Proceedings of I-KNOW '09 and I-SEMANTICS '09
2-4 September 2009, Graz, Austria

Reaching Semantic Agreements through Interaction

Manuel Atencia **Marco Schorlemmer**
(IIIA-CSIC, Artificial Intelligence Research Institute, Spain
{manu,marco}@iiia.csic.es)

Abstract: We address the complex problem of semantic heterogeneity in multiagent communication by looking at semantics related to interaction. Our approach takes the state of the interaction in which agents are engaged as the basis on which the semantic alignment rests. In this paper we describe an implementation of this technique and provide experimental results on interactions of varying complexity.

Key Words: interaction model, alignment protocol, alignment mechanism
Category: I.2.11, I.2.12

1 Introduction

We tackle the problem of semantic heterogeneity as it arises when combining separately engineered software entities in open and distributed environments. In particular, we focus on how to reach mutual understanding of the terminology that occurs in communicated messages during a multiagent interaction. Semantic heterogeneity is most commonly addressed either by having recourse to shared ontologies, or else by resolving terminological mismatches by ontology mapping [Kalfoglou and Schorlemmer 2003, Euzenat and Shvaiko 2007]. Ontologies may indeed be very useful for stable domains and closed communities, but the cost of guaranteeing global semantics increases quickly as the number of participants grows. Ontology mapping allows for more dynamism and openness, but current techniques compute semantic similarity in an interaction-independent fashion, for instance, by exploring the taxonomic structure of ontologies or by resorting to external sources such as WordNet, where semantic relations like synonymy, among others, were determined prior to interaction and independently from it. Hence, in general, these techniques do not address the fact that the meaning of a term is also relative to its use in the context of an interaction.

In this paper we aim at proving that this more pragmatic context may guide interacting agents in reaching a mutual understanding of their respective local terminologies. For this we make an empirical evaluation of an implementation of the Interaction-Situated Semantic Alignment (I-SSA) technique (Section 2), originally formalised in [Atencia and Schorlemmer 2008]. Our implementation of I-SSA lets two agents interact through communicative acts according to two separate interaction models locally managed by each agent. All terminological mismatches during communication are handled at a meta-level in the context of an alignment protocol. As interaction-modelling formalism we have initially chosen finite-state automata (FSA), because they are the basis of more complex interaction-modelling formalisms such as Petri nets or electronic institutions [Arcos et al. 2005]. We set out to answer two Research Questions:

1. Is there a gain in communication accuracy —measured in the number of successful interactions, i.e., interactions reaching a final state— by repeated semantic alignment through a meta-level alignment interaction?
2. If so, how many repeated interactions between two agents are needed in order to get sufficiently good alignments —measured in the probability of a successful interaction?

The experimentation results (Section 3) give a positive answer to the first question and relate the number of interactions to the probability of a successful interaction on the basis of a collection of interaction models.

2 Interaction-Situated Semantic Alignment

We model a multiagent system as a set MAS of agents. Each agent in MAS has a unique identifier and may take one (or more) roles in the context of an interaction. Let *Role* be the set of roles and Id the set of agent identifiers. We write $(id : r)$, with $r \in Role$ and $id \in Id$, for the agent in MAS with identifier id playing role r.

Each agent is able to communicate by sending messages from a set M, which is local to the agent. We assume that a set $\mathfrak{I}_\mathrm{P}$ of *illocutionary particles* (such as "inform", "ask", "advertise", etc.) is shared by all agents (see, for example, KQML [Labrou and Finin 1997] or FIPA ACL [O'Brien and Nicol 1998]).

Definition 1. Given a non-empty set M of messages, the set of *illocutions generated by* M, denoted by $\mathfrak{I}(M)$, is the set of all tuples $\langle \iota, (id : r), (id' : r'), m\rangle$ with $\iota \in \mathfrak{I}_P$, $m \in M$, and $(id : r), (id' : r')$ agents such that $id \neq id'$. If $i = \langle \iota, (id : r), (id' : r'), m\rangle$ is an illocution then $(id : r)$ is the *sender* of i and $(id' : r')$ is the *receiver* of i. In addition, $\langle \iota, (id : r), (id' : r')\rangle$ and m are called the *head* and *content* of i, respectively.

In this work, we treat messages as propositions, i.e., as grounded atomic sentences, leaving the generalisation to first-order sentences for future work.

2.1 Interaction Models

We model an interaction model as a deterministic finite-state automaton whose transitions are labelled either with illocutions, or with special transitions such as, for instance, timeouts or null transitions (also λ-transitions):

Definition 2. An *interaction model* is a tuple $\mathrm{IM} = \langle Q, q^0, F, M, C, \delta \rangle$ where:

- Q is a finite set of *states*,
- q^0 is a distinguished element of Q named the *initial state*,
- F is a non-empty subset of Q which elements are called *final states*,
- M is a finite non-empty set of *messages*,
- C is a finite set of *special transitions*, and
- δ is a partial function from $Q \times (\mathfrak{I}(M) \cup C)$ to Q called the *transition function*.

Every interaction model is related with an automaton in a natural way. The notion of history associated to an interaction model presented bellow is very similar to a string accepted for an automaton. The clear difference is that the former one takes into account the states explicitly.

Definition 3. Let IM be an interaction model, where $\mathrm{IM} = \langle Q, q^0, F, M, C, \delta \rangle$. An IM-*history* or *history* associated with IM is a finite sequence:

$$h = q^0, \sigma^1, q^1, \ldots, q^{k-1}, \sigma^k, \ldots, q^{n-1}, \sigma^n, q^n$$

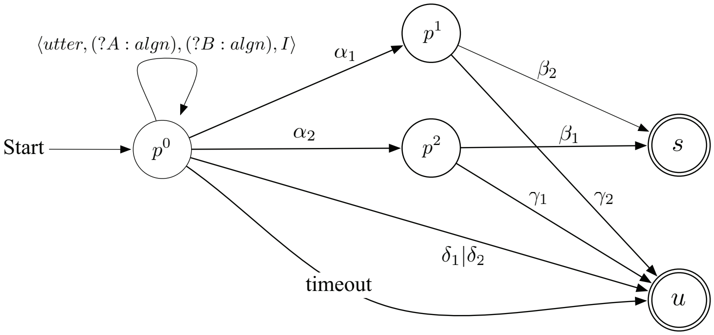
where $q^n \in F$ and for each k: $q^k \in Q$, $\sigma^k \in \mathfrak{I}(M) \cup C$ and $\delta(q^{k-1}, \sigma^k) = q^k$.

2.2 Alignment as Interaction

We study a scenario where two agents want to take part in an interaction, but with the thorny problem that the agents will follow different interaction models. So we have two agents A_1 and A_2 associated with interaction models IM_1 and IM_2, respectively, and we assume that these interaction models are distinct but they are about the same kind of interaction (e.g., a sealed-bid auction, a travel reservation or a bargaining process).

With agents knowing that they may follow different interaction models and that semantic mismatches are likely to occur, communication requires to be processed in another level. For this reason, we define a meta-level *alignment protocol* (AP) (see Figure 1) that links interaction models: any communication act according to the object level interaction models becomes ineffective and has an effective counterpart according to the meta-level AP.

There are two final states by name of letters s and u. If the state s is reached, then the interaction is considered *successful*, otherwise it is considered *unsuccessful*. In this sense, we distinguish for the moment only two sorts of interactions. Regarding transitions, all of them are listed below the figure except one that has a special status. Notice that agents can adopt only one role, namely, the



$$\alpha_i = \langle inform, (id_i : algn), (id_j : algn), \texttt{final\_state}\rangle$$
$$\beta_i = \langle confirm, (id_i : algn), (id_j : algn), \texttt{final\_state}\rangle$$
$$\gamma_i = \langle deny, (id_i : algn), (id_j : algn), \texttt{final\_state}\rangle$$
$$\delta_i = \langle inform, (id_i : algn), (id_j : algn), \texttt{failure}\rangle$$

Figure 1: The alignment protocol

'aligner' role, or *algn* in short. There are two kind of messages: `failure` and `final_state`. Moreover, the former one can be tagged with the illocutionary particle *inform*, and the later one with *inform*, *confirm* and *deny*.

Each agent follows both AP and its own interaction model. When agents agree to initiate an interaction, both of them are in state p^0 wrt AP. In addition, agent A_i is in state q_i^0 wrt IM_i $(i = 1, 2)$. Imagine agent A_i is in state q_i, where q_i is an arbitrary element of Q_i. There can be several possibilities.

1. A_i decides to utter $\mu = \langle \iota, (id_i : r), (id_j : r'), m)\rangle$ in accord with IM_i, where $\mu \in \delta_i(q_i, \cdot)$.[1] The communication act must be carried out via AP so agent A_i sends illocution $\langle utter, (id_i : algn), (id_j : algn), \mu\rangle$ to A_j. Therefore, the state remains the same in the AP context, whereas q_i turns to $q_i' = \delta_i(q_i, \mu)$ in the IM_i context.

2. A_i prompts a state change by a special transition $c_i \in C_i$ in the IM_i context. Thus q_i turns to $q_i' = \delta_i(q_i, c_i)$. This action is not reflected in AP since it does not entail any communication act.

3. A_i receives $\langle utter, (id_j : algn), (id_i : algn), \mu\rangle$ where $\mu = \langle \iota, (id_j : r), (id_i : r'), m)\rangle$ with regard to AP. Recall that from A_i's viewpoint, m is a foreign message so it is considered semantically different from all local messages. Consequently m **is to be mapped** with one of those messages that A_i expects to receive at state q_i in the IM_i context. Furthermore, we can make

---

[1] $\delta_i(q_i, \cdot)$ is the function defined from $\Sigma_i = \Im(M_i) \cup C_i$ to Q_i in a natural way.

a selection and just consider those messages encased in illocutions which head is equal to that of μ. In this way, A_i is to choose an element of the following set:

$$\mathcal{R} = \{a \mid \langle \iota, (id_j : r), (id_i : r'), a\rangle \in dom(\delta_i(q_i, \cdot))\}$$

There can be two possibilites: $\mathcal{R}$ is empty or not.

3.1 As long as $\mathcal{R}$ is not empty, A_i can select an element a of $\mathcal{R}$ making use of the **alignment mechanism** (AM) explained further below. So q_i turns to $q_i' = \delta_i(q_i, \nu)$ where $\nu = \langle \iota, (id_j : r), (id_i : r'), a)\rangle$.

3.2 In case $\mathcal{R}$ is empty, then no mapping is possible. The interaction is considered unsuccessful. In order to state it, A_i sends a failure message to A_j by uttering $\delta_i = \langle inform, (id_i : algn), (id_j : algn), \texttt{failure}\rangle$. Thus p_0 turns to u in the AP context.

4. If q_i is a final state and A_i considers the interaction finished, it can send illocution $\alpha_i = \langle inform, (id_i : algn), (id_j : algn), \texttt{final\_state}\rangle$ to A_j. In this case, p_0 turns to p_i and A_i expects to receive illocutions β_j or γ_j $(j \neq i)$, either confirming or denying the interaction end, respectively. If it receives β_j, then p_i turns to s and the interaction is considered successful; if it receives γ_j, p_i turns to u and the interaction is considered unsuccessful.

5. Finally, we have to take into account the possibility of a deadlock. This is the case when, for example, successive mappings have led the agents to states where both of them only can receive. In order to avoid deadlocks, the special transition *timeout* is linked to the initial state p_0 in AP. When a specific period of time is exceeded, this transition leads agents to finish the interaction considered unsuccessful.

The alignment mechanism associates every foreign message with a categorical variable ranging over local messages, such that a variable assignment represents a mapping element. The mechanism further computes frequency distributions of all these variables on the basis of past successful interactions. Agents mapping choices are determined by virtue of these distributions.

Assume agent A_i tackles a situation like the one described above in case 3.1. Message m is associated with a variable X that takes values in M_i. The equality $X = a$ represents a *mapping element* (the fact that m is mapped to a), also written $[m/a]$. If there is no past experience, $[m/a]$ is chosen with probability

$$p = \frac{1}{n}$$

where n is the cardinality of $\mathcal{R}$.

Now, things are different as long as agents have interacted successfully in the past. In order to reason about past experiences, agents have to keep track of these ones. A *history* is a sequence of the form:

$$h = q_i^0, \sigma_i^1, q_i^1, \ldots, q_i^{k-1}, \sigma_i^k, \ldots, q_i^{n-1}, \sigma_i^n, q_i^n$$

computed recursively as follows:

- q_i^0 is the initial state of IM_i, and
- if A_i is in case 1, then $[\imath, q_i']$ is queued in h,
- if A_i is in case 2, then $[c_i, q_i']$ is queued in h,
- if A_i is in case 3.1, $[\langle \iota, (id_j : r), (id_i : r'), [m/a] \rangle \rangle, q_i']$ is then queued in h,
- q_i^n is a final state of IM_i.

Notice that unsuccessful interactions are not considered.

Agents resort to all past histories in order to calculate the frequency distributions. Remember foreign messages do not occur in isolation: each message is the content of a specific illocution which is received at a particular state. To capture this dependency two more variables are considered: Q and H. Q takes values in the set of states Q_i and H can be instantiated with heads of illocutions.

So coming back to a situation like the one described in 3.1, agent A_i wonders whether $X = a$, where a varies in M_i, given that m is the content of an illocution with head $H = \langle \iota, (id_j : r), (id_j : r') \rangle$ that has been received at state $Q = q_i$. Using the corresponding frequency distribution:

$$f_r[X = a \mid Q = q_i, H = \langle \iota, (id_i : r), (id_j : r') \rangle] = \frac{v}{w} \in \mathbb{Q}$$

and $[m/a]$ is chosen with probability

$$p = \frac{v}{w}$$

Note that this option prevents agents from discovering new mapping elements. Alternatively, we can "contaminate" this distribution probability with the uniform distribution over $[1, \ldots, N_0]$, where N_0 is the number of zeros of the former frequency distribution. In this case, $[m/a]$ is chosen with probability

$$p = q\frac{v}{w} + (1-q)\frac{1}{N_0}$$

where q is a number close to 1.

3 Experimentation

3.1 Design

In this section we will explain our experiment design. The alignment protocol and mechanism are implemented in Sicstus Prolog and all random operations were executed with the Sicstus Prolog random library.

In our simulations only two agents are considered. This assumption is by no means very restrictive, since it is always possible to split an interaction among several agents into several interactions between two agents.

To overcome the lack of sufficiently complex examples with which to run our implementation and experiments, we have used the FSA Utilities toolbox [van Noord 1996] as follows. First, an abstract alphabet made up of arbitrary illocutions and special transitions is generated. Second, a regular expression is built upon this alphabet and prefixed numbers of Kleene star, concatenation and alternation operators. Finally, the regular expression is compiled into an automaton using the FSA library. Table 1 shows all variables considered in this process and the range of values they may take.

Name	**Variable**	**Range**
Number of illocutions	N_{ill}	$\mathbb{N}^*$
Number of illocutionary particles	N_{ip}	$\mathbb{N}^*$
Number of roles	N_{role}	$\mathbb{N}^*$
Number of messages	N_{msg}	$\mathbb{N}^*$
Number of special transitions	N_{spt}	$\mathbb{N}$
Number of Kleene star operators	N_{star}	$\mathbb{N}$
Number of concatenation operators	N_{con}	$\mathbb{N}$
Number of alternation operators	N_{alt}	$\mathbb{N}$

Table 1: Simulation variables

In practice, the ranges of these variables are bounded. One expects N_{ip} not to be much greater than 30 (KQML performatives, for instance, do not exceed this value). A reasonable upper bound for N_{role} is 20, and our recent experience within the OpenKnowledge project has confirmed this (see, for example, [Marchese et al. 2008], where an eResponse interaction model with no more than 10 roles is defined).[2] Likewise the number of special transtions is no likely to be greater than 5. Though ontologies vary in size from a few hundred terms to

---

[2] http://www.openk.org.

hundreds to tens of thousands of terms, these amounts reduce when limited to appear in specific interactions. For this reason, interaction models with more than 100 messages are not considered. Now, operators measure the complexity of the interaction model. Again, experience within the Openknowledge project has shown that interaction model complexity do not go over the complexity entailed by a few hundreds of operators. Finally, the number of illocutions N_{ill} is bounded by the following inequalities which must hold to ensure that all symbols considered so far appear in the resulting interaction model:

$$N_{ill} \geq \max\left\{N_{ip}, N_{msg}, \left\lceil \frac{N_{role}}{2} \right\rceil + 1\right\} \tag{1}$$

$$N_{ill} + N_{spt} \leq N_{con} + N_{alt} + 1 \tag{2}$$

We generated five interaction models corresponding with the variable groundings of Table 2 (with the same variable order as in Table 1).

Interaction model	**Variable instantiations**
imodel1	$15, 1, 1, 5, 0, 2, 10, 15$
imodel2	$20, 1, 2, 10, 0, 5, 15, 10$
imodel3	$30, 2, 3, 15, 2, 10, 20, 25$
imodel4	$50, 1, 1, 40, 0, 15, 30, 25$
imodel5	$100, 4, 5, 80, 2, 20, 50, 80$

Table 2: Interaction models

3.2 Execution and Evaluation

Remember that in our model agents consider all foreign messages semantically different, even when they match sintactically with any local ones. This fact justifies our decision to let agents follow the same interaction model, since agents will deal the situation as if they conform to disparate models.

In total three experiments were performed. In the first one, we simulated two agents interacting through the alignment protocol and taking advantage of the alignment mechanism. In the second one, agents only made use of the alignment protocol and no update alignment was carried out ever. Now, some series of interactions were simulated. Specifically, we ran both implementations with all interaction models in series of $N = 2^n$ interactions, where $n = 1, 2, \ldots, 12$ (thus we let agents interact at most 4096 times). Each batch of interactions was performed 50 times recording the average of failures $F(N)$ (or F). In order to compare both experiments we computed the ratio of failures to interactions, that is, $R = \frac{F}{N}$. Figure 2 exposes the results corresponding to imodel4. It is straightforward to check that when using the alignment mechanism the number of failures decreases considerably, while the alignment protocol alone yields a higher and almost constant number of failures. Similar results were obtained with the rest of interaction models (Figure 3). This answers positively Research Question 1 stated in the Introduction.

In the third experiment, we first let agents interact as in the first experiment so as to compute an alignment, again in series of $N = 2^n$ interactions, where $n = 1, 2, \ldots, 12$. This alignment was then used by the agents to interact 50 times without using the alignment mechanism. We recorded this time the ratio of successes to interactions, that is, $R = \frac{S}{50}$. Figure 4 shows the results with the five interaction models. In all cases R approaches 1. Actually, no more than 256 interactions are needed to achieve an alignment that ensures a probability close to 0.8 to interact successfully. This answers Research Question 2.

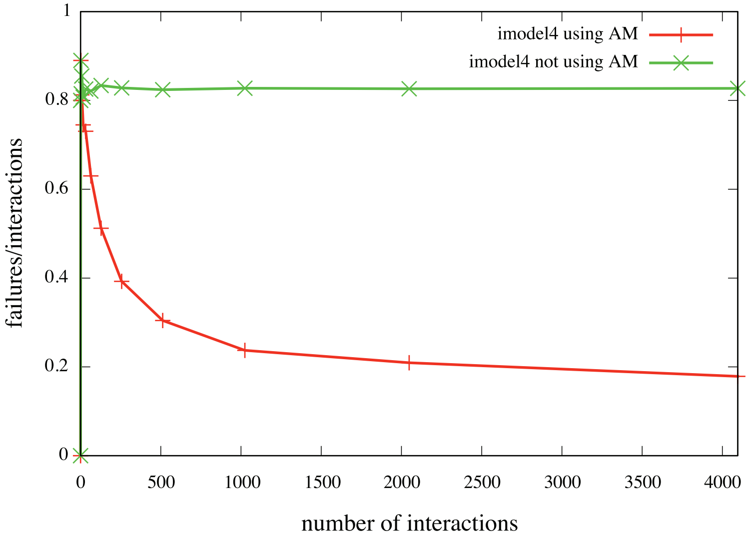
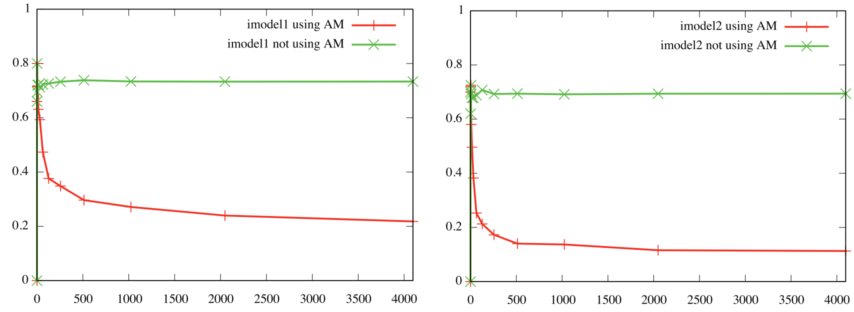


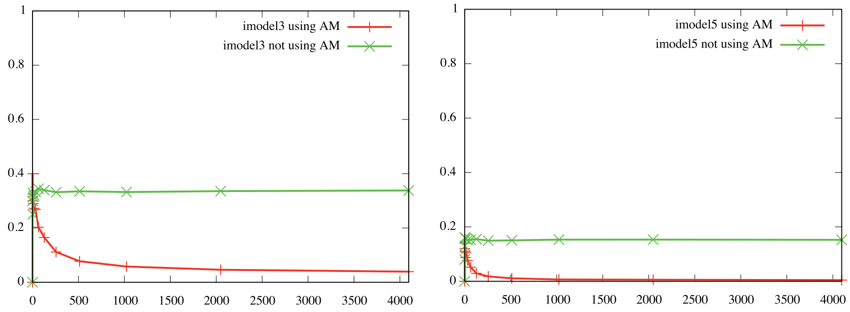
Figure 2: Experiments 1 and 2 with imodel4

4 Conclusions and Further Work

We have shown that, by guiding the interaction of agents that employ different terminologies by means of a meta-level alignment protocol, interacting agents are capable of significantly increasing their communication accuracy by repeated interactions. This meta-level alignment protocol takes the state into account when establishing semantic relationships between mismatching terminology.



(a) Experiments 1 and 2 with imodel1 (b) Experiments 1 and 2 with imodel2



(c) Experiments 1 and 2 with imodel3 (d) Experiments 1 and 2 with imodel5

Figure 3: Experimentation results

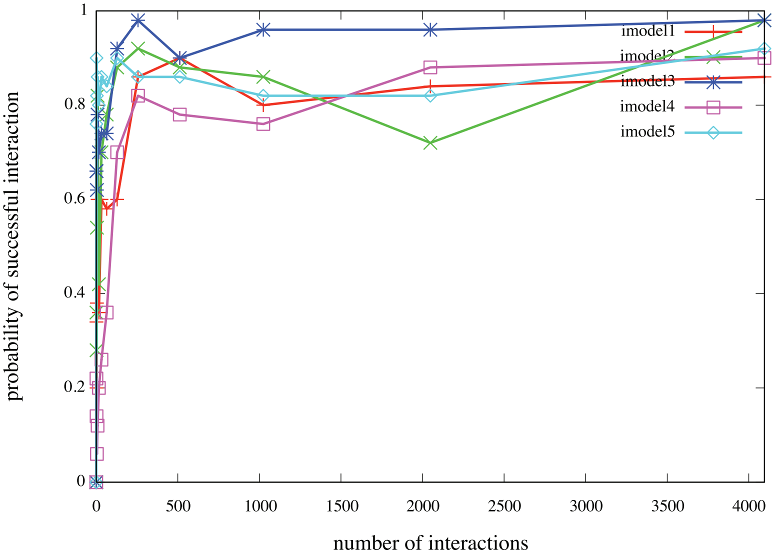


Figure 4: Experiment 3 with all interaction models

The alignment accuracy that agents are capable to achieve by resorting only to the interaction context as captured in the alignment protocol is relative to the expressiveness of the interaction-modelling language. For the case of simple FSA-based interaction models as those considered in this paper, semantic alignment is bounded by the mathematical product of interaction models as defined in [Atencia and Schorlemmer 2008]. In order to get more accurate interaction-situated semantic alignments we plan to extend our initial approach to more expressive interacion-modelling formalisms and richer communication languages.

Concerning the experimentation, we plan to test the significant relationship among the independent variables $(N_{ill}, N_{ip}, N_{role}, N_{msg}, N_{spt}, N_{star}, N_{con}, N_{alt})$ and the dependent variables (both ratios above). This will give us information about the kind of interaction models specially suitable for our approach.

5 Related Work

Other approaches share with ours the insight that semantics is often interaction-specific. In [Besana and Robertson 2007] the authors opt to attach probabilities to meanings of terms that are determined by earlier, similar interactions, and these probabilities are used to predict the set of possible meanings of a message. Meaning is also defined relative to a particular interaction, but the authors aim at reducing the search space of possible a priori mappings (in a classical sense), namely by assessing those ones with highest probability in the context of an interaction.

In [Rovatsos 2007] a dynamic semantics for agent communication languages (ACLs) is proposed. With the same spirit, Rovatsos bases his notion of dynamic semantics on the idea of defining alternatives for the meaning of individual speech acts in an ACL semantics specification, and transition rules between semantic states (collections of variants for different speech acts) that describe the current meaning of the ACL. One of our initial premises leads to an ACL to be shared by all agents. We believe that to agree on a pre-defined ACL is not a big assumption that can significantly help to solve the semantic heterogeneity brought by the existence of different agent content languages.

In tune with the previous work, Bravo and Velázquez present an approach for discovering pragmatic similarity relations among agent interaction protocols [Bravo and Velázquez 2008]. Besides the objection already explained above, the authors do not take into account state histories when measuring their notion of pragmatic similarity, but separate state transitions. This certainly leaves out relations among messages that may be crucial in certain scenarios.

Acknowledgments

This work is supported under Spanish projects IEA (TIN2006-15662-C02-01), MULOG 2 (TIN2007-68005-C04-01) and Agreement Technologies (CSD2007-0022), sponsored by the Spain Ministry of Science and Innovation; and under the Generalitat de Catalunya (2005-SGR-00093).

References

[Arcos et al. 2005] Arcos J.L., Esteva M., Noriega P., Rodríguez-Aguilar J.A., Sierra C.: "Engineering open evironments with electronic institutions"; Engineering Applications of Artificial Intelligence, 18, 2 (2005), 191-204.

[Atencia and Schorlemmer 2008] Atencia M., Schorlemmer M.: "I-SSA: Interaction-Situated Semantic Alignment"; 16th International Conference on Cooperative Information Systems (CoopIS'08), Monterrey, Mexico, November 9-14, 2008, Proceedings, Lecture Notes in Computer Science 5331, Springer Berlin, 445-455.

[Besana and Robertson 2007] Besana P., Robertson D., "How service choreography statistics reduce the ontology mapping problem"; 6th International Semantic Web Conference (ISWC'07), Busan, Korea, November 11-15, 2007, Proceedings, Lecture Notes in Computer Science 4825, Springer Berlin, 44-57.

[Bravo and Velázquez 2008] Bravo M., Velázquez J.: "Discovering pragmatic similarity relations between agent interaction protocols"; 4th International Workshop on Agents and Web Service Merging in Distributed Environments (AWeSoMe'08), Monterrey, Mexico, November 9-14, 2008, Proceedings, Lecture Notes in Computer Science 5333, Springer Berlin, 128-137.

[O'Brien and Nicol 1998] O'Brien P.D., Nicol R.C.: "FIPA – Towards a standard for software agents"; BT Technology Journal, 16, 3 (1998), 51-59.

[Euzenat and Shvaiko 2007] Euzenat J., Shvaiko P.: "Ontology matching"; Springer (2007).

[Kalfoglou and Schorlemmer 2003] Kalfoglou Y., Schorlemmer M.: "Ontology mapping: the state of the art"; The Knowledge Engineering Review, 18, 1 (2003), 1-31.

[Labrou and Finin 1997] Labrou Y., Finin T.: "Semantics and conversations for an agent communication language"; 15th International Joint Conference on Artificial Intelligence (IJCAI'97), Nagoya, Japan, August 23-29, 1997, Proceedings, 584-591.

[Marchese et al. 2008] Marchese M., Vaccari L., Trecarichi G., Osman N., McNeill F. : "Interaction models to support peer coordination in crisis management"; 5th International Conference on Information Systems for Crisis Response and Management (ISCRAM'08), Washington DC, USA, May 4-7, 2008, Proceedings, 230-242.

[van Noord 1996] van Noord, G.: "FSA utilities: a toolbox to manipulate finite-state automata"; 1st International Workshop on Implementing Automata (WIA'96), London, Ontario, Canada, August 29-31, 1996, Proceedings, Lecture Notes in Computer Science 1260/1997, Springer Berlin, 87-108.

[Rovatsos 2007] Rovatsos M.: "Dynamic semantics for agent communication languages"; 6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'07), Honolulu, Hawaii, USA, May 14-18, 2007, Proceedings, Electronic Edition, ACM DL.