

Reasoning about Communication – A Practical Approach based on Empirical Semantics

Felix Fischer and Michael Rovatsos

Department of Informatics
Technical University of Munich
85748 Garching, Germany
{rovatsos, fischerf}@cs.tum.edu

Abstract. Given a specification of communication rules in a multiagent system (in the form of protocols, ACL semantics, etc.), the question of how to design appropriate agents that can operate on such a specification is a very important one. In *open systems*, the problem is complicated even further by the fact that adherence to such a supposedly agreed specification cannot be ensured on the side of other agents.

In this paper, we present an architecture for dealing with communication patterns that encompass both a surface structure of admissible message sequences as well as logical constraints for their application. This architecture is based on the InF-FrA social reasoning framework and the concept of interaction frames. It assumes an *empirical semantics* standpoint by which the meaning of communication is pragmatically interpreted through decision-theoretic optimality considerations of a reasoning agent. We introduce the abstract architecture and a formal model and present experimental results from a complex domain to illustrate its usefulness.

1 Introduction

The process of designing agent communication languages, interaction protocols and conversation policies is primarily concerned with what goes on *between* rather than *inside* agents. It is therefore only natural that ACL research has rarely attacked the problem of how to build agents in accordance with a given set of semantic rules, protocol structures, etc.

In the light of *open* multiagent systems, in which agents are not controlled by a central entity or pursue common goals and need not be benevolent, this design problem is complicated even further. Not knowing how other agents will apply a given specification concerning the semantic and pragmatic properties of the provided means of communication, the question for an agent is not only how to apply it by himself, but also whether to comply with the supposedly agreed specification at all.

This is exactly the issue this paper deals with: *Given the specification of a number of communication patterns together with logical constraints for their application, how can we design an agent that uses them appropriately to further his own goals, not knowing whether other agents will comply?* As a possible answer to this question, we propose an agent architecture based on the concept of *empirical semantics* [10]. This architecture can be seen as a “communication reasoning” component that takes as inputs the

belief state and utility function of an agent as well as a set of models of communication patterns (so-called *interaction frames*) and outputs communication decisions. The architecture is characterised by two essential features:

1. It uses decision-theoretic principles to make optimal action decisions while taking previous communication *experience* into account.
2. It combines empirical knowledge with *a priori* specified logical constraints for communication patterns.

Most importantly, the reasoning heuristics used in this architecture are computationally tractable and can be implemented directly. We thus contribute to the task of engineering agents that are able to operate flexibly on a given set of communication protocols and semantic rules and under the assumption that the empirical semantics view is followed.

The remainder of this paper is structured as follows: In section 2 we review the foundations of the empirical semantics approach. Subsequently, we introduce the InFFrA social reasoning framework that the proposed architecture is based on. Section 4 establishes a formal model of empirical semantics called m^2 InFFrA that uses InFFrA frames to capture probabilistic expectations about communication. In section 5 we then present the apparatus necessary for making decisions in this model. Experimental results from a concrete implementation of the model in a complex domain are given in section 6 to underline its practical usefulness. Section 7 rounds up with some closing remarks and conclusions as well as an outlook to future work.

2 Empirical Semantics

In [10], we proposed a new model for communication based on *empirical semantics*. The basic idea behind this model is that communication obtains its meaning through the interactions within which it occurs. The central instrument to establish this meaning are the expectations constructed by agents participating in or observing an interaction. These expectations “contain” the current semantics of communication in a given social context and their evolution over time mirrors the evolution of meaning.

To make things a bit more concrete, we quote the following central elements of the empirical semantics approach from [10]:

1. 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 levels of effects can be distinguished:
 - (a) The immediate reactions of other agents and oneself to the message.
 - (b) 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.
2. 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.
3. 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.

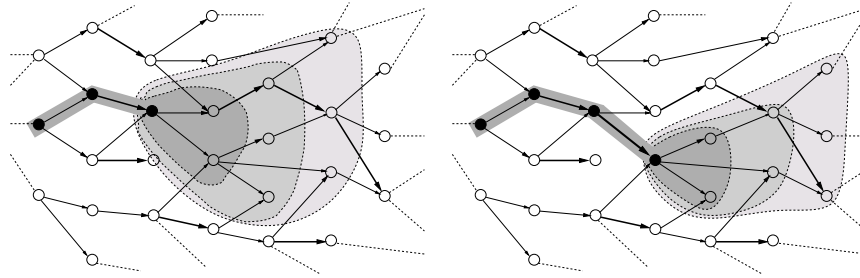


Fig. 1. Empirical semantics and its evolution

In other words, our view is that the semantics of communication can be reduced to its pragmatics as perceived by agents who are using communication. While this is an overtly restrictive interpretation of meaning for general communication, it can be very useful for practical agent reasoning in *open* multiagent systems, in which “normative” semantics (e.g. commitment-based speech act semantics [12]) may be proven wrong by others’ inability or unwillingness to comply with them. This is because of the fact that it can be used as a *minimal-assumption* method for reasoning about communication which at least ensures that the agent will not compromise his own welfare for the sake of complying with a pre-specified semantics (when others do not comply, either).

Generally speaking, if the meaning of an utterance (or sequence of utterances) lies in the expected consequences, what is needed to capture semantics is a model of causality and correlation that allows to predict these consequences for a given (sequence of) utterance(s). This model will then be adapted with new observations so as to enable the agent to extrapolate past correlations into the future.

Expectation networks [6, 7] have been suggested as a very general method of capturing this kind of expectations that relies on a probabilistic model of “continuations” between interrelated messages derived from statistical observation. Instead of going into the details of this formalism, we sketch the basic idea using the illustrations of figure 1. In this figure, nodes represent communicative actions, edges correlations between them (variable line width is used to indicate different degrees of correlation). The shaded node sequence is used to describe the recently observed portion of the network, e.g. an ongoing conversation. The decreasingly dark shaded regions of predicted future actions denote that predictions regarding “distant” events are increasingly vague. The transition from the situation shown on the left to that on the right occurs upon observation of a new action that is appended to the currently relevant path. With this new observation, the correlation between the message previously observed and the current message increases compared to alternatives that did not occur.

Taking the standpoint of empirical semantics enables an agent to adapt his expectations with new communication experience and to adjust his own behaviour to the expected reactions of others. However, this conceptual view of semantics does not say anything about how agents can use such expectations in practice.

3 The InFFrA Architecture

The Interaction Frames and Framing Architecture InFFrA [11] has originally been developed as a meta-framework for social reasoning architectures based on the notions of “interaction frames” and “framing”.

The central idea behind InFFrA is to employ models of classes of interaction called interaction frames to guide agents’ social behaviour. The process of applying frames appropriately in interaction situations is referred to as framing. In the abstract architecture InFFrA, a frame is a data structure that contains information about

- the possible courses of interaction (so-called *trajectories*) characteristic to a particular frame,
- *roles and relationships* between the parties involved in an interaction of this class,
- *contexts* within which the interaction may take place, and
- *beliefs*, i.e. epistemic states of the interacting parties.

In computational terms, the trajectory model is usually a representation of a set of admissible message and action sequences, while the latter three elements can be collapsed into a single set of logical constraints which then have to be verified using the agent’s internal belief state (usually represented by the contents of a knowledge base).

InFFrA makes use of a number of frame-based data structures to conduct the steps necessary for framing:

- the *active frame*, the unique frame currently activated to describe the expected course of events,
- the *perceived frame*, an interpretation of the currently observed state of affairs,
- the *difference model* containing the differences between perceived frame and active frame,
- the *trial frame*, used when alternatives to the current frame are sought for,
- and the *frame repository*, in which the agent locally stores its frame knowledge.

Using these data structures, an InFFrA agent performs the following steps in each reasoning cycle:

1. *Interpretation & Matching*: Update the perceived frame and compare it with the active frame.
2. *Assessment*: Assess the usability of the active frame in terms of
 - (i) adequacy (compliance of frame conditions with the current situation),
 - (ii) validity (the degree to which the active frame’s trajectory matches the perceived encounter) and
 - (iii) desirability (depending on whether the implications of the frame correspond to the agent’s private goals).
3. *Framing decision*: If the active frame seems appropriate, continue with 5. Else, proceed with 4 to find suitable alternatives.
4. *Adjustment/Re-framing*: Search the frame repository for better frames. “Mock-activate” them as trial frames iteratively and go back to 1; if no suitable frame is found, end the encounter.
5. *Enactment*: Derive action decisions by applying the active frame.

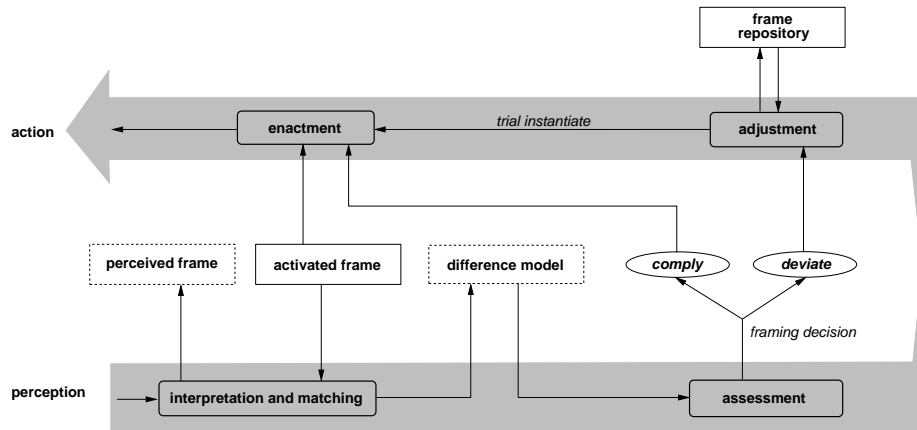


Fig. 2. Overview of the framing process

This entire (simplified) framing process is depicted in figure 2. It should be emphasised that InFFrA only describes the social layer of agent reasoning; in order to obtain an integrated agent architecture, it has to be combined with a suitable component for local rational reasoning (e.g. a BDI [8] reasoner), so that the agent’s private goals and preferences can be taken into consideration during the *assessment* phase. Looking at InFFrA from an empirical semantics point of view, we can re-interpret the frames in an agent’s repository as the expectations about communication patterns that the agent holds. This provides us with a reasoning scheme for processing expectations which is missing in the pure “expectation-modelling” view of expectation networks laid out in the previous section. Moreover, if we ensure that frame conceptions are adapted with new experience, InFFrA is ideally suited for applying pre-specified communication patterns (e.g. traces of protocol execution as envisioned by the designer) in a goal-oriented fashion while making sure that others’ (or one’s own) deviance from these normative patterns influences the expectations about validity and relevance of these patterns in the future.

4 A Formal Model of Frame-Based Empirical Semantics

To obtain a formal computational model for representing and reasoning about communicative expectations based on InFFrA, we have developed m^2 InFFrA, a simple yet expressive model for describing two-party, discrete, turn-taking *encounters* which can be thought of as conversations between two agents. m^2 InFFrA uses a sequence of message patterns (i.e. messages containing variables) as its trajectory model specifying the surface structure of the encounters described by a particular frame, together with a list of *substitutions* to capture concrete values of these variables in previously experienced interactions. Each substitution also corresponds to a set of logical *conditions* that were required for and/or precipitated by execution of the trajectory in the respective encounter. Finally, *trajectory occurrence* and *substitution occurrence* counters record the

frequency with which the frame has occurred in the past. Formally, a frame in $m^2InFFrA$ is defined as follows:

Definition 1. A frame is a tuple $F = (T, \Theta, C, h_T, h_\Theta)$, where

- $T = \langle p_1, p_2, \dots, p_n \rangle$ is a sequence of message patterns $p_i \in \mathcal{M}$, the trajectory of the frame,
- $\Theta = \langle \vartheta_1, \dots, \vartheta_m \rangle$ is an ordered list of variable substitutions,
- $C = \langle c_1, \dots, c_m \rangle$ is an ordered list of condition sets, such that $c_j \in 2^{\mathcal{L}}$ is the condition set relevant under substitution ϑ_j ,
- $h_T \in \mathbb{N}^{|T|}$ is a trajectory occurrence counter list counting the occurrence of each prefix of the trajectory T in previous encounters, and
- $h_\Theta \in \mathbb{N}^{|\Theta|}$ is a substitution occurrence counter list counting the occurrence of each member of the substitution list Θ in previous encounters.

In this definition, \mathcal{M} is a language of speech-act [1] like message and action patterns of the form $\text{perf}(A, B, X)$ or $\text{do}(A, Ac)$. In the case of messages (i.e. exchanged textual signals), perf is a performative symbol (request, inform, etc.), A and B are agent identifiers or agent variables and X is the propositional content of the message taken from a logical language \mathcal{L} . In the case of physical actions (i.e. actions that manipulate the physical environment) with the special “performative” do , Ac is the action executed by A (a physical action has no recipient as it is assumed to be observable by any agent in the system). Both X and Ac may contain non-logical substitution variables that are used for generalisation purposes (as opposed to logical “content variables” used by agents to indicate quantification or to ask for a valid binding). We further use $\mathcal{M}_c \subset \mathcal{M}$ to denote the language of “concrete” messages that agents use in communication (and that do not contain variables other than “content variables”).

To illustrate these concepts and to explain the semantics of a frame, we will consider an example of how the FIPA contract net protocol [4] can be implemented in $m^2InFFrA$. For ease of presentation, we will write $T(F)$, $C(F)$, etc. to denote the respective elements of a frame F and use the compact notation $\langle T_h(F), C(F), \Theta_h(F) \rangle$ instead of $(T, C, \Theta, h_T, h_\Theta)$, where

$$T_h(F) = \xrightarrow{h_T(F)[1]} p_1 \xrightarrow{h_T(F)[2]} p_2 \cdots \xrightarrow{h_T(F)[n]} p_n$$

and

$$\Theta_h(F)[i] = \xrightarrow{h_\Theta(F)[i]} \Theta(F)[i].$$

Table 1 shows an interaction frame for the success path of the contract net protocol, the following should be noted about this example:

- As can be seen, the first condition set (corresponding to the empty substitution) contains feasibility preconditions of the respective performatives (in FIPA-SL [3] with additional timestamps of the form $@i$ indicating at which trajectory step a condition has to hold), as far as they are relevant for frame execution.
- Definition 1 does not constrain which conditions are to be stored for a specific enactment of a frame (even allowing for empty condition sets), and the task of extracting *relevant* or even *crucial* information from the agent’s knowledge is clearly

$$\begin{aligned}
F_{cn} = & \left\langle \left\langle \xrightarrow{5} \text{cfp}(A_1, A_2, \langle R, P \rangle) \xrightarrow{3} \text{propose}(A_2, A_1, Q) \right. \right. \\
& \left. \xrightarrow{3} \text{accept-proposal}(A_1, A_2, Q) \xrightarrow{2} \text{do}(A_2, A_1, R) \right\rangle, \\
& \left\langle \{ \iota X (P = Q) = Y, \right. \\
& \quad \neg B_{ref} A_1 (\text{any } X I_{A_2} \text{Done}(R, P)) \wedge \neg B_{A_1} I_{A_2} \text{Done}(R) @1, \\
& \quad B_{A_2} I_{A_2} \text{Done}(R, Q) @2, \\
& \quad B_{A_1} I_{A_1} \text{Done}(R, Q) \wedge B_{A_1} I_{A_2} \text{Done}(R, Q) @3, \\
& \quad \left. B_{A_2} Q @4 \}, \{ \}, \{ \text{damaged}(\text{tires}) \} \right\rangle, \\
& \left\langle \xrightarrow{0} [] \right. \\
& \quad \xrightarrow{1} [A_1 / \text{agent}_1, A_2 / \text{agent}_2, P / \text{priceOf}(\text{tire}, X), \\
& \quad \quad Q / \text{priceOf}(\text{tire}, 75), R / \text{sell}(\text{tire}, 4)], \\
& \quad \xrightarrow{1} [A_1 / \text{agent}_3, A_2 / \text{agent}_1, P / \text{priceOf}(\text{tire}, X), \\
& \quad \quad \left. Q / \text{priceOf}(\text{tires}, 400), R / \text{replace}(\text{tires})] \right\rangle,
\end{aligned}$$

Table 1. Interaction frame for the success path of the FIPA contract net protocol

nontrivial. However, the reasoning framework to be defined in the following section primarily uses conditions to identify similarities in encounters, while expectation is drawn from utilities obtained during frame execution, making this approach less sensitive to the way conditions are selected.

As for the different failure cases covered by the contract net protocol, these could either be modelled implicitly by using timeouts and interpreting “silence as disapproval” or explicitly by virtue of additional frames. For example, the frame of table 2 models the refusal to submit a proposal.

Together, F_{cn} and F_{cr} capture the following observations about previous encounters:

- Five encounters started with a message matching $\text{cfp}(A_1, A_2, \langle R, P \rangle)$, three of them continued with a proposal by A_2 , the other two with a refusal. In two of the former three cases, the proposal was accepted by A_1 and the respective physical action carried out by A_2 .
- One encounter has terminated after the second message or was continued with a message not matching $\text{accept-proposal}(A_1, A_2, Q)$.
- For four of the five encounters, substitutions are available. Two contracts were made, one about the delivery of four tires for a price of 75 each, one about the replacement of a set of tires for 400. Two calls for proposals for the delivery of four tires were refused because no tires were available (which is a generalisation over two encounters, leaving A_2 unspecified).

To use $m^2\text{InFFrA}$ frames for the prediction of future encounters, however, we need a formal model of their *prospective* rather than *retrospective* semantics. Here, the idea is (as in [10]) to use an entire repository $\mathcal{F} = \{F_1, \dots, F_n\}$ of frames, each representing a set of message/action sequences by virtue of the substitutions that can still be

$$\begin{aligned}
F_{cr} = & \langle \langle \xrightarrow{5} \mathbf{cfp}(A_1, A_2, \langle R, P \rangle) \xrightarrow{2} \mathbf{refuse}(A_2, A_1, Q) \rangle, \\
& \langle \{ \neg B_{ref} A_1 (any X I_{A_2} Done(R, P)) \wedge \neg B_{A_1} I_{A_2} Done(R) @1, \\
& B_{A_2} (\neg Feasible(R) \wedge Q \wedge \neg Done(R) \wedge \neg I_{A_2} Done(R)) @2 \}, \\
& \{ B_{A_1} (inStock(tire)) @1 \} \rangle, \\
& \langle \xrightarrow{0} [] \rangle \\
& \xrightarrow{2} [A_1/agent_1, P/priceOf(tire, X), \\
& Q/\neg inStock(tire), R/sell(tire, 4)]
\end{aligned}$$

Table 2. Interaction frame modelling the refusal to submit a proposal in the FIPA contract net protocol

applied to its trajectory. Given an *encounter prefix*, i.e. a sequence of messages already uttered in the current encounter, and the agent's current belief state, we can filter out those paths that either (a) do not match the encounter prefix or (b) are labelled with logical conditions not satisfied under current knowledge base content. Considering the remaining (i.e. relevant) paths, we can then assign probabilities to all the possible *encounter continuations* (or postfixes) using the counter values of $F \in \mathcal{F}$. Additionally, domain-dependent *case-based reasoning* [5] techniques can be applied by introducing a similarity measure on messages, such that different instances of a postfix pattern have different probabilities depending on their similarity with past enactments of the respective frame. Hence, \mathcal{F} represents a simplified version of an expectation network that has the form of a tree.

More formally, let $w \in \mathcal{M}_c^*$ the encounter prefix and

$$\vartheta_{fixed}(F, w) = \mathit{unifier}(w, T(F)[1:|w|])$$

the most general unifier (MGU) of w and the corresponding trajectory prefix $T(F)[1:|w|]$ of F (where $\mathit{unifier}(\cdot, \cdot)$ returns the most general unifier for two message patterns or sequences thereof, or \perp if they cannot be unified).

For a given knowledge base $KB \in 2^{\mathcal{L}}$ describing the belief state of an agent (KB is assumed to be encoded in the same propositional language \mathcal{L} as the content of messages for reasons of simplicity), this allows us to define

$$\Theta_{poss}(F, KB, w) = \{ \vartheta \mid \exists \vartheta'. \vartheta = \vartheta_{fixed}(F, w)\vartheta' \wedge \exists i. KB \models C[i]\vartheta \}$$

as the set of substitutions still *possible* under F , KB , and w . The elements of Θ_{poss} are extensions of ϑ_{fixed} for which at least one condition in $C(F)$ is satisfied.¹

¹ We use the notation $L\vartheta$ here for the result of applying ϑ to each element of a list or set L , and $\vartheta\vartheta'$ for the substitution that results from applying ϑ' after ϑ . Further, we implicitly assume that only *minimal* substitutions are considered for ϑ' , only replacing variables that actually occur in $T(F)$ or in some $C(F)[i]$.

For a given *similarity measure* $\sigma : \mathcal{M}^* \times \mathcal{M}^* \rightarrow \mathbb{R}$ on message pattern sequences, we can define

$$\sigma(\vartheta, F) = \sum_{i=1}^{|\Theta(F)|} \overbrace{\sigma(T(F)\vartheta, T(F)\Theta(F)[i])}^{\text{similarity}} \overbrace{h_{\Theta(F)}[i]}^{\text{frequency}} \overbrace{c_i(F, \vartheta, KB)}^{\text{relevance}}$$

to assess to which extent ϑ is “applicable” to F . In this definition, $h_{\Theta(F)}[i]$ is used to take the frequency of a past condition/substitution into account and c_i expresses how relevant a particular frame condition $C[i]$ is in determining the applicability of ϑ .² This quantity can be used to derive a conditional probability distribution over the different substitutions ϑ F may be enacted under:

$$P(\vartheta|F, w) = \begin{cases} \alpha \cdot \sigma(\vartheta, F) & \text{if } \vartheta \in \Theta_{\text{poss}}(F, KB, w) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

for some normalisation constant α . Finally, combining this formula with the frequency

$$P(F|w) = \begin{cases} \frac{h_{\mathcal{T}(F)}[|T(F)|]}{\sum_{F' \in \mathcal{F}, \text{unify}(T(F')[1:|w|], w)} h_{\mathcal{T}(F')}[|T(F')|]} & \text{if } \text{unify}(T(F)[1:|w|], w) \\ 0 & \text{else} \end{cases}$$

of F matching any past encounter starting with w allows us to compute the *continuation probability* with which an encounter that started with w will be concluded with w' :

$$P(w'|w) = \sum_{F \in \mathcal{F}, ww' = T(F)\vartheta} P(\vartheta|F, w)P(F|w)$$

Looking back at figure 1, this equation defines a probability distribution for the possible continuations of an encounter given a current “path” in the expectation network induced by \mathcal{F} . Next, we will show how this semantics can be used to conduct rational reasoning about communication in m²InFFrA.

5 Reasoning and Making Decisions About Communication

Based on the formal model presented in the previous section, the general principles of InFFrA can be concretised and embedded into an agent architecture to endow agents with the ability to reason and make decisions about communication. To illustrate how this is done, we will again go through the individual steps of the abstract reasoning cycle depicted in figure 2.

² A simple definition of c_i , considering only those conditions currently satisfied, is for example given by

$$c_i(F, \vartheta, KB) = \begin{cases} 1 & \text{if } KB \models C(F)[i]\Theta(F)[i]\vartheta \\ 0 & \text{otherwise.} \end{cases}$$

Interpretation and Matching/Assessment At the beginning of each reasoning cycle, the knowledge base KB and the encounter prefix w are updated from the peer’s last utterance.

As for matching and assessment, the agent checks if $\Theta_{poss}(F_a, KB, w) \neq \emptyset$, i.e. if the trajectory $T(F_a)$ of the active frame F_a matches w and the remaining steps of $T(F_a)$ can still be executed under KB . Considerations about the *desirability* of F_a could also play a role in the framing decision, and a possible definition for such a desirability criterion will be given in our description of frame enactment.

Returning to the example of the previous section, we consider an encounter with $\mathcal{F} = \{F_{cn}, F_{cr}\}$ and $F_a = F_{cn}$ for both agents and prefix $w = \text{cfp}(agent_2, agent_1, (\text{sell}(\text{tire}, 1), \text{priceOf}(\text{tire}, X)))$, so that $agent_1$ can successfully match F_a against w . If for example $agent_1$ did only sell tires in fours, executability of F_a would still fail and require a re-framing.

Adjustment/Re-Framing The idea behind grouping different courses of interaction into frames is to exploit the fact that (usually) similar types of interaction (e.g. negotiation dialogues, contracting, etc.) exist which differ only in the specific content of messages, but not in what they achieve for the interacting parties. This hierarchical view enables agents to optimise *within* the current frame while disregarding other frames during a conversation. However, if the active frame cannot be carried out any longer, the search space for appropriate continuations has to be expanded.

In this case, a variety of frame selection heuristics can be used to find suitable alternatives. In our implementation, experimental results for which are presented in the following section, we use hierarchical reinforcement learning techniques [13] to learn an optimal frame selection strategy over time. Also, we construct new frames through concatenation in a planning-like manner to achieve the original goal of a conversation that went awry. An extensive treatment of the resulting architecture can be found in [2].

At the end of an encounter – in order to maintain a concise model of past interactions – the active frame is augmented by a substitution that unifies its trajectory with w , along with a set of conditions that were required for or precipitated by the execution.

Enactment If the active frame contains no further steps, the agent simply terminates the encounter, as is the case if no active frame could be found. Else, the next message or action is chosen by applying the locally optimal substitution ϑ^* to the next step of the active frame’s trajectory.

To determine ϑ^* , we assume that the agent’s preference towards different world states is expressed by means of a real-valued *utility function* $u : \mathcal{M}^* \times 2^{\mathcal{L}} \rightarrow \mathbb{R}$, where $u(w, KB)$ is the utility associated with a message sequence w being executed for initial knowledge base KB .

Since some variables of ϑ^* will be bound by the agent himself, while concrete values for others will be “selected” by his peer with a certain conditional probability, the optimal substitution is defined as the one with the highest *expected* utility. Normally, this notion of expected utility will also be used during frame assessment to determine the desirability of a specific substitution or entire frame. For example, one could force a re-framing unless executing the postfix of the active frame under the most desirable substitution yields a positive utility.

If we write Θ_s and Θ_p for the sets of possible substitutions the agent and his peer in the current encounter can apply, respectively, the expected utility of executing a frame F under $\vartheta_s \in \Theta_s$ is given by

$$E[u(\vartheta_s|F, w, KB)] = \sum_{\vartheta_p \in \Theta_p} u(\text{postfix}(T(F), w)\vartheta_s\vartheta_p, KB) \cdot P(\vartheta_p|\vartheta_s, F, w),$$

where $\text{postfix}(T(F), w)$ is the postfix of F corresponding to prefix w (which can be determined by applying the most general unifier of w and the corresponding prefix of $T(F)$ to the respective rest) and $P(\vartheta_p|\vartheta_s, F, w)$ is the probability with which the peer will conditionally choose some substitution $\vartheta_p \in \Theta_p$ depending on the agent's choice $\vartheta_s \in \Theta_s$.

An approximation of $P(\vartheta_p|\vartheta_s, F, w)$ can be computed from the past cases stored in F . Using Bayes' rule and applying equation 1 to both numerator and denominator, we can approximate

$$P(\vartheta_p|\vartheta_s, F, w) = \frac{\sigma(\vartheta_{\text{fixed}}(F, w)\vartheta_s\vartheta_p, F)}{\sum_{\vartheta} \sigma(\vartheta_{\text{fixed}}(F, w)\vartheta_s\vartheta, F)}$$

if $\vartheta_{\text{fixed}}(F, w)\vartheta_s\vartheta_p \in \Theta_{\text{poss}}(F, KB, w)$ (and 0, else). By means of standard expected utility maximisation, the optimal substitution ϑ^* is then given by

$$\vartheta^*(F, w, KB) = \arg \max_{\vartheta_s \in \Theta_s} E[u(\vartheta_s|F, w, KB)].$$

Concluding the above example under the assumption that no re-framing was necessary, *agent*₁ now has to select an appropriate proposal. That is, each of the elements of Θ_{poss} yields a different price for a tire, and the search for the best substitution is a search for the highest possible price (as u will usually increase with higher profit) such that *agent*₂ will still accept (as the probability for the accept will usually decrease with an increasing price) and will be guided by similar past cases stored in $\Theta(F_{cn})$, i.e. by past prices for tires.

6 Experimental results

In order to show their performance in practice, the concepts presented so far have been implemented and tested in the multiagent-based link exchange system **LIESON** [9]. In this system, agents representing Web sites engage in communication to negotiate over mutual linkage with the end of increasing the popularity of one's own site and that of other preferred sites.

Available physical actions in this domain are the addition and deletion of numerically rated links originating from one's own site and the modification of ratings (where the probability of attracting more traffic through a link depends on the rating value).

LIESON provides a highly dynamic and complex interaction testbed for the following reasons:

- Agents only have a partial and incomplete view of the link network. In particular, agents engage in non-communicative goal-oriented action in between encounters, so that the link network (and hence the agents' utility situation) may change while a conversation is unfolding.

- The number of possible link configurations is vast, and agents can only predict possible utilities for a very limited number of hypothetical future layouts.
- There is no notion of commitment – agents choose frames in a self-interested way and may or may not execute the physical actions that result from them. Also, they may undo their effects later on.

LIESON agents consist of a non-social BDI [8] reasoning kernel that projects future link network configurations and prioritises goals according to utility considerations. If these goals involve actions that have to be executed by other agents, the m²InFFrA component starts a framing process which runs until the specific goal has been achieved or no adequate frame can be found. We report on experiments in which agents were equipped with frames with the following six trajectories:

```

request(A, B, X) → accept(B, A, X) → confirm(A, B, X) → do(B, X)
request(A, B, X) → propose(B, A, Y) → accept(A, B, Y) → do(B, Y)
request(A, B, X) → propose – also(B, A, Y) → accept(A, B, Y) →
do(B, X) → do(A, Y)
request(A, B, X) → reject(B, A, X)
request(A, B, X) → propose(B, A, Y) → reject(B, A, Y)
request(A, B, X) → propose – also(B, A, Y) → reject(B, A, Y)

```

The first three frames allow for accepting to perform a requested action X , making a counter-proposal in which Y is suggested instead of X , or using `propose – also` to suggest that B executes X if A agrees to execute Y . The last three frames can be used to explicitly reject a request or proposal. In that, X and Y are link modification actions; each message is available in every state and incurs a cost that is almost negligible compared to the utilities gained or lost through linkage actions (yet high enough to ensure no conversation goes on forever). Also, agents can always send a `stop` action to indicate that they terminate an encounter if they cannot find a suitable frame.

After their termination, encounters are stored in the frame from which they have originated. For example, agent a_1 would store the encounter `request($a_1, a_2, add(a_2, a_1, 2)$) → reject($a_1, a_2, add(a_2, a_1, 2)$)` by adding a substitution $[A/a_1, B/a_2, X/add(a_2, a_1, 2)]$ to the respective frame together with an automatically generated list of conditions that were required for physical action execution.

As state abstraction, we use generalised lists of statements of the form $\{\uparrow|\downarrow\}(\{I, R\}, \{I, R, T\}, \{+, -, ?\})$ representing the physical actions talked about in an encounter. \uparrow and \downarrow stand for a positive or negative link modification (i.e. addition/deletion of a link or an increase/decrease of its rating value), I/R for the initiator and responder of the encounter, T for a third party; $+/-/?$ indicates whether the (learning) agent likes, dislikes or doesn't know the target site of the link modification. For example, if a_1 and a_2 talk about `do($a_1, deleteLink(a_1, a_3)$)` in an encounter initiated by a_1 (while the learning agent a_2 is the responder and likes a_3 's site) this is abstracted to $\downarrow(I, T, +)$. If in the same conversation a_2 suggests to modify his own link toward

a_1 (whom he does not like) from a rating value of 1 to 3, the state (*viz subject*) of the encounter becomes $\{\downarrow(I, T, +), \uparrow(R, I, -)\}$. The intuition behind this state abstraction method is to capture, in a generalised form, the *goal* of the conversation that can currently be realised while at the same time reducing the state space to a reasonable size.

Figure 3 shows a comparison for a system with ten agents with an identical profile of private ratings (preferences) towards other agents (both plots show the performance of the best and the worst agent in the group as well as the average utility over all agents). In the first plot, agents employ BDI reasoning and additionally send requests to others whenever they favour execution of someone else's action according to their BDI queue. These requests are then enqueued by the recipient as if he had "thought of" executing the respective action himself. Thus, it depends on the recipient's goal queue and on his utility considerations whether the request will be honoured or not. As one can see, after a certain amount of time agents do no longer execute any of the actions requested by others, and cannot find any profitable action to execute themselves, either. The system converges to a stable state.

The second plot shows the results of a simulation with the same setup as above but using $m^2\text{InFFrA}$ agents. Again, agents issue requests whenever they identify that someone else could do something useful. After this initial message, the framing procedure takes over. Quite clearly, despite the fact that there is a greater variation in maximal, minimal and average agent utility, the average and the best agent perform significantly better than in the BDI case, while the weakest agent performs just as good as in the BDI case on the average.

While on first glance the performance of $m^2\text{InFFrA}$ agents might not be strikingly different from those using plain BDI, it should be noted that the results establish a lower bound on the performance gained by using $m^2\text{InFFrA}$. In environments with (possibly non-benevolent) peer agents showing non-stationary behaviour, the performance of agents using prespecified communication protocols or assuming a fixed semantics of communication can become arbitrarily bad, while $m^2\text{InFFrA}$ includes the ability to learn what to expect from a given peer in a specific interaction situation. However, additional experiments will be necessary to allow for any quantitative statements.

Yet another interesting interpretation can be drawn from the fact that the average utility of $m^2\text{InFFrA}$ agents lies within the range of the two horizontal lines in the plot. These denote the average utilities for two special linkage configurations: the lower one of the two corresponds to a fully connected linkage graph, in which each agent (honestly) displays the ratings of his out-links (i.e. reveals his true opinions about others). The slightly higher utility shown by the upper line is attained if agents do not lay any links toward agents they dislike. It is an interesting property of the utility function used in *LIESON*, that acting "politically correct" is slightly better than being honest. The fact that agent utilities evolve around these benchmarks indicates that they truly strive to make strategic communication moves and exploit the advantages of concealing certain beliefs.

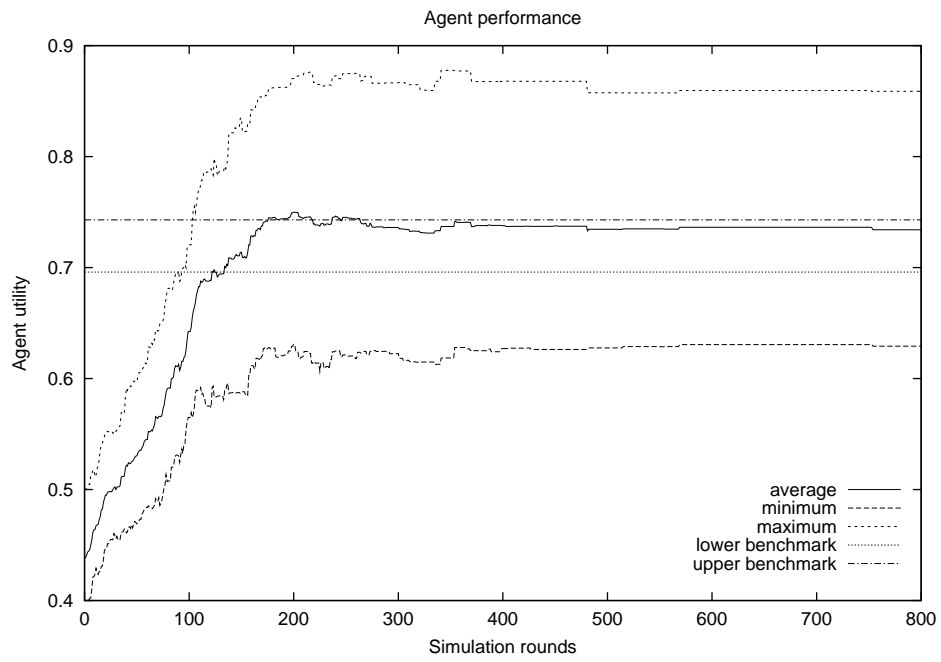
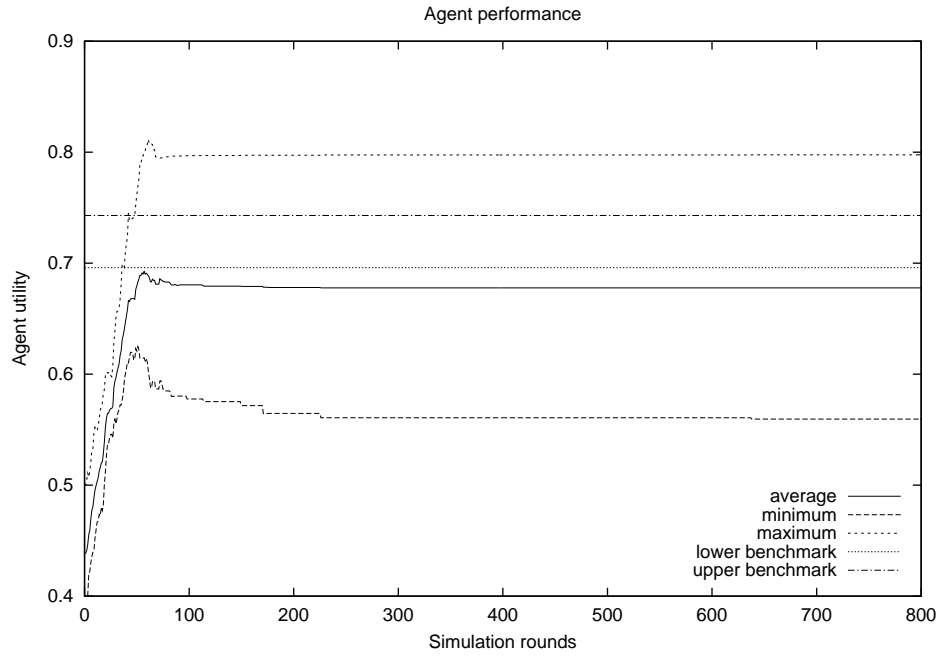


Fig. 3. Performance plots

7 Conclusions

In this paper we have presented an architecture for reasoning about communication patterns within the framework of empirical semantics. From a practical point of view, this contributes to the “agent side” of ACL research, as it offers methods to construct and implement agents that are able to deal with given specifications of rules that govern the communication in a system. Three features are particularly interesting about our approach:

1. The $m^2InFFrA$ frames used in our formal model combine information about the surface structure of communication sequences, logical constraints (frame conditions) and empirical data (counters). This allows for great flexibility with respect to *what* is defined in commonly agreed protocols, semantics, etc. For example, in non-exploitable protocols as those used in mechanism design, it may suffice to just specify admissible message sequences (because it does not matter what the agents think, for example), while other types of interactions such as contracting may require commitment rules, descriptions of agents’ mental states, etc.
2. Agents are capable of exploiting past communication experience. Thus, they are able to “start out” with a predefined set of patterns and to test to which degree their peers comply with them, which offers a major advantage in open systems where adherence to protocols, truthfulness constraints, etc. cannot be guaranteed. In particular, this paves the way for the use of *machine learning* methods (as we have done) that make minimal assumptions about others’ behaviour and simply accumulate communication knowledge as they go.
3. The architecture combines decision-theoretic (probabilistic) reasoning methods with symbolic communication. This constitutes an important contribution to the practical design of communicating agents, but also an initial step to explore the special character of communication as opposed to general action. In [10], for example, we have argued for trading off optimality against predictability in communication so that more reliable communication patterns can occur. Using decision theory as a firm foundation for rational reasoning can help formalising what agents (should) do in communication and whether and in which way this differs from the models used in, e.g. game theory and economics.

A major advantage of our approach is that it allows us to combine the decision-theoretic power of RL models with the knowledge-based aspects of symbolic agent communication, interaction protocols and ACL research in general. It is this aspect that makes rational action and learning possible for high-level agent architectures that employ logical reasoning.

In the future, we intend to look at more complex models of interaction frames with trajectories more expressive than simple sequences. Also, we want to investigate to which degree $m^2InFFrA$ agents are capable of exchanging meta-frame information to reach consensus about which frames to use. Finally, a lot of work needs to be done on automatically transforming ACL and interaction protocol specifications into readily usable $m^2InFFrA$ frames.

References

1. J. L. Austin. *How to do things with Words*. Clarendon Press, 1962.
2. F. Fischer. Frame-Based Learning and Generalisation for Multiagent Communication. Diploma Thesis, Department of Informatics, Technical University of Munich, Munich, Germany, December 2003.
3. FIPA (Foundation for Intelligent Physical Agents). FIPA Communicative Act Library Specification, 2002. Available under URL <http://www.fipa.org/specs/fipa00037/SC00037J.pdf>.
4. FIPA (Foundation for Intelligent Physical Agents). Fipa Contract Net Interaction Protocol Specification, 2002. Available under URL <http://www.fipa.org/specs/fipa00029/SC00029H.pdf>.
5. J. L. Kolodner. *Case-Based Reasoning*. Morgan Kaufmann, San Francisco, 1993.
6. M. Nickles and M. Rovatsos. Communication Systems: A Unified Model of Socially Intelligent Systems. In K. Fischer and M. Florian, editors, *Socionics: Its Contributions to the Scalability of Complex Social Systems*, LNCS. Springer-Verlag, Berlin, Germany, 2004. To appear.
7. M. Nickles, M. Rovatsos, and G. Weiss. Empirical-Rational Semantics of Agent Communication. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-04)*, New York, NY, 2004.
8. 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.
9. M. Rovatsos. *LIESON – User’s Manual and Developer’s Guide*. <http://www7.cs.tum.edu/~rovatsos/lieson/users-manual.pdf>, 2002–2003.
10. M. Rovatsos, M. Nickles, and G. Weiß. Interaction is Meaning: A New Model for Communication in Open Systems. In J. S. Rosenschein, T. Sandholm, M. Wooldridge, and M. Yokoo, editors, *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-03)*, Melbourne, Australia, 2003.
11. M. Rovatsos, G. Weiß, and M. Wolf. An Approach to the Analysis and Design of Multiagent Systems based on Interaction Frames. In Maria Gini, Toru Ishida, Cristiano Castelfranchi, and W. Lewis Johnson, editors, *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-02)*, Bologna, Italy, 2002. ACM Press.
12. M.P. Singh. A social semantics for agent communication languages. In *Proceedings of the IJCAI Workshop on Agent Communication Languages*, 2000.
13. R. S. Sutton, D. Precup, and S. Singh. Between MDPs and semi-MDPs: A Framework for Temporal Abstraction in Reinforcement Learning. *Artificial Intelligence*, 112:181–211, 1999.