Interactive Decisions in COSMA-Negotiations*

Achim Schupeta
German Research Center for Artificial Intelligence (DFKI)
Project AKA-MOD
Stuhlsatzenhausweg 3
D-66123 Saarbrücken
Germany
E-mail: schupeta@dfki.uni-sb.de
Tel.: ++49 681 302 5320
Fax: ++49 681 302 5341

Abstract

Comunicational interaction of agents can be described at different levels of complexity and abstraction: The lowest level introduces possible *message types*. The *protocol* level links messages together to create communicational contexts. The *decision* layer consists of all (usually domain-dependent) criteria and measurements to rank and compare different suggestions and proposals arising during a negotiation. The usage of these criteria determines an agent's actual decisions. Within the framework of the protocol these decisions compose the *negotiation* strategy of an agent. The *cooperative behavior* can be described by considering the society of all interacting agents and it emerges from the interdependencies of the different negotiation strategies of the agents.

This interaction model is exemplified by the problem of appointment scheduling. COSMA agents (COoperative Schedule Management Agents) are designed to act as personal assistants to maintain their user's calendar. They are provided with competence for negotiating about scheduling appointments with other COSMAs. COSMA is an ongoing research project and several features are not yet fully developed. Therefore, throughout the whole paper we will also discuss future extensions to the system.

1 Introduction

Distributed systems and the metaphor of a society of cooperating autonomous agents have become more and more attractive in AI in recent years and has led to the development of DAI. In addition to classical AI which relies on intelligence based on representation and use of knowledge, this approach explores the role of interactional intelligence, i.e. the aspects of intelligent behavior that emerge from the interaction of individual agents with local perception, knowledge, beliefs, goals, plans and so on (cf. [BG88], [GH89], [WD92]). Problems arising when designing a multi-agent system (MAS) involve, among others, communication, conflict resolution methods, negotiation leading to mutual committed decisions, coordination of plans, epistemic knowledge and appropriate representations of all these.

COSMA[1] (COoperative Schedule Management Agents) deals with the application domain of appointment scheduling to approach these problems and develop methods for the design of multi-agent systems in general. The problem of making appointments with other people is an everyday problem for most of us. Usually an initiator suggests one or more possible time intervals for a meeting to a set of invitees. They in turn must check their calendars for an interval that fits best and they inform the initiator of their results. In most cases, especially if more than two invitees are involved and their calendars are already rather dense, it is not immediately possible to obtain an appointment date that is convenient for all. Rather, it is necessary for the participants to first negotiate about proposals and counterproposals to arrive at a mutually acceptable compromise which reconciles the local preferences of the different agents. The problem is an instance of decision making with several participants with different preferences, but with incomplete information about the other's preferences.

Appointment scheduling is an excellent example of how interaction and cooperation among autonomous agents leads to mutually committed multi-agent plans and is, therefore, a good domain for research in distributed AI (cf. [SD92], [Lux92], [EE92], [Boc93]). The process of finding a solution involves communication and negotiation about scheduling proposals, mechanisms of cooperation for resolving possible conflicts and methods of taking decisions. The problem is inherently distributed because each agent maintains its local private calendar and even tries to hide information not relevant for a specific appointment under discussion. Therefore, any centralized approach to this problem would rely on

*This work was supported by the German Ministry of Research and Technology under grant ITW9104

[1] We will use this name for any single agent as well as for the COSMA system composed of several agents.

the unnatural assumption that all users are willing to give unlimited access to their private data. Furthermore, with a decentralized system we avoid the problem of having a single bottleneck and attain more reliability in the case of failure of a component. Though appointment scheduling is a real world problem, it additionally has the advantage of being relatively simple to implement, because any interaction that occurs is based on communication. Therefore, it is ideal for designing, testing and comparing various strategies for decision taking and conflict resolution. The domain can be modelled at increasing levels of complexity. In the current version of COSMA, we restrict the goal of negotiation to reach an agreement about the starting-time of a meeting, but the duration, the place, the topic or the list of participants could also be a matter of discussion in future versions.

Our approach designs the solution for the appointment scheduling problem as a multi-agent system of several COSMAs, where each COSMA acts as the personal assistant for its user which maintains his dairy and has the means and competence to negotiate about the time of meetings. While the definitive final decision about a meeting is still taken by the human user, COSMA relieves its user of the process of negotiation. In addition, the user can hand over decision competence to COSMA by installing appropriate default behaviors for COSMA e.g. to handle the case of temporary personal absence.

Once one of the users specifies an appointment, his COSMA initiates a negotiation by sending invitations to the other participants' COSMAs. The negotiation is guided by the decisions the agents make based upon their time-preferences and the priority of the meeting under discussion. Thereby, each agent computes a utility which allows it to compare different proposals and to compare its utility with the utilities that other agents assigned to proposals. A mechanism based on the comparison of utilities is introduced to represent the concession an agent makes when joining a compromise. This allows the agents to maintain a history in terms of the concessions made in previous negotiations and to use this to influence their actual decisions. This mechanism is designed to model typical human behavior to break symmetry in deadlocked negotiations according to the scheme: "I'll make a concession this time if you make a concession next time in a similar situation".

In the following, we will give an overview of the different interaction levels and their relations in our application scenario with a special emphasis on the decisions taken by the agents. A more detailed description of the COSMA system can be found in [SS93].

2 Interaction

Interaction of COSMAs will be considered according to the following five components (cf. Figure 1):

- The **message** layer defines format and types for all possible communicational actions. Message delivery services are assumed to be provided by the basic system.
- Using these messages, the **protocol** specifies which possible sequences of messages are allowed and sensible. We say that the communicational context determines via the protocol which messages are possible in a certain situation. The protocol still leaves open the decisions within the actual negotiation (e.g. accept or reject a proposal).
- Therefore, besides the messages and protocol we need evaluation mechanisms which provide **decision criteria** for ranking and comparing different scheduling proposals. They are, of course, extremely dependent upon the application domain. For appointment scheduling, we use a simple model of utility based on an appointment's importance and a user's preferences for specific times.
- The application of these criteria at certain decision points defined by the framework of the protocol is what defines an agent's **negotiation strategy**.
- With the abstract level of **cooperation** we denote those results of an interaction that can only be described as emergent effect from the behavior of all agents within a society. (E.g. a traffic jam cannot be described by considering any single car.)[2]

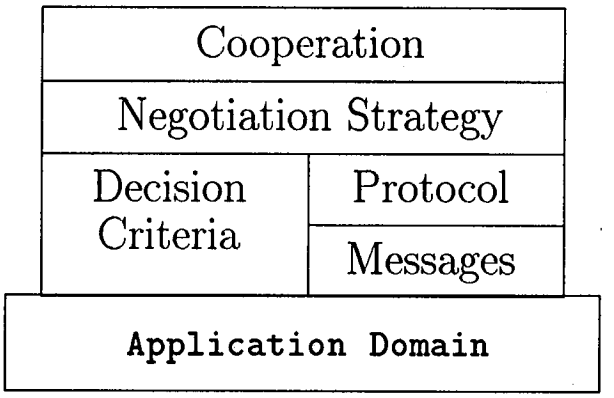

Figure 1: Ingredients of Interaction on different Levels of Abstraction

3 Application Domain

The application domain builds the basis for every MAS. Appointment scheduling is a very convenient application domain because interaction among the agents is based on communication. There are no physical actions to be done by the COSMA, neither in the real world nor in a simulated environment. The only interfaces to COSMA are a device for sending and receiving messages from other COSMAs on the one side and an interface to get commands from and display information to the user on the other side.

[2]This example also shows that, in general, we do not suppose a benevolence behind the term cooperation. We just take it as if it would abbreviate "*co*ncurrent *operation*". Nevertheless, in the context of appointment scheduling via COSMAs we want to achieve "*co*ordinated *operation*" with a positive intention.

The domain can be modelled at different levels of complexity. For the current version the following knowledge and parameters are relevant:

- *Participants* : For each meeting a list of participants has to be specified. This is done by the initiator, i.e. the user that initially creates the meeting. (The initiator is not necessarily a participant of a meeting.)
- *Time* : Time is the most central issue for COSMA. Time expressions are used to describe the duration of a meeting and to specify the time a meeting starts or is proposed to start. Time expressions must be capable of representing exact time points, time intervals and conjunctions of intervals. Defaults can be used to assign exact durations to different types of meetings (like e.g. class → 45 min., colloquium → 20 min., talk → 2 hrs. ...) or to decode higher level specifications like "tomorrow morning" or "next week". A resolution of under-specified expressions like "as soon as possible" is also possible.
- *Preferences* : Associated with times are a user's preferences for accepting a meeting as described in Section 6.
- *Priority* : A priority is specified by the initiator for each meeting. It encodes the importance assigned to the meeting and is used to obtain a user's utility of a scheduling proposal by comparing it with the preference.
- *Topic* : The topic is provided to inform the user and might influence his assessment of the importance of a meeting.

Additionally, the following features are under consideration for the enhanced version of COSMA currently under development:

- *Resources* : Special devices such as blackboards, overhead projectors, microphones, etc. or rooms can be modeled as passive participants of a meeting represented by their own modified instances of COSMAs. Of course passive COSMAs will need a slightly different functionality and interface.
- *Place* : Besides the notion of a room as a special resource (see above), the place of a meeting may also put additional time constraints on a negotiation. If we assume the participants of a meeting reside at distant locations, some of them may have to allocate appropriate travelling times before and after the meeting in their calendars.
- *Authority* : Usually there are hierarchical structures between the employees of a company which could be modelled by assigning different authorities to the participants. These may be used to bias negotiations towards the preferences of e.g. the project leader.
- *Roles* : Depending on the type of a meeting, different roles may be associated temporarily with different participants (e.g. lecturer, assistant, listener, moderator, secretary, ...). Specialized protocols may be necessary to handle the demands of different roles.

4 Messages

According to the central idea of speech act theory (cf. e.g. [App85]) we view communicational actions as planned actions. In general, planning and communication are tightly entangled: Plans contain communicational acts and communicational acts influence an agent's plans. In COSMA all actions are communicational acts and, therefore, we don't have any problems with the integration of planning and communication here.

For COSMA we designed a communication protocol using basically ten[3] different message types whose intended meaning is sketched in the following:

ARRANGE:	To initialize a negotiation an agent sends this message containing a specification of the priority of the meeting, the list of participants and a specification of suggested start-times to all participants.
REFINE:	As an answer to a proposal this message specifies a subset of the originally proposed time-specification.
MODIFY:	As an answer to a proposal this message rejects the proposed start-times and proposes an alternative start-time.
ACCEPT:	Indicates agreement to a proposed time-specification.
REJECT:	Irrevocable refusal to meet the appointment. This message finishes the negotiation unsuccessfully.
NEGOTIATE:	Special message for conflict resolution. When the participants have refined a proposed time-specification in a mutually incompatible way, the initiator tries to negotiate about the different possible subsets of the proposal. While the refine messages only contain information about their prefered interval, the initiator now demands an exact evaluation of the ranking for specific time intervals. NEGOTIATE provides information exchange for this negotiation.
COND-ACCEPT:	Conditional agreement to a proposal. An agent may conditionally agree to a time proposal depending on the ranking that other agents gave to that proposal. But if the knowledge about their rankings is not available, the agent replies with a COND-ACCEPT containing its decision threshold. The initiator uses

[3] Additionally, there are some technical message types between the COSMA and the graphical user interface which are not relevant for the description of COSMA's negotiation capabilities.

that threshold to evaluate the decision when the knowledge about the other's rankings is complete.

CONFIRM: This message finishes the negotiation and fixes a specific start-time for the meeting when an agreement of all participants is reached.

CHANGE: This message cancels the date of an already arranged appointment and makes a proposal to fix it into another interval. It reinitiates an already finished negotiation.

CANCEL: This message cancels an already arranged appointment. There is no alternative date for the specified meeting because the sender refuses to be a participant in that meeting at all. (The Initiator of the meeting has to decide if he is going to arrange a modified meeting with a modified list of participants.)

For the future development of COSMA's message layer we plan a closer adaption to the conceptual model of cooperation presented in [LBS92]. This is based on cooperation objects (i.e. a unit of work or matter of cooperation), cooperation primitives, which correspond to the message-types, and cooperation methods that can be seen as procedures prescribing how the agents can efficiently conduct a cooperation. COSMA can be seen as an application of this conceptual model designed for a specific domain and using additional special messages. We envisage two ways to reach a better adaption: Firstly, it should be possible to implement special messages as cooperation methods composed of several cooperation primitives. Secondly, it may be promising to introduce a hierarchically structured typology of message types with general primitives on top and specialized instances used in specific applications at the bottom.

5 Protocol

The protocol specifies the possible communicational interactions between the agents and, thus, links the messages to communicational contexts. We think of the protocol as a set of possible actions within a given communicational context. E.g. an answer is usually given in the context of a question.

A central part of the basic (top-level) protocol is shown in Figure 2 which represents internal actions and message exchange from the local view of a single agent. Nodes represent internal actions or states of agents or, as in the case of the "negotiate" node, represent subgraphs introduced for clarity and modularity. Arcs represent messages received by an agent, state transitions or simply different outcomes from subgraphs. The role of the agent as initiator or other participant is determined at the beginning of a negotiation from the origin of the first message (user interface or other COSMA). The "negotiate" node encapsulates a sub-protocol that attempts to resolve a conflict arising from incompatible refine messages.

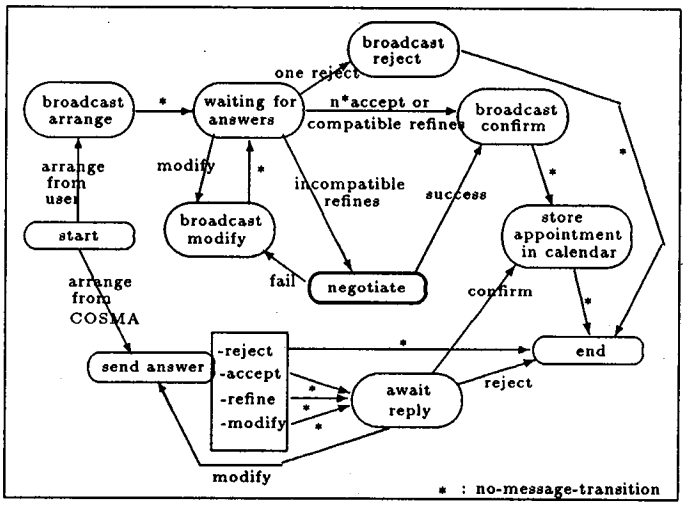

Figure 2: COSMA Negotiation Protocol

The basic idea of the protocol in COSMA is simple: A negotiation starts with sending the arrange-message of the initiator to all participants of the meeting. It specifies the priority of the meeting, the list of participants and a first time proposal. The answers to this message could fall into four categories:

- At least one of the participants rejects. In this case, the agent has to send a rejection of the meeting to all other participants, which terminates the negotiation. (We leave it to the user to decide, if he wants to rearrange that meeting with a modified list of participants.)
- Some participants reply with the suggestion to modify the time proposal. In this case, COSMA has to broadcast a modified time proposal to all participants, which it has to compute based on local preferences and the modify-proposals.
- All participants accept the proposal or refine the proposed time in such a way that the different refinements overlap in one acceptable time-interval. The COSMA has to fix a specific time point for the beginning of the meeting within that interval and has to broadcast this with a confirm-message to the participants.
- All participants agree in principle to the proposed time interval, but refine this in a mutually incompatible way, i.e. there is no sufficient overlap for the different refinements. This case could, in principal, be handled like the second one (i.e. propose some other interval) but our protocol is designed to negotiate about the different subintervals of the proposal to reveal the criteria on which the agent's replies are based, and possibly resolve the conflict.

The last case causes a negotiation in which conflict resolution is tried by the initiator. Therefore, the initiator splits the interval under discussion into possible subintervals. It orders these subintervals according to its local

preferences and sequentially works on this list of proposals. For each entry it asks the other agents to evaluate their ranking of this proposal. This is done by using the message type NEGOTIATE. The answers to this message are of type REJECT or ACCEPT or COND-ACCEPT. The decisions of the participants are based on their preferences and on the history of negotiations as will be described in the next two sections.

The cases of reinitiation of a negotiation and calling off an appointment by CHANGE and CANCEL, respectively, are not shown in the figure. In the current version a reinitiation is handled in the same manner as an initial arrangement, i.e. information about the participants time preferences must be collected again. This is justified by the assumption that the participants' calendars may have changed since the first negotiation. Assuming that a user assigned maximal autonomy to his COSMA, a meeting with a high priority may also displace a lower priority appointment, which then has to be rescheduled by a reinitialized negotiation. In the case of dense calendars this may cause rippling effects. On the other hand, we did not reinitiate negotiations of already scheduled meetings when caused by the cancellation of another meeting a more appropriate time-interval becomes available.

In general, the intention of the protocol is to capture the notion of a context in a negotiation. By specifying all sensible sequences of messages it builds the frame for interaction. On the other hand, it installs an implicit model of the other agents on the assumption that they follow the same protocol in a negotiation. We are currently looking for mechanisms to provide more flexibility on the protocol level. Different protocols appropriate for different situations or roles of participants will be developed. The fragmentation into sub-protocols that may be used from different top level protocols is a first approach in this direction.

6 Time File

The time file represents the availability of a user's time. Time representation is based on exact time-points specified by the tuple [*year, day, hour, minute*]. An interval is given by a pair of time-points. We define a preferenced interval to be an interval together with an associated preference $x \in [0, 1]$. By a time file we mean a contiguous sequence of preferenced intervals. Formally, a time file can be seen as a stepfunction with values in $[0, 1]$. The meaning of a preference is to reflect the planned allocation of activities to a time interval, i.e. a preference of 1 says that this interval is already allocated for some maximally important activity while a preference of 0 is associated with an available interval. Values between 0 and 1 express all possible intermediate degrees of importance. The availability of time depends on several factors. In the current implementation we consider four sources of time-preferences:

- Common knowledge about general preferences. For example, most people do not want to work at night or will have a very low preference for working during the lunch time. The common knowledge is provided by a common knowledge time file which can be predefined or generated and maintained by constantly recording and averaging the meetings over a representative period of time. This file should be the same for all agents.
- User-specific preferences express the users' personal preferences to allocate meetings. It is represented as a user-specific time file and represents any regular preferences like e.g. an aversion to schedule meetings before ten o'clock in the morning, as well as any other time consuming activities, that are not associated with meetings. The user interface has to provide the functionality to specify these preferences.
- Calendar dates represent the allocation of time for already arranged meetings. They are also represented as a time file where the preference for a time equals the priority of the meeting scheduled during that time. To prevent too much rescheduling, the priority of scheduled meetings may be lifted up slightly. The source of this information is, of course, the user's calendar itself.
- The action memory, which is also represented as a time file, contains temporary information about the time proposals an agent is making during a negotiation. This is used to coordinate concurrent negotiations. If, in one negotiation a COSMA proposes a specific time interval I for a meeting and is asked in the context of another negotiation about another meeting if a time within I would be acceptable, it should deny this (supposing the first meeting is of higher priority). The action memory thus temporarily blocks the intervals of current proposals but has to release them as soon as a proposal is out of date. The exact mechanism of coordinating parallel negotiations is another parameter for designing and testing different negotiation strategies in future research.

These different sources are combined into a single time file which is at last used to compute the utility that the agent assigns to a proposal. User-specific preferences and the common knowledge time file are combined by averaging the preferences of each time-point. Thus neither of these sources dominates the other. The other two sources are merged into this time file by taking the maximum of the preferences for each time-point, because a scheduled meeting or a blocked interval should dominate the specific and common preferences.

7 Decision = Utility + Concession

7.1 Utility

Decisions in COSMA are based on the utility that an agent assigns to a proposal. The utility is computed by comparing the priority of the meeting under discussion with the preference of the interval that was proposed. Precisely, we define the utility of a proposal to be the difference of the priority of the meeting and the preference the agent assigns to the proposed interval relative to the duration:

Let p_m be the priority and d_m the duration of meeting m. If a proposal suggests starting m at time t_1 the agent assigns the utility ut to that proposal based on its current time file tf according to the formula

$$ut(m, tf, t_1) = \frac{p_m * d_m - \int_{t_1}^{t_1+d_m} tf(t)\,dt}{d_m}$$

The integral reduces to a sum because we assume tf to be a stepfunction. If a proposal specifies a set of several start-points, as is usually the case, the agent bases its decision on the maximal utility possible according to its time file. However, if the agent accepts the proposal it may happen that the meeting will be fixed at a time with a lower utility for the agent than that maximum. To avoid this, a more cautious policy for the agent would be to reply with a refine message, which restricts the possibilities to those with maximal utility. On the other hand, such a cautious policy may reduce the chance of finding a time suitable for all participants or at least increase the duration of negotiations. This discussion shows the influence of different negotiation strategies.

As a first approach towards testing different strategies we distinguish three different usages of the utility:

- *Absolute decision*: In an absolute decision the agent agrees to any scheduling proposal that has positive utility.
- *Relative decision*: In a relative decision the agent agrees if its utility is greater than the average of the utility of all participants of that meeting.
- *Concessive decision*: In a concessive decision the agent decreases its utility by subtracting from it the degree of concession and agrees to a proposal if the value obtained is greater than the average utility of all participants of the meeting.

The relative decision is obviously a little bit whimsical because not every agent can gain greater utility than the average utility-gain[4]. We considered this decision strategy only as a basis for the concessive decision, in which the degree of concession is used.

7.2 Concession

The degree of concession was introduced into COSMA to model typical human behavior for breaking symmetry in deadlocked situations: "I'll make a concession this time if you make a concession next time in a similar situation". The degree of concession encodes information about the utilities of previous meetings and, thus, has the function of a memory. In analogy to game theory we could interpret it as the introduction of the concept of an iterated game.

A large degree of concession means that an agent has made many concessions in previous negotiations. To compute the degree of concession, the agents have to maintain a simple partner model to be aware of what has happened in past negotiations. An agent that has gained relatively high utility out of the meetings in the past should make more concessions in future negotiations, i.e. should agree to proposals even if it gains less utility compared to the other participants. (The degree of concession C is negative $\Rightarrow$ The agent has to make more concessions in future.) On the other hand an agent that has gained little utility out of past negotiations should run a more uncompromising negotiation strategy. (The degree of concession C is positive $\Rightarrow$ The agent should try to gain more utility from meetings in future.) These rules, supposing all participants follow the strategy of concessive decisions, should have the effect of distributing utilities fairly among the agents over a history of negotiations. An agent that tries to exploit this strategy by constantly making no concessions will soon face very uncooperative behavior from the other agents. Scheduling meetings with that agent in the list of participants will become more and more difficult.

The degree of concession is **used during** negotiation to support the decision and **updated after** an agreement is reached for usage in future negotiations.

Concerning the update, an agent has to maintain, for each partner X it ever met in a negotiation, two values: X's utility U_x and its own utility U_x^o in negotiations with X. After each successfully finished negotiation, in which X was also a participant, these values are updated by weighted averaging, where $w_1 + w_2 = 1$ holds:

$$U_x \leftarrow w_1 * U_x + w_2 * ut_x$$

$$U_x^o \leftarrow w_1 * U_x^o + w_2 * ut^o$$

Here ut_x stands for the utility of X and ut^o for the agents' own utility in the current negotiation. With this simple mechanism of averaging we need not store the whole history of negotiations and besides gain the effect that more recent utilities have more influence than older ones. The actual weighting of recent utilities versus older ones is determined by the values of w_1 and w_2. But even if w_2 is chosen, to be very small, the influence of each single older utility dies out gradually with every new negotiation. This effect is also reasonable, as in human cooperation the argument "You have to make a concession, because I did so last week" is more persuasive than "You have to make a concession, because I was concessive two years ago".

Concerning the usage for a decision within the actual negotiation the degree of concession is computed by the difference between the average utility and the agent's own utility. The following example assumes there are three other agents, X, Y and Z in the list of participants of the currently negotiated meeting. The degree of concession C is then computed as follows:

$$C = U_{av} - U_o$$

$$\text{where} \quad U_{av} = \frac{U_o + U_x + U_y + U_z}{4}$$

$$\text{and} \quad U_o = \frac{U_x^o + U_y^o + U_z^o}{3}$$

[4] In the current implemented protocol, the initiator is the one who will gain less utility than the average, if the meeting can be scheduled at all.

8 Conclusion and Outlook

The design of the interaction of the COSMAs involved several aspects: Starting from the selection of an appropriate set of messages and the specification of meaningful sequences of messages via a protocol, we developed a time model based on preferences that supports ranking and comparison of proposals to guide the decisions to be made in the negotiation. The effect of the local behavior of the agents within a negotiation is cooperative allocation of appointments within the calendars of the participants. The cooperation of COSMAs results in the distributed local calendars. These are globally consistent in the sense that all participants of successfully negotiated meetings will have an entry for that meeting in their calendar at the same time interval. Furthermore, with the degree of concession and its usage, we modelled a mechanism to distribute utility fairly among the agents. This resembles typical human behavior. When someone makes concessions in a negotiation such that a compromise can be found, then the next time it should be someone other's turn to be indulgent. The major problem with this is a commonly accepted measurement of "concession" if we want to apply this mechanism in more complex domains. Furthermore, our agents implicitly rely on the assumption that all other agents behave in the same manner. In contrast to this, it would be desirable that the agents explicitly refer to and mutually agree to the cooperation mechanism of making timely distributed concessions, i.e. the cooperation mechanism itself should be an explicit matter in the discussion of the agents.

All our experiments succeeded in finally finding an acceptable time slot except when one of the participants rejects or cancels the meeting. This is no surprise, because our protocol is designed in a way to produce a theoretically infinite sequence of proposals that place the meeting further and further in the future. There is still no deadline for a meeting. Also, we did not give criteria that quantitatively describe the quality of one scheduling solution compared to another. Our intention in this study was not to design an optimal distributed algorithm solving the scheduling appointment problem, but to investigate the negotiation and the special case of conflict resolution approach to decision making. We think it is even doubtful if the definition of an optimality-criterion in this dynamically changing environment makes sense. For future development we will introduce the concept of deadlines and some notion of resources (like meeting rooms). This would make the application domain more complex and extend possible applications to the field of general scheduling problems.

Acknowledgement

We would like to thank all members of the research groups AKA-MOD and KIK-Teamware at DFKI for fruitful discussions and comments on the paper. Special thank is given to Pierre Sablayrolles, who did most of the implementational work.

References

[App85] D.E. Appelt. *Planning English Sentences.* Cambridge University Press, 1985.

[BG88] A. Bond and L. Gasser. *Readings in Distributed Artificial Intelligence.* Morgan Kaufmann, Los Angeles, CA, 1988.

[Boc93] S. Bocionek. CAP II & RAP: Lernfähige, verhandelnde Agenten für die Büroautomatisierung. In J. Müller, editor, *Beiträge zum Gründungsworkshop der Fachgruppe Verteilte Künstliche Intelligenz*, pages 161 – 174. DFKI-Document, D-93-06, Saarbrücken, April 1993.

[EE92] N. Eisinger and N. Elshiewy. Madman - multi-agent diary manager. internal report ECRC-92-7i, ECRC, München, 1992.

[GH89] L. Gasser and M.N. Huhns. *Distributed Artificial Intelligence, Volume II.* Research Notes in Artificial Intelligence. Morgan Kaufmann, San Mateo, CA, 1989.

[LBS92] A. Lux, F. Bomarius, and D. Steiner. A model for supporting human computer cooperation. In *AAAI Workshop on Cooperation among Heterogeneous Intelligent Systems*, July 1992.

[Lux92] A. Lux. A multi-agent approach towards group scheduling. Research Report RR-92-41, German Research Center for AI (DFKI), Saarbrücken, August 1992.

[SD92] S. Sen and E.H. Durfee. A formal analysis of communication and commitment in distributed meeting scheduling. In *11th Int. Workshop on DAI*, pages 333 – 344, Homestead, Glenn Arbor, Feb. 1992.

[SS93] P. Sablayrolles and A. Schupeta. Conflict Resolving Negotiation for COoperative Schedule Management Agents (COSMA). Technical Report TM-93-02, DFKI Saarbücken, May 1993.

[WD92] E. Werner and Y. Demazeau, editors. *Decentralized A.I. 3.* North-Holland, 1992.