Cunha, L.M., Fuks, H. & Lucena, C.J.P. Setting Groups of Learners using Matchmaking Agents. in Uskov, V. (Ed), Proceedings of the IASTED International Conference on Computers and Advanced Technology in Education - CATE 2003, June 30 - July 2, Rhodes - Greece, pp 321-326, ISBN:0-88986-361-X, ISSN: 1482-7905, 2003.

SETTING GROUPS OF LEARNERS USING MATCHMAKING AGENTS

Leonardo Magela Cunha, Hugo Fuks, Carlos José Pereira de Lucena
Software Engineering Laboratory - Computer Science Department - Catholic University of Rio de Janeiro
R. Marquês de São Vicente, 225, Gávea – 22453-900 – Rio de Janeiro – RJ
Brazil
{leocunha, hugo, lucena}@inf.puc-rio.br

ABSTRACT

Nowadays, the complexity of work and the dissemination of the information and communication technologies value and make group work a potential allied for the organizations. The computer support offered under the umbrella of groupware systems is based on the research from Computer Supported Cooperative Work (CSCW). The support for the workers should be given since the group formation until its dissolution, going through the group work itself. In Software Engineering, multi-agent systems provide a properly level of abstraction for the treatment of complex and distributed problems. One of these problems is characterized by environments for group work and/or learning in groups. In this paper it is presented the study and how a multi-agent system was implemented in the AulaNet environment, which is a groupware for Web-based teaching and learning, to help group formation.

KEY WORDS

Computer assisted learning and instruction, collaborative software, software agents, group formation.

1. INTRODUCTION

Computer support for group learning is augmenting by the dissemination of computer technology networks. There is a great variety of new applications and possibilities, although one of the major challenges is the transition period required for transporting the applications, methods, methodologies and techniques of the real world to the virtual world. This is where the work in the fields of Computer Supported Cooperative Work (CSCW) and Software Engineering join up. Through the studies for group work and the methods for developing software it is possible to model and design software that is proper for supporting groups which is called groupware.

The access to large quantities of information distributed by the networks and the complexity of the real world has quickly changed how people learn. Agent technology has been presented as a promising strategy to be applied to the current challenges of modern educational environments that are increasingly more influenced by technologies such as the Internet and Artificial Intelligence. With the emergence of Distributed Artificial Intelligence [1] it can be seen the strengthening of the software agents concepts and also a redirecting of these concepts towards Software Engineering [2].

The objective of this paper is to present how a multi-agent system (MAS) was implemented for establishing groups of learners in classes during a course taught in the AulaNet environment [3], which is a groupware developed for Web-based teaching and learning.

The paper is structured as follows: the next section presents, based upon the group support that AulaNet offers, an investigation into software agent technologies and their applicability to the environment. In section 3, it is shown how a multi-agent system was used to support group formation within the environment it is also discussed some future work. For example, the creation of a federation of AulaNet servers, where the formation of groups with members (on different servers) of different classes within a course is possible. And last, final considerations about the work that was carried out are presented.

2. SOFTWARE AGENTS

For Jennings *et alli* [4], autonomous agents and multi-agent systems represent a new way of analyzing, designing and implementing complex software. The agent abstraction has a wide gamut of applications, ranging from the creation of personal assistants to air traffic control systems, electronic commerce and the group work support.

2.1 AGENTS FOR THE AULANET ENVIRONMENT

According to Aroyo and Kommers [5], agents can influence different aspects in educational systems. They supply new educational paradigms, support theories and can be very helpful both for learners and for teachers in the task of computer-aided learning. The application of agents in the educational sector comes about mainly in the form of personal assistants, user guides, alternative help systems, dynamic distributed system architectures, human-system mediators and others.

As a result of all of the changes that have taken place in the educational system, one now sees the increasing emergence of complex and dynamic educational infrastructure that needs to be efficiently managed. Corroborating this, new (types of) educational mechanisms and services need to be developed and supplied.

In particular these services need to satisfy a series of requirements such as personalization, adaptation, support for user mobility, support for users while they are dealing with new technologies, among others. Agents emerge to provide solutions for these requirements in a way that is more efficient when compared to other existing technologies [5].

Lees and Ye [6] believe that the application of the agent paradigm to CSCW potentially can: make the exchange of information more fluid among the participants of groupware systems (as decision-making systems), help in control of the process flows and also supply groupware interfaces. These ideas also are applicable to other domains, such as is the case of interactive learning.

The AulaNet environment was developed based upon CSCW umbrella and contains a range of different pedagogical opportunities as represented by its services. The following are highlights of possible elements of the environment that would benefit from the application of the multi-agent system paradigm: **Course content**: educational content could be dynamically linked through the use of pedagogical agents that determine the best sequence of presentation or method of exhibition to learners, for example based upon their profiles; **Asynchronous communication**: a greater exchange between the participants of a course could be obtained by using personal agents to filter messages [7] as well as the creation of link structures for messages related to the interests of the participants; **Synchronous communication**: the use of agents can assist the teachers to mediate online debates [8] in a manner that improves the learning that takes place during these synchronous exchanges; **Support of group work**: software agents can be used for the formation [9] and the support for group work, classes or even entire courses; Other possibilities are the use of agents to exchange content and to the use of virtual reality in distance learning courses.

As presented in the above list, a learning environment can become complex enough to instigate the use of the agent paradigm. With the development of technology based on mobility, through the use of personal assistants (PDAs) and cellular telephones, a new challenge also has been presented regarding the way of accessing and presenting educational content.

In terms of the geographical distribution of the participants, which is one of the most publicized advantages of the Web-based education environments, there is much to gain through the use of the agent paradigm. Through computation distribution there can be a significant reduction in the demand for computational resources on the servers and greater customization for the clients. How to configure the different personal agents to carry out all of these tasks, or how to configure a course to make use of all of these types of agents are questions that still are open and lacking in research.

The use of agents for the formation of groups within the AulaNet is the first attempt to incorporate a multi-agent system into the environment. Next, we present the justification for the use of the agent-oriented paradigm based upon some characteristics that an agent should present [10].

Autonomy: the use of the concept of autonomy permits the encapsulation of the interests of the participants of a course. For example, a learner's agent can give preference to participation in groups where the learner has a greater level of interest in the topics; **Interactivity**: to exist communication between the agents in order to discover partners it is necessary that they use a communication protocol; that is, it is necessary that they are interactive; **Collaboration**: the learner agents need to collaborate, supplying and receiving information about which ones will be the best partners in order to put together the group as intended by the teachers of the course; **Pro-activity**: the capacity to act with or without the need for user interference can be used to permit agents to suggest the formation of groups, based upon the identification of the interests of the participants, given topics and the activities related to a course. Another form of applying pro-activity would be the awareness of the specific needs of a group of learners and the suggestion of the formation of a workgroup; **Mobility**: the mobility of the agents would make it possible to use systems for forming groups from different servers, which is especially interesting in organizations that are far-flung geographically.

3. GROUP FORMATION ON THE AULANET

To design a multi-agent system (MAS) to support group formation it was necessary to model the learners of the AulaNet environment. We next present some considerations regarding the modeling of these learners, their respective implementation in the environment as well as concepts about how to find agents that supply the services or the information desired by other agents. Then we present the MAS for the group formation. Subsequently the concept of a federation of AulaNet servers is presented where it would be possible to create groups with members (on different servers) of different classes of the same course.

3.1. MODELING THE LEARNERS

According to Kay [11], in the first computer-assisted teaching environments the idea was to build "teachers" who could transmit knowledge to the learners. Currently, these types of environments are more geared up for exploration on the part of the learners, designing, building and using adaptive systems as tools. These environments also are being built to give greater responsibility to the learners regarding aspects of the learning process, and especially regarding control of its model, which is the central aspect in the adaptability of the tools.

For McCalla *et alli* [12], learner models may have a variety of purposes depending upon the type of knowledge that needs to be stored and processed. For them, the computation of all of the learner (sub)models of an environment can be computationally expensive and not always necessary. In the work cited four purposes are presented for a model: reflection, validation, matchmakers and negotiation.

For Kay [11], there are potential problems from the learners' point of view. One is the increase in the power of choice and control over the model. This could increase the learners' workloads or even turn into a distraction. In this case, the learners should take advantage of the moments such as the end of a course or a topic to evaluate and reflect upon their participation and the learning process. Another potential problem is incorrect data being supplied by the learners. The solution adopted in this work for that problem was to store the type of information learners are providing and the type the environment extracts.

To implement the learner models within the AulaNet environment, it was decided to use the specification defined by the Instructional Management Systems Global Consortium, Inc.(IMS) called IMS Reusable Competency Definitions (RCDs) [13]. In the final version of this specification IMS changed the name of RCD to RDCOE (Reusable Definition of Competency or Educational Objective) [14]. The work of Soltysiak and Crabtree [15] is recommended for a detailed review of the use of user modeling for agents.

In the specification of the IMS, the word competence is used in a general manner, including meanings such as skill, knowledge, task and learning outcome. It was thus seen that the best meaning for using in the environment was as knowledge, making it possible to record reusable knowledge definitions and, subsequently, the creation of learner models based upon these definitions.

According to the IMS, the reusable competence definitions provide a means for creating common understanding that appear as part of a career plan, prerequisites for a course or for educational objectives. They can be used for exchange between learning environments or human resources systems among others. The reusable definitions of competence were created for an exchange between machines although the information they currently contain is for human understanding. Basically, a definition contains a unique identifier and a non-structured textual description.

In order to make it possible for AulaNet participants to organize their models and create groups, the reusable competence definitions were associated with the courses, the class specific contents and the participants.

It is left up to the teachers of a course (Coordinators and Teacher Co-Authors) to associate a set of RCDs with their specific courses and/or activities. For example a course about databases could be related with these RCDs: Database Architecture; Database Models; Relational Model and SQL (Structured Query Language), among others.

The participants are allowed to provide information about the set of RCDs of the server or of a specific course. The information that will be supplied by the participants is a grade or concept about how much they know about or are interested in the RCD; it also is possible to insert a comment about the attribution of the concept they are making.

Considering the need to distinguish between the information of the learner model generated by the environment and that supplied by the learner, it was decided to use the "Type" attribute in the records for the RCDParticipant table. There are three dimensions or types of information of an RCD in the AulaNet: Interest, Qualification and Competence.

Interest is the information that is supplied by the learners about their level of interest regarding a given RCD. Qualification is also information that is supplied by the learners regarding their experience with a given RCD. And, last, Competence is the information generated by the environment based on the evaluation of a learner by the teachers for a given activity associated with an RCD.

3.2. MATCHMAKING AND BROKERING CONCEPTS

One of the problems in a multi-agent system project is the way to discover which agents have a specific piece of information or skill. Many environments and specifications define agents that offer white page services, which are directories of agents; and yellow page services, which are directories with the features offered by the agents. These are the cases of FIPA-OS [16] and SACI [17]. Some agent communication languages such as KQML also offer special performatives for this behavior, such as Recruit, Broker and Forward [18]. For Ivezic *et alli* [19], matchmakers and brokers work as intermediate agents between agents that supply services and agents that need these services.

For Foner [20], the use of centralized architecture for matchmaking can be valid, for example, in cases where the agents are unable to discover each other and request that a central entity provide a solution for the problem. However, there also are disadvantages to this type of architecture - for example, its tolerance for mistakes is low since it has a central point where attacks or even incidents may occur. Another disadvantage is the potential computational bottlenecks that could arise as a result of the increase in the number of agents.

He also states that the use of some techniques that already have been applied in networks, such as the hierarchical organization of entities (such as on the Internet's domain name systems and newsgroups) does not reduce problems like the computational bottleneck. This occurs because of the non-existence of a standard hierarchy. For example, why would the interests of one agent come ahead of another? In order to propose a solution to these problems, Foner used some ideas based on computational ecology [21]. The main ideas are: i) To compare agent information in a decentralized manner (peer-to-peer); ii) To use references from one agent for the others and an algorithm that remember hill-climbing to find other partners to; iii) Build clusters or clumps of agents with common interests, and; iv) To use these clusters of agents with common interests to present users to each other; v) To use a persistent agent that is active for long periods of time and not an agent that the user initiates, obtains a result and then deactivates. In this way, more agents can be consulted and more appropriate clusters can be formed. Next, we present a multi-agent system that helps teachers in forming groups of learners.

3.3. A MULTI-AGENT SYSTEM FOR GROUP FORMATION

It is possible on AulaNet to create a course, and within the course it is possible to have different classes, each with their respective mediators. The environment was prepared for the use by groups within classes using the Tasks service. Furthermore, it is possible to form groups manually. An agent is found in an environment and interacts with this environment. In comparison with the environment concept, in the SACI tool there is a concept of society where the agents are united and can communicate through KQML agent communication language using their identities. An identity is a name that uniquely identifies an agent within its society [22].

In the SACI tool, an agent's life cycle is as follows. The agent enters a society and receives an identification. In this society it can send or receive messages from other agents of the same society, announce its skills to the society and, finally, it may leave the society, thereupon losing its identity.

The creation of groups in a class is related to the dynamics of the course, that is, the person responsible for creating groups is the mediator of the class. The mediator requests that a mediating agent (AgMediator) create groups in his class, supplying the information that is necessary for this purpose, such as the number of groups, what are the RCDs that must be analyzed in the learner models, what is the level of difference between learners and if the learners can be repeated in the groups, among other data.

In order to determine if the maintenance of agents representing learners in execution during a long period of time is feasible or not, it would be necessary to analyze the average number of participants on the AulaNet servers. Thus, it was decided to permit that the AgMediator enter into the society of the class that requested the creation of the groups and instantiate an agent for forming groups (AgGroups) that, for its part, instantiates the agents for all of the learners (AgLearner).

On AgGroups' initialization, it receives from the AgMediator which RCDs and respective aspects it should represent. The AgLearner agents are initialized and enter into the society, publishing which learner they represent and if they can satisfy the needs of the RCDs and aspects requested by the AgGroups. The AgMediator then requests that the AgGroups form the groups, presenting the goal containing the levels of difference of each RCD and its respective aspects. The AgGroups search the society for the AgLearners that can respond to the formation of groups and passes along the request of the AgMediator to the AgLearner. Those that are able to form groups collaborate among themselves to suggest groups that satisfy the request.

It can be seen that the formation of K groups with n participants satisfying a given degree of difference between the participant models is a NP-Complete problem. The 3-Dimensional matching problem [23] is a problem that can have polynomial time in the case the elements are repeated in the matchings that are carried out, or if the dimension is less than 3. However, in our case, it is possible that we will have numbers of groups greater than or equal to 3.

One heuristic adopted to solve this problem allows agents to pass along references from other agents they know, thus reducing the quantity of messages exchanged. Another heuristic adopted was the use of a greedy strategy [24]. At the moment an AgLearner succeeds in forming the groups that were requested in accordance with the parameters received from AgGroups, it stops collaborating, so informing the AgGroups and leaving the society.

The AgGroups filter these groups, if there are repeated groups. AgGroups supplies this information to the AgMediator. For its part, the AgMediator gets back the information to the mediator who requested the formation of the groups through the AulaNet's group formation interface. In case a timeout occurs during the attempt to form the groups determined by the AgMediator, it requests that the AgGroups and the AgLearner give up carrying out the formation, halting collaboration and leaving the society. Each AgLearner knows the model of its learner. This model is composed of the learner's competence, interest and qualification aspects from the RCDs. Upon requesting the formation of a group, the AgGroups define which aspects of the model must be taken into consideration for negotiation by the AgLearners.

For example, a mediator might want learners, independent of their preferences, to form groups where all have the same prior knowledge about a given RCD. He should define that the interest and competence aspects will not be taken into account in the negotiation and that the degree of difference be minimal in the qualification aspect. Other combinations of aspects may be created, assisting the mediator to apply different tactics for forming groups. The collaboration model for learner agents was inspired in the matchmaking algorithm found in [20].

One of the questions that arise from the use of software agents is: what do these agents learn? Do they learn from their interaction with the users or with the other agents?

According to the way the system was designed and implemented, it is not possible to learn more about the interaction between agents unless they continue to execute for a longer amount of time. Regarding the learners, the agents could learn more about them through the changes that occur in their competence, as of their use of the environment. Why do the AgLearners collaborate? Would it not be better to create a central agent for matchmaking? The decision to use a distributed structure was based on [20] who states that for a reduced number of agents the computational gains are not considerable in relation to the central structure. In classes with few learners the centralized structure perhaps would be more efficient; however, this work is also concerned with future versions of the AulaNet environment and, thus, support for the formation of inter-class and inter-server groups is also important. For example, in Figure 1, it is possible to foresee a future step for the formation of inter-class groups within the same course.

3.4. FEDERATION OF AULANET SERVERS

For example, if a coordinator wants to create groups of learners for co-authorship, he can request that the mediators of the course offer him groups or learners with particular characteristics found in some RCDs. These groups could be used directly for attribution to the desired activity. Or the agents representing the elected learners could be asked to move to the coordinator's society and collaborate in the formation of groups with members from different classes.

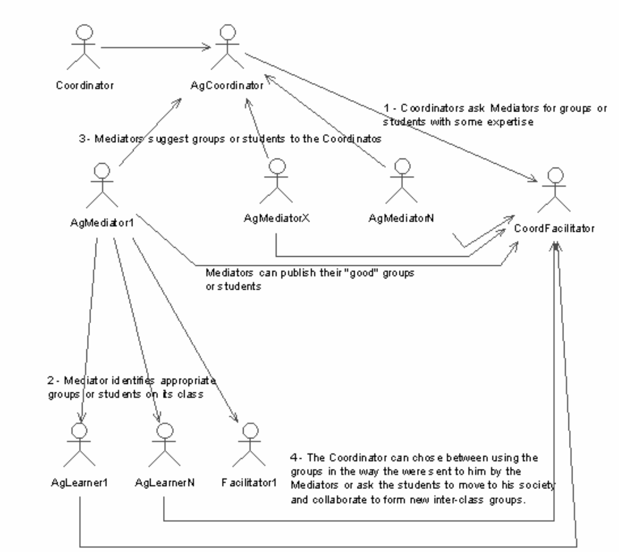

Figure 1 - Group formation using software agents

One of the purposes of the AulaNet project is the creation of server federations, where it would be possible to participate in courses hosted in different servers. Silva *et alli* [25] propose a framework for the interoperability of educational content using IMS specifications called ContentNet.

A federation of servers in this paper is a set of servers with a single identification, for example the server's IP address, that is capable of exchanging information about the learners and their interaction in different courses - and not only educational content, as is the case of ContentNet.

A course in a federation of servers is created in one server and has classes in it or in other servers. The interaction of the learners is stored in the class servers and the content relating to the entire course is stored in the course server. Thus, this server must be capable of handling a larger number of accesses.

Based upon a federation of servers, the use of software agents for forming groups, or even for the customization of content, stands out as an appropriate solution. Especially for the formation of groups, we fear that the storage of centralized information about the learners could overload the course server, whereas maintaining class servers means the knowledge remains distributed and it is easily accessible by the agents. The agents can travel around and supply information or negotiate with the agents of other learners in order to meet goals that have been established, for example, by the coordinator of the course.

4. FINAL CONSIDERATIONS

The use of workgroups within the AulaNet environment provides a view of the needs of the learners and teachers and collaborates with a contextualized survey. Moreover, the concern with using IMS standards and the groupware approach adopted by the system makes it possible to have easy similarity between the educational world and the workplace of the market job. The use of the AulaNet as a tool for supporting work is being researched and looks promising. The resemblance of a course with a project, of a class with a team, of a learning group with a workgroup, is a very stimulating view in this research process. The use of learner modeling in the AulaNet environment also satisfies one of the concerns of modern organizations, which is knowledge management.

By providing support for forming and working in a group, the software agents also are supporting project learning and collaborative learning. When the groups that have been formed demonstrate a high degree of heterogeneity, interdisciplinary attitude and practice also may benefit. In the same way that these aspects are influenced, it is necessary to point out that professional skills can be developed and influenced through group work, such as the capacity for self-monitoring, listening, presenting new ideas and persuasion, among others.

5. ACKNOWLEDGEMENT

The AulaNet project is partially financed by the Fundação Padre Leonel Franca, by the Ministry of Science and Technology through its Program of Excellence Nuclei (PRONEX) grant 76.97.1029.00 (3366) and by the

Software Engineering for Multiagent Systems project (ESSMA) grant 552068/2002-0. And also through individual grants awarded by CNPq – Brazilian National Research Council to: Carlos José Pereira de Lucena 300031/1992-0, Hugo Fuks 303055/02-2 and Leonardo Magela Cunha 140895/2002-7.

REFERENCES

[1] G. Weiss (ed.), *Multiagent systems: a modern approach to distributed artificial intelligence* (Cambridge, MA: MIT Press, 1999).

[2] N. Jennings, & M. Wooldridge, Agent-Oriented Software Engineering, *In*: J. Bradshaw (ed.), *Handbook of Agent Technology* (AAAI/MIT Press, 2000).

[3] C. J. P. Lucena, H. Fuks, R. Milidiú, C. Laufer, M. Blois, R. Choren, V. Torres, & L. Daflon, AulaNet: Helping Teachers to Do Their Homework, *Multimedia Computer Techniques in Engineering Education Workshop*, Graz, Austria: Technische Universitat Graz, 1999.

[4] N. Jennings, K. Sycara, & M. Wooldridge, A Roadmap of Agent Research and Development, *Autonomous Agents and Multi-Agent Systems*, *1*, Kluwer Academic Publishers, 1998, 7-38.

[5] L. Aroyo, & P. Kommers, Preface - Intelligent Agents for Educational Computer-Aided Systems. *Journal of Interactive Learning Research, 10* (3/4), 1999, 235-242.

[6] B. Lees, & Y. Ye, Preface of the Proceedings of ASCW01 – Workshop of Agent-Supported Cooperative Work, *In: The Fifth International Conference on Autonomous Agents,* Montreal, CA, 2001.

[7] P. Maes, Agents that Reduce Work and Information Overload. *Communications of the ACM*, *37*(7), 1994, 31-40.

[8] P. A. Jaques, A. F. Andrade, M. C. Moraes & M. C. Móra, Uma Arquitetura de Agentes para a Análise Qualitativa da Interação em Ambientes de Educação à Distância. *In: Simpósio Brasileiro de Informática na Educação*, Maceió, Alagoas, Brazil, 2000, 282-289 [In Portuguese].

[9] C. J. M. Olguín, A. L. N. Delgado, & I. L. M. Ricarte, An Agent Infrastructure to Set Collaborative Environments. *IFETS/IEEE Educational Technology & Society. 3*.(3), 2000, 65-73.

[10] OMG (2000). Object Management Group. Agent Technology – Green Paper. Version 1.0, 2000, Object Management Group.

[11] J. Kay, Learner Control. *User modelling and User Adapted Interaction*, *11*, Kluwer Academic Publishers, Netherlands, 2001,111-127.

[12] G. Mccalla, J. Vassileva, J. Greer, & S. Bull,. Active Learner Modeling, *In: Gautier, Frasson & Vanlehn (Ed.). Proceedings of ITS'2000*, Springer LNCS 1839, 2000, 53-62.

[13] IMS RCD (2001) Instructional Management Systems Global Consortium, Inc. (online) IMS Reusable Competency Definition Specification. Version .01, approved by the IMS Technical Board in February 2000 <http://www.imsproject.org/rcd/index.html> (Consultation: 2001, July 3rd).

[14] IMS RDCOE (2003) Instructional Management Systems Global Consortium, Inc. (online) Reusable Definition of Competency or Educational Objective Specification. Version 1.0, approved by the IMS Technical Board in October 2002 <http://www.imsglobal.org/competencies/index.cfm> (Consultation: 2003, January 31st).

[15] S. Soltysiak, B. Crabtree, Knowing Me, Knowing You: Practical Issues in the Personalisation of Agent Technology. *In: International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology - PAAM'98*. London, UK, 1998, 467-484.

[16] S. Poslad, P. Buckle, & R. Hadingham, The FIPA-OS Agent Platform: Open Source for Open Standards. *In: International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology - PAAM 2000,* Manchester, UK, 2000, 355-368.

[17] J. F. Hübner, & J. S. Sichman, SACI: Uma ferramenta para Implementação e Monitoração da Comunicação entre Agentes. *In: Brazilian Symposium on Artificial Intelligence (SBIA)/7th Ibero-American Conference on AI (IBERAMIA),* Atibaia, São Paulo, 2000, 47-56 [In Portuguese].

[18] K. Decker, M. Williamson & K. Sycara K, Matchmaking and Brokering, *In: The Second International Conference on Multi-Agent Systems (ICMAS-96)*, Kyoto, Japan, 1996.

[19] N. Ivezic, M. Barbacci, D. Libes, T. Potok, T. & J. Robert, An Architecture Analysis of Agent-based Systems. *In: Workshop Proceedings of ESAW'00. Engineering Societies in the Agents' World*, 2000.

[20] L. Foner, A Multi-Agent Referral System for Matchmaking. *In: The First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM 96)*, London, UK, 1996, 245-262.

[21] B. A. Huberman (ed.) *The ecology of computation* (Elsevier Science Publishers B. V., 1988).

[22] J. F. Hübner, & J. S. Sichman, SACI Programming Guide. Version 0.8, Universidade de São Paulo, Brazil, 2001.

[23] M. Garey, & D. Johnson, *Computers and intractability – A guide to the theory of NP-completeness* (W.H. Freeman and Co., San Francisco, USA, 1978).

[24] T. H. Cormen (Ed), C. E. Leiserson, R. L. Rivest, & C. Stein, *Introduction to algorithms* (MIT Press. 2nd edition, 2001).

[25] V. Silva, C. J. P. Lucena, & H. Fuks, ContentNet: A Framework for the Interoperability of Educational Content Using Standard IMS. *Computers & Education Journal*,*37*(3/4), Elsevier Science Press, 2001, 273-295.