Multi-Agent Meeting Scheduling: Preliminary Experimental Results

Leonardo Garrido
Center for Artificial Intelligence, ITESM
Sucursal de Correos "J"
Monterrey, N.L. 64849 México
lgarrido@campus.mty.itesm.mx

Katia Sycara
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213 USA
katia@cs.cmu.edu

Abstract

We view meeting scheduling as a distributed task where each agent knows its user's preferences and calendar availability in order to act on behalf of its user. Although we may have some intuitions about how some parameters could affect the meeting scheduling efficiency and meeting quality, we run several experiments in order to explore the tradeoffs between different parameters. Our experiments show how the calendar and preference privacy affect the efficiency and the meeting joint quality under different experimental scenarios. The results show how the meeting scheduling performance is more stable and constant when agents try to keep their calendar and preference information private. We believe that these parameters play a key role in the distributed meeting scheduling task, specially if we are interested in building distributed systems with truly autonomous and independent agents where there is not a fixed control agent.

Introduction

In our daily life, meeting scheduling is a naturally distributed task which is time-consuming, iterative, and somewhat tedious. It can take place between two persons or among several persons. Sometimes, people just try to schedule one meeting. However, most of the time people need to schedule many meetings at the same time taking into account several constraints.

Each potential attendee needs to take into account his/her own meeting preferences and calendar availability. Most of the time, each attendee has some uncertain and incomplete knowledge about the preferences and calendar of the other attendees; in fact, people usually try to keep their calendar and preference information private. During the meeting scheduling process, all attendees should consider the main group goal (i.e. to schedule a meeting) but they also take into account individual goals (i.e. to satisfy their individual preferences).

There exist several commercial products but they are just computational calendars with some special features (e.g. availability checkers, meeting reminders); in (Taub 1993), a review of several of these products can be found. None of these products is a truly autonomous agent capable of communicating and negotiating with other agents in order to schedule meetings in a distributed way taking into account the user's preferences and calendar availability.

However, there has been much research work in meeting scheduling. Most of the earliest work reached interesting but limited success (Kelley & Chapanis 1982; Greif 1982; Kincaid, Dupont, & Kaye 1985).

We can find some interesting approaches in Artificial Intelligence such as (Mitchell *et al.* 1994) and (Maes 1994); these approaches focus in learning user preferences but they do not take much attention to the social and distributed implications of the distributed meeting scheduling process.

We can also find other investigations in Distributed Artificial Intelligence. A very interesting work in distributed meeting scheduling is (Sen & Durfee 1991; 1992; 1993; 1994; 1995); this work has been focused on solving the meeting scheduling problem using a central host agent capable of communicating with all other agents in order to schedule meetings using a negotiation based on contracts (Smith 1980); the main purpose of the host agent is to coordinate the search for a feasible schedule taking into consideration attendees' calendars. However, user preferences are not taken into account during the meeting scheduling process. They have focused their research on several search biases to get different density profiles in agents' calendars.

Another recent work in distributed meeting scheduling is (Ephrati, Zlotkin, & Rosenschein 1994). They presented an alternative approach which is economic in flavor. Using three centralized monetary-based meeting scheduling systems, they analyzed tradeoffs between the mechanism complexity and information preferences and they introduced the Clarke Tax Mechanism

as a method for removing manipulability from them.

In (Sycara & Liu 1994), it is presented another approach based on modeling and communication of constraints and preferences among the agents. Here the agents are capable of negotiating and relaxing their constraints in order to find and reach agreements on schedules with high joint utility. Using this model, agents also can react and revise the schedule in response to dynamic changes.

In our work, we view meeting scheduling as a distributed task where each agent knows its user preferences and calendar availability in order to act on behalf of its user. We implemented our system which consists of truly autonomous software agents running, as independent processes, on different computers and we have run several experiments; in this paper, we present some of our preliminary experimental results.

Multi-Agent Meeting Scheduling System

Our distributed meeting scheduling system does not have a fixed central control. This means that there is not a specialized control agent and each agent is able to try to schedule a meeting via negotiation taking into account individual preferences and calendar availability under a dynamic protocol and coordination mechanism.

Since we are primary interested in investigating the behavior of truly autonomous and independent agents, one of our main concerns is information privacy. In our system, each agent knows only its user meeting preferences and calendar information. However, agents can exchange information during negotiation.

A meeting has a date, start-time and duration and it is scheduled when all agents reach an agreement on values for these attributes. Agents negotiate values for theses attributes taking into account three *time windows* given as input:

Date window. It indicates the range of days where the meeting can be scheduled.

Start-time window. It indicates the range of start-times to schedule the meeting.

Duration window. It indicates the range of durations where the meeting can be scheduled.

Each agent has a set of meeting preferences given as inputs for each particular meeting; this set represents its user's meeting preferences (i.e. the three meeting attributes values that are the most preferred by its user). Here it is interesting to note that there are other approaches (Mitchell *et al.* 1994) which focus on learning user preferences using induction trees and taking into consideration several meeting attributes. Eventually, our agents could have communication with one of these systems in order to get the set of user meeting preferences.

The main goal is to schedule meetings considering the three time windows given as inputs. However, each agent has its own individual goal: to schedule the meeting maximizing its individual meeting preferences (i.e. the agents try to schedule the meeting in the calendar interval with the closest attributes to its user meeting preferences).

In order to schedule meetings, agents face two problems: one is that the agents can have different available calendar intervals and the other one is that they can also have different individual meeting preferences. Furthermore, the distributed environment and the information privacy are, of course, sources of other aspects that need to be taken into account.

In our system, the agents are able to negotiate proposing and bidding values for the three meeting attributes (date, start-time and duration) according to their individual preferences and available time intervals. In our distributed system, agents can exchange their meeting preferences and calendar information according to some privacy policy which has been an adjustable parameter in our experiments. Basically, each agent is able to relax three different time constraints: date, start-time and duration. In addition to its preferences, each agent has weights (values between 0 and 1) that indicate how to relax each time constraint.

We implemented a simplified version of the coordination mechanism and communication protocol presented earlier in (Sycara & Liu 1994). In our experiments, agents communicate and negotiate via message passing. Each agent is able to relax its preferences when conflicts arise. The protocol we implemented is as follows:

1. All the agents in the group are randomly enumerated.

2. An agent is selected randomly; this agent becomes the first task coordinator who is responsible for broadcasting the first proposal.

3. Each agent that receives a proposal accepts or rejects it by replying the message. If the agent accepts it, it may share the priority value that this agent has assigned to the accepted time interval.

4. When the current task coordinator receives all the replying messages and the proposal was accepted by all agents, the coordinator sums up the priorities to get the group utility measure for that meeting

and broadcasts a final confirmation message notifying that the last proposal has been accepted and its group utility value.

5. However, when the proposal is rejected by at least one agent, a new task coordinator is selected which is the following agent in the enumeration that opposes the current proposal. The new task coordinator relaxes its time constraints; it broadcasts a new proposal and the process is repeated from point 3.

The original coordination mechanism and communication protocol discussed in (Sycara & Liu 1994) takes into account multiple-meeting schedules and dynamic changes after a meeting have been approved (e.g. an agent can ask for further negotiation if it realizes that, due to dynamic local changes, it is possible to schedule the meeting again achieving higher utility).

During negotiation, agents look for free calendar intervals to be proposed. This search is biased using individual utility functions; the utility function gives a priority value to each available calendar interval according to the particular time windows, individual meeting preferences and relaxation weights. This gives us a search mechanism that can be biased by different relaxation heuristics.

Let us define a *calendar interval* j as a vector, $\vec{I}^j$, with three attributes: date, start-time and duration. Also, let us define the *preferences of the agent* k as a vector, $\vec{P}^k$, with the same three attributes. Furthermore, let us define the *relaxation weights of the agent* k (the weights assigned by agent k for relaxing each of the three time constraints) also as a vector, $\vec{W}^k$, with the same three attributes.

Now, we can define the *weighted distance* ($WDist$) between an *interval* j and the *preferences of the agent* k, taking into account its *relaxation weights*, as:

$$WDist(\vec{I}^j, \vec{P}^k, \vec{W}^k) = \sum_{i=1}^{3} [\vec{W}_i^k \cdot Dist(\vec{I}_i^j, \vec{P}_i^k)]$$

Here, *The subscript* i indicates the meeting attribute (1 is *date*, 2 is *start-time* and 3 is *duration*). $Dist(\vec{I}_i^j, \vec{P}_i^k)$ is the distance between $\vec{I}_i^j$ and $\vec{P}_i^k$ (i.e. the number of possible different instances of the attribute i between the attribute value of the interval j and the attribute value that is most preferred by the agent k).

Finally, we can present the *General Utility Function* we used in our experiments:

$$Priority_k(\vec{I}^j) = \frac{\sum_{i=1}^{3} [\vec{S}_i - 1] - WDist(\vec{I}^j, \vec{P}^k, \vec{W}^k)}{\sum_{i=1}^{3} [\vec{S}_i - 1]}$$

As we can see, this function is just the *normalized* weighted distance taking into account the *window sizes* which are seen as a vector,$\vec{S}$, with the three time attributes. In other words, $\vec{S}_i$ is the window size of the meeting attribute i (i.e. the number of possible different instances of the attribute i in the time window given as input). Using this function, each agent k can assign a priority value to each available time interval j. The maximum possible priority value is 1 and the minimum possible value is 0. The calendar interval with the highest value is the best option to propose.

Experiments Description

In the experiments we present here, we consider the negotiation/scheduling process between three agents. We have identified some experimental variables and we have been investigating the effect of each parameter in isolation from the other.

In order to avoid intractable combinatorics, we considered calendars of three days with three hours per day. Also, we used a *relaxation step* of 30 minutes (i.e. any possible calendar interval will have a duration and start-time multiple of 30 minutes).

So let us say that our calendar days are 1, 2, and 3. The start-times are 0, 30, 60, 90, 120, and 150. The possible durations are 30, 60, 90, 120, and 150. All these values define our three time windows discussed earlier: the *date window*, *start-time window*, and *duration window*.

The number of busy hours in a calendar may vary. Let us define the *calendar density* for a particular calendar as the number of busy hours (without fractions) in that calendar. In our experiments, we vary the calendar density from zero to nine and we run each different experimental meeting scheduling scenario for each different possible calendar under each different calendar density.

We repeated each experimental scenario under every different calendar using each different calendar density with one agent while the rest of the agents had empty calendars. We calculated the average efficiency and average joint quality when the last run of each calendar density is accomplished.

We measured the efficiency in terms of the number of proposals and we measured the meeting quality as a joint quality, using the following formula:

$$JointQuality(\vec{I}^j) = \frac{\sum_{i=1}^{n} Priority_i(\vec{I}^j) \cdot 10}{n}$$

In this formula, the maximum possible value is 10 and the minimum one is 0. The variable n is the number of agents.

In this way, we have obtained two graphs for each particular experimental scenario: the first one is an efficiency graph with the number of proposals and calendar density as axes; the second one is the meeting group quality graph with the joint quality and calendar density as axes.

As we said before, it is highly desirable to keep the agents' information private. Let us see how we varied the amount of information exchanged:

Calendar Information. Basically, we experimented with two kinds of calendar information exchanges: *total calendars* and *partial calendars*. In the former case, agents exchange all their available calendar intervals. In the latter case, agents exchange a portion of their available calendars. Furthermore, in this case we can have other cases varying the size of the exchanged calendars (e.g. proposing only two slots instead of four slots).

Preferences. We have here two kinds of preference information exchanges: *public preferences* and *private preferences*. In the former case, agents inform the preference value of each calendar interval that they propose. In the latter case, agents inform their preference values after they have reach an agreement on a time interval; this is done in order to measure the joint quality of the meeting.

It is interesting to note that if the agents are exchanging *public preferences* they can use the joint quality formula discussed previously instead of using their individual utility functions. Furthermore, it is also interesting to note that if they are exchanging *public preferences* and *total calendars*, it is possible to get an agreement with the highest possible joint quality in only "one shut". This is possible because each agent knows all the information (i.e. there is no information privacy at all).

In the following paragraphs we present the experimental scenarios we set up:

Varying individual goals with total calendars. Here we varied the individual preference values and we plotted two curves. The first curve shows the meeting scheduling process when all agents have the same preferences; let us call it *Common-Goals* curve. The second one, called *Disparate-Goals* curve, shows the process when each agent has different preferences.

Here agents exchange *total calendars* but they do not maintain *public preferences*. Therefore, they are using their individual utility functions to guide the search and they do not exchange their preferences until they reach an agreement. After this, each agent has the opportunity of proposing another time interval and this cycle ends until no agent has new proposals.

Varying individual goals with partial calendars. In this scenario, we varied the preference values again. As we said in the previous experimental scenario, the first curve, *Common-Goals*, shows the meeting scheduling process when all agents have the same preference values. The second one, *Disparate-Goals*, shows the process when each agent has different preferences.

The difference between this scenario and the previous one is that this scenario was fixed to the exchange of *partial calendars* instead of *total calendars* and, as in the previous scenario, the agents maintain *private preferences*. So that they also use their individual utility functions and not the joint utility formula to guide the search.

Varying individual goals with public preferences. Here we varied again the agents' individual goals, plotting two curves. The first one, called *Common-Goals*, shows the process when all agents have the same preferences. The second one, called *Disparate-Goals*, shows the process when each agent has different preferences.

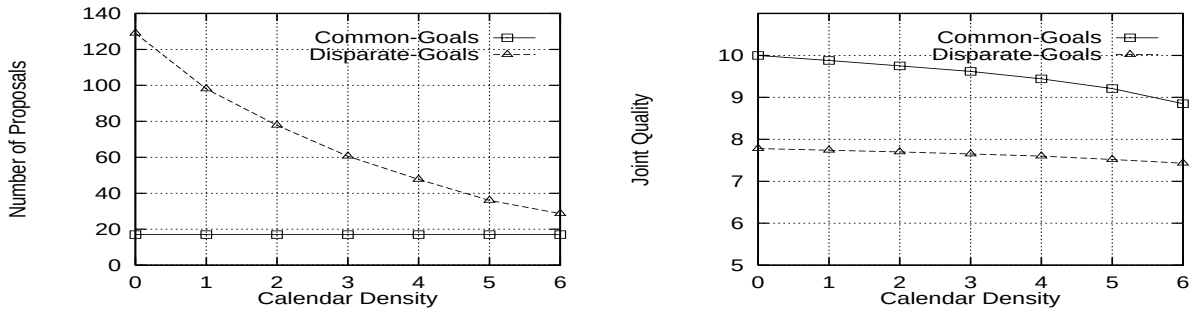
Here, agents exchange *total calendars* and *public preferences*. Since they are working under the *public preferences* scheme they also use the joint utility formula, discussed earlier, instead of their individual utility functions.

Varying preference privacy with total calendars. As we discussed earlier, we have basically two kinds of preferences exchanges: *public preferences* and *private preferences*. We plotted one curve for each of them and we call them *Public-Goals* and *Private-Goals* respectively.

All agents exchange *total calendars* in this experimental scenario. The agents' utility functions were fixed to *Disparate-Goals*. That is, every agent has a different set of preferences.

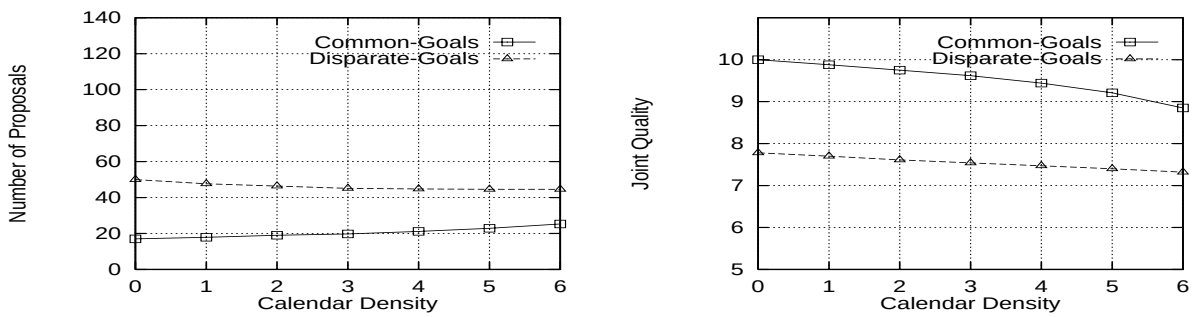
Varying preference privacy with partial calendars. As in the previous experimental scenario, we plotted two curves: one called *Public-Goals* and the other one *Private-Goals*.

We fixed now the calendar exchange scheme to *partial calendars*. As in the previous scenario, the agents' utility functions were fixed to *Disparate-Goals*.



(a) Efficiency (b) Quality

Figure 1: Varying individual goals with total calendars



(a) Efficiency (b) Quality

Figure 2: Varying individual goals with partial calendars

Experimental Results and Analyses

Figure 1 displays the results we obtained varying individual goals with *total calendars*. As we expected, when all agents have the same preferences (*Common-Goals* curve) the efficiency and meeting quality are, in general, better than when agents have different preferences (*Disparate-Goals* curve).

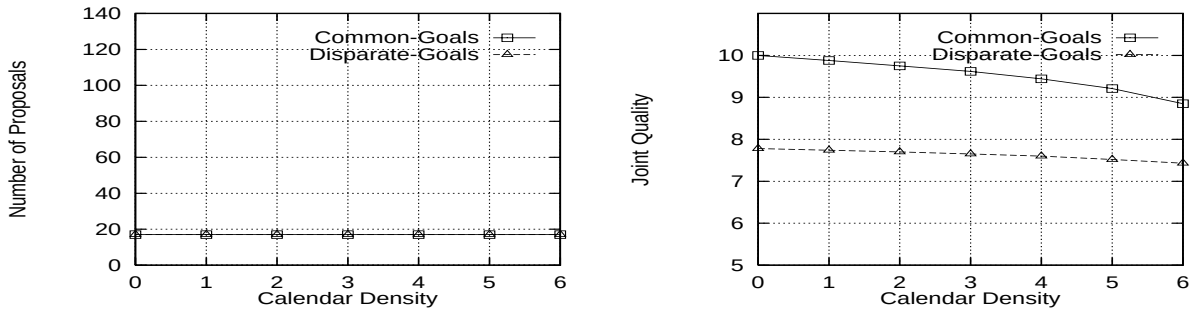
Also, we can see that the efficiency showed by the *Common-Goals* curve is constant and the meeting quality is decreasing when the calendar density is increasing. As we will see in the following graphs, this meeting quality is the best one that three agents can reach using our coordination mechanism and communication protocol.

On the other hand, the efficiency showed by the *Disparate-Goals* curve is not constant; in this case, the efficiency increases when the calendar density increases. However, this efficiency is always worse than that showed by the *Common-Goals* curve. The meeting quality of the *Disparate-Goals* is also decreasing but not as fast as the *Common-Goals*. However, the former always is lower than the latter.

Now, the results we obtained varying the individual preferences with *partial calendars* are shown in figure 2. Let us first note that, as in the previous case, the *Common-Goals* curve shows better results than the *Disparate-Goals* curve. We can also see that the meeting quality graph is very similar to the previous one. However, we can see now a really different efficiency graph.

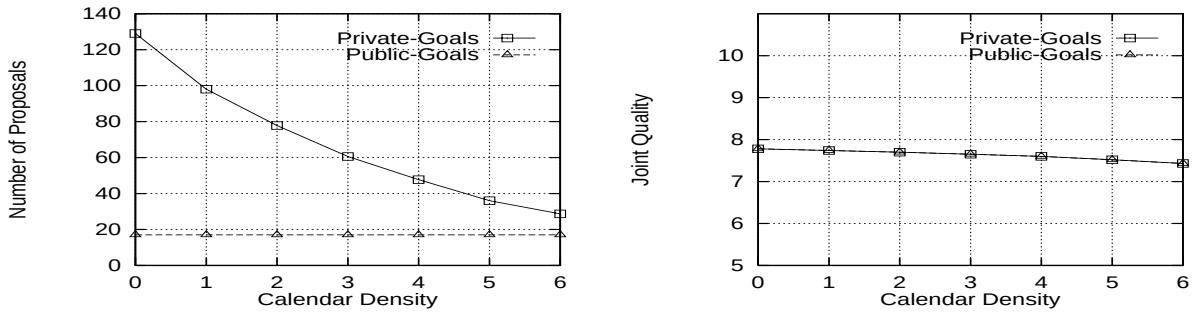
It is interesting to note the difference between this efficiency graph and the previous one. Now, the efficiency showed in figure 2 by the *Disparate-Goals* curve is improved when the calendar density is less or equal to 4. In fact, this efficiency remains almost constant under every calendar density. As we saw in the previous graph, it was not the case with the *Disparate-Goals* curve. On the other hand, the efficiency showed by the *Common-Goals* curve is increasing now, while it was constant in the previous graph. Fortunately, as we can see in figure 2, the increment is small and the curve does not raise very quickly.

We can explain these differences noting that, in general, agents explore less search space when they ex-



(a) Efficiency (b) Quality

Figure 3: Varying individual goals with public preferences



(a) Efficiency (b) Quality

Figure 4: Varying preference privacy with total calendars

change *partial calendars* than when they exchange *total calendars*. However, sometimes they need to make more iterations because the *partial calendars* does not have a common intersection, as we saw earlier in the *Common-Goals* curve.

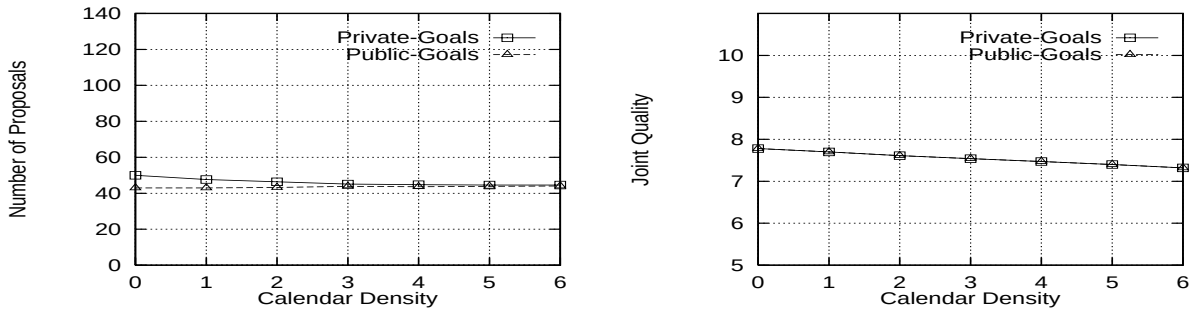
On the other hand, we can see that the joint quality in figure 1 is slightly better than that shown in figure 2. This can be explained noting that when agents explore *partial calendars* they do not evaluate all the possible calendar intervals. So that it is possible to reach agreements on meetings with joint qualities lower than those qualities reached with *total calendars*.

Now, figure 3 shows the results obtained when we varied the individual goals with *public preferences* and also with *total calendars*. As we can see, both *Disparate-Goals* and *Common-Goals* curves show exactly the same high and constant efficiency. Also, we can see that the meeting quality is the same as in the graph shown in figure 1.

This results are explained noting that in this experimental scenario there is not information privacy at all. Although agents have different individual goals the efficiency is as fast as when they have common goals; this is because they are exchanging *public preferences* and they are able to use the joint quality formula to guide the search through the exchanged *total calendars*.

Now, let us look at figure 4; this figure displays the results varying preference privacy with *total calendars*. Remember that in this experimental scenario agents have different individual goals. As we can see, we obtained now a very similar efficiency graph to that shown in figure 1. The difference is that we are varying now the preference privacy (remember that, in the experimental scenario shown in figure 1, agents exchanged *total calendars* and both curves showed the results when agents used different individual goals).

As we can see in figure 4, when agents exchange *total calendars* and they maintain *Public-Goals*, they get high and constant efficiency. This is because they are using the joint quality function to guide the search. However, if they exchange *private preferences* (look at *Private-Goals* curve), we can see that they spend more time because they use their individual utilities functions instead of the joint utility formula discussed



(a) Efficiency

(b) Quality

Figure 5: Varying preference privacy with partial calendars

earlier.

On the other hand, it is interesting to note that both curves have similar shape in the joint quality graph. This is because the agents have different goals and *total calendars* in both cases. Obviously, sooner or latter (depending on the preference privacy policy), they reach the same results. Of course, we are assuming that agents prefer to relax their constraints before failing to reach an agreement.

Figure 5 also shows our results when varying preference privacy but now with *partial calendars*. Again, both curves show agents with different individual goals. However, they exchange now *partial calendars*. As we expected, we obtained again similar meeting quality results for both curves.

However, we can see different results in the efficiency graph. We see that we can obtain almost equal efficiency with calendar densities greater or equal to three. In fact, both curves are, in general, very similar. They are almost constant with similar values. It is interesting to contrast this graph with the previous efficiency graph where both curves were very different. Meeting scheduling efficiency remains more constant varying preference privacy with *partial calendars* than when varying it with *total calendars*.

Discussion

First of all, we need to note that our work has a primary focus on social aspects of the meeting scheduling task. We are specially interested in aspects such as information privacy, negotiation, and group adaptation. However this does not mean that the inherent scheduling problem, which is known as an NP-complete problem (Garey & Johnson 1979), does not have enough importance to be considered.

In general, our expectations were confirmed by the experimental results. However, we have discovered some interesting results that we would like to discuss here.

We can expect that the efficiency decreases when the difference between individual goals increases. This is particularly true when agents exchange *total calendars*. However, when they exchange *partial calendars*, the performance is better in general. When agents exchange *partial calendars* the efficiency is more stable and better, in average, than when they exchange *total calendars* (remember figure 1 and figure 2). This result is interesting since we are primary interested in scenarios where agents try to keep their calendar information private. Furthermore, the meeting quality is not greatly affected when we move from the *total calendars* graph to the *partial calendars* graph.

On the other hand, when agents are exchanging *public preferences*, they are able to look for an agreement using the joint utility formula reaching optimal meeting joint qualities with the fewest possible number of exchanged proposals (remember figure 3). However, we should remember that in our experiments we are working under the honesty assumption. This means that agents use preference information in order to really try to reach an agreement with the highest joint quality. However, this is not always true in real life.

Now, it is interesting to note that varying preference privacy when agents exchange *partial calendars* does not affect the efficiency too much (remember figure 5). This is interesting since in realistic scenarios people usually try to keep both calendar and preference information private.

Our multi-agent system shows how we can provide automated support for the meeting scheduling task taking into account user preferences and keeping information private.

However, we have been working under some key assumptions: our agents accept in advance the protocol

and coordination policies; they agree in meeting locations, since we have not modeled this parameter in our system yet; our agents are honest (that is, they do not try to take advantage of the exchanged information in order to manipulate the outcome of the negotiation); they always try to reach the optimal joint meeting quality; finally, as human beings, they first relax their preferences before failing to reach an agreement (Raiffa 1982).

Conclusions and Future Directions

We have presented some of the preliminary results we obtained through experimentation with our multi-agent meeting scheduling system. This system is based on the communication protocol presented earlier in (Sycara & Liu 1994).

The experiments presented in this paper show some of the relationships between different experimental variables, such as calendar and preference privacy. The results show how the meeting scheduling performance is more stable and constant when agents try to keep their calendar and preference information private. We believe that these variables play a key role in the distributed meeting scheduling task, specially if we are interested in building distributed systems with truly autonomous and independent agents where there is not a fixed control agent.

We intend to continue our research towards more realistic scenarios relaxing some of our assumptions discussed in the previous section. We intend to let agents learn and infer other agents' mental attitudes and behaviors in order to model more complex and realistic scenarios where agents need to adapt to the whole group in decentralized environments.

Acknowledgments

This research has been sponsored in part by CONACYT under their scholarship program and by ITESM Campus Monterrey under the PCP program. Most of this work has been conducted at The Robotics Institute in Carnegie Mellon University.

References

Ephrati, E.; Zlotkin, G.; and Rosenschein, J. S. 1994. A non-manipulable meeting scheduling system. In *13th International Workshop on Distributed Artificial Intelligence*.

Garey, M. R., and Johnson, D. S. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman and Co.

Greif, I. 1982. Pcal: A personal calendar. Technical Report TR-213, MIT Laboratory for Computer Science, Cambridge, Mass.

Kelley, J. F., C., and Chapanis, A. 1982. How professional persons keep their calendars: Implications for computarization. *Journal of Occupanional Psychology* 55:141–156.

Kincaid, C.; Dupont, P.; and Kaye, A. 1985. Electronic calendars in the office: An assessment of user needs and current technology. *ACM Transactions on Office Information Systems* 3(1).

Maes, P. 1994. Agents that reduce work and information overload. *Communications of the ACM* 37(7):30–40.

Mitchell, T.; Caruana, R.; Freitag, D.; McDermott, J.; and Zabowski, D. 1994. Experience with a learning personal assistant. *Communications of the ACM* 37(7):80–91.

Raiffa, H. 1982. *The Art and Science of Negotiation*. Cambridge, Mass.: Harvard University Press.

Sen, S., and Durfee, E. H. 1991. A formal study of distributed meeting scheduling: Preliminary results. In *ACM Conference on Organizational Computing Systems*.

Sen, S., and Durfee, E. H. 1992. A formal analysis of communication and commitment. In *11th International Workshop on Distributed Artificial Intelligence*.

Sen, S., and Durfee, E. H. 1993. The effects of search bias on flexibility in distributed scheduling. In *12th International Workshop on Distributed Artificial Intelligence*.

Sen, S., and Durfee, E. H. 1994. Adaptive surrogate agents. In *13th International Workshop on Distributed Artificial Intelligence*.

Sen, S., and Durfee, E. H. 1995. Unsupervised surrogate agents and search bias change in flexible distributed scheduling. In *First International Conference on Multi-Agents Systems*.

Smith, R. 1980. The contract net protocol: High-level communication and control in distributed problem solver. *IEEE Transactions on Computers* 29(12):1104–1113.

Sycara, K., and Liu, J. 1994. Distributed meeting scheduling. In *Sixteenth Annual Conference of the Cognitive Society*.

Taub, E. 1993. Sharing schedules. *Mac User* 155–162.