Coherence Constraints for Agent Interaction

Joris Hulstijn, Frank Dignum, and Mehdi Dastani

ICS, Utrecht University
{jorish,dignum,mehdi}@cs.uu.nl

Abstract. This paper describes the use of coherence constraints as a means to regulate agent interaction. Coherence constraints describe relationships between the content of utterances, and the context. They can be used for example to express that an answer must refer back in a meaningful way to the question that it answers. We also discuss several possible ways in which the enforcement of coherence constraints can be implemented in a multi-agent system. Finally we describe a possible implementation in the 3APL platform, which shows the feasibility of this form of interaction regulation.

1 Introduction

In the field of agent communication languages we observe a trend from specifying small dedicated interaction protocols towards specifications of flexible or open protocols [9, 32, 23], which are more widely usable. Moreover, interaction protocols tend to be viewed more as resources provided by an electronic institution in which the agents interact, than as fixed specifications attached to the agents. One of the reasons for this last trend is that protocols are increasingly studied in conjunction with the social organization that enforces them, and the software infrastructure that enables them [11].

Most agent communication infrastructures are based on the standards developed by FIPA. FIPA has provided a standard for structuring messages [12] and for simple interaction protocols [14]. The backbone of these standards is based on speech act theory [3, 29]. Like other actions, speech acts can be combined into protocols or plans to achieve a goal. The semantics of a speech act is commonly given by the preconditions and intended effect on the mental state of an agent, expressed using modal operators for belief and intention.

The FIPA standardization effort has been a relative success, although it has been criticized heavily, e.g. [26, 31]. Here are two points of criticism.

1. It is impossible to verify the correct usage of a speech act, since for most realistic multi-agent settings the mental state of an agent is inaccessible. Agents may well be lying. This makes it impossible to verify protocols under common assumptions regarding multi-agent systems [31]. What is needed instead is a semantics that is based on *public information* about what agents are committed to, on the basis of what they have said.

2. Policies and protocols are often only defined in terms of speech act types, like request, accept or reject. Protocol definitions are thus mainly concerned with the form of interaction; nothing much is said about the content and function of the messages. Thus agents may conform to the 'letter' of the protocol, while not being coherent. What is needed is a way to extend protocols with so called *coherence constraints* on the content of messages.

The second criticism can be countered by stating that constraints on the coherence of messages can only be given if a certain representation language for the content of the messages is assumed. Because FIPA tries to remain as general as possible it abstracts over the content language and thus the coherence constraints cannot be expressed. Although this is true, we think that in many cases the platform in which the agents interact can in fact put constraints on the content of messages.

In general, interaction behavior is determined by a public protocol that offers a repertoire of messages and rules to define what sequences of messages are well-formed, and the strategy of individual participants to generate messages and accept or reject messages from other participants. A protocol formalizes conventional interaction patterns based on the underlying activity or application. We believe that for many applications, coherence constraints are an essential component of the protocol; they are conventional, just like the message order.

What we mean by coherence is illustrated by the following examples.

(1) propose($s, b, 40$); propose($b, s, 30$); propose($s, b, 35$); propose($b, s, 20$)

(2) cfp(a, b,price_quote(shoes,x)); bid(b, a,price_quote(soles,80))

Example (1) shows a simple type of concessive negotiation. Participants are supposed to make concessions, until they reach agreement. The last bid of buyer b is incoherent, because it is not a concession with respect to the previous bid. Note that what counts as a concession, depends on the content of the message, on the previous messages and on the role of the participant. Example (2) concerns a call for proposals. The example shows that coherence crucially depends on the background knowledge that can be assumed for participants. The bid by b seems incoherent, because it does not match the price quote for shoes that was called for. However, if we suppose that it is commonly known that soles are parts of shoes, and that therefore the price of shoes is partly determined by the price of soles, the bid does count as a coherent, though partial, response.

First, we investigate in this paper the use of coherence constraints as a part of the protocol definition. Because coherence constraints are a very general notion, they can be used to define the order of message types as well as constraints on the content of messages. One might thus use them both in addition to a traditional protocol, and to replace it. In the second case the order of messages can be left more open. Similar to other declarative formulations, like commitments [32] or landmarks, coherence constraints may specify the motivation behind a message sequence. This produces more flexible protocols, that leave more decisions to the strategies of the individual agents. Note however, that the issue of flexible protocols is orthogonal to the use of coherence constraints.

Second, we investigate multi-agent architectures to verify coherence constraints. If the constraints are incorporated in the agents, compliance checking can be done by the agents themselves. The interactions can also be verified by the platform through which the communication takes place. The first option will, in general, be more efficient, but places a heavy burden on the agents. The second option is less efficient, but does not require agents to have additional reasoning capabilities. We will discuss a number of solutions to this trade-off, and show how they can be implemented using the 3APL platform [8], which provides a development platform for the 3APL agent programming language [17].

The rest of the paper is structured as follows. In section 2 we introduce the notion of coherence more fully. In section 3 we show how coherence constraints can be used in combination with existing protocol descriptions. In section 4 we describe a number of alternatives for the implementation of coherence checking in the communication infrastructure, and illustrate it using the 3APL platform.

2 Coherence

We review the notion of coherence as used in linguistics. Intuitively, a discourse (text or dialogue) can be called coherent when its parts 'belong together'. Coherence has been studied in natural language semantics and pragmatics under the header of discourse structure. There are many approaches, e.g. [7, 15, 30, 2].

Aspects of coherence that have to do with form are also called *cohesion* [16]. In natural language, cohesion shows by the use of a consistent vocabulary, a consistent style and parallel syntactic constructions. The use of anaphora and ellipsis to refer to objects mentioned earlier gives the impression of a coherent discourse. Coherence is strongly related to topic structure. A discourse of which the topics of the utterances are related, for example because they are subtopics, makes a more coherent impression than a text with frequent topic shifts.

A common approach to analyze coherence is rhetorical structure theory [22]. The content expressed by an utterance is related to the previous discourse, by a rhetorical relation, such as elaboration, explanation or contrast. Rhetorical relations are also called *coherence relations*. They are typically marked by adverbs like 'because' (explanation), or 'however' (contrast). If no explicit or implicit coherence relation can be found to link an utterance to the context, not even the 'neutral' elaboration relation, the discourse can be said to be incoherent. Coherence also relates to the purpose of utterances. For goal-directed discourse, whenever an utterance contributes to the underlying goal of the discourse, for example to convince the reader or explain something, this will increase coherence. Goal-based notions of coherence occur in models of misunderstanding [1].

A so called discourse context is used to record the contributions of each of the utterances to the over-all meaning of a discourse. A coherence relation determines how the content of an utterance is added to the context. For example, the content of an elaboration can be added straightaway, provided it is logically consistent and does not already follow. But a contrast relation suggests a conflict with a previous utterance, and an explanation induces a causal relation.

Using a discourse context, the global notion of coherence can be reduced to a local notion of coherence with respect to the context. Although this simplification does not hold for discourse in general, it does hold for the multi-agent applications we have in mind.

1. An utterance U is coherent with context C iff a coherence relation $R(U', U)$ can be found that connects U with some part U' of C.
2. In this case, a new context $C' = C + U$ is created which adds the content of U to C in a way that depends on $R(U, U')$.
3. A sequence of utterances $U_1, ..., U_n$ is called coherent, iff each U_k is coherent with context C_{k-1}, which represents the content of $U_1, ..., U_{k-1}$.

This analysis also applies to dialogue. Coherence relations for dialogue are based on the dialogue genre, like negotiation, persuasion or information exchange. For example, Asher and Lascarides [2] analyze question-answer sequences in terms of two coherence relations: IQAP (indirect question answer pair) and Q-ELAB (question elaboration). The IQAP relation expresses answerhood. Two utterance representations U_1 and U_2 stand in relation IQAP when U_1 somehow triggers a question, and U_2 conveys information that counts as an answer to that question. The Q-ELAB relation expresses that asking a question should contribute to the goals of the asker. So, two utterances U_1 and U_2 stand in relation Q-ELAB when U_2 is a question of which the answers will help to achieve part of the apparent goal suggested by U_1.

We believe that this pattern generalizes to other goal-directed types of dialogue. An initiative, such as a question or command, is coherent when the expected response contributes to the apparent goal of the initiator, where the goal may be induced from previous utterances, from the dialogue setting, or from the initiative itself. A response is coherent when it contributes to resolving the problem or goal expressed by the initiative.

3 Protocol Definitions

Our protocol definitions are based on the idea of a dialogue game as expressed by Mann [21] and McBurney and Parsons [24], although we use a simplified terminology. Based on the notions discussed so far we show how to define two simple dialogue games: information exchange and concessive negotiation. These dialogue games are merely meant to illustrate aspects of multi-agent interaction. Although inspired by linguistics, they are bound to over-simplify the complex aspects of coherence found in natural language.

3.1 Dialogue Game Rules

Each dialogue game is defined by five sets of rules.

1. The *initial context* defines the circumstances under which the dialogue game begins, including the expectations and commitments of the dialogue participants upon entering the game.

2. The *dialogue acts* define the repertoire of messages. A dialogue act consist of a dialogue act type (called performative by FIPA), a speaker, an addressee and an expression that represents the content. For example, in dialogues for information exchange we find $\texttt{inform}(i, j, \varphi)$ and $\texttt{question}(i, j, \psi)$. Here i is the speaker, j is the addressee, φ is a proposition and ψ is an expression that represents an issue: the semantic content of a question. In concessive negotiation we find $\texttt{propose}(i, j, \varphi)$ where φ represents the price of a previously selected object.
3. *Combination rules* define under what circumstances a dialogue act is permitted or not, or obligatory or not. These rules incorporate the feasibility and sincerity preconditions of speech act theory. Combination rules implement conventional interaction patterns. For example, questions must be followed by inform acts that count as answers. The combination rules also incorporate coherence constraints that specify, for example, that an answer must be relevant to some question.
4. *Update rules* define the meaning of the content of each dialogue act in terms of the changes to the commitment states of the participants. We write $C_i + \varphi$ to describe the result of updating agent i's commitments with φ, where φ can be a proposition, an issue or an offer.
5. The *end contexts* define the circumstances under which a dialogue may end successfully or unsuccessfully.

Note that the approach is similar to the levels distinguished by Prakken and Sartor [27]. The *logical level* is concerned with the logical relations between the content of messages, such as consistency or relevance, or the attack relation between arguments in a dispute [10]. The *dialectical level* determines how moves affect the commitments of the participants. At this level we put our update rules. The *procedural level* is concerned with the way in which moves can be phrased, and in what order. At this level we find the dialogue acts, and combination rules. Coherence constraints are also placed at this level, but make use of the logical level. Finally, there is a fourth orthogonal *strategic* or *heuristic level*, which is concerned with the motivations of the agents themselves. Only in conjunction with individual strategies does a protocol determine the course of interaction.

Our example dialogue games are interpreted relative to a dialogue context $C = \langle C_1, ..., C_n, H, E\rangle$. A dialogue context consists of the commitments C_i of all participating agents i, along with a history H of all the acts that were uttered, and an environment model E, that contains basic facts about the dialogue situation, such as time, location and information about the roles of the participants.

$$\begin{array}{ccc} \circ & -\texttt{inform}(i, j, \varphi) \rightarrow & \circ \\ | & & | \\ \langle C_i, C_j, H, E\rangle & & \langle C_i + \varphi, C_j + \varphi, H; \texttt{inform}(i, j, \varphi), E\rangle \end{array}$$

Fig. 1. Accepted inform act and corresponding update

Now consider for example an $\texttt{inform}(i, j, \psi)$ act (Figure 1). By the principle of sincerity, the inform commits the speaker to believing φ, so we must update C_i. The actual beliefs of i are irrelevant; what matters are the beliefs that i is committed to uphold on the basis of what was said. In other words: the speaker must appear to be sincere. If we suppose that addressee j accepts the information that φ, j is also committed to believe φ, so in that case we must also update C_j. Else we leave C_j as it is. Acceptance is signalled explicitly by acknowledgement, or implicitly by a coherent continuation. The history H is updated too.

We assume the contents of the messages to be expressed in a first order language L based on which we define a communication language L_C. The definitions are dependent on the specific dialogue game d.

Definition 1 (Language). Let $A = \{i, j, \ldots\}$ be the set of agent names, P_d be a set of dialogue act types for some dialogue game d, with $\alpha \in P_d$ and $\varphi \in L$. Then L_C is defined as follows:

$$\pi ::= \ \alpha(i, j, \varphi) \mid \pi; \pi' \mid \epsilon.$$

Intuitively, $\alpha(i, j, \varphi)$ means that agent i performs a dialogue act of type α towards agent j with semantic content φ. Sequence $\pi; \pi'$ means that first π and then π' should be done. Notation ϵ represents the empty sequence. The language can be further extended to a full protocol definition language, with for example if-then-else constructs, while-loops, or deadlines. For multi-party dialogue, j can be replaced with a set of addressees.

3.2 Information Exchange

Current theories of information exchange in natural language semantics assume that a dialogue context involves the *issues under discussion* [19, 20], in addition to the factual information being exchanged. Therefore, for each agent we record the issues it is apparently interested in, as well as the information it is committed to uphold, based on what has been said.

Definition 2 (Dialogue Context - Information Exchange).
For each agent $i \in A$, let the *commitment state* be a tuple $C_i = \langle S_i, I_i \rangle$, where S_i is a set of closed formulae representing the information to which i is committed, and I_i is a set of formulae representing the issues to which i is committed.
Let *history* H be a sequence of dialogue acts, represented by a formula from L_C.
Let *environment* E be a set of ground formulae (no variables), representing basic facts about the dialogue situation. We can demand $E \subseteq S_i$ for all agents i, in which case the environment is part of the common ground of the agents, but this assumption is not necessary.
Now a *dialogue state* between agents $A = \{1, ..., n\}$, denoted as C, is an n-tuple consisting of the commitment states of the individual agents, the history, and the environment: $C = \langle C_1, \ldots, C_n, H, E \rangle$.

The following update definitions are rather straightforward. More elaborate definitions result, if we take 'grounding' into account [6]. In that case, an update by the addressee would only take place after an utterance has been accepted. Acceptance is then either signalled explicitly by an acknowledgement, or implicitly by a coherent continuation. For now we restrict ourselves to $P_{\texttt{inf\_ex}} = \{\texttt{inform}, \texttt{question}\}$. Since π can be a complex expression, we define $C + \pi$ in a compositional way.

Definition 3 (Update - Information Exchange). Let $C = \langle C_1, ..., C_n, H, E\rangle$, $i, j \in A$, $\varphi \in L$ and $\pi, \pi_1, \pi_2 \in L_C$. Then an update $C + \pi$ is defined as follows:

$$\begin{aligned}
C + \texttt{inform}(i, j, \varphi) &= \langle\langle S_i \cup \{\varphi\}, I_i\rangle, \langle S_j \cup \{\varphi\}, I_j\rangle, H; \texttt{inform}(i, j, \varphi), E\rangle \\
C + \texttt{question}(i, j, \psi) &= \langle\langle S_i, I_i \cup \{\psi\}\rangle, \langle S_j, I_j \cup \{\psi\}\rangle, H; \texttt{question}(i, j, \varphi), E\rangle \\
C + \pi_1; \pi_2 &= (C + \pi_1) + \pi_2 \\
C + \epsilon &= C
\end{aligned}$$

Relative to the issues recorded in the commitment states of an agent, a restricted notion of relevance can be defined [19]. A proposition is relevant when it partly resolves one of the issues in the context. Issues are represented here by first order formulas, possibly containing free variables, similar to Prolog queries. We say a formula φ *resolves* an issue ψ relative to a set of formulas S, in case there exists an assignment θ of variables to constants, such that $S, \varphi \models \psi\theta$. An issue ψ itself is relevant$_I$, whenever its resolution will resolve some other, more embedded, issue χ. Other definitions are quite possible. For example, we may stipulate that all questions are relevant. Note that we can not define relevance of questions by referring to the goals of the asker, since we only have access to what was said.

In our restricted version of information exchange, we demand that a contribution to an information exchange must also be informative and consistent. Consistency is the usual notion. A proposition is considered informative, when it is not already derivable from previous commitments; likewise for an issue. A fourth restriction, that information should not be over-informative, is left out for simplicity. Although these constraints roughly correspond to the maximes of Grice, they do not necessarily assume that agents are cooperative; they just require that agents behave as if they are cooperative.

In general we use notation $\texttt{coherent}(d, C, \pi)$ to denote that dialogue π of type d is coherent in dialogue context C. Again, it is defined compositionally.

Definition 4 (Coherence - Information Exchange). Let C be a dialogue context, $\pi_1, \pi_2 \in L_C$ and $i, j \in A$ and $\varphi \in L$. Then, $\texttt{coherent}(\texttt{inf\_ex}, C, \pi)$ is defined as follows:

$$\begin{aligned}
&\texttt{coherent}(\texttt{inf_ex},\ C,\ \texttt{inform}(i, j, \varphi)) \Leftrightarrow \\
&\quad \texttt{consistent}(C_i,\ \varphi)\ \& && \text{(consistent for speaker)} \\
&\quad \texttt{relevant}(C_j,\ \varphi)\ \& && \text{(relevant for addressee)} \\
&\quad \texttt{informative}(C_i,\ \varphi), \& && \text{(informative for both)} \\
&\quad \texttt{informative}(C_j,\ \varphi), \\
&\texttt{coherent}(\texttt{inf_ex},\ C,\ \texttt{question}(i, j, \psi)) \Leftrightarrow \\
&\quad \texttt{relevant}_I(C_i,\ \psi)\ \& && \text{(relevant for speaker)} \\
&\quad \texttt{informative}_I(C_i,\ \psi), \& && \text{(informative for both)} \\
&\quad \texttt{informative}_I(C_j,\ \psi),
\end{aligned}$$

where for $x \in \{i, j\}$

$\texttt{consistent}(\langle S_x, I_x\rangle,\ \varphi)$ iff $S_x, \varphi \not\models \bot$

$\texttt{relevant}(\langle S_x, I_x\rangle,\ \varphi)$ iff there is a $\psi \in I_x$ and $S_x, \varphi \models \psi\theta$, for some θ

$\texttt{relevant}_I(\langle S_x, I_x\rangle,\ \psi)$ iff there is a $\chi \in I_x$ and $S_x, \psi \models \chi$ i.e. for all θ

$\texttt{informative}(\langle S_x, I_x\rangle,\ \varphi)$ iff $S_x \not\models \varphi$

$\texttt{informative}_I(\langle S_x, I_x\rangle,\ \psi)$ iff $I_x \not\models \psi$

$$\begin{aligned}\texttt{coherent}(\texttt{inf\_ex}, C, \pi_1; \pi_2) &\Leftrightarrow \texttt{coherent}(\texttt{inf\_ex}, C, \pi_1)\ \& \\ &\quad\ \texttt{coherent}(\texttt{inf\_ex}, C + \pi_1, \pi_2) \\ \texttt{coherent}(\texttt{inf\_ex}, C, \epsilon) &\Leftrightarrow \top\end{aligned}$$

An information exchange presupposes a so called *information potential*: there is some issue that the 'novice' is interested in, and that it expects to be known by the 'expert'. Unfortunately, we can only use public commitments; not the real interests of the agents, so we cannot express this constraint in the initial context. Therefore we have the following rather weak definition of an initial context.

Definition 5 (Initial Context - Information Exchange). Let $A = \{i, j\}$ be the participants. Let ϵ be the empty formula and E_0 the initial environment, then $C^0 = \langle\langle\emptyset, \emptyset\rangle, \langle\emptyset, \emptyset\rangle, E_0, \epsilon\rangle$ is the initial state of an information exchange.

Termination of an information exchange means that all issues are resolved.

Definition 6 (End State - Information Exchange). Let $A = \{i, j\}$ be the participants, then $C^+ = \langle\langle S_i, I_i\rangle, \langle S_j, I_j\rangle, E, H\rangle$ is a successful end state of an information exchange, iff $C^0 + H = C^+$, and $\texttt{coherent}(\texttt{inf\_ex}, C^0, H)$ and for all $\psi \in I_i \cup I_j$, $S_j \models \psi\theta$ or $S_i \models \psi\theta$, for some assignment θ.

3.3 Concessive Negotiation

A concessive negotiation is a rather restricted type of dialogue. It is inspired by the *monotonic concession protocol*, which has been studied extensively, see for example [28]. We assume that participants are making bids on some object, of which it has previously been decided that one agent, the buyer, wants to buy it and that the other agent, the seller, wants to sell it. We let $P_{\texttt{co\_neg}} = \{\texttt{propose}\}$ and we require that the content of all messages is of the form $\texttt{price}(x)$ where x is a positive amount.

For the dialogue contexts, we just re-use definition 2, although we do not use the issues. Note that we could have left out the commitment states altogether; all work can be done by the history. That would also remove the need for update rules. But as soon as we allow versions of the protocol with exceptions, or decommitment, we must record commitments separate from history.

Definition 7 (Dialogue Context - Concessive Negotiation).
Identical to definition 2.

The main coherence relation is that of a bid being a concession. In this restricted version, a concession means a lower price for the seller, and a higher price for the buyer. Obviously more interesting definitions of a concession exist for less well-defined domains, such as tasks in a household:

(3) A. Can you put the garbage out?
B. Only if you will do the dishes.

In both cases, a concession means that a proposal is made which is less preferred than the previously made proposals, where the preference order is partly related to the role (buyer, seller) and partly to personal preferences. Note that the notion of consistency does not apply, as each price is strictly speaking inconsistent with previously mentioned prices. Informativeness follows from concession, which is a much stronger notion. Relevance would also follow from concession, if each agent were interested in the issue $\texttt{price}(x)$, i.e. what price is going to be paid.

Definition 8 (Coherence - Concessive Negotiation). Let $A = \{b, s\}$ with x, y variables ranging over A and $C = \langle C_b, C_s, H, E\rangle$, such that $E \models \texttt{buyer}(b) \wedge \texttt{seller}(s)$, then define:

$$\texttt{coherent}(\texttt{co\_neg},\ C,\ \texttt{propose}(x, y, \texttt{price}(u))) \Leftrightarrow \texttt{concession}(C, x, \texttt{price}(u)).$$

$$\texttt{coherent}(\texttt{co\_neg}, C, \pi_1; \pi_2) \Leftrightarrow \texttt{coherent}(\texttt{co\_neg}, C, \pi_1)\ \&\ \texttt{coherent}(\texttt{co\_neg}, C + \pi_1, \pi_2)$$

$$\texttt{coherent}(\texttt{co\_neg}, C, \epsilon) \Leftrightarrow \top$$

where

$$\texttt{concession}(\langle C_x, C_y, \pi; \texttt{propose}(y, x, \texttt{price}(v)), E\rangle, x, \texttt{price}(u)) \Leftarrow E \models \texttt{buyer}(x) \wedge \texttt{seller}(y)\ \&\ u \leq v\ \&\ C_x \models \texttt{price}(u')\ \&\ u' < u.$$

$$\texttt{concession}(\langle C_x, C_y, \pi; \texttt{propose}(x, y, \texttt{price}(u)), E\rangle, y, \texttt{price}(v)) \Leftarrow E \models \texttt{buyer}(x) \wedge \texttt{seller}(y)\ \&\ u \leq v\ \&\ C_y \models \texttt{price}(v')\ \&\ v < v'.$$

The update rules are again rather straightforward. Each bid, replaces the previous ones. Updates do not need to be monotonic.

Definition 9 (Update - Concessive Negotiation). Let $A = \{b, s\}$, with x, y variables ranging over A, and $C = \langle C_b, C_s, H, E\rangle$, then define

$$C + \texttt{propose}(x, y, \texttt{price}(v)) = \langle\langle (S_x \setminus \{\texttt{price}(u)\}) \cup \{\texttt{price}(v)\}, I_x\rangle, H; \texttt{propose}(x, y, \texttt{price}(v)), E\rangle$$

$$C + \pi_1; \pi_2 = (C + \pi_1) + \pi_2$$

$$C + \epsilon = C$$

In the initial context, no agent has made a bid yet.

Definition 10 (Initial Context - Concessive Negotiation). Let $A = \{b, s\}$ and $E_0 \models \texttt{buyer}(b) \wedge \texttt{seller}(s)$, then $C^0 = \langle\langle\emptyset, \emptyset\rangle, \langle\emptyset, \emptyset\rangle, \epsilon, E_0\rangle$ is an initial state of a concessive negotiation.

At the end contexts, an agreement about the price must have been reached.

Definition 11 (End Context - Concessive Negotiation).
$C^+ = \langle\langle\{\texttt{price}(u)\}, \emptyset\rangle, \langle\{\texttt{price}(v)\}, \emptyset\rangle, H, E\rangle$ is a successful end state of a concessive negotiation, iff $u = v$, $C^0 + H = C^+$, and $\texttt{coherent}(\texttt{co\_neg}, C^0, H)$.

3.4 Dispute and Commissive Dialogues

The approach can be extended to dispute. In that case, the semantic *attack* relation [10], is the main ingredient used in coherence constraints.

Although we believe the approach would also work for dialogues that produce commitments, i.e. accepted requests, accepted proposals, and closed negotiations and deliberation in general, it is currently an open problem how to formalize the *fulfillment* relation for commitments, that would be the main candidate as a semantic relation underlying coherence for such dialogues. Under what conditions can we say a commitment has been fulfilled, or a promise has been kept? This will be left for further research.

4 Coherence in Agent Interaction

Several communication infrastructures for multi-agent systems have become available [8, 4, 18], some of which conform to the FIPA recommendations [13]. Many of these systems provide a message passing mechanism that ensures messages are delivered, provided they are addressed using the right identifier. Alternatively, we may assume a shared data space with coordination mechanisms to regulate communication [5, 25].

4.1 Approaches to Coherence Validation

Given an agent communication platform, how can violations of coherence constraints be detected and reported? We discuss some possible architectures.

1. By a central *director* agent, that controls the interaction. An example is the role of an auctioneer in an auction. The auctioneer will only recognize well-formed and coherent bids. Another example is the role of a chairman in a meeting, that assigns turns, manages speaking time and sets the topic for each speaker. In case of violations, a director has the authority to sanction the violator, for example by banishment from the group.
2. By connecting each agent to an individual *governor* agent, as in ISLANDER [11]. The governor provides an API-like interface that allows an external agent to interact with other agents in the system and with the environment. The interface works as a filter: potential violations are simply blocked.
3. By getting each *participant* individually to play the role of a governor, and decide whether or not to accept messages. Incoherent or non well-formed messages are either ignored or explicitly rejected. One might say that in this case the protocol has been reduced to the message layer, and that all the work is now done by the strategy of the individual agents. In other words: rejecting a message has become a strategic action. Note that this solution puts a heavy requirement on the knowledge and reasoning capabilities of an agent. Each agent should know the protocol, be able to detect violations, and decide whether it is more beneficial to reject a message, or leave incoherencies unnoticed and maintain good relations with the speaker.

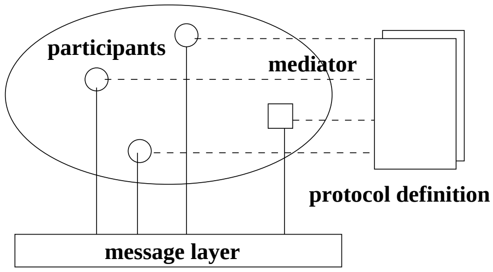

Fig. 2. Incoherence detection through a mediator on the platform

4. By means of a hybrid approach, in which a *mediator* agent monitors the interactions to detect and report violations as they occur. Such a mediator agent can have the same knowledge and reasoning capacities as other agents in the environment; it needs no special privileges except for access to the messages. The protocol can be made accessible on the platform, so all agents know how to behave. The advantage of having a mediator, is that the agents do not require a complex violation detection mechanism. With respect to violations, several different sanction policies can be devised. For example, the mediator can suffice with sending all affected agents a violation report. Agents can then decide a sanction for themselves, e.g., to abandon the interaction, put the violator on a black list, or continue after all. Alternatively, the mediator might request the agent management system to have the agent removed from the platform, or be banned for future occasions.

In the remainder of this section we explore the last option. The general set up of a mediator in a platform, is depicted in figure 2.

But before we can explain the implementation of a mediator, we first need to be more specific about the agent platform. We take the 3APL agent platform as an example; the approach is also possible in other platforms.

4.2 Agent Platform

Figure 3 shows an overview of the 3APL agent platform, which follows FIPA recommendations [13] and is therefore rather general. Such a platform provides communication and coordination facilities, access to knowledge sources, and a way of mediating between different agents. Many of these facilities are accessed through the *agent management system* (AMS) which also controls entry and exit of agents on the platform.

In the case of 3APL, the platform can also be used to assist programmers during development. A graphical user interface (GUI) enables the programmer to load agents from a library, implement and execute them, and observe their behavior.

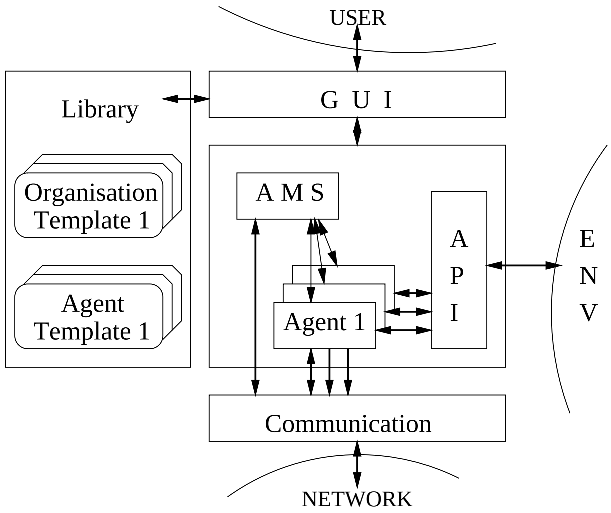

Fig. 3. Agent Platform

Individual Agents Figure 4 shows an example of a 3APL agent program. 3APL closely follows the BDI paradigm. It contains a *goal base*, and a *belief base*. These constructs together represent an agent's mental state throughout the execution of the program. The belief base is currently implemented using a Prolog interpreter. It allows one to reason, and match queries to beliefs. The agents *capabilities* are of the form {`Pre`}`Capability`{`Post`}, where both `Pre` and `Post` are formulas representing beliefs. A program transforms the mental state by means of a number of *practical reasoning rules*. These rules generally are of the form `Goal` ← `Guard | Body`, where the guard is a test on the belief base. The body contains goals, capabilities or combinations of these using sequence, while, if-then-else or a test. A test succeeds if a Prolog query to the belief base succeeds. By sending and receiving messages, facts of the form `sent` or `received` are added to the belief base.

Shared knowledge file For some applications, an agent needs general background knowledge or skills. Therefore an external Prolog file can be loaded. The agent behaves as if the clauses in the Prolog file are part of its personal belief base. This facility is used to distribute the protocol and make the coherence relations accessible to all agents, including the mediator.

Communication Management The 3APL agent platform provides communication by means of message passing. A message will be delivered by the message layer, provided the agent management system knows the identifier of the agent being addressed. The messages themselves have the structure of speech acts, with a sender, receiver and a content, which is compliant with the FIPA standards for agent communication [12].

```
PROGRAM "mediator"

LOAD "H:\protocol.pl" \\ classify\2, update\2, coherent\3, sanction\3

CAPABILITIES:
{session(SID, DialGame, As, C)} Store(C1)
   {NOT session(SID, DialGame, As, C), session(SID, DialGame, As, C1)},
{} StoreViolation(Message,Session) { violation(Message, Session) }
\\ ...

BELIEFBASE:
\\ example of a session fact, with (C_harry, C_sally, Hist, Env)
\\ session(023,info_exch,[harry,sally],
\\  c(c_harry([],[dep_time(kl108,?Time)]),c_sally([],[]),
\\  [m(023,harry,[sally],question,dep_time(kl108,?Time))],[ams, 9:34]))

GOALBASE:
monitor_detect_report(),sanction()

RULEBASE:
monitor_detect_report() <- received(Id,Sender,Addrs,Perf,Phi) |
    BEGIN                                                 // monitor
        classify( m(Id,Sender,Addrs,Perf,Phi), DialGames)?;
        member(DialGame, DialGames)?;
        session(SID, DialGame, As, C)?;
        IF  member(Sender,As) AND subset(Addrs, As) AND   // detect
            coherent(DialGame, C, m(Id,Sender,Addrs,Perf, Phi))
        THEN
            BEGIN
                update(DialGame, m(Id,Sender,Addrs,Perf,Phi), C, C1)?;
                Store( C1 );
            END
        ELSE
            BEGIN                                        // report
                Send(99, [Sender|Addrs], announce,
                     violation( m(Id,Sender,Addrs,Perf,Phi),
                                session(SID, DialGame, As, C)));
                StoreViolation( m(Id,Sender,Addrs,Perf,Phi),
                                session(SID, DialGame, As, C))
            END
    END,

sanction() <-  violation(Message,Session) |
    BEGIN
        sanction_policy(Message, Session, Sanction)?;
        execute(Sanction)                        // to be filled in
    END.
```

Fig. 4. 3APL template for a mediator agent

4.3 Mediator

A 3APL template for a mediator agent is shown in figure 4. This version applies a simple message driven monitor-detect-report loop. For each message that is sent on the platform, a copy is sent to the mediator. The mediator maintains a list of interaction sessions, containing a representation of the context with a message history, the particular dialogue game, the participating agents and their roles. The session list is matched against the incoming message. Based on a classification procedure, the mediator decides which protocol may be applicable. The classification is based on the set of dialogue acts that may occur in a certain dialogue game, but also takes the current participants into account. Because one dialogue act type can occur in several protocols, there may be a list of several candidate sessions. When possible, the message identifier and the session identifiers are used to speed up this matching process.

If the mediator can match the incoming message to one of the candidate sessions such that it makes a coherent contribution, the message is stored as part of that session. If no such match can be made, the message is deemed incoherent. In that case the mediator sends a so called *violation report* to all participants involved: the violator and the addressees of the message. The violation report contains references to the message and the session. Violations are also stored for later reference. Based on a sanctioning policy, further action may be initiated. A slot for such action is indicated in the template.

5 Conclusion

We have proposed a way to formulate coherence constraints on the content of messages in agent communication protocols, in addition to the usual syntactic constraints. The notion is inspired by linguistic research. Protocols of two kinds of dialogue types, information exchange and concessive negotiation, can be reformulated in terms of coherence constraints.

We have discussed several ways in which coherence checks can be implemented in an agent communication infrastructure: using a central director, a governor for each individual agent, by the agents themselves or by means of a mediation service provided by the platform. An architecture using a mediator, is presented in detail. The approach is illustrated using the the 3APL communication infrastructure.

The design is currently being implemented in the 3APL platform. Experiments for empirical evaluation of the approach are envisioned.

With respect to the two criticisms raised against the FIPA approach in the introduction, we may conclude the following.

1. A mediator agent with similar capabilities and privileges as other agents, turns the semantics of the speech acts and protocol into a public affair, and removes the necessity of accessing the mental states of agents.
2. Coherence constraints make it possible to specify protocols in terms of the content and semantics of messages, and not just their syntax.

References

1. Ardissono, L., Boella, G., and Damiano, R. A plan-based model of misunderstandings in cooperative dialogue. *Human Computer Studies*, 48:649–679, 1998.
2. Asher, N. and Lascarides, A. *Logics of Conversation.* Cambridge University Press, 2003.
3. Austin, J. *How to do things with words.* Harvard University Press, Cambridge Mass., 1962.
4. Bellifemine, F., Poggi, A., and Rimassa, G. JADE: a FIPA2000 compliant agent development environment. In *Proceedings of the fifth international conference on autonomous agents (Agents'01)*, pages 216–217. ACM, 2001.
5. Carriero, N. and Gelernter, D. Linda in context. *Communications of the ACM*, 32:444–458, 1989.
6. Clark, H. and Brennan, S. A. Grounding in communication. In Resnick, L., Levine, J., and Teasley, S., editors, *Perspecives on socially shared cognition*, pages 127–149. APA BOOKs, Waskington, D.C., 1991.
7. Craig, R. T. and Tracy, K., editors. *Conversational Coherence: Form, Structure and Strategy.* Sage Publications, Beverly Hills, 1983.
8. Dastani, M., Hulstijn, J., Dignum, , and F. Meyer, J.-J. C. Issues in multiagent system development. In *Third International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'04)*, to appear.
9. Dastani, M., Hulstijn, J., and van der Torre, L. Dialogue games and negotiation. In *Proceedings of the Fifth International Conference on Autonomous Agents (AA'2001)*, pages 180–181. ACM Press, 2001.
10. Dung, P. M. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.
11. Esteva, M., de la Cruz, D., and Sierra, C. ISLANDER: an electronic institutions editor. In *First Interantional Joint Conference on Autonoumous Agents and Multiagent Systems (AAMAS'02)*, pages 1045 – 1052. ACM Press, 2002.
12. FIPA, . ACL message structure specification. Technical Report XC00061, Foundation for Intelligent Physical Agents, 2001.
13. FIPA, . Agent management specification. Sc0023j, Foundation for Intelligent Physical Agents, 2002.
14. FIPA, . Interaction protocol specifications. Technical Report SC00026 – SC00036, Foundation for Intelligent Physical Agents, 2002.
15. Grosz, B. J. and Sidner, C. L. Attentions, intentions and the structure of discourse. *Computational Linguistics*, 12:175–204, 1986.
16. Halliday, M. and Hasan, R. *Cohesion in English.* Longman, New York, 1976.
17. Hindriks, K., de Boer, F., van der Hoek, W., and Meyer, J.-J. Agent programming in 3APL. *Autonomous Agents and Multi-Agent Systems*, 2(4):357–401, 1999.
18. Howden, N., Ronnquist, R., Hodgson, A., and Lucas, A. Jack summary of an agent infrastructure. In *5th International Conference on Autonomous Agents.* 2001.
19. Hulstijn, J. *Dialogue Models for Inquiry and Transaction.* PhD thesis, University of Twente, Enschede, 2000.
20. Larsson, S. and Traum, D. Information state and dialogue management in the TRINDI dialogue move engine toolkit. *Natural Language Engineering*, 6(3-4):323 – 340, 2000.
21. Mann, W. C. Dialogue games: Conventions of human interaction. *Argumentation*, 2:511–532, 1988.

22. Mann, W. C. and Thompson, S. A. Rhetorical structure theory: Towards a functional theory of text organization. *Text*, 8(3):243–281, 1988.
23. Mazouzi, H., Seghrouchni, A. E. F., and Haddad, S. Open protocol design for complex interactions in multi-agent systems. In *Proceedings of the 1st international joint conference on Autonomous agents and multiagent systems (AAMAS'02)*, pages 517 – 526. ACM Press, 2002.
24. McBurney, P. and Parsons, S. Games that agents play: A formal framework for dialogues between autonomous agents. *Journal of Logic, Language and Information*, 11(3):315–334, 2002.
25. Omicini, A. Towards a notion of agent coordination context. In Marinescu, D. and Lee, C., editors, *Process Coordination and Ubiquitous Computing*, page 187200. CRC Press, 2002.
26. Pitt, J. and Mamdani, A. Some remarks on the semantics of FIPA's agent communication language. *Autonomous Agents and Multi-Agent Systems*, 2(4):333–356, 1999.
27. Prakken, H. and Sartor, G. Modelling reasoning with precedents in a formal dialogue game. *Artificial Intelligence and Law*, 6:231–287, 1998.
28. Rosenschein, J. S. and Zlotkin, G. *Rules of Encounter: designing conventions for automated negotiation among computers*. MIT Press, Cambridge Mass., 1994.
29. Searle, J. R. *Speech acts: an Essay in the Philosophy of Language*. Cambridge University Press, Cambridge, 1969.
30. Webber, B. Computational perspectives on discourse and dialogue. In Schiffrin, D., Tannen, D., and Hamilton, H., editors, *The Handbook of Discourse Analysis*. Blackwell Publishers Ltd., 2001.
31. Wooldridge, M. J. Semantic issues in the verification of agent communication languages. *Journal of Autonomous Agents and Multi-Agent Systems*, 3(1):9–31, 2000.
32. Yolum, P. and Singh, M. P. Flexible protocol specification and execution: Applying event calculus planning using commitments. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS'02)*, pages 527–534. ACM Press, 2002.