

Optimization of Communication Principles in Multi-Agent Economic System

Petr BLECHA

University of Hradec Kralove, Hradec Kralove, Czech Republic
petr.blecha@uhk.cz

Abstract. The paper is focused on the memory optimization for the communication mechanisms on markets (goods, services) in the agent-based computational economy – Virtual Economy Project (VEP). For the realization of the business, the facilitators such as matchmaker and broker are used. These agents allow effective distribution of goods and services and thus constitute a key component of supply chain management within the whole model. The design emphasizes the high adaptive potential of the entire trading system and the ongoing adaptation to changing environmental conditions. Mechanisms of indirect communication allow effective scaling of trading on markets of different sizes, it means that they are used for both - retail and wholesale transactions. The main focus is on the possibilities to optimization of the allocation memory for the used algorithms, their comparing and on the prediction for the future extension of the model. The paper is also focused on the description of the key features of the market mechanisms of the model, the description of the algorithms for the implementation of the business, arguments for their introduction in this form, and the description of the practical implementation of the facilitation agents in the AnyLogic 8 environment used for system development.

Keywords: Agent, Optimization, Broker, Agent-based Economic Model, Travel Market.

1 Introduction

Communication is a deliberate exchange of information in multi-agent systems that deliver the production and perception of brands that are drawn from a shared system [10]. Such a shared system can be understood as a language dictionary based on the multi-agent system, which describes standards and rules of communication among the agents.

Communication can be divided into two basic categories according to the aspects from which we can perceive it. If there is a communication principle between two agents where the message from the sender gets directly to the recipient, the communication is direct. On the other hand, if a third agent is used as a mediator, is needed to mediate communication - indirect communication [12].

One of the basic issues of multi-agent systems is the search for certain agents that have (or may have) information the initiating agent searches for [7]. The issue helps

to solve the facilitation agents, which appear in the system as already mentioned mediators. Within this text, attention will be focused primarily on the use of such facilitation agents for the realization of the functioning of the goods and services markets in the environment of the agent-oriented computational economy.

2 Coordination of Agents Through Mediators

In a highly dynamic, competitive and comprehensive environment of goods and services markets, consumer decision-making depends on several individual factors. It can be a network of contacts, innate consumer dynamics, or various external environmental influences [8]. Similarly, it can be considered for non-end consumers - companies, factories, etc. However, this approach assumes that all agents have their own dynamic database that contains accurate information about other agents (location, services provided ...) and which is continuously updated during the system run.

However, the maintaining such a database is very impractical for large multi-agent systems (MAS) - it leads to increased computational difficulty, updating of individual agent databases, or more complex definitions of agent behavior (all types that we assume to be on the goods and services market) unreasonably high computational demands [9].

The mediating agents (mediators) do not contribute directly to addressing the desired goal, but they support the flow of information in the MAS community [14]. At the same time, they enable the distribution of the communication load to logical units or sets of agents according to regional, type, or other competencies. Thanks to their knowledge, which contains the necessary information about all the agents in the system (or a sufficiently large subset of them) [6], they can find agents who are able to meet the target.

3 Virtual Economy Project

The VEP (Virtual Economy Project) is largely oriented to effective distribution of all types of products, which are used in the model. There are applied some principles of indirect communication or principles of offer and demand [13].

Multi-agent model AVE is developed by Anylogic tool (actual version 8.2.). Anylogic is a development environment which runs on the JAVA platform [14]. The platform allows to use three basic approaches to modeling systems [12]:

- Agent-based approach,
- Discrete event,
- System dynamics.

All of these ways are mutually combinable, so it is possible to use all of them in one project [2].

As an example, the behavior of a factory agent (a factory-oriented agent (manufacturing company) - FA) can be mentioned. FA can use behavioral charts

to define behavior that includes communication with facilitators, and system dynamics as well as the process itself.

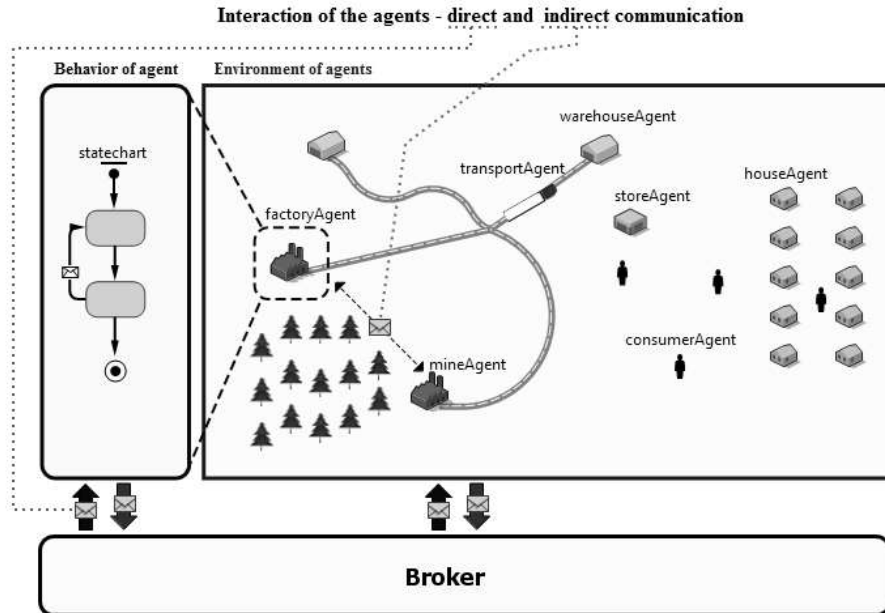


Fig. 1. Graphic representation of the agent interactions [2].

AnyLogic also enables the development of visually rich interactive environments with a simple yet sophisticated animation function [2]. In the figure (see Fig. 1), there is a simplified indication of how the MAS graphic environment can look like, how the agent's behavior can be defined, and what kind of communication between agents is already used in VEP [5].

3.1 Principles of the Markets in AVE

The retail markets serve to distribute products for both end-users (consumer-type agents, CA) and for endless consumers – factories (FA), shops (SA), service providers (transport agents - TA). Retail and wholesale levels are distinguished [1].

The retail market is focused on satisfying the needs of the consumption of end customers, i.e. CA, which in the AVE model represent human individuals. Within the retail market, they constitute an offer of Store agents (Business Agents, SA). These SAs receive FA goods in standardized contracts (in the VEP context in standardized batches, e.g. the volume of goods that the lorry takes) at wholesale goods market. Conversely, they convert from a wholesale format to a retail consumer market format.

Some FAs also enter this market in the form of a seller, despite the fact that they do not have a comparatively wide range of goods compared to SAs that could offer. Nevertheless, they can have a significant share of the retail goods market due to lower

prices for selected products. SAs usually have a higher price due to import charges, marketing costs, and mediation gains and other similar factors.

Compared to the retail market, the wholesale market is primarily focused on non-end customers, primarily FA, which represent the production companies in the model. FAs need certain inputs - energy, labor, and in particular the material needed for production, which they derive from the wholesale commodity market. They distribute their merchandise to Mine Agents (mining agents, they do not need input materials, MA), and several FA ones.

But FAs are the main buyers. Their goal is to produce a certain kind of goods, which is then trying to sell as much as possible. Each FAs has a production chain in which it is defined what components the product consists of, and also in what proportion one the individual components are represented by for a given time period (e.g., day or shift).

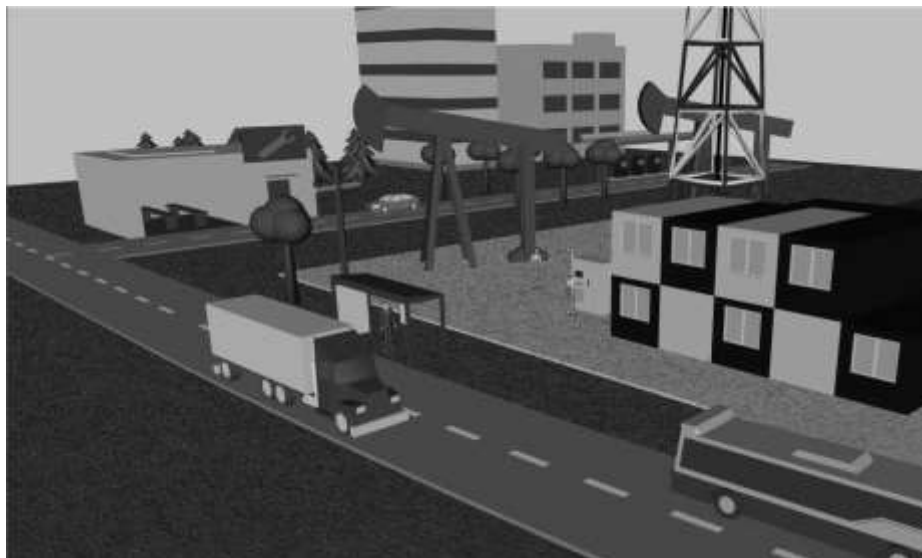


Fig. 2. Representation of the 3D scene in Anylogic [2].

The next type of the markets in the VEP is service market. The market is specific from the previous types because there are variety of types of services – labor, travel, cosmetic, car etc. The labor services are the most important in the VEP that the separate market has been implemented [4].

3.2 Implementation of the Communication in the VEP project

The first phase of verification of the functionality of the model was carried out by ensuring the basic cycle of raw materials and goods in the model, especially among the FAs, who would not be able to realize the production without the required inputs. For the initial verification, whether the theory of raw material flow from MA to FAs will work properly (or whether functional status diagrams for defining FA behavior

were created), several transport agents (TA) were created to fill a warehouse at MA to convert the corresponding amount to the FA that these raw materials require.

Once this method was found to be functional, it was already possible to model the communication itself, where agents could follow a standardized system orders (requirements) and bring the project closer to the realistic principles of the economic market.

Direct communication in the VEP. At the beginning of the project development, there were about 250 instances of the agents. The communication was implemented as a standard procedure where all market participants can communicate with all others to accomplish their own goals [4].

This approach was realized by implementing the complex behavior of each type of agent. There are some parts of the behavior of each agent type:

- Communication protocols with different agent type - different types of agents have different messages that they can respond. Due to this fact, several new communication functionalities of agents were implemented.
- Database of the other agents – agents need to know about the others. It is necessary to determine which agent could help to reach the actual goal (i.e. SAs can sell the food products to satisfy the CA need).
- Comparing mechanisms of the offers (product, raw material, transport offer...). Every agent wants to reach its goal in the most appropriate way. On the retail market of goods it is i.e. the lowest price for the required product, but in the large scale – with calculating with the travel costs due to the location of the shop and the agent location (house).

In a short run of the simulation (about three months of the simulation time) the simulation ran fluently and the behavior of the whole system worked correctly – the Mine agents mined the raw material (coal, iron ore, corn, grain, meat...), distributed their products on the wholesale market database by themselves where the Factory agents bought raw materials, produced the products which have been distributed on the wholesale market database, where the products have been bought by Store agents, then distributed on the retail market database and bought by Consumer agents...

In a long run of the simulation (from three months up to two years of the simulation time) the simulation was not run in fluent and several of the computations were not be finished correctly. The issue was caused by insufficient system performance. There was a high load of the memory and high CPU usage. The performance issue was caused by a large amount of data of the market and agents databases.

All agents used the own databases and the market databases to find the best product for the satisfying their needs. But with the increased data in this databases, the processing algorithm of each of the agent took much longer time than usual and required more memory allocation. In some cases, the free memory was not available and several processes fail.

Within the scalability of the project where the rapid increase of the agents was expected in the final version (more than 100 000), the solution for this issue was proposed - implementation of the indirect communication among the agents.

Indirect communication in the VEP – Broker agent. For the intermediation of indirect communication on the commodity market, a broker agent (broker) was chosen whose operation will be described in detail below.

In general, a broker is an agent that maintains a database of other agents in the system and tries to mediate their request to meet the desired goal. Such a process involves three basic agent roles [7]:

- Applicant - an agent whose goal can be achieved only by another agent,
- Broker – mediator agent,
- The server of the Broker - broker database (can be also an agent).

In our case, the applicant may be an FA (or CA), the broker here is an intermediary that intermediates the chain of steps to be taken to achieve the desired goal, and the broker database can be i.e. classic database (MySQL), an Excel workbook data or special agent. An Excel workbook was used in the first versions of the VEP as a local database but from the Anylogic 8.0 is for the current version used integrated MySQL server within the optimization of the large data processing.

In the model, the broker represents the intermediary between the buyer and the seller in the wholesale/retail market of goods. It handles orders, distributes goods to the market, and mediates product transport from the provider to the client.

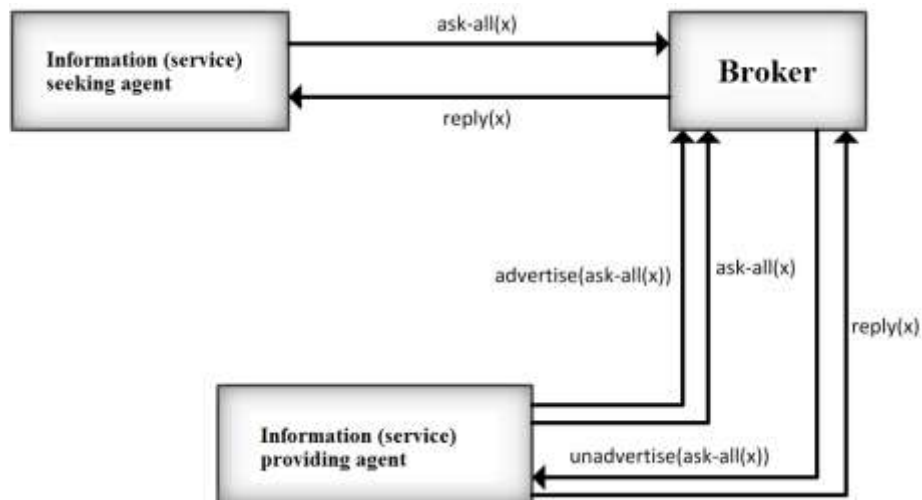


Fig. 3. Broker principle and operations [4].

The broker allows agents to request mediation of the planned deal to reach the agent's goal. Such an agent does not maintain a database of other agent information, he knows only the address of the broker he can contact to satisfy any of his needs (i.e. product distribution, product purchase).

This leads to a significant reduction of the communication flow and higher efficiency of communication. With a higher number of agents in the model (thousands), when

the communication would be enabled among each other, the execution would be extremely demanding for CPU and memory.

To create a more effective indirect communication across product ordering, the Order class was created to define all the necessary ordering features that are important to the broker. The Order class contains a unique identifier for the client (idClient), whereby the broker determines who is the client of the order, the product that the client requests/distributes and the amount. The Broker from this information can calculate the final price which the client is willing to pay.

The Broker has implemented methods to (a) finding other agents to communicate with, (b) finding the most available products in the required amount on the market, (c) recalculating prices on the market (discount before the expiration – usable for the Store agents), (d) updating market databases about new products or sold products, (e) optimize database table (archiving of the records to free up the memory and to optimize finding processes).

These Broker's functions are on the one location in the memory and the behavior of the other agents is much simpler than with the direct communication [3]. With this implementation, the simulation run was more efficient than before. Now, the simulation run fluently for the more than 5 years of the simulation run.

In the latest version of the VEP, the new functionality to optimize the communication was added. Now there are several Broker agents available in the simulation in-stead of one – always one City Broker is implemented per City and one main Broker for communication between City Brokers. It was due to constantly expanding virtual economy environment (from city to district, district to state...).

Overview of the potential memory saves by indirect communication approach.

The indirect communication approach is used within the consumer's finding algorithm where the appropriate service providers are searched. For the potential memory saves overview the procedures from [11] were used. The implemented algorithm is described in Fig. 4.

Table 1. Objects and agents used in the provider finding a cycle.

Object/Agent	Number of occurrences	Memory allocation shortcut (1 unit)
Consumer	N_c	M_c
Provider	N_p	M_p
Broker	$0 \dots 1$	M_b
List	$1 \dots N$	M_l

During the executing of the algorithm, several objects will be processed. The objects and agents will require the relevant allocation of the memory. The over-view of the objects and agents used in service-provider finding algorithm is described in the Table 1.

The total memory usage for the all consumers (TM_c) and providers (TM_p) will be always allocated by the simple formulas:

$$TMc = Mc * Nc \quad (1)$$

$$TMp = Mp * Np \quad (2)$$

In the principle of the direct communication, every consumer agent has to contain the List object for the found providers. The final formula for the memory usage $O(n)$ in this algorithm will be:

$$O(n) = TMc + TMp + Nc * Ml \quad (3)$$

A full algorithm with the better-displayed dependency of the inputs (number of consumers) for total memory usage:

$$O(n) = Nc * (Mc + Ml) + Np * Mp \quad (4)$$

which is the final formula for the total memory usage for all consumers and providers with total memory usage for the all consumer's lists.

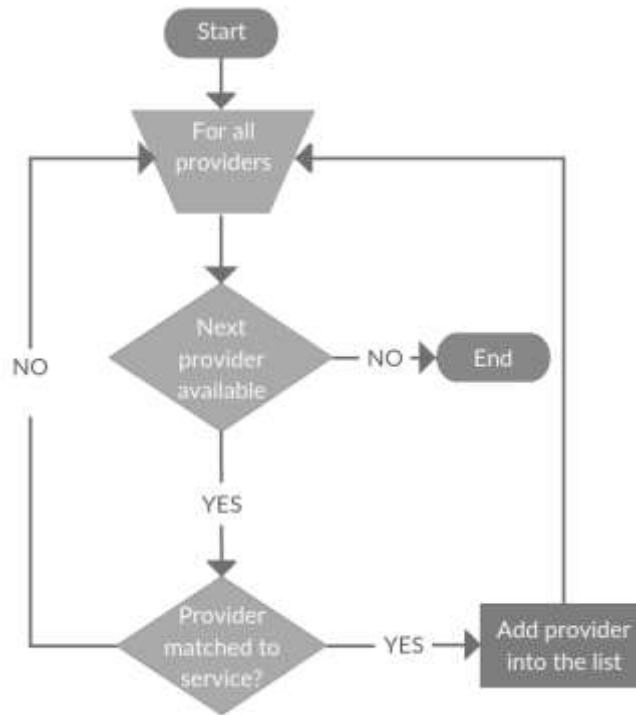


Fig. 4. Diagram of the finding provider algorithm.

In the indirect communication usage, only the Broker agent (which is not implemented in the direct communication approach) contains the List of the providers. The final formula for the memory usage in this algorithm is:

$$O(n) = TMc + TMp + Mb + Ml \quad (5)$$

A full algorithm with the better-displayed dependency of the inputs (number of consumers) for total memory usage:

$$O(n) = Nc * Mc + Np * Ml + Mb + Ml \quad (6)$$

which is the final formula for the total memory usage for all consumers and providers with memory usage for one Broker agent with one list. The predicted formulas can be compared with the measured values.

Real usage of the memory with different communication approaches. For the verification of the optimization the memory usage analysis, the simulation runs with the identical settings were provided for both project implementation - direct and indirect communication approaches.

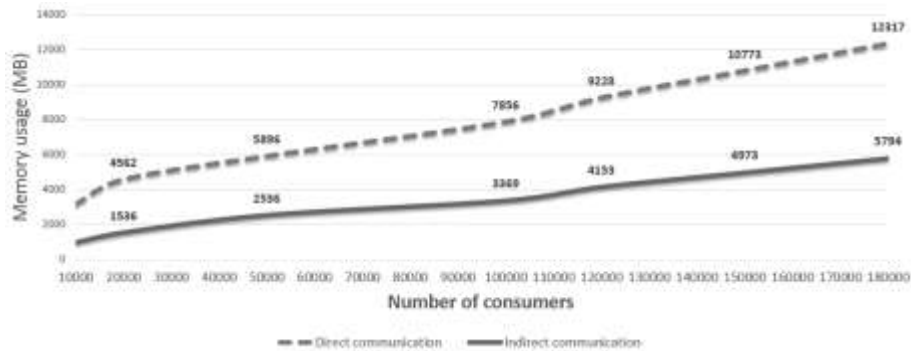


Fig. 5. Comparison of the communication approaches.

In the application the consumers searched for the providers of the tourism services. There were different kinds of the services – based on the locations, prices and capacity. The comparison results are shown in the Fig. 5.

The simulations ran for both approaches for **10 times** (20 in total). The start value of the consumers was **10 000**. After each run (experiment gathers the data for the **100** simulation days) the number of the consumer agents was increased about the next **10 000**.

As it is shown on the graph the simulation ran on the old VEP project version - with the direct communication approach – the memory usage was almost three times higher than on the newer version of the application with indirect communication approach. For example, for the **10 000** of the consumer agents the usage of the memory was **4562** MB for direct communication and **1536** MB for indirect. For the **180 000** of the consumer agents was the memory usage **12317** MB for direct and **5794** MB for indirect.

Compared with the predicted formulas the results reflect the linear function. If we consider that the number of consumer agents is the only value which could be changed (other parameters are constants):

$$O(n) = Nc * (Mc + Ml) + Np * Mp \quad (7)$$

$$O(n) = Nc * (CONST_1 + CONST_2) + (CONST_3 * CONST_4) \quad (8)$$

$$O(n) = Nc * CONST_{12} + CONST_{34} \quad (9)$$

the formulas for the direct communication can be recorded as a classic linear function formula $y = \mathbf{ax} + \mathbf{b}$.

The same principle could be applied to the indirect communication formula:

$$O(n) = Nc * Mc + Np * Mp + Mb + Ml \quad (10)$$

$$O(n) = Nc * CONST_5 + (CONST_6 * CONST_7) + (CONST_8 + CONST_9) \quad (11)$$

$$O(n) = Nc * CONST_{5} + (CONST_{67} + CONST_{89}) \quad (12)$$

$$O(n) = a * x + b \quad (13)$$

where $O(n)$ is total allocated memory: $y = \mathbf{ax} + \mathbf{b}$.

The real formula from the measured data results was determined by the statistic functions in the **R** program. For the direct communication the formula was determined as follows:

$$y = 0.0509x + 3102.7 \quad (14)$$

and for the indirect communication:

$$y = 0.027 + 900.94 \quad (15)$$

The results correspond with the predicted formulas – the direct communication has the higher constant value than indirect ($Mc + Ml$ instead of only Mc) and smaller b constant value than indirect (only $Np * Mp$ instead of $Np * Mp + Mb + Ml$). The linear increase of the memory was proved for both approaches.

4 Conclusion

The optimization of the memory and CPU usage is very important part of the continuously extended multi-agent systems. Within the increasing of the number of the agents and more complex behavior of the agents in the environment, the simulation run needs more HW performance. This could be solved by sequential updating of the HW - better CPU, more memory etc., but the price will be high.

The other solution is to optimize the processes in the model. In this paper, the optimization of the model simulation was implemented by changing the communication approach from the direct communication, when all agents need to be informed about the others and the behavior of this agents had to be more complex, to indirect.

Within the indirect communication, the complex algorithms are only in the facilitation agent Broker which was responsible for the providing of the whole request of the other agents.

The optimization was verified by the comparing of the memory usage across the simulation runs on both communication approaches.

Acknowledgements. This work was supported by the FIM UHK under Grant of Specific Research Project “Information and knowledge management and cognitive science in tourism”.

References

1. Algarvio, H., Lopes, F., Santana J.: Multi-agent retail energy markets: bilateral contracting and coalitions of end-use customers. 12th International Conference on the European Energy Market (EEM), pp. 1-5, IEEE, Lisbon (2015).
2. ANYLOGIC Homepage, <http://www.anylogic.com>, last accessed 2018/05/27.
3. Baert, Q., Caron, A., Morge, M., Routier, J.: Fair multi-agent task allocation for large datasets analysis. Knowledge and Information Systems, vol. 55, pp. 591–615, Springer, London (2017).
4. Blecha, P.: Labor Market In Multi-Agent Environment Modeling Virtual Economy. In: Hradec Economic Days, vol. 8, pp. 114–120, Hradec Kralove (2018).
5. Bures V., Blecha P., Tucnik P.: Agent-Based Modelling and Simulations as an Integral Functionality of the Business Intelligence Framework. In: Dregvaite G., Damasevicius R. (eds) Information and Software Technologies. Communications in Computer and Information Science, vol. 538, pp. 236–245, Springer, Druskininkai (2015).
6. Byrski, A., Drezewski, R., Siwik, L., Kisiel-Dorohinicki, M.: Evolutionary multi-agent systems. The Knowledge Engineering Review, pp. 171–186, Cambridge University Press (2015).
7. Decker, K., Williamson, M., Sycara, K.: Matchmaking and brokering. In: Proceedings of the Second International Conference on Multi-Agent Systems, ICMAS-96, vol. 432. (1996).
8. Garifullin, M., Borshchev, A., Popkov, T.: Using AnyLogic and agent-based approach to model consumer market. In: Proceedings of the 6th EU ROSIM Congress on Modelling and Simulation, pp. 1–5. (2007).
9. Parker, J.: Coordination in large scale multi-agent systems for complex environments. In: Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems, pp. 1751–1752 International Foundation for Autonomous Agents and Multi-agent Systems, Paris (2014).
10. Russell, S. J., Norvig, P.: Artificial intelligence: a modern approach. Pearson Education Limited, Malaysia (2016).
11. Sipser, M.: Introduction to the Theory of Computation. Course Technology, 2nd edition, (2005).
12. Spiro, R. J., Bertram C. B., Brewer, W. F.: Theoretical issues in reading comprehension: Perspectives from cognitive psychology, linguistics, artificial intelligence and education. vol. 11. Routledge (2017).

13. Tucnik, P., Blecha, P., Kovarnik, J.: Adaptation to Market Development Through Price Setting Strategies in Agent-Based Artificial Economic Model, vol. 10448, pp. 548–557, ICCI, Cyprus (2017).
14. Wagner, T., Rana, O.: Infrastructure for agents, multi-agent systems, and scalable multi-agent systems. In: International Workshop on Infrastructure for Scalable Multi-Agent Systems, Barcelona, pp. 3–7, Spain (2001).