Request for Action Reconsidered as a Dialogue Game Based on Commitments

Brahim Chaib-draa[1], Marc-André Labrie[1], and Nicholas Maudet[2]

[1] Dépt. Informatique et Génie Logiciel,
Université Laval, Ste-Foy, PQ, Canada
{chaib,labrie}@iad.ift.ulaval.ca
[2] IRIT, Université Paul Sabatier
Toulouse, France
maudet@irit.fr

Abstract. This paper follows recent work in the field of dialectical models of inter-agent communication. The request action as proposed by Winograd & Flores is reconsidered in an original dialogue game framework, as a composition of different basic games (small conversation policies). These basic games are based on commitments of participants and are handled through a "contextualization game" which aims at defining how games are opened or closed through the dialogue, and what combinations are legal. We show how such a model offers more practical flexibility, and covers more situations than the classical protocol. Finally, we give an overview on the game simulator that we are currently developing.

1 Introduction

Many authors have convincingly argued that classical approaches to agent communication languages (ACL) [7,9] suffer from a lack of verifiability (due to their mentalistic semantics) [18,3,24] and still require to be used with protocols (because there is simply to much continuations to be explored in practice) [11]. An interesting approach is then to use protocols as a basis to define (some part of) the semantics of ACL. The problem is that protocols usually lack of practical flexibility, genericness, and compositionality, as well as rigorous specifications (due to their underlying models, often finite state machines in the case of sequential interactions). Recently, researchers have begun to address these issues and have explored means to define more appropriate conversation policies (CPs), *i.e.* general constraints on the sequences of semantically coherent messages leading to a goal [11] —as opposed to protocols which refer too often to some specific underlying formalism. A significant amount of approaches on CPs have been inspired, more or less directly, by the work of formal dialectic of Hamblin [12] and Walton and Krabbe [22]. This trend of research really now forms a field of dialectical models of interaction (*computational dialectics*) [10,17]. We have argued elsewhere that this influence has in fact produced two sorts of approaches: *commitment-based* and *dialogue-game-based* CPs [14]. Commitment-based CPs aims at defining semantics of the communicative acts in terms of public notions,

M.-P. Huget (Ed.): Communications in Multiagent Systems, LNAI 2650, pp. 284–299, 2003.

e.g. social commitments. Dialogue-game-based CPs, in addition, consider that protocols are captured within appropriate structures (games), and that these structures can be composed in different ways to form the global structure of dialogue. In this paper, we aim at exploring in practice (on the classical request for action interaction) why and how such a dialogue-game-based approach can be useful.

The rest of the paper is as follows: Section 2 presents the case for conversation policies, using the “request for action” described in [23] as a motivating example. Section 3 introduces the basic material of the dialogue-game-based framework that forms the backbone of our approach. Section 4 reconsiders “request for action” within our framework. A discussion concludes the paper.

2 A Case for Conversation Policies

As suggested by speech-act-based tradition to interagent communication, we assume that messages exchanged between agents have the form of some *dialogue moves*, composed of a dialogue move *type* —or *performative* to use FIPA terminology— (*e.g.* assertion, question, etc.) accompanied with some *content* expressed in a content language. Until recently ACL research issues have primarily related to the generation and interpretation of individual of these ACL messages. Nowadays researchers on ACLs try to address the gap between these individual messages and the extended message sequences, or dialogues, that arise between agents. As part of its program code, every agent must implement tractable decision procedures that allow that agent to select and produce ACL messages that are appropriate to its private state. But these decision procedures must also take into consideration the context of prior ACL messages and other agent events, so that the agent produce messages that are appropriate to this interactional context. While the private state of its agent is not accessible, the interactional context is typically public and accessible to all dialogue participants, which makes easier to verify that the agent indeed conforms to the semantics of its communicative acts. The other advantage is that taking this context into account can actually *simplify* the computational complexity of ACL message selection for an agent. By engaging in preplanned or stereotypical conversations, much of the search space of possible agent responses can indeed be eliminated. The specification of these conversations is usually accomplished by means of protocols. In this paper, we will be interested in interactions between agents where no concurrency is involved (that is, mainly, interactions between two agents involved in a single dialogue). In this context, finite state machines (FSMs) are arguably an adequate and popular formalism to define these protocols. The states of the automaton maps the possible state of the conversation given a message by the participants and the previous state of the conversation. Carefully designed and highly complex protocols have been proposed using these techniques in the literature, and implemented in real applications, see for instance COOL [2]. The Winograd and Flores’s [23] (see Fig. 1) “request for action” is a classical example of such protocols.

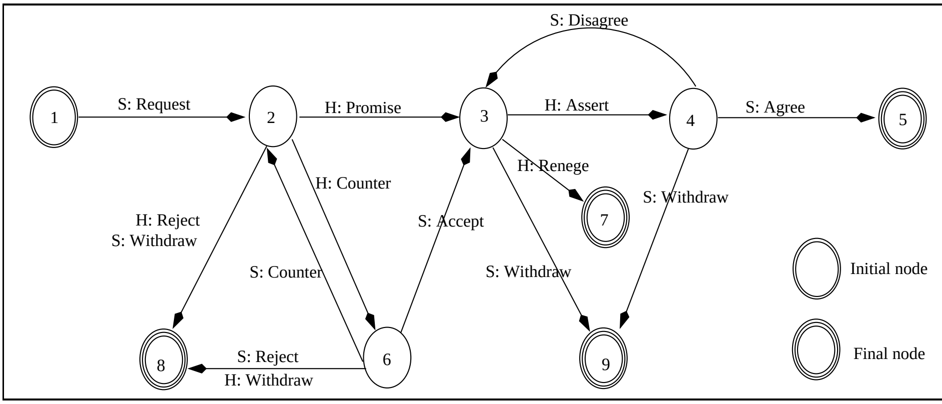

Fig. 1. The request for action protocol

Since this example is central in our paper, we will focus on this latter protocol and describe the dialogue behavior specified by this protocol. Conversation begins in the initial state (1), by a request from speaker S. In state (2), the dialogue can successfully be followed by the promise from H to realize the requested action, or come into a "negotiation cycle" with a counter-proposal from H, or fail with a reject leading to state (8). At state (3), the addressee will signal that the task has been achieved, or eventually decide to renege (leading to the final state (7)) and S will in turn positively (state(5)) or negatively evaluate the achievement of the action (state(3)). Now, in light of the above description and also of the critics and discussion of [11,16,18,19,21,25], we list below some problems encountered by protocols when described as FSMs:

- The state of the conversation is *practically* difficult to describe on the basis of several previous moves. As a result, protocols often describes very tightly constrained interactions (*e.g.* avoiding backtracking moves) and designers possibly omit transitions that may be, intuitively, allowed.
  Example: To determine the state of the dialogue, the "request for action" protocol refers in most cases to *the* previous move only. An exception is the "withdraw" move by S, which can lead to different states of the dialogue (however, all these states are final unsuccessful nodes, so the distinction seems rather misleading). Also one can argue, following [19], that S could make an offer to H without any explicit request from S, in which case the protocol would effectively starts at state (2), and this case is not covered by the protocol.
- The state of the conversation is only the result of the sequence of dialogue move types while in some cases it can be necessary to describe more expressively the state of the conversation.
  Example: Typically, it can useful to constrain the interaction on the basis of some information related to the content of the move. In our example, it could the case that the initial request of the agent also determines a

deadline, and consequently that the declaration that the action is done will be considered legal only if it occurs before this deadline. Alternatively the counter-proposals made by the agents during the negotiation cycle can be required to meet some constraints (for instance, S could only *postpone* the deadline for the action, etc.).

– The protocol is composed of different phases which could be used in the definition of others protocols.
  Example: When considered carefully, our example protocol seems to be composed of different "phases" (or small protocols), not identified as such and not specific to the particular case of the request for action (firstly, the agents will try to negotiate a task for H to do. Next S and H will discuss the correct achievement of this task.)
– Protocols don't allow the agents to consider possibilities to misbehave wrt. the specified interaction (even if this should be discouraged in general).
  Example: The agent which has been requested to undertake the action may decide (on the basis of some private reasons) that it is crucial to postpone its answer. The consequences of this unexpected behaviour should be made explicit, so that the agent can take this parameter into consideration when making its decision.

These remarks emphasis the need to define declarative *conversation policies* (CPs), which are *"general constraints on the sequences of semantically coherent messages leading to a goal"* [11]. The now popular commitment-based approach argues that protocols can be viewed as a set of commitments associated with each state of the dialogue. In this paper, we take more specifically the road of an approach mixing dialogue games and commitments. In the final section of the paper, we discuss whether this approach is appropriate to tackle the problems discussed earlier. But let us start by describing the details of our framework.

3 A Dialogue Game Framework Based on Commitments

We take the picture of two software agents involved in some interaction. To communicate, our agents exchange some communicative acts as can be found in the KQML or FIPA-ACL frameworks. As explained earlier, we assume that their communicative behavior is handled through a notion of dialogue game. The main features of this approach is that the notion of commitments (Sect. 3.1) is used to express the different notions defining the game structure (Sect. 3.2), and that these games can be are grounded (Sect. 3.3) and composed in different ways (Sect. 3.4).

3.1 Commitments

To start with, we give some details about the notion of commitment that we use in our approach. The notion of commitment is a social one, and should not be confused with some psychological notion of commitment. Crucially, commitments are contracted towards a partner or a group. More precisely, commitments are expressed as predicates with an arity of 6:

$$C(x, y, \alpha, t, s_x, s_y)$$

meaning that x is committed towards y to α at time t, under the sanctions s_x and s_y. The first sanction specifies conditions under which x can withdraw from this commitment, and the second specifies conditions under which y reneges the considered commitment. For instance, the following commitment

$$c_1 = C(Al, Bob, sing(Al, midnight), now, 10, 20)$$

states that agent Al is committed towards agent Bob to *sing* at *midnight.* If Al eventually decides to withdraw the commitment he will pay the penalty 10. If Al decides to renege the commitment to sing, he will pay 20. We concede that this account of penalties is extremely simple in this version. A more complete account could be similar to the one of Toledo and al. [6]

The notation is inspired from [19], and allows us to compose the actions involved in the commitments: $\alpha_1|\alpha_2$ classically stands for the choice, and $\alpha_1 \Rightarrow \alpha_2$ for the conditional statement that the action α_2 will occur in case of the occurrence of the event α_1. Finally, the operations on the commitments are just creation and cancellation.

$$c_2 = C(Al, Bob, sing(Al, midnight)|dance(Al, midnight), now, 10, 20)$$

and

$$c_3 = C(Al, Bob, music(Bob, midnight) \Rightarrow create(c_2), now, 10, 20)$$

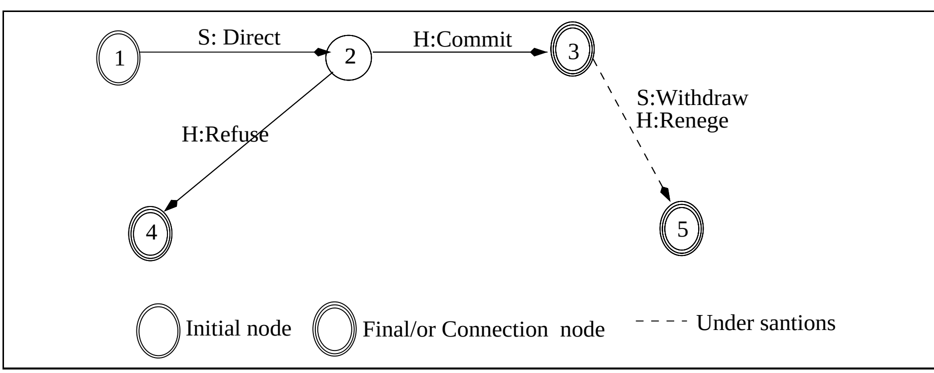

Fig. 2. The different status of the commitments

The commitment c_2 captures that the agent Al is committed towards Bob to *sing* or *dance* at midnight. The commitment c_3 captures that the agent Al is committed to contract the preceding commitment (c_2) if agent Bob plays *music*. All commitments hold a time concerning when they were contracted (*now*). From now, for the sake of readability, we will ignore the *create* operation. We also permit propositional commitments, that we regard as collections of commitments centering on some proposition p, in the line of [22]. Such commitments are typically the result of assertive moves.

Now we need to describe the mechanism by which the commitments are discussed and created during the dialogue. This mechanism is precisely captured within our game structure. To account for the fact that some commitments are established within the contexts of some games *and only make sense within this context*, we make explicit the fact that this commitments are specialized to game g. This will typically be the case of the dialogue rules involved in the games, as we will see below.

3.2 Game Structure

We share with others [8,4,15] the view of dialogue games as structures regulating the mechanism under which some commitments are discussed through the dialogue. Unlike [4,15] however, we adopt a strict commitment-based approach within game structure and express the dialogue rules in terms of commitments. Unlike [8] on the other hand, we consider different ways to combine the structures of the games, and we precise how to derive all other games from some *basic* dialogue games —considering only the degree of strength [21].

In our approach, games are considered as bilateral structures defined by *entry conditions* (which must be fulfilled at the beginning of the game, possibly by some accommodation mechanism), *success conditions* (defining the goals of the participants when engaged in the game), *failure conditions* (under which the participants consider that the game reached a state of failure), and *dialogue rules*. As previously explained, all these notions, even dialogue rules, are defined in terms of (possibly conditional) commitments. Technically, games are conceived as structures capturing the different commitments created during the dialogue.

To sum up, we have Entry conditions (E), Success conditions of initiator (SI) and partner (SP), Failure conditions of initiator (FI) and partner (FP), and dialogues Rules (R) for each game. We also assume that there is a constant sanction s_g to penalize the agents that will not follow the expected dialogic behavior (as described in the Dialogue Rules). Within games, conversational actions are time-stamped as "turns" (t_0 being the first turn of dialogue within this game, t_f the last). To make things more concrete, let us illustrate these ideas with a directive game presented in Fig. 3.

Imagine that agent *Al* and agent *Bob* have entered the directive game. *Al* is committed to play a directive move towards agent *Bob*, and *Bob* is committed to create a commitment to play a commit (i.e., an accept) or a refuse if *Al* honors his commitment. The game follows the automata as described in Fig. 2, but note that the agents have the possibility to play some moves not expected (in this case, they have the penalty s_g). When the game expires (successfully or not), the commitments that were specialized to this game, those which are dependent on the context "g", are automatically cancelled. The others are known as "persistent" commitments. Thereafter, we will omit the sanctions in our games specifications for better readability.

$$
\begin{array}{l|l}
E_{bd} & \neg C(y, x, \alpha, t_0, s_y, s_x) \\
Si_{bd} & C(y, x, \alpha, t_f, s_y, s_x) \\
SP_{bd} & \text{Nil} \\
FI_{bd} & C(y, x, \neg\alpha, t_f, s_y, s_x) \\
FP_{bd} & \text{Nil} \\
R_{bd} & C_g(x, y, \mathit{direct}(x, y, \alpha), t_0, s_g, s_g) \\
 & C_g(y, x, \mathit{direct}(x, y, \alpha) \Rightarrow \\
 & C_g(y, x, \mathit{commit}(y, x, \alpha) | \mathit{refuse}(y, x, \alpha), t_1, s_g, s_g), t_0, s_g, s_g) \\
 & C_g(y, x, \mathit{commit}(y, x, \alpha) \Rightarrow \\
 & C(y, x, \alpha, t_2, s_y, s_x), t_0, s_g, s_g) \\
 & C_g(y, x, \mathit{refuse}(y, x, \alpha) \Rightarrow \\
 & C(y, x, \neg\alpha, t_2, s_y, s_x), t_0, s_g, s_g)
\end{array}
$$

Fig. 3. Definition of a basic directive game.

Now, from this basic directive game, all other combinations as *S: direct*, *S: request*, *S: demand*, *S: order*, *S: command*, etc., and *H: commit*, *H: accept*, *H: promise*, *H: certify*, etc. can be negotiated between the two participants. To do that, S and H should negotiate the degree of strength of the two main speech acts composing the game. Now, let us explain what do we mean by the degree of strength. Speech acts are expressed with different *degrees of strength* depending on the illocutionary force. For example, the degree of strength of a supplications is greater than that of a request, because an initiator who supplicates expresses a stronger desire than a speaker who requests. According to this, the relations of comparative strength that exist between English illocutionary forces in virtue of semantic definitions of English performative verbs can be exhibited in semantic tables by constructing logical trees in accordance with the following rules [20]:

1. All nodes of a semantic table are speech act verbs with the same force;
2. A verb is the immediate successor of another verb if and only if the force that it names can be obtained from the force named by the other verb by adding new composants or increasing the degree of strength.

Such tree exhibits in fact relations of comparative strength between illocutionary forces. These trees can be reflected by some integers which measure the degree of strength of illocutionary forces. By convention, we select 0 (zero) to represent the *neutral* degree of strength that is characteristic of the primitive illocutionary force, +1 the next stronger degree, +2 the next stronger, etc. Similarly, −1 represents the greatest degree of strength smaller than zero, etc. Now, if we consider "assert", "commit", "direct", "declare", as the *primitives* of assertives, commissives, directives and declaratives, we can represent the four trees as follows:

– *for assertives:* suggest(degree = −1), assert(degree = 0), tell(degree = +1), inform(degree = +2), reveal(degree = +3), divulge(degree = +4), etc.
– *for commissives:* commit(degree = 0), accept(degree = +1), promise(degree = +2), certify(degree = +3), etc.

– *for directives:* suggest(degree = −1), direct(degree = 0), request(degree = +1), demand(degree = +2), order(degree = +3), etc.
– *for declaratives:* declare(degree = 0), renonce(degree = +1), terminate(degree = +2), cancel(degree = +3), etc.

Our assumption here is that this degree of strength will define in turn the sanctions (s_x, s_y) under which agents can withdraw, renonce, renege, etc. their commitments, as well as the sanctions s_g applied to the rules of the game. Both participants in conversation can then negotiate this degree of strength before entering the game. Formally, a game g is then represented by the following statement

$$\langle basic\_game(ds_I, ds_P), (E, S, F, R) \rangle$$

where *basic_game*, reflects some basic game as the directive game; (ds_I, ds_P) reflects degrees of strength from the basic type game for the initiator (I) and the partner (P), and the tuple (E, S, F, R) reflects the structure of the proposed game. Note that in any game, we only take into account the speech act of initiator as *request*, *ask*, *etc.* and the "acceptance" of partner as *commit, promise, assert, etc.* In other words, we do not take into account the "refusal" of partner and that is why we only consider two degrees of strength.

3.3 Grounding the Games

The specific question of how games are grounded through the dialogue is certainly one of the most delicate [13]. Following [17], we assume that the agents can use some meta-acts acts of dialogue to handle game structure and thus propose to enter in a game, propose to quit the game, and so on. Thus agents can exchange messages as

$$propose.enter(Al, Bob, g_1)$$

where g_1 describes a well-formed game structure (as detailed above). This message is a proposal of the agent *Al* to agent *Bob* to enter the game g_1. This means that games can have different status: they can be *open*, *closed*, or simply *proposed*. How this status is discussed in practice is described in a *contextualization* game which regulates this meta-level communication. As a simple first account of this game, we could adopt the intuitive view of games simply opened through the successful exchange of a propose/accept sequence. However, things are getting more complicate if we want to take account different kinds of combinations. All these kinds of structurations are considered within a contextualization game that we do not detail here.

3.4 Composing the Games

As explained before, the possibility to combine the games is a very attractive feature of the approach. The seminal work of [22] and the follow-up formalisation of [17] have focused on the classical notions of *embedding* and *sequencing*, but recent works extends the kinds of combinations studied [15]. We now detail the games' compositions that we use in our framework. Describing these kinds of combinations, we precise the conditions under which they can be obtained, and their consequences. Ultimately, such conditions and consequences should be included in the contextualization game we are working on [13].

Sequencing noted $g_1; g_2$, which means that g_2 starts immediately after termination of g_1.
Conditions: game g_1 is *closed*.
Effects: termination of game g_1 involves entering g_2.

Choice noted $g_1|g_2$, which means that participants play either g_1 or g_2 non-deterministically. Not surprisingly, this combination has no specific conditions nor consequences.

Pre-sequencing noted $g_2 \rightsquigarrow g_1$, which means that g_2 is opened while g_1 is proposed.
Conditions: game g_1 is *proposed*.
Effects: successful termination of game g_1 involves entering game g_2.
Such pre-sequencing games can be played to ensure that entry conditions of a forthcoming game are actually established —for instance to make public a conflictuel position before entering a persuasion game. In case that the first game is not successful, the second game is simply ignored.

Embedding noted $g_1 < g_2$, which means that g_1 is now opened while g_2 was already opened.
Conditions: game g_1 is *open*.
Effects: (conversational) commitments of the embedded games are considered having priority over those of the embedding game. Much work needs to be done to precisely define this notion within this framework, but this may be captured by constraining the sanctions related to the embedded game to be greater than those of the embedding game ($s_{g2} > s_{g1}$).

4 Request for Action

Our aim in this paper is to reconsider the famous "request for action" (see Fig. 1) protocol within our dialogue-game-based framework. Considering the protocol as initially stated by Winograd and Flores (see figure), we have found that such protocol requires four basics building dialogue games : (1) a "request" game (rg); (2) an "offer" game (og), (3) an "inform" game (ig) and (4) an "ask" game (ag).

4.1 Request Game (rg)

The request game as specified by Winograd and Flores captures the idea that the initiator (I) "request" the partner (P) and this latter can "promise" or "reject".

In our framework, this starts with the contextualization game in which I and P negotiate the establishment of the following game

$$\langle basic\_directive(+1, +2), (E_{rg}, S_{rg}, F_{rg}, R_{rg})\rangle$$

Both agents should also adapt their new conditions and rules from those of the primitive directive. The new conditions and rules are described in Fig. 4.

$$\begin{array}{l|l}
E_{rg} & \neg C(y, x, \alpha, t_0) \\
SI_{rg} & C(y, x, \alpha, t_f) \\
SP_{rg} & \text{Nil} \\
FI_{rg} & C(y, x, \neg\alpha, t_f) \\
FP_{rg} & \text{Nil} \\
R_{rg} & C_g(x, y, request(x, y, \alpha), t_0) \\
 & C_g(y, x, request(x, y, \alpha) \Rightarrow \\
 & C_g(y, x, promise(y, x, \alpha) | refuse(y, x, \alpha), t_1), t_0) \\
 & C_g(y, x, promise(y, x, \alpha) \Rightarrow C(y, x, \alpha, t_2), t_0) \\
 & C_g(y, x, refuse(y, x, \alpha) \Rightarrow C(y, x, \neg\alpha, t_2), t_0)
\end{array}$$

Fig. 4. Conditions and rules for the request game.

Notice that I and P "request" and "promise" and consequently they should be more committed in this case than in the case where the first one "suggest" and the second "commit". Such increase in the degree of commitment should be reflected by sanctions which should be greater in the first case than in the second case.

4.2 Offer Game (*og*)

An offer is a promise that is conditional upon the partner's acceptance. To make an offer is to put something forward for another's choice (of acceptance or refusal). To offer then, is to perform a conditional commissive. The game can be described as

$$\langle basic\_offer(0, +1), (E_{og}, S_{og}, F_{og}, R_{og})\rangle$$

Precisely, to offer α on condition that the partner accept α. Conditions and rules are in this case are described in Fig. 5.

4.3 Information Game (*ig*)

This game starts with the couple $I : assert$ and $P : agree$ or $P : disagree$ which denotes in fact the couple with (0,+1) according to the tree of strength.

$$\langle basic\_assertive(0, 0), (E_{ig}, S_{ig}, F_{ig}, R_{ig})\rangle$$

$$
\begin{array}{l|l}
E_{og} & \neg C(x,y,\alpha,t_0) \\
SI_{og} & C(x,y,\alpha,t_f) \\
SP_{og} & \text{Nil} \\
FI_{og} & C(x,y,\neg\alpha,t_f) \\
FP_{og} & \text{Nil} \\
R_{og} & C_g(x,y,\mathit{offer}(x,y,\alpha),t_0) \\
 & C_g(y,x,\mathit{offer}(x,y,\alpha) \Rightarrow \\
 & C_g(y,x,\mathit{accept}(y,x,\alpha)|\mathit{refuse}(y,x,\alpha),t_1),t_0) \\
 & C_g(x,y,\mathit{accept}(y,x,\alpha) \Rightarrow C(x,y,\alpha,t_2),t_0) \\
 & C_g(x,y,\mathit{refuse}(y,x,\alpha) \Rightarrow C(x,y,\neg\alpha,t_2),t_0)
\end{array}
$$

Fig. 5. Conditions and rules for the offer game

Notice that a partner can be in the disposition of being in accord or agreement with someone without uttering any words. He can also agree by doing a speech act. In this case, he agrees when he can assert a proposition p while presupposing that the initiator has previously put forward p and while expressing his accord or agreement with this initiator as regards p. To disagree is to assert $\neg p$ when the other has previously put forward p. In this game, we assume that the successful termination is when an agreement is reached about the proposition p. The conditions and rules for this couple are detailed in Fig. 6.

$$
\begin{array}{l|l}
E_{ig} & C(y,x,p,t_0) \text{ or } C(y,x,\neg p,t_0) \\
SI_{ig} & C(y,x,p,t_f) \text{ and } C(x,y,p,t_f) \\
SP_{ig} & \text{Nil} \\
FI_{ig} & \text{Nil} \\
FP_{ig} & \text{Nil} \\
R_{ig} & C_g(x,y,\mathit{assert}(x,y,p),t_0) \\
 & C_g(y,x,\mathit{assert}(x,y,p) \Rightarrow \\
 & C_g(y,x,\mathit{assert}(y,x,p)|\mathit{assert}(y,x,\neg p),t_1),t_0) \\
 & C_g(x,y,\mathit{assert}(x,y,p) \Rightarrow C(x,y,p,t_1),t_0) \\
 & C_g(y,x,\mathit{assert}(y,x,p) \Rightarrow C(y,x,p,t_2),t_0)
\end{array}
$$

Fig. 6. Conditions and rules for the inform game

4.4 Ask Game (*ag*)

We use "ask" in the sense of asking a question, which consists to request the partner to perform a future speech act that would give the initiator a correct answer to his question (in the context of this protocol, the questions will have the form "is the work W finished" and will expect an assertion or a denial that W is finished as a possible answers). According to these remarks, we propose for the *ask* game, described as

$\langle \textit{basic\_question}(0,0), (E_{ag}, S_{ag}, F_{ag}, R_{ag})\rangle$

The structure pertaining to this game is described in Fig. 7.

E_{ag}	Nil
SI_{ag}	$C(y,x,p,t_f)$ or $C(y,x,\neg p,t_f)$
SP_{ag}	Nil
FI_{ag}	Nil
FP_{ag}	Nil
R_{ag}	$C_g(x,y,\textit{question}(x,y,p),t_0)$ $C_g(y,x,\textit{question}(x,y,p) \Rightarrow$ $C_g(y,\textit{assert}(y,x,p)\vert\textit{assert}(y,x,\neg p),t_1),t_0)$ $C_g(y,x,\textit{assert}(y,x,p) \Rightarrow C(y,x,p,t_2),t_0)$

Fig. 7. Conditions and rules for the ask game

4.5 Request Action Reconsidered

Taking for granted that our agents both have access to the basic building games as described above —and handle these games through the use of a contextualization game that we have sketched— we will now first study the detail of how a conversational behavior following the Winograd and Flores (WF) request for action protocol can be captured. We also illustrates the flexibility of the formalism by adding situations not considered in the initial protocol. An example concludes the section.

To start with, it is clear that WF basically consists of a request game followed with an evaluation phase. How will the result of the action be evaluated? In the WF protocol, it is assumed that the partner informs the initiator when the action is done. The combination is typically a pre-sequencing, since it only makes sense to play the *inform* game in case of acceptance of the request.

$rg \leadsto ig$

Now, as illustrated by the WF, it is possible that the agents enter some negotiation cycle about the requested action. This means that, after the initial request, we could find a sequence of different offers and requests made by the agents. We use the shortcut $(*)$ to stipulate that the sequence can be repeated a number of times, with different *request* and *offer* games, of course (conditions on these requests and offers are not detailed here).

$(rg; og)^* \leadsto ig$

As described so far, the resulting structure simply captures the classical WF protocol. Now, we consider in addition that the initiator may want to ask himself whether the action is completed. Thus we have the following amended structure, capturing that participants may choose an *inform* or an *ask* game to trigger the evaluation.

$(rg; og)^* \leadsto (ig|ag)$

Also, an important possibility not considered is that the agents may have some conflictuel position about the achievement of the action. In this case, they may want to enter some *persuasion* game to convince the other. Such a *persuasion* game (pg) is not detailed here, but can be regarded as another subtype of a directive game, where the initiator challenges the partner and demands some justification to support some proposition p. Thus we may have the following combination of games, where it is possible to embed in the *inform* (or *ask*) game a *persuasion* game to reach agreement:

$(rg; og)^* \leadsto (ig|ag) < (pg)^*$

All this assumes that the basic WF is initially described as a pre-sequence of a *request* and an *evaluation* game. Note that others combinations might be considered. For instance, we could assume that the initiator's request will be honored without any explicit feedback from the partner. In this case, the combination of games could be

$rg < (ig|ag)$

4.6 Example

To make things more concrete, we include an example involving *Al* and *Bob*. The dialogue starts when *Al* requests *Bob* to support him in the course of a reviewing process.

$propose.enter(Al, Bob, rg \leadsto ig)$

Bob would like to help his friend, but he is very busy at the moment so he wouldn't like to be penalize at this level. *Bob* refuses. *Al* proposes as an alternative the game *suggest* (sg) where the penalties are more acceptable (the *suggest* game is just another subtype of the basic directive game).

$propose.enter(Al, Bob, sg \leadsto ig)$

$accept.enter(Bob, Al, sg \leadsto ig)$

The preceding moves are examples of the game-level negotiation that we have discussed in the paper. The immediate consequence is that all the commitments described in the game are created. In particular, *Bob* has contracted the following commitment :

$C(Bob, Al, review(Bob, monday), now, 10)$

Now it is Monday in the world of our agents, and *Bob* informs *Al* that he has completed the review.

$inform(Bob, Al, done(review))$

Unfortunately, agent *Al* does not seem quite satisfied with the received review form. He does not agree that the action was done as requested.

$$assert(Al, Bob, not(done(review)))$$

Note that the current state of the dialogue (where a conflict of opinion is explicit) makes possible to enter a persuasion game, where for instance, agent *Bob* would challenge *Al* and ask why he is not happy with the review form. *Al* could in turn explains that *Bob* has choosen a borderline recommendation, a case forbidden by the review guidelines. The detail of how such persuasion dialogue can be managed within dialogue game framework might be found in [1]. In case of successful termination of the persuasion game, an agreement is found and the protocol ends.

4.7 Towards a Game Simulator

The currently in progress agent Dialogue Game Simulator adopts some facets presented in this paper. In particular, it offers a graphical environment which is able to simulate conversations between agents with dialogue games. Precisely, each user must initially choose a "scenario file" which is assigned to his agent. This file, created beforehand, contains the actions than he will carry out during simulation at a predetermined time. A file which describes a game is composed of entry conditions, success conditions, failures conditions, as well as rules of the game. Moreover, for each speech act forming part of a game, we can define a constraint which indicates the hierarchical relation which must exist between the sender and the receiver to be able to play the action in question.

For each initiated dialogue, a workspace is created containing the following: implicated agents, a stack of dialogue games as well as a chart of sent messages (similar to the sequence diagram of UML). The management of most of the compositions of dialogues is ensured by the stack which is in charge to keep the trace of dialectical shifts allowing thus to (a) manage the coherence of the conversation, (b) avoid conflict situations and, (c) possibly detect fallacies.

Notice also that each agent has an agenda, a structure containing its commitments in action as well as the propositional commitments that this agent had contract during the simulation. An agent's agenda also contain commitments of others agents which were contracted in a dialogue with him. As mentioned previously, the commitments established in a particular context are withdrawn from the agendas when the game which generated them, is closed.

The simulator is coded in JAVA, agents are developed with JACK —an agent development tool in JAVA, and files concerning the games are written in XML.

5 Discussion

Let us now return to the problems discussed in section 2, and investigate whether dialogue-game-based approaches as the one suggested here can help to meet these requirements:

- Identifying the state of the dialogue as a set of commitments (precisely those commitments that are relevant to constrain the interaction at this stage) is a key point to make CPs more easy to design. It allows the designer to use some declarative rules referring to these meaningful commitments, thus facilitating the design of most games while improving clarity and expressiveness. As a consequence, the resulting CPs are practically more flexible.
- Keeping track of the commitments of the participants also makes possible future references to these commitments instead of just referring to the previous moves of dialogue. For instance, it is possible to design dialogue rules where a request can obligates the other agent to promise to undertake an action before a given deadline.
- The use of small CPs offers more flexibility and genericness in the sense that those basic games can be composed for any other complex game. The contextualization game sketched in the paper offers the possibility to shift from game to game during conversation, but needs to be studied more carefully. One can specify, for instance the case where the initiator's request can be honored without an explicit acceptation of the partner; or the case where the initiator can ask if the commitment is satisfied.
- Public commitments as suggested in our approach motivates agents to conform to some expected behavior and thus facilitate coordination in the dialogue. However, agents remain autonomous and may decide to violate the commitments and pay the sanctions if they find good reasons to do so, offering a promising balance between normativeness and autonomy.

It is clear that much work needs to be done regarding some foundational aspects of the approach: the mechanisms of the commitments and the sanctions need to be explored to be exploitable by some computational agents, the contextualization and combinations of games will certainly need corrections when faced with to others case studies. However, by trying to investigate in the light of a classical example how this promising dialogue-game-based approach can be used, we feel that this paper contributes to the development of future flexible conversational policies.

References

1. L. Amgoud, N. Maudet, and S. Parsons. Modelling dialogues using argumentation. In *Proceedings of the 4th International Conference on Multi-Agent Systems (ICMAS00)*, pages 31–38, Berlin, 2000. IEEE Press. 297
2. M. Barbuceanu and M. Fox. Cool: A language for describing coordination in multiagent systems. In *Proceedings of the first International Conference on Multi-Agent Systems (ICMAS)*, pages 17–25, 1995. 285
3. M. Colombetti. Commitment-based semantics for agent communication languages. In *Proceedings of the First Workshop on the History and Philosophy of Logic, Mathematics and Computation (HPLMC00)*, San-sebastian, 2000. 284
4. M. Dastani, J. Hulstijn, and L. V. der Torre. Negotiation protocols and dialogue games. In *Proceedings of the 5th International Conference on Autonomous Agents (Agents2001)*, pages 180–181, Montreal, Canada, 2001. ACM Press. 289, 289

5. F. Dignum and M. Greaves, editors. *Issues in agent communication*, volume 1916 of *Lecture Notes in Computer Science*. Springer-Verlag, 2000. 299, 299, 299
6. C. Excelente-Toledo, R. A. Bourne, and N. R. Jennings. Reasoning about commitments and penalties for coordination between autonomous agents. In *Proceedings of fifth International Conference on Autonomous Agents (Agents2001)*, pages 131–138, Montreal, Canada, 2001. ACM Press. 288
7. T. Finin, Y. Labrou, and J. Mayfield. KQML as an agent communication language. In J. Bradshaw, editor, *Software agents*, pages 291–316. AAAI/MIT Press, 1997. 284
8. R. F. Flores and R. C. Kremer. To commit or not to commit: Modelling agent conversations for action. *Computational intelligence —Special Issue on Agent Communication Languages*, 18(2), 2002. 289, 289
9. Foundation for Intelligent and Physical Agents (FIPA). Communicative act library specification (xc00037h). http://www.fipa.org/spec, 2001. 284
10. T. F. Gordon. Computational dialectics. In P. Hoschka, editor, *Computers as assistants - A new generation of support systems*, pages 186–203. Lawrence Erlbaum Associates, 1996. 284
11. M. Greaves, H. Holmback, and J. Bradshaw. What is a conversation policy? In *[5]*, pages 118–131. 2000. 284, 284, 286, 287
12. C. L. Hamblin. *Fallacies*. Methuen, 1970. 284
13. N. Maudet. Negotiating games. *Journal of autonoumous agents and multi-agent systems* (research note, to appear). 291, 292
14. N. Maudet and B. Chaib-draa. Commitment-based and dialogue-game based protocols: new trends in agent communication languages. *The Knowledge Engineering Review*, 17(2), 2002. 284
15. P. McBurney and S. Parsons. Games that agents play: a formal framework for dialogue between autonomous agents. *Journal of Logic, Language, and Information —Special issue on logic and games*, 11(3), 2002. 289, 289, 292
16. P. McBurney, S. Parsons, and M. Wooldridge. Desiderata for agent argumentation protocols. In C. Castelfranchi and W. L. Johnson, editors, *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2002)*, pages 402–409, Bologna, Italy, 2002. AAMAS. 286
17. C. Reed. Dialogue frames in agent communication. In *Proceedings of the Third International Conference on MultiAgent Systems (ICMAS98)*, pages 246–253, Paris, France, 1998. IEEE Press. 284, 291, 292
18. M. P. Singh. Agent communication languages: rethinking the principles. *IEEE Computer*, pages 40–47, 1998. 284, 286
19. M. P. Singh. A social semantics for agent communication language. In *[5]*, pages 31–45. 2000. 286, 286, 288
20. D. Vanderveken. *Meaning and speech acts*. Cambridge University Press, 1991. 290
21. L. Vongkasem and B. Chaib-draa. Acl as a joint project between participants. In *[5]*, pages 235–248. 2000. 286, 289
22. D. Walton and E. Krabbe. *Commitment in dialogue: basic concepts of interpersonal reasoning*. State University of New York Press, Albany, NY, 1995. 284, 289, 291
23. T. Winograd and F. Flores. *Understanding computers and cognition: a new foundation for design*. Addison-Wesley, MA, 1986. 285, 285
24. M. Wooldridge. Semantic issues in the verification of agent communication languages. *Journal of Autonomous Agents and Multi-Agent Systems*, 3(1):9–31, 2000. 284
25. P. Yolum and M. P. Singh. Flexible protocol specification and execution: applying event calculus planning using commitments. In *Proceedings of the first International Conference on Autonomous Agents and Multi-agent Systems (AAMAS2002)*, pages 527–534, 2002. 286