IBIMA Publishing
Journal of e-Learning & Higher Education
http://www.ibimapublishing.com/journals/JELHE/jelhe.html
Vol. 2013 (2013), Article ID 261432, 9 pages
DOI: 10.5171/2013.261432

A Multi-agent System for Resource Allocation to Education Programmes in Higher Education Institutions

Constanta-Nicoleta Bodea and Radu-Ioan Mogos

Economic Informatics and Cybernetics Department, Academy of Economic Studies, Bucharest, Romania

Abstract

Universities are often facing with the problem of resource allocation to the education Programmes they provide. The paper presents a multi-agent based solution, which is developed for a big Romanian university, delivering hundreds of programmes every year. The multi-agent system receives inquiries for different resources and performs a transparent negotiation process between programmes and the resources managers, in order to find a good solution for the resources allocation problem. There is also taken into account a predefined set of criteria, which have to be fulfilled in order to allocate the resources. Finally, the solution is tested by several people. Their opinions were the subject of a questionnaire. This paper provides evidence that a multi-agent system based solution can be appropriate to solving the resource allocation problem. The findings call for further specification and more resources may be used to develop the system.

Keywords: Multi-agent systems, Negotiation process, Resource allocation problem, Petri networks.

Introduction

In the research study by Dolgov and Durfee (2006), the resource allocation problem (RAP) is described as being the allocation process of the available resources to the clients (scheduled tasks or agents, either cooperative or self-interested) in a way which maximizes the global utility. As an optimization problem, RAP is a multi-objective and over-constrained problem. Even if the optimality is rarely needed in the real-world situations, the optimal solution is expected by the users. For solving RAP, a centralized or distributed approach could be undertaken. Ridder et al. (2012) mentioned that a distributed constraint optimization problem (DCOP) is more adequate in a large problem space. In this case, the problem is split into agents, each of them having a specific set of variables and constraints as well as local optimization criterion. The goal is to find a feasible solution with the highest ranking by all agents.

Developing academic class timetable is a real-world RAP. This problem involves the allocation of students, teachers and rooms, within certain restrictions (related to the regulations, proper utilization of resources, and satisfaction of people's preferences), for executing education activities in specific time-slots. Effective timetables for academic courses delivery are crucial for the efficient utilization of university resources and for ensuring the student satisfaction. Timetable is usually developed manually, based on valid solutions applied in the previous years. This approach does not even guarantee finding an optimal solution, even a valid one. Silvia et al. (2004) suggested that the problem complexity is increasing when multiple

Contact author: Constanta-Nicoleta Bodea E-mail: bodea@ase.ro

criteria and types of classes are considered. Computational intelligence methods are successfully applied to solve the academic class timetable problem. The earlier works on the class timetabling problem, using neural network, are found in (Gislen et al., 1992) and (Looi, 1992). Carrasco and Pato (2004) define two neural network architectures to solve the timetable problem, one based on continuous potts neurons and the other on discrete winner-take-all neurons. A constraint solver is defined by Rudova and Murry (2003) for developing a demand-driven academic timetable. It allows the hard constraint propagation together with the preference propagation for the soft constraints.

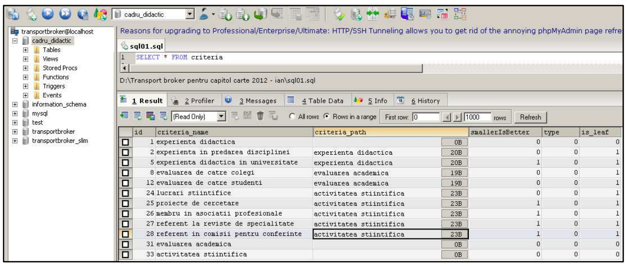
Multi-Agent Systems and Agent-Based Methodologies

Bodea et. al. (2011) agree that the multi-agent system (in short, MAS) represents an important paradigm in the field of distributed computation. Promoting the collaborative approach in complex problem solving, MAS is a natural implementation of the distributed constraint optimization (DCOP) approach. When MAS is adopted, a collaborative planning and participatory environment in decision making can be applied, allowing the involvement and participation of multiple experts and stakeholders (Mendoza and Prabhu, 2003). The MAS is composed by several agents that communicate and work together in an environment (application field or problem). Each of these agents is designed to have a goal. According to this goal it will act in a specific way and at a specific moment of time. For obtaining resources, the agent's utility should be calculated. The utility is usually defined by what the agent can achieve using these resources, which is a non-trivial task, because the agent's actions might have long-term and nondeterministic effects and they are conditioned by the resources it will obtain. This leads to cyclic dependencies, with no possibility to calculate parameterized solutions. Determining classes of utility functions (deVries & Vohra, 2003), iterative algorithms for resource allocation, preference elicitation (Sandholm & Boutilier, 2006), and concise languages for expressing agents' preferences (Boutilier, 2002) there are only some solutions for addressing this issue.

Implementing MAS is a complex and laborious process and the role of the applied methodology is essential. For this reason, a lot of research is done in the agent-oriented methodologies field, revealing that methodologies are even more important in developing MAS than in another software engineering projects (Brinkkemper, 1996), (IEEE Standards Board, 1990), (Sturm, Shehory, 2003). Agent-based methodologies consider the enterprise as divided into sub-organizations where agents play one or more roles, interacting with each other. Concepts as "role", "social dependence" and "organizational rules" are used not only to model the system environment, but also to model the system itself. One of the most important aspects addressed by an ABM is the description of interactions between agents, simulating the dependencies between agents and their roles in the system. Each methodology must have a high enough degree of abstraction in agent modeling. This is why the object-oriented methodologies are not suitable for developing multi-agent systems. An agent-oriented methodology should be in focused on agents, the roles they have made in systems and interaction protocols.

A Multi-Agent System Development for Resources Allocation in Universities

Let's consider that the resources allocation is made based on a predefined set of criteria, which have to be fulfilled in order to allocate the resources. How better the criteria must be fulfilled is interactively decided during a negotiation process. If the expectation level of the client (programme/course) is too high, it is a high risk to not be able to allocate resources. Contrary, if the expectations are too low, the risk is to get low quality resources. This is why, a negotiation process, running several negotiation rounds is a better solution than other approaches. The figure 1 presents the criteria for the resource *Professors*.



Reasons for upgrading to Professional/Enterprise/Ultimate: HTTP/SSH Tunneling allows you to get rid of the annoying phpMyAdmin page refre

```
1  SELECT * FROM criteria
```

D:\Transport broker pentru capitol carte 2012 - ian\sql01.sql

	id	criteria_name	criteria_path		smallerIsBetter	type	is_leaf
□	1	experienta didactica		0B	0	0	0
□	2	experienta in predarea disciplinei	experienta didactica	20B	0	0	1
□	5	experienta didactica in universitate	experienta didactica	20B	1	0	1
□	8	evaluarea de catre colegi	evaluarea academica	19B	0	0	1
□	12	evaluarea de catre studenti	evaluarea academica	19B	0	0	1
□	24	lucrari stiintifice	activitatea stiintifica	23B	0	0	1
□	25	proiecte de cercetare	activitatea stiintifica	23B	1	0	1
□	26	membru in asociatii profesionale	activitatea stiintifica	23B	1	0	1
□	27	referent la reviste de specialitate	activitatea stiintifica	23B	1	0	1
□	28	referent in comisii pentru conferinte	activitatea stiintifica	23B	1	0	1
□	31	evaluarea academica		0B	0	0	0
□	33	activitatea stiintifica		0B	0	0	0

Fig. 1. The Allocation Criteria for Professors

The criteria applied in the negotiation process for acquiring the "professor" resources are teaching experience (in the university and in the programme/course topic), academic evaluation (the results of the evaluation made by the students and by the peers), and the scientific activity (scientific papers, research projects, member of the professional associations, reviewer at scientific journals and conference). All these criteria are used for implementing soft constraints in the timetable engine, as preferences of programmes/courses in terms of resources (professors and teaching rooms). We assume that the resource inquirers are weakly-coupled, meaning that the programmes/courses only interact through the shared resources, and once the resources are allocated, their state transitions are independent. While this allocation assumption introduce a limitation in our approach, it also allows us to address more efficient the non-cooperative situations and to avoid the state space explosion.

The methodology applied for the development of the system is TROPOS. According to TROPOS, the following development phases have to be executed, sequential and iterative: the requirement analysis, the system description in relationship with the operational environment, the design of overall architecture, the design of detailed architecture and the system implementation.

The Requirements Analysis

The environment of the system contains the following entities (figure 2):

- Inquirers, asking for resources and representing different programmes/courses ;

- Resource managers, owning the resources in the university (heads of department/char);

- Resources (professors and rooms for didactical activities)

- Negotiator, the entity proposed by the organization to find solution for a good utilization of the resources. We consider that the negotiator is representing the organization interest for a better utilization of the resources.

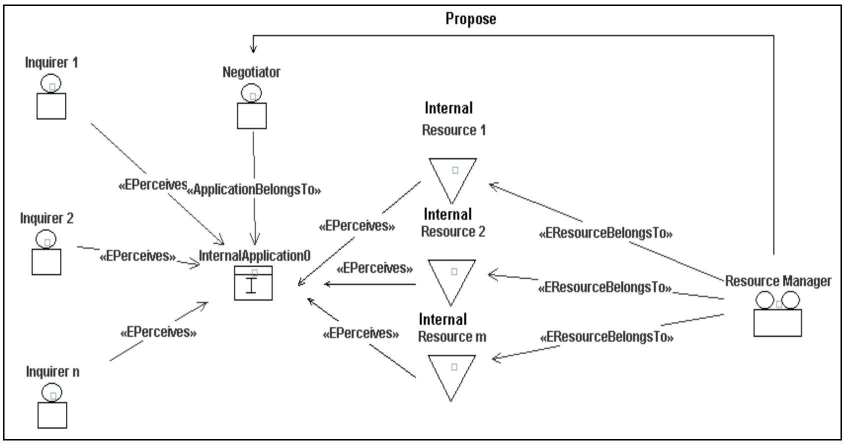


Fig. 2. Environment Diagram

The System Description in Relationship with the Operational Environment

Figure 3 presents, in hierarchical form, the goals of the participants in the resource allocation process.

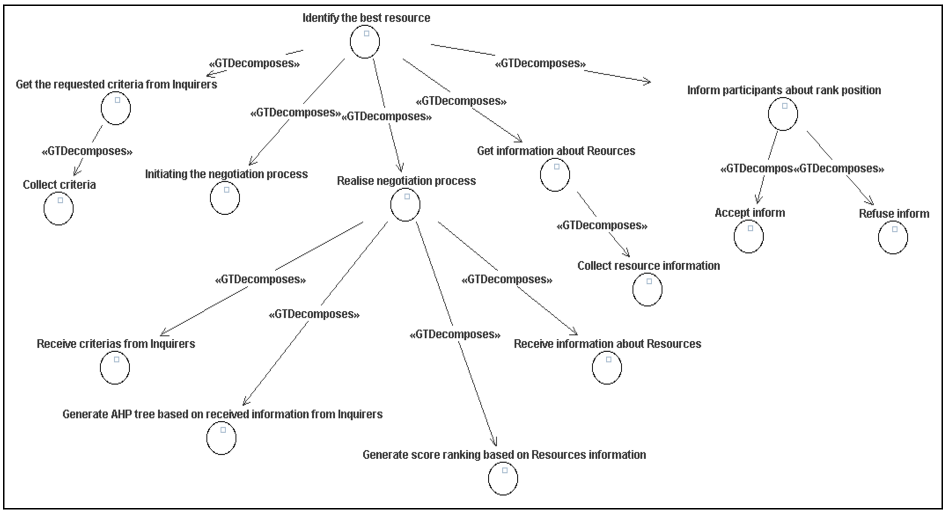

Identify the best resource
«GTDecomposes»
«GTDecomposes»
Get the requested criteria from Inquirers
Inform participants about rank position
«GTDecomposes»
«GTDecomposes»
«GTDecomposes»
«GTDecomposes»
Get information about Reources
Collect criteria
Initiating the negotiation process
Realise negotiation process
«GTDecompos
«GTDecomposes»
Accept inform
Refuse inform
«GTDecomposes»
Collect resource information
«GTDecomposes»
«GTDecomposes»
«GTDecomposes»
«GTDecomposes»
Receive criterias from Inquirers
Receive information about Resources
Generate AHP tree based on received information from Inquirers
Generate score ranking based on Resources information

Fig. 3. The Multi-Agent System Goal Tree

The Design of Overall Architecture

The overall architecture is defined in term of sub-systems, relationships and dependences.

Figure 4 presents the model of roles and agents. The correspondences between agents and roles are presented in Table 1.

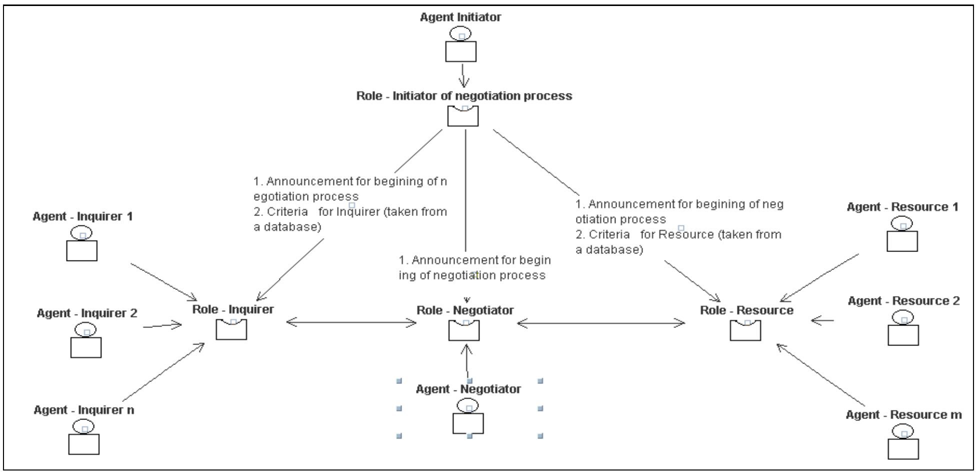

Agent Initiator
Role - Initiator of negotiation process
1. Announcement for begining of n egotiation process
2. Criteria for Inquirer (taken from a database)
1. Announcement for begining of neg otiation process
2. Criteria for Resource (taken from a database)
1. Announcement for begin ing of negotiation process
Agent - Inquirer 1
Agent - Inquirer 2
Agent - Inquirer n
Role - Inquirer
Role - Negotiator
Role - Resource
Agent - Negotiator
Agent - Resource 1
Agent - Resource 2
Agent - Resource m

Fig. 4. The Roles and Agents Model

.

Table 1. The Correspondences between Agents and Roles

Agent	The associated role
Inquirer Agent	Inquirer
Resource Agent	Resource
Negotiator Agent	Negotiator
Initiator Agent	Initiator of negotiation process

The overall architecture of the UNIRA system is presented in figure 5. The exchange of messages between agents is presented in Table 2.

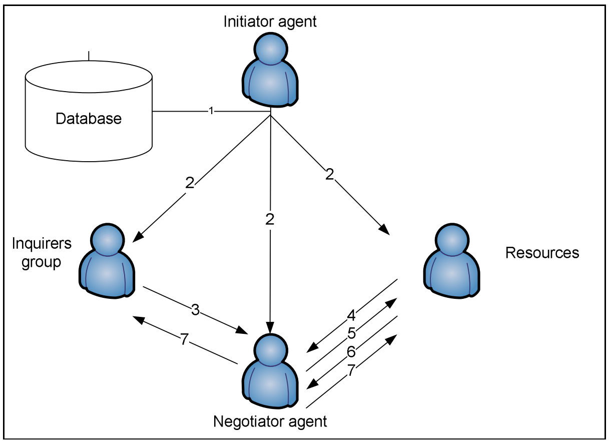


Fig.5. The Overall Architecture of the UNIRA System

Table 2. The Message Exchange

Agent name	Send message to	Content of the message
Inquirer agent	Inquirer agent	(2) Negotiation process is started and other information
	Negotiator agent	(2) Negotiation process is started and other information
	Resource agent	(2) Negotiation process is started
Inquirer agent	Negotiator agent	(3) Allocation criteria with correspondent values
Resource agent	Negotiator agent	(4) An offer is sent (6) An improved offer is sent
Negotiator agent	Inquirer agent	(7) The winning resource agent is announced
	Resource agent	(7) The winning resource agent is announced

The Design of Detailed Architecture

Due to their ability to model dynamic systems, the object-oriented Petri nets are chosen to define the multi-agent system detailed architecture. The detailed architecture contains the main components (agents), the interconnections between components, and the inputs and outputs for each of them. To get clarity, the agency is represented only by inputs, outputs and a generic transaction including all places and transitions. Figure 6 presents a multi-agent architecture based on object-oriented Petri net and figure 7 presents the detailed architecture of the system.

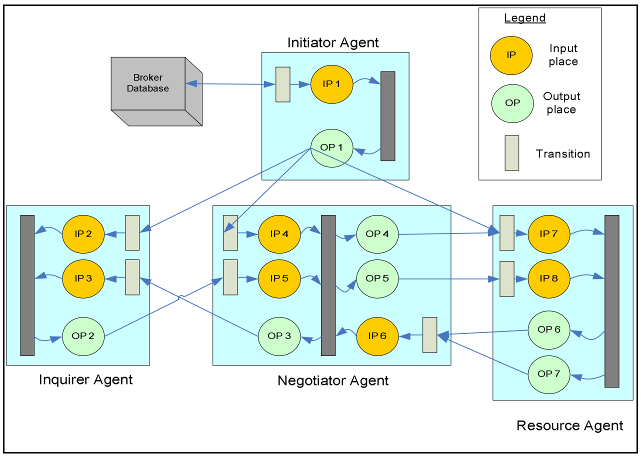


Fig. 6. The Architecture Based on Object-Oriented Petri Nets

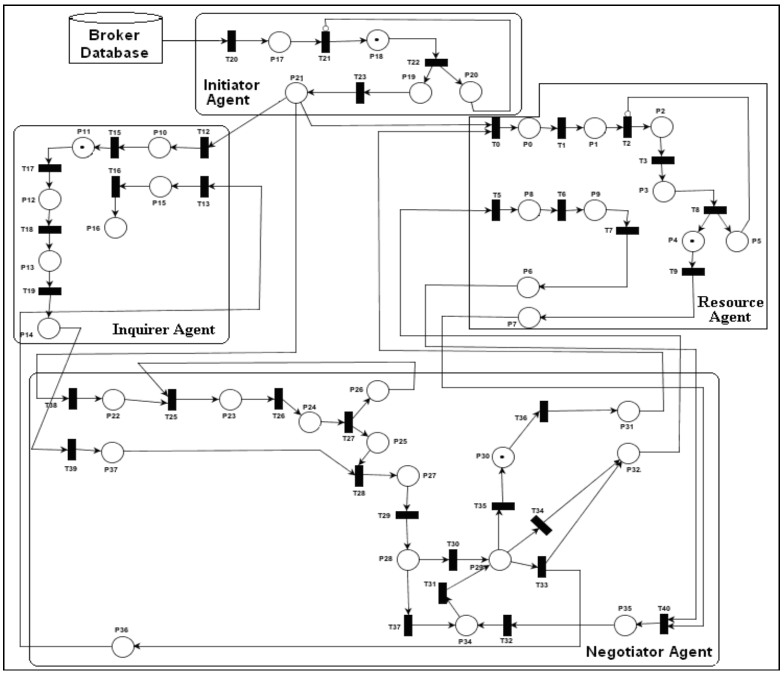


Fig.7. The Detailed Architecture of the System

The negotiation agent, using the information sent by the Inquirer agents create an AHP tree, which is be used for the scores calculation, for each offer issued by resource agents.

System Validation

The system validation starts with the most important characteristics of the system, assuring the approach originality. The originality of the system relies mainly on the negotiation process, executed in a very transparent way. In order to validate the user satisfaction in using the system, a questionnaire was designed and applied to 50 persons, representing 25 applicants, 20 resource managers, and 5 negotiators. The questionnaire has the following questions: how intuitive is the interface of your agent (1 - very little intuitive, 5 - very intuitive); have you identified factors which you consider unnecessary for the interface? (1-No, 0-Yes) (max: 25); did you experience difficulties in the execution? (1-No, 0-Yes); you needed advanced knowledge in order to work with the application? (1-No, 0-Yes); did you experience errors in the communication with the server? (1-No, 0-Yes); how do you rate the quality of the allocated resources in relation to your request? (1-totally inadequate, 5 - totally appropriate) – only for the Inquirer agent ; how do you consider the ranking position that you have placed (1-completely unacceptable, 5 - total acceptable) – only for the resource agent and how do you rate the communication with other entities? – only for the negotiator Agent.

The degree of interface understanding (the question one) is lower to the applicants (only 84%), because the grading criteria are made by comparison (the elements belonging to the same leaf node are evaluated comparing them to all others). This approach produces confusion for the users working for the first time with the application. The resource managers get a higher score (around 90%), mainly because the values for the resource evaluation are directly input in the system. The negotiators consider that the interface as intuitive, getting a score around 92%. The reason for not getting a higher sore is the data aggregation and calculations, a lot of the intermediary results being shown on the negotiation interface. At second question, about 70% of the negotiators consider that some information, mainly the intermediary results may be removed from the interface. At question three, only 2 out of 5 negotiators, 9 out of 20 resource managers, and 19 out of 25 inquirers declare that they experienced difficulties during the usage of the system. At question four, regarding the level of knowledge required for the system usage, 80% of the users declare that they do not need prior knowledge about the system. The question five is addressing the most common problem, the communication with the server. Only 21 out of 50 persons did not experience communication problems. Regarding the question 6.1, 60% of inquirers are satisfied with the allocation result, considering that the allocated resources meet the requirements. At question 6.2, 85% of the resource managers consider the negotiation results as acceptable. At question 7, the communication problems for the negotiators are more frequent than other entities. This is because the negotiator sends and receives the largest amount of information in the system.

Conclusions and Future Work

The paper presents a model based on MASs for solving the problem of allocating resources in an academic environment. The novelty of the system is that it applies the negotiation process for incremental satisfying of the soft constraints, in a transparent way (the intermediate results are sent to all participants and their preferences might change accordingly). This approach reduces the computational complexity of the resources allocation problem. The model is relevant not exclusively for the academic resources management, but also for a large variety of domains, including the load distribution, production planning, computer scheduling, portfolio selection, apportionment, and so on. The system limitation is that, for the moment, it is a stand-alone system, not being integrated into the ERP system of the university. In the near future, the authors will integrate system in the ERP system, in order to generate the academic timetables and different reports about the resources management.

References

Bodea, C. N., Badea, I. R. & Purnus, A. (2011). "Distributed Research Project Scheduling based on Multi-Agent Methods," in: *Journal of Applied Computer Science & Mathematics,* no. 10 (5) /2011, pag. 20-26, ISSN:1843-1046, Suceava, http://www.jacs.usv.ro/?pag=info.

Boutilier, C. (2002). "Solving Concisely Expressed Combinatorial Auction Problems," In Proceedings of the Eighteenth National Conference on Artificial Intelligence (AAAI-02), Pp. 359–366.

Brett, S. W., Ridder, J. P. & Signori, D. T. (2012). Distributed Algorithms for Resource Allocation Problems, *The 17th International Command and Control Research and Technology Symposium,* 19-21 June, Fairfax VA USA.

Brinkkemper, S. (1996). "Method Engineering: Engineering of Information Systems Development Methods and Tools," *Information & Software Technology,* 38, Pp. 275-280.

Carrasco, M. P. & Pato, M. V. (2004). "A Comparison of Discrete and Continuous Neural Network Approaches to Solve the Class/Teacher Timetabling Problem," *European Journal of Operational Research* 153 (1), 65–79.

De Vries, S. & Vohra, R. V. (2003). "Combinatorial Auctions: A Survey," *INFORMS Journal on Computing,* 15(3), 284–309.

Dolgov, D. A. & Durfee, E. H. (2006). "Resource Allocation among Agents with MDP-Induced Preferences," *Journal of Artificial Intelligence Research* 27, 505–549.

Gislen, L., Peterson, C. & Soderberg, B. (1992). "Complex Scheduling with Potts Neural Networks," *Neural Computation* 4, 805–831.

IEEE Standards Board (1990). 'Standards Coordinating Committee of the Computer Society of the IEEE,' *IEEE Standard Glossary of Software Engineering Terminology, IEEE.*

Looi, C. K. (1992). "Neural Network Methods in Combinatorial Optimization," *Computers and Operations Research* 19 (3/4), 191–208.

Mendoza, G. A. & Prabhu, R. (2003). "Qualitative Multi-Criteria Approaches to Assessing Indicators of Sustainable Forest Resource Management," *Forest Ecol. Manage.* 174, 329–343.

Rudova, H. & Murry, K. (2003). "University Course Timetabling with Soft Constraints," In: Burke E. K. and Causmaecker P. D. (Eds.), Proceedings of the Practice and Theory of Automated Timetabling IV, Fourth International Conference, Gent, Belgium, Lecture Notes in Computer Science (LNCS), *Springer,* Pp. 310–328.

Sandholm, T. & Boutilier, C. (2006). Preference Elicitation in Combinatorial Auctions. In: Cramton, Shoham, & Steinberg (Eds.), Combinatorial Auctions, *MIT Press.*

Silva, J. D. L., Burke, E. K. & Petrovic, S. (2004). "An Introduction to Multiobjective Metaheuristics for Scheduling and Timetabling," *Lecture Notes in Economics and Mathematical Systems, Springer,* Volume 535, Pp. 91–129.

Sturm, A. & Shehory, O. (2003). A Framework for Evaluating Agent-Oriented Methodologies, În Giorgini P. and Winikoff M. (Eds.), Proceedings of the Fifth International Bi-Conference Workshop on Agent-Oriented Information Systems, Melbourne, Australia, Pag. 60-67.