Privacy/Efficiency Tradeoffs in Distributed Meeting Scheduling by Constraint−Based Agents [1]

Eugene C. Freuder, Marius Minca and Richard J. Wallace
Constraint Computation Center
Department of Computer Science, University of New Hampshire
Durham, NH 03824, USA
ecf−, mminca−, rjw@cs.unh.edu

Abstract

In many cases of negotiation or cooperative problem solving agents may be of 'two minds' about communicating with other agents. While some communication is necessary, privacy issues place a potential cost on such communication. We study this privacy / efficiency tradeoff in the context of meeting scheduling. Agents propose meeting times and places consistent with their own schedules while responding to other proposals by accepting or rejecting them. The information in their responses is either a simple accept/reject or an account of schedule conflicts, i.e. meetings in their schedule that conflict with the proposed meeting. In some conditions agents simply store the information gained about other agents' meetings; in others agents make inferences using simple consistency processing to eliminate more possibilities. In all situations we measure efficiency of problem solving and loss of private information ("privacy"). The latter includes either possible meetings (time and place) or meetings in the original schedules that have been identified by other agents. We find that, without making inferences, greater information exchange leads, naturally, to a greater loss of privacy, but that this loss is *not* compensated by greater efficiency. In contrast, when inferences are made, there is a marked enhancement in efficiency, which also leads to a diminished loss of privacy. We conclude that when privacy concerns are overriding, no explicit information should be exchanged, but if efficiency is also a concern, the best method is to combine a minimum of explicit information exchange with constraint−based inferences.

1. Introduction

As a result of the growth of the Internet and World Wide Web, it has become possible to automate a number of cooperative functions, even among widely dispersed participants. One area of application receiving considerable attention is meeting scheduling (Eaton et al., 1998; Garrido & Sycara, 1996; Sen & Durfee, 1995). This is a task which might be profitably delegated to software agents communicating over networks. In many cases it is expected that the agents will exhibit a degree of independence, since each is working for an individual with distinct affiliations.

In cooperative communication involving independent agents an important issue that arises is privacy. There will be cases where individuals will be interested in restricting the information communicated to other individuals to prevent sensitive information from being received by others. At the same time, the necessary cooperation involves some minimum amount of information exchange.

However, privacy in such contexts may incur costs. In particular, when information shared with other agents is kept to a minimum, cooperative decision making may become less efficient.

In this paper we set out to study this privacy/efficiency tradeoff empirically.

[1] Supported by NSF Grant No. IIS−9907385 and Nokia, Inc.

We use a multi−agent paradigm, where independent agents communicate with each other to solve a problem of mutual interest. In Section 2 we describe the basic situation and the experimental design. In Section 3 we discuss the results of our basic experiments. In Section 4 we consider the effects of agent−inference involving a simple form of constraint reasoning. Section 5 gives conclusions and future directions.

2. A Meeting Scheduling System

2.1. Overview

We implemented a *multi−agent meeting scheduling system* in which each agent has its own calendar, which consists of appointments in various cities (Boston, Philadelphia, New York, Los Angeles, San Francisco) at different hours (from 9am to 6pm) and different days (from Sunday to Saturday).

The meeting to be scheduled must have a day, a start−time and city, and the problem is solved when all agents reach an agreement on values for these attributes.

The basic constraints are the times required for travel between meetings in different cities. These are indicated in Figure 1. Times between cities on one coast are shown beside arcs connecting cities; the arc between the two circles represents the constraints between any city on one coast and the cities on the other.

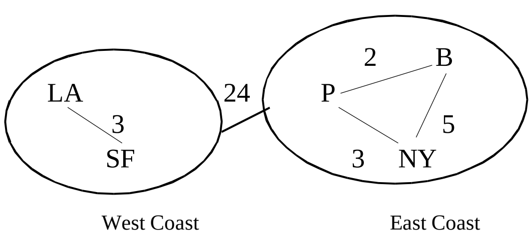

Figure 1. Time constraint graph

Agents negotiate by having one agent propose a meeting, which the other agents accept or reject, based on whether or not it fits (does not violate the time constraints within) their own schedules. Depending on the type of experiment, an agent will respond to a proposal in different ways with respect to the amount of data it decides to share (privacy factor).

We consider two extreme cases: a) agents respond to a proposal by giving only the minimum amount of information to the agent who made the proposal (e.g. "I'm sorry, I cannot meet then"), and b) agents respond by giving all their conflicts (e.g. "I cannot meet then, because I have meeting at 13:00 in Boston on Wednesday and a meeting in Los Angeles at 18:00 on Thursday"). We also include an intermediate case, where only one conflict is given as a reason for rejection. In the latter two cases, we compare the situation where agents use arc consistency to remove further possible meetings with one where they do not. Finally, we examine the situation where one agent uses arc consistency and the others do not. In each case we are concerned with the relationships between loss of privacy (how much information each agent gives the other agents) and efficiency (how quickly the agents arrive at a mutually acceptable meeting).

2.2. Experimental Strategy

The present experiments involve three agents. Each agent starts the negotiation having a number of initial meetings in its calendar. These initial meetings are generated so that there is at least one solution to the problem.

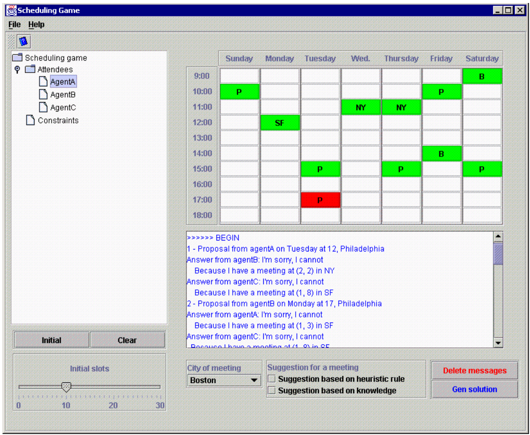

Figure 2. Multi−agent meeting sheduling system.

A sample test run is shown in Figure 2. The calendar shown is for agent A (see left−hand panel). There are 10 possible times for a meeting on each of the seven days (70 meeting slots). Agent A's initial schedule has 10 appointments, which are highlighted on the calendar (in green on the original GUI, here, in gray). Agents B and C have similar calendars. The darkened slot in the schedule (red in the original) is the meeting chosen as the "guaranteed solution" for this experiment (cf. below). In this condition, all conflicts are given and arc consistency is not used.

The following is the complete protocol for this experimental run:

>>>>>> BEGIN
1 − Proposal from agentA on Tuesday at 12, Philadelphia
Answer from agentB: I'm sorry, I cannot
  Because I have a meeting at (11, Tuesday) in NY
Answer from agentC: I'm sorry, I cannot
  Because I have a meeting at (17, Monday) in SF
2 − Proposal from agentB on Monday at 17, Philadelphia
Answer from agentA: I'm sorry, I cannot
  Because I have a meeting at (12, Monday) in SF
Answer from agentC: I'm sorry, I cannot
  Because I have a meeting at (17, Monday) in SF
3 − Proposal from agentC on Tuesday at 9, Los Angeles
Answer from agentA: I'm sorry, I cannot
  Because I have a meeting at (15, Tuesday) in P
Answer from agentB: I'm sorry, I cannot
  Because I have a meeting at (11, Tuesday) in NY
  Because I have a meeting at (11, Monday) in NY
4 − Proposal from agentA on Monday at 11, San Francisco
Answer from agentB: I'm sorry, I cannot
  Because I have a meeting at (11, Monday) in NY
  Because I have a meeting at (14, Sunday) in B
Answer from agentC: I'm sorry, I cannot
  Because I have a meeting at (14, Sunday) in B
5 − Proposal from agentB on Saturday at 11, Philadelphia
Answer from agentA: I'm sorry, I cannot
  Because I have a meeting at (9, Saturday) in B
Answer from agentC: I'm sorry, I cannot
  Because I have a meeting at (9, Saturday) in B
6 − Proposal from agentC on Friday at 17, Philadelphia
Answer from agentA: OK
Answer from agentB: OK
Result:
 − #proposals: 6
 − #values deleted: 44
 − #meetings identified: 9
 − amount of travel required: 0
<<<<<< END

In this situation, the basic parameters are number of agents, number of cities, number of initial meetings, and the time constraints. In the present experiments the only parameter varied is the number of initial meetings.

We run our experiments using the following scenario:

1. The number of initial meetings in an agent's calendar is the same for all agents and is constant for a given set of experimental runs. The values for this parameter in different sets of runs varies from 5 to 40 in steps of 5, giving the cases 5, 10, 15, 20, 25, 30, 35 and 40.
2. For each number of initial meetings there are 100 experimental runs.

In each experiment our main measures are an efficiency measure in the form of the number of proposals made before an acceptable meeting is found and two measures of privacy lost, which we take to be the amount of information communicated about an agent's own schedule: the number of possible meetings that have been discarded ("values deleted") and the number of prior meetings that have been identified.

3. Efficiency and Privacy − Empirical Results

3.1. Baseline Experiments: No Explicit Information Given

In the baseline experiments the amount of information exchanged during the communication is minimal: when an agent considers the proposal unacceptable, his answer is "I'm sorry, I cannot meet then", giving no specific information about his calendar. In this case, each rejection is counted as privacy loss of 1 because that meeting can be eliminated as a possibility for that agent. Since an acceptance does not allow one to make such a deduction, the privacy loss in this case is 0.

The protocol for each experimental run is as follows:

Set up:

1. Choose a solution at random (this means meeting time − day and hour − and city)
2. Generate the initial meetings for each agent (time and city) that satisfy the time constraints and guarantee that the initial solution is valid.

   All three agents will make proposals in a Round Robin order. The first agent (the proposer − or current agent) is selected at random.

repeat

generate a proposal (time and city) at random

check the proposal to see if:

− the time−slot is empty in the calendar of the current agent

− the proposal has no conflict in the current agent's calendar

− the proposal has not been already rejected

if the above conditions are not satisfied, go to the top of the loop and repeat the process

increment the number of *rounds*

the current agent makes the proposal to the other agents

if the both answers are positive, then a solution has been found, otherwise select a new agent using the Round Robin method

until (a solution is found)

The results of the baseline experiments are given in Figure 3. There is a curvilinear relation between number of initial meetings for each agent and the number of rounds (proposals) required to find a solution. This appears to be due to two factors: the number of possible solutions decreases as the number of meetings increases, causing the efficiency measure to increase. (In this case an increase means less efficiency.) But at the same time the number of proposals that satisfy the criteria for acceptance (cf. pseudocode above) decreases, which makes it easier to locate an acceptable solution.

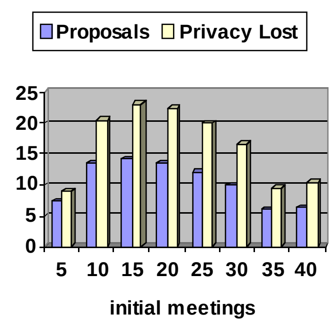

Figure 3. Efficiency and privacy measures in experiments where no explicit information is given about an agent's schedule. Values are means for each measure across 100 experimental runs. Privacy lost is equal to the number of meetings that have been eliminated.

3.2 Exchanging Explicit Information

In these conditions, when a proposal is made the other two agents respond by giving meetings that are in conflict with the proposed meeting. Here, we measure privacy in two ways: by counting possible meetings that have been eliminated and by counting each meeting communicated. For example, if agent C cannot attend a meeting in Boston at 12 PM on Wednesday because of a 10 AM meeting on that day in New York and an 11 AM meeting in Los Angeles on Thursday, then this may be counted as 9 meetings eliminated and two meetings identified.

In these conditions, each agent keeps "views" of the schedules of the other agents. This means that when an agent who makes a proposal gets a negative answer followed by information about the meetings that are in conflict with this proposal, it will update its view about the agent who rejected the proposal.

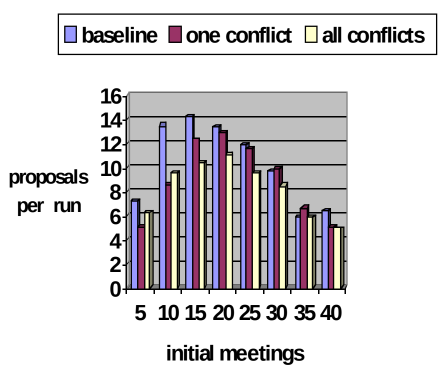

Figure 4. Efficiency measure (mean number of proposals per run) for experiments with explicit information exchange, compared with previous baseline experiments. Each value is based on 100 experimental runs.

In these experiments as in the first, a proposal is first chosen at random. Then, following the basic checks described in the last section, this proposal is checked against the current views that the proposer has of the other agents' schedules. In other respects, the experimental protocol is like that of the first experiment.

Figure 4 shows the difference in efficiency when explicit information is given about conflicts; for convenience, the baseline condition is shown together with the conditions in which reasons are given in the form of a single conflict or all conflicts. Although the expected trend is found overall, in that with explicit information, fewer proposals are required on average, the effects are usually quite modest. It appears that the effect is greater when the number of initial meetings is small (5−15). In addition, the effect of giving all conflicts rather than only one appears to be restricted to larger numbers of initial meetings (15−30).

In contrast, the amount of privacy lost is markedly affected, whether this is in the form of possible meetings eliminated (Figure 5) or the proportion of initial meetings that were identified (Figure 6). (The latter measure is zero in the baseline condition, since no meetings could be identified on the basis of value (meeting) elimination alone.)

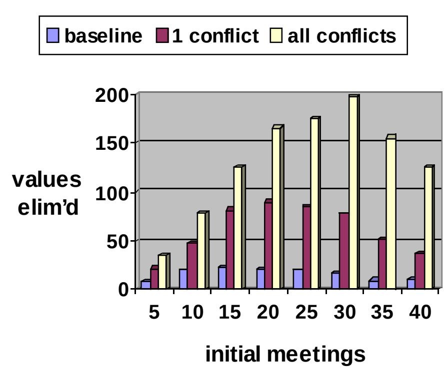

Figure 5. First measure of privacy lost per run for experiments with explicit information exchange, compared with baseline experiments. Same experiments as in Figure 4.

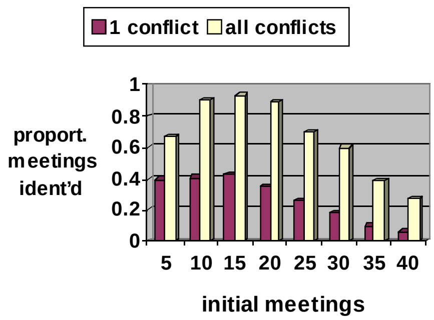

Figure 6. Proportion of meetings identified by other agents averaged over runs, for experiments with explicit information exchange. Same experiments as in Figure 4.

3.3 Making Inferences

If an agent maintains views of other agents, then it will be able to reason about possible meetings from these views. Here, we use a simple form of inference based on arc consistency.

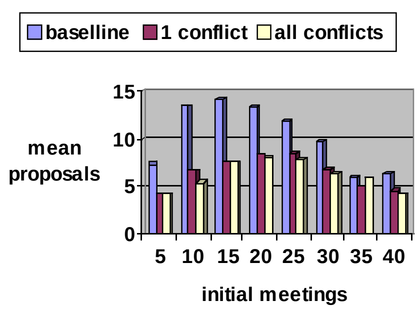

Figure 7. Efficiency measure (mean number of proposals per run) for experiments with explicit information exchange and arc consistency processing, compared with baseline experiments.

For example, suppose that one of agent B's meetings that is in conflict with agent A's proposal has the following parameters: Wednesday, 12 PM, Boston. Agent A can then update its view about agent B's schedule according with the time constraints like this: the only city available for a possible meeting on Wednesday, 1 PM is Boston. The same conclusion can be made for the following hours: 10 AM, 11 AM, 1 PM, and 2 PM. In the same way, agent A knows that at 3 PM on Wednesday agent B can only have a meeting in Boston or Philadelphia, and so on.

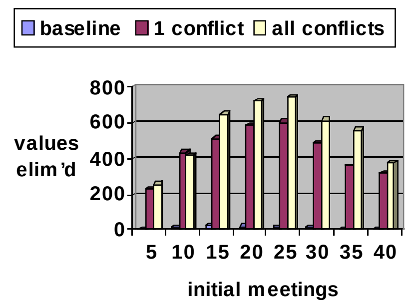

Figure 8. First measure of privacy lost (mean values deleted per run) for experiments with explicit information exchange and arc consistency processing, compared with baseline experiments. Same experiments as in Figure 7.

If agents make inferences of this sort as they gain information about other agents' conflicts, this results in a dramatic improvement in efficiency as long as the number of initial meetings is not large (Figure 7). For higher numbers of initial meetings, the number of candidate solutions decreases considerably, so that even the baseline condition is efficient.

Not surprisingly, the first privacy measure also shows a marked increase in comparison to the case of explicit information exchange without processing, since many more values can be eliminated through inference (Figure 8). On the other hand, the second measure shows a decrease (Figure 9). This shows that making inferences allows us to ameliorate the tradeoff between efficiency and privacy. Equally important, the gain in efficiency is almost as great when one conflict is communicated, while the loss of privacy is naturally reduced.

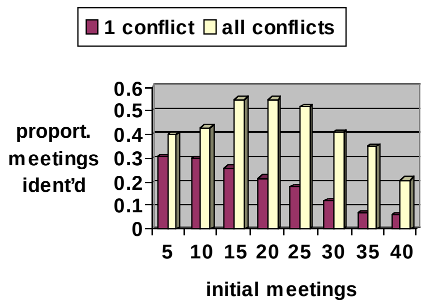

Figure 9. Proportion of meetings identified by other agents averaged over runs, for experiments with explicit information exchange and arc consistency processing. Same experiments as in Figure 7.

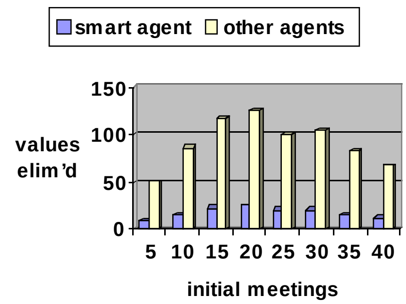

Figure 11. First measure of privacy lost (mean values deleted per run) for experiments with one smart agent. Agent using arc consistency processing ("smart agent") is shown separately from other agents. Agents give one conflict as reason for rejecting proposals.

4. The advantage of being clever

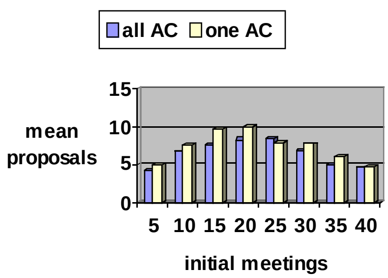

Figure 10. Mean number of proposals per run for experiment with one smart agent, compared with all−AC experiment (darker bars). Agents give one conflict as reason for rejecting proposals

In these experiments we allow only one agent to make inferences, based on information communicated about conflicts. Rather surprisingly, under these conditions efficiency falls off very little; Figure 10 shows the results when one conflict is communicated, and similar results are found for the all−conflicts condition. At the same time, the 'smart' agent gives up very little privacy in the form of elimination of possible meetings (Figure 11; similar but less marked results are found for the all−conflicts condition).

On the other hand, the proportion of meetings identified is not reduced under these conditions. In the first place, there is little difference in this case between the 'smart' agent and the others (Figures 12−13).

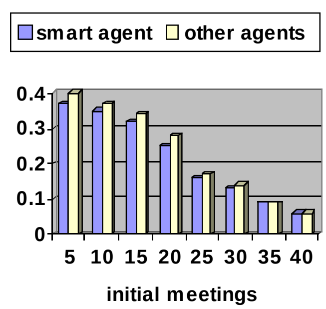

Figure 12. Proportion of meetings identified for experiments with one smart agent. Agent using arc consistency processing ("smart agent") is shown separately from other agents. Agents give one conflict as reason for rejecting proposals.

Secondly, the results for the 'smart' agent under these conditions differ very little from the conditions where all agents make inferences. To see this, compare the results for the smart agent in Figures 12 and 13 with mean proportion of meetings identified when one or all conflicts are communicated, as shown in Figure 9. Together, these results indicate that making inferences based on arc consistency is not sufficient to deduce meetings in the other agents' schedules.

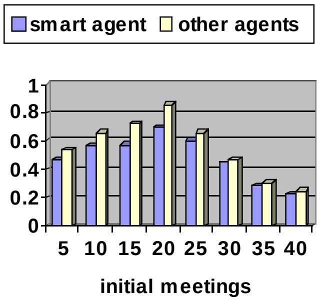

Figure 13. Proportion of meetings identified for experiments with one smart agent. Agent using arc consistency processing ("smart agent") is shown separately from other agents. Agents give all conflicts as reasons for rejecting proposals.

5 Conclusions and Future work

We have demonstrated the expected tradeoff between privacy and efficiency in the domain of agent negotiation. But we have also shown how it can be ameliorated. When more private information is exchanged, overall efficiency is not improved unless agents use some means of inference to reduce the search space further. In this case, the gain in efficiency forestalls the loss of private information. We have also shown that most of this gain is possible when agents exchange smaller amounts of explicit information (namely, giving only one conflict to explain why a proposal is rejected), and if only one agent (out of three) makes use of such inferences. In addition, these effects are dependent on the number of solutions to the problem. When the number of meetings in the initial schedules is large enough to reduce the number of possible meetings, then greater information exchange and constraint−based deductions do not improve on a simple guessing strategy that involves a minimum of privacy loss.

The next step in our work involves straightforward extensions to the present experiments. First and foremost is the extension of the preliminary work on reasoning with arc consistency (Section 3.3 above). We must also verify the present results under more conditions, and in so doing expand the space of parameter values tested (e.g. different numbers of agents, different time constraints).

In a later stage of our research, we will elaborate on the negotiation process and consider agent preferences and strategies. A communication between agents can involve a negotiation. Each agent can set its preferences and the possibility of changing the time, the date or the city for some meetings. Preferences may be based on the amount of travel necessary to get to a meeting and the price of this travel.

We can distinguish two basic strategies, that we call "selfish" and "unselfish". When a proposal is received by an agent and this proposal conflicts with his schedule, but he accepts it by changing one of his meetings, we say that the agent is unselfish. In contrast, a selfish agent will demand that the proposal be changed. A third case could involve a combination of these strategies: the agent who receives a proposal may change one of its meetings but also stipulate that the proposal should be changed.

Clearly, the distinction between selfish and unselfish is orthogonal to the private−public dimension. We can imagine an unselfish agent who simply reports an acceptance while making extensive changes to its schedule, or one that reports these alterations in great detail. Similarly, a selfish agent can simply signal rejection or give reasons why as well. However, we would like to know how variations in privacy interact with these strategies. In particular, it would be interesting to know if it makes sense for an agent to adopt one strategy or the other depending on the amount of information it has to reveal. For example, if an agent is unselfish and a maximal amount of information is being exchanged, will it be at a greater disadvantage vis a vis a selfish agent, in comparison to a situation where a minimal amount of information is exchanged?

Clearly, there are many issues to pursue in examining the importance of privacy in the context of negotiation among independent agents.

References

Eaton, P. S., E. C. Freuder, & R. J. Wallace (1998). Constraints and agents. Confronting ignorance. AI Magazine, Summer 1998, 51−65.

Garrido, L., & K. Sycara (1996). Multi−agent meeting scheduling: Preliminary experimental results. In: Proceedings, Second International Conference on Multi−Agent Systems, ICMAS−96. Menlo Park: AAAI Press.

Sen, S., & E. H. Durfee (1995). Unsupervised surrogate agents and search bias change in flexible distributed scheduling. In: Proceedings, First International Conference on Multi−Agent Systems, ICMAS−95, p. 336−343.