INFORMATYKA EKONOMICZNA BUSINESS INFORMATICS 3(25) · 2012
ISSN 1507-3858

Jerzy Korczak, Maciej Bac, Krzysztof Drelczuk, Aleksander Fafuła
Wrocław University of Economics, Poland
e-mail: {jerzy.korczak, maciej.bac, krzysztof.drelczuk, aleksander.fafula}@ue.wroc.pl

MULTI-AGENT SYSTEM FOR STOCK TRADERS[1]

Abstract: The authors of this paper present the architecture of a multi-agent system which supports investment decisions on the stock market. The individual components of the system, the manner of communication between them and the mechanism of assessing the individual agents are discussed. New methods of pre-processing financial series, the concept of never-ending learning, the behavioral model of stock exchange traders and the manners of translating the modeled patterns into the open and close position signals are described. The results of the research are described and the directions of the further development of the platform are provided in the conclusion.

Keywords: multi-agent decision support systems, financial markets, stock trading expertise, artificial intelligence, time series analysis.

1. Introduction

Generally, the algorithms used in stock trading decision support systems can be based on mathematics, statistics, economics and artificial intelligence [Allen, Karjalainen 1999; LeBaron 2011; Dempster, Jones 2001; Lipiński 2004]. In this paper a multi-agent system to advise traders on foreign exchange markets will be presented. One of the first multi-agent systems was proposed by K.P. Sycara, D. Zeng and K. Decher [2002]. The system enabled the user to cooperate with multiple specialized agents which had access to financial models and supervise the operation of the system on the market. Recently, various interesting concepts of multi-agent systems have been published. C. Chiarella, R. Dieci and L. Gardini [2011] describe a system where two groups of agents, applying the methods of the fundamental and technical analysis, shape market dynamics. Similar research was conducted by F. Westerhoff [2004]. V. Bohm and J. Wenzelburger [2005] present the evaluation of the shares portfolio optimization strategy by three agents: rational agent, interference agent and technical

[1] Selected parts of this article were published under non-exclusive copyright in *Proceedings of the Federated Conference on Computer Science and Information Systems FedCSIS 2012* (see [Korczak et al. 2012]).

analysis agent. Among numerous other studies on the applications of multi-agent systems, the overview of active and passive forms of learning of the agents operating on financial markets by B. LeBaron [2011] is also worth mentioning.

There are several standards describing the principles of designing systems based on agent technologies. One of them is the standard developed by the FIPA (Foundation for Intelligent Physical Agents) organization. It describes the manner of management, communication and cooperation between agents. Another standard is the MAF (Mobile Agent Facility) defined by the Object Management Group – Agent Working Group. The Agents Society and the Knowledge Sharing Effort were also involved in the standardization of agent technologies. The principles established by these associations have been respected in our project and agent implementation.

The aim of this paper is to present the concept of the scalable and open multi-agent system which would enable the best possible support for investment decisions due to the integration and cooperation of the agents. In the system, called A-Trader, the accuracy of predictions, the orientation on online trading, the improvement of own knowledge base and the ability to adapt to the changing environment were important requirements. Many ideas have been taken from our previous works, notably on the internet Bourse Expert – Agent Technology system (iBE-AT), developed in the LSIIT-CNRS [Korczak, Lipiński 2008; Lipiński 2004]. In the iBE-AT system, several rule-based agents were generated by the genetic algorithm to recommend purchase and sale decisions. In the new systems, the agents may be composed of various functions, not necessarily technical analysis ones, and the supervisor strategy is extended.

The paper is divided into three main sections. The first one presents the functional architecture of the system. The individual components of the system and the manner of communication between them are discussed. In the second one, the issue of data pre-processing is characterized and the operation of the agent responsible for preparing the information is described. The new idea of learning agents and an agent modeling the behavior of stock exchange traders to discover patterns of the open and close position signals are also described. The last part is a description of the results of the research carried out and an indication of its further development direction.

2. System architecture

The key ideas in designing A-Trader were its openness for integration of new agents and functionalities and the efficient communication between the various agents. The service oriented architecture and cloud computing have solved the problem of computing efficiency. Open and easy to implement, the communication protocol (SOAP) has significantly facilitated the integration of individual agent solutions (see further in [WWW 2–4]). Ultimately, the implemented communication protocol has insured fast data transfer. Popular in distributed systems, PUSH technology has been used to propagate information within the system (see further in [Agarwal 2011]).

In the A-Trader architecture, the following agents and components can be distinguished:
- Notify Agent (NA),
- Historical Data Agent (HDA),
- Cloud of Computing Agents (CCA),
- Market Communication Agent (MCA),
- User Communication Agent (UCA),
- Supervisor (S),
- Database System (DS).

The Notify Agent (NA) ensures efficient communication within the system and is an intermediary in sending signals between agents (see Figure 1). The Notify Agent forwards the information on the status change of a given agent to all the agents that are recorded in the notify register as the clients/observers of its signals. The notification is performed by triggering an appropriate Web method (SOAP) in all the agents from the list of listeners to the indicated signal. Next, it records the information on the status change of the Notify Agent in the database. This capability of the Notify Agent makes the system flexible and scalable, provides the possibility to add and remove agents easily and ensures the independence of the system from the physical position of the agent.

Another system agent downloads data from the Database System (SD) and delivers them to the agents according to their needs. This is the Historical Data Agent (HDA). This agent is responsible for supplying agents with the historical data in the learning processes.

The next system component is the Cloud of Computing Agents (CCA), consisting of Basic Agents Cloud (BAC), Intelligent Agents Cloud (IAC) and User Agents Cloud (UAC). The Basic Agents Cloud (BAC) is a group of agents which pre-process data and compute the basic technical analysis indicators. These agents most often use unprocessed data and perform predefined simple operations. This enables fast computing and the use of the results by agents with more advanced logic. The agents which possess their own knowledge base, which can learn and change their parameters and their internal state, create another component of the agents cloud, called the Intelligent Agents Cloud (IAC). This group of agents includes all the solutions based on artificial intelligence (genetic algorithms, neural networks, expert systems, etc.), agents analyzing text messages and agents observing user behavior. The possibility to monitor and learn the results supplied by other agents makes the agents capable of improving their operations and effectiveness. In the next sections, the important IAC agents will be detailed, notably agents for the pre-processing of financial series, agents responsible for modeling the behavior of stock exchange gamblers and agents based on never-ending learning.

The User Agents Cloud (UAC), in turn, is the agents cloud containing the agents created by external users. Separating this part of the system provides the possibility to integrate the User Agents with the system without the necessity to install the agent

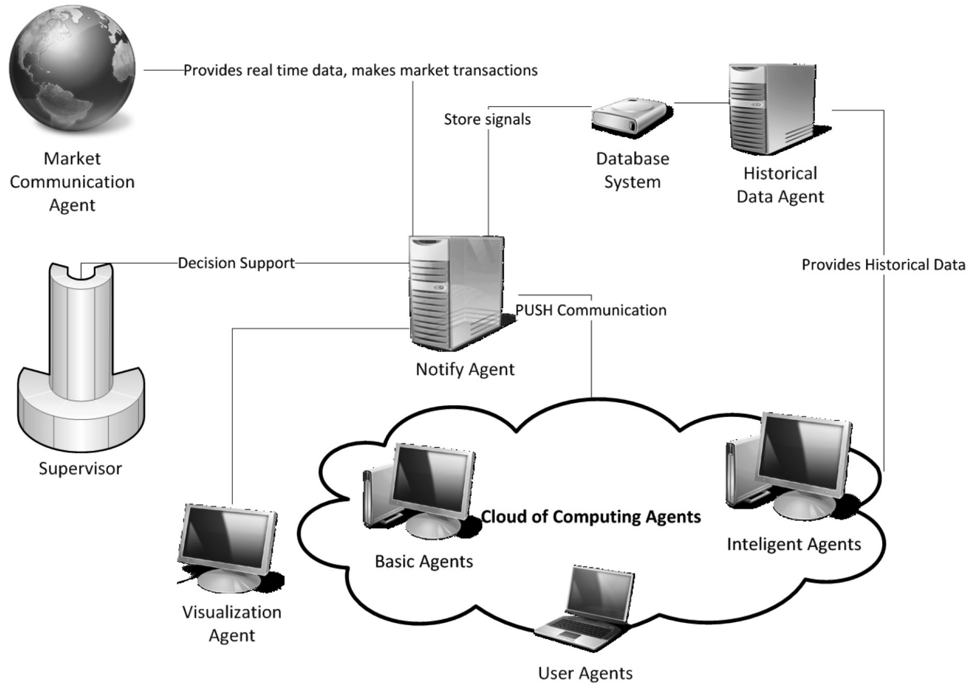

Figure 1. A-Trader system architecture

Source: own elaboration.

on the servers. Such an approach ensures that the user does not have to reveal the secret of its agent's operation while being able to receive and send signals at the same time – and the external agent does not perturb the stability of the system operation. Owing to that, external users may freely add their agents to the system, and the added agents may use all the information/signals supplied by the system. The Intelligent Agents may receive and interpret the signals supplied by the UAC. After obtaining the appropriate quality, some of the User Agents may be moved to the Intelligent Agents Cloud.

The communication of the system with the external environment is ensured by the Market Communication Agents (MCA). On the one hand, these agents supply the news from financial markets and quotations of the available securities; on the other hand, they are responsible for performing open and close position orders. This group includes the agents analyzing the information from companies and browsing the content of the portals devoted to financial markets. They can, among other things, supply information on investor/speculator sentiments by analyzing financial experts' blogs or browsing relevant posts on community portals.

Fast and easy visualization of the results of the agent or the entire system's work is an important aspect in verifying the correctness of its operation. This is possible in the system due to the User Communication Agents (UCA). Visualization tools permit the fast analysis of the agent's operation, which may significantly contribute to simplifying the works on improving its effectiveness. It is also a manner of presenting the results of the platform to end-users.

The Communication Agent allows the user to communicate its own suggestions to the Intelligent Agents. It enables the change of the parameters of a selected agent or the suggestion for the Supervisor on which mechanisms are supposed to influence investment decisions and to what extent. UCA permits the user to prepare and analyze learning sets for the Intelligent Agents. The presentation of the signals supplied by the agents of the system does not require the implementation of their own tools but it is possible due to the integration of the agents with the software supplied by brokers. The operation of A-Trader is schematically illustrated in Figure 2.

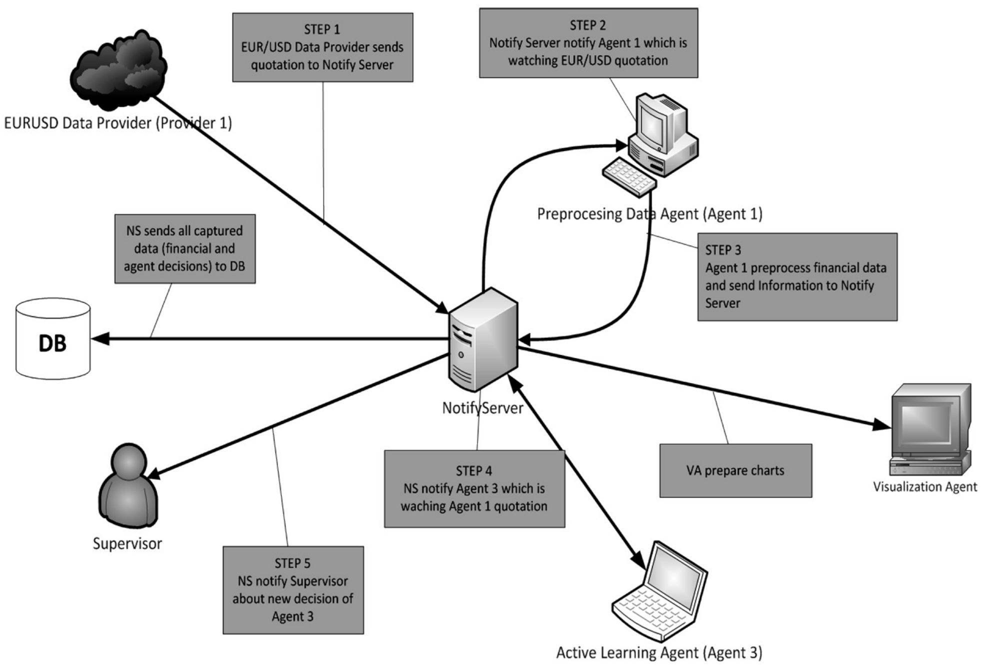

Figure 2. Schema of A-Trader operation

Source: own elaboration.

The multi-agent system is characterized by the agents being capable of generating independent decisions. These may be mutually consistent or completely contradictory. Such mutually exclusive decisions are e.g. the open and close position suggestions generated by two independent agents at the same time. The conflicts are resolved by Supervisors, which observe the decisions of all the Cloud of Computing

Agents and the Intelligent Agents and assess their effectiveness in investing and risk. They decide which agents are taken into consideration when making an investment decision and whose advice is ignored based on the collected information. Decision trees or negotiation methods may be applied for resolving conflicts [Korczak et al. 2012]. Implementing the Supervisor, based on the consensus method also proved to be effective [Hernes 2011]. The decision taken by means of the consensus method accounts for the signals of all agents, which reduces the number of wrong decisions. The Supervisor, apart from making the final decision, regulates the volume of the positions and their maintenance time. It is accountable for the investment security of the entire system and discontinues the investments in cases where exceptional situations, or those not predicted by the system, occur on the market.

The proposed architecture is compliant with the FIPA standard. All the required components and services are ensured by the Notify Agent. The Notify Agent is the Agent Management System. It stores the information on the available agents, their location (White Pages) and available services (Yellow Pages) and fulfils the role of a directory service (Directory Facilitator). It also intermediates in communicating messages between the individual agents (Message Transport). The specificity of the problem being solved (generating open and close position signals), due to the dynamically changing environment, requires an extremely high system efficiency. The advanced communication between the agents occurs solely in the process of learning by the whole system or its individual agents. When starting its work in the online mode, the agent learns or uses its knowledge base and supplements the required historical data. When it is prepared for operation, its work comes down to the analysis of the data sent from the Notify Agents by means of the knowledge collected earlier and the communication of its decision to its Notify Agent. When the decision needs to be made immediately, the process of an advanced inter-agent negotiation, such as establishing a connection or approving the messaging protocol, does not occur.

In order to make the implementation easier, each agent is activated within its container, which isolates it from the environment and which encapsulates the communication with the central part of the system – the Notify Agent. Multiple containers may be activated on one machine. The priorities when designing the container were as follows:

- the maximum simplification of the agent creation process and its integration with the system, which is virtually transparent for the user,
- the maximum simplification of the already created agent and including it in the cloud.

The objective of the isolated agents and their transfer to the cloud is to ensure asynchronous cooperation and enable the performance of specialized operations on the environments dedicated therefore, e.g. computing algorithms, which may be performed synchronously on computers equipped with multi-processor NVIDIA graphic cards. Visualization agents, in turn, may also be activated on the mobile equipment which facilitates user access to the results of the cloud computing.

3. Functional description of selected agents

3.1. Data pre-processing agents

Pre-processing data according to the methodology CRISP-DM (Cross Industry Standard Process for Data Mining) [Chapman et al. 2000] developed by Daimler-Chrysler Analyst, SPSS, and NCR, is known to be the second and third stage of the process of data mining, that is, the "understanding of the data" and "data preparation".

Most of the data stored in databases is rough, incomplete and noisy. For example, the database may include: values that are obsolete or unnecessary, missing values, outliers, data in the wrong format, etc. The overall objective of the process is to minimize irrelevant data that enters the model and often causes erroneous results.

A difficulty with the initial analysis and data processing is that the algorithms work on real data while preparing the data for such models, which requires a very short reaction time. Designers of data mining models are very often balancing between these two guidelines, settling for compromises on data quality to make them available in an acceptable time.

Agents responsible for initial processing, in the presented system, are part of cloud computing agents. An important criterion posed against these agents is the speed of processing. The advantage of extracting such a group of agents is that it makes possible the effective management of the process of preliminary data analysis. This will allow using the automatic optimization of this process and, perhaps most importantly, enable the parallelization of processing allowing the preparation of high quality data within an acceptable time frame.

3.2. Never-ending learning agents

The agents responsible for never-ending learning are a component of the already described agents cloud, and specifically of the Intelligent Agents group. The never-ending learning concept is in this case implemented based on four groups of agents: grouping agents, trainer agents, assessment agents and a group of agents which are supposed to be the subject of the algorithm's operation.

The suggested never-ending learning method is based on the following assumptions:

a) The method does not restrict the user in respect to the selection of financial instruments. The actual data from the currency market are used in the platform design, yet the algorithm will operate efficiently on all other high frequency time series.

b) The number of agents in unlimited.

c) The agent algorithm may be based on any algorithm which can be adjusted to the changing conditions of the financial market by changing the parameters. They include neural networks [Bac 2010], evolutionary algorithms [Barbazon, O'Neill

2005], association rules and expert systems [Dymova, Sevastianov, Bartosiewicz 2002]. Parameterized algorithms are also e.g. technical analysis functions (e.g. RSI, MACD, moving average with the time window size parameter) or other algorithms which do not change their state of knowledge but which are modifiable by input parameters.

The group of never-ending learning agents consists of the following: grouping agent, trainer agent, assessment agents and self-learning agent group. The grouping (association) agent distinguishes the agents that cooperate with each other effectively. Some of the agents will have a higher effectiveness of the decisions taken when other agents are active (a specific value of the output signal). This is how the grouping agent will be capable of distinguishing the agents which operate effectively in conditions of high or low changeability or of the sold-out or bought-out market. Moreover, the grouping agent divides the periods on the observed time series into clusters, describing them by means of specific values of output signals of the individual agents. Next, the trainer agent creates new instances of significantly more effective agents in the specified conditions based on the clusters. Such an operation results in creating extremely specialized agents whose investment performance for particular situations/phases faced by the market is above average. The assessment agents verify the performance of the agents and identify the cause of low performance (excessive number of transactions, premature or too long detention of open/close trading decisions, incorrect TP – Take Profit or SL – Stop Loss). This significantly accelerates the specialization of the agents with auto-adaptive capacities. The formation of new agents detecting completely new market phases will allow the grouping agent to create new clusters and enable the evolution of the system in accordance with the never-ending learning concept.

4. Experimental behavioral agent

One of the top modern elements of the A-Trader system is a module suitable for the on-line tracking of the behavior of investors. The approach used in this subsystem relies on decision rules that have previously been discovered as negative or unwanted. In particular, they could signal the possibility of an investor being under the influence of one of the following cognitive biases: Gambler's Fallacy or Hot Hand.

The rules of this system were extracted from the data collected during multiple experiments. A special stock market simulator was built for this purpose [Fafuła 2010]. This is an experimental and pioneer approach, but the agent is already capable of identifying four crucial situations:

– Gambler's Fallacy (when buying and selling): a belief in trend reversal [Sycara, Decker, Zeng 2002],
– Hot Hand (when buying and selling): a belief in trend continuity [Luna, Perrone (Eds.) 2002].

During the development of these rules, approximately one hundred people were tested for more than thirty days. They were observed during trading and asked many questions related to trading psychology. Finally, the data was processed using various data mining algorithms to reveal the top influencing factors causing erroneous assumptions by investors.

In the end, the behavioral agent can be used for various purposes. One of them is to warn investors about potential mistakes that they could avoid. On the other hand,

Figure 3. Examples of behavioral indicators' signals

Source: own elaboration.

it is possible to point out to them the moments at which it is most likely that other users of the market would make a mistake. Therefore, it is possible to use this agent as a system for suggesting early market entry points.

To describe the matrix of the events, the agent focuses on controlling the price and volume using many techniques derived from the technical analysis. The rules are initially visualized and discussed to confirm their general correctness. Rules can have the format of "if-then" clauses, as in the example below:

IF (bollinger_bands = in [bbands middle to lower] AND williams%range = w%r normal AND
macdhistdir = macd hist rising AND atr = elevated AND Volume > average (Volume, k = 10))
THEN R_possible-fallacies = Gambler's Fallacy

Some of the long positions suggested by these cognitive indicators are displayed in the figures (see Figure 3) including stocks: PFIZER, INTEL and MSFT. They are presented using different time ranges but it is evident that they occur in visually similar formations.

Currently the extracted rules are encoded and tested on real data on NYSE and NASDAQ, revealing results above average. The present goal of this system is to improve performance and determining if the chosen company is more subjective to swing, speculative investing or mainly driven by other forces.

5. Conclusion

The first trials of the implemented multi-agent environment proved extremely encouraging. The Supervisor decreased the investment risk by restricting the independent operations of more risk-taking agents for joint decisions of the entire environment. The cooperating agents made profitable decisions more frequently and closed the loss-generating positions considerably earlier.

The SOAP accepted for communication permitted the fast implementation of the agent containers in the C# environment. The authors are planning to implement the containers in the Java, PHP languages and the environment for the solutions developed in the Matlab software in their further works. The application of the containers will enable the replacement of the SOAP communication protocol for a binary standard ensuring faster communication.

The platform allows the integration of new intelligent solutions and methods operating within the area of the financial series analysis. It permits the use of the signals generated by the already implemented agents, owing to which it is possible to test the effectiveness of own methods in combination with the algorithms implemented so far. By constructing our own agents, it is possible to make use of the information of the assessing agents in order to improve our own solutions. The described multi-agent system makes testing and validating new algorithms easier by supplying the basic functionalities and data. It enables the concentration of works on constructing new agents while not being concerned about the basic data and message supply mechanisms.

Creating agents simulating the behavior of market trade participants is an innovative idea for taking advantage of the infrastructure. Developing a cross-section by investor "types", accounting for the proportion between the number of investors of each type and accounting for capital resources of the individual investor categories will enable the capital flow forecast. It is possible to make an attempt to apply this idea to the changes in the capital allocation on the observed securities and the changes between the securities on the selected market and alternative investment possibilities (securities, bonds, goods, currencies).

The A-Trader platform is now in the testing and expansion phase. The number and scope of the applied methods is being continuously expanded. New agents based on the most recent artificial intelligence methods are being created. On the basis of the concluded research, the solutions operating on various markets and using the available algorithms are being developed. This is an open platform for testing the trading models used in the academic works of the research team and students' theses.

References

Agarwal S., Toward a push scalable global Internet, [in:] *Proceedings of Global Internet Symposium*, IEEE Infocom, Shanghai, China, 2011.

Allen F., Karjalainen R., Using genetic algorithms to find technical trading rules, *Journal of Financial Economics* 1999, Vol. 51, pp. 245–271.

Bac M., Self Organizing Map (SOM) network application support for short-term investment decisions, [in:] J. Korczak (Ed.), *Data Mining and Business Intelligence*, Prace Naukowe Uniwersytetu Ekonomicznego we Wrocławiu nr 16, Wrocław 2010.

Barbazon A., O'Neill M., *Biologically Inspired Algorithms for Financial Modelling*, Springer, 2005.

LeBaron B., Active and passive learning in agent-based financial markets, *Eastern Economic Journal* 2011, Vol. 37, pp. 35–43.

Bohm V., Wenzelburger J., On the performance of efficient portfolios, *Journal of Economic Dynamics and Control* 2005, Vol. 29, Issue 4, pp. 721–740.

Chapman P., Clinton J., Kerber R., Khabaza T., Reinart T., Shaerer C., Wirth R., *CRISP-DM Step-by--Step. Data Mining Guide*, 2000, http://www.crisp-dm.org.

Chiarella C., Dieci R., Gardini L., Asset price and wealth dynamics in a financial market with heterogeneous agents, *Journal of Economic Dynamics and Control* 2002, Vol. 49, pp. 173–197.

Dempster M., Jones C., A real time adaptive trading system using genetic programming, *Quantative Finance* 2001, Vol. 1, pp. 397–413.

Dymova L., Sevastianov P., Bartosiewicz P., A new approach to the rule-base evidential reasoning: Stock trading expert system application, *Expert Systems with Applications* 2002, Vol. 37, Issue 8, pp. 5564–5576.

Fafuła A., A prototype of platform for data-driven approach to detection of cognitive biases, [in:] J. Korczak (Ed.), *Data Mining and Business Intelligence*, Research Papers of Wrocław University of Economics No. 104, Business Informatics 16, UE, Wrocław 2010, pp. 71–78.

Hernes M., *Metody consensusu w rozwiązywaniu konfliktów wiedzy w wieloagentowym systemie wspomagania decyzji*, Praca doktorska, Uniwersytet Ekonomiczny we Wrocławiu, 2011.
Korczak J., Lipinski P., Systemy agentowe we wspomaganiu decyzji na rynku papierów wartościowych, [in:] S. Stanek et al. (Ed.), *Rozwój informatycznych systemów wieloagentowych w środowiskach społeczno-gospodarczych*, Placet, 2008, pp. 289–301.
Korczak J., Bac M., Drelczuk K., Fafuła A., A-Trader – consulting agent platform for stock exchange gamblers, [in:] M. Ganzha, L. Maciaszek, M. Paprzycki (Eds.), *Proceedings of the Federated Conference on Computer Science and Information Systems FedCSIS 2012*, Polskie Towarzystwo Informatyczne, IEEE Computer Society Press, Warsaw, Los Alamitos, 2012, pp. 963–968.
Lipiński P., *Evolutionary Data-Mining Methods in Discovering Stock Market Expertise from Financial Series*, PhD thesis, Universite Louis Pasteur, Strasbourg 2004.
Luna F., Perrone A. (Eds.), *Agent-based Methods in Economics and Finance: Simulations in Swarms*, Springer, 2002.
Sundali J., Croson R., Biases in casino betting: The hot hand and the gambler's fallacy, *Judgment and Decision Making* 2006, Vol. 1, No. 1.
Sycara K.P., Decker K., Zeng D., Intelligent agents in portfolio management, [in:] N. Jennings, M. Wooldridge (Eds.), *Agent Technology*, Springer, 2002, pp. 267–282.
Tversky A., Kahneman D., Judgment under uncertainty: Heuristics and biases, *Science* 1982, Vol. 185 (4157), pp. 1124–1131.
Westerhoff F.H., Multiasset market dynamics, *Macroeconomic Dynamics* 2004, Vol. 8, pp. 596–616.

Websites

[WWW1] http://www.w3.org/TR/2003/WD-soap12-mtom-20030721/.
[WWW2] http://www.w3.org/TR/soap/.
[WWW3] http://www.xml.com/pub/a/2003/02/26/binaryxml.html.
[WWW4] http://jade.tilab.com/.

WIELOAGENTOWY SYSTEM DLA GRACZY GIEŁDOWYCH

Streszczenie: W artykule zaprezentowano architekturę wieloagentowego systemu wspierającego decyzje inwestycyjne. Omówiono poszczególne elementy systemu, sposób komunikacji pomiędzy nimi, mechanizm oceniania poszczególnych agentów, łączenie ich wspólnych sygnałów otwierania/zamykania pozycji i ponowne uczenie przy pomocy dobranych danych, co tworzy koncepcję procesu niekończącego się uczenia. Opisane zostały nowe metody transformacji szeregów finansowych, behawioralny model zachowań graczy giełdowych i sposoby przełożenia zamodelowanych wzorców na sygnały otwarcia i zamknięcia pozycji. W podsumowaniu opisano wyniki badań oraz wskazano kierunki dalszego rozwoju platformy.

Słowa kluczowe: system wieloagentowy, rynki finansowe, systemy doradcze na giełdzie, sztuczna inteligencja, analiza szeregów czasowych.