Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 35 (2014) 1416 – 1425
Procedia Computer Science

18th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems - KES2014

MAREA – from an agent simulation application to the social network analysis

Dominik Vymětal, Roman Šperka*

Silesian University in Opava, School of Business Administration in Karviná, Univerzitní nám. 1934/3, 733 40, Karviná, Czech Republic

Abstract

The aim of the paper is to present an enhanced software simulation application based on a trading company control loop and to validate agents' behavior from simulation experiments using simple Petri net, and social network analysis. The main purpose of the application is to improve existing decision support systems with the use of a simulation. The ERP system using the REA ontology approach is used as a measuring element in the application. The system has been developed in cooperation between Silesian University in Opava, School of Business Administration in Karvina, Czech Republic and REA technology Copenhagen, Denmark. After the prototype tests at the end of the year 2011, we presented it at the beginning of 2012 for the very first time. Firstly, the enhanced framework with several types of agents and negotiation possibilities is described. This is followed by the decision function explanation, which is the core of the price negotiation. Secondly, a brief look on the graphical user interface and main parts of MAREA simulation monitor is provided. Brief results of the model validation performed by means of ProM software are presented. To conclude, MAREA is a software application with simulation possibilities, which can be used to present trading behavior of a company for decision support.

Peer-review under responsibility of KES International.

Keywords: decision support; MAREA; agents; simulation; application; social network

1. Introduction

The companies have to ensure the flexibility of their behavior, speed of decisions, and customer satisfaction leading to the optimal market share, profits and other key performance indicators (KPIs) in order to survive in a

* Corresponding author. Tel.: +420 596 398 282.
E-mail address: sperka@opf.slu.cz.

Peer-review under responsibility of KES International.
doi:10.1016/j.procs.2014.08.198

global and turbulent market environment. The aim of this paper is to introduce the usage of an application prototype used for simulations of a business company to be dealt as a decision support system.

Simulations supporting decision support systems are typically based on business process modeling treated by many researchers; e.g., Axenath et al.[1], Davenport[2], Eriksson and Penker[3], Van der Aalst[4], Bucki and Suchánek[5], and Šperka and Spišák[6-7]. Alternative enterprise modeling methods - value chain oriented models have obtained much attention from researchers in the accounting domain and later, from enterprise modeling. Value chain modeling idea originally based on well-known Porter´s model concentrates on the value flows both inside the enterprise and on the value exchange with the environment. Currently, the most popular value chain enterprise methodologies are e3-value in Gordijn[8], and the REA (Resources, Events, Agents) ontology; e.g., McCarthy[9], Hruby et al.[10], and Dunn et al.[11]. REA was proposed by McCarthy[9] with the aim to resolve issues specific to the double-entry bookkeeping. It was gradually expanded upon by Geerts and McCarthy; e.g.[12] into the enterprise ontology. It is generally accepted that the REA concept has a potential to be implemented within the design of new ERP systems according to Van den Bossche and Wortmann[13], and recently also to auditing; e.g., Weigand and Elsas[14].

However, business process and value chain modeling methods often omit the fact that any business process runs in an environment with complex social relations. This is why they often meet difficulties when some social behavior like negotiation, management specific methods, market disturbances and others come into consideration. In such cases, some local intelligence within a business process model can be of an advantage. This is probably the main reason, why a new software modeling paradigm came into existence – namely the multi-agent modeling approach. Per definition, software agents can act to some extent independently and use their own targets, aims and cooperation in order to the global target of the modeled system. Modeling and simulation using multi-agent systems (Agent-based modeling and simulation) can be seen as a new perspective of a system modeling, especially for decision-making support systems; e.g., Macal and North[15], and Wooldridge[16]. In Vymětal and Scheller[17] we presented a general agent-oriented simulation application MAREA (Multi-agent REA framework).

Keeping this application in mind, further research focusing on a possibility to define model evaluation methods based on the model outputs is presented in this paper. The result of this research is an extended, REA based prototype in which the REA database is used for data definition, collection and evaluation and the processes are represented by software agents. The actions of agents are logged for further evaluation. The prototype can be used both in standalone PC environment and over Internet. We present the developed prototype, its general structure, the graphical user interface and also some methods of the model validation. The paper is structured as follows. First, a general model structure is presented. Next, graphical user interface is introduced. As a result of a typical simulation run, a typical output log file is used as an input for the model validation by means of a simplified Petri net and social network. Conclusion sums up the results obtained and outlines of the next research targets.

2. Simulation model

The model implemented in the simulation application utilizes a virtual business company using the REA value chain and the multi-agent system as the active element. The general model structure is presented in Fig. 1. The model uses the control loop paradigm. The internal parts of the company are represented by several agent types such as sales representative, purchase representatives and marketing agents. The outputs of the company are measured by the REA based ERP system. The communication among the agents is carried out with a set of defined messages and corresponding message handlers. These activities are recorded in a special XES-type log file for further use in the model analysis and evaluation (see later). The market environment is represented by customer and vendor agents. Note that all the agents mentioned exist in a large number of instances. The difference between measured outputs and targets is used as a feedback for the management agent, who regularly (e.g. once per month) takes necessary management actions. The whole company structure can be extended by other agents such as production line manager controlling the production of a company or an accountant agent who registers the REA based data and computes KPIs such as overhead costs and others (in Fig. 1 are these agents presented in dotted line).

The interaction between the customers and sales representatives and also between the vendors and purchase representatives is represented by sales and purchase quote requests, the quotes themselves followed by price negotiations. This is modeled by the classical contract net protocol. The customer decides if he should accept the quotation based on the decision function presented earlier in e.g., Šperka and Vymětal[18]. If the proposal is not accepted, the sales representative changes the price accordingly (this is one of the parameters that can be changed by the modelers). As the decision function can be seen as a core part of the model, we present here this function like in Šperka and Vymětal[18].

The decision function for *m*-th sales representative negotiating with *i*-th customer is represented by equation 1.

$$c_n^m = \frac{\tau_n T_n \gamma \rho_m}{O v_n} \tag{1}$$

- c_n^m - price of n-th product offered by m-th sales representative,
- τ_n - market share of the company for n-th product $0 < \tau_n < 1$,
- T_n - market volume for n-th product in local currency,
- γ - competition coefficient, lowering the success of the sale $0 < \gamma \leq 1$,
- ρ_m - m-th sales representative ability to sell $0.5 \leq \rho_m \leq 2$,
- O - number of sales orders for the simulated time,
- v_n - average quantity of the n-th product, ordered by i-th customer from m-th sales representative.

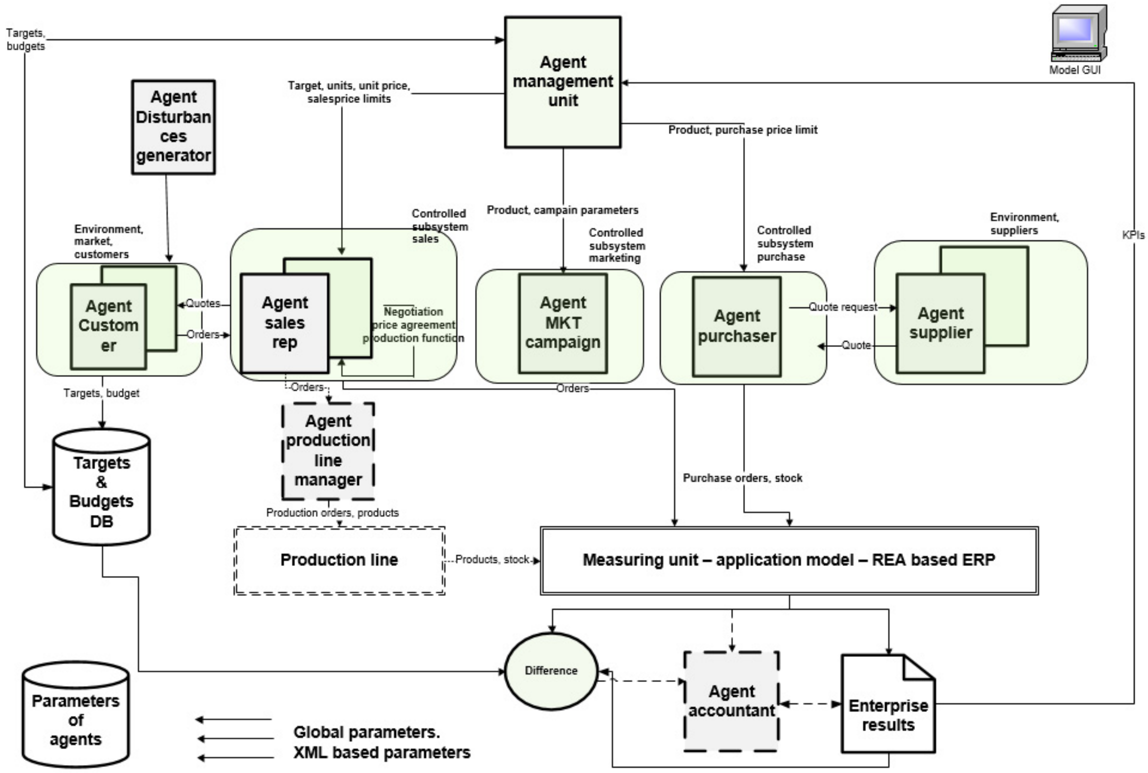

Fig. 1. Generic model of a business company. Source: adapted from[17,18].

Similar decision function is used in the vendor – purchase representatives negotiation. The aforementioned parameters represent global simulation parameters set for each simulation experiment. Other global simulation parameters are: lower limit sales price, number of customers, number of sales representatives, number of iterations, and mean sales request probability. The more exact parameters can be delivered by the real company, the more realistic simulation results can be obtained. In case we would not be able to use the expected number of sales orders *O* equation 2 can be used.

$$O = ZIp \text{ where} \tag{2}$$

- Z – number of customers
- I – number of iterations,
- P – mean sales request probability in one iteration.

The presented decision function is generally based on the overall market balance for each product. In this sense it does not reflect the idea of agent´s independency. Quite different approach can be used for the customers' decision whether to buy or not. This approach is based on the utility theory and needs data for the preferences and budget constraints. The utility-based approach is out of scope this paper and an object for further research.

Among the actions that the management agent can take is the change of the purchase limit price, a decision upon a sales representative or purchase representative education, start of a marketing campaign, and others. As an example of the closed control loop we can use a case of a sales representative education. This education increases his ability to sell (ρ_m). This again helps to achieve a better negotiated price (see equation 1). The parameters of management actions can be set by the modelers. With such general structure, the modelers are able to configure the agent types, the management action parameters and others and to observe the behavior of the system reacting to the state of the environment generally.

3. MAREA introduction

Two main components, the multi-agent system (MAS) and the REA based Enterprise Resource Planning system (ERP) comprise the MAREA application. The simulation designer or modeler can interact with the system by means of a Graphical user interface (GUI). The general overview is presented in Fig. 2.

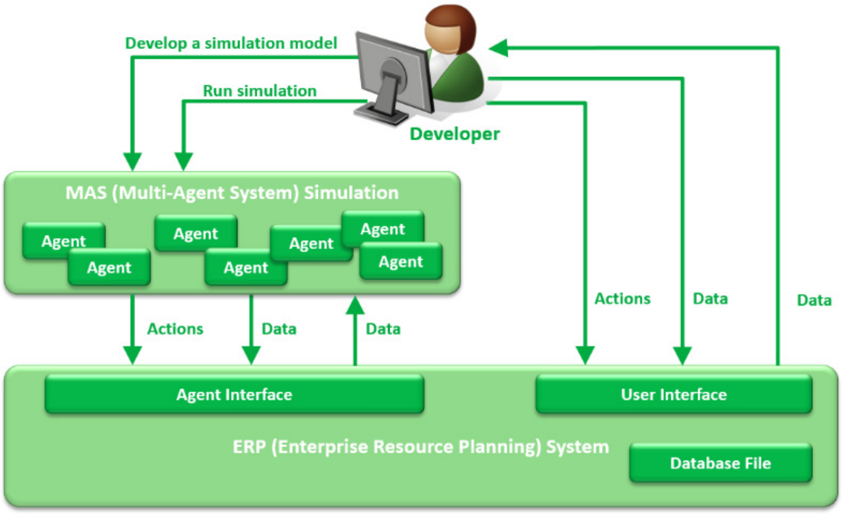

Fig. 2. Application overview. Source: adapted from REA Technology official documentation.

Within the ERP system next important elements of the application namely the Message viewer and the log file are hidden. A modeler can either use the ERP system directly, or can configure or add new intelligent agents to perform the same activities that a human user can perform. For example, a simulation designer can use the ERP system directly to create initial data for a simulation, then start the agent platform to run a simulation. By means of the ERP system the designer or modeler can inspect the simulation results, and even adjust the data (within the rules implemented in the ERP system) and then start the agent platform to continue running the simulation. Both agents and a human user can read data from the ERP system, write data to the ERP system, and perform actions, such as sending a purchase order.

The modeler can set up the trading company parameters (see the example in Table 1) and run trading simulation for a specific time to interpret the development of KPIs of this company. Similarly, the designer can evaluate the log files to evaluate the correctness of the model.

Table 1. Basic simulation parameters.

Parameter	Example	Remark
Number of iterations	365	52 weeks also possible
Number of customers	100	Up to several thousand
Number of vendors	5	Typically
Mean quantity in one sales order	5 units	
Probability of sales order request	0,1	
Number of sales representatives	2-3	One for e-business modeling
Sales representative ability	0,7	For start

The main screen called Simulation monitor (Fig. 3) consists of five panes:

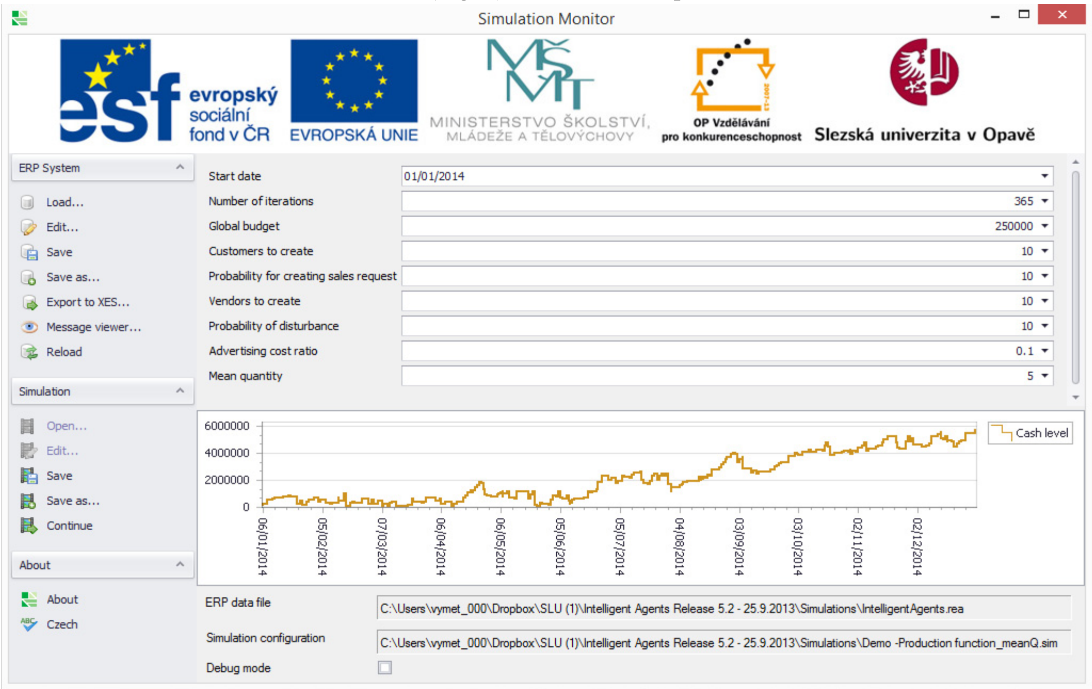

Fig. 3. Description of the simulation monitor. Source: own.

- Logo pane with sponsors of the project.
- Simulation properties. These properties are determined by the agent simulation model, and different simulation models might contain different simulation properties.
- ERP system menu. Within the ERP system menu, two very important buttons can be seen: the Export to XES button and the Message viewer button. The functions and usage of these two buttons will be described in the next section.
- Simulation menu.

- A graph of the cash level of the company, indicating the progress of the simulation. The cash level fluctuations represent the money inflow as a result of realized sales orders, and the cash outflow caused by purchase orders, sales representatives' commissions and other expenses such as overhead costs.

The agents communicate and cooperate by means of messages. The application uses a simple communication scheme based on a tuple message – message handler. An example of the message handler is presented in Fig. 4. All messages sent among the agents including the messages the agents send to each other during negotiation are recorded in a message log file. They can be seen by means of the Message viewer and can be also filtered on the message type. The example negotiation between sales representative agent and the customer agent is presented in Fig. 5. Here we can see a result of a multi-round negotiation. The originally proposed price of the Table (2500) was negotiated down to 1250. Here again: the parameters of a price reduction within a negotiation are a part of global system parameters that can be changed by the modellers.

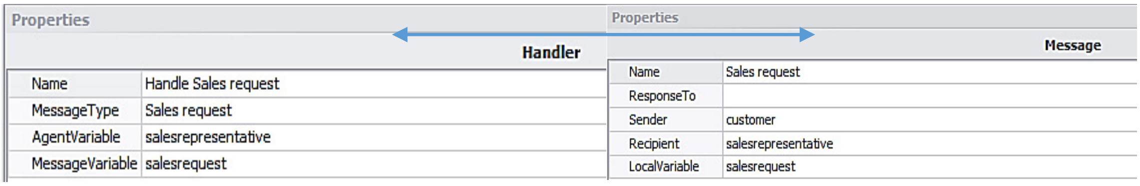

Fig. 4. Description of the simulation monitor. Source: own.

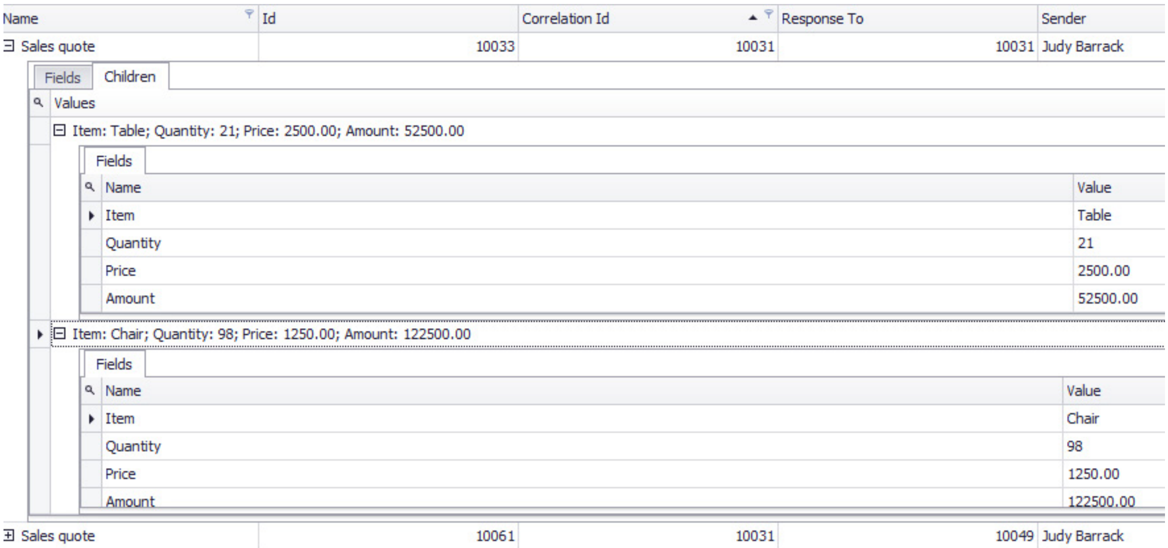

Fig. 5. Sample negotiation. Source: own.

4. Analysis of simulation outputs

Simulation monitor allows to start, pause, resume, and continue the simulations and to open the ERP files. Before the simulation starts, the initial values of the ERP records such as the customers, the vendors, the products etc. can be set up (Fig. 6). The results of the simulation run are recorded in the ERP system during each simulation step. The values of the most important KPIs can be exported to MS Excel files for further analysis. The complete ERP database file can be saved for future use.

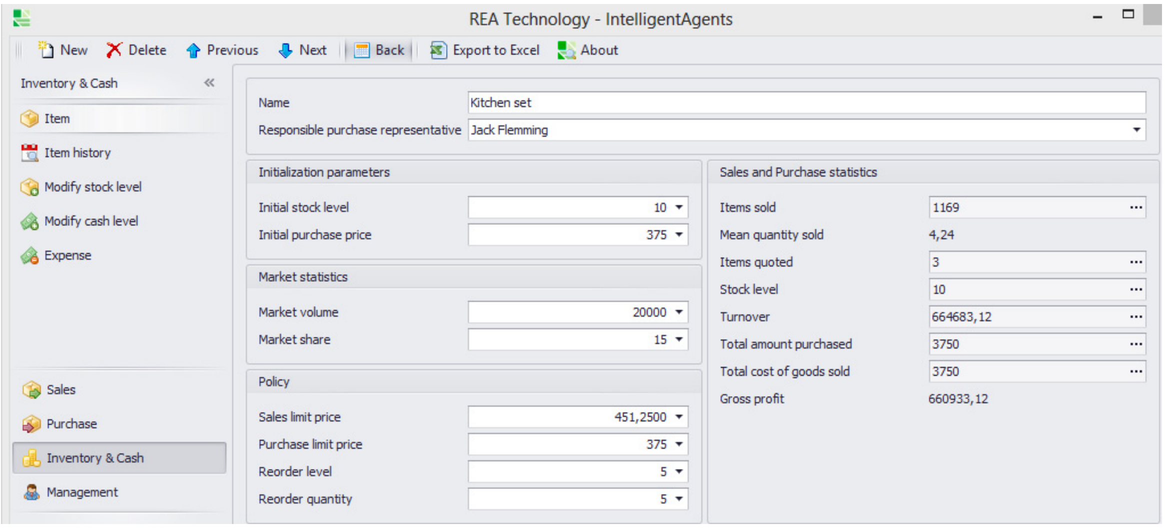

Fig. 6. Part of the resulting ERP – simulation results. Source: own.

After the simulation run, the validation of the agents' structure can be carried out. For this analysis, the agents actions are recorded in a special log file in XES format; e.g. Van der Aalst[19]. The XES log file is then analysed by means of the ProM 6.3 software developed by the Technical University of Eindhoven. Two outputs form the ProM software are used in order to check the structure of the system. The simplified Petri net was generated using the Alpha-algorithm defined in Van der Aalst[19]. This Petri net allows for workflow and control structure evaluation.

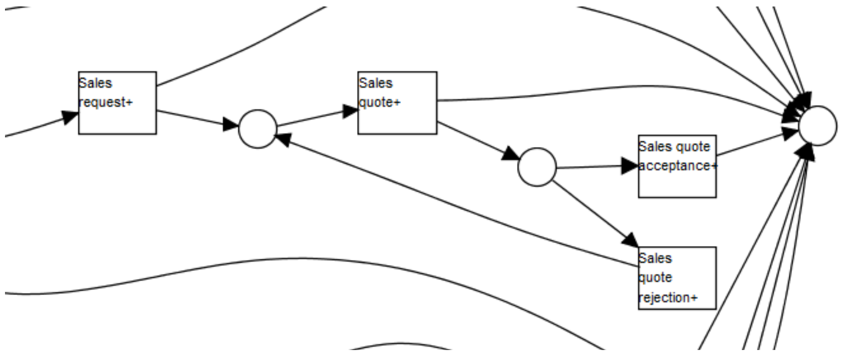

Fig. 7. Fragment of negotiating Petri net. Source: own

In Fig.7, a part of the resulting Petri net concerning the sales negotiation is presented. It demonstrates that the contract net protocol modeled by the customer and sales representative agents was realized correctly. No other ProM process mining tools were used at this stage of the research, the aim of the evaluation of control processes could be achieved by means of the Alpha-algorithm. The next step of the organizational model is the evaluation targeted on the correct co-operation of the agents, in other words, how the "work" flows through the model. So e.g., all vendor agents should exchange information with the purchase representative agents, all customers should be connected with sales representatives and so on. No agent of the model should be modeled as a standalone one. ProM tool proposes several perspectives for such type of analysis that can be used for this purpose based on social network analysis principles. A plethora of notions and metrics has been defined to analyse social networks and the importance of the relationships between the nodes (e.g., Wasserman and Faust[20]). In our case, the nodes of the social network were the agents, and the arcs correspond to relationships between them. From five types of ProM social network analysis namely, the reassignment analysis, similar work analysis, subcontracting analysis, handover-of-work analysis and working together analysis, the last one with "simultaneous appearance ratio" parameter was used for the model evaluation The reason for this choice was the aim of the evaluation – to prove that work flows correctly from one agent type to another one. However, it could be noted that similar result could be achieved by means of "handover-of-work" type of analysis.

In Fig. 8 a working together type of social network analysis is presented. Here, for better readability the result of only 10 customer model was used. We can see here two clusters: the customer cluster concentrated around Peter Hanson – the sales representative and the vendor cluster around Jack Fleming – the purchase representative. Please note that this final result was achieved after several model corrections based on the analysis results. Taking the results of Petri net analysis and working together analysis in consideration it can be assumed that the model structure reflects real process structure of the modeled company. Using such types of analysis the system structure can be validated step-by-step and the agents' cooperation algorithms as well as their internal structure can be improved.

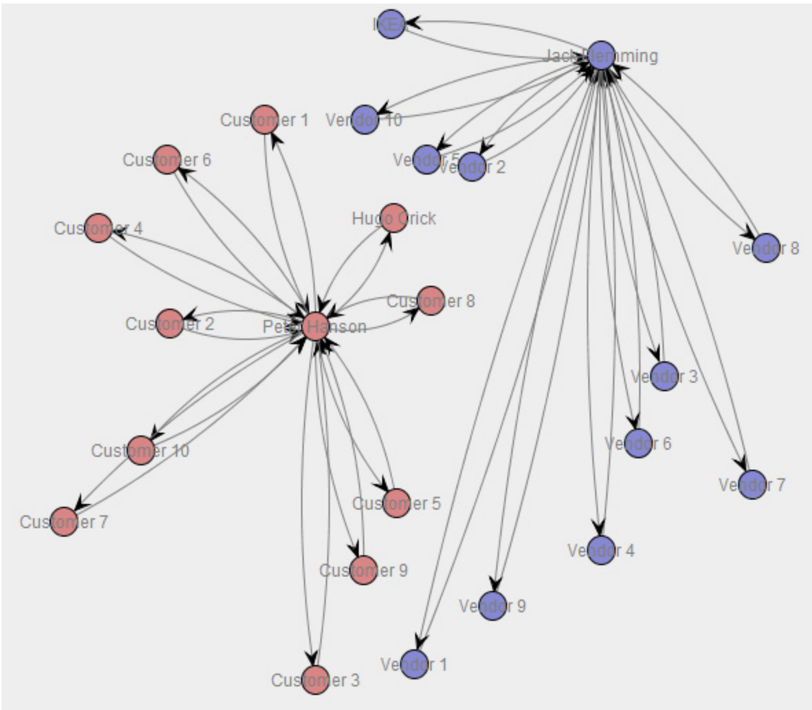

Fig. 8. Social network for working together perspective. Source: own.

5. Conclusion

We presented basic parts of MAREA application in this paper as a prototype of a simulation based software framework. A general structure of the simulation model, basic participants and simulation steps were presented. The MAREA application serves for the decision support of company's management and can also be used for educational purposes. The setup of the application provides possibilities to edit the company parameters and to run trading simulations. This allows users to analyze trading behavior back-to-back according to the parameters setup. The prototype was tested using real data gathered from the ERP system of high-tech Slovakian company and using randomly generated data as well. The most important features of MAREA are: model parameterization, a possibility to configure the model also on the source code level of the agents and simulation speed of the framework. The new features have been added to the original application, namely the possibility to create the process log and the message viewer. Both new features allow a deep evaluation of the model structure using Petri net and social net analysis be means of process mining software. Future research will concentrate on further log files analysis to give us feedback about processes in the running simulation experiment and for new approach to the customer decision function based on the utility theory.

Acknowledgements

This work was supported by the grant Innovation of Study Programs at Silesian University in Opava, School of Business Administration in Karvina, Reg. no. CZ.1.07/2.2.00/28.0017.

References

1. Axenath B, Kindler E, Rubin V. AMFIBIA: a meta-model for integrating business process modeling aspects. *International Journal of Business Process Integration and Management*, **2** (2); 2007, pp. 120-131.
2. Davenport T. *Process Innovation: Re-engineering Work through Information Technology*. Boston: Harvard Business School Press, 1992.
3. Eriksson HE, Penker M. *Business Modeling with UML*. New York: John Wiley and Sons, Inc., 480 p. ISBN 978-0471295518, 2006.
4. Van der Aalst, WMP. Business Process management: a Personal View. *Business Process Management Journal*, **10**(2), p.5, 2004.
5. Bucki R, Suchánek P. The Method of Logistic Optimization in E-commerce. *Journal of Universal Computer Science*. Volume 18, Issue 10, p. 1238-1258. ISSN 09486968, 2012.
6. Šperka R, Spišák M. Transaction Costs Influence on the Stability of Financial Market: Agent-based Simulation. *Journal of Business Economics and Management*, Taylor & Francis, London, United Kingdom, 2013. Volume **14**, Supplement 1, pp. S1-S12, DOI: 10.3846/16111699.2012.701227. Print ISSN 1611-1699, Online ISSN 2029-4433. Available from: <http://www.tandfonline.com/doi/abs/10.3846/16111699.2012.701227#.Ur80j9LuLy0>. (Accessed 31 January 2014), 2013.
7. Šperka R, Spišák M. Tobin Tax Introduction and Risk Analysis in the Java Simulation, In: *Proc. 30th International Conference Mathematical Methods in Economics*. Part II. Silesian University in Opava, Karvina, Czech Republic, pp. 885-890, ISBN 978-80-7248-779-0, 2012.
8. Gordijn J, Akkermans H. Value Based Requirements Engineering: Exploring Innovative e-Commerce Ideas. *Requirements engineering*, **8**, No 2. Available from: <http://www.springerlink.com/content/5ne64nnr0jun0nxk>. (Accessed 22 March 2010), 2003.
9. McCarthy WE. The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment. *The Accounting Review* (July 1982); 1982, p. 554-578.
10. Hruby P, Kiehn J, Scheller C. *Model/Driven Design Using Business Patterns*. Berlin Heidelberg: Springer Verlag. ISBN: 3-540-30154-2, 2006.
11. Dunn CL, Cherrington OJ, Hollander AS. *Enterprise Information Systems: A Pattern Based Approach*. New York: McGraw-Hill/Irwin, 540 p. ISBN 0-07-2404-29-9, 2004.
12. Geerts GL, McCarthy WE. Policy-Level Specification in REA Enterprise Information Systems. *Journal of Information Systems*. Vol. **20**, No. 2; 2006, p. 37-63.
13. Van den Bossche PEA, Wortmann JC. Why accounting data models from research are not incorporated in ERP systems. Paper presented at the *2nd International REA Workshop*, June 25, Fira Santorini Island, Greece, 2006.
14. Weigand H, Elsas P. Model-based auditing using REA. *International Journal of Accounting Information Systems*, Vol. **13**, Issue 3 (2012); 2012, p. 287 – 310.
15. Macal C, North MJ. Tutorial on agent-based modeling and simulation, Part 2: How to model with agents, In: *Proceedings of the 2006 Winter simulation conference*. [online], Available from: <http://www.informs-sim.org/wsc06papers/008.pdf>. (Accessed 20 December 2010), 2006.
16. Wooldridge M. *MultiAgent Systems : An Introduction*. 2nd edition. Chichester: John Wiley & Sons Ltd, 2009. 461 p. ISBN 978-0-470-51946-2, 2009.
17. Vymětal D, Scheller C. MAREA: Multi-Agent REA-Based Business Process Simulation Framework. In: *Proc. International Scientific Conference ICT for Competitiveness*. Silesian Universitz in Opava, p. 301-310. ISBN 978-80-7248-731-8, 2012.
18. Šperka R, Vymětal D. MAREA - an Education Application for Trading Company Simulation based on REA Principles. In: *Proc. Advances in Education Research*. Vol. **30**. Information, Communication and Education Application. USA. pp. 140-147. ISBN 978-1-61275-056-9, 2013.

19. Van der Aalst WMP. *Process Mining. Discovery, Conformance and Enhancement of Business Processes*. Berlin Heidelberg: Springer-Verlag, Germany, XVI, 352 p. ISBN 978-3-642-19345-3, 2011.
20. Wasserman S, Faust K. *Social Network Analysis: Mathods and Applicaions.* Cambridge: Cambidge university press, 825 p. ISBN: 0-521-38707-8, 1994.