

# Interaction is Meaning: A New Model for Communication in Open Systems

Michael Rovatsos, Matthias Nickles, Gerhard Weiss

{rovatsos, nickles, weissg}@cs.tum.edu  
Department of Informatics  
Technical University of Munich  
85748 Garching bei München  
Germany

## ABSTRACT

We propose a new model for agent communication in open systems that is based on the principle that the meaning of communicative acts lies in their experienced consequences. A formal framework for analysing such evolving semantics is defined. An extensive analysis of example interaction processes shows that our framework allows for an assessment of several properties of the communicative conventions governing a multiagent system. Among other advantages, our framework is capable of providing a very straightforward definition of communicative conflict. Also, it allows agents to reason about the *effects* of their communicative behaviour on the structure of communicative expectations as a whole when making decisions.

## Categories and Subject Descriptors

I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence—*Multiagent Systems, Languages and structures*

## General Terms

Theory, Measurement

## Keywords

Agent communication languages, evolutionary semantics.

## 1. INTRODUCTION

Defining the semantics of agent communication languages (ACLs) is one of the most important aspects of ACL research. In terms of *speech act theory* [1], which is the most widely accepted theory used for designing ACLs, this involves explaining the link between *illocution* and *perlocution*, i.e. to describe the *effects* of utterances (those desired by the sender and those brought about by the recipient of

the message) solely in terms of the speech acts used. Various proposed semantics suggest, however, that it is necessary to either resort to the mental states of agents [4, 3, 17] or to publicly visible commitments [5, 8, 13, 16] in order to capture the semantics of speech acts, i.e. to aspects of the system that are *external* to the language itself.

In the context of *open* multiagent systems (MAS) [6], characterised by dynamically changing populations of self-interested agents whose internal design is not (completely) accessible to others, it is not clear how specifications of mental attitudes or systems of commitments can be linked to the observed interactions. How can we make predictions about agents' future actions, if the semantics of their communication is defined in terms of mental states or commitments not related to the design of these agents?

In response to this problem, this paper suggests a view of communication semantics that is *empirical*, *consequentialist* and *constructivist* in nature. It is based on abandoning the distinction between illocution and perlocution in favour of defining the meaning of illocutions solely *in terms* of their perlocutions. This means that any utterance bears the meaning of its consequences (hence consequentialist), i.e. other utterances or observable physical actions, as derived from previous experience (hence empirical); also, this meaning is always regarded from the standpoint of a self-interested, locally reasoning agent with decision-theoretic (bounded) rationality (hence constructivist).

By grounding meaning in interaction and viewing semantics as an emergent and evolving phenomenon, this model of communication has the capacity to provide a basis for talking about agent communication that will prove useful as more and more MAS applications move from closed to open systems. Its practical use lies in the possibilities it offers for analysing agent interactions and for deriving desiderata for agent and protocol design. At a more theoretical level, our framework provides a very simple link between autonomy and control and introduces a new, powerful notion of conflict defined in purely communicative terms, which contrasts mentalistic or resource-level conflict definitions such as those suggested in [11]. As a central conclusion, "good" protocols are proven to be both autonomy-respecting and contingency-reducing interaction patterns, which is shown through an analysis of example protocols with our framework.

The remainder of this paper is structured as follows: section 2 presents the assumptions underlying our view of com-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

AAMAS'03, July 14–18, 2003, Melbourne, Australia.  
Copyright 2003 ACM 1-58113-683-8/03/0007 ...\$5.00.

munication, and in section 3 we lay out requirements for agents our model is suitable for. Sections 4 and 5 describe the model itself which is defined in terms of simple consequentialist semantics and entropy measures. An analysis of several interaction scenarios follows in section 6, and we round up with some conclusions in section 7.

## 2. BASICS

Our framework is based on a set of observations about communication, which follow quite naturally, once a certain view of agents in open systems is adopted.

We assume that agents are situated in an environment that is co-inhabited by other agents they can communicate with. They have preferences regarding different states of the world, and they strive to achieve those states that are most desirable to them. To to this end they *deliberate*, i.e. they take action to achieve their goals. Also, agents' actions have effects on each other's goal attainment – agents are *inter-dependent*.

In open, dynamic and unpredictable systems, it is useful to organise experience into a cause-and-effect model (which will depend much more on statistical correlation rather than on “real” causality) of the behaviour of their environment in order to take rational action. This is not only true of the physical environment, but also of other agents. Therefore, we regard the foremost function of communication to lie in providing such a causal model for the behaviour of other *agents* that an agent can use in a similar way as rules that it discovers regarding the physical environment.

However, two distinctions have to be made between physical actions executed to manipulate the environment and communicative interaction between agents: firstly, the *autonomy* of agents stands in contrast to the rules that govern physical environments – agents receive messages but are free to fulfil or disappoint the expectations [2] associated with them. Secondly, communication *postpones* (or replaces) “real” physical action<sup>1</sup>: it allows for the establishment of causal relationships between symbols and subsequent symbols or physical actions.

With this in mind, we claim the following:

- (1) Past experience with communication creates expectations for the future.
- (2) Agents employ information about expectations strategically.
- (3) Communicative expectations are generalised.
- (4a) Uncertainty regarding expectations should be reduced in the long run.
- (4b) Undesirable expectations need to be broken.

Statement (1) simply states that causal models can be built by agents from experience and used for predicting future behaviour. (2) is a consequence of (1) and the assumptions made about agent rationality (we can expect agents to use *any* information they have to achieve their goals).

The first interesting claim is (3) which points at a distinct property of communication. It means that in contrast to other causal models, the meaning of symbols used in communication is supposed to hold for *any* agent (cf. sociological models of communication [9, 10]). The fact that

<sup>1</sup>Of course, communication takes place in physical terms and hence *is* physical action. Usually, though, exchanging messages is not supposed to have a strong impact on goal achievement since it leaves the physical environment virtually unmodified.

illocutions (which usually mark certain paths of interactions) represented by performatives in speech act theory are parametrised with “sender” and “recipient” roles conforms with this intuition. Without this generalisation (which is ultimately based on a certain homogeneity assumption among agents [10]), utterances would degenerate to “signals” that spawn particular reactions in particular agents.

Claims (4a) and (4b) provide a basis for the design criteria applied when building agents that are to communicate effectively. Unfortunately, though, the goals they describe may be conflicting. (4a) states that the uncertainty in expectations should be reduced to a minimum. From a “control” point of view, ideally, an agent's peers would react to a message in a mechanised, fully predictable way so that any contingency about their behaviour can be ruled out. At the same time, the agent itself wants to be free to take any decision at any time to achieve its own goals. Since its plans might not conform with existing expectations, she may have to break them as stated by (4b). Or she might even desire some *other* peer to break an existing expectation, if, for example, the existing “habit” does not seem profitable anymore. We can summarise these considerations by viewing any utterance as a *request*, and asking *what* is requested by the utterance: the confirmation, modification or novel creation of an expectation.

These considerations lead to several desiderata for semantic models of communication:

- The meaning of a message can only be defined in terms of its *consequences*, i.e. the messages and actions that are likely to follow it. Two<sup>2</sup> levels of effects can be distinguished:
  1. The immediate reactions of other agents and oneself to the message.
  2. The “second-order” impact of the message on the expectation structures of any observer, i.e. the way the utterance alters the causal model of communicative behaviour.
- Any knowledge about the effects of messages must be derived from *empirical* observation. In particular, a semantics of protocols cannot be established without taking into account how the protocols are *used* in practice.
- Meaning can only be *constructed* through the eyes of an agent involved in the interaction, it strongly relies on relating the ongoing communication to the agent's own goals.

Following these principles, we have developed a framework to describe and analyse communication in open systems that will be introduced in the following sections.

## 3. ASSUMPTIONS ON AGENT DESIGN

### 3.1 The InFFrA social reasoning architecture

In order to present the view of communication that we propose in this paper, we first need to make certain assumptions regarding the type of agents it is appropriate for. For

<sup>2</sup>A third level of effects can be identified, that of *non-observable* effects. Since they cannot be correlated with utterances in the way discussed here, they are not considered.

this purpose, we shall briefly introduce the InFFrA social reasoning meta-architecture that has previously been described in full detail in [15]. We choose InFFrA to describe this view of communication, because it realises the principles laid out in the previous section, while making only fairly general assumptions about the kind of agents our models are suitable for.

InFFrA is based on the idea that agents organise the interaction situations they find themselves into so-called *interaction frames* [7], i.e. knowledge structures that represent certain categories of interactions. These frames contain information about

- the possible interaction *trajectories* (i.e. the courses the interaction may take in terms of sequences of actions/messages),
- *roles and relationships* between the parties involved in an interaction of this type,
- *contexts* within which the interaction may take place (states of affairs before, during, and after an interaction is carried out) and
- *beliefs*, i.e. epistemic states of the interacting parties.

While certain attributes of the above must be assumed to be shared knowledge among interactants (so-called *common attributes*) for the frame to be carried out properly, agents may also store their personal experience in a frame (in the form of *private attributes*), e.g. utilities associated with previous frame enactments, etc. What makes interaction frames distinct from interaction protocols and conversation policies is that (i) they provide comprehensive characterisations of an interaction situation (rather than mere restrictions on the range of admissible message sequences), and (ii) they always include information about experience with some interaction pattern, rather than just rules for interaction.

Apart from the interaction frame abstraction, InFFrA also offers a control flow model for social reasoning and social adaptation based on interaction frames, through which an InFFrA agent performs the following steps in each reasoning cycle:

1. *Matching*: Compare the current interaction situation with the currently activated frame.
2. *Assessment*: Assess the usability of the current frame.
3. *Framing decision*: If the current frame seems appropriate, continue with 6. Else, proceed with 4.
4. *Re-framing*: Search the frame repository for more suitable frames. If candidates are found, “mock-activate” one of them and go back to 1; else, proceed with 5.
5. *Adaptation*: Iteratively modify frames in the repository and continue with 4.
6. *Enactment*: Influence action decisions by applying the current frame. Return to 1.

This core reasoning flow, that is supposed to be performed by InFFrA agents in addition to their local goal-oriented reasoning processes (e.g., a BDI [14] planning and plan monitoring unit) is reasonably generic to cater for almost any kind of “socio-empirically adaptive” agent design.

Using the InFFrA meta-architecture, we can specify a “minimal” set of properties of agents that is in accordance with the principles laid out for our framework in section 2.

### 3.2 “Minimal” InFFrA agents

The simplest InFFrA-compliant agent design that can be conceived of is as follows: we consider agents that engage in two-party turn-taking interactions that occur in discrete time and whose delimiting messages/actions can always be determined unambiguously. This means that agents always interact only with one peer at a time, that these encounters consist of a message exchange in which agents always take turns, and that an agent can always identify the beginning and end of such an encounter (e.g. by applying some message timeout after which no further message from the other agent is expected anymore).

We also assume the existence of some special “deictic” message performative  $\mathbf{do}(A, X)$  that can be sent by agent  $A$  to indicate it is executing a physical (i.e. non-communicative) action  $X$  in the environment<sup>3</sup>.

Further, we assume that agents store these encounters as “frames”  $F = (C, w, h)$  in a (local) frame repository  $\mathcal{F}$  where  $C$  is a condition,  $w$  is a message sequence and  $h$  is a vector of message counters.

The message sequence of a frame is a simple kind of trajectory that can be seen as a word  $w \in \Sigma^*$  from some alphabet of message symbols  $\Sigma$  (which include the  $\mathbf{do}$ -symbols that refer to physical actions). Although agents may invent new symbols and the content language of messages (e.g. first-order logic) may allow for an infinite number of expressions,  $\Sigma$  is finite, since it always only contains symbols that have already occurred in previous interactions.

Since specific encounters are relevant/possible under particular circumstances only, we assume that the agent has some knowledge base  $KB$  the contents of which are, at any point in time, a subset of some logical language  $L$ , i.e.  $KB \in 2^L$ . Then, provided that the agent has a sound inference procedure for  $L$  at its disposal, it can use a condition (expressed by a logical formula  $C \in L$ ) to restrict the scope of a message sequence to only those situations in which  $C$  holds:

$$(C, w, h) \in \mathcal{F} \Leftrightarrow (KB \models C \Rightarrow w \text{ can occur})$$

In practice,  $C$  is used to encode any information about roles and relationships, contexts and beliefs associated with a frames as described in section 3.1.

As a last element of the frame format we use, agents employ “usage counters”  $h \in \mathbb{N}^{|w|}$  for each message in a frame trajectory. The counter values for all messages in some prefix trajectory sequence  $w \in \Sigma^*$  is incremented in all frames who share this prefix word whenever  $w$  occurs, i.e.

$$(w \text{ has occurred } n \text{ times} \wedge |w| = i) \Rightarrow \forall (C, wv, h) \in \mathcal{F}. \forall i \leq |w|. h_i = n$$

(for some  $v \in \Sigma^*$ ). This means that  $h$  is an integer-valued vector that records, for each frame, how often an encounter has occurred that started with the same prefix  $w$  (note that during encounters,  $h_i$  is incremented in *all* frames that have shared prefixes  $w$  if this is the message sequence just perceived until the  $i$ th message). Therefore,  $\text{count}(F)[i] \geq \text{count}(F)[i + 1]$  for any frame  $F$  and any  $i \leq |\text{traj}(F)|$  (we use functions  $\text{cond}(F)$ ,  $\text{traj}(F)$  and  $\text{count}(F)$  to obtain the

<sup>3</sup>More precisely,  $\mathbf{do}(A, X)$  is actually a shortcut for an observation action of the “recipient” of this message by which she can unambiguously verify whether  $A$  just executed  $X$  and which she interprets as part of the encounter; it need *not* be some distinguished symbol that has been agreed upon.

values of  $C$ ,  $w$  and  $h$  in a frame, respectively). To keep  $F$  concise, no trajectory occurs twice, i.e.

$$\forall F, G \in \mathcal{F}. \text{traj}(F) \neq \text{traj}(G)$$

and if a message sequence  $w = \text{traj}(F)$  that has been experienced before occurs (describing an *entire* encounter) under conditions  $C'$  that are not compatible with  $\text{cond}(F)$  under any circumstances (i.e.  $\text{cond}(C') \wedge C' \models \text{false}$ ),  $F$  is modified to obtain  $F' = (\text{cond}(F) \vee C', w, h)$ .

As a final element in this agent architecture, we assume the existence of a utility function

$$u : 2^L \times \Sigma^* \rightarrow \mathbb{R}$$

which will provide to the agent an assessment of the utility  $u(KB, w)$  of any message/action sequence  $w$  and any knowledge base content  $KB$ .

Minimal InFFrA agents who construct frame repositories in this way can use them to record their interaction experience: In any given situation, they can filter out those frames that are irrelevant under current belief and compute probabilities for other agents' actions and for the expectations others have of themselves given their own previous behaviour. They can assess the usability of certain frames by consulting their utility function, and they use the trajectories in  $\mathcal{F}$  both to determine the frames that are applicable and to pick their next actions.

## 4. EMPIRICAL SEMANTICS

As mentioned before, the semantic model we want to propose is purely *consequentialist* in that it defines the meaning of utterances in terms of their effects.

Let  $2 \cdot H \in \mathbb{N}$  be some upper bound on the possible length of encounters, and let  $\Delta(\Sigma^H)$  be the set of all discrete probability distributions over all words from  $\Sigma^*$  no longer than  $H$ .

We define the interpretation  $I_{\mathcal{F}}$  induced by some frame repository  $\mathcal{F}$  as a mapping from knowledge base states and current encounter sequence prefixes to the posterior probability distributions over all possible postfixes (conclusions) of the encounter. Formally,  $I_{\mathcal{F}} \in (2^L \times \Sigma^H \rightarrow \Delta(\Sigma^H))$  with

$$I_{\mathcal{F}}(KB, w) = \lambda w'. P(w'|w)$$

where

$$P(w'|w) = \alpha \cdot \sum_{\substack{F \in \mathcal{F}, \text{traj}(F) = ww', \\ KB \models \text{cond}(F)}} \text{count}(F)[|\text{traj}(F)|]$$

for any  $w, w' \in \Sigma^H$  and some normalisation constant  $\alpha$ .

This means that, considering those frames only whose conditions hold under  $KB$ , we compute the ratio of experienced conclusions  $w'$  to the already perceived prefix encounter  $w$  and the number of all potential conclusions to  $w$ .

The intuition behind this definition is that during an interaction encounter, if the encounter started with the initial sub-sequence  $w$ , the interpretation function  $I_{\mathcal{F}}$  will yield a probability distribution over all possible continuations  $w'$  that may occur in the remainder of the current interaction sequence.

Finally, given this probability distribution, we can also compute the expected ‘‘future utility’’ of any message se-

quence  $w$  by computing

$$\bar{u}(w) = \sum_{w' \in \Sigma^H} I_{\mathcal{F}}(KB, w)(w') \cdot u(KB', w')$$

if  $KB'$  is the state of the knowledge base after  $w'$  has occurred<sup>4</sup>.

The definitions in this section resemble the framework of Markov Decision Processes (MDPs) very much, and to capture the fact that probabilities of communication effects are *affected* by the decision-making agent herself, the MDP model would have to be modified appropriately. For the purposes of the present analysis, though, defining some simple measures on expectation structures will suffice.

## 5. ENTROPY MEASURES

With the above definitions at hand, we can now return to the principles of communication laid out in section 2. There, we claimed that an agent strives to reduce the uncertainty about others' communicative behaviour, and at the same time to increase its own autonomy.

We can express these objectives in terms of the *expectation entropy*  $EE$  and the *utility deviation*  $UD$  that can be computed as follows:

$$EE_{\mathcal{F}}(w, KB) = \sum_{w' \in \Sigma^H} -P(w'|w) \log_2 P(w'|w)$$

$$UD_{\mathcal{F}}(w, KB) = \sqrt{\sum_{w' \in \Sigma^H} (u(w', KB) - \bar{u}(w', KB))^2}$$

Total *entropy*  $\mathcal{E}_{\mathcal{F}}(w, KB)$  of message sequence  $w$  is defined as follows:

$$\mathcal{E}_{\mathcal{F}}(w, KB) = EE_{\mathcal{F}}(w, KB) \cdot UD_{\mathcal{F}}(w, KB)$$

How can these entropy measures be interpreted? The expectation entropy assesses the information-theoretic value of having performed/perceived a certain sequence  $w$  of messages. By computing the information value of all potential continuations,  $EE$  (again, we drop subscripts and arguments whenever they are obvious from the context) expresses the entropy that is induced by  $w$  in terms of potential continuations of this encounter prefix: the lower  $EE$ , the higher the value of  $w$  with respect to its ability of reducing the uncertainty of upcoming messages/actions. Thus, by comparing expectation entropies for different messages in the process of selecting which message to utter, the agent can compare their values or regard the system of all possible messages as an ‘‘encoding’’ for future reactions.

Utility deviation, on the other hand, is defined as the standard deviation between the utilities of all possible continuations of the encounter given  $w$  so that the importance of the potential consequences of  $w$  can be assessed. Its power lies in being closely related to the expected utility of the encounter, while at the same time providing a measure for the *risk* associated with the encounter sequence perceived so far.

Returning to the observation we made regarding the ‘‘request’’ nature of any communicative action in section 2, we can now rephrase this view in terms of the mathematical tools introduced in the above paragraphs: Any message

<sup>4</sup>This is because  $w'$  might involve actions that change the state of the environment. Unfortunately, this definition requires that the agent be able to predict these changes to the knowledge base *a priori*.

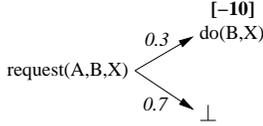


Figure 1: A frame repository based on the SRP, compiled into a tree.

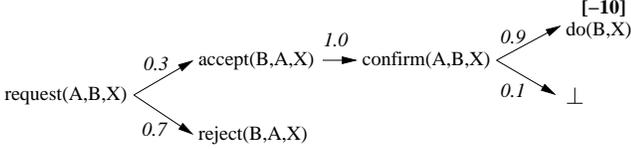


Figure 2: RAP frame repository tree.

$v \in \Sigma$  considered in the context of an encounter has an expectation entropy associated with it, so that  $EE(wv, KB)$  can be used to predict how much using  $v$  will help to “settle” the communication situation, i.e. to reduce the number of potential outcomes of the entire encounter. At the same time  $UD(wv, KB)$  can be used to check how “grave” the effects of different outcomes would be.

By combining these two measures into  $\mathcal{E}$ , the agent can trade off the reduction of uncertainty against sustainment of autonomy depending on its willingness to conform with existing expectations or to deviate in order to pursue goals that contradict the expectations held towards the agent.

## 6. ANALYSIS

To see how the above framework may help interpret the meaning of utterances and guide the agent’s behaviour, we will compare three different interaction scenarios, in which the frame repositories of some agent  $a_1$  have been compiled into the trees shown in figures 1, 2 and 3, respectively (we use trees of interaction trajectories as defined in [2] instead of sets of sequences as a more compact representation). The nodes which represent messages are connected by edges that are labelled with transition probabilities in *italics* (computed using  $count(F)$ ). We use variables  $A, B, X$  etc. to capture several “ground” situations by a single tree. The substitutions that are needed to reconstruct past interactions using the tree are not displayed in the examples, but form part of the private attributes (cf. section 3.1).

Where the direct utility associated with an action is not zero, the increase/decrease in total utility is printed on top of the action in **bold** face in square brackets  $\square$  (if communication preceding these “utility nodes” comes at a cost, this has been already considered in the utility of the leaf node). For simplicity, we also assume the trees presented here to be the result of combining all frames that are consistent with the current knowledge base, i.e. frame conditions have already been checked.

### 6.1 Interaction scenarios

The repository shown in figure 1 summarises experience with a “simple-request” protocol (SRP) where one agent starts by requesting an action  $X$  and the other may simply execute the requested action or end the encounter (the  $\perp$

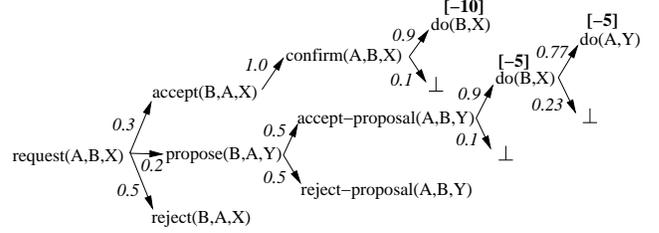


Figure 3: RCOP tree.

symbol is used to denote encounter termination whenever termination probability is below 1.0) – in a sense, this is the most “minimal” protocol one can think of. So far, only 30% out of all requests have been fulfilled, all others went unanswered. We now picture a situation in which agent  $a_1$  is requested by agent  $a_2$  to execute some action, but this action has a utility of  $-10$  for  $a_1$ . Note that the probabilities in the tree are derived from observing *different* interactions where  $a_1$  may have held both participating parties’ roles in different instances, but the utility decrease of 10 units is computed on the grounds of the current situation, by instantiating variable values with agent and action names (e.g.  $A = a_2, B = a_1$  and  $X = deliver(quantity = 100)$ ).

Figure 2 shows a “request-accept” protocol (RAP) that leaves some more options to the requestee as she may accept or reject the request. After confirmation of the requesting agent (which is certain), the requestee executes the request with a probability of 90%; in 10% of the cases, the agent who agreed to fulfil the request is unreliable.

The “request-counter-offer” protocol (RCOP) in figure 3 offers more possibilities still: it includes “accept” and “reject” options, but it also allows for making a proposal  $Y$  that the other agent may accept or reject in turn, and if this proposal is accepted, that other agent is expected to execute action  $Y$  if the first agent executes  $X$ . The distribution between **accept/propose/reject** is now 0.3/0.2/0.5, because it is realistic to assume that in 20% of the cases in which the initial offer would have been rejected in the RAP, the requestee was able to propose a compromise in the RCOP. As before, the requestee fails to perform  $X$  with probability 0.1, and this unreliability is even larger (23%) for the other agent. This is realistic, because the second agent is tempted to “cheat” once its opponent has done her duty. In the aforementioned scenario, we assume that the “compromise” actions  $X$  and  $Y$  (e.g.  $X = deliver(quantity = 50)$ ,  $Y = pay\_bonus$ ) both have utility  $-5.0$ , i.e. the compromise is not better than the original option  $deliver(quantity = 100)$ .

Now let us assume  $a_1$  received the message

**request**( $a_2, a_1, deliver(quantity = 100)$ )

from  $a_2$  who starts the encounter. The question that  $a_1$  finds herself in is whether she should perform the requested action despite the negative utility just for the sake of improving the reliability of the frame set or not<sup>5</sup>.

### 6.2 Entropy decrease vs. utility

<sup>5</sup>Ultimately, this depends on the design of the agent, i.e. in which way this reliability is integrated in utility computation.

First, consider the case where she chooses to perform the action. In the SRP, this would decrease  $UD(\mathbf{request})$  from 5.39 to 5.36<sup>6</sup>, but it would increase  $EE(\mathbf{request})$  from 0.8812 to 0.8895. The total entropy  $\mathcal{E}(\mathbf{request})$  would increase from 4.74 to 4.76. In case of not executing the requested action utility deviation would rise to 5.40, expectation entropy would decrease to 0.8776, and the resulting total entropy would be 4.73.

How can we interpret these changes? They imply that choosing the more probable option  $\perp$  reduces entropy while performing the action increases it. Thus, since most requests go unanswered, doing nothing reassures this expectation. Yet, this increases the risk (utility deviation) of  $\mathbf{request}$ , so  $a_1$ 's choice should depend on whether she thinks it is probable that she will *herself* be in the position of requesting an action from someone else in the future (if e.g., the utility of  $\mathbf{do}$  becomes +10.0 in a future situation and  $a_1$  is requesting that action). But since the difference in  $\Delta\mathcal{E}$ <sup>7</sup> is small (0.02 vs. -0.01), the agent should only consider sacrificing the immediate payoff if it is *highly* probable that the roles will be switched in the future.

Let us look at the same situation in the RAP case. The first difference to note here is that

$$UD(\mathbf{accept}) = UD(\mathbf{confirm}) = 6.40 > 4.76 = UD(\mathbf{request})$$

This nicely illustrates that the “closer” messages are to utility-relevant actions, the greater the potential risk, unless occurrence of the utility-relevant action is absolutely certain. This means that the 0.9/0.1 distribution of  $\mathbf{do}/\perp$  constitutes a greater risk than the 0.7/0.3 distribution of  $\mathbf{reject}/\mathbf{accept}$ , even though  $EE(\mathbf{confirm}) < EE(\mathbf{request})$ !

If  $a_1$  performs the requested action, the total entropy of  $\mathbf{request}$  increases from 4.86 to 4.89, if she doesn't (by sending a  $\mathbf{reject}$ ), it decreases to 4.84. Since this resembles the entropy effects in the SRP very much, what is the advantage of having such a protocol that is more complex?

### 6.3 External paths and path criticality

The advantages of the RAP become evident when looking at the entropies of  $\mathbf{accept}$  and  $\mathbf{confirm}$  after a  $\mathbf{reject}$ , which remain unaffected (since they are located on different paths than  $\mathbf{reject}$ ). So RAP is, in a sense, superior to SRP, because it does allow for deviating from a certain expectation by *deferring* the expectations partly to messages on unaffected *external paths*. Effectively this means that after a  $\mathbf{reject}$ , a  $\mathbf{request}$  becomes riskier in future encounters, but if the agent waits until the  $\mathbf{accept}$  message in a future interaction, she can be as certain of the consequences as she was before. Of course, in the long run this would render  $\mathbf{request}$  almost useless, but if used cautiously, this is precisely the case where autonomy and predictability can be combined to serve the needs of the agents.

The most dramatic changes to entropy values will be witnessed if the agent doesn't perform the action, but promises to do so by uttering an  $\mathbf{accept}$  message:  $\mathcal{E}(\mathbf{request})$  increases from 4.86 to 5.05,  $\mathcal{E}(\mathbf{accept})$  and  $\mathcal{E}(\mathbf{confirm})$  both increase from 3.00 to 3.45. This is an example of how our analysis method can provide information about *path criti-*

<sup>6</sup>The small changes are due to the fact that the frame repository is the product of 100 encounters – a single new encounter induces only small changes to the numerical values.

<sup>7</sup> $\Delta\mathcal{E}$  is defined as the difference between entropies after and before the encounter.

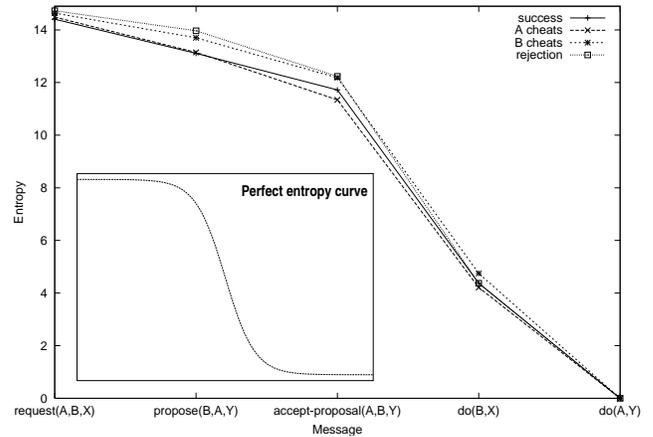


Figure 4: RCOP entropies along “success” path for all four interaction cases.

*cality*: it shows that the normative content of  $\mathbf{accept}$  is very fragile, both because it is closer to the utility-relevant action and because it has been highly reliable so far.

### 6.4 Trajectory entropy shapes

Let us now look at the RCOP and, once more, consider the two alternatives of executing the request right away or rejecting the request. Now, the total entropy decreases from 14.41 to 14.38 and 14.35 in the case of  $\mathbf{accept}/\mathbf{reject}$ , respectively. This is similar to the SRP and the RAP, even though the effects of different options are now less clearly visible (which due to the fact that refusal and acceptance are now more evenly distributed). Also, the total entropy of  $\mathbf{request}$  that is more than three times higher than before (with comparable utility values). This suggests that it might be a good idea to split the RCOP into two frames that start with different performatives, e.g.  $\mathbf{request-action}$  and  $\mathbf{request-proposal}$ .

Of course, the  $\mathbf{propose}$  option is what is actually interesting about the RCOP, and the final step in our analysis will deal with this case. If  $a_1$  analyses the possible runs that include a  $\mathbf{propose}$  message, she will compare the effects of the following encounters on the frame tree with each other:

Short name	Encounter
“success”:	$\mathbf{request}(A, B, X) \dots \rightarrow \mathbf{do}(A, Y)$
“A cheats”:	$\mathbf{request}(A, B, X) \dots \rightarrow \mathbf{do}(B, X)$
“B cheats”:	$\mathbf{request}(A, B, X) \dots \rightarrow \mathbf{accept-proposal}(A, B, Y)$
“rejection”:	$\mathbf{request}(A, B, X) \rightarrow \mathbf{reject-proposal}(B, A, X)$

Figures 4 and 5 show the values of  $\mathcal{E}(w)$  and  $\Delta\mathcal{E}(w)$  (the change in total entropy before and after the encounter) computed for the messages along the path

$$w = \mathbf{propose}(A, B, X) \rightarrow \dots \rightarrow \mathbf{do}(A, Y)$$

A first thing to note is the shape of the entropy curve in figure 4 which is typical of meaningful trajectories. As illustrated by the boxed “perfect” entropy curve, reasonable trajectories should start with an “autonomy” part with high entropy which gives agents several choices, and then continue with a “commitment” part in which entropy decreases rapidly to make sure there is little uncertainty in the consequences of the interaction further on.

Secondly, figure 5 which shows the changes to the node entropies before and after the respective interaction proves that as in the RAP, cheating has a negative impact on entropies. Moreover, the effects of “A cheats” appear to be much worse than those of “B cheats” which reassures our intuition that the closer utterances are to the final outcome of the encounter, the more critical will the expectations about them be.

Thirdly, as before, the “rejection” dialogue and the “success” dialogue are acceptable in the sense of decreasing entropies of `propose` and `accept-proposal` (note that the small entropy increase of `request` is due to the 0.1/0.23 probabilities of cheating after `accept-proposal` and `do(B, X)`). The fact that “success” is even better than “rejection” suggests that, in a situation like this, there is considerable incentive to compromise, if the agent is willing to sacrifice current payoff for low future entropies.

## 6.5 Conflict potential

Looking at the plots in figure 5, a more general property of communication becomes evident: we can imagine an agent reckoning what to do in an ongoing encounter who evaluates the potential entropy changes to relevant paths after each message.

For this purpose, let  $\mathcal{F}'$  be the result of adding a new encounter  $w'$  to the current repository  $\mathcal{F}$  (we assume  $\text{count}(w)$  and  $\text{cond}(w)$  are computed as described in section 3). The *entropy change* induced on trajectory  $w \in \Sigma^*$  by performing encounter  $w' \in \Sigma^*$  is defined as

$$\Delta\mathcal{E}_{\mathcal{F}}(w, w') = \mathcal{E}_{\mathcal{F}'}(w) - \mathcal{E}_{\mathcal{F}}(w)$$

This quantity provides a measure of the *expectation-affirmative* or *expectation-negating* character of an utterance. In other words, it expresses to which degree the agents are saying “yes” or “no” to an existing expectation.

The *conflict potential* of an encounter can be derived by comparing the *expected* entropy change to the *occured* entropy change, and thus revealing to which degree the agents exceeded the expected change to expectation structures. We can define the conflict potential exerted by the occurred encounter  $w''$  on encounter  $w$  if the expected encounter was  $w'$  as

$$\mathcal{CP}_{\mathcal{F}}(w'', w', w) = \int_{w[1]}^{w[|w|]} \Delta\mathcal{E}_{\mathcal{F}}(w, w'') - \Delta\mathcal{E}_{\mathcal{F}}(w, w') dw_i$$

This is the area under the “conflict curve” in figure 5, that computes

$$\Delta\mathcal{E}(\text{“success”}, \text{“A cheats”}) - \Delta\mathcal{E}(\text{“success”}, \text{“success”})$$

This curve shows how the difference between expected and actual entropy change grows larger and larger, until the encounter is terminated unsuccessfully. This increases the probability that the participating agents will stop trusting the expectation structures, and that this will inhibit the normal flow of interaction, especially if  $\mathcal{CP}$  is large for several paths  $w$ .

A noteworthy property of this view of conflict is that in cases where, for example, entirely new performatives are tried out, the conflict potential is 0 because the expected entropy change (which is very large, because the agents know nothing about the consequences of the new performative) is identical to that actually experienced. So what matters about conflict is not whether the expectations associated

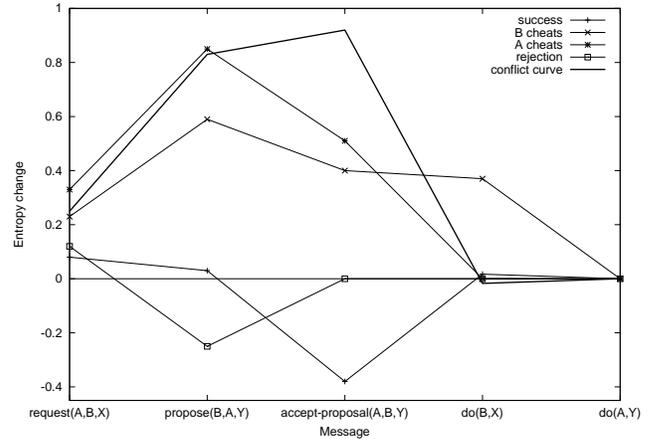


Figure 5: RCOP entropy changes  $\Delta\mathcal{E}$  along the `propose(A, B, X) → ... → do(A, Y)` path.

with a message are clear, but rather whether the effect of uttering them comes close to our expectations about that effect on the expectation structures – a property we might call *second-order expectability*.

## 7. CONCLUSIONS

In this paper, we have presented a novel model for defining and analysing the semantics of agent communication that is based on the experience of agents (or external observers) with the ongoing communicative behaviour in a MAS. This model is very general in that it only relies on a statistical analysis of observed communication and makes no assumptions regarding the domain of application. It does impose some restrictions on the design of the agents by assuming them to fulfil minimal InFFrA criteria, i.e. to be able to record and statistically analyse observed interaction sequences. Subsequently, we defined entropy measures that proved to be very powerful when analysing different interaction situations.

The central conclusion from this analysis is that appropriate expectation structures are structures that leave enough room for autonomy but are at the same time reliable once certain paths are chosen by interactants – they are *autonomy-respecting* and *contingency-reducing* at the same time.

Such structures are characterised by the following features:

- *external paths* whose entropies remain unaffected by agent’s choices in the early phases of an encounter;
- *low expectation entropy* where *utility deviation* is high – the higher the potential loss or gain of a path, the more predictable it should be (esp. towards the end of an encounter);
- *alternatives* for different utility configurations; paths that are likely to have a wider range of acceptable outcomes for the partners (e.g. by containing `do`-actions for all parties, cf. RCOP) are more likely to become stable interaction procedures, as they will be used more often.

One of the strengths of our framework is that empirical semantics suggest including considerations regarding the usefulness of “having” a certain semantics in the utility-guided decision processes of agents. Agents can compute entropy measures of message trajectories prior to engaging in the actual communication and assess the first- and second-order effects of their actions under current utility conditions or using some long-term estimate of how the utility function might change (i.e. which messages they will want to be reliable in the future). The fact that agents consider themselves being in the position of someone else (when computing entropy changes) links the protocol character of communication to the self-interested decision-making processes of the participating agents, thus making communication truly meaningful.

This *role-taking* [10] in communication necessitates the use of speech acts in our framework to generalise over agents. The “content” slot of the speech acts used is necessary to parametrise them with additional symbols, thus constructing a *topic* of the conversation, the ultimate objective being to use performatives as markers for certain interaction trajectories (and probability distributions) that occur in a variety of situations. This allows us to view frame repositories as ontologies of performatives with a probabilistic, quantitative annotation – rotating the frame trees of section 6 clock-wise by 90° suffices to make them look like semantic network style ontologies. In our understanding, the meaning of things always depends on “how we act upon them”, so the relationship between expectation-based semantics and ontologies certainly goes beyond the above observation and deserves further investigation.

Critics may object that there is more to communication than statistical correlations between messages and actions, because the purpose of communication is not always physical action (but also, e.g., exchange of information) and that many (in particular, normative) aspects of communication are neglected by reducing semantics to an empirical view. We still believe that such empirical semantics can serve as a “greatest common denominator” for divergent semantic models of different agents, if no other reliable common knowledge about the meaning of messages is available. If, on the other hand, such knowledge is available, our framework can still be used “on top” of other (mentalistic, contractual) semantics.

Our definition of conflict potential is another strength of our framework, because conflict is defined as the situation in which agents’ trust in the communication system becomes weaker. Sudden, unexpected “jumps” in entropies that become bigger and bigger render the expectation structures questionable, the meaning of communicative acts becomes more and more ambiguous. This definition of computational conflict is very powerful because it does not resort to domain-dependent resource or goal configurations and is defined solely in terms of communicative processes. However, we have not yet suggested resolution mechanisms for such conflict interactions. We believe that *reifying* conflict in communication (i.e. making it the *topic* of communication) is key when it comes to conflict resolution. We are currently developing protocols in which agents exchange information about their entropy changes in order to resolve conflicts and these seem very promising.

Another interesting issue to look at in future work is the emergence of system-wide, stable interaction patterns from

scratch, in particular analysing how and when agents introduce new types of messages and message sequences, and how they re-combine expectation structures to make sense out of new communication structures.

Finally, a more elaborate decision-theoretic framework is needed that views expectation structures as uncertain environments for which rational decision-making processes with provable optimality criteria can be defined. We are currently exploring the possibilities of adapting reinforcement learning algorithms for this purpose.

## 8. REFERENCES

- [1] J. L. Austin. *How to do things with Words*. Clarendon Press, 1962.
- [2] W. Brauer, M. Nickles, M. Rovatsos, G. Weiß, and K. F. Lorentzen. Expectation-Oriented Analysis and Design. In *Proceedings of the 2nd Workshop on Agent-Oriented Software Engineering (AOSE-2001)*, volume 2222 of *LNAI*, 2001. Springer-Verlag, Berlin.
- [3] P. R. Cohen and H. J. Levesque. Communicative actions for artificial agents. In *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95)*, pages 65–72, 1995.
- [4] P. R. Cohen and C. R. Perrault. Elements of a Plan-Based Theory of Speech Acts. *Cognitive Science*, 3:177–212, 1979.
- [5] N. Fornara and M. Colombetti. Operational specification of a commitment-based agent communication language. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS’02)*, pages 536–542, 2002. ACM Press.
- [6] L. Gasser. Social conceptions of knowledge and action: DAI foundations and open systems semantics. *Artificial Intelligence*, 47:107–138, 1991.
- [7] E. Goffman. *Frame Analysis: An Essay on the Organisation of Experience*. Harper and Row, New York, NY, 1974.
- [8] F. Guerin and J. Pitt. Denotational Semantics for Agent Communication Languages. In *Proceedings of the Fifth International Conference on Autonomous Agents (Agents’01)*, pages 497–504. ACM Press, 2001.
- [9] N. Luhmann. *Social Systems*. Stanford University Press, Palo Alto, CA, 1995.
- [10] G. H. Mead. *Mind, Self, and Society*. University of Chicago Press, Chicago, IL, 1934.
- [11] H.-J. Müller and R. Dieng, editors. *Computational Conflicts – Conflict Modeling for Distributed Intelligent Systems*. Springer-Verlag, Berlin, 2000.
- [12] J. Pitt and A. Mamdani. Designing agent communication languages for multi-agent systems. In *Proceedings of the Ninth European Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW-99)*, volume 1647 of *LNAI*, pages 102–114. Springer-Verlag, 1999.
- [13] J. Pitt and A. Mamdani. A protocol-based semantics for an agent communication language. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI-99)*, 1999.
- [14] A. S. Rao and M. P. Georgeff. BDI agents: From theory to practice. In *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95)*, pages 312–319, 1995.
- [15] M. Rovatsos, G. Weiß, and M. Wolf. An Approach to the Analysis and Design of Multiagent Systems based on Interaction Frames. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS’02)*, 2002. ACM Press.
- [16] M. Singh. A social semantics for agent communication languages. In *Proceedings of the IJCAI Workshop on Agent Communication Languages*, 2000.
- [17] M. P. Singh. A semantics for speech acts. *Annals of Mathematics and AI*, 8(1–2):47–71, 1993.