Acquiring and Adapting Probabilistic Models of Agent Conversation

Felix Fischer
fischerf@cs.tum.edu
Department of Informatics
Technical University of Munich
85748 Garching, Germany

Michael Rovatsos
mrovatso@inf.ed.ac.uk
School of Informatics
University of Edinburgh
Edinburgh EH8 9LE, UK

Gerhard Weiss
weissg@cs.tum.edu
Department of Informatics
Technical University of Munich
85748 Garching, Germany

ABSTRACT

Communication in multiagent systems (MASs) is usually governed by agent communication languages (ACLs) and communication protocols carrying a clear cut semantics. With an increasing degree of *openness*, however, the need arises for more flexible models of communication that can handle the uncertainty associated with the fact that adherence to a supposedly agreed specification of possible conversations cannot be ensured on the side of other agents.

As one example for such a model, *interaction frames* follow an *empirical semantics* view of communication, where meaning is defined in terms of expected consequences, and allow for a combination of existing expectations with empirical observation of how communication is used in practice.

In this paper, we use methods from the fields of case-based reasoning, inductive logic programming and cluster analysis to devise a formal scheme for the acquisition and adaptation of interaction frames from actual conversations, enabling agents to autonomously (i.e. independent of users and system designers) create and maintain a concise model of the different classes of conversation in a MAS on the basis of an initial set of ACL and protocol specifications. This resembles the first rigorous attempt to solve this problem that is decisive for building truly autonomous agents.

Categories and Subject Descriptors

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

General Terms

Languages, Theory

Keywords

Agent Communication, Evolutionary Semantics, Inductive Generalisation, Cluster Validation

AAMAS'05, July 25-29, 2005, Utrecht, Netherlands

1. INTRODUCTION

Traditional approaches to agent communication, with their roots in speech act theory [1], do not respect the *autonomy* [10] of individual agents in that they suppose effects of communication on agent's mental states [18, 2] or a normative quality of publicly visible commitments [7, 19]. In environments involving some degree of *openness* like, for example, design heterogeneity or dynamically changing populations, such a "normative" attitude is put into question by the fact that adherence to supposedly agreed modes of communication cannot be ensured on the side of other agents. While this problem stems from a fundamental conflict between agent autonomy and the need for cooperation (and communication) with other agents towards a joint goal, there is also a practical side to it that can be phrased in the form of two questions:

1. If strict adherence to communication languages and protocols cannot be taken for granted, how can meaningful and coherent communication be ensured?
2. Observing the course of conversations that take place in a MAS, how can agents effectively organise this kind of knowledge and relate it to existing specifications, so that they can actually benefit from it?

What is obviously required to answer these questions is a *probabilistic* model of agent conversation. Generic "purely" probabilistic models, however, are not very well suited for this task, since symbolic agent communication is not at all "random", and we would rather like to identify patterns and relational properties of communication (in the same way as communication protocols containing variables resemble patterns).

Interaction frames [16] are such a model of agent conversation, capturing both the surface structure of possible messages or message sequences and logical conditions regarding the context of their execution. What distinguishes interaction frames from the methods commonly used for the specification of ACL and protocol semantics is that they allow for an explicit representation of *experience* regarding their practical use. Instead of being interpreted normatively, they are assigned an *empirical communication semantics* [15], where the meaning of an utterance (or sequence thereof) is defined solely in terms of it's expected consequences, as given by past experience with a frame (to say it in terms of speech act theory [1], the meaning of illocutions are defined solely in terms of their expected perlocutions). Currently two different "flavours" of empirical communication semantics exist. While interaction frames view empirical semantics from the perspective of symbolic interactionism, expectation networks [11] take the point of view of social systems theory.

As a matter of fact, empirical semantics derives from actual interactions and hence has to be acquired and adapted dynamically from these using empirical observation. In this paper, we use methods from the fields of case-based reasoning, inductive logic programming and cluster analysis to devise a formal *frame learning scheme* FLeaS for the acquisition and adaptation of interaction frames from the actual conversations conducted in a MAS, enabling agents to autonomously (i.e. independent of users and system designers) create and maintain a concise model of the different classes of conversation on the basis of an initial set of ACL and protocol specifications. This resembles the first rigorous attempt to solve this problem, which is a crucial one for building agents that communicate and act in full appreciation of the autonomy of their respective peer.

The remainder of this paper is structured as follows: In the following section we give a formal definition of interaction frames and their semantics and identify desirable properties of methods for their acquisition and adaptation. In section 3 we develop such a method that views frames as clusters in the space of interactions and aims at maximising the quality of the overall clustering. Section 4 closes with some conclusions and a perspective of possible future work.

2. INTERACTION FRAMES

Before turning to the acquisition and adaptation of frame-based empirical semantics, we quote [4] for a formal definition of a particular instance of the interaction frame data structure. This definition uses a language $\mathcal{M}$ of speech-act [1] like message and action patterns of the form $\texttt{perf}(A,B,X)$ or $\texttt{do}(A,Ac)$. In the case of messages (i.e. exchanged textual signals), `perf` is a performative symbol (e.g. `request`, `inform`), A and B are agent identifiers or agent variables and X is the content of the message taken from a first-order language $\mathcal{L}$. In the case of physical actions (i.e. actions that manipulate the physical environment) with the pseudo-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 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).

This said, frames are formally defined as follows:

Definition 1 (interaction frame) *An* interaction frame *is a tuple* $F = (T, \Theta, C, h_T, h_\Theta)$, *where*

- $T = \langle p_1, p_2, \ldots, p_n\rangle$ *is a sequence of message and action patterns* $p_i \in \mathcal{M}$, *the* trajectory *of the frame,*
- $\Theta = \langle \vartheta_1, \ldots, \vartheta_m\rangle$ *is an ordered list of* variable substitutions,
- $C = \langle c_1, \ldots, c_m\rangle$ *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 conversations, 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 conversations.*

While the trajectory $T(F)$ models the surface structure of message sequences that are admissible according to frame F, each element of $\Theta(F)$ resembles a past binding of the variables in $T(F)$, and the corresponding element of $C(F)$ lists the conditions required for or precipitated by the execution of F in this particular case. $h_T(F)$ finally indicates how often F has been executed completely or just in part, $h_\Theta(F)$ is used to avoid duplicates in $\Theta(F)$ and $C(F)$.

Example 1 *Consider the following frame (for the sake of readability, we write the* h_T *and* h_Θ *values next to the corresponding trajectory steps and substitutions):*

$$
\begin{aligned}
F = \Big\langle & \Big\langle \xrightarrow{6} \texttt{request}(A_1,A_2,X) \xrightarrow{4} \texttt{accept}(A_2,A_1,X) \\
& \quad \xrightarrow{3} \texttt{confirm}(A_1,A_2,X) \xrightarrow{3} \texttt{do}(A_2,X)\Big\rangle, \\
& \Big\langle \{self(A_1), other(A_2), can(A_2, \texttt{do}(A_2,X)\}, \\
& \quad \{owns(A_1, ticket(Y, CPE))\}\Big\rangle, \\
& \Big\langle \xrightarrow{2} \langle [A_1/\texttt{a2}], [A_2/\texttt{a1}], \\
& \qquad [X/\texttt{book}(\texttt{flight}(\texttt{MEX},\texttt{CPE}))]\rangle, \\
& \quad \xrightarrow{1} \langle [A_1/\texttt{a3}], [A_2/\texttt{a1}], \\
& \qquad [X/\texttt{book}(\texttt{hotel}(\texttt{CPE}))]\rangle \Big\rangle \Big\rangle
\end{aligned}
$$

According to F, *a total number of six requests has been issued, four of which have been accepted by the respective peer. Three of these were then followed by a confirmation and execution of the designated action, and substitutions and conditions exist for the cases in which the frame has been "executed" as a whole. While F implicitly states how many conversations have ended prematurely or turned out differently, any further information would have to be stored outside F (i.e. in another frame).*

The semantics of frames has been defined accordingly as a probability distribution over the possible continuations of an interaction that has started with $w \in \mathcal{M}_c$ and is computed by summing up over a set $\mathcal{F}$ of known frames:

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

To reduce the complexity associated with reasoning about a particular interaction, an agent can alternatively select a single frame as a (normative) model of this interaction and restrict reasoning to this frame. For this hierarchical approach to be reasonable as well as successful, however, it is required that the different frames concisely capture the different classes of conversations that can take place. This requirement has to hold as well for frames used by external observers to model, analyse or describe the interactions in a MAS.

What is hence required is a method for the acquisition and adaptation of interaction frames from the actual interactions in a MAS, such that the resulting set of frames corresponds to the different classes of interactions as perceived by the agent or external observer. We propose such a method in the following section.

3. ACQUISITION AND ADAPTATION OF FRAMES

As we have said, the need for its acquisition and adaptation from actual interactions is an inherent property of empirical semantics. Using a set of interaction frames for representation, we have further argued that these frames need to model different classes of interactions within a MAS.

We will now present a method for the adaptation and acquisition of empirical semantics using the previous section's formalisation of interaction frames. For this, we will introduce a metric on the space $\mathcal{M}_c^*$ of finite-length message sequences and then extend it to a metric between frames. This allows us to interpret a frame repository (i.e. a set of known frames) as a (possibly fuzzy) clustering in the "conversation space", and hence to measure the quality of a frame acquisition and adaptation method in terms of the quality of the clustering it produces (referred to as "cluster validity" in [8]).

According to this interpretation, adaptation from a new conversation either introduces a new cluster (viz frame) or it adds to an existing one with or without modifying the trajectory of the respective frame. The different alternatives can be judged heuristically in terms of the corresponding cluster validities, which we will use to devise an algorithm for the adaptation of frame repositories. To perform the necessary frame modifications in any of the above cases, we will also present a generic algorithm for merging two frames into one.

3.1 A distance metric on message sequences

As a basis of our interpretation of interaction frames as clusters, we will start by introducing a distance metric on the set of possible messages and then extend it to finite-length message sequences. Since messages as defined above are essentially first-order objects, we could simply use a general purpose first-order distance like the one proposed in [17]. Instead, we introduce a family of mappings on messages that are parametrised on two functions d_s and D_s and allow us to add a "semantic" flavour in the form of domain-specific knowledge. As we will see, the most basic (and domain-independent) instance of this family is in fact a metric on messages (i.e. it particularly satisfies the triangle inequality), which can easily be extended to message sequences.

Definition 2 *Let $d_s : S \times S \mapsto [0,1]$ a (normalised) metric on the set S of primitive symbols (i.e. function and predicate names) of $\mathcal{M}$ and $\mathcal{L}$. Let $D : (S \times \mathbb{N})^2 \mapsto [0,1]$ a (normalised) mapping on pairs of symbols and their respective parameters. We then define a mapping $d_p : \mathcal{M}_c \times \mathcal{M}_c$ parametrised on d_s and D_s with*

$$d_p(m,n) = \frac{1}{1+\sum_{i,j} D_s(\underline{m},i,\underline{n},j)} \cdot \left(d_s(\underline{m},\underline{n}) + \sum_{i=1}^{|m|}\sum_{j=1}^{|n|} D_s(\underline{m},i,\underline{n},j) \cdot d_p(m_i,n_j)\right)$$

where $\underline{x}$, $|x|$ and x_i denote the operator symbol (or "head"), number of arguments (i.e. arity of the operator symbol) and ith argument of x, respectively.

Hence, the distance of two messages m and n is computed from the distances of both their heads and their arguments, where $D(\underline{m},i,\underline{n},j)$ determines in how far the distance between m and n depends on the distance between the ith argument of m and the jth argument of m.

Example 2 *Consider a genealogy domain with predicates parents$(\cdot,\cdot,\cdot)$, mother$(\cdot,\cdot)$ and child$(\cdot,\cdot)$. The following (partial) definition of d_s induces an intuitive (but otherwise arbitrary) similarity between these concepts:*

	parents	*mother*	*child*
parents	0	1/2	1/2
mother	1/2	0	1/2
child	1/2	1/2	0

We further define $D_s(x,\cdot,y,\cdot)$ to be the identity matrix and assume that $D(x,i,y,j) = D(y,j,x,i)$. The following partial definitions of $D(x,i,y,j)$ establish a connection between the corresponding parameters of different *predicates (e.g., the parameters of parents denote mother, father and child, whereas those of child denote child and parent):*

		parents		
		mother	*father*	*child*
child	*child*	0	0	1
	parent	1	1	0

		parents		
		mother	*father*	*child*
mother	*mother*	1	0	0
	child	0	0	1

		child	
		child	*parent*
mother	*mother*	0	1
	child	1	0

For three individuals Al, Bo, and Zoe and using a trivial definition of d_s for these symbols, d_p for example takes the following values:

m	n	$d_p(m,n)$
child(Zoe,Al)	*child(Zoe,Bo)*	1/3
child(Zoe,Bo)	*parents(Bo,Al,Zoe)*	3/8
parents(Bo,Al,Zoe)	*child(Zoe,Al)*	3/8
mother(Bo,Zoe)	*child(Zoe,Al)*	2/3

So what are the general requirements on d_s and D_s such that d_p is a metric? Formally, to resemble a metric, a mapping δ needs to satisfy the following three conditions:

1. $\delta(m,n) \geq 0$ with equality iff $m = n$
2. $\delta(m,n) = \delta(n,m)$ (symmetry)
3. $\delta(m,o) \leq \delta(m,n) + \delta(n,o)$ (triangle inequality)

If the latter part of the first condition is dropped, the resulting mapping is called a pseudometric. For example, this would be the case for d_s if it was to encode the fact that two symbols denote the same individual. It could be argued, however, that such equalities (and more complex ones like $fatherOf(Bert) = Craig$) should be treated on the knowledge (i.e. semantic) rather than symbol (i.e. syntactic) level. On the other hand, the above example shows that d_s and D_s can indeed be used to treat certain features of the application domain at the symbolic level. d_s and D_s might even be adjusted depending on the way different symbols are used in actual communication (hence learning how different symbols, predicates and functions relate to each other).

Since we require d_s to be a metric (i.e., it particularly satisfies the first condition), d_p trivially satisfies this condition as well. If additionally D_s is symmetric, i.e. $D(m,i,n,j) = D(n,j,m,i)$ for all m,n,i,j, then d_p can easily be shown to satisfy the second condition by means of structural induction (this is the reason why in the above example values of D have only been given for one direction). In [13], measures that satisfy the first two properties are called similarity measures. When used in clustering, however, such similarity measures tend to cause strange behaviour.

A formal treatment of the triangle inequality could again be done by means of structural induction, imposing specific constraints on D_s. This is beyond the scope of this paper, though, and will be omitted for lack of space. Instead, we will henceforth concentrate on the following generic and domain-independent definitions of d_s and D_s (observe that d_s is indeed a metric on S):

$$d_s(x,y) = \begin{cases} 0 & \text{if } x = y \\ 1 & \text{otherwise} \end{cases}$$

$$D_s(x,i,y,j) = \begin{cases} \frac{1}{|x|} & \text{if } x = y \text{ and } i = j \\ 0 & \text{otherwise} \end{cases}$$

This means that every two distinct elements of S have maximum distance, only the distances of corresponding arguments of the same predicate or function are taken into account, and the overall distance is made up in equal parts by the distance of the operator symbols and the average distance of the arguments. We will now show that d_p is indeed a metric for this definition of d_s and D_s.

Proposition 1 *$d = d_p|_{d_s,D_s}$ with d_s and D_s as defined above is a metric on $\mathcal{M}_c$.*

Proof: For the above definitions of d_s and D_s, d_p can be written in simplified form as

$$d(m,n) = \begin{cases} \frac{1}{|m|+1} \sum_{i=1}^{|m|} d(m_i,n_i) & \text{if } \underline{m} = \underline{n} \\ 1 & \text{otherwise.} \end{cases}$$

In order to prove that this resembles a metric, we have to show that the three conditions given above hold for all $m,n,o \in \mathcal{M}_c$. The first two conditions are trivially satisfied if $\underline{m} \neq \underline{n}$ or $\underline{m} = \underline{n}$ and $|m| = |n| = 0$ and can be show to hold for all m and n by means of structural induction.

As for the triangle inequality, we only need to consider the case $m \neq o$, since otherwise $d(m,o) = 0$ and the condition is trivially satisfied. Since $d(x,y) \leq 1$ for all $x,y \in \mathcal{M}_c$ (which is due to the fact that d is normalised and can again be shown by means of induction), so that the inequality holds if either $\underline{m} \neq \underline{n}$ or $\underline{n} \neq \underline{o}$, we can restrict the proof to the case $\underline{m} = \underline{n} = \underline{o}$. Thus, we have

$$d(m,o) = 1/(|m|+1) \sum_{i=1}^{|m|} d(m_i,o_i),$$

$$d(m,n) = 1/(|m|+1) \sum_{i=1}^{|m|} d(m_i,n_i), \text{ and}$$

$$d(n,o) = 1/(|n|+1) \sum_{i=1}^{|n|} d(n_i,o_i),$$

such that the inequality is trivially satisfied for $|m| = 0$ and we can again use structural induction to show that it holds for all $m,n,o \in \mathcal{M}_c$. □

Example 3 *The distance of the two messages*

$$m = \texttt{request}(\texttt{a2},\texttt{a1},\texttt{book}(\texttt{flight}(\texttt{MEX},\texttt{CPE}))) \textit{ and}$$
$$n = \texttt{request}(\texttt{a3},\texttt{a1},\texttt{book}(\texttt{hotel}(\texttt{CPE})))$$

is given by

$$\begin{aligned} d(m,n) = 1/3 \cdot \big(d(\texttt{a2},\texttt{a3}) + d(\texttt{a1},\texttt{a1}) + \\ + d(\texttt{book}(\dots),\texttt{book}(\dots))\big) = \\ = 1/3 \cdot (1 + 0 + 1/2) = 1/2. \end{aligned}$$

To finally obtain a metric $d_* : \mathcal{M}_c^* \times \mathcal{M}_c^* \mapsto [0,1]$ on message sequences, we simply compute the mean pairwise distance of the corresponding elements for sequences of equal length.

Definition 3 (distance between message sequences) *Let $|v|$ and v_i denote the length and ith element of sequence v. We define*

$$d_*(v,w) = \begin{cases} \frac{1}{|v|} \sum_{i=1}^{|v|} d(v_i,w_i) & \textit{if } |v| = |w| \\ 1 & \textit{otherwise.} \end{cases}$$

Proposition 2 *d_* is a metric on the set $\mathcal{M}_c^*$ of finite-length message sequences.*

Proof: Again, we have to show that d_* satisfies the three conditions for being a metric. The first two conditions follow directly from the definition of d_* and proposition 1.

As for the triangle inequality

$$d_*(u,w) \leq d_*(u,v) + d_*(v,w),$$

we again use the fact that d_* is normalised and $d_*(v,w) \leq 1$ for all $v,w \in \mathcal{M}_c^*$. We distinguish three different cases.

If $|u| \neq |v|$ or $|v| \neq |w|$, then the r.h.s. is ≥ 1, while the l.h.s. is ≤ 1, which satisfies the condition.

If $|u| \neq |w|$, the l.h.s. equals 1, but for the r.h.s. to be ≤ 1 we would require that $|u| = |v|$ and $|v| = |w|$, which violates the assumption.

If finally $|u| = |v| = |w|$, the inequality can be written as

$$\sum_{i=1}^{|u|} d(u_i,w_i) \leq \sum_{i=1}^{|u|} d(u_i,v_i) + \sum_{i=1}^{|v|} d(v_i,w_i).$$

Since d is a metric on $\mathcal{M}_c$ and hence

$$d(u_i,w_i) \leq d(u_i,v_i) + d(v_i,w_i)$$

holds for all i, this is satisfied as well. □

3.2 A metric between frames

Having defined a metric d_* on the set of finite-length message sequences, we will now extend this metric (a metric on *points*, so to speak) to a metric on frames by interpreting these as sets of the message sequences they represent (i.e., point *sets*).

[13] proposes a general formalism to define a distance metric between finite sets of points in a metric space. The distance between two sets A and B is computed as the weight of the maximal flow minimal weight flow through a special distance network between the elements of the two sets. Additionally, one can assign weights to the elements of A and B in order to alleviate the difference in cardinalities between the two sets. Interpreting (integer) weights as element counts yields a metric on *multisets*, which is ideally suited to measure the distance between interaction frames in which multiple instances of a particular message sequence have been stored (corresponding to a substitution count larger than one). We will briefly outline the general idea behind this metric and quote the relevant definitions. First recall some basic definitions regarding transport networks.

Definition 4 (integer flow network) *Let (V,E) a loop-free connected finite directed graph with $s,t \in V$ and $|\{x \in V | (x,s) \in E\}| = |\{x \in V | (v,t) \in E\}| = 0$. Let cap a function $cap : E \mapsto \mathbb{N}$. Let w a function $w : E \mapsto \mathbb{N}$. Then $N(V,E,cap,s,t,w)$ is called an* integer flow network.

Definition 5 (flow) *Let $N(V,E,cap,s,t,w)$ an integer flow network. Then a function $f : E \mapsto \mathbb{N}$ is a flow for N iff*

- *$f(e) \leq cap(e)$ for all $e \in E$ and*

- $\sum_{u\in V} f(v,u) = \sum_{u\in V} f(u,v)$ *for all* $v \in V\backslash\{s,t\}$ *(and* $f(v,u) = 0$ *if* $(v,u) \notin E$*).*

For a flow f*,* $val(f) = \sum_{v\in V} f(s,v) = \sum_{v\in V} f(v,t)$ *is called the* value *of* f *and* $w(f) = \sum_{e\in E} w(e)\cdot f(e)$ *is called the* weight *of* f*.*

Definition 6 (maximal flow minimal weight flow) *Let* f *a flow for* $N(V,E,cap,s,t,w)$*.* f *is called a maximal flow for* N *iff for all flows* f' *for* N*,* $val(f') \leq val(f)$*.* f *is called a maximal flow minimal weight flow for* N *iff for all maximal flows* f' *for* N*,* $w(f') \geq w(f)$*.*

The following definition is used to assign integer weights to the elements of a set.

Definition 7 (integer weighting function) *Let* X *a set. Then a function* $W : 2^X \mapsto (X \mapsto \mathbb{N})$ *is an* integer weighting function *on* X*.* $size_W : 2^X \mapsto \mathbb{N}$ *denotes the size of a set under weighting function* W*, i.e.* $size_W(A) = \sum_{a\in A} W(A)(a)$*.*

For given X *and* W*, we further define* $Q_X^W = \max_{A\in 2^X} size_W(A)$ *as the tight upper bound for the size of any subset of* X *under* W*.*

Based on that, a special distance network is defined through the elements of A and B.

Definition 8 (distance network) *Let* X *a set with metric* d *and weighting function* W*,* M *a constant. Then for all finite* $A,B \in 2^X$*, a* distance network $N[X,d,M,W,A,B] = N(V,E,cap,s,t,w)$ *is defined as follows:*

- *V is a set of vertices, given by* $V = A\cup B\cup\{s,t,a_-,b_-\}$ *(such that* $s,t,a_-,b_- \notin A\cup B$*);*
- *E is a set of edges given by* $E = (\{s\}\times(A\cup\{a_-\}))\cup((B\cup\{b_-\})\times\{t\})\cup((A\cup\{a_-\})\times(B\cup\{b_-\}))$*;*
- *cap assigns a capacity to each edge, such that for arbitrary* $a\in A$ *and* $b \in B$*:* $cap(s,a) = W(A)(a)$*,* $cap(b,t) = W(B)(b)$*,* $cap(s,a_-) = Q_X^W - size_W(A)$*,* $cap(b_-,t) = Q_X^W - size_W(B)$*, and* $cap(a,b) = cap(a_-,b) = cap(a,b_-) = cap(a_-,b_-) = \infty$*; and*
- *w assigns a weight to each edge, such that for arbitrary* $a\in A$ *and* $b\in B$*:* $w(a,b) = d(a,b)$*,* $w(s,a) = w(b,t) = w(s,a_-) = w(b_-,t) = w(a_-,b_-) = 0$*, and* $w(a_-,b) = w(a,b_-) = M/2$*.*

Figure 1 shows an example of such a distance network and the capacities and weights assigned to the different edges.

Now, since $cap(s,a_-) \geq 0$, $cap(b_-,t) \geq 0$, and $cap(s,a_-) + \sum_{i=1}^m cap(s,a_i) = cap(b_-,t) + \sum_{i=1}^m cap(b_i,t) = Q_X^W$, the flow of the maximal flow minimal weight flow from s to t of a distance network equals Q_X^W. [13] uses this fact for the following definition of a distance between sets of points in a metric space.

Definition 9 (netflow distance) *Let* X *a set with metric* d *and weighting function* W*,* M *a constant. Then for all* $A,B \in 2^X$*, the* netflow distance *between* A *and* B *in* X*, denoted* $d^N_{X,d,M,W}(A,B)$*, is defined as the weight of the maximal flow minimal weight flow from* s *to* t *in* $N[X,d,M,W,A,B]$*.*

[13] further shows that $d^N_{X,d,M,W}(A,B)$ is a metric on 2^X and can be computed in polynomial time (in $size_W(A)$ and $size_W(B)$ and in the time needed to compute the distance between two points) if all weights are integers. Also, this metric is claimed to be much better suited for applications where there is likely a point with a high distance to any other point than, for example, the Hausdorff metric (which only regards the maximum distance of any point in one set to the closest point in the other set).

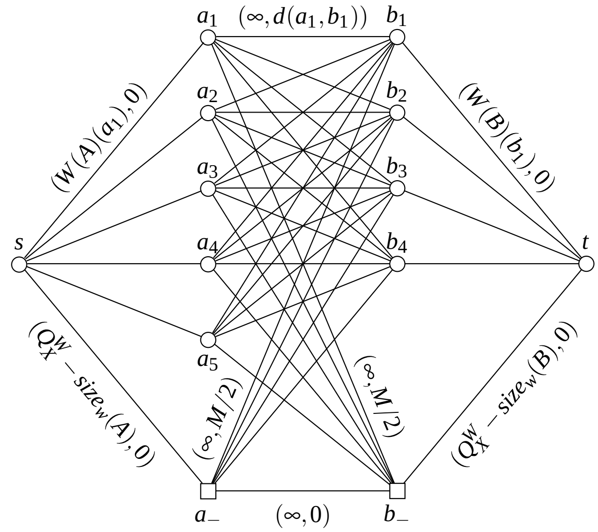

Figure 1: Distance network for two sets $A = \{a_1,\dots,a_5\}$ **and** $B = \{b_1,\dots,b_4\}$ **(adopted from [13]). Edge labels are given in the form** $(cap(a,b),w(a,b))$ **for an edge between** a **and** b**.**

Mapping each frame to the set of messages it represents and weighting each element with the respective substitution count, we directly obtain a metric d_f on frames.

Definition 10 (distance between frames) *Let*

$$m_f(F) = \{m \in \mathcal{M}_c^* | \exists\vartheta \in \Theta(F).\ m = T(F)\vartheta\}$$

the set of message sequences stored in frame F*. Let*

$$W(m_f(F))(m) = h_\Theta(F)[i] \quad iff \quad m = T(F)\Theta(F)[i]$$

a weighting function for elements of $m_f(F)$*. Then, the distance between two frames* F *and* G*, denoted* $d_f(F,G)$*, is defined as the maximal flow minimal weight flow from* s *to* t *in the transport network* $N[\mathcal{M}_c^*, d_*, 1, W, m_f(F), m_f(G)]$*.*

Proposition 3 d_f *is a metric on the set of frames.* $d_f(F,G)$ *can be computed in polynomial time in* $\sum_{i<|\Theta(F)|} h_\Theta(F)[i]$*,* $\sum_{i<|\Theta(G)|} h_\Theta(G)[i]$ *and the time required to compute* d_**.*

Proof: The former follows directly from proposition 2 and from theorem 7 of [13]. The latter follows directly form the definition of W and from theorem 8 of [13]. □

Observe that since d_* is normalised, we can safely set $M = 1$. Further, viewing a particular frame repository $\mathcal{F}$ and assigning a weight of zero to each message sequence not stored in any of the frames in $\mathcal{F}$, we obtain $Q^W_{\mathcal{M}_c^*} = \sum_{F in \mathcal{F}} \sum_{i\leq|\Theta_F|} h_\Theta(F)[i]$.

3.3 Validity of frame modifications

Based on the metrics defined in the previous sections, we can interpret interaction frames as clusters of points in the space of message sequences, which in particular allows us to define the quality of a set of frames as a model for actual interactions in terms of the quality of the corresponding clustering.

[8] refers to this quality as *cluster validity* and defines the validity of a particular cluster as the ratio between its compactness,

i.e. average distance between points within this cluster, and its isolation, i.e. minimum distance to any other cluster. Accordingly, we define the compactness and isolation of a frame using the metrics d_* and d_f on message sequences and frames, respectively.

Definition 11 (frame compactness and isolation) *Let $\mathcal{F}$ a set (repository) of frames, $F \in \mathcal{F}$ a single frame. The* compactness *of F is then defined as the (normalised) average distance between the individual messages stored in it, weighed by their respective occurrence counts:*

$$c(F) = \Big(1 \Big/ \sum_{i=1}^{|\Theta(F)|} \sum_{j=i+1}^{|\Theta(F)|} h_i \cdot h_j \Big) \cdot \sum_{i=1}^{|\Theta(F)|} \sum_{j=i+1}^{|\Theta(F)|} h_i \cdot h_j \cdot d_*\big(T(F)\vartheta_i, T(F)\vartheta_j\big)$$

where $\vartheta_i = \Theta(F)[i]$ and $h_i = h_\Theta[i]$ denote the ith substitution of F and the corresponding count. The isolation *of F in $\mathcal{F}$ is defined as the minimal distance to any other frame in $\mathcal{F}$:*

$$i(F, \mathcal{F}) = \min_{G \in \mathcal{F} \setminus F} d_f(F, G)$$

Since $c(F)$ uses d_* for distances within a single frame F only, there exists a more efficient way of computing it. If we write $w(v,m)$ to denote the *weight* of a variable v in a message pattern m (i.e. the sum of coefficients of $d(v,\cdot)$ in $d_*(m, m\vartheta)$ for some substitution ϑ), then we can precompute $w(v, T(F))$ for any variable v in the trajectory of F, and rewrite $c(F)$ to

$$c(F) \propto \sum_{i=1}^{|\Theta(F)|} \sum_{j=i+1}^{|\Theta(F)|} h_i \cdot h_j \cdot \sum_v w\big(v, T(F)\big) \cdot d_*\big(v\vartheta_i, v\vartheta_j\big)$$

According to definition 11, $c(F)$ is zero for frames with only one distinct substitution, so defining overall validity as the sum or product of individual validities $i(F, \mathcal{F})/c(F)$ is not a good idea. Instead, we define the validity of a frame repository $\mathcal{F}$ as the ratio between average isolation and average compactness for all the frames in $\mathcal{F}$, taking special care of situations where only frames with a single substitution exist.

Definition 12 (frame validity) *Let $\mathcal{F}$ a set (repository) of frames. The* validity *of $\mathcal{F}$ is then defined as*

$$v(\mathcal{F}) = \begin{cases} \frac{\sum_{F \in \mathcal{F}} i(F, \mathcal{F})}{\sum_{F \in \mathcal{F}} c(F)} & \text{if } \exists F \in \mathcal{F} \,.\, |\Theta(F)| > 1 \\ \frac{1}{|\mathcal{F}|} \sum_{F \in \mathcal{F}} i(F, \mathcal{F}) & \text{otherwise} \end{cases}$$

In analogy to cluster analysis we conjecture that the higher the validity $v(\mathcal{F})$ of a frame repository $\mathcal{F}$ built from a particular set of concrete interactions, the better it models the different classes of conversation in a MAS. Facing different alternatives for the incorporation of an interaction that has just been perceived, each of them corresponding to a specific modification of $\mathcal{F}$, we can judge their quality simply by measuring $v(\mathcal{F})$ before and after this modification and hence devise an algorithm that tries to maintain a frame repository with the highest possible validity.

3.4 Frame abstraction and merging

Before we can apply the results of the previous section to an algorithm for the acquisition and adaptation of interaction frames from actual interactions, we will first have to make explicit the actual modifications that can be performed on interaction frames and sets thereof in order to adapt them to newly observed interactions. We do so by providing a general algorithm for merging two interaction frames into one. This algorithm can then be used to simply add a new message to an existing frame (by interpreting the message as a "singular" frame with ground trajectory and only the empty substitution) or to reorganise a whole repository. In order to distinguish these two activities, and according to the point in time they are performed relative to the actual interactions, we might refer to them as online and offline merging.

Starting with frame trajectories and following Occam's Razor, the trajectory of the frame obtained from merging F and G should be the least general message pattern sequence that can be unified with both $T(F)$ and $T(G)$ using standard first-order unification, i.e. the *least general generalisation* (lgg) [12] of the two, denoted $lgg(T(F), T(G))$. The following inductive definition of least general generalisation for message sequences can be turned into a simple algorithm for its computation.

Definition 13 (least general generalisation) *The least general generalisation (lgg) of two terms is given by*

$$lgg(f(s_1,\ldots,s_k), g(t_1,\ldots,t_l)) = \begin{cases} f(lgg(s_1,t_1),\ldots,lgg(s_k,t_k)) & \text{if } f = g \text{ and } k = l \\ x & \text{otherwise,} \end{cases}$$

where x is a new variable (i.e. does not occur in any s_i or t_i) such that $lgg(s,t)$ is unique for any subterms s and t throughout the lgg (i.e. equal terms are replaced with the same variable).

The lgg of two messages is only defined for messages with equal performatives and is given by

$$lgg(p(a,b,x), p(c,d,y)) = p(lgg(a,c), lgg(b,d), lgg(x,y)).$$

The lgg of two message sequences with the same length is given by

$$lgg((m_1,\ldots,m_k),(n_1,\ldots,n_k)) = (lgg(m_1,n_1),\ldots,lgg(m_k,n_k)).$$

As before, it has to be ensured that $lgg(s,t)$ is unique throughout the lgg for any two subterms s and t.

In an algorithm, uniqueness of the lgg is usually achieved by means of a table that holds the lggs computed so far for any pair of arguments.

Along with the lgg, definition 13 also yields two substitutions, namely the most general unifier (mgu) of the lgg with each of its arguments, and we use the abbreviation $\vartheta_m(m,n) = mgu(m, lgg(m,n))$.

Example 4 *The lgg of the two messages m and n of example 3 yields*

$$\begin{aligned} lgg(m,n) &= \mathtt{request}(A, \mathtt{a1}, \mathtt{book}(X)) \\ \vartheta_m(m,n) &= [A/\mathtt{a2}, X/\mathtt{flight}(\mathtt{MEX}, \mathtt{CPE})] \\ \vartheta_m(n,m) &= [A/\mathtt{a3}, X/\mathtt{hotel}(\mathtt{CPE})] \end{aligned}$$

To obtain the substitutions and conditions of the merged frame, the ϑ_m have to be applied to the substitutions and conditions of the respective frame. For this, let F one of the frames to merge, let t denote the trajectory of the resulting frame and c_j and ϑ_j the condition and substitution of the resulting frame that correspond to $C(F)[j]$ and $\Theta(F)[j]$. If the new frame is to hold all the conversations of F, then $t\vartheta_i = T(F)\Theta(F)[i]$ has to hold for $1 \le i \le |\Theta(F)|$.

The definition of ϑ_m implies that $T(F) = t\vartheta_m(T(F),\cdot)$ and thus $t\vartheta_m(T(F),\cdot)\Theta(F)[i] = t\vartheta_i$.

If accordingly ϑ_i is computed as $\vartheta_i = \vartheta_m(T(F),\cdot)\Theta(F)[i]$, however, information might be lost about correlations between multiple conversations originating from the same frame. To retain this kind of information, substitutions should be concatenated rather than applied unless the right side of $\vartheta_m(T(F),\cdot)$ is a variable (which is quite common, as it results from the introduction of a new variable for a variable in the course of computing the lgg). The following definition formalises this concept of selective application of a substitution.

Definition 14 *Let $\vartheta = [v_1/t_1,\dots,v_n/t_n]$ a single variable substitution and $\Theta = \langle s_1,\dots,s_m\rangle$ a list of substitutions. Then, $\vartheta \rtimes \Theta$ denotes the list of substitutions that results from* selectively prepending *ϑ to each element of Θ and is given by*

$$\vartheta \rtimes \Theta = \langle r_1,\dots,r_m\rangle$$

where

$$r_i = [v_1/r_{i1},\dots,v_n/r_{in}]\cdot s_i$$

and

$$r_{ij} = \begin{cases} t_j s_i & \text{if } t_j \text{ is a variable} \\ t_j & \text{otherwise} \end{cases}$$

Example 5 *Recall message pattern $p := lgg(m,n)$ and substitutions $\vartheta_1 := \vartheta_m(m,n)$ and $\vartheta_2 := \vartheta_m(n,m)$ of example 4. Further generalisation to a message pattern $q = \mathtt{request}(B,C,Y)$ (observe that $lgg(p,q) = q$) yields the intermediary result*

$$\vartheta_m(p,q) = [B/A, C/a1, Y/book(X)]$$

and

$$\begin{aligned} &\vartheta_m(p,q) \rtimes \langle\vartheta_1,\vartheta_2\rangle = \\ &= \langle[B/\mathtt{a2}, C/\mathtt{a1}, Y/\mathtt{book}(X), X/\mathtt{flight}(\mathtt{MEX},\mathtt{CPE})], \\ &\quad [B/\mathtt{a3}, C/\mathtt{a1}, Y/\mathtt{book}(X), X/\mathtt{hotel}(\mathtt{CPE})]\rangle \end{aligned}$$

as the list of substitutions corresponding to q.

As for the conditions of the merged frame, $c_i\vartheta_i = C(F)\Theta(F)[i]$ has to hold analogously. Replacing ϑ_i with the above result yields $c_i\vartheta_m\Theta(F)[i] = C(F)\Theta(F)[i]$ and thus $c_i\vartheta_m = C(F)$. Writing ϑ^{-1} for the "inverse" of a substitution ϑ (replacing terms by variables), c_i can hence be defined as $c_i = C(F)\vartheta_m^{-1}$.

This finally leads us to the following definition of a merging operation on frames:

Definition 15 (frame merging) *Let F and G two interaction frames with $|T(F)| = |T(G)|$. Then, the result of* merging *F and G, denoted by $M(F,G)$, is given by*

$$\begin{aligned} &M(F,G) = \\ &\big\langle lgg\big(T(F),T(G)\big), \\ &\ C(F)\vartheta_m\big(T(F),T(G)\big)^{-1}\cdot C(G)\vartheta_m\big(T(G),T(F)\big)^{-1}, \\ &\ \vartheta_m\big(T(F),T(G)\big)\rtimes\Theta(F)\cdot\vartheta_m\big(T(G),T(F)\big)\rtimes\Theta(G), \\ &\ hmax(F,G), \\ &\ h_\Theta(F)\cdot h_\Theta(G)\big\rangle, \end{aligned}$$

where $hmax(F,G) = \langle h_1,h_2,\dots\rangle$ with

$$h_i = \begin{cases} max\big\{h_T(F)[i], h_T(G)[i], \\ \qquad \sum_k h_\Theta(M(F,G))[k]\big\} & \text{if } i = |T(F)| \\ max\big\{h_T(F)[i], h_T(G)[i], h_{i+1}\big\} & \text{if } i < |T(F)|. \end{cases}$$

The rather complex definition of the step counter values for the merged frame stems from the fact that it is impossible to determine the value $h_T(merge(F,G))$ would have taken if $merge(F,G)$ had been in the repository during all the conversations stored in F and G just from the information provided by F and G. On the other hand, it is also impossible to determine which additional conversations would have been stored in $merge(F,G)$ if this had been the case, so it seems fair to approximate h_T based on the following observations: Obviously, $\max(h_T(F),h_T(G))$ is a lower bound for $h_T(merge(F,G))$. In addition to that, the sum of the values of h_Θ is a lower bound for the value of $h_T[|T|]$, since it resembles the exact number of past conversations stored in the frame. Finally, for each i, $h_T[i]$ is a lower bound for $h_T[j]$ with $j < i$. Hence, as we cannot infer any upper bounds from the counter values alone, we simply choose the values of $h_T(merge(F,G))$ such that the bounds are tight. If only online merging is used, this approximation always yields accurate values for h_T.

3.5 An algorithm for learning frames

Based on the formal notion of validity of a set of frames presented in section 3.3, which extends cluster validity to the space of multi-agent conversations, and on the frame merging procedure given in section 3.4, the following simple algorithm computes the best way to incorporate a newly observed message sequence m into a frame repository $\mathcal{F}$:

function *flea*$(\mathcal{F},m)$ **returns** a frame repository
inputs: frame repository $\mathcal{F}$, message sequence m
 /* compute the singular frame F for m ***/**
 $F := \big(m, C_m, \{\}, \langle 1,\dots,1\rangle, \langle 1\rangle\big)$
 /* compute the set $\mathbb{F}$ of alternatives for inclusion of m ***/**
 $\mathbb{F} := \big\{\mathcal{F}\cup\{F\}\big\}\cup\bigcup_{F'\in\mathcal{F}}\big\{\mathcal{F}\setminus F'\cup M(F',F)\big\}$
 /* return the most valid frame repository ***/**
 return $\arg\max_{\mathcal{F}'\in\mathbb{F}} v(\mathcal{F}')$

While the surface structure of a particular message sequence equals the message sequence itself, identification of a set C_m of logical conditions that held during a conversation (according to the observer's world model) and that were *relevant* or *crucial* is clearly a nontrivial task. If frames exist, however, the execution of which was hindered due to reasons of context (especially if pre-specified "protocol" frames are used), these can be used to identify conditions other than those (physically) required for the execution of the individual messages.

Since the above algorithm only considers a single frame at a time for inclusion into the repository, it is unable to detect structures in the space of interactions that develop over time. This corresponds to a more general problem of *order dependence* in incremental unsupervised learning and might in practice result in several frames actually modelling the same class of interactions. This problem can be handled, though, by supplementing the above online merging algorithm with one that periodically checks if two frames in the repository can be merged to increase its overall validity.

4. CONCLUSIONS

In this paper, we have presented a novel approach for building and maintaining a probabilistic model of agent conversations from an initial set of communication primitives and protocols and from the actual conversations that take place in a MAS. Agents in open environments that communicate according to high-level pre-specified conversational patterns can use this approach to augment these patterns with empirical observation of actual conversations, such that they can be attributed an empirical semantics.

A formal scheme FLeaS has been provided which uses a particular instance of the interaction frame data structure for representing the probabilistic model. This allows for an integration of the results presented here with previous work on interaction frames, particularly an architecture for reasoning about communication within the framework of empirical semantics [5, 4] and an application of hierarchical reinforcement learning to the task of learning communication strategies [14, 6]. The basic principles of our approach, however, could also be applied to other, possibly more complex, forms of representation.

The scheme itself uses distance metrics between message sequences and between frames to interpret a set of frames as a clustering in the space of possible conversations and tries to maintain a good quality of this clustering as new conversations are stored. It is thus properly grounded in the theory of clustering and cluster analysis.

Our current work focuses on an experimental exploration of the benefits and limitations of our approach in real-world "communication learning" tasks (some initial results are reported on in [3]). Depending on the perspective from which empirical observations are taken, different applications of interaction frames are possible. As shown in [5, 4], individual agents can put them into relation to their private goals and use them to derive their communicative actions in order to "communicate optimally" towards these. From the perspective of an external observer, on the other hand, interaction frames can be interpreted as a global model of the communication in a MAS and hence used to measure the performance of the MAS or of individual agents w.r.t. communication, to design new communication protocols or to devise open ontologies [9] that dynamically capture concepts and how communications refer to them.

An open issue that has to be dealt with in future work to allow for the creation of interaction frames from scratch is the discovery of conditions that were relevant or crucial for the execution of a specific conversation. While inductive logic programming techniques may again be the appropriate means to attack this problem, a transition to relative least general generalisation (which might be required to handle background knowledge already available for a particular class of conversation) would make this one disproportionately harder to solve.

5. ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their valuable suggestions to improve the quality of the paper.

6. REFERENCES

[1] J. L. Austin. *How to do things with Words*. Clarendon Press, 1962.

[2] P. R. Cohen and H. J. Levesque. Communicative actions for artificial agents. In *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS)*, pages 65–72, 1995.

[3] F. Fischer. Frame-based learning and generalisation for multiagent communication. Diploma Thesis. Department of Informatics, Technical University of Munich, 2003.

[4] F. Fischer and M. Rovatsos. Reasoning about communication: A practical approach based on empirical semantics. In *Proceedings of the 8th International Workshop on Cooperative Information Agents (CIA)*, volume 3191 of *Lecture Notes in Artificial Intelligence*, Erfurt, Germany, 2004. Springer-Verlag.

[5] F. Fischer and M. Rovatsos. An Empirical Semantics Approach to Reasoning About Communication. *Engineering Applications of Artificial Intelligence, Special Issue on Best Papers of CIA 2004*, 18(4), 2005. To appear.

[6] F. Fischer, M. Rovatsos, and G. Weiß. Hierarchical reinforcement learning in communication-mediated multiagent coordination (poster). In *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, New York, NY, 2004.

[7] 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)*, pages 536–542, Bologna, Italy, 2002. ACM Press.

[8] A. K. Jain and R. C. Dubes. *Algorithms for clustering data*. Prentice Hall, Upper Saddle River, NJ, 1988.

[9] M. Nickles, T. Froehner, and G. Weiß. Social annotation of semantically heterogeneous knowledge. In *Proceedings of the 4th International Workshop on Knowledge Markup and Semantic Annotation (SemAnnot)*, 2004.

[10] M. Nickles, M. Rovatsos, and G. Weiß, editors. *Agents and computational autonomy. Potential, risks, and solutions*, volume 2969 (Hot Topics) of *Lecture Notes in Artificial Intelligence*, Berlin, Germany, 2004. Springer-Verlag.

[11] 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)*, New York, NY, 2004.

[12] G. Plotkin. A note on inductive generalization. *Machine Intelligence*, 5:153–163, 1971.

[13] J. Ramon and M. Bruynooghe. A polynomial time computable metric between point sets. *Acta Informatica*, 37(10):765–780, 2001.

[14] M. Rovatsos, F. Fischer, and G. Weiss. Hierarchical Reinforcement Learning for Communicating Agents. In *Proceedings of the Second European Workshop on Multiagent Systems (EUMAS)*, Barcelona, Spain, December 15-17 2004.

[15] M. Rovatsos, M. Nickles, and G. Weiß. Interaction is Meaning: A New Model for Communication in Open Systems. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, Melbourne, Australia, 2003.

[16] M. Rovatsos, G. Weiß, and M. Wolf. An Approach to the Analysis and Design of Multiagent Systems based on Interaction Frames. *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, Bologna, Italy, 2002. ACM Press.

[17] M. Sebag. Distance induction in first order logic. In *Proceedings of the Seventh International Workshop on Inductive Logic Programming*, 1997.

[18] M. P. Singh. A semantics for speech acts. *Annals of Mathematics and Artificial Intelligence*, 8(1–2):47–71, 1993.

[19] M. P. Singh. A social semantics for agent communication languages. In *Proceedings of the IJCAI Workshop on Agent Communication Languages*, 2000.