A Genetic Fuzzy Decision Agent based on Personal Ontology for Meeting Scheduling Support System*

Chang-Shing Lee
Department of Information Management
Chang Jung University
Tainan, 711, TAIWAN
leecs@mail.cju.edu.tw

Meng-Ju Chang
Department of Information Management
Chang Jung University
Tainan, 711, TAIWAN
r76901106@mail1.cju.edu.tw

Chen-Yu Pan
Department of Engineering Science
National Cheng Kung University
Tainan, 701, TAIWAN
n9692412@ccmail.ncku.edu.tw

Chyi-Nan Chen
Department of Engineering Science
National Cheng Kung University
Tainan, 701, TAIWAN
cnc@mail.ncku.edu.tw

***Abstract* - A *Genetic Fuzzy Decision Agent (GFDA)* based on personal ontology for *Meeting Scheduling Support System (MS3)* is proposed in this paper. We apply the concept of personal ontology to the *MS3*, and extend the four-layered object-oriented ontology to *Personal Meeting Scheduling Ontology (PMSO)* for the *MS3*. When an organization member sends a meeting request with required information to *MS3*, the *MS3* will retrieve the invitees' information from *Personal Meeting Scheduling Ontology (PMSO) Repository* and send a response to *GFDA*. Then, *GFDA* utilizes these invitees' information along with the *PMSO* and meeting information to infer the suitable meeting time slots. Therefore, *GFDA* can analyze each invitee's attendance possibility based on soft-computing technology. The experimental result shows that the proposed *GFDA* is feasible, efficient and usable for meeting scheduling support system.**

I. INTRODUCTION

Ontology is a collection of key concepts and their interrelationships collectively providing an abstract view of an application domain. With the support of ontology, both user and system can communicate with each other by the shared and common understanding of a domain [1]. Moreover, an ontology is a computational model of some portions of the world. It is often captured in a semantic network and represented by a graph whose nodes are concepts or individual objects and whose arcs represent relationships or associations among the concepts [2]. On the other hand, an agent is a program that performs unique tasks without direct human supervision. J. Ferber [3] gives another definition of an agent such as "an agent is capable of acting in an environment and can communicate directly with other agents". An intelligent agent is more powerful than an agent because of the reasoning and learning capabilities [4]. Employees in an enterprise may spend much of their time scheduling and attending meetings. The process of searching for an available meeting time can be complicated by communication delayed and by other concurrently scheduled meeting. Automating meeting scheduling not only can save the user time and effort, but also lead to more efficient schedulers and improvements in how the enterprise exchanges information.

S. Sen [5] proposes a system using intelligent meeting-scheduling agents that can negotiate with other agents without compromising their user-specified constrains. In addition, K. Sugihara et al. [6] propose a meeting scheduler for office automation. They took the priorities of persons and meetings as considerations in the real office environment, and use a heuristic algorithm for timetable rearrangement. T. Haynes et al. [7] propose an automated meeting scheduling system that utilizes user preferences. W. S. Jeong et al. [8] focuse on how the meeting-scheduling agent can reduce failures when there is no common time-slot. They solve the failure condition with utilizing the cooperation and the rescheduling strategy. E. Mynatt et al. [9] present a calendar system extension that uses a Bayesian model to predict the likelihood of one's attendance at the events listed on his or her schedule. A. Ashir et al. [10] propose a multi-agent based decision mechanism for distributed meeting-scheduling system. J. A. Pino et al. [11] accommodate users' availability according to their own preferences and restrictions of schedule meetings. F. Bergenti et al. [12] describe an agent-based computer-supported cooperative work system designed to promote the productivity of distributed meetings by means of agent. C. Glezer [13] proposes and evaluates a comprehensive agent-based architecture for an inter-organizational intelligent meeting-scheduler. C-S Lee et al. [14] use an intelligent fuzzy meeting agent to infer attendance possibility. In this paper, we develop a *Genetic Fuzzy Decision Agent (GFDA)* based on personal ontology for *Meeting Scheduling Support System (MS3)*.

This paper is organized as follows. In Section II, we apply the concept of personal ontology to meeting scheduling support system, and use a four-layer object-oriented ontology for this application. Section III introduces the architecture of meeting scheduling support system. The experimental results are presented in Section IV. Finally, the conclusion is given in Section V.

II. PERSONAL ONTOLOGY FOR MEETING SCHEDULING SUPPORT SYSTEM

With an ontology, we can organize keywords and database concepts by capturing the semantic relationship among the keywords or among the tables and fields in databases [2]. The semantic relationship can provide an abstract view of the information space for our schedules. T. R.

* This work is partially supported by National Science Council of Republic of China under grants NSC92-2213-E-309-005.

Rayne et al. [15] propose a distributed meeting scheduling agent, called RETSINA Calendar Agent (RCal), that processes schedules marked up on the Semantic Web, and imports them into the user's personal information manager. In this paper, we apply the concept of personal ontology [2] to the meeting scheduling support system, and extend the four-layered object-oriented ontology [16] to *Personal Meeting Scheduling Ontology* (PMSO) for the *MS3*.

In the proposed ontology, we define four layers including *Domain Layer*, *Category Layer*, *Class Layer*, and *Instance Layer*, and four kinds of relationships including association, generalization, aggregation, and instance-of for meeting scheduling application domain. The association relation belongs to non-taxonomic relation. On the other hand, the generation, aggregation, and instance-of belong to taxonomic relation. Besides, the aggregation is a whole-part relationship.

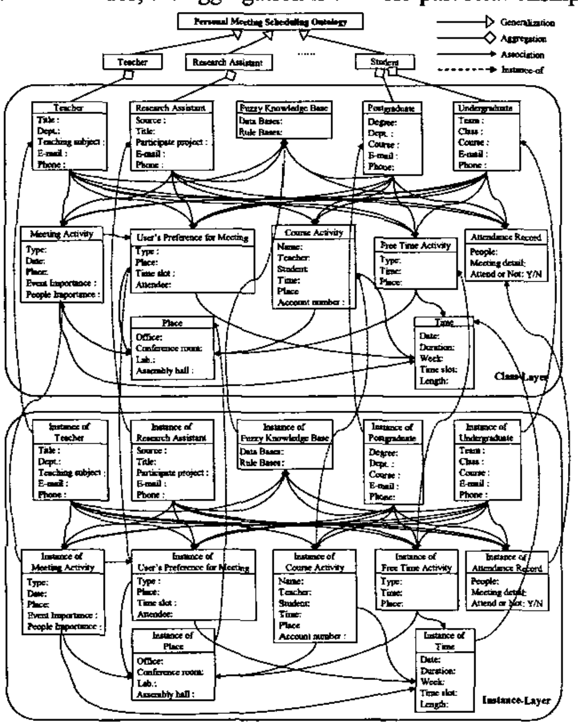

Fig. 1 The architecture of Personal Meeting Scheduling Ontology.

Fig. 1 shows the architecture of *Personal Meeting Scheduling Ontology (PMSO)*. In the MS3, the *Category Layer* is classified into "*Teacher Category*", "*Research Assistance Category*", and "*Student Category*". Fig. 2 shows a part of *Personal Meeting Scheduling Ontology (PMSO)* for "Decision Support & Artificial Intelligence (DSAI)" Lab. at Chang Jung University in Taiwan. There exist semantic relations between the concept pairs in the *Instance Layer*. For example, there are two *Concepts* "Chang-Shing Lee: Teacher" and "Meng-Ju Chang: Postgraduate" in *Instance Layer*. The association relationship, "*Participate*" and "*Guide*", are existed between the concept pairs, (*Teacher*, *Department Meeting*) and (*Teacher*, *Postgraduate*), respectively.

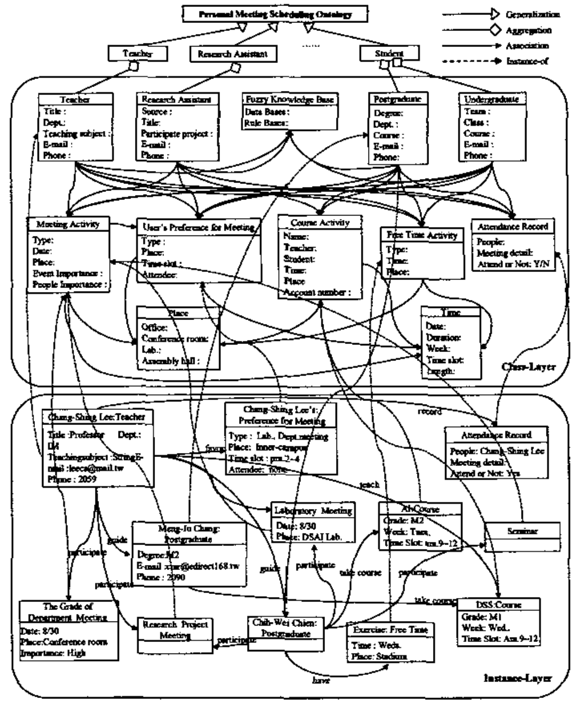

Fig. 2 A part of Personal Meeting Scheduling Ontology for DSAI Lab.

III. The Architecture of Meeting Scheduling Support System

A. The Architecture of Meeting Scheduling Support System

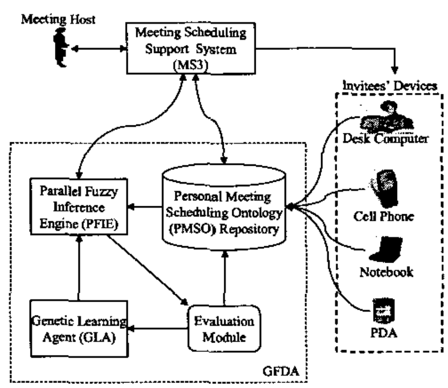

Fig. 3 The architecture of meeting scheduling support system.

Fig. 3 shows the architecture of *MS3*. At the beginning, the meeting host sends a meeting requirement to *MS3*, and *MS3* can obtain the invitees' schedules from the *Personal Meeting Scheduling Ontology (PMSO) Repository*. *MS3* then computes the common free time or removable working time for all invitees, and responds the computing results to *Parallel Fuzzy Inference Engine (PFIE)*. The *PFIE* performs fuzzy inference by utilizing the invitees' information, the fuzzy knowledge base of *PMSO*, and meeting information. In this architecture, the devices of end users include desk computer, cell phone, notebook, PDA, etc. Hence, the invitees can get the meeting information by various platforms. Furthermore, the attendance records of the invitees will be keep in the *PMSO* for generating invitee's personal fuzzy knowledge base. In the specific time, *Genetic Learning Agent (GLA)* will start to learn the fuzzy knowledge base for each user. In addition, an evaluation module is used to evaluate the learning process. The *Genetic Fuzzy Decision Agent* will be introduced in next section.

B. *Genetic Fuzzy Decision Agent*

The Genetic Fuzzy Agent consists of a parallel Fuzzy Inference Engine, a Genetic Learning Agent, an Evaluation Module and a Personal Meeting Scheduling Ontology Repository. We utilize a *Parallel Fuzzy Inference Engine* to infer attendance possibility of each invitee (See Fig. 4) [14][17].

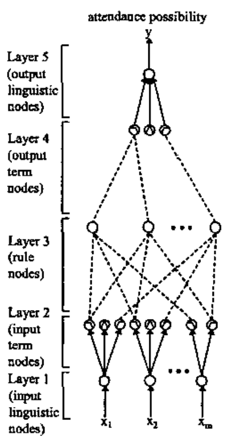

Fig. 4 The architecture of Parallel Fuzzy Inference Engine.

1) *Parallel Fuzzy Inference Engine*:

There are seven input fuzzy variables used in the *Parallel Fuzzy Inference Engine* for *MS3*. The *User_Priority (UP)* fuzzy variable denotes the importance of invitees. The *Meeting_Event_Priority (MEP)* fuzzy variable denotes the importance of the meeting. The length of meeting time is denoted as the *Meeting_Time_Length (MTL)*. *Meeting_Place_Preference (MPP)* fuzzy variable denotes the preference of meeting place for each invitee. The *Meeting_Subject_Preference (MSP)* fuzzy variable denotes the preference of meeting subject for each invitee. And the last two fuzzy variables are *Meeting_Time_Preference_1 (MTP1)*, and *Meeting_Time_Preference_2 (MTP2)*. *MTP1* fuzzy variable denotes the preference of meeting time for each invitee. For example, if one people likes to attend the meeting in the morning at AM10:00, and the meeting time is AM9: 30, then *PFIE* may infer the possibility of attending this meeting for the invitee is high. *MTP2* fuzzy variable is used to consider the work priority of each invitee's schedule. For example, if a meeting event priority for user A is low, and A has a high priority work to do at the same time, then A may not attend the meeting at this time. The output fuzzy variable in the *PFIE* is *Attend_Meeting_Possibility (AMP)*, it denotes the possibility of attending the meeting for each invitee. Fig. 5 shows an example of membership functions for fuzzy variable *UP*.

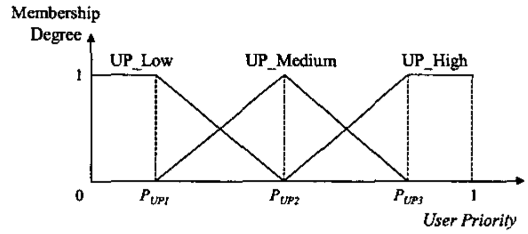

Fig. 5 An example of membership functions for fuzzy variable *UP*.

2) *Genetic Learning Agent*:

We modify the genetic learning method [18] and data-driven method [19] to generate the fuzzy knowledge base. The knowledge base includes the data base (DB) and rule base (RB). The DB consists of the number of linguistic terms and the parameters of membership functions of each fuzzy variable *UP, MEP, MTL, MPP, MSP, MTP1, MTP2* and *AMP*. The RB consists of fuzzy rules.

In the beginning, we encode the DB to be composed of two parts in each chromosome. One is the number of linguistic terms C_1, and the other is the parameters of membership functions C_2. The number of linguistic terms for each variable is stored into a vector C_1 as follows:

$$C_1 = (L_1, L_2, \ldots, L_m) \tag{1}$$

where L_i represents the number of linguistic terms of *i*-th fuzzy variable. The value of each L_i is restricted in the set {*2, 3, 4*}. On the other hand C_2 represents the parameters of each membership function. Fig. 6 shows the membership functions used in *Genetic Learning Agent*.

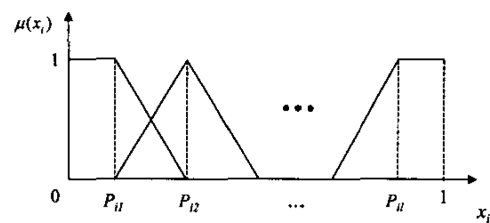

Fig. 6 The membership functions used in Genetic Learning Agent.

The parameters of membership function for each variable are stored into a vector C_2 as follows:

$$C_2 = (C_{21}, C_{22}, \ldots, C_{2m}) \tag{2}$$

where C_{2i} represents the parameters of membership functions of *i*-th fuzzy variable, and

$$C_{2i} = (P_{i1}, P_{i2}, \ldots, P_{il}) \text{ , and } P_{i1} < P_{i2} < \ldots < P_{il} \tag{3}$$

where P_{ij} represents the peak value of *j*-th linguistic term in the *i*-th fuzzy variable. Therefore, the chromosome

$C = C_1C_2$ represents to the membership functions of each fuzzy variable.

In addition, we utilize data-driven method to generate fuzzy rules from attendance records in *PMSO* and chromosomes. The attendance record is the set of input-output data pairs as follows:

$$\begin{aligned}&(x_1^1, x_2^1, \ldots, x_{m-1}^1, y_d^1),\\&(x_1^2, x_2^2, \ldots, x_{m-1}^2, y_d^2),\\&\quad\vdots\\&(x_1^n, x_2^n, \ldots, x_{m-1}^n, y_d^n)\end{aligned} \tag{4}$$

where x_i^j denotes the input values of *i*-th fuzzy variable for the *j*-th attendance record, y_d^j denotes the actual decision for the *j*-th attendance record. Each data pair will generate a fuzzy rule, but it is highly possible that there will generate some conflicting rules. Therefore, we assign a degree to each rule and take the *j*-th rule with maximum degree $D(Rule^j)$ to resolve conflicting rules as follows:

$$D(Rule^j) = \mu(x_1^j) \times \mu(x_2^j) \times \ldots \times \mu(x_{m-1}^j) \times \mu(y_d^j) \tag{5}$$

3) Evaluation Module:

In this paper, we use the mean square error (*MSE*) to be the fitness function as follows:

$$MSE = \frac{1}{2 \times T} \sum_{t=1}^{T} (y^t - y_d^t)^2 \tag{6}$$

where T denotes the number of the training data, y^t denotes the output of *PFIE* for the *t*-th training data, and y_d^t denotes the *desired output* of the *t*-th training data.

Next, we briefly describe the related genetic operators used in this paper as follows: (a) Selection: The selection probability calculation follows linear ranking. The selection probability is computed by using the nonincreasing assignment function; (b) Crossover: The standard crossover operator is applied over the two parts of the chromosomes. When C_1 is crossed at a random point, the corresponding values in C_2 are also crossed in the two parents; (c) Mutation: Two different operators are used in C_1 and C_2, respectively. In C_1 part, we random select the number of linguistic terms of the variable and change it to the immediately upper or lower value. In C_2 part, Michalewicz's nonuniform mutation operator [20] is used.

IV. Experimental Results

In order to evaluate the effectiveness of the proposed *GFDA* on *MS3*, we adopt the actual attendance records retrieved from *PMSO* repository as the experimental data. We evaluate the efficiency of genetic learning in the first experiment and adopt the latest 100 attendance records of five Lab. members. Then, 80 and 20 attendance records are adopted as the training and testing data respectively. Table I demonstrates the computing results. The definition of each column for Table I is described as follows:

1) User: the member of DSAI Lab.

2) Granularity: the number of labels for each fuzzy variable.

3) Rules: the number of rules.

4) MSE_{tra}: MSE over the training data set.

5) MSE_{tst}: MSE over the testing data set.

TABLE I
The Computing Results of 5 Members in DSAI Lab

User	Granularity	Rules	MSE_{tra}	MSE_{tst}
1	4 4 3 3 4 3 3 3	60	0.005226	0.010887
2	4 2 3 3 3 4 3 4	48	0.004999	0.004000
3	4 4 2 3 4 4 2 4	62	0.008749	0.037133
4	3 4 2 2 2 4 2 2	43	0.005536	0.029250
5	3 3 3 4 3 3 3 4	65	0.008808	0.041291

In the second experiment, we adopt one of the attendance records from DSAI Lab. members. Whenever the *MS3* increases five attendance records of the member, *GLA* will start to learn the *Fuzzy Knowledge Base* of personal ontology. Then, the *Fuzzy Knowledge Base* generated by *GLA* will assist *PFIE* to process more correct inference for the next meeting.

As shown in Fig. 7, the experimental result illustrates the frequency of the correct inference after learning. Obviously, the more attendance records that increased in the *MS3*, the more correct inference can be inferred by the *GFDA*. In this experiment, the times of correct inference are 83, and the total times for fuzzy inference are 95, hence the average correct rate is 83/95=87.37%. This experimental result shows that the *GFDA* can work effectively.

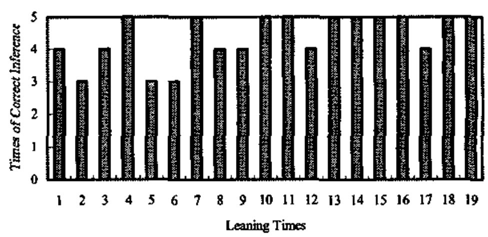

Fig 7 The Histogram of Times of Correct Inference.

V. Conclusions

A Genetic Fuzzy Decision Agent based on personal ontology for Meeting Scheduling Support System is proposed in this paper. The experimental results shows that the *GFDA* is feasible, efficient and usable for meeting scheduling support system. In the future, we will extend the capability of *GFDA* to be able to generate the semantic meeting information by personal ontology.

References

[1] V-W Soo and C-Y Lin, "Ontology-based information retrieval in a multi-agent system for digital library," *Proceedings of the 6th Conference on Artificial Intelligence and Applications*, Taiwan, 2001, pp. 241-246.

[2] M. N. Huhns and L. M. Stephens, "Personal Ontologies," *IEEE Internet Computing*, vol. 3, no. 5, pp. 85-87, Sep./Oct. 1999.

[3] J. Ferber, Multi-Agent Systems, ADDISON- WESLEY, New York, 1999.

[4] H. Chen, A. Housion, J. Nunamaker and J. Yen, "Toward Intelligent Meeting Agents," *IEEE Computer*, vol. 29, no. 8, pp. 62-70, Aug. 1996.

[5] S. Sen, "Developing an Automated Distributed Meeting Scheduler," *IEEE Expert*, vol. 12, no. 4, pp. 41-45, July/Aug. 1997.

[6] K. Sugihara, T. Kikuno and N. Yoshida, "A Meeting Scheduler for office Automation," *IEEE Trans. on Software Engineering*, vol. 15, no. 10, pp. 1141-1146, Oct. 1989.

[7] T. Haynes and S. Sen, N. Arora and R. Nadella, "An automated meeting scheduling system that utilizes user preferences," *Proceedings of the 1st international conference on Autonomous agents*, USA, 1997, pp. 308-315.

[8] W. S. Jeong, J. S. Yun and G. S. Jo, "Cooperation in Multi-Agent System for Meeting Scheduling," *Proceedings of the Region 10 conference on IEEE TENCON*, Korea, 1999, vol. 2, pp. 832-835.

[9] E. Mynatt and J. Tullio, "Inferring Calendar Event Attendance," *Proceedings of the 6th International Conference on Intelligent user Interfaces*, USA, 2001, pp. 21-128.

[10] A. Ashir, K. H. Joo, T. Kinoshita and N. Shiratori, "Multi-Agent Based Decision Mechanism for Distributed Meeting Scheduling System," *Proceedings of the International Conference on Parallel and Distributed Systems*, Korea, 1997, pp. 275-280.

[11] J. A. Pino and H. A. Mora, "Scheduling Meetings with Guests' approval," *Proceedings of the XVII International Conference of the Chilean on IEEE Computer Science Society*, Chile, 1997, pp. 182-189.

[12] F. Bergenti and A. Poggi, "An Agent-Based Approach to Manage Negotiation Protocols in Flexible CSCW Systems," *Proceedings of the 4th International Conference on Autonomous Agents*, Spain, 2000, pp. 267-268.

[13] C. Glezer, "A conceptual model of an interorganizational intelligent meeting-scheduler (IIMS)," *Journal of Strategic Information Systems*, vol. 12, no. 1, pp. 47-70, March 2003.

[14] C-S Lee and C-Y Pan, "An Intelligent Fuzzy Agent for Meeting Scheduling Decision Support System," *Fuzzy Sets and Systems*, vol. 142, no. 3, pp. 467-488, 2004.

[15] T. R. Payne, R. Singh and K. Sycara, "Processing Schedules Using Distributed Ontologies on the Semantic Web," *Workshop on Web Services, E-business and the Semantic Web*, Springer-Verlag Heidelberg, pp. 203-212, 2002.

[16] C-S Lee, Y-F Kao, Y-H Kuo and I-H Meng, "An Episode-based Fuzzy Inference Mechanism for Chinese News Ontology Construction," *The 7th World Multiconference on Systemics, Cybernetics and Informatics*, USA, 2003, pp. 453-458.

[17] C. T. Lin, C. S. G. Lee, "Neural-Network-Based Fuzzy Logic Control and Decision System," *IEEE Trans. on Computers*, vol. 40, no. 12, pp. 1320-1336, Dec. 1991.

[18] O. Cordon, F. Herrera, and P. Villar, "Generating the Knowledge Base of a Fuzzy Rule-Based System by the Genetic Learning of the Data Base," *IEEE Trans. on Fuzzy Systems*, vol. 9, no. 4, pp. 667-674, Aug. 2001.

[19] L. X. Wang and J. M. Mendel, "Generating Fuzzy Rules by Learning from Examples," *IEEE Trans. Systems Man, and Cybernetics*, vol. 22, no. 6, pp. 1414-1427, Nov./Dec. 1992.

[20] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs. Vienna, Austria: Springer-Verlag, 1996.