Publishing House
AKAPIT

COMPUTER METHODS IN MATERIALS SCIENCE
Informatyka w Technologii Materiałów

Vol. 11, 2011, No. 2

INDUSTRIAL APPLICATIONS OF MULTI-AGENT SYSTEMS

STANISŁAWA KLUSKA-NAWARECKA[1], GRZEGORZ DOBROWOLSKI[2*], EDWARD NAWARECKI[2]

[1] *Foundry Research Institute, Cracow, Poland, Academy of Information Technology, Łódź, Poland*
[2] *AGH University of Science and Technology, Kraków, Poland*
**Corresponding author: grzela@agh.edu.pl*

Abstract

The paper outlines concepts and solutions specific to the given class of the applications of multi-agent systems. The considerations are illustrated with examples of the results ob-tained in tests and computational experiments carried out us-ing pilot implementations.

Key words: multi-agent systems, metal product defects, supply chains, computational in¬telligence

1. INTRODUCTION

When on the turn of the late 80's and early 90's of the twentieth century the first studies on the agent paradigm appeared referred to intelligent information processing methods and complex support systems for decision-making processes, many skeptics felt that this area does not have any broader development perspective. Despite these gloomy predictions, in the late 90's, an intensive development of multi-agent systems was observed, both in the sphere of general concepts and theoretical foundations, as well as practical applications and related development tools.

Currently, the number of studies published in the widely understood issues of the agent systems is growing constantly, but most of them contain discussions kept at a high level of abstraction, or refer mainly to the problems of tools and implementations. Reports are still lacking that would clarify the concept of the creation of specific solutions and applied methodologies.

The material presented in this paper refers to this last area, showing the concepts, methods and structure of agent systems oriented to the solution of specific industrial problems. Discussing solutions available in these areas of the knowledge management, logistics and computational intelligence, an attempt was made to highlight their diversity, while paying attention that they are all comprised in one common model.

The considerations are illustrated with results obtained during pilot implementations carried out at the Department of Computer Science of the AGH University of Science and Technology, Foundry Research Institute in Krakow and the AGH Department of Applied Computer Science and Modeling.

2. MULTI-AGENT SYSTEM AND FIELDS OF ITS APPLICATION

In the process of creating the theoretical backgrounds and concepts for an application of agent paradigms, numerous definitions and formal models of both agents and respective systems are created (Wooldridge, 2002, Weiss, 1999, Cetnarowicz, 1999, Dobrowolski, 2002). To confer these concepts

ISSN 1641-8581

a possibly general dimension, one can refer to definitions proposed by the classics in this field.

The above definitions are of an intuitive character and as such are not able to create direct backgrounds for the implementation of practical solutions of more utilitarian character. The first step in the creation of Multi-Agent System (MAS) concept of predetermined functionality is adopting the appropriate (if only general) shape of a formal model defining more strictly the character of the designed solution.

To satisfy the needs of the discussion carried out further, the model of an agent and of an agent system has been adopted in the form of aggregates:

$$ag =< Goal, Act, com, env, res, atr > \qquad (1)$$

where: *Goal* – a set of the agent's local goals, *Act* – a set of its actions, *com* – means of communication in the shape of specialized actions, *env* – the agent's relations with the environment (e.g. location), *res* – resources consumed or produced by the agents as the effect of its actions, *atr* – the other agent's attributes not constituting directly the system.

$$\text{MAS} =< Ag, Env, Com, Res, Org > \qquad (2)$$

where: *Ag* – a set of the system agents, *Env* – a common environment, *Com* – support for inter-agent communication, *Res* – resource representation in the environment, *Org* – organization of the system generated by the global goal of the system.

Examining the meaning of the above given formulae, it is easy to observe that even the agent itself as an autonomous entity creates wide possibilities for an interpretation and hence a great variety of possible modes of its realization (software or physical). Still larger number of possibilities of this type offer the concept of an agent system where, besides a set of agents, numerous other factors (relations, restrictions) responsible for operation of these agents are also taken into account.

Considering the above, it is difficult to speak about a reasonable fine-grained classification of the agent systems, and the more about the formulation of general rules for their designing and use. As a consequence, the construction of an effectively operating agent system (in terms of certain predetermined requirements) is still a serious problem of both scientific and engineering nature.

Analyzing the available data on the already existing and operating agent systems, it seems reasonable and justified to adopt as a criterion of their classification the goal they are supposed to serve and not the principles of their construction or algorithms used in their operation.

With this approach adopted, the most typical areas of industrial applications (goals) of agent systems embrace:

– simulation of processes and phenomena, where the goal is: acquiring possibilities to examine properties of the investigated process through observations (measurements) of parameters not available under real conditions;
– management and control of complex processes and systems of a decentralized character which makes application of the centralized decision mode impossible;
– intelligent computational systems, in which the agent paradigm is used in solving of complex computational problems (most often related to the task of optimization);
– identification of complex systems (also multi-agent ones), where information sources are heterogeneous and scattered in both space and time, and whose operation is of decentralized character (not subject to common control).

3. KNOWLEDGE MANAGEMENT IN THE DIAGNOSIS OF DEFECTS IN METAL PRODUCTS

The task of the agent system described in this study is management of the diagnostic knowledge on defects in metal castings (Kluska-Nawarecka et al., 2007*b*, Kluska-Nawarecka et al., 2007*c*, Dobrowolski et al., 2003). The execution of diagnostic functions requires the ability to use the knowledge comprised in autonomous modules, where the mode in which it is represented is of a heterogeneous character (rules, fuzzy rules, tables, photographs). So, in such a system, the knowledge management will consist in rendering relevant fragments (components) of the knowledge and in integration of them especially in the presence of their heterogeneity (Dobrowolski et al., 2003).

In the given solution, the agent tasks are assigned to individual modules of the knowledge, thus forming a system whose logical structure is schematically presented in figure 1. Apart from these modules, the structure of the system includes an agent responsible for the ontological service (integrating knowledge components) and an agent responsible for the knowledge management, i.e. providing user with the knowledge elements necessary to solve the problem.

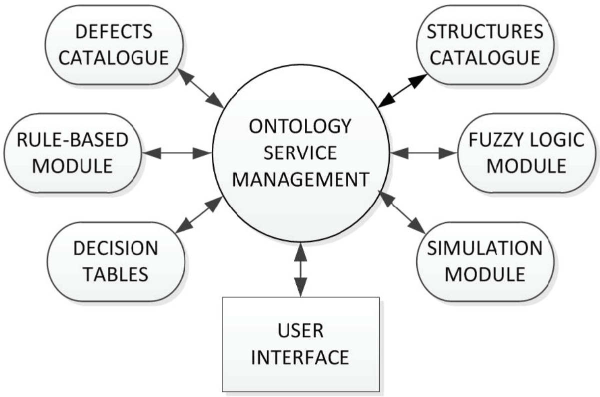

Fig. 1. General structure of the system.

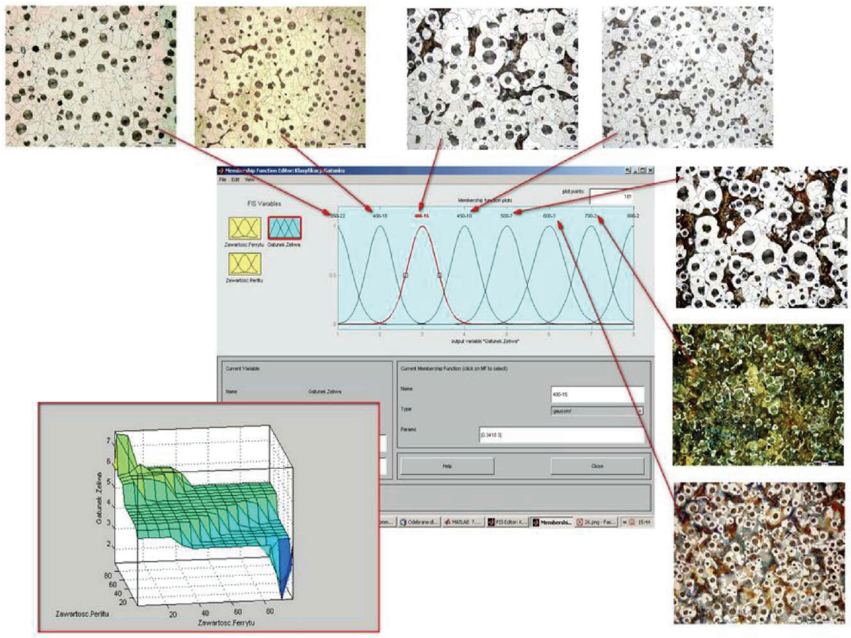

Fig. 2. Visualization of the chosen results for the case of fuzzy analysis of casting defects.

The description of figure 1 in terms of agent systems requires the determination of individual modules, first, and of their entire configuration, next, in accordance with models given in the form of aggregates 1 and 2.

Considering, for example, module called DECISION TABLES, according to 1, the following assignments can be made:

goal = [defect classification]
Act = [receiving message, reasoning, sending message]
com = [graph of connections, protocols] (3)
env = [ontologies, input data]
res = [available knowledge resources]
atr = [defect attributes]

In the same way one can specify the concepts defining agents that represent other knowledge modules. On the other hand, quite different form assumes the specification of module ONTOLOGY SERVICE. Here, the following elements can be specified:

goal = [knowledge integration]
Act = [receiving message in sender's ontology, changing ontology, sending message in receiver's ontology]
com = [graph of connections, protocols in different ontologies] (4)
env = [partial ontologies of individual modules]
res = [common ontology, integrating procedures]
atr = [attributes of partial ontologies]

Knowledge integration requires the application of complex system procedures, which in the described solution are executed by OntoGrator system (Kluska-Nawarecka et al., 2007*a*), operation of which is based on data described in OWL language. To picture the complex character of these procedures, figure 2 gives visualization of the chosen results for the case of fuzzy analysis of casting defects.

After specification of individual agents one can describe the whole system in accordance with formula 2:

Ag = {ag1, ag2, . . . , agn}
Env = [ontology, database field]
Com = [graph of connections, protocols] (5)
Res = [agents knowledge, databases]
Org = [knowledge management procedure, agents cooperation mode]

The procedures of knowledge management co-operating with the user's interface should enable the execution of preset global system goal, which is the best possible satisfaction of the user needs.

4. AGENT-BASED SIMULATION OF THE SUPPLY CHAINS

The great complexity of problems related to the management and optimizing of decisions for systems defined as supply chains (understood as processes of production and distribution of various types of goods) is the reason why problems of this class are often solved with the use of complex simulation models. However, the methods to design such models also present a serious research problem and therefore are all the time at the stage of intensive development.

From numerous studies in this field e.g. (He et al., 2006, Moyaux et al., 2006) it follows that agent systems are fully consistent with the specific characteristics of operation of the supply chains, and hence their application in construction of simulation models of a given class is both well justified and very promising.

An important difference, in respect of the case disclosed in Section 3 concerning management of knowledge resources, consists in the fact that the simulation model operates in a virtual environment of the designed agent system, while the former MAS is, as we might say it, embedded in the real environment representing the knowledge resources which were the subject of the carried out activities.

As a consequence in this case, the first stage of model designing will be identification in a real system (supply chains) production units which have significant effect on the operation of this model. In practice, it consists in determination of a set of agents present in the model done by equation 2.

Basing on the known from literature, descriptions of the supply chains functioning, the following set of agents operating in the designed model is adopted (Koźlak et al., 2007*a*, Koźlak et al., 2007*b*):

Customer (Cust) – represents a final customer, ordering goods produced by the company;

Buyer (Buy) – places orders for goods needed in interaction with the market role;

Seller (Sell) – sells goods on the market;

Market (Mar) – associates buy and sell requests coming from Agents Buyer, Customer and Seller;

Inventory Manager (InM) – informs Buyer which components should be bought also predicts the demand;

Producer (Prd) – performs a production process and, basing on the available actions and resources possessed, performs production of output goods, which are being sold by Seller;

Strategy Planner (SPl) – manages other agents representing parts of the company and the configuration of their goods, makes choice of the production strategy and choice of the price ranges offered.

More precise characteristics of all these agents would require very detailed studies, and this is why the present study is limited to demonstration of the elements of a model of an Agent Strategy Planner, which is the most important one in a production company.

In this case, the components of aggregate 1 are determined in the following way:

The goal of SPl agent is maximization of a quality index, given as a composition (e.g. weighted sum) of two components:

$$goal(\mathrm{SPl}) \equiv f_{\mathrm{SPl}}(\mathrm{financing},\ \mathrm{prestige}) \rightarrow \max \qquad (6)$$

where: financing – are company's global financial assets or its income obtained within an assumed time interval, prestige – is understood as the evaluation of the company in relation to other (competitive) companies, based on e.g. the quality of produced goods, punctual deliveries, etc.

Actions of this agent consist of decisions regarding strategy in respective areas of its activity:

$$\begin{aligned} act(\mathrm{SPl}) \equiv [&\text{set strategies for :}\\ &\text{production, storage, selling}] \end{aligned} \qquad (7)$$

The resources (*res*) of agent SPl are the available (within certain time interval) financial means, the owned means of production and the warehouse stock.

The SPl agent's environment (*env*) is composed of the above mentioned agents from its company and agents described below acting outside the company as well as those who belong to other companies.

The agent's communication (*com*) is defined by a graph representing the company's internal communication network as well as the communication means, available to the agent within the environment in which it operates.

The agent's model can be completed with additional attributes (*atr*), which determine, among others, the applied algorithms, specific preferences, etc.

Attention deserves the fact that Customer and Market agents are operating outside the company (they belong to the environment) and can render services to other companies operating within the simulated environment.

Having defined the role of all agents, a conceptual MAS model can be formulated, determining the individual elements of an aggregate (2).

The designed set of agents (*Ag*) comprise all agents who perform the roles described above, these belonging to the companies covered by simulation, and agents operating in an environment outside the market (Cust, Mar).

The whole environment (*Env*), communication (*Com*) and resources (*Res*) are defined as a sum of the corresponding sets assigned to individual agents or companies.

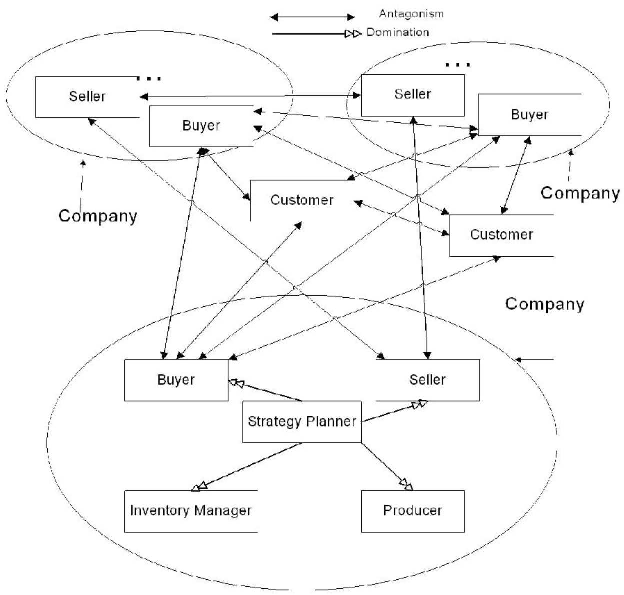

Fig. 3. Relations between agents: antagonism and domination.

However, an important problem affecting the structure of a simulation model is specifying the principles of its organization (*Org*). The components of an MAS organization structure are communication and access to resources. Here, both these components results in a natural way from the real characteristics of the simulated companies, which are taken into consideration as early as at a level of the agent models ($ag \in Ag$). The third component of the MAS organization is a set of relations, which occur between agents. Each of the relations is determined on Cartesian product $rel \subseteq Ag \times Ag$ which means that to each pair of agents a formula determining their mutual relations is ascribed.

The determination of these relations is an important project task, since at the level of implementation it determines, among others, the choice of decision algorithm, restrictions, sometimes even strategy.

In the solution described in the present study, the following set of relations (*Rel*), which seemed to be most characteristic of the examined class of companies, was determined:

independence – no relations occur for a given pair of agents;

cooperation – agents are mutually dependent, because they decide to jointly pursue their goals;

domination the decisions of one of the agents (the senior) are taken arbitrarily and have an impact on the operation of the second one (the subordinate);

antagonism (conflict) – agents are interdependent, and at the same time their goals oppose each other (an antagonistic game).

For a particular environment, a graphical representation can be designed in the form of a network of links between agents corresponding to a given mutual relationship. The network illustrating the relations of antagonism and domination are shown in figure 3.

Other relationships between agents (e.g. all company's agents remain in the relationship of Subordination in terms of Agent Strategy) can be represented in a similar manner.

The above described components and aspects of the operation of a real system represented by agents and links that occur between them constitute a basis in order to design a simulation model of the MAS, which after testing allows examination of different scenarios of the operation of the analyzed supply chain.

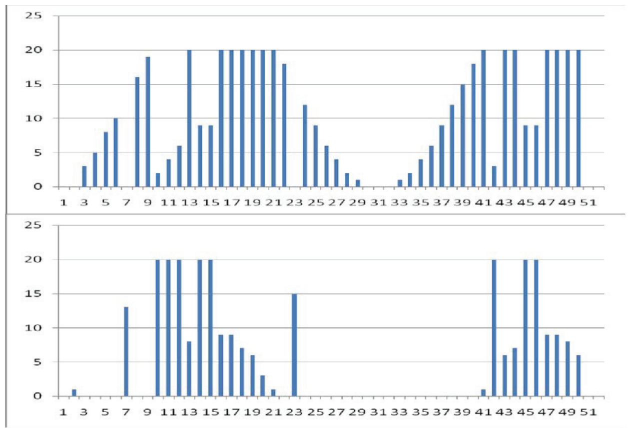

Fig. 4. Sale of two companies applied different production strategies: aggressive and conservative.

It is obvious that the considerations presented here sketch the idea of designing a model of MAS only, while appropriate concepts and activities take shape of formalized procedures and algorithms during implementation.

As an illustration, figure 4 shows the results of one of the performed tests. The selected scenario describes the effects of the operation of two companies using different production strategies (conservative and aggressive) in a situation when the demand for the product being manufactured has been changing over time in a sinusoidal way.

5. EVOLUTIONARY AGENT AND COMPUTATIONAL INTELLIGENCE SYSTEMS

Notion *computational intelligence* describes methods and system solutions that serve solving problems (most often optimization ones) that due to their complexity cannot be successfully treated with other classical methods (of analytic or heuristic types).

In the case of agent system dedicated to solving problems of computational intelligence, the problem concerns the creation of a virtual environment in which the population of agents is subjected to evolutionary changes aimed at finding a solution to the computational task (Kisiel-Dorohinicki, 2002, Byrski et al., 2002, Byrski & Kisiel-Dorohinicki, 2007). The notion of computational intelligence is related with the methods and system solutions used to solve problems (mostly optimization), which due to their complexity cannot be solved by classical methods (analytical methods, the typical heuristic algorithms).

The basic analogy between classical Genetic Algorithm (GA) and Evolutionary Multi-Agent System (EMAS) is assigning the role of solutions carriers to the agents similarly to the chromosomes of GA. However, the mechanism organizing the process of evolution is radically different in EMAS, where selection based on global ratings is replaced by a fully decentralized self-assessment process carried out by individual agents. General concept of EMAS activities is sketched out in figure 5.

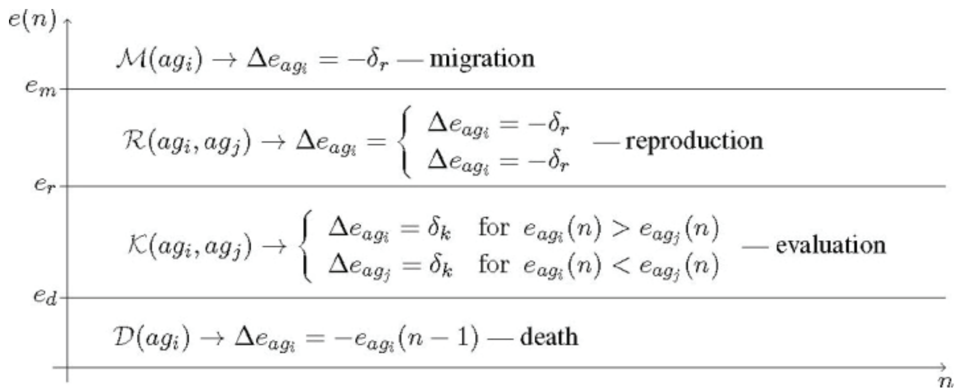

Fig. 6. *Levels of the agent's vital energy and its energy-determined actions.*

The basis for such an assessment is provided by the introduced concept of the agent „vital energy". This original concept presented in (Dobrowolski et al., 2001) has found various developments in the later work of the team, among others in (Dobrowolski et al., 2002, Kisiel-Dorohinicki, 2004, Byrski & Kisiel-Dorohinicki, 2009, Dreżenzewski & Siwik, 2008), which abled creation of different variants of EMAS oriented to solving several classes of computational tasks (multimodal function optimization, multi-objective optimization, probabilistic problems).

The vital energy is available to agents depending on the performance of their operation. In general, this relationship can be written as:

$$e_{ag_i}(n) = e_{ag_i}(0) + \sum_{k=1}^{n-1} \Delta e_{ag_i}(u_{ag_i}(k)) \tag{8}$$

where: $e_{ag_i}(n)$ the energy of agent agi at the nth stage of the evolution process; $\Delta e_{ag_i}(u_{ag_i}(k))$ the change in the energy of agent agi resulting from the performed operation $u_{ag_i}(k)$.

The way of defining the action $u_{ag_i}(k)$ and determination of the corresponding energy changes $\Delta e_{ag_i}(u_{ag_i}(k))$ differ somewhat for the individual EMAS versions though, generally, they are based on distinguishing the characteristic energy levels of an agent. The idea is presented schematically in figure 6.

According to this scheme, the drop of agent energy below the level of death ed makes him deleted from the system. In the interval $e_d \leq e_{ag_i}$ (n) $\leq e_r$ (the competition area) meeting of agents can take place, where agent of a larger energy absorbs the energy δ_e of the „weaker" agent. Agents whose energy exceeds the level of reproduction er can generate a new offspring, and each of the agents will provide this offspring with a part of its own energy δ_r.

In some variations of EMAS (island systems), an energy level of migration e_m is additionally distinguished, upon exceeding of which agents can take action to move to another area of the environment. At the same time, „elite" areas are formed, where only agents with high energy, it means representing good approximation of solutions to a given computational problem, can reside.

In addition to energy constraints referred above, the specification of EMAS takes into account a number of detailed procedures determining how to generate an initial population, the principle of agents meeting (competition, recombination) and their migration (island systems). Different variants of these procedures are constructed within the specific implementation of

EMAS. Referring to the general model of the agent expressed by formula 1, the following interpretation of its components is adopted:

goal – maintaining the high level of energy (allowing to say alive), at the same time striving to maintain the stability of the population (recombination) and possible creation of favorable conditions for further action (migration);

Act – determined by actions as described above, corresponding to the current energy level of the agent;

env – virtual space in which agents operate; it has the capacity to supply energy (creation of the agent) or to consume energy (destruction of the agent), and can also create some opportunities for communication;

com – determined by the adopted organization of EMAS; it may be comparison of the energy during meetings (competition, reproduction) and information about the average energy level in a specific area of the environment (island variant – migration);

res – the energy state of the agent $e_{ag_i}(n)$ (as described above), determining its ability to function;

atr – basic attributes of an agent are represented by the solution of a computational task it can offer (encoded in its genotype).

These terms are appropriately transferred to the level of the whole system (EMAS), described by a general formula 2, while within the organization scheme (*Org*) its structure (possibly with islands) is determined, as well as the values of individual parameters (energy levels, quanta of energy transferred, and parameters of procedures not specified here).

The above sketched rules for construction of EMAS result in adding to the process of evolution some entirely new properties (with respect to the typical GA), among which as the main ones the following should be pointed:

– conferring decentralized nature on evolution, where individuals (agents) decide themselves about the course of their transformations;
– eliminating arbitrarily implemented selection procedures, which could cause the loss of diversity in a new population;
– introduction of new evolutionary operators (recombination, migration), which leads to an enrichment of opportunities to seek for new solutions;
– increasing possibilities to influence the process of evolution through appropriate selection of parameters, such as power thresholds, the size of the energy quanta, etc.

The global consequence of these modifications is increasing the stability of the population while maintaining the diversity of its individuals, which in most cases leads to improved efficiency of the computational process (convergence, accuracy of solutions).

An interesting aspect of the cognitive nature may be an analogy between the operation of EMAS and the processes of evolution in natural environments (social, economic). In both cases, these processes may depend not only on the general conditions and development trends, but also on the behavior of different individuals (e.g. in the emergency and crisis situations).

At the same time, the fact worth noting is that in the considered embodiment of EMAS, the contact of the virtual environment with the reality involves only the calculation (measurement) of utility function, allowing for evaluation of various solutions.

The computational experiments conducted for several of the EMAS realization oriented at different classes of the optimization problems showed the effectiveness of the approach based on the energy model of an agent. As one of the successful implementations of systems of this class is EMAS system running on a NEvol platform (Net Evolutionary Computing Platform) realized in the AGH Department of Computer Science.

6. FINAL REMARKS

Against the background of the general characteristics of agent systems, the results of the conducted studies concerning MAS application in the areas of knowledge management in foundry technology were presented together with modeling of supply chains and computational intelligence.

In the carried out discussion it was sought to highlight the concepts and ideas characteristic of an embodiment of these applications, while indicating that they form methodology for constructing certain class of computer systems.

As key elements of the presented solutions that decide of their originality, one ought to underline:

– with respect to knowledge management – solving problems with representation and integration of technological knowledge;

– in relation to supply chains simulation – definitions of the agents' roles and their interrelations;
– within the scope of computational intelligence – the concept of vital energy that enables autonomous evolution of the agents.

The authors believe that the material presented here is not only a relation of the conducted research but can also provide some guidance and inspiration when undertaking the work related with application of an agent approach.

ACKNOWLEDGEMENTS

The presented work has been partly financed by the Polish Ministry of Science and Higher Education, grant no. N N516 366236.

REFERENCES

Byrski, A., Kisiel-Dorohinicki, M., 2007, Agent-based evolutionary and immunological optimization, Computational Science -ICCS 2007, *7th International Conference, Beijing*, China, May 27 30, 2007, Proceedings, Springer.

Byrski, A., Kisiel-Dorohinicki, M., 2009, Agent-based model and computing environment facilitating the development of distributed computational intelligence systems, Proc. of the Computational Science -ICCS 2009, *9th International Conference*, Baton Rouge, U.S.A., May 25-27, 2009, Springer.

Byrski, A., Kisiel-Dorohinicki, M., Nawarecki, E., 2002, Agent-based evolution of neural network architecture, M. Hamza, ed, *Proc. of the IASTED Int. Symp.: Applied Informatics*, IASTED/ACTA Press.

Cetnarowicz, K., 1999, *Problemy projektowania i realizacji systemów wieloagentowych*, Vol. 80 of *Rozprawy Monografie*, UWND AGH, Kraków.

Dobrowolski, G., 2002, *Technologie agentowe w zdecentralizowanych systemach informacyjno-decyzyjnych*, Vol. 107 of *Rozprawy Monografie*, Uczelniane Wydawnictwa Naukowo-Dydaktyczne Akademii Górniczo-Hutniczej im. S. Staszica, Kraków.

Dobrowolski, G., Kisiel-Dorohinicki, M., Nawarecki, E., 2001, Evolutionary multi-agent system in multiobjective optimisation, M. Hamza, ed, *Proc. of the IASTED Int. Symp.: Applied Informatics*, IASTED/ACTA Press.

Dobrowolski, G., Kisiel-Dorohinicki, M., Nawarecki, E. , 2002, Evolutionary multi-agent system for multiobjective balancing of production lines, E. Camacho, L. Basanez, A. de la Puente, ed, Preprints of the 15th World Congress of IFAC. Final Program. Book of Abstracts + CD, PERGAMON, Barcelona, Spain, p. 440.

Dobrowolski, G., Marcjan, R., Nawarecki, E., Kluska-Nawarecka, S., Dziaduś, J., 2003, Development of INFOCAST: Information system for foundry industry, *TASK Quarterly* 7(2), 283-289.

Dreżewski, R., Siwik, L., 2008, Agent-based co-operative co-evolutionary algorithm for multi-objective optimization, L. Rutkowski, ed., Artificial Intelligence and Soft Computing — ICAISC 2008, Vol. 5097 of *Lect. Notes in Comp. Sci.*, Springer, p. 388-397.

He, M., Rogers, A., Luo, X., Jennings, N., 2006, Designing a successful trading agent for supply chain management, Proc. of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems, *ACM*, 1159-1166.

Kisiel-Dorohinicki, M., 2002, Agent-oriented model of simulated evolution, W. I. Grosky, F. Plasil, ed, SofSem 2002: *Theory and Practice of Informatics*, Vol. 2540 of *LNCS*, Springer-Verlag.

Kisiel-Dorohinicki, M., 2004, Agent-based models and platforms for parallel evolutionary algorithms, Proc. of the Computational Science - ICCS 2004, *4th International Conference*, Kraków, Poland, June 6-9, 2004, Springer, LNCS 3038.

Kluska-Nawarecka, S., Dobrowolski, G., Nawarecki, E., Marcjan, R., 2007*a*, OntoGRator – an intelligent access to heterogeneous knowledge sources about casting technology, *Computer Methods in Materials Science* 7(2), 324-328.

Kluska-Nawarecka, S., Marcjan, R., Adrian, A., 2007*b*, The role of knowledge engineering in modernisation of new metal processing technologies, *Archives of Foundry Engineering* 7(2), 169-172.

Kluska-Nawarecka, S., Smolarek-Grzyb, A., Wilk-Kołodziejczyk, D., Adrian, A., 2007*c*, Knowledge representation of casting metal defects by means of ontology, *Archives of Foundry Engineering* 7(3), 75-78.

Koźlak, J., Dobrowolski, G., Nawarecki, E., 2007*a*, Agent-based modeling of supply chains in critical situations, Y. Shi, G. D. van Albada, J. Dongarra, P. Sloot, ed, Computational Science – ICCS 2007, Vol. 4488 of *Lect. Notes in Comp. Sci.*, Springer, 944-951.

Koźlak, J., Marszalek, J., Siwik, L., Zygmunt, M., 2007*b*, Selection of efficient production management strategies using the multi-agent approach, H.-D. Burkhard, G. Lindemann, R. Verbrugge, L. Varga, ed, Multi-Agent Systems and Applications V, Vol. 4696 of *Lect. Notes in Comp. Sci.*, Springer, 337-339.

Moyaux, T., Chaib-draa, B., D'Amours, S., 2006, Supply chain management and multiagent systems: An overview, B. Chaibdraa, J. Muller, ed, *Multiagent Based Supply Chain Management*, Vol. 28, Springer, 4–45.

Weiss, G., ed, 1999, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, The MIT Press.

Wooldridge, M., ed, 2002, *An Introduction to Multiagent Systems*, John Wiley & Sons.

PRZEMYSŁOWE ZASTOSOWANIA SYSTEMÓW AGENTOWYCH

Streszczenie

Artykuł przedstawia w zarysie koncepcje i realizacje specyficzne dla danej klasy zastosowań systemów agentowych. Rozważania ilustrowane są przykładami wyników uzyskanych podczas testów i eksperymentów obliczeniowych wykonanych z użyciem odpowiednich implementacji pilotowych.

Received: November 11, 2010
Received in a revised form: December 6, 2010
Accepted: December 10, 2010