ACL: Specification, Design and Analysis All Based on Commitments

Mathieu Bergeron and Brahim Chaib-draa

Université Laval, Département d'Informatique et de Génie Logiciel,
Sainte-Foy, Québec, Canada G1K 7P4
{bergeron, chaib}@damas.ift.ulaval.ca

Abstract. In recent years, social commitment based approaches have been proposed to solve problems issuing from previous mentalistic based semantics for agent communication languages. This paper follows the same line of thought since it presents a dialogue game based agent communication language (called DIAGAL) which allows agents to manipulate the public layer of social commitments through dialogue, by creating, cancelling, . . . , updating their social commitments. Then we show how we can study the commitments dynamic to model agent dialogue and we present some metrics that can be used to evaluate the quality of dialogue. Next, we use an example (summer festival organization) to show how DIAGAL can be used in analyzing and modelling automated conversations in offices. Finally, we present the results and analysis of the summer festival simulations that we realized through our dialogue game simulator (DGS).

1 Introduction

Dialogue games reflect interactions between different participants in dialogue or conversations. In such interactions, each participant intervenes by making utterances according to a pre-defined set of rules. Typically, the rules define how the dialogue may or must start, what statements may or must be uttered in a given context and, how the dialogue may or must terminate. Such games have found many applications in the past. For example, they have been used in ancient and medieval philosophy for argumentation, and more generally, for logical thinking. In modern philosophy, they have been used for argumentation theory related to the contextual analysis of fallacious reasoning.

Dialogue games have also been applied in computational linguistics, computer science and cognitive science. Recently, dialogue games have been proposed as the basis for "conversation policies" for autonomous software agent communication. To this end, work has focussed on persuasion dialogues [1]; negotiation dialogues [2, 13]; agent-team formation dialogues [4]; commitment dialogues [8]; dialogues for rational interactions [10], etc. However, none has considered an approach entirely based on commitments to specify, design and analyse agents conversations. This paper attempts to fill this gap by proposing (i) a new dialogue game adopting a strict commitment approach within game structure; (ii) a method that studies the commitments dynamic for modelling agents conversations and; (iii) metrics based on commitments to analyse agents conversations.

2 DIAGAL: A Dialogue Game Agent Language

2.1 Commitments

As our approach is based on commitments, we start with details about the notion of commitment. The notion of commitment is a social one, and should not be confused with its other side based on the psychological notion. Crucially, commitments are contracted towards a partner and they are expressed as predicates with an arity of 7. Thus, an action commitment takes the form :

$$C(x, y, \alpha, t, s_x, s_y, \texttt{Sta})$$

meaning that agent x is committed towards y to do α at time t (commitment creation time), under the sanctions s_x and s_y. The first sanction specifies what happens if x cancels or violates the commitment, and the second specifies what happen if y tries to withdraw the commitment.[1] Since a commitment can be in different states, we add the `Sta` parameter to specify the commitment state. An accepted propositional commitment would have propositional content p instead of α. We use propositional commitments to commit on the current state of the world and action commitments to commit to realize an action in the future. This commitment notation is inspired from [14], and allows us to compose the actions or propositions 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 the case of the realisation of the action α_1.

Commitments that can exist between the various agents of a MAS evolve: their states change. A commitment can be in one of the following states:

- *Inactive* (`Ina`): By default, a commitment is inactive.
- *Created* (`Crt`): A commitment is in this state if it has been "socially established" in a dialogue game.
- *Canceled* (`Cnl`): A commitment is in this state if it has been withdrawn by the debtor.
- *Released* (`Rel`): A commitment is in this state if it has been withdrawn by the creditor.
- *Fulfilled* (`Ful`): A commitment is in this state if the debtor has satisfied its content.
- *Violated* (`Vio`): A commitment is in this state if the debtor hasn't respected its content.
- *Failed* (`Fal`): A commitment is in this state if an attempt to socially established it has failed.

[1] A sanction can be apply to the creditor because the debtor can really want to respect his commitment. For example, a company which invests a lot of time and money to respect a contract does not want that this contract can be canceled.

Now we need to describe the mechanism by which the commitments states are modified during the dialogue. This mechanism is precisely captured within our game structure.

2.2 Game Structure

The main particularity of social commitments is that they must be socially established in order to hold. This means that every change on the social commitment layer should be grounded by the conversing agents. We share with others [3, 6, 10] the view of dialogue games as structures regulating the mechanism under which some commitments are discussed through the dialogue. Unlike others [3, 10] however, we adopt a strict commitment-based approach within game structure and express the dialogue rules in terms of *dialogical commitments* [8]. 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 those *dialogical commitments* are particular to game g (by indicating g as a subscript). This will typically be the case for the dialogue rules involved in the games, as we will see below. In our approach, games are considered as bilateral structures defined by:

- *entry conditions* (E): expressed in terms of extra-dialogical commitments, entry conditions are conditions which must be respected to enter the game;
- *dialogue rules* (R): expressed in terms of dialogical commitments, dialogue rules specify what the conversing agents are "dialogically" committed to do. The fulfilment of those rules will lead to reaching either the success or the failure conditions of the game;
- *success conditions* (S): success conditions indicate the result (the effect in terms of extra dialogical commitments) of the dialogue game if the modification of the public layer, which was the purpose of the game, has been socially accepted;
- *failure conditions* (F): failure conditions indicate the result of the dialogue game if the modification of the public layer has been socially rejected.

In these conditions, a game g is formally defined as a 4-tuple (E_g, S_g, F_g, R_g), where E_g is the set of commitments that specify the entry conditions, S_g the set of commitments that specify the success conditions, F_g the set of commitments that specify the failure conditions and R_g the set of commitments that specify the rules of the dialogue game.

2.3 Dialogue Games

We have defined four basic dialogue games for the creation of social commitments, which are exactly those which lead (in case of success) to the four types of commitments which can hold between two agents x and y.

1. Agent x uses a *Request* game to attempt to have an action commitment from y toward x accepted;

2. Agent x uses an *Offer* game to attempt to have an action commitment from x toward y accepted;
3. Agent x uses an *Ask* game to attempt to have a propositional commitment from y toward x accepted;
4. Agent x uses an *Inform* game to attempt to have a propositional commitment from x toward y accepted.

Four other games allow retracting the four types of commitments which can hold between two agents x and y.

1. Agent x uses a *CancelActionC* game to attempt to retract an action commitment from x toward y;
2. Agent x uses a *ReleaseActionC* game to attempt to retract an action commitment from y toward x;
3. Agent x uses a *CancelPropC* game to attempt to retract a propositional commitment from x toward y;
4. Agent x uses a *ReleasePropC* game to attempt to retract a propositional commitment from y toward x.

If an agent wants to modify a commitment (change any attribute(s) of the commitment except the debtor or the creditor), he can try to retract the commitment and create a new one with the new attribute(s). However, such cancellation may result in some undesirable sanctions being applied. This is why, we have defined two more games that allow attempts to update commitments without having to face sanctions. These games are:

1. Agent x and y use an *UpdateActionC* game to attempt to update an action commitment;
2. Agent x and y use an *UpdatePropC* game to attempt to update a propositional commitment.

As example of conditions and rules, let us consider the *Request* game which captures the idea that the initiator (x) "requests" an action α from the partner (y) who can "accept" or "refuse". The conditions and rules of the *Request* game are presented in figure 1. Sanctions were omitted in the *Request* game definition for better readability. In this paper, we only present the *Request* game, but the reader interested by the others games can refer to [11].

Now let's see how a commitment state changes. When an agent needs to change a commitment state, he simply uses the right dialogue game among the ten previously presented. For instance, if an agent x wants to create a commitment in action α of which he is the creditor and y is the debtor, then x should propose to y to play a *Request* game concerning α. Table 1 presents games that an agent x should play relatively to the desired operation on commitments.

Now that DIAGAL's dialogue games have been presented, we must describe how the agents can use them to communicate.

Request game (rg)
E_{rg} $\neg C(y, x, \alpha, t_c, \texttt{Crt})$ and $\neg C(y, x, \neg\alpha, t_c, \texttt{Crt})$ $\forall\ t_c\ , t_c < t_j$
S_{rg} $C(y, x, \alpha, t_f, \texttt{Crt})$
F_{rg} $C(y, x, \alpha, t_f, \texttt{Fal})$
R_{rg} 1) $C_g(x, y, \mathit{request}(x, y, \alpha), t_j, \texttt{Crt})$
2) $C_g(y, x, \mathit{request}(x, y, \alpha) \Rightarrow$ $C_g(y, x, \mathit{accept}(y, x, \alpha) \mid \mathit{refuse}(y, x, \alpha), t_k, \texttt{Crt}), t_j, \texttt{Crt})$
3) $C_g(y, x, \mathit{accept}(y, x, \alpha) \Rightarrow C(y, x, \alpha, t_f, \texttt{Crt}), t_j, \texttt{Crt})$
4) $C_g(y, x, \mathit{refuse}(y, x, \alpha) \Rightarrow C(y, x, \alpha, t_f, \texttt{Fal}), t_j, \texttt{Crt})$

Fig. 1. Conditions and rules for the *Request (rg)* game.

Commitment Type	Current State	Desired State	Dialogue Game
$C(x, y, \alpha)$	Inactive	Created	*Offer*
	Created	Cancelled	*CancelActionC*
$C(y, x, \alpha)$	Inactive	Created	*Request*
	Created	Released	*ReleaseActionC*
$C(x, y, p)$	Inactive	Created	*Information*
	Created	Cancelled	*CancelPropC*
$C(y, x, p)$	Inactive	Created	*Question*
	Created	Released	*ReleasePropC*

Table 1. Games that an agent x should use in order to obtain the different commitment states.

2.4 Grounding and Composing the Games

The specific question of how games are grounded through the dialogue is certainly one of the most delicate [8]. *Grounding* refers to the process of reaching mutual belief (or common ground). Following [12], we assume that agents can use some meta-acts of dialogue to handle game structure and thus propose to enter a game, propose to leave a game, and so on. This means that games can have a 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. Table 2 indicates the current contextualisation moves and their effects in terms of commitments. For example, when a proposition to enter a game g ($\mathit{prop.in}(x, y, g)$) is played by agent x, agent y is committed to accept ($\mathit{acc.in}$), to refuse ($\mathit{ref.in}$) or to propose entering another game g' ($\mathit{prop.in}(y, x, g')$).

Concerning the possibility of composing the games, the seminal work of [15] and the follow-up formalisation of [9] have focused on the notions of *iteration*, *sequencing* and *embedding*:

Iteration: If g is a game, then g^n means that game g is repeated n times, where each occurrence being undertaken until closure and then being followed by the next one.

Move	Operations
$prop.in(x, y, g)$	$create(y, C_g(y, x, acc.in(y, x, g)$ $\|ref.in(y, x, g)\|prop.in(y, x, g')))$
$prop.out(x, y, g)$	$create(y, C_g(y, x, acc.out(y, x, g)$ $\|ref.out(y, x, g)))$
$acc.in(x, y, g)$	create dialogical commitments for game g
$acc.out(x, y, g)$	suppress dialogical commitments for game g
$ref.in(x, y, g)$	no effect on the public layer
$ref.out(x, y, g)$	no effect on the public layer

Table 2. DIAGAL contextualisation game.

Sequencing: If g_1 and g_2 are both dialogue games, then $g_1; g_2$ means that game g_2 starts immediately after termination of game g_1.

Embedding: Here also if g_1 and g_2 are both dialogue games, then $g_2 < g_1$ means that g_2 is embedded in g_1. In fact, it means that g_2 is now opened while g_1 was already opened. Dialogical commitments of the embedded games are considered having priority over those of the embedding game.

3 Commitments Network

As the DIAGAL's dialogue games are based on commitments, it would be interesting to look at how we could specify the agents dialogical behavior of a multiagent system (MAS) using such commitments. In fact, it is interesting to study the links that exist between the different commitments of a MAS. For example, one can wonder when an agent must try to create a particular commitment or try to modify an existing commitment. Actually, it's possible and even probable that an agent attempt to modify the state of a commitment if he notices that the state of another commitment has changed. For example, the cancellation of a commitment C_i can lead an agent to attempt to create another commitment C_j.

Therefore, it's interesting to enumerate the consequences of the commitments states changes in a MAS in order to study the dynamic of the system by the dynamic of the commitments. Moreover, as the dialogue games allow agents to manipulate social commitments, one can try to specify the dialogical behavior of the agents of a MAS by studying the dynamic of the commitments. Consequently, elaborating a commitments network that enumerates the consequences of all the commitments states changes in a MAS to guide the agents in their conversations is very useful.

3.1 Definition

Some preliminary concepts should be defined before specifying a commitments network. Precisely, we need to define the following sets:

- $A = \{a_1, a_2, \ldots, a_k\}$ is the set containing all the agents of a MAS.
- $C = \{c_1, c_2, \ldots, c_n\}$ is the set containing all the commitments that can exist in a MAS.
- $S = \{$`Ina`, `Crt`, `Cnl`, `Rel`, `Ful`, `Vio`, `Fal`$\}$ is the set containing all the states that a commitment can take. These states have been introduced in section 2.1.

It's also important to define the notions of *commitment state modification cause* and *commitment state modification desire*:

- A *commitment state modification cause* is represented by a triplet of the type $(a, c, s)_{cau}$ where $a \in A$, $c \in C$ and $s \in S$. The triplet $(a, c, s)_{cau}$ specifies that the commitment c takes the state s because of agent a. The state modification is observed by the debtor and the creditor of the commitment when it occurs.
- A *commitment state modification desire* is represented by a triplet of the type $(a, c, s)_{des}$ where $a \in A$, $c \in C$ and $s \in S$. The triplet $(a, c, s)_{des}$ specifies that the agent a desires that the commitment c pass in the state s if it's possible.

We can now define the commitments network concept. A commitments network $W = (A, C, L)$ is defined by :

- A : a set of agents.
- C : a set of commitments.
- L : a set of couples of the type $((a_1, c_1, s_1)_{cau}, (a_2, c_2, s_2)_{des})$ where $a_1, a_2 \in A$, $c_1, c_2 \in C$ and $s_1, s_2 \in S$. In fact, this set contains all the causality links that exist between all the commitments of the set C. The link $((a_1, c_1, s_1)_{cau}, (a_2, c_2, s_2)_{des})$ specifies that agent a_2 desires that the commitment c_2 pass in the state s_2 if he notices that te commitment c_1 pass in the state s_1 because of agent a_1.

3.2 Example

Let's suppose that a simple MAS contains three agents (ag_1, ag_2 and ag_3) and that only two commitments can exist in that MAS: `C1` $= C(ag_1, ag_2, \alpha)$ and `C2` $= C(ag_1, ag_3, \beta)$. Now, let's suppose that the creation of the commitment `C1` by the agent ag_2 causes the cancellation of the commitment `C2` by the agent ag_1. Starting from these considerations, we can determine a commitments network for the MAS. The commitments network $W_{ex_1} = (A_{ex_1}, C_{ex_1}, L_{ex_1})$ of this example is defined as follows:

1. $A_{ex_1} = \{ag_1, ag_2, ag_3\}$

2. $C_{ex_1} = \{\texttt{C1},\texttt{C2}\}$
3. $L_{ex_1} = \{((ag_2, \texttt{C1}, \texttt{Crt})_{cau}, (ag_1, \texttt{C2}, \texttt{Cnl})_{des})\}$

So, if the agent ag_1 notices that the commitment `C1` is created by the agent ag_2, he tries (if he wants to follow the commitments network) to cancel the commitment `C2`. To achieve such cancellation, ag_2 should use the game *CancelActionC* as proposed in table 1. In these conditions ag_2 proposes to ag_3 to play the game *CancelActionC* since ag_3 is the creditor of `C2`.

We also developed a graphic method to represent a commitments network which is inspired from the commitments causality diagram of Wan and Singh [16]. At first, a commitment is simply represented by a circle containing the debtor, the creditor and the identifier of the commitment. In addition, a link (arrow) is addded between two commitments (circles) if a state modification of the first commitment causes a state modification desire of the second commitment. The graphic version of the commitments network of the preceding example is presented in figure 2. The link between the two commitments means that the creation of the first commitment by ag_2 ($\texttt{ag}_2\texttt{:Crt}$) results in an attempt by the agent ag_1 of cancelling ($\texttt{ag}_1\texttt{:Cnl}$) the second commitment.

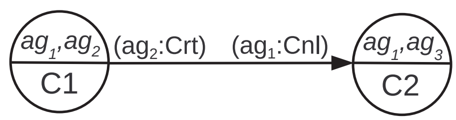

Fig. 2. An example of link between two commitments

So, we can see how the commitments networks can help MAS designers to specify the dialogical behavior of their agents by studying the dynamic of the commitments.

4 Summer Festival Example

We are currently studying how we can use the commitments network and the dialogue games to model offices as systems of communicative actions. Through dialogue games, participants engage in actions by making promises, asking for information, stating facts, etc. and through these actions, they create, modify, cancel, release, ..., fulfill commitments that bind their current and future behavior. The illustrative example on which we focus here concerns the organization of a summer festival. This festival which lasts several days consists of groups of artists coming from various countries. We want all the necessary management of such an event to be done by software agents.

Our summer festival example is a MAS containing five various types of agents who have different tasks to realize. These agents are as follows:

- *AgArtist*: A type of agent representing an artist in the system. An instance is represented by ar_i where i is used to indicate that potentially several artists will be present in the simulation. Such type of agent can accept or refuse an invitation regarding the requested fee.
- *AgPlanner*: An instance of *AgPlanner* (*pl*) is an interface between an agent of type *AgArtist* and the remainder of the system. He is responsible for finding the artists to be invited, for managing a budget as well as a schedule. He also delegates some tasks to the agent *AgSecretary*.
- *AgSecretary*: An instance of *AgSecretary* (sc_i) is an interface between an agent of type *AgPlanner* and the other resource agents (*AgHotels* and *AgTravelAgency*). He is responsible for the hotel and flight reservations.
- *AgHotels*: An instance of *AgHotels* (ht_i) is an agent which represents a conglomerate of hotels.
- *AgTravelAgency*: An instance of *AgTravelAgency* (ta_i) is an agent which represents a conglomerate of airline companies.

4.1 The Example as a Commitments Network

As the complete MAS of the summer festival example contains a lot of commitments, we will only study a small part of it: the invitation of an artist by the planner. At the beginning of the invitation, the planner and the artist must try to find an agreement on the date and the hour of the artist's performance. Then, the planner and the artist must reach an agreement on the fee for the artist's performance. Finally, the planner have to ask the artist if he wants an airline ticket and an hotel room. With this description, it's now possible to define a commitments network for the invitation of an artist.

To specify the commitments network, we must find the commitments that are hidden in the descrition of the invitation of an artist. Initially, the planner must try to *commit* the artist to come to the festival. If the artist accepts the invitation, the planner must *commit* himself to pay the artist for his performance at the festival. Finally, the planner must *commit* himself reserving the airline ticket and the hotel room if the artist accepts. We thus note that the invitation of an artist can lead to the creation of four different commitments. These commitments are presented in the table 3.

Id	Commitment
C1	$C(ar_i, pl, ComeToFesti(ar_i, date, time))$
C2	$C(pl, ar_i, PayFee(pl, ar_i, fee))$
C3	$C(pl, ar_i, ReserveRoom(pl, ar_i, date))$
C4	$C(pl, ar_i, ReserveTicket(pl, ar_i, date))$

Table 3. Commitments that can exist between the planner and artists.

After inventorying the commitments of the summer festival example, we can now elaborate a commitments network of the invitation of an artist. All we have to do is to determine the causality links that exist between the commitments. In our case, the links emerge from the description of the invitation of an artist and from the commitments of table 3. The commitments network of the invitation of an artist $W_{inv} = (A_{inv}, E_{inv}, L_{inv})$ is defined as follows:

1. The agents set $A_{inv} = \{pl, ar_1, ar_2, \ldots, ar_n\}$.
2. The commitments set $C_{inv} = \{\mathtt{C1}, \mathtt{C2}, \mathtt{C3}, \mathtt{C4}\}$.
3. The causality links set $L_{inv} =$
   $\{ ((-,-,-)_{cau}, (pl, \mathtt{C1}, \mathtt{Crt})_{des}),$
   $((pl, \mathtt{C1}, \mathtt{Crt})_{cau}, (pl, \mathtt{C2}, \mathtt{Crt})_{des}),$
   $((pl, \mathtt{C1}, \mathtt{Rel})_{cau}, (pl, \mathtt{C2}, \mathtt{Cnl})_{des}),$
   $((ar_i, \mathtt{C1}, \mathtt{Cnl})_{cau}, (pl, \mathtt{C2}, \mathtt{Cnl})_{des}),$
   $((pl, \mathtt{C2}, \mathtt{Fal})_{cau}, (ar_i, \mathtt{C1}, \mathtt{Cnl})_{des}),$
   $((pl, \mathtt{C2}, \mathtt{Crt})_{cau}, (pl, \mathtt{C3}, \mathtt{Crt})_{des}),$
   $((pl, \mathtt{C2}, \mathtt{Crt})_{cau}, (pl, \mathtt{C4}, \mathtt{Crt})_{des}),\ \}.$

A graphic version of the commitments network is also given in figure 3. With this network, we can for instance see that the establishment of the commitment C1 (which commits the artist to come to the festival) leads (under some conditions) to the creation of C2 (which commits the planner to pay the artist fees).

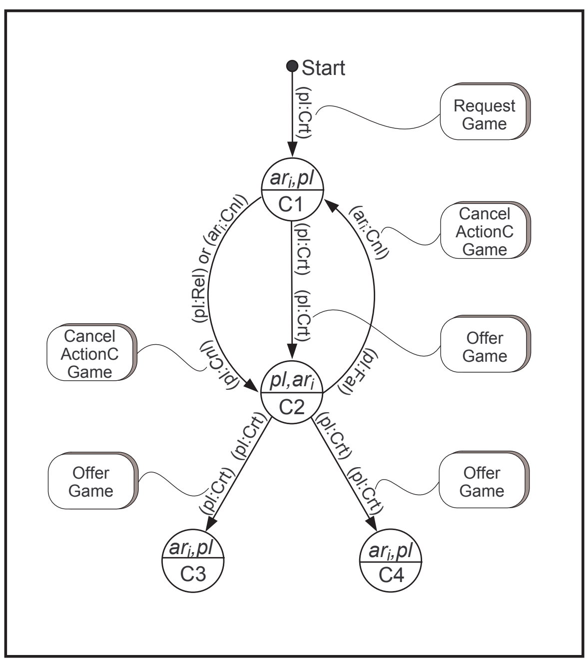

Fig. 3. The commitments network of the invitation of an artist.

Therefore, the commitments network allows us to identify all the commitments states changes that have a consequence on the invitation of an artist. If a commitment state modification is not included in the commitments network, it means that this modification has no consequences. Consequently, the dialogical behavior of the agents is implicitly specified through the commitments network. On figure 3, we specified the dialogue games (for a better comprehension) that the agents need to use to follow correctly the commitments network. In general, it is not necessary to specify the dialogue games because the agents can easily find the games to use according to the commitment type and the state desired for this commitment as we saw earlier in table 1. For example, the planner can easily determine that he has to use a *Request* game to create the commitment `C1`. He has to use a *Request* game because he is the creditor of `C1` and `C1` is an action commitment.

5 Simulation and Results

We have developed a tool (called DGS—Dialogue Game Simulator) which allows us to simulate conversations between agents who use dialogue games to communicate. In fact, the DGS is an implementation of the DIAGAL's theoretical concepts and allows us to study several aspects of the communication between software agents. DGS aims to be an effective tool of validation as well as a means of analyzing dialogues, diagrams and structures concerning the various games.

The main component of the DGS is the agenda. This structure contains action commitments as well as propositional commitments deduced from dialogue rules when an action is played. With it, users can follow the effects of the agents' actions i.e., check the creation, cancellation, fulfillement, . . . of commitments between the agents. In fact, an agenda is a kind of *Commitment Store* where commitments are classified according to the time they were contracted. In addition, the DGS contains an action board that offers the possibility of observing the exchange of messages between agents (UML sequence diagram). Such a board acts as a visual component for the simulator user, to help him understand and analyze what occurred in a dialogue between two agents.

In addition, we defined several metrics to evaluate and analyse the quality of the conversations which take place into the DGS. These metrics make it possible to check and validate the dialogical behavior of the agents of a MAS and are defined on three levels :

- Metrics for the entire MAS.
- Metrics for a dialogue between two agents.
- Metrics for an agent.

Moreover, for each one of these levels, we defined three types of metrics: metrics concerning commitments, metrics concerning dialogue games and general metrics. Some of the metrics we have defined for the entire MAS are presented in table 4.

Type	Metrics
Commitment	Number of action commitments
	Number of prop commitments
	Number of fulfilled commitments
	Number of withdrawn commitments
Dialogue game	Number of dialogue games used
	Proportion of dialogue games used
General	Number of agents in the MAS.
	Number of dialogue in the MAS
	Avg number of dialogue per agents
	Total number of dialogue turn

Table 4. Metrics for the entire MAS.

5.1 Summer Festival Agents

The agents of the summer festival example has been implemented using the JACK language which is an agent-oriented programming language. As we created these agents to test the DIAGAL language, the commitments network, the simulator and the metrics, we did not equip them with a too complex decision-making mechanism. In fact, we used behavior files to model the decision-making mechanism of the summer festival agents. In the behavior files we specify if an agent has a tendency to cancel his commitments, if he respects his commitments and if he often commits himself. For example, an agent can cancel some commitments with a probability of 0.5, respect some commitments with a probability of 0.8 and accept to commit himself to realise an action α with a probability of 0.9. It's important to mention that the behavior files are private to the agent and that the agents negotiate the content (salary of the artist for example) of their commitments even if their decisions are random.

In addition, the summer festival agents must determine which agents to interact with to organize the festival effectively. For example, the planner must try to invite artists who respect and don't cancel their commitments. Consequently, the planner must try to learn the behavior of the artists if he wants to organise the festival more effectively. For this reason, we propose using the concepts of trust and agreement degree. Like Marsh [7], we use the notation $T_x(y)$ ($-1 \leq T_x(y) \leq 1$) to specify the degree of trust that an agent x has in agent y. This trust degree is fixed at the beginning (generally at 0) and evolves according to the actions of the agents (fulfill commitment, cancel commitment, ...). We also use the notation $A_x(y)$ to specify the agreement degree that agent x has toward agent y. This agreement degree is also fixed at the beginning and also evolves according to the actions of the agents. Consequently, the planner must make a good compromise between the trust and agreement degrees to determine which artists he should invite. If A stands for the artist set, then the planner will choose to interact with the agent y who is such as:

$$y = \operatorname*{argmax}_{a \in A}(\gamma\ T_{pl}(a) + \mu\ A_{pl}(a))$$

The γ and μ parameters balance the importance allocated to the two degrees (confidence and agreement). Finally, the summer festival agents use a commitments network to guide their conversations. The commitments network is specified in an XML file which is loaded when the simulation starts.

5.2 Experiments

We simulated 500 organizations of the summer festival to analyze the evolution of the agents from one simulation to the other. We have particularly focused on the interactions between the planner and the artists. The planner has to find 9 artists in a group of 25 artists who have different behaviors and preferences (salary, disponibility, ...). We used our metrics to analyse the summer festival simulations and the work of the planner. We show here the results of the analysis of two metrics that we find significant to evaluate the quality of the planner's work : the percentage of fulfilled commitments by the artists and the number of commitments cancelled by the artists.

The graph in figure 4 presents the average percentage of fulfilled commitments by the artists for each block of 10 simulations. Hence, the graphic contains the average percentage of fulfilled commitments for 50 blocks of 10 simulations. The graph in figure 5 presents the average number of commitments cancelled by the artists for each block of 10 simulations.

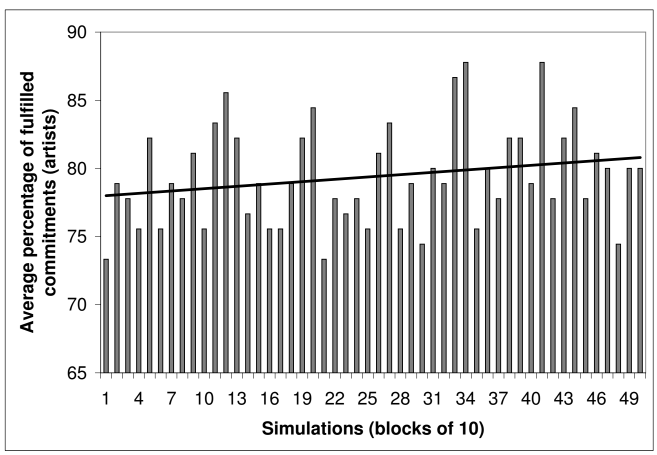

Fig. 4. Average percentage of commitments fulfilled by the artists.

These two graphs show that the planner learns to organise the festival more effectively. In fact, the planner learns to find better artists for the festival. The erratic variation between the results of simulations is attributable to the random behavior of the agents. Indeed, even if the planner invites artists who respect their commitments with a very strong probability, it is possible that several of

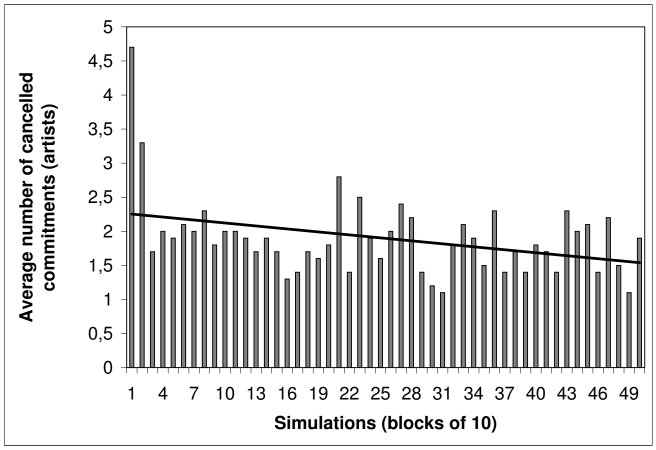

Fig. 5. Average number of commitments cancelled by the artists.

these agents do not respect their commitments. All things considered, these two graphs show us that the metrics are efficient to analyse agents conversations.

6 Conclusion

In this article, we have shown how commitments can be used to specify, design and analyse agent dialogue. At first, we presented our commitment-based agent communication language DIAGAL. At the syntactic level, the DIAGAL's dialogue games appears to be a good alternative between strictly "mentalistic" or "social" approaches and protocols. At the semantic level, dialogue games are defined in terms of entry conditions, success conditions and failure conditions expressed in terms of verifiable extra-dialogical social commitments. Moreover, the conventional part of pragmatics is expressed in terms of conditional dialogical social commitments specifying the rules of the games. Besides, the contextualisation game ensures the grounding of dialogue games (taking into account the attentional level of agents) while dialogue games ensure the grounding of each modification in the social commitments layer. Consequently, DIAGAL offers a complete set of tools to manipulate the social commitments layer.

Thereafter, we introduced the commitments network which allows specifying the dialogical behavior of the agents of a MAS by studying the dynamic of the commitments. Comparatively to Wan and Singh [16] that use commitments causality to analyse conversations examples, we propose to use commitments causality to specify the dialogical behavior of the agents of a MAS. In addition, we showed how we can use metrics based on commitments to analyse agents conversations.

Finally, in our approach, the speech acts make the dialogue games go forward, the dialogue games manipulate the commitments layer and the commitments networks make the joint activity go forward.

References

1. L. Amgoud, N. Maudet, and S. Parsons. Modelling dialogues using argumentation. In *Proceedings of the 4th Conference on Multi-Agent Systems (ICMAS)*, Boston, 2000.
2. L. Amgoud, S. Parsons, and N. Maudet. Arguments, dialogue, and negotiation. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, Berlin, 2000.
3. M. Dastani, J. Hulstijn, and L. V. der Torre. Negotiation protocols and dialogue games. In *Proceedings of the BNAIC*, 2000.
4. F. Dignum, B. Dunin-Keplicz, and R. Vebrugge. Agent theory for team formation by dialogue. In C. Castelfranchi and Y. Lespérance, editors, *Intelligent Agent VII: Proceedings of the Seventh International Workshop on Agent Theories, Architectures and Languages (ATAL 2000)*, pages 150–166, LNAI, 1986, Berlin, Germany, Springer, 2000.
5. F. Dignum and M. Greaves, editors. *Issues in agent communication*. LNAI N0 1916, Springer-Verlag, Berlin, 2000.
6. R. F. Flores and R. C. Kremer. A formal theory for agent conversations for actions. *Computational intelligence*, 18(2)(120-173), 2002.
7. S. P. Marsh. *Formalising Trust as a Computational Concept*. PhD thesis, University of Stirling, April 1994.
8. N. Maudet and B. Chaib-draa. Commitment-based and dialogue-game based protocols–new trends in agent communication language. *The Knowledge Engineering Review*, 17(2):157–179, 2002.
9. P. McBurney and S. Parsons. Games that agents play: A formal framework for dialogues between autonomous agents. *J. of Logic, Lang. and Inf.*, 11(3):315–334, 2002.
10. P. McBurney, S. Parsons, and M. Wooldridge. Desiderata for agent argumentation protocols. In *Procceedings of the First International Conference on Autonomous Agents and Multi-Agents*, Bologna, 2002.
11. P. Pasquier, M. Bergeron, and B. Chaib-draa. Diagal: a generic acl for open systems. In *Proceeding of the Fifth International Workshop Engineering Societies in the Agents World (ESAW)*, volume 3451 of *Lecture Notes in Artificial Intelligence (LNAI)*, pages 139–152. Springer-Verlag, 2004.
12. C. Reed. Dialogue frames in agent communication. In *Proceedings of the Third International Conference on MultiAgent Systems (ICMAS)*, 1998.
13. F. Sadri, F. Toni, and P. Torroni. Logic agents, dialogues and negotiation: an abductive approach. In M. Schroeder and K. S. A. 2001), editors, *Symposium on Information Agents for E-Commerce, AI and the Simulation of Behaviour Conference*, York, UK, 2001. AISB.
14. M. P. Singh. A social semantics for agent communication language. In *[5]*, pages 31–45. 2000.
15. D. Walton and E. Krabbe. *Commitment in Dialogue*. State University of New York Press, 1995.
16. F. Wan and M. P. Singh. Commitments and causality for multiagent design. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 749–756. ACM Press, 2003.