Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 963–968
ISBN 978-83-60810-51-4

A-Trader – consulting agent platform for stock exchange gamblers

Jerzy Korczak, Maciej Bac, Krzysztof Drelczuk, Aleksander Fafuła
Wroclaw University of Economics ul. Komandorska 118/120, 53-345 Wroclaw, Poland
Email: {jerzy.korczak, maciej.bac, krzysztof.drelczuk, aleksander.fafula}ue.wroc.pl

Abstract—**The authors of this paper present the architecture of a multi-agent system which supports investment decisions. The individual components of the system, the manner of communication between them, the mechanism of assessing the individual agents are discussed here. Combining their common open/close position signals and relearning with the use of the selected data create the never-ending learning concept. New methods of transforming financial series, the behavioural model of stock exchange gamblers and the manners of translating the modelled 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.**

I. Introduction

UNDERSTANDABLY the issue of stock exchange data exploration has been meeting with unabated interest for years. A range of more or less effective methods which are aimed at the market analysis and which assist the decision-making process have been developed [2;5;9;14]. Generally, the methods are based on statistics, economics, artificial intelligence, financial knowledge. This study focuses solely on multi-agent systems and suggests a technological solution aiding the purchase and sale decision-making on the securities market.

One of the first solutions of this type is the multi-agent system proposed by K. P. Sycara, D. Zeng and K. Decher [16]. The system enabled the user to cooperate with multiple specialised agents which have access to financial models and supervise the operation of the system, the situation on the market, the environment and the pursuit of the user priorities on the ongoing basis. Numerous interesting concepts of multi-agent systems have been published in the recent years. C. Chiarella, R. Dieci and L. Gardini [8] 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 [18]. V. Bohm and J. Wenzelburger [6] present the evaluation of the shares portfolio optimisation strategy by three agents: rational agent, interference agent and technical 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 is also worth mentioning [LeBaron, 2010].

There are several standards describing the principles of designing the systems based on agent technologies. One of them is the standard developed by the FIPA (Foundation for Intelligent Physical Agents) organisation. It describes the manners of management, communication and cooperation between agents. Another standard is 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 standardisation of agent technologies. The results of the works of the above mentioned associations facilitate agent implementation and integration and their transfer between the existing systems. The available implementations of certain standards are mainly experimental implementations and some of them are no longer maintained. The most popular platform among the JADE ones is maintained in two versions as an open-source platform [24] and independently as a commercial platform – JADE 7 [25]. The commercial variety of JADE is positioned as a base framework, which supplies the basic mechanisms for business applications. It is characterised by a greater speed and scalability than the free variety. Unfortunately, the professional JADE version is free of charge only for non-commercial use and its use licence does not permit the modification of the source code. A platform which is supposed to be the basis for constructing a multi-agent open system for the analysis of financial time series needs to be fast, light and simple to use. The aspect of security and possibility to modify/improve the engine is also important. Speed is not only the instant messaging protocol but also triggering agent responses immediately; the agents should process the demanded information. The lightness of the platform is a low resource consumption overhead related to the information receiving and sending process itself. Unfortunately, the above listed platforms do not satisfy the necessary conditions to serve as the base for the open multi-agent system for analysing financial time series despite the fact that they supply multiple tools and capabilities which would undoubtedly be extremely useful.

The aim of this publication is to present the design and implementation 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 based on any methods. The accuracy of predictions, the orientation on obtaining information continuously, the never-ending improvement of own knowledge base and the ability to adapt to the changing environment were important requirements in the system implementation.

The internet Bourse Expert – Agent Technology (iBE--AT) system developed in the LSIIT-CNRS laboratory in Strasbourg was a prototype of the suggested solution assisting investment decision-making on the securities market [12], [20]. In the iBE-AT system several types of agents were designed: agents responsible for registering and pre-processing of the information from the securities market, rapid response agents which directly observe the market in order to detect anomalies immediately, agents analysing the collected financial data and preparing investment decisions suggested to the user, agents managing the knowledge base and artificial financial experts used to analyse financial data, agents analysing text data, particularly press news related to the companies quoted on the stock exchange, agents visualising financial data and the prepared investment decisions, and security agents monitoring user access to the system. Artificial financial experts recommending purchase and sale decisions were subsets of technical analysis rules selected by the genetic algorithm. Each rule was a function suggesting one of three investment decisions: PURCHASE, SELL or KEEP, calculated based on the recent quotations of prices and share volumes. The genetic algorithm sought such subsets of rules that optimised the objective function specifying the financial efficiency of the expert in a given period.

Another example of the agent technologies application when aiding investment decision-making was the Evolutionary Multi Agent System (EMAS) [1], [12], developed in the Institute of Computer Science, University of Wrocław, consisting of agents which are functionally similar to iBE-AT agents, ones that analyse the situation on the market and suggest investment decisions. The agent society was subject to evolution controlled by several system agents in the computational process, owing to which the agents with low investment skills were eliminated and replaced with the agents that were better adjusted to market conditions. The EMAS system was based on the FIPA standards and implemented with the use of the JADE library, which enabled, among others, the activation of the system on many computers and the maintenance of distributed computing. The iBE-AT and EMAS systems were tested on the actual data obtained, among others, from the Paris, New York and Warsaw stock exchange.

The iBE-AT and EMAS solutions presented above did not fully meet the user expectations. The applications were characterised by an exceedingly slow operation and costly maintenance. Moreover, due to the outdated technologies on the basis of which they were created, there were problems with their development or integration with the applications of other stock exchange gamblers. The A-Trader platform was established as a response to the former drawbacks. The A-Trader project assumes the integration of the platform with the Meta Trader system supplied with data online, including ticks of any securities, goods or currency pairs. The created agents have access to the selected raw data and to the pre-processed data. The paper describes the solution tested on the actual data of the FOREX market related to the major currency pairs.

The paper is divided into three main parts. The first one presents a draft of the physical and logical architecture of the suggested solution. The individual components of the system and the manner of communication between them are discussed. In the second one the issue of the data pre-processing is characterised and the operation of an example agent responsible for preparing the information is described. The concept of the never-ending learning process is explained. The new idea of an agent modelling the behaviours of stock exchange gamblers and the manner of translating the modelled patterns into the open and close position signals are also described. The last – conclusion – part is a description of the results of the carried out research and the indication of its further development direction.

II. System architecture

The presented system is a multiagent solution that supports the analysis of time series of high frequency, such as trading instruments. The main features are its openness for integration and development of new system functionality and ensuring adequate communication between the various agents. The service orientated architecture and cloud computing solves the problem of computing power. Open and easy to implement the communication protocol (SOAP) significantly facilitated the integration of individual solutions (see further in [21], [22], [23]). Ultimately, this protocol will be changed to its own communication protocol which will ensure required fast speed data transfer. Popular in distributed systems PUSH technology significantly accelerates the propagation of information within the system (see further in [19]). Multiagent system allowed us to customize solutions for arbitrary systems of surveillance methods to end users, as well as supervising implemented intelligent agents. Agents of evaluating existing solutions, preparing the training data sets and methods that are able to learn and adapt to ensure the continuing evolution of knowledge describing the behavior of financial markets.

The following agents and components are distinguished in the A-Trader architecture:

- 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. It is an intermediary in sending signals between agents in accordance with the declared indications (see Fig. 1). Each agent the state of which changes informs its Notify Agent. The Notify Agent forwards the information on the status change of a given agent to all 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 newly forming or learning agents according to their needs. This is the Historical Data Agent (HDA). This agent is responsible for supplying agents with the historical data which permit their initiation or which are applied in the learning process.
The next system component is the Cloud of Computing Agents (CCA), consisting of the following (see Fig. 1):

- Basic Agents Cloud (BAC),
- Intelligent Agents Cloud (IAC),
- 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; the data already processed by other CCA agents are rarely applied. They perform predefined operations of a low memory and computing complexity. It enables fast obtainment of the results and forwarding them for further use by the agents with more advanced logic. The result of the operations of the Basic Agents Cloud is a fixed set of data which describe the market and which are used by other agents. From the viewpoint of the solution implementation, SAC may be an integral part of the Notify Agent, which will allow the data transfer and grouping time to be saved when sending them to the target agents.

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). The possibility to observe the results supplied by other agents, to process, "memorise" and learn them makes the agents capable of continuous changes of their operating logic and improvements of their effectiveness. This group of agents includes all the solutions based on artificial intelligence (genetic algorithms, neural networks, expert systems, etc.), agents analysing text messages, agents observing user behaviours. The following are distinguished among these agents: agents responsible for pre-transformation of financial series, agents responsible for modelling the behaviours of stock exchange gamblers, 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 entire solution without the necessity to install the agent on the research team servers. Such an approach ensures that a researcher outside the team does not have to reveal the secret of its agent's operation while being able to receive and send signals, on the one hand. And the external agent does not threaten the stability of the operation of the characterised system, on the other. 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

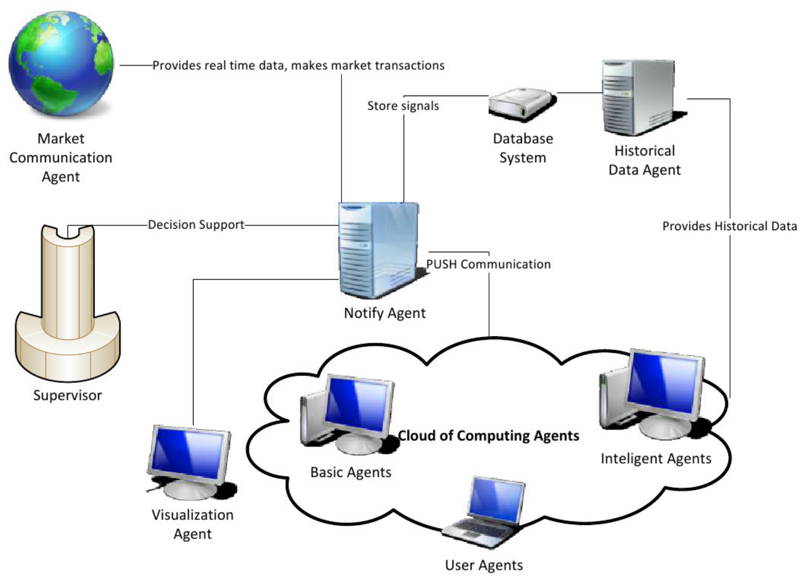

Fig.I. A-Trader system architecture. Source: own work

UAC. After obtaining appropriate stability, the User Agents may be transferred 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 analysing the information from companies and browse the content of the portals writing about financial markets. They can, among others, supply the information on the investor/speculator sentiments by analysing financial experts' blogs or browsing relevant posts on community portals.

Fast and easy visualisation of the results of the agent or the entire system's work is an important aspect in verifying the correctness of its operation. It is possible in the system due to the User Communication Agents (UCA). Visualisation tools permit a 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 analyse learning sets for the Intelligent Agents. The presentation of the signals supplied by the agents of the characterised system does not require the implementation of own tools but is possible due to the integration of the agents with the software supplied by brokers.

The multi-agent system is characterised by the agents being capable of generating independent decisions. These may be mutually consistent or completely contradictory decisions. Such mutually exclusive decisions are e.g. the open and close position suggestions generated by two independent agents at the same time. A conflict between agents occurs then. The conflicts are resolved by Supervisors (S), which observe the decisions of all the Cloud of Computing Agents and the Intelligent Agents, assess their effectiveness in investing and risk. They are responsible for resolving the conflict. 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 [13]. Implementing the Supervisor based on the consensus method may also prove effective [Hernes, 2011]. The decision taken with the use 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 the case where exceptional situations or ones 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), fulfils the role of a directory service (Directory Facilitator). It 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 of 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 with the use 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 to enable the performance of specialised operations on the environments dedicated therefore, e.g. computing algorithms, which may be performed synchronously on the computers equipped with multi-processor NVIDIA graphic cards. Visualisation agents, in turn, may also be activated on the mobile equipment which facilitate the user access to the results of the cloud computing.

A container was implemented on the .NET platform as part of the prototype, which enables the activation of the agents on the computers with the Windows and Unix families systems. The creation of containers for JAVA agents and ones enabling the integration with popular environments, such as MATLAB, is planned at the subsequent stages.

III. Functional description of selected agents

A. Data pre-processing agents

Pre-processing data according to the methodology CRISP-DM (Cross Industry Standard Process for Data Mining) [7] developed by Daimler-Chrystler Analyst, SPSS, and NCR, is known to be the second and third stage of the process of data mining, that is "understanding of the data" and "data preparation".

Most of the data stored in databases is a rough, incomplete and noisy. For example, database may include:

- values that are obsolete or unnecessary,
- missing values,
- outliers,
- data in the wrong format,
- multidimensional dependent variables, etc.

The overall objective of the process is to minimize irrelevant data that enters the model and often cause 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, so it requires a very short time of their reaction. Designers of data mining models, very often are balancing between these two guidelines, to settle 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. It will allow using the automatic optimization of this process and, perhaps most importantly, enable parallelization of processing allowing the preparation of high quality data within an acceptable time.

B. 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 a subject of the algorithm's operation.

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

- The method does not restrict the user with 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.
- The number of agents in unlimited.
- 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 [4], evolutionary algorithms [3], association rules and expert systems [10]. Parameterised 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 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 the 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 with the use 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 specialised 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). It significantly accelerates the specialisation 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.

C. Experimental Behavioural Agent

One of the top-modern elements of the A-Trader system is a module suitable for 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 [11]. 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 [17],
- Hot Hand (when buying and selling) – a belief in trend continuity [15].

During 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.

Finally, the behavioral agent can be used for various purposes. One of them is to warn investors about potential mistakes they could avoid. On the other hand it is possible to point out to them the moments where it is most likely other users of the market would commit a mistake. Therefore it is possible to use this agent as a system for suggesting early market entry points.

Currently the extracted rules were 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 is mainly driven by other forces.

IV. Conclusion

The first attempts to 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 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 thus far. By constructing own agents, it is possible to make use of the information of the assessing agents in order to improve 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 not caring about the basic data and message supply mechanisms.

Creating agents simulating the behaviours 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 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 student theses.

References

[1] Adamczyk, I., *Ewolucyjny system wieloagentowy,* Praca magisterska, Uniwersytet Wrocławski, 2007.

[2] Allen. F., Karjalainen, R., *Using Genetic Algorithms to find Technical trading Rules,* Journ. Of Financial Econ., 51, pp.245-271.

[3] Bac M., Kwaśnicka H., *Możliwości zastosowania algorytmów genetycznych w systemach informacyjnych wspomagających proces podejmowania decyzji gracza giełdowego,* [w:] Inżynieria i systemy ekspertowe, red. A. Grzech, K. Juszczyszyn, H. Kwaśnicka, Akademicka Oficyna Wydawnicza Exit, Warszawa 2009, s. 663-676.

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

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

[6] Bohm, V., Wenzelburger**,** *On the Performance of Efficient Portfolios,* Journal of Economic Dynamics and Control, Volume 29, Issue 4, April 2005, ss. 721-740.

[7] 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.

[8] Chiarella C., Dieci R., Gardini L., Asset price and wealth dynamics in a financial market with heterogeneous agents, Journal of Economic Dynamics and Control, 2011.

[9] Dempster M., Jones, C., A Real Time Adaptive Trading system using Genetic Programming, Quantative Finance, 1, pp. 397-413.

[10] Dymova L., Sevastianov P., Bartosiewicz P., A new approach to the rule-base evidential reasoning: Stock trading expert system application, [w:] Source: EXPERT SYSTEMS WITH APPLICATIONS Volume: 37 Issue: 8 Pages: 5564-5576 DOI: 10.1016/j.eswa.2010.02.056 Published: AUG 2010

[11] Fafuła A., A prototype of platform for data-driven approach to detection of cognitive biases, [in:] Data Mining and Business Intelligence, Ed. J. Korczak, Research papers of Wrocław University of Economics No. 104, Business Informatics 16, UE, Wrocław, str. 71-78.

[12] Korczak J., Lipinski P., Systemy agentowe we wspomaganiu decyzji na rynku papierów wartościowych, [in], Rozwój informatycznych systemów wieloagentowych w środowiskach społeczno-gospodarczych, ed. S. Stanek et al., Placet, 2008, pp. 289-301.

[13] Korczak J., Lipiński P., Technology of intelligent agents used in financial data analysis, [in] Proceedings of 5th Ogólnopolska Konferencja Naukowa Nowoczesne Technologie Informacyjne w Zarządzaniu, NTIZ2006, Prace Naukowe AE, Wydawnictwo Akademii Ekonomicznej, Wrocław, 2006r.

[14] Luna F., Perrone A. (eds), Agent-based Methods in Economics and Finance: Simulations in Swarms, Springer, 2002.

[15] Sundali J., Croson R., Biases in Casino Betting: The Hot Hand and the Gambler's Fallacy, *Judgment and Decision Making*, Vol. 1, Nr 1, (7/2006).

[16] Sycara, K.P., Decker, K., Zeng, D., Intelligent Agents in Portfolio Management, Agent Technology, eds. N. Jennings, M. Wooldridge, Springer, 2002, pp.267-282.

[17] Tversky A.; Kahneman D., Judgment under uncertainty: Heuristics and biases. Science 185 (4157): str. 1124–1131.

[18] Westerhoff F. H., Multiasset market dynamics. Macroeconomic Dynamics, 8, ss. 596—616.

[19] Agarwal S., Toward a Push Scalable Global Internet, Proc. Global Internet Symposium, IEEE Infocom, Shanghai, China, 2011.

[20] Lipiński P., Evolutionary Data-Mining Methods in Discovering Stock Market Expertise from Financial Series", PhD thesis, Universite Louis Pasteur, Strasbourg, 2004.

[21] [WWW1]http://www.w3.org/TR/2003/WD-soap12-mtom-20030721/

[22] [WWW2]http://www.w3.org/TR/soap/

[23] [WWW3]http://www.xml.com/pub/a/2003/02/26/binaryxml.html

[24] [WWW4]http://jade.tilab.com/

[25] [WWW5]http://www.jade.co.nz/jade/index.htm