Control Loop Model of Virtual Company in BPM Simulation

Roman Šperka[1], Marek Spišák[1], Kateřina Slaninová[1],
Jan Martinovič[2] and Pavla Drážd ilová[2]

[1] Silesian University in Opava,
School of Business Administration in Karviná,
Univerzitní nám. 1934/3a,
733 40, Karvin, Czech Republic
{sperka,spisak,slaninova}@opf.slu.cz
[2] VŠB Technical University in Ostrava,
Faculty of Electrical Engineering,
17. listopadu 15/2172,
708 33 Ostrava, Czech Republic
{jan.martinovic,pavla.drazdilova}@vsb.cz

Abstract. The motivation of the paper is to introduce agent-based technology in the business process simulation. As in other cases, such simulation needs sufficient input data. However, in the case of business systems, real business data are not always available. Therefore, multi-agent systems often operate with randomly (resp. pseudo randomly) generated parameters. This method can also represent unpredictable phenomena. The core of the paper is to introduce the control loop model methodology in JADE business process simulation implementation. At the end of this paper the analysis of agent-based simulation outputs through process mining methods and methods for analysis of agents' behavior in order to verify the correctness of used methodology is presented. The business process simulation inputs are randomly generated using the normal distribution. The results obtained show that using random number generation function with normal distribution can lead to the correct output data and therefore can be used to simulate real business processes.

1 Introduction

Simulations used in experiments in the paper could be described as agent-based simulations [13] of Business Process Management (BPM). Business process is an activity adding the value to the company. Usual business process simulation approaches are based on the statistical calculation (e.g. [14]). But only several problems can be identified while using the statistical methods. There are a lot of other influences those are not able to be captured by using any business process model (e.g. the effects of the collaboration of business process participants or their communication, experience level, cultural or social factors). This method has only limited capabilities of visual presentation while running the simulation. Finally, an observer does not actually see the participants of business process dealing with each other.

V. Snasel et al. (Eds.): SOCO Models in Industrial & Environmental Appl., AISC 188, pp. 515–524.
springerlink.com

Agent-based simulations dealing with a company simulation can bring several crucial advantages [4], [11]. They can overcome some of the problems identified above. It is possible to involve unpredictable disturbance of the environment into the simulation with the agents. All of the mentioned issues are the characteristics of a multi-agent system (MAS).

One of the problems the simulations of business processes tackle with is the lack of real business data. Many researchers [9], [20] use randomly generated data instead. On the basis of our previous research, we used the normal distribution in our simulation experiments. We reported on more issues dealing with the business process and financial market simulations [16], [17]. The simulation approach described in this paper uses bellow mentioned control loop model [17], [2] as a core principle. The influence of randomly generated parameters on the simulation outputs while using different kinds of distributions is presented in [18].

The novel methodology and workflow described in this paper are implemented in the form of MAS [19]. JADE [3] development platform was chosen for the realization. JADE provides robust running and simulation environment allowing the distribution of thousands of software agents. Multi-agent system is used as a BPM framework in this paper. When finished, it shall cover the whole company structure from supply of the material, through the production process, up to the selling and shipment. The overall idea of the proposed novel methodology is to simulate real business processes and to provide predictive results concerning the management impact. This should lead to improved and effective business process realization.

To achieve the suitable design of the proposed system a mechanism for verification is needed. As selected input data are generated randomly, or can be set for various types of simulations, the appropriate verification of agents' behavior is required. Process mining methods [1] were used for identification of agents' behavior as well as the methods for identification of behavioral patterns working with similarity of sequences [10], [8]. Used method for behavioral pattern identification was described more detailed e.g. in [15], where behavioral patterns of students in e-learning system were identified.

This paper is structured as follows. Section 2 briefly informs about the control loop model. Multi-agent system implementation and mathematical definition of production function are presented in Section 3. In Section 4 the simulation results are introduced. In Section 5 the process mining methods are used in order to verify the correctness of implemented model. In Section 6 the conclusion and future research steps find their places.

2 Control Loop Methodology

While analytical modeling approaches are based mostly on the mathematical theories [12] the approach followed in this paper is based on experimental simulations. The generic business company used for these simulations based on the control loop paradigm is presented in Figure 1.

The methodology is based on the idea that the company could be presented as a control loop, where the market conditions as well as customers behavior are seen as an external part of the modeled system while the internal company behavior is subject

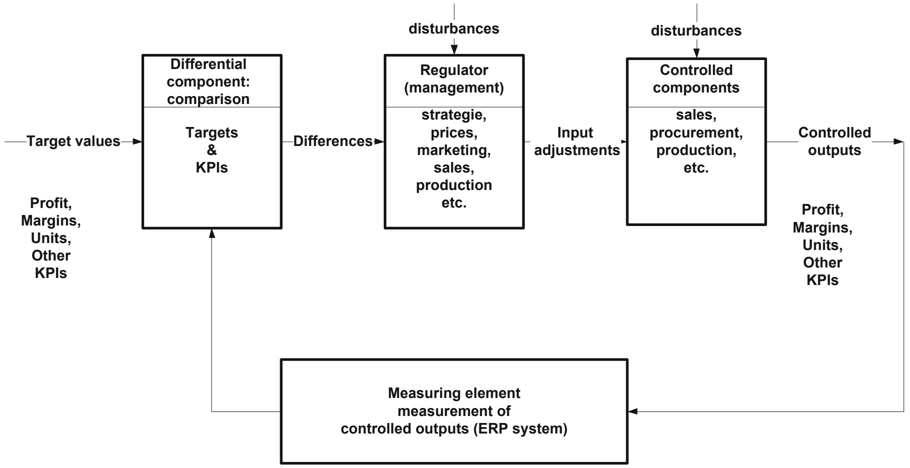

Fig. 1. Generic model of a business company (source: own)

to the simulation. We simulate one of the core business processes of a typical business company. The simulated business process is the selling of goods by sales representants (sales reps in the rest of the paper) to the customers. This process is only one part of the whole control loop. The subject presented in this paper consists of following types of agents: seller agents, customer agents, an informative, and a manager agent. The seller agent interacts with the customer agent according to the multi-agent approach. The interaction is based on the FIPA contract-net protocol [6]. This simplified system was extended by random disturbances influencing the agents' behavior. The number of customer agents is significantly higher than the number of seller agents. The behavior of agents is influenced by two randomly generated parameters using the normal distribution.

The control loop consists of controlled units like sales, purchase, production and others, managed by a regulator unit (the management of the company). The outputs of the controlled units are measured by the measuring unit and compared with the key performance indicators (KPIs). The differences found are sent to the regulator unit, which takes necessary measures in order to keep the system in the closeness with of the KPI values.

The overall workflow of the system proposed can be described as follows. The customer agents randomly generate the requests to buy random pieces of goods. The seller agents react to these requests according to their own internal decision functions and follow the contracting with the customers. The purpose of the manager agent is to manage the requests exchange. The contracting results into the sales events. More indicators of sale success like revenue, amount of sold goods, incomes, and costs are analyzed (more can be found in Section 4).

The motivation of this paper is to present a part of the whole system consisting of seller and customer agent types. In Section 3 the multi-agent system implementation is in detail described.

3 Multi-agent System Implementation

In this section, the implementation steps of the multi-agent system are described in details. The mathematical definition of production function is proposed. Production function is used during the contracting phase of agents' interaction. It serves to set up the limit price of the customer agent as an internal private parameter.

Every simulation step is stored as a time record in the log file. The complete log file is used as a subject for the verification. Only one part of the company's control loop, defined earlier, was implemented. This part consists of sales reps and customers trading with stock items (e.g. tables, chairs). One stock item simplification is used in the implementation. Participants of the contracting business process in our system are represented by the software agents - the seller and customer agents interacting in course of quotation, negotiation and contracting. There is an interaction between them. The behavior of the customer agent is characterized in our case by proposed customer production function defined bellow (Equation 1).

Each period turn (here we assume a week), the customer agent decides, if to buy something. His decision is defined randomly. If the customer agent decides not to buy anything, his turn is over; otherwise he creates a sales request and sends it to his seller agent. The seller agent answers with a proposal message (concrete quote starting with his maximal price – limit price $*1.25$). This quote can be accepted by the customer agent or not. The customer agents evaluate the quotes according to the production function. The production function was proposed to reflect the enterprise market share for the product quoted (market share parameter), sales reps' ability to negotiate, total market volume for the product quoted etc. (in e.g. [17]). If the price quoted is lower than the customer's price obtained as a result of the production function, the quote is accepted. In the opposite case, the customer rejects the quote and a negotiation is started. The seller agent decreases the price to the average of the minimum limit price and current price (in every iteration is getting effectively closer and closer to the minimum limit price), and resends the quote back to the customer. The message exchange repeats until there is an agreement or reserved time passes.

The sales production function for the m-th sales representative pertaining to i-th customer determines the price that i-th customer accepts [18].

$$c_n^m = \frac{\tau_n T_n \gamma \rho_m}{Z M \gamma_n^{mi}}, \tag{1}$$

where:

c_n^m - price of the n-th product quoted by m-th sales representative,
τ_n - company market share for the n-th product,
T_n - market volume for the n-th product in local currency,
γ - competition coefficient lowering the sales success,
ρ_m - quality parameter of the m-th sales representative,
Z - number of customers,
M - number of sales representatives in the company,
γ_n^{mi} - requested number of the n-th product by the i-th customer at m-th sales representative.

Customer agents are organized in groups and each group is being served by concrete seller agent. Their relationship is given; none of them can change the counterpart. Seller agent is responsible to the manager agent. Each turn, manager agent gathers data from all seller agents and makes state of the situation of the company. The data is the result of the simulation and serves to understand company behavior in a time depending on the agents' decisions and behavior. The customer agents need to know some information about the market. This information is given by informative agent. This agent is also responsible for the turn management and represents outside or controllable phenomena from the agents' perspective.

4 Simulation Results

In order to include randomly generated inputs, two important agents' attributes were chosen to be generated by pseudo random generator. Firstly, the seller agent's ability, and secondly the customer agent's decision about the quantity for purchase were used.

For generating random numbers from normal distribution (Gaussian) the Java library called Uncommon Maths written by [5] was used. For the values generation random `MerseneTwisterRNG` class was implemented. The class is a pure Java port of Makoto Matsumoto and Takuji Nishimura's proven and ultra-fast *Mersenne Twister Pseudo Random Number Generator for C*. The parametrization of MAS is listed in Table 1.

Table 1. List of agents' parameters

AGENT TYPE	AGENT COUNT	PARAMETER NAME	PARAMETER VALUE
Customer Agent	500	Maximum Discussion Turns	10
		Mean Quantity	50
		Quantity Standard Deviation	29
Seller Agent	50	Mean Ability	0.5
		Ability Standard Deviation	0.3
		Minimal Price	5
Manager Agent	0	Purchase Price	4
Market Info	1	Item Market Share	0.5
		Item Market Volume	5 000 000

One year of the selling/buying processes (52 turns weeks) was simulated (Figure 2). The output values are mostly closer to the mean than to the extremes. Obtained results demonstrate realistic KPIs of company simulated.

In Table 2 aggregated data can be seen, which reflects the most important characteristic of the normal distribution.

Both attributes the seller ability and also the selected quantity should be taken from the normal distribution for more realistic scope of the generation. The main significant data series from *Incomes* was used and the correlation analysis was made. The correlation coefficient -0.018 doesn't prove tight correlation binding between the results

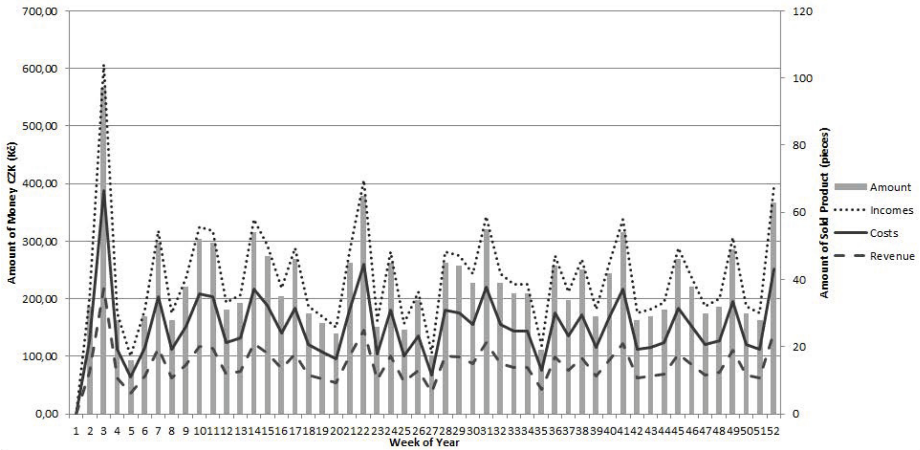

Fig. 2. The generation graph with normal distribution of attributes (source: own)

of simulation and random attributes generation with normal distribution. For the KPIs evaluation the amount, incomes, costs and revenue were counted. The values resulted from the 53 weeks of the company trading. Implemented MAS provides necessary results in the form of KPIs every week during one year of trading. Obtained KPIs could be compared from one simulation experiment to another. This could be used to analyze different simulation parameterizations and the impact on the company performance.

Table 2. Random Distribution Generation Output Results

AGGREGATION	AMOUNT (PIECES)	INCOMES (CZK)	COSTS (CZK)	REVENUE (CZK)
SUM	1 983.00	11 803.48	7 932.00	3 871.48
AVG	38.13	226.99	152.54	74.45
STD. DEV.	16.05	95.76	64.19	32.03

In Section 5 will be used the process mining method in order to verify the proposed model. If the verification confirms correct behavior of agents, the used model works also correct.

5 Verification of Simulation Implementation

The verification of implemented simulation of proposed multi-agent system was performed through the analysis of the agents' behavior. The system records all the actions performed by the agents into the log file. The log file structure corresponds to the following description: the rows of the log file represent events, which can be described by attributes like timestamp, performer (agent), type of action, and additional information (ability, stock items, price and other).

The agents' behavior was analyzed through the methods of process mining. Aalst in [1] defines a methodology for the analysis of business processes. This methodology was used for the extraction of sequences of actions performedx by the agents in the system. The sequences were extracted for each negotiation between the seller and customer agents; the other types of agents were analyzed as well. However, for the verification process, the behavior of the seller and customer agents was more important. More detailed description of sequence extraction and behavioral pattern identification using algorithms for the comparison of categorical sequences is described in our previous work, for example in [15], where behavioral patterns of students in e-learning system were analyzed. For the experiments in this paper, the method T-WLCS was used for finding the behavioral patterns of the agents.

After the sequence extraction phase, we have obtained 2 097 sequences. Using T-WLCS method for comparison of sequence similarity, we have constructed the similarity matrix, which can be represented using tools of graph theory. The finding of behavioral patterns of agents was performed through a weighted graph $G(V,E)$, where vertices V were the sequences and edges E represented the relations between the sequences on the basis of their similarity. The weight of edges w was defined by the sequence similarity (T-WLCS method). Detailed information of the original sequence graph is presented in Table 3.

Table 3. Description of Sequence Graph

INFORMATION	VALUE
Nodes	1975
Isolated Nodes	2
Edges	942 068
Connected Components	7

The sequence graph consisted of large amount of similar sequences. Moreover, it was dense and very large for further processing. Better interpretation of results was possible by finding the components, which can represent the behavioral patterns of agents. That was the reason, why we have used spectral clustering by Fiedler vector and algebraic connectivity [7]. The original sequence graph was divided into 7 components with similar sequences. Description of obtained components, their size and type of agents is presented in Table 4.

Visualization of obtained components using graph theory can be seen in Figure 3. Each component, determined by its color, consists of similar sequences and represents similar behavior. Each sequence is described by the information of its performer (agent). Therefore, we are able to identify groups of agents with similar behavior to verify the model.

The verification process of the implementation was based on finding behavioral patterns and groups of agents with similar behavior. Proposed method can be used successfully for better interpretation and description of real model working and of real behavior of the agents in the model. There are several input parameters, which can influence real behavior of the agents in the system. Proposed method can facilitate the analysis of

Table 4. Component Description

COMPONENT	SIZE	TYPE OF AGENT
C0	1 034	CustomerAgents
C1	498	CustomerAgents
C2	253	CustomerAgents
C3	136	SellerAgents
C4	51	SellerAgents
C5	1	ManagerAgent
C6	1	DisturbanceAgent

real behavior of the agents during the system simulation. As an example we can mention ability, which was set up randomly for the seller agents, where the values followed the normal distribution function. The ability is the parameter, which influents the seller agents' behavior. The higher the value is, the more skillful the agent is in the negotiation process. In the other words, the agent is able to finish the negotiation process in shorter time. This corresponds to shorter sequences extracted from the log file for this agent.

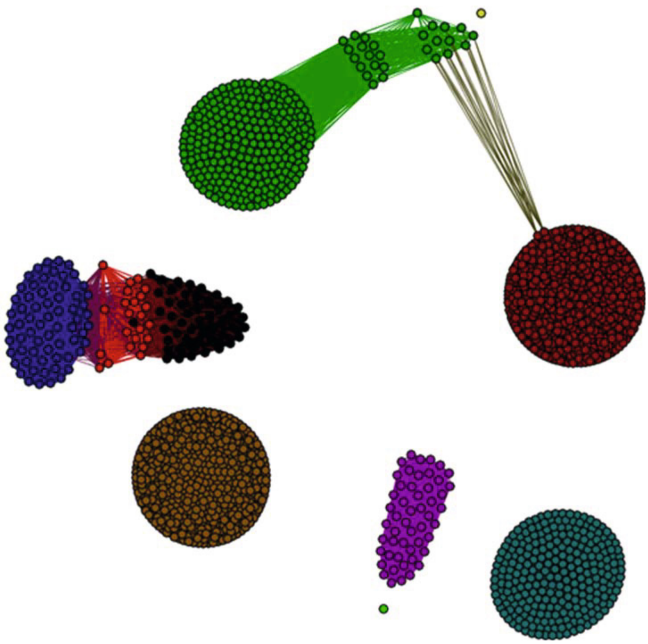

Fig. 3. Components with Similar Sequences

6 Conclusion

The BPM simulation experiment in the form of MAS was introduced in this paper. Proposed simulation model was implemented in order to simulate business process participants in virtual company. Overall methodology is based on the company's control loop. The simulation provides useful information about core business processes. Process mining methods used for the model verification confirmed the model correctness.

The verification process of the implementation was based on finding behavioral patterns and groups of agents with similar behavior. Proposed method can be used successfully for better interpretation and description of real model working and of real behavior of the agents in the model.

The next steps of our research will follow the rest parts of the whole control loop. The purchase side and the disturbance module will be implemented in order to analyze the performance of virtual company. The using of implemented MAS as a decision support tool for the management of company will be the leading idea in the future research.

Acknowledgment. This work was supported by the Bio-Inspired Methods: research, development and knowledge transfer project, reg. no. CZ.1.07/2.3.00/20.0073 funded by Operational Programme Education for Competitiveness, co-financed by ESF and state budget of the Czech Republic and by SGS, VSB – Technical University of Ostrava, Czech Republic, under the grant No. SP2012/151 Large graph analysis and processing.

References

1. van der Aalst, W.M.P.: Process Mining. Discovery, Conformance and Enhancement of Business Processes, p. 352. Springer, Heidelberg (2011) ISBN: 978-3-642-19344-6
2. Barnett, M.: Modeling & Simulation in Business Process Management. Gensym Corporation, pp. 6–7 (2003), `http://bptrends.com/publicationfiles/11-03%20WP%20Mod%20Simulation%20of%20BPM%20-%20Barnett-1.pdf` (accessed January 16, 2012)
3. Bellifemine, F., Caire, G., Trucco, T.: Jade Programmer's Guide. Java Agent Development Framework (2010), `http://jade.tilab.com/doc/programmersguide.pdf` (accessed January 16, 2012)
4. De Snoo, D.: Modelling planning processes with TALMOD. Master's thesis, University of Groningen (2005)
5. Dyer, D.W.: Uncommons Maths - Random number generators, probability distributions, combinatorics and statistics for Java (2010), `http://maths.uncommons.org` (accessed January 16, 2012)
6. Foundation for Intelligent Physical Agents (FIPA 2002) FIPA Contract Net Interaction Protocol. In Specification, FIPA, `http://www.fipa.org/specs/fipa00029/SC00029H.pdf`
7. Fiedler, M.: A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. Czechoslovak Mathematical Journal 25, 619–633 (1975)
8. Guo, A., Siegelmann, H.: Time-Warped Longest Common Subsequence Algorithm for Music Retrieval, pp. 258–261 (2004)

9. Hillston, J.: Random Variables and Simulation (2003), http://www.inf.ed.ac.uk/teaching/courses/ms/notes/note13.pdf (accessed January 16, 2012)
10. Hirschberg, D.S.: Algorithms for the longest common subsequence problem. J. ACM 24, 664–675 (1977)
11. Jennings, N.R., Faratin, P., Norman, T.J., O'Brien, P., Odgers, B.: Autonomous agents for business process management. Int. Journal of Applied Artificial Intelligence 14, 145–189 (2000)
12. Liu, Y., Trivedi, K.S.: Survivability Quantification: The Analytical Modeling Approach. Department of Electrical and Computer Engineering, Duke University, Durham, NC, U.S.A. (2011), http://people.ee.duke.edu/~kst/surv/IoJP.pdf (accessed January 16, 2012)
13. Macal, C.M., North, J.N.: Tutorial on Agent-based Modeling and Simulation. In: Proceedings: 2005 Winter Simulation Conference (2005)
14. Scheer, A.-W., Nüttgens, M.: ARIS Architecture and Reference Models for Business Process Management. In: van der Aalst, W.M.P., Desel, J., Oberweis, A. (eds.) Business Process Management. LNCS, vol. 1806, pp. 376–389. Springer, Heidelberg (2000)
15. Slaninova, K., Kocyan, T., Martinovic, J., Drazdilova, P., Snasel, V.: Dynamic Time Warping in Analysis of Student Behavioral Patterns. In: Proceedings of DATESO 2012, pp. 49–59 (2012)
16. Spisak, M., Sperka, R.: Financial Market Simulation Based on Intelligent Agents - Case Study. Journal of Applied Economic Sciences VI(3(17)), 249–256 (2011) Print-ISSN 1843-6110
17. Vymetal, D., Sperka, R.: Agent-based Simulation in Decision Support Systems. In: Distance Learning, Simulation and Communication 2011. Proceedings, pp. 978–980 (2011) ISBN 978-80-7231-695-3
18. Vymetal, D., Spisak, M., Sperka, R.: An Influence of Random Number Generation Function to Multiagent Systems. In: Proceedings: Agent and Multi-Agent Systems: Technology and Applications 2012, Dubrovnik, Croatia (2012)
19. Wooldridge, M.: Multi Agent Systems: An Introduction to, 2nd edn. John Wiley & Sons Ltd., Chichester (2009)
20. Yuan, J., Madsen, O.S.: On the choice of random wave simulation in the surf zone processes. Coastal Engineering (2010), http://censam.mit.edu/publications/ole3.pdf