossless translation from $\mathcal{O}_i$ to $\mathcal{O}_j$ iff f is a sound translation and* *for any atomic concept* $C \in \mathscr{B}_{\Phi_i}$ *there is no sound translation* $T'_{ij} : \mathscr{B}_{\Phi_i} \to \mathscr{B}_{\Phi_j}$ *such that*

$$ext(\mathscr{O}_i, C) \cap ext(\mathscr{O}_j, T_{ij}(C)) \subseteq ext(\mathscr{O}_i, C) \cap ext(\mathscr{O}_j, T'_{ij}(C))$$

and

$$ext(\mathscr{O}_i, \neg C) \cap ext(\mathscr{O}_j, T'_{ij}(C)) \subset ext(\mathscr{O}_i, \neg C) \cap ext(\mathscr{O}_j, T_{ij}(C)).$$

Similarly, we say of any $C' \in \mathscr{B}_{\Phi_j}$ *to be a lossless translation for* $C \in \mathscr{B}_{\Phi_i}$ *if there is a lossless translation* T_{ij} *from* $\mathscr{O}_i$ *to* $\mathscr{O}_j$ *such that* $T_{ij}(C) = C'$.

This property means that the translation of an atomic concept is the most specific translation possible in the target ontology. In other words, while the translation of an atomic concept may differ in extension from the original one since a complete preservation may not be possible, the difference in the extension is minimal (w.r.t. set inclusion).

The following properties state the meaning preservation properties of lossless translations. First we show that, up to extensional equivalence, the translated concept preserves the meaning of the original concept.

Proposition 1 *Let* $\mathscr{O}_i$ *be an ontology and* T_{ii} *a lossless translation from* $\mathscr{O}_i$ *to* $\mathscr{O}_i$. *Then the translation of any concept C is (extensionally) equivalent to C, i.e.*

$$\forall C \in \mathscr{B}_{\Phi_i} \; (ext(\mathscr{O}_i, C) = ext(\mathscr{O}_i, T_{ii}(C))))$$

It is easy to see that this proposition holds from the definition of lossless translation. Notice that C is maximal element concerning the properties of Definition 6. Since any other translation must include the extension of C, by maximality of C, their extensions must be equal.

The following proposition states that the information $T_{ij}(A) = B$, where T_{ij} is a lossless translation, can be identified as a (defeasible) subsumption axiom $A \sqsubseteq B$ in the union of the ontologies.

Proposition 2 *Let* $\mathscr{O}_i$, $\mathscr{O}_j$ *be ontologies,* T_{ij} *a lossless translation from* $\mathscr{O}_i$ *to* $\mathscr{O}_j$ *and* $\mathscr{O}_T = \{C_1 \sqsubseteq C_2 \mid C_2 \neq \bot \wedge T_{ij}(C_1) = C_2\}$. *Then* $\mathscr{O}_i \cup \mathscr{O}_j \cup \mathscr{O}_T$ *is consistent.*

Notice that, by definition of soundness, $O_1 \vDash C(a)$, then $O_2 \nvDash \neg T_{ij}(C)(a)$. Thus, there is no individual a s.t. $C(a)$ and $\neg T_{ij}(C)(a)$ are derivable from the ontology. For this reason, the proposition above holds.

From the definitions above, we can easily construct functions to compute the set of possible sound and lossless translations. We provide algorithms for the computation of those function (Figures 2.a and 2.b), given that this is a central step in the strategy for ontology negotiation in our mechanism. Notice that other techniques in instance-based matching [2] can be easily integrated to order or select translations, as some similarity measure between concepts. Particularly, since a successful translation is dependent on shared individuals, other methods may provide alignments between individuals.

The algorithm *CompSound* in Figure 2.a for computing admissible sound translations for an atomic concept C based on its positive and negative instantiations works by testing for each atomic concept in the hearer's ontology if the soundness condition is

```
Algorithm CompSound($\mathscr{O}$, $Pos$)
Input:
    Knowledge base $\mathscr{O}$
    Set $Pos$ of positive instantiations of a concept
Output:
    set of sound translations
  $Sound$ := $\{\}$
  for each $C$ concept name in $\mathscr{O}$
      if $Pos \cap ext(\mathscr{O}, \neg C) = \{\}$ then
          $Sound$ := $Sound \cup \{C\}$
  return $Sound$
```

(a) Algorithm for computing sound translations of a concept

```
Algorithm TransCand($\mathscr{O}$, $Pos$, $Neg$)
Input:
    Knowledge base $\mathscr{O}$
    Set of positive and negative instantiations $Pos, Neg$
    of a concept
Output: set $S$ of lossless translations
  $S$ := $CompSound(\mathscr{O}, Pos)$
  repeat
    $S'$ := $S$
    for each $C \in S'$
        for each $C' \in S' \setminus \{C\}$
            if $ext(\mathscr{O}, C) \cap Pos \subseteq ext(\mathscr{O}, C') \cap Pos$ and
                $ext(\mathscr{O}, \neg C') \cap Neg \subset ext(\mathscr{O}, \neg C) \cap Neg$
            then $S$ := $S \setminus \{C\}$
  until $S = S'$
  return $S$
```

(b) Algorithm for computing lossless translations of a concept

Fig. 2. Algorithms for computing translations

satisfied by this concept. If soundness is violated, the concept is rejected as a possible translation.

The algorithm *TransCand* in Figure 2.b computes candidates for lossless translations of a concept C, based on its positive and negative instantiations. It works by, first selecting all atomic concepts in the knowledge base $\mathscr{O}$ that are sound translations for C and testing for each one if it satisfies the lossless condition of Definition 6. If this condition is violated, the concept is rejected as a possible translation. By the following result, we have that the algorithm presented in Figure 2.b is correct. Notice that since the set of concept and role names are finite, so it is the set of atomic concepts, and thus the algorithms presented always terminate.

Proposition 3 *Let $\mathscr{O}_i$ and $\mathscr{O}_j$ be two ontologies with $\mathscr{B}_{\Phi_i}, \mathscr{B}_{\Phi_j}$ their respective sets of atomic concepts, and $C \in \mathscr{B}_{\Phi_i}$ s.t. $ext(\mathscr{O}_i, \neg C) = Neg$ and $ext(\mathscr{O}_i \neg C) = Pos$. For all $C' \in TransCand(O_j, Pos, Neg)$, C' is a lossless translation of C in $\mathscr{O}_j$.*

It is not difficult to see that the algorithm is correct, since it test for every atomic concept of the ontology whether the requirements in Definitions 5 and 6 hold. Also, notice that the algorithm always terminates, since it is an iteration on a finite set of concepts.

5 Integrating Ontology Negotiation in Agent Communication

Once established the main notions used in this work, we begin the description of a communication mechanism allowing terminological negotiation. To specify such mechanism, we will use a simple model for an agent ag as a tuple consisted of is composed of an ontology $\mathscr{O}$, a collection translation functions T and a message base $M = \langle In, Out, Susp, Hist \rangle$ with the agent's messages Inbox, Outbox, Suspended Messages, and History of Messages, with all messages sent by the agent respectively. An agent is, thus, a triple $ag = \langle \mathscr{O}, M, T \rangle$.

When an agent $ag_i = \langle \mathscr{O}_i, M_i, T_i \rangle$, for example, wants to send a message to agent $ag_j = \langle \mathscr{O}_j, M_j, T_j \rangle$ she executes an action `.send`. Messages in the outbox M_{Out} have the form $\langle mid, id, ilf, cnt\rangle$, where *mid* is the message identifier, *id* is the hearer's identifier, *ilf* is the illocutionary force or type of the message and *cnt* its content. A message in the inbox M_{In} has the same format except that *id* is the identification of the agent that has sent the message.

A multiagent system is composed of agents $\langle ag_1, \dots ag_n\rangle$ asynchronously communicating with each other. In a multiagent system with n agents, the component T of each agent i has $n-1$ concept translation functions $T_{ij} : \mathscr{B}_{\Phi_i} \rightarrow \mathscr{B}_{\Phi_j}$, one for every other agent j.

The operational semantics of our communication mechanism is given by a set of rules that define a transition relation between configurations $\langle \mathscr{O}, M, T\rangle$. Intuitively the notation $\langle \mathscr{O}, M, T\rangle \longrightarrow \langle \mathscr{O}', M', T'\rangle$ means that, after one step in its execution, the components of agent $\langle \mathscr{O}, M, T\rangle$ may have been modified to $\langle \mathscr{O}', M', T'\rangle$

Since the main components of the configuration are tuples, we will use the subscript when referring to a specific component, e.g. M_{In} will be used to refer to the inbox *In* in the tuple M. We will also make use of *selection functions* that are defined by the agent programmer, e.g. the function S_M selects a message from the agent's message boxes, such as M_{In}, to be processed next.

We will assume that, unless negotiating the addition or explanation of a concept, the agents always translate their messages before sending them. This assumption can be easily implemented by taking the semantics of sending a message $\langle mid, id, ilf, cnt\rangle$ to automatically translate the contents of the message.

We begin the description of the performatives by explaining how the different ontologies affect the rules for assertional communication. Then, we introduce the main contributions of this work, i.e. the rules for terminological negotiation. We will explain the rules of the communication mechanism by instantiating them in the interaction between two agents (agent i and j).

Assertional Communication Assertional communication refers to the communication about ground facts commonly available in multiagent systems and data exchange mechanisms, such as proposed in [14] and [13]. Usual performatives available for the agents are ones as *Tell/Inform* for communicating to an agent some information and *Ask/Confirm* for querying an agent whether an information is valid or for which individuals it is true. In this work, we will limit our discussion to the *Tell* performative to illustrate the change in the semantics that results from the inclusion of ontologies and negotiation in the mechanism. The other performatives in the literature, such as those in [14] may be constructed similarly.

As in the ontology negotiation protocols described in [5] when the hearer detects a possible loss of information, she must proceed to request further explanation. In the simplest case when no loss of information occurs, the semantics of the communication is straight forward.

When an agent j receives an atomic formula $C_j(a)$ from agent i as a *Tell*, and there is no information loss, where C_j is a concept in agent j's terminology, then agent j must update her Abox with that information. To represent the detection or non-detection of an information loss, we will use the $InfLoss$ predicate of Definition 4. Also, we use an Update function to include a set new factual information in the Abox. This function is highly dependent on DL used to represent the ontological information, but methods for Abox updating have been proposed for a wide range of description logics [16–18].

$$\frac{\begin{array}{c} S_M(M_{In}) = \langle m_0, i, Tell, C_j(a) \rangle \\ \langle \mathcal{O}_j, T_{ji} \rangle \vDash \neg InfLoss(C_j(a)) \end{array}}{\langle \mathcal{O}_j, M_j, T_j \rangle \longrightarrow \langle \mathcal{O}'_j, M_j, T_j \rangle} \qquad (\textsc{Tell})$$

$$\begin{array}{l} \textit{where:} \\ \qquad \mathcal{O}'_j = Update(\mathcal{O}_j, C_j(a)) \end{array}$$

The more interesting case for us happens when an information loss has been detected. As in the examples discussed before, the agent must request further clarification of the message. In our mechanism, we include the performative *ReqSpec* to represent this case. To request further specification of an information in the message, the agent will send a *ReqSpec* message with the information for which the agent has detected a possible information loss.

When an agent j receives an atomic formula $C_j(a)$ from agent i as a *Tell* and there is a potential information loss, she will reply to agent i with a request for a further specification of the information.

$$\frac{\begin{array}{c} S_M(M_{In}) = \langle m_1, i, Tell, C_j(a) \rangle \\ \langle \mathcal{O}_j, T_{ji} \rangle \vDash InfLoss(C_j(a)) \end{array}}{\langle \mathcal{O}_j, M_j, T_j \rangle \longrightarrow \langle \mathcal{O}_j, M'_j, T_j \rangle} \qquad (\textsc{TellInfLoss})$$

$$\begin{array}{ll} \textit{where:} & \\ M'_{jOut} & = M_{jOut} \cup \{\langle m_1, i, ReqSpec, C_j(a)[Tell] \rangle\} \end{array}$$

The request for a specification will begin an interaction for terminological exchange. While this terminological exchange is being performed, however, the assertional exchange that initiated it - the *Tell* message above - will be suspended, for it to be restarted after a new concept is introduced to express correctly the information agent i wished to convey.

Notice that the request for a specification occurs in every communication act in which an information loss is detected. We are exploring the specific case of a *Tell* message, but it could be generated by any other assertional communication performative of [14], such as an *Ask* message. For agent i to be able to resend the original information after the terminological negotiation is over, agent j will annotate the illocutionary force, i.e. the type of the message, in the *ReqSpec* so agent i can reconstruct it after the negotiation is over.

Receiving a ReqSpec message The *ReqSpec* is aimed to request the expansion of the conversation vocabulary when a possible information loss is detected by the hearer. When agent i receives a *ReqSpec* message from agent j, with content $C_j(a)$, she must add a new concept to the vocabulary that will allow a lossless communication between the agents. This action is performed by sending *AddConcept* messages, explaining the (known) extensional meaning of the concept.

Notice that the information $C(a)$ that agent i initially wished to convey has been translated to $C_j(a)$ before the first message was sent. Because of that, to further explain it, agent i must reacquire the original information, stored the her message history.

Since a *ReqSpec* message initiates a negotiation process, the assertional message agent i wanted to send to agent j must be suspended, waiting the end of the negotiation. Agent i, thus, will reconstruct the original message she wished to convey and store it in her suspended message, for it to be processed after the terminological exchange is over.

$$\frac{\begin{array}{c} S_M(M_{In}) = \langle m_1, j, ReqSpec, C_j(a)[Tell]\rangle \\ \langle m_1, j, Tell, C(a)\rangle \in Hist \end{array}}{\langle \mathcal{O}_i, M_i, T_i\rangle \longrightarrow \langle \mathcal{O}_i, M'_i, T_i\rangle} \qquad (\text{REQSPEC})$$

$$\begin{array}{ll} \textit{where:} & \\ Hist' & = Hist \setminus \{\langle m_1, j, Tell, C(a)\rangle\} \\ S & = \{C(b) \mid \mathcal{O}_i \models C(b)\} \cup \{\neg C(b) \mid \mathcal{O}_i \models \neg C(b)\} \\ M'_{iSusp} & = M_{iSusp} \cup \{\, \langle m_1, j, Tell, C(a)\rangle\} \\ M'_{iOut} & = M_{iOut} \cup \{\langle m_1, j, AddConcept, S\rangle\} \end{array}$$

Receiving an AddConcept message The *AddConcept* is aimed to inform the hearer of the (extensional) meaning of a new concept C to be used in communication. An agent may send an *AddConcept* as a means to introduce a new concept she wants to use or as a response to a request for a terminological specification.

When agent j receives from agent i an *AddConcept* message with the set S containing the extension to the concept C, she must search for the concepts in her ontology that constitute a good translation for this new concept and inform agent i this information. To compute the "good" candidates for the translation of C in terms of the hearer's concepts, we will use a function *TransCand*, presented in Section 4. In the rule, we use an auxiliary function S_T that selects one among the candidates for translation. This selection function may be provided by the programer or based on some ordering on the concepts, not unlike the selection function in Definition 2.

$$\frac{S_M(M_{In}) = \langle m_1, i, AddConcept, S]\rangle}{\langle \mathcal{O}_j, M_j, T_j\rangle \longrightarrow \langle \mathcal{O}_j, M'_j, T_j\rangle} \qquad (\text{ADDCONCEPT})$$

$$\begin{array}{ll} \textit{where:} & \\ P & = \{a \mid C(a) \in S\} \\ N & = \{a \mid \neg C(a) \in S\} \\ B & = S_T(TransCand(\mathcal{O}_j, P, N)) \\ M'_{jOut} & = M_{jOut} \cup \{\langle m_1, i, Translate, C \sqsubseteq B\rangle\} \end{array}$$

Receiving a Translate message The *Translate* message is a response to a previous *AddConcept* message. It contains one information: a terminological axiom $A \sqsubseteq B$, where A is the concept she wants to use in communication and B is the translation computed by the sender of the *Translate* message to this concept. As a result, the receiver will update her translation function to include this new information.

$$\frac{S_M(M_{In}) = \langle m_1, j, Translate, A \sqsubseteq B]\rangle}{\langle \mathcal{O}_i, M_i, T_i\rangle \longrightarrow \langle \mathcal{O}_i, M'_i, T'_i\rangle} \qquad (\textsc{Translate})$$

$$\begin{array}{ll} \textit{where:} & \\ T'(C) & = \begin{cases} B & \text{, if } C = A \\ T(C) & \text{, otherwise} \end{cases} \\ M'_{iSusp} & = M_{iSusp} \setminus \{\langle m_1, j, Tell, C(a)\rangle\} \\ M'_{iOut} & = M_{iOut} \cup \{\langle m_1, j, Tell, T(C)(a)\rangle\} \end{array}$$

6 Conclusions

In this work, we presented an integration of ontology negotiation and an agent communication mechanism. Using the notion of translation between ontologies, and the algorithms provided to compute such translations, we guarantee the communication to be meaning-preserving, focusing on extensional meaning. It is important to notice that we focus on the integration of a negotiation protocol within a broader communication mechanism for agent communication within practical multiagent systems and not on a method for ontology mediation *per si*.

About the translation method, it is important to notice two things. Firstly, by giving up on the use of classifiers as in [5], the successful communication between agents relies on the existence of shared individuals between the agents, i.e. that there exist some individuals known by both agents. Secondly, it is also important to notice that the choice to translate concept names into concept names has expressibility consequences. Allowing the translation of a concept name to be a DL formula would provide a much more fine-grained way of expressing the relationship between the concepts of the agents' ontologies, as in [8]. The choice we made in our work was based on two reasons: firstly most communication mechanisms available rely on the exchange of ground literals, not complex formulas; secondly, allowing complex formulas as translations - even if restricted to conjunctions of atomic concepts - yields in an exponential complexity for computing the translation candidates. Also, we would like to point out that a method of translation from concept names to formulas is highly dependent on which DL is used to axiomatize the ontologies while our method is general.

Regarding complexity, our approach requires only a linear number of query answering requests for the ontology reasoner. If the underlying Description Logic is limited enough, the computation of translations is tractable. We don't consider the integration of ontology negotiation in the communication mechanism to introduce a considerable overhead to the system. The reason for this belief is that translations are cumulative throughout the execution and are only computed when needed. This leads us to conclude that our method is scalable to large and open-ended systems, without creating a great overhead.

In future work, we would like to explore more deeply the connection between translation functions and defeasible subsumption rules. We believe the semantics developed for defeasible description logics may provide a rich understanding of ontology negotiation as a reasoning problem.

References

1. J. de Bruijn, M. Ehrig, C. Feier, F. Martín-Recuerda, F. Scharffe, and M. Weiten, “Ontology mediation, merging and aligning,” *Semantic web technologies*, pp. 95–113, 2006.
2. J. Euzenat, P. Shvaiko *et al.*, *Ontology matching*. Springer, 2007, vol. 18.
3. F. FIPA, “Ontology service specification,” *FIPA XC00086D*, 2001.
4. S. C. Bailin and W. Truszkowski, “Ontology negotiation between intelligent information agents,” *The Knowledge Engineering Review*, vol. 17, no. 1, pp. 7–19, 2002.
5. J. Van Diggelen, R.-J. Beun, F. Dignum, R. M. Van Eijk, and J.-J. Meyer, “Ontology negotiation: goals, requirements and implementation,” *International Journal of Agent-Oriented Software Engineering*, vol. 1, no. 1, pp. 63–90, 2007.
6. V. Mascardi, D. Ancona, R. H. Bordini, and A. Ricci, “CooL-AgentSpeak: Enhancing AgentSpeak-DL Agents with Plan Exchange and Ontology Services,” in *Intelligent Agent Technology - IAT*, 2011, pp. 109–116.
7. A. B. Williams, “Learning to share meaning in a multi-agent system,” *Autonomous Agents and Multi-Agent Systems*, vol. 8, no. 2, pp. 165–193, 2004.
8. Y. Tzitzikas and C. Meghini, “Ostensive automatic schema mapping for taxonomy-based peer-to-peer systems,” in *Cooperative Information Agents VII*. Springer, 2003, pp. 78–92.
9. D. Calvanese, G. De Giacomo, M. Lenzerini, and R. Rosati, “Actions and programs over description logic ontologies,” in *Description Logics – DL*, 2007.
10. F. Baader, S. Brandt, and C. Lutz, “Pushing the el envelope,” in *IJCAI*, vol. 5, 2005, pp. 364–369.
11. W. A. Carnielli, M. E. Coniglio, and I. M. D’ottaviano, “New dimensions on translations between logics,” *Logica Universalis*, vol. 3, no. 1, pp. 1–18, 2009.
12. I. Horrocks, O. Kutz, and U. Sattler, “The even more irresistible sroiq.” *KR*, vol. 6, pp. 57–67, 2006.
13. T. Finin, R. Fritzson, D. McKay, and R. McEntire, “Kqml as an agent communication language,” in *International conference on Information and knowledge management*. ACM, 1994, pp. 456–463.
14. R. Vieira, A. Moreira, M. Wooldridge, and R. H. Bordini, “On the formal semantics of speech-act based communication in an agent-oriented programming language,” *Journal of Artificial Intelligence Research*, vol. 29, no. 1, pp. 221–267, Jun. 2007.
15. T. Klapiscak and R. H. Bordini, “JASDL: A practical programming approach combining agent and semantic web technologies,” in *Declarative Agent Languages and Technologies – DALT*, 2008, pp. 91–110.
16. H. Liu, C. Lutz, M. Milicic, and F. Wolter, “Updating description logic aboxes.” in *KR*, 2006, pp. 46–56.
17. C. Halaschek-Wiener, B. Parsia, E. Sirin, and A. Kalyanpur, “Description logic reasoning for dynamic aboxes,” in *Proceedings of the Int. Workshop on Description Logics, DL*, vol. 6, 2006.
18. D. Calvanese, E. Kharlamov, W. Nutt, and D. Zheleznyakov, “Evolution of dl- lite knowledge bases,” in *The Semantic Web–ISWC 2010*. Springer, 2010, pp. 112–128.