An Agenda-scheduling System Based on Persuasion Among Agents

Takayuki ITO Toramatsu SHINTANI
Department of Intelligence and Computer Science,
Nagoya Institute of Technology,
Gokiso, Showa-ku, Nagoya, 466 JAPAN.
E-Mail:{itota,tora}@ics.nitech.ac.jp

Abstract

In this paper, we propose a Multi-Agent Agenda-scheduling System. In our daily life, meeting scheduling is a time-consuming, iterative and somewhat tedious task. In multi-agent meeting scheduling, agents who act autonomously in the network can schedule meetings on individuals' behalf. In the workplace, agendas are often scheduled for meetings. In this paper, we schedule agendas for meetings. Ideally, all users' preferences should be accepted. It is, however, often difficult to reach an agreement among agents in consideration of all users' preferences. This persuasion mechanism facilitates making a collective agreement among agents in consideration of users' preferences. In agenda-scheduling, agents must quantify the user's subjective preferences. In order to quantify the user's subjective preferences, we employ the Analytic Hierarchy Process. In this paper, we present experiments that show how agents can reach a collective agreement by using a persuasion protocol.

1 Introduction

The term *agent*, used in a variety of ways, has recently commanded much attention in the field of AI. Agents should have their own knowledge and beliefs and act autonomously in an open network environment. In multiagent environments, agents' goals can be common, independent, or conflicting. Ideally, agents should be working toward an agreement in order to resolve conflict or encourage cooperation. Reaching an agreement is one of the most important task of agents.

In order to reach an agreement among agents, we have proposed a persuasion protocol[6, 7, 8]. The persuasion protocol is based on the rationality of agents. Agents should satisfy some criteria of rationality, e.g., economic rationality, logical rationality, etc. For example, economic rationality is satisfied by maximizing (expected) utility[9]. Logical rationality is satisfied by maintaining logical consistency.

In this paper, we chose a multiagent meeting scheduling domain as a testbed for the performance of the persuasion protocol. Ideally, in order to reflect users' preferences in group decision, many proposals should be accepted by agents. It is, however, difficult to accept all the proposals since the proposals can sometimes conflict. In this paper, we show how negotiation using the persuasion protocol can reach more agreements compared with existing negotiation protocols, e.g., Contract Net Protocol[16], Multistage Negotiation[2], and Unified Negotiation Protocol[12].

In the process described by this paper, agents schedule the meeting's agendas. One problem in agenda scheduling is how to allocate agendas to time intervals. Each agent has individual preferences regarding agendas and costs on time intervals. The cost on a time interval means how busy the agent's user is in the time interval. Each agent proposes both an agenda and an time interval according to his/her preferences. Other agents receive the proposal and accept or reject the proposal according to his/her own preferences.

Another problem in agenda scheduling is how agents measure the user's subjective preferences. Since agents are computer programs, we must quantify the user's subjective preferences. In order to quantify the subjective preferences, we employ the Analytic Hierarchy Process (AHP)[13].

This paper is organized as follows: Section 2 presents the persuasion protocol. Section 3 shows an agenda scheduling system. Section 4 presents agent's behaviors. In here, we show the how agents reduce the pairwise comparisons for their users and negotiation protocols with/without the persuasion protocol. Section 5 shows some experimental results in the mul-

Information Systems and Technologies for Network Society, Fukuoka, Japan, September 1997

tiagent agenda scheduling. Section 6 shows related works. Section 7 presents some concluding remarks.

2 Persuasion Protocol

In the persuasion protocol, an agent who persuades another agent is called a **persuader** and an agent who is persuaded by a persuader is called a **compromiser**. The outline of the persuasion protocol can be shown as follows:

1. **Request**: The persuader sends a proposal to the compromiser in order to reach an agreement.
2. **Belief revision**: The compromiser receives the proposal. If the compromiser is able to accept the proposal, he/she needs not revise his/her belief. If he/she is unable to accept the proposal, the compromiser tries to revise his/her belief (needs, decisions, or preference) in order to accept the proposal.
3. **Reply**: As the result of the belief revision, if the compromiser is able to accept the proposal, he/she replies an agreement message. If not, the compromiser replies a reject message.

The compromiser should satisfy some criteria of rationality, e.g. logical consistency, economic rationality. For example, we can describe how the compromiser satisfies the economic rationality as follows: The persuader and the compromiser use evaluation functions in order to evaluate the proposals. The persuader uses evaluation function f. The compromiser uses evaluation function g. Now, proposal A and proposal B exist. The evaluation of proposals are $f(A) \geq f(B)$ and $g(B) \geq g(A)$. If the persuader send a proposal A to the compromiser, the compromiser should revise his/her belief. In order to satisfy the economic rationality, as the result of the belief revision, the compromiser's evaluation of the proposals should be at least $g(A) \geq g(B)$.

3 Multi-Agent Agenda-scheduling System

3.1 System Architecture

Figure 1 shows the architecture of our system. In our system, users' computers are connected by a network. Each user's decision-making is supported by their own hierarchical decision support module in our system. Agents manage their particular user's hierarchical decision support module, and negotiate based on the information it supplies. The agent's tasks are described in section 4. The hierarchical decision support module

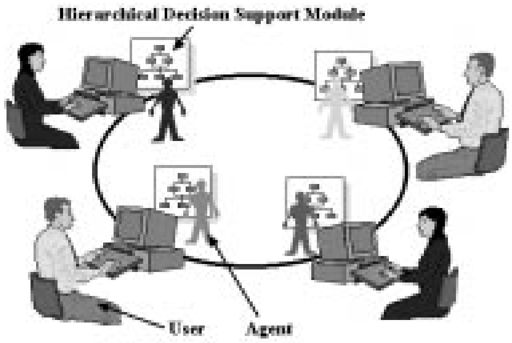

Figure 1: System architecture

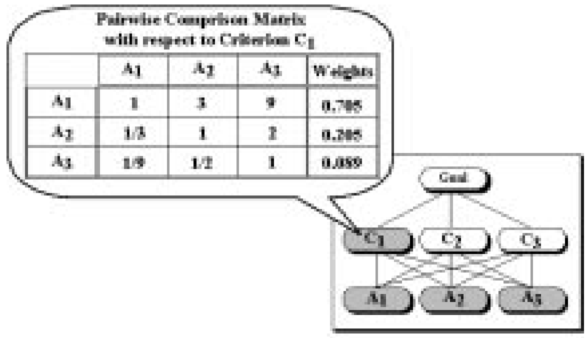

Figure 2: Analytic Hierarchy Process

has functions to make judgements for pairwise comparisons, and to construct a hierarchy.

3.2 Quantifying Subjective Judgements

In order to measure subjective judgements of users, we employ AHP in our system. The AHP is a method for making decisions that are hard to analyze quantitatively. It combines both systems approach and subjective judgements, and its primary purpose is to maximize the user's intuition and experience.

In the AHP, users decompose the problem into a hierarchy that consists of a goal, criteria (and possibly sub-criteria), and alternatives. The judgement of the pairwise comparison between factors (in Figure 2, alternatives A_1,A_2 and A_3) on a certain level is made with respect to the criterion that is a factor (in Figure 2, criterion C_1) on the upper level. By interpreting a set of values of judgements as a matrix (top left of Figure 2), the weights (i.e., measurement of criteria) of factors are calculated analytically. To put it more concretely, we can achieve the weights of each factor as the eigen-vector for the max eigen-value of the pairwise comparison matrix. As a whole hierarchy, the weights of the alternatives can be calculated by composing the weights of the criteria.

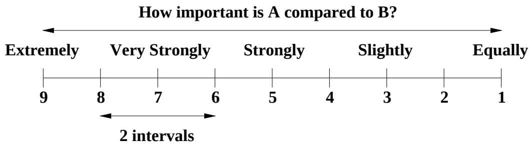

Figure 3: Scale of measurement for AHP

A pairwise comparison matrix is characterized by the following: (1) diagonal values are 1; and (2) the values of elements in a matrix are $a_{ij} = 1/a_{ji}$. For judgements of pairwise comparisons in the AHP, we can use a 9 point scale consisting of five words (equally, slightly, strongly, very-strongly and extremely (Figure 3)) and four intermediate levels (e.g., between slightly and strongly)

AHP provides a measure the inconsistency in each set of judgements. This measure is called the inconsistency ratio (I.R.) and is defined as follows: $I.R. = (\lambda_{max} - n)/(n-1)$. Here, $\lambda_{max}$is the max eigen-value described above and n is the size of the pairwise comparison matrix. Ideally, a set of judgements in a pairwise comparison matrix will be consistent, and the inconsistency ratio will be 0. If the inconsistency ratio is no more than 0.1, this means that the pairwise comparison matrix is consistent.

3.3 Agenda-scheduling

The process for agenda scheduling is described as follows: First, a host user proposes a meeting to other users who will be participating in the meeting. Second, the host user generates agendas, time intervals, broadcast agendas, and time intervals to the users. Third, each user decides the weights of the agendas and the costs of the time intervals by employing AHP. Fourth, agents negotiate with each other based on their users' subjective weight. Negotiation among agents is described in section 4. Finally, the result of the negotiation is reported to all users.

Figure 4 shows an example of agenda scheduling. The top right of Figure 4 is a set of agendas from the second International Conference on Multi-Agent Systems (ICMAS-96). Agenda scheduling decides how to sort these agendas within the time intervals. Users preferences are reflected in the position of the sorted agendas. For example, "It is difficult for me to arrive early in the morning" or "Invited talks should take place in the afternoons." When all agendas are allocated to the time intervals, the agenda scheduling is success.

The benefits of agenda scheduling by autonomous agents are described as follows: We can construct a system in which users' private data (in this paper, weights of agendas and costs of time intervals) are not exchanged and shared. Because the users' private data are not exchanged and shared, we can preserve the users' privacy.

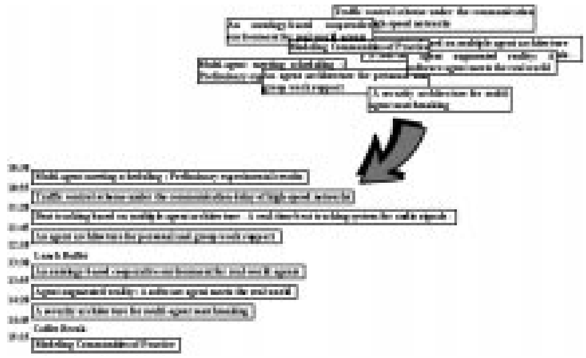

Figure 4: An example of agenda-scheduling

4 Agent's Behavior

4.1 Reducing the Pairwise Comparisons

Each agent manages the decision hierarchy and the pairwise comparison matrix constructed by its user. In general, the AHP requires too many judgements of its user. If there are n factors for a criterion, users have to make $n(n-1)/2$ judgements. If the number of levels of the decision hierarchy or the number of factors of the level is increased, more and more judgements are required. Acting on the user's behalf, the user's agent effectively reduces the number of judgements, which leads the user to make consistent judgements dynamically using the following methods.

In general, users have to judge all pairwise comparisons in the AHP. In fact, it is very hard for a user to judge all pairwise comparisons. In our system, the initial value of the elements of all pairwise comparison matrixes is 1 (i.e.,"Equally Important"). Because there are initial values, the user changes only the value that the user wants to judge.

In the AHP, the value of an element a_{ij} in the matrix equals W_i/W_j. Here, the W_i expresses the weight of the factor i. If a pairwise comparison matrix is consistent (i.e., $I.R. = 0$), a certain element can be inferred from the other two elements. For example, in a matrix, because the value of an element a_{12} is W_1/W_2 and the value of an element a_{23} is W_2/W_3, we can infer that the value of an element a_{13} is $a_{12} \times a_{23} = W_1/W_2 \times W_2/W_3 = W_1/W_3$. In the system, in order to reduce the number of judgements, agents infer the value of an element in a matrix using this feature of the AHP. When the agent infers the value of an element, if the element was judged by the user, the agent asks the user to change the judgement. If the element has never been judged, the agent changes the value of the element to the new value inferred.

In general, in order to reduce the number of judgements, the Harker Method [5] is now widely employed. However, to use the Harker method, the covering condition must be satisfied on a matrix. In order to satisfy the covering condition, many judgements, from which all elements in the matrix must be inferred, are required. In a real system, it is also hard for the user to make judgements with the covering condition. If the user has a chance to satisfy the covering condition, the system exploits the Harker method instead of the simple method mentioned above to omit some redundant pairwise comparisons and check the consistency of each comparison dynamically. In a pairwise comparison matrix, when the value of $I.R.$ is more than 0.1, this means that the matrix is inconsistent and the user must remake all the judgements. Naturally, this remaking creates additional work load for the user. In our system, while the user is making judgements in a matrix, the agent is watching the $I.R.$ dynamically. When the $I.R.$ is more than 0.1, the agent requests that the user remake the present judgement.

4.2 Persuasion Based Negotiation

In this section, we show the agent negotiation based on the persuasion protocol for agenda scheduling. $N = \{a_1, a_2, \ldots, a_l\}$ is the set of agents. $T = \{t_1, t_2, \ldots, t_m\}$ is the set of time intervals. $M = \{m_1, m_2, \ldots, m_n\}$ is the set of agendas. t_{m_k} represents the time interval where the agenda m_k is located. As a condition to locate the agenda m_k at t_{m_k}, n_{m_k} is the number of supporters of the agenda m_k. We call this condition the agreement condition. $C_{a_i}(t_j)$ is the cost of agent a_i for the time interval t_j. $W_{a_i}(m_k)$ is the weight of the agent a_i for the agenda m_k. $C_{a_i}(t_j)$ and $W_{a_i}(m_k)$ are decided by agent a_i's user with AHP.

Next, we show the basic negotiation protocol based on the contract net protocol. The basic negotiation protocol does not use the persuasion protocol. Respectively, Announcement, Bidding, and Award in contract net protocol correspond to Propose, Report Accepting or Rejecting, and Sum up in the basic negotiation protocol.

Step 1 Propose An agent $a_{proposer}$ is selected using a particular method; for example, according to relevant users' authority. In the experiments of Section 5, $a_{propser}$ is selected randomly. $a_{proposer}$ sends a proposal $< m_k, t_j >$ to all other agents. The proposal $< m_k, t_j >$ means that agenda m_k is located at time interval t_j. m_k is the agenda upon which an agreement has not been reached in the set of agendas M. m_k and t_j maximize $W_{a_{proposer}}(m_k) - C_{a_{proposer}}(t_j)$. In other words, $< m_k, t_j >$ represents the most preferable pair of m_k and t_j.

Step 2 Report Acceptance or Rejection The agent who receives the proposal pledges that he negotiates on the proposal $< m_k, t_j >$ *only* (This pledge is called *commitment*[15]). Each agent reports the rejection or acceptance of the proposal $< m_k, t_j >$. If the cost of the time interval t_j for the agent a_i equals 0, $C_{a_i}(t_j) = 0$, a_i accepts the proposal $< m_k, t_j >$. If $C_{a_i}(t_j) > 0$, a_i rejects the proposal $< m_k, t_j >$. Namely, if the time interval t_j is convenient for the agent, he accepts this proposal. If not, he/she rejects this proposal.

Step 3 Sum up The agent $a_{propser}$ sums up the acceptance or rejection reports. If the acceptance number is bigger than the condition of the number of supporters n_{m_k}, the agents reach an agreement on the proposal $< m_k, t_j >$. If the agents reach an agreement, this negotiation is success. The agent $a_{proposer}$ broadcasts to other agents whether or not this negotiation is successful. Then, if the agents here reached agreements for all agendas, they finish the negotiation. If not, they restart the negotiation process from the Step 1 Propose.

In order to reach more agreements, we apply the persuasion protocol to the basic negotiation protocol. Step A and Step B are the same as Step 1 and Step 2 of the basic negotiation protocol.

Step A Propose An agent $a_{proposer}$ is selected. $a_{proposer}$ sends a proposal $< m_k, t_j >$ to all other agents.

Step B Report Acceptance or Rejection The agent who receives the proposal commits to the agent $a_{proposer}$ on the proposal $< m_k, t_j >$ only. Each agent reports acceptance or rejection of the proposal $< m_k, t_j >$.

Step C Sum up The agent $a_{propser}$ sums up the acceptance or rejection reports. If the acceptance number is bigger than the condition of the number of supporters n_{m_k}, the agents reach an agreement on the proposal $< m_k, t_j >$. If the agents reach an agreement, this negotiation is success. The agent $a_{proposer}$ broadcasts to other agents whether or not this negotiation is successful. The proposal on which an agreement could not be reached is suspended by the agent $a_{proposer}$. The agent $a_{proposer}$ broadcasts the suspended proposal. Since the weight of the agenda and the cost of time interval of the suspended proposal are private data for $a_{proposar}$, they are not broadcasted. Each agent a_i has a suspended list L_{a_i}. When the agent a_i receives the suspended proposal, the proposal is registered with the suspended list L_{a_i}. The possibility of suspending a proposal is a feature of this protocol.

Step D Persuasion If the agents could not reach an agreement on the proposal $< m_k, t_j >$, the agent $a_{proposer}$ persuades the other agent. Here, the suspended proposal $< m_{k'}, t_{j'} >$ is proposed previously by the agent $a_x, where\ x \neq proposer$. The proposal $< m_{k'}, t_{j'} >$ is included in the suspended list L_{a_i} of each agents. Now, the agent $a_{proposer}$ plays the role of the persuader. The agent a_x is the compromiser. If $C_{a_{proposer}}(t_{j'}) < W_{a_{proposer}}(m_k)$, the persuader $a_{proposer}$ persuades the compromiser a_x, where $C_{a_{proposer}}(t_{j'})$ is the cost of the time interval of the suspended proposal $< m_{k'}, t_{j'} >$ and $W_{a_{proposer}}(m_k)$ is the weight of the agenda $m_{k'}$ of the suspended proposal $< m_{k'}, t_{j'} >$.

- **request**: The persuader $a_{proposer}$ requests of the compromiser a_x that a_x accepts the proposal $< m_k, t_j >$ proposed by $a_{proposer}$ and in compensation for this, the persuader $a_{proposer}$ accepts the proposal $< m_{k'}, t_{j'} >$ proposed by a_x.
- **belief revision**: The compromiser a_x receives the proposal $< m_k, t_j >$ from the persuader $a_{proposer}$. If $C_{a_x}(t_j) < W_{a_x}(m_{k'})$, the persuader $a_{proposer}$ accepts the proposal $< m_k, t_j >$. If not, he/she rejects the proposal $< m_k, t_j >$.
- **reply**: The compromiser a_x sends the report accepting or rejecting the proposal $< m_k, t_j >$ to the persuader $a_{proposer}$.

Step E Sum up The agent $a_{propser}$ receives the report of the acceptance or rejection of proposal $< m_k, t_j >$. If the proposal is accepted, the agent $a_{proposer}$ accepts the suspended proposal $< m_{k'}, t_{j'} >$ and conveys the acceptance to agent a_x. If the number accepting is bigger than the condition of the number of supporters n_{m_k} and $n_{m_{k'}}$, the agents reach an agreement on proposals $< m_k, t_j >$ and $< m_{k'}, t_{j'} >$ respectively. If agents reached agreements for all agendas, they finish the negotiation. If not, they restart the negotiation process from the Step A.

Figure 5 shows an example of the persuasion. Here, agent a is the persuader and agent b is the compromiser. Each agent has individual preferences shown at the middle of Figure 5. Each histogram at the middle of Figure 5 shows the cost of time intervals. For example, the persuader has cost 0.1 at time interval 10 and weight 0.5 for agenda A. In order to preserve the users' privacy, these individual preferences are not shared and not compared with each other's. The persuader proposed the proposal 1 and the proposal was rejected by other agents. The compromiser suspended the proposal 2 previously.

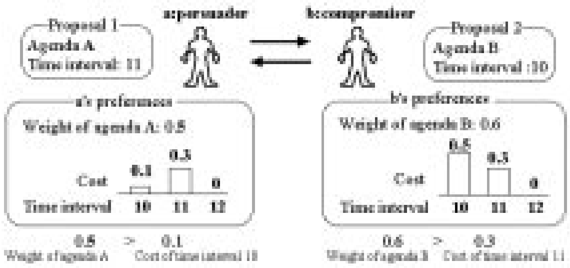

Figure 5: An example of the persuasion

The persuasion protocol progresses as follows: The persuader compares his/her weight of the agenda A with his/her cost of the time interval 10 of the agenda B. Since the weight of the agenda A (0.5) is bigger than the cost of the time interval 10 (0.1), the persuader can accept proposal 2 according to the economic rationality. Then, the persuader requests of the compromiser that the compromiser accepts the proposal 1 proposed by persuader, and in compensation for this, the persuader accepts the proposal 2 proposed by the compromiser. The compromiser compares his/her weight of the agenda B with his/her cost of the time interval 11 of agenda A. Since the weight of the agenda B (0.6) is bigger than the cost of the time interval 11 (0.3), the compromiser can accept proposal 1 according to the economic rationality. Now, each proposal, proposal 1 and proposal 2, are accepted by the persuader and by the compromiser.

An advantage of the persuasion protocol is that agents need not compare their own preferences with each other's. In a decision support domain such as agenda scheduling, the user's individual preferences should not compare directly and simply. The persuasion protocol described above can be realized only according to the economic rationality of the agent.

5 Experimental Results

Our preliminary set of experiments involve five agents, 8 or 18 agendas, and 8 or 18 time intervals. Each agent has costs in some time intervals as noise. Each agent considers noise during some time intervals as those intervals' noise. The noise represents how

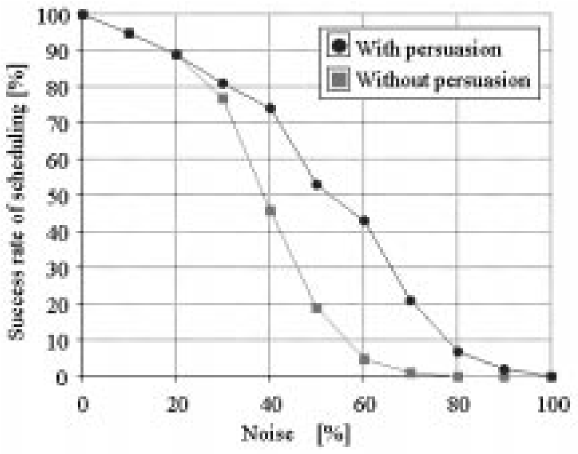

Figure 6: Condition : 3 , Agendas : 8

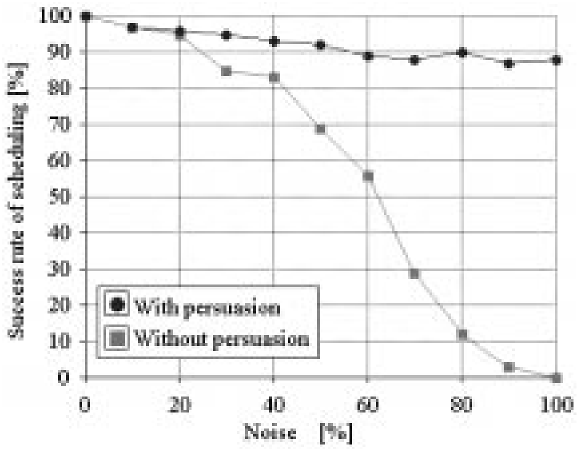

Figure 7: Condition : 2 , Agendas : 8

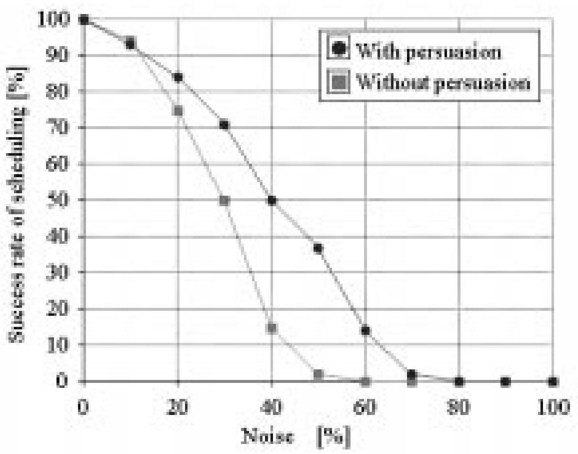

Figure 8: Condition : 3 , Agendas : 18

many time intervals has cost. If the time interval t_j has cost, $C_{a_i}(t_j)$ is bigger than 0, $C_{a_i}(t_j) > 0$. For example, if the noise is 50%, a half of the time intervals has cost. The aim of the experiment is to collect data confirming the benefits of the persuasion protocol. In order to show the performance of the persuasion protocol, we assume that costs and weights are random value without AHP in these experiments.

"Without persuasion" is the basic negotiation in which agents negotiate according to Step 1, Step 2, and Step 3. Here, one turn consists of Step 1, Step 2, and Step 3. "With persuasion" is the negotiation with the persuasion protocol in which agents negotiate according to Step A, Step B, Step C, Step D, and Step E. Here, one turn consists of Step A, Step B, Step C, Step D, and Step E. We observed whether agenda schedulings succeeded or failed in 100 turns. One trial consisted of 100 turns. We varied the noise from 0% to 100% and observed how many schedulings were successful in 100 trials. Figure 6,Figure 7 and Figure8 were created by averaging 10 charts.

Figure 6 shows the experimental scenario wherein all agreement conditions $n_{m_k} = 3$. Figure 7 shows the experimental scenario wherein all agreement conditions $n_{m_k} = 2$. In Figure 6 and Figure 7 shows that agents can reach a lot of agreements in the case "with persuasion" rather than in the case "without persuasion."

When the agreement condition is 3 (Figure 6), during which the noise is from 30% to 90% agents can reach a lot of agreements in the case "with persuasion" more than in the case "without persuasion." In particular, in conditions where the noise is from 40% to 70%, the success rate of scheduling with persuasion is 30% higher than the success rate of scheduling without persuasion.

When the agreement condition is 2 (Figure 7), the success rate of scheduling with persuasion over all the noise density levels is higher than the success rate of scheduling without persuasion. In this case, the success rate with persuasion could be much improved. Since the persuasion protocol is based on two agents negotiating, it is easier to reach an agreement if the agreement condition is 2.

Figure 8 shows a more realistic scenario where the number of agendas are 18. The agreement condition is 3. In this case, agents also can reach a lot of agreements with persuasion more than without persuasion during noise is from 20% to 70%.

The results of the experiments demonstrate that the persuasion protocol can improve the rate of agreement in agenda scheduling.

6 Related Work

An interesting study in distributed meeting scheduling is Sen and Durfee[14] who focused on solving the meeting scheduling problem using a central host agent. For their agents, they chose to adapt the multistage negotiation protocol[2], which is a generalization of the contract net protocol[16]. They proposed some heuristic strategies such as Search Biases, Announcement Strategies, Bidding Strategies, and Commitment Strategies and have analyzed the strategies' performance. However, in their study user preferences are not taken into account during the meeting scheduling process.

Garrido and Sycara[4] focused on decenterized meeting scheduling. They 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. In their protocol, each agent is able to relax its preferences when conflicts arise. In other words, the agent compromise without rationality. In our work, if the agent's rationality is satisfied, the agent can compromise when the agent is persuaded. Moreover, they did not establish how to measure the subjective judgements of users. In this paper, in order to measure the judgements of users, we employ the Analytic Hierarchy Process (AHP) [13].

We can also find other investigations in multiagent meeting scheduling. Bui, Kieronska, and Venkatesh[1] described the integration of a learning module into a communication-intensive negotiating agent architecture. This learning module gives the agents the ability to learn about other agents' preferences via past interactions. This approach is illustrated with an example from the distributed meeting scheduling domain. They have been focused on learning, while we have been focused on negotiation among agents.

Ephrati, Zlotkin, and Rosenschein[3] presented an alternative approach which is economic in flavor. They presented three scheduling mechanisms based on vote. However, these mechanisms are all sensitive to manipulation. In order to remove manipulability from them, the authors introduced the Clarke Tax Mechanism. In general, a group reaches consensus by use of a vote. But the results of voting are often inconsistent, largely due to the inconsistency of voting rules: Majority Rule, Single Voting Rule, etc[9]. In our work, we propose a persuasion protocol rather than voting methods for negotiation among agents.

Maes[10] has focused on learning the user's preference by agent. She implemented an agent for electronic mail handling, an agent for meeting scheduling, an agent for electronic news filtering, and an agent that recommends books, music or other forms of entertainment. She did not, however, focus on negotiation among agents.

The main difference between these studies and this paper is described as follows. These researches have been focused on when the meeting is held. In this paper, we focused on how the agenda of the meeting is drawn up.

In the field of Computer Supported Cooperative Work (CSCW)[11], meeting support systems have been developed. That can schedule a meeting. Busbach[17] presented EuroCoOp Task Manager. Busbach implemented a meeting support system using the EuroCoOp Task Manager and pointed out that meeting support systems should be able to support a variety of different meeting types, such as kick-offs, project team meetings, presentations, training, seminars, etc. Busbach has focused on supporting a meeting. In our work, we have focused on scheduling meetings and agendas based on multiagent negotiation.

7 Conclusion

In this paper, we proposed the Multi-Agent Agenda-scheduling System. Reaching agreements are one of the important tasks of autonomous agents. In this paper, we applied agreements-reaching among agents to the agenda scheduling domain. In this domain, it is important to reflect users' preferences in agenda scheduling. In order to reflect users' preferences, agents should accept many proposals and reach a collective agreement. In order to reach many agreements among agents, we proposed the persuasion protocol. It is also important to quantify the user's subjective preferences. We employed AHP in order to quantify the user's subjective preferences. We ran experiments in which we introduced varying amounts of noise into the time intervals, the agreements condition and the number of agendas. The experiments show how agents can reach a collective agreements by using the persuasion mechanism.

References

[1] H. H. Bui, D. Kieronska, and S. Venkatesh: Learning other agents' preferences in multiagent negotiation, *Proceedings of the Thirteenth National Conference on Artificial Intelligence(AAAI-96)*, pp. 115–119, 1996.

[2] S. Conry, R. A. Meyer, and V. R. Lesser: Multistage Negotiation in Distributed Planning, *Readings in Distributed Artificial Intelligence* (Bond, A. H. and Gasser, L.(eds.)), Morgan Kaufman, pp. 367–384, 1988.

[3] E. Ephrati, G. Zlotkin, and J. S. Rosenschein: A Non-manipulable Meeting Scheduling System, *The Thirteenth International Distributed Artificial Intelligence Workshop*, pp. 105–125 ,1994.

[4] L. Garrido and K. Sycara: Multi-Agent Meeting Scheduling: Preliminary Experiment Results, *Proceedings of Second International Conference on Multi-Agent Systems(ICMAS-96)*, AAAI Press, pp. 95–102 ,1996.

[5] Patrick T. Harker: Incomplete pairwise comparisons in the analytic hierarchy process, *Mathmatical Modelling*, 9:353–360, 1987.

[6] T. Ito and T. Shintani: An Approach to a Multiagent Based Scheduling System Using a Coalition Formation, *Proceedings of the Ninth International Conference on Industrial & Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE-96)*, Gordon and Breach Science Publishers, p. 780 ,1996.

[7] T. Ito and T. Shintani: Implementing an Agent Negotiation Protocol Based on Persuasion, *Proceedings of the Second International Conference on Multiagent Systems(ICMAS-96)*, AAAI Press, p. 443 ,1996.

[8] T. Ito and T. Shintani: Persuasion among Agents : An Approach to Implement a Group Decision Support System Based on Multi-Agent Negotiation, *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence(IJCAI-97)* ,1997 (to appear).

[9] R. D. Luce and H. Raiffa: Games and decisions, Dover Publications, 1985.

[10] P. Maes: Agents that Reduce Work and Information Overload., *Communications of the ACM*, Vol. 37, No. 7, pp. 31–40 ,1994.

[11] W. Paul: Introducing cscw - what it is and why we need it. In Stephen A.R. Schrivener, editor, *Computer-Supported Cooperative Work*, Applied Information Technology series, chapter 1, pp. 1–18. UNICOM, 1994.

[12] J. S. Rosenschein and G. Zlotkin: Rules of Encounter, The MIT Press ,1995.

[13] Thomas L. Saaty : The Analytic Hierarchy Process, McGraw Hill, 1980.

[14] S. Sen and E. H. Durfee: On the design of an adaptive meeting scheduler, *The Tenth IEEE Conference on Artificial Intelligence for Applications*, pp. 40–46 ,1994.

[15] S. Sen and E. H. Durfee: The role of commitment in cooperative negotiation, *International Journal on Intelligent and Cooperative Information Systems*, Vol. 3, No. 1, pp. 67–81 ,1994.

[16] R. G. Smith: The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver, *IEEE Transactions on Systems, Man, and Cybernetics*, pp. 1104–1113 ,1980.

[17] B. Uwe and K. Thomas: Support for meetings using the eurocoop task manager. In Stephen A.R. Schrivener, editor, *Computer-Supported Cooperative Work*, Applied Information Technology series, chapter 10, pp. 149–170. UNICOM, 1994.