



HAL
open science

Simulation Experiment in a Virtual Laboratory Environment as a Ground for Production Competencies Acquiring

Natalia Bakhtadze, Oleg Zaikin, Andrzej Żylawski

► **To cite this version:**

Natalia Bakhtadze, Oleg Zaikin, Andrzej Żylawski. Simulation Experiment in a Virtual Laboratory Environment as a Ground for Production Competencies Acquiring. IFIP International Conference on Advances in Production Management Systems (APMS), Sep 2021, Nantes, France. pp.535-545, 10.1007/978-3-030-85874-2_57. hal-04030396

HAL Id: hal-04030396

<https://inria.hal.science/hal-04030396v1>

Submitted on 15 Mar 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

Simulation Experiment in a Virtual Laboratory Environment as a Ground for Production Competencies Acquiring

Natalia Bakhtadze^{1[0000-0001-9319-6951]}, Oleg Zaikin² and Andrzej Żyławski³

¹ V.A. Trapeznikov Institute of Control Sciences RAS, Moscow, Russia

^{2,3} Warsaw School of Computer Science, 00-169, Warsaw, Poland
sung7@yandex.ru ozaikin@poczta.wysi.edu.pl

Abstract. The paper describes advantages of teaching and application of modelling manufacturing systems. Two paradigms of modelling: Queuing systems and Simulation are briefly presented and analysed. Advantages of distance learning these approaches worldwide are presented. Furthermore, a combined way of learning these two methods, with a focus on the modelling and simulating selected basic processes of manufacturing systems, is proposed in briefly described case study. The concept provides division of this method depending on student's education level.

Keywords: Manufacturing Competencies, Queuing Systems and Simulation, Virtual Laboratory.

1 Introduction

Simulation is a technique for practice and learning that can be applied to many different disciplines and trainees. It is a technique to replace and amplify real processes or experiences with controlled ones, that replicate substantial aspects of the real world in a fully interactive fashion. Simulation-based learning can be the way to develop professionals' knowledge, skills, and attitudes, whilst protecting students from labour-intensive costs and unnecessary risks. Simulation-based education can be a platform which provides a valuable tool in learning to theoretical bases and resolve practical tasks in manufacturing systems [1].

Simulation-based training techniques, tools, and strategies can be applied in designing structured learning experiences, as well as be used as a measurement tool related to targeted teamwork competencies and learning objectives. In manufacturing, simulation offers good scope for training of interdisciplinary project teams. The realistic scenarios and equipment allows for retraining and practice till one can master the necessary knowledge or skill. Teamwork training conducted in the simulated environment may offer an additive benefit to the traditional methods of learning, enhance performance, and also help reduce errors in designing and decision - making in manufacturing.

A simulation experiment can prove to be a useful ground for providing the necessary skills in a very broad sense: from problem (task) statement, through problem analysis, mathematical and simulation modelling to conducting a research experiment and interpreting the results. The simulation modelling is a multi-stage and interactive process. Dividing it into stages is a consequence of qualities the simulation process contains a research method. Each of the stages requires identification of the type and the contents of applicable knowledge and that fact becomes the foundation for development of a problem-specific tasks sequence. It is therefore reasonable to claim that precise measuring of competencies acquisition process is equally effective as it is in the traditional learning environment.

Using the principles presented herein, it is possible to establish an instance of a laboratory of simulation. The laboratory is based on the use of a typical model repository classified according to Kendall's notation [2]. That problem has already been analysed in the context of a procedure for personal competencies acquisition. The content and order of problem simulation-oriented tasks is a result of a widely accepted routine dealing with preparation and formulation of stages in the simulation experiment [3], [4] and [5]. Let us examine the particular stages of conducting the simulation procedure in the perspective of the tasks provided to the learning individual .

2 Methodology for Developing the Simulation Model

2.1 Statement of simulation experiment

Development of the simulation model requires initial problem conceptualization and specification of the given problem using terms of a relevant domain. Problem conceptualization may usually be represented by inconsistent structure and is often implemented using unsophisticated description formed using natural language. The inherent diversity of concepts in that case and limited precision of specification of the problem statement create major problems at the stage of modelling. The most complex issue is maintaining accuracy in the process of transformation of concepts of the analysed domain to terms and functions applied in a given simulation tool, i.e. to a certain portion of procedural knowledge [6].

Simulation software usually includes libraries of predefined templates for modelling different problem domains. Unfortunately, in most cases they are fairly limited in scope and applicability and their are designed to respond to select, or most typical problems. On the other hand, they effectively handle entire classes of problems within a scope of given mathematical topic, i.e. fundamental (theoretical) knowledge that is represented by well defined structure and concept taxonomy. In such case it is possible to identify and formulate an official analytical model. Moreover, a mathematical apparatus and formal notation are available for precise analytical task formulation. We can refer to the example of Kendall's notation serving the queuing system theory [2].

Therefore, the stage of problem statement using the concepts of universally known mathematical engine and its relevant notation can be seen as less complex in performing transformation of the domain concepts to formulas used in the given simulation software than in the example of direct transformation to the terms of a chosen domain characterized by limited precision of its concepts. That is why the previously mentioned stages of simulation

experiment statement conducted in the environment of virtual laboratory should be complemented by several iterations of mapping of the verbal problem representation onto the simulation model [7] (Fig..1). In that context, documents such as the 2000 Mathematics Subject Classification in which mathematical taxonomy is introduced play a pivotal role in facilitating the process of identification of adequate mathematical engine upon formalizing the problem domain.

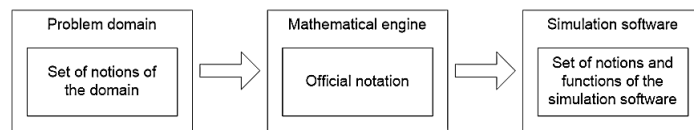


Fig.1. The stages of mapping the conceptual model in the process of simulation experiment statement (source: [8])

As it has been shown on Fig.1, the simulation experiment statement is exemplified by use of the concepts that are specific to the given simulation software. The concepts form a language for defining the problem and requirements essential for conducting the simulation experiment. The preparation of the concepts is not possible without completing the stage of forming the assumptions about the specific problem using the concepts of the domain that the analysed problem is pertinent to. The simulation experiment statement performed in such approach is difficult to be translated to the language (terms and formulas) of the simulation software. Hence, additional modelling process is performed using a mathematical apparatus that is adequate to the given problem. Using that measure significantly simplifies the simulation process of the mathematical model using the given simulation software.

It is vital for the process of formulation of initial assumptions about the researched problem to identify a criterium for performing analysis of the model. Once the criterium has been determined, its examination is required in order to identify its characteristics (continuous, or discrete), or to find other significant parameters. The problem containing properly defined criterium can be subject to modelling using the procedure of mapping the concepts of fundamental knowledge onto procedural knowledge. The mapping can be carried out through identification of direct equivalents between distinct conceptual models [7].

2.2 Algorithm for determining simulation experiment settings

The algorithm for determining the simulation experiment settings will be discussed using an example of a research problem defined using queuing systems theory. Using a queuing systems theory, an elementary event occurring at the input to the system is to be determined. Naturally, the event is considered the most significant component of the model. In addition, all the parameters involved in the event input stream are to be determined, their average intensity (i.e. a number of events on a given time frame and their characteristic: continuous, or discrete). If the input event stream is of stochastic nature, a probability distribution function of time intervals between the incoming events should be determined. In a queuing theory system, a numerous types of events can appear, and for that reason the process of the

system identification and determination of the operational parameters should be performed for each of the type respectively.

The process of defining the elementary events is key importance to the queuing system model. Thus, it should be repeated some number of times in order to thoroughly investigate the given problem. Once the research activities have been completed resulting in the definition being sound and acceptable, it then possible to identify the rational laying foundations for the simulation experiment. The assumptions for the experiment are determined using the ARENA simulation software. The previously deliberated mathematical-to-simulation model transformation process is conducted according to the principles of concept mapping [8]. To conclude, the algorithm for determining rationale of the simulation experiment for queuing system theory using the ARENA software is contained in the following steps:

1. Given a problem domain, perform a task statement.
2. Determine an elementary and determine parameters of the event flow.
3. Refine the event definition through recurrent operation with steps (1) and (2). The process is terminated when a researcher is convinced that the problem can be solved using queuing systems theory.
4. Identify a structure of the queuing systems-based system.
5. Determine an initial outline (specification) of the experiment.
6. Create a corresponding simulation model using the ARENA software [9].

3 Queuing Modelling of the Virtual Laboratory Management System

3.1 Structure of the virtual laboratory management system

Let us consider the process of management in virtual laboratory as a typical closed queuing system containing several special types of servers and one class of learners, hereafter referred to as students [8]. Virtual laboratory as a queuing system can be described very simply because we can use existing analytical models. For example, we can consider the problem of optimization of queuing system parameters. To solve this task, it is important to learn some of the characteristics typical of queuing systems, such as:

- average number of students in the class,
- average waiting time in the queue,
- average service time,
- average time thinking for each type of order or the use of servers.

Let us consider a situation typical for virtual laboratory (VL) - a finite population of users, a local network (e.g. Intranet) and no restrictions on the length of the waiting queue. In our deliberations, we will focus on closed queuing system. The structure of such a system in the context of VL is presented in the Fig.2. In the case of a closed system, we will divide the simulation process into six main components:

- 1) Server (class) of students.
- 2) Learning server - a repository of didactic materials.
- 3) Teacher server.

- 4) Simulation server - a repository of simulation models.
- 5) Tutor's server.
- 6) Administrator server - SQL registration and identification database.

The student class generates orders with random time intervals. The order process is carried out in the following order:

- 1) The first student's order is sent to the administrator's server for registration and identification. Each student has the same characteristics of arrival and service.
- 2) The student generates orders to the teacher's server to solve the input test and receive a theoretical task. The order to the teacher's server can be sent several times.
- 3) Student in order to complete the theoretical task generates orders for:
 - learning server with access to a repository of didactic materials,
 - teacher's server (for consultation),
 - administrator's server (in case of failure).
- 4) After the entry test and sending the answer to the theoretical question, the student asks the tutor's server to ask for a practical task, in this case the task with the simulation model.
- 5) The student generates orders for:
 - a model repository to conduct a simulation experiment,
 - tutor server (for consultation) once again to
 - administration server
- 6) The student leaves the system.

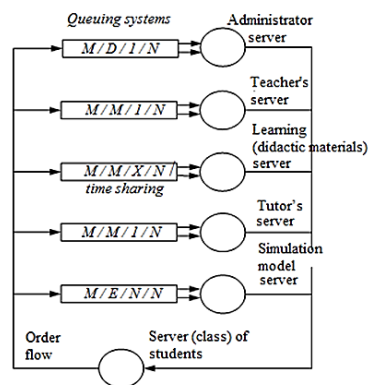


Fig. 2. The structure of queuing systems in simulation experiment process (own study based on [10])

Statement of the Task. With a large number of students and their independence, the flow of orders generated in the student class and the flow of incoming orders to each

type of server can be considered as stationary. The process of generating orders of student class can be described by the Poisson distribution. The service time in the teacher's server is assumed as exponential.

Consider the situation when students need to find didactic materials in the learning server or the relevant exam dates in the administration server. Because in our model the teacher is a consultant, student orders in the learning server are handled in consultation mode.

There are several types of servers and corresponding queuing systems in the open distance learning environment. Each server has different assignment and productivity. We use the *Kendall notation* [2] to describe them.

$A/B/X/Y/Z$, where:

A - distribution of the order arrival process,

B - service time distribution in the server,

X - number of parallel servers,

Y - system capacity,

Z – queue service discipline.

In the queuing system, student works are treated as user orders. Consider a finite population of customers. In general, different students need different time to think. This is related to the cognitive characteristics of the student. In addition, it is important to ensure that the material can be selected and delivered on time. We simplify the real situation and assume that every student has the same cognitive model. In the case of the open distance learning system, we distinguish the following queuing systems.

1. *Administrator server* $M/D/1/N$ - orders from students arrive according to the Poisson process, the time of service is deterministic (permanent). The number of servers is equal to 1. System capacity - equal to the number of students (N). Servicing discipline – *FIFO*, i.e. first came - first served.

2. *Teacher's server* $M/M/1/N$ - orders from students arrive in accordance with the Poisson process, the service time distribution is exponential. The system contains only one teacher. The capacity of the system is equal to the number of students (N). Servicing discipline – *FIFO*, the first came - first served.

3. *Learning server (didactic materials)* $M/M/X/N/time\ sharing$ - orders from students arrive in accordance with the Poisson process, the service time distribution is exponential. The number of servers is equal to X , where the size of X is determined by economic factors. The capacity of the system is equal to the number of students (N). Server operation time is evenly distributed among all orders in the system (*time sharing*).

4. *Tutor's server* $M/M/1/N$ - orders from students arrive according to the Poisson process, the service time distribution is exponential. The number of servers is equal to 1. The capacity of the system is equal to the number of students (N). Servicing discipline – *FIFO*, first came - first served.

5. *Server of simulation models* $M/E/N/N$ - orders from students arrive in accordance with the Poisson process, the service time distribution is Erlang's function. The number of servers is equal to 1. The capacity of the system is equal to the number of students (N). Servicing discipline - *FIFO*, first came- first served. The Erlang time of

service is due to the fact that the process of servicing each task is multi-phase, where k is the number of phases of the simulation experiment. The time of servicing each phase of the experiment can be described by exponential function

6. *Student class* $M / M / N / N$ - generates orders completely randomly. The time interval between orders is random, both the number of students and the capacity of the system is limited by the number N .

To build a model of VL system, it is necessary to determine the average time of staying of orders in queue, average server usage and average time spent in the whole system. In general, the queuing system q_s can be defined by the following parameters: the rate of the arrival of orders λ_s , the intensity of servicing μ_s and average time of servicing $\tilde{\tau}_s = 1/\mu_s$, the number of parallel servers N_s , the size of the input buffer b_s and the use of the server ρ_s . It is necessary to specify the number of servers in each queuing system, keeping in mind that each type of server has different productivity and downtime costs. Below the formal model for allocating resources in a closed queuing system is presented.

Input data. N is number of students in the classroom, $\nu_i, i = 1, 2, \dots, I$ are rate of arrival of student's orders to the server (from one student), $\tilde{\tau}_i^s, i = 1, 2, \dots, I$ are average servicing time for server $s_i, \mu_i = 1/\tilde{\tau}_i^s$ are average rate of servicing for server s_i .

Control parameters. $X_i, i = 1, 2, \dots, I$ are numbers of machines working in parallel for each of the servers.

Criteria function. The criterion function contains three components:

1. Total processing time of all customer orders arriving per unit of time:

$$T_{\Sigma} = \sum_i N \nu_i \tau_i^s = \sum_i \lambda_i \tilde{\tau}_i^s(X_i),$$

where: $\lambda_i = \nu_i N$ - total rate of arrival of client orders to the server $s_i, \tilde{\tau}_i = \tilde{\tau}_i^w + \tilde{\tau}_i^s$ is average time of order stay in the server $s_i, \tilde{\tau}_i^w$ - average waiting time of the order in the server queue $s_i, \tilde{\tau}_i^w = F(X_i), \tilde{\tau}_i^s$ - average time of servicing the order in the server s_i .

2. Total server cost (hardware and software) $C_{\Sigma} = \sum_i X_i c_i$, where c_i are the costs of one type server s_i ,

3. Total costs of work and server downtime (personnel, hardware and software) are

$$U_{\Sigma} = \sum_i [(1 - \frac{\lambda_i}{\mu_i X_i}) \delta_i + \frac{\lambda_i}{\mu_i X_i} \gamma_i],$$

where δ_i, γ_i , are expenses caused by work or downtime of server s_i per unit of response time.

Then **the task of modelling the work flow** can be formulated as follows.

For a given set of server types, $s_i, i = 1, 2, \dots, I$ included in open distance learning system and parameters of each type server ($N, \nu_i, \tilde{\tau}_i^s, \mu_i$) it is need to determine number of parallel servers of each type $X_i, i=1, \dots, I$, guaranteeing a minimum total time needed to complete all orders received:

$$T_{\Sigma} = \sum_i N \nu_i (\tilde{\tau}_i^w + \tilde{\tau}_i^s) = \min, (X_i, i = 1, \dots, I)$$

With the restriction: for total costs of educational resources (personal, equipment and software):

$$\alpha C_{\Sigma} + U_{\Sigma} = \sum_t \left[X_i c_i + \left(1 - \frac{\lambda_i}{\mu_i X_i}\right) \delta_i + \frac{\lambda_i}{\mu_i X_i} \gamma_i \right] \leq C_0$$

3.2 Method of Solution

The task formulated in this way is the task of integer programming with a non-linear objective function with non-linear constraints. In the general case, with any distribution of orders' flow in the distance learning system and any distribution of servicing time, this task can not be solved only on the basis of analytical methods. It is necessary to perform an experiment on a simulation model. Suppose the exam in distance learning system described above consists of a fixed number of questions for each student in the class. The total time of the exam may be expressed by the following formula:

$$T_{exam} = qN(\tilde{T}_c + \tilde{T}_R) + q \sum_{k=0}^{N_0} \tilde{T}_A(N),$$

where: q is the number of questions for each student, $\tilde{T}_c + \tilde{T}_R$ is average time of thinking and response, \tilde{T}_A is average total time spent by the student in the system, N_0 is the initial number of students in the class. The average total time \tilde{T}_A can be expressed as the weighted sum of time spent by the student in each type of server:

$$\tilde{T}_A = \sum_i v_i \tilde{\tau}_i = \sum_i v_i (\tilde{\tau}_i^w + \tilde{\tau}_i^s).$$

Define p_i as the probability of referring to the server s_i . From the condition of normalization $\sum_i p_i = 1$ it follows that

$$p_i = \frac{v_i}{\sum_i v_i}$$

In the general case, the rate of the arrival of orders in a closed queuing system can be determined by means of a set of linear equations (*traffic equation*)

$$\lambda_i = \sum_{j=1}^M \lambda_j p_{ij} = \lambda_c p_i, i = 1, \dots, I, \quad \lambda_c = \sum_{j=1}^M \lambda_j,$$

where: M is number of servers, $P = \|p_{ij}\|$ is transition matrix. The average number of orders waiting in the queue of server s_i can be determined by using:

$$E[k_i] = \sum_{k=0}^{k_0} k P[k_i = k];$$

$$P[k_i = k] = \sum_{\tilde{k} \in S, k_{\tilde{k}} = k} P(\tilde{k}).$$

In [11] it was shown that for this type of closed queuing system the following form of the product, shown below, is true:

$$P(\tilde{k}) = P(\tilde{0}) \prod_i \left(\frac{\lambda_i}{\mu_i}\right)^{k_i},$$

$$P(\tilde{0}) = \left\{ \sum_{k \in S} \prod_i \left(\frac{\lambda_i}{\mu_i}\right)^{k_i} \right\}^{-1},$$

where: $P[k_i = k]$ is probability of waiting for k_i orders in the queue, $P(\tilde{k})$ is probability of state \tilde{k} , $P(\tilde{0})$ is the probability of a "zero" state.

4 Simulation model of teacher and student collaboration

This process is related to behaviours and preferences of humans, which can change and are difficult to predict. For this reason simulation seems like an adequate mechanism, which allows obtaining information about the analysed object of study and is the basis for further activities related to the functioning of this object.

On the basis of the developed simulation model it is possible to study the adapted *strategy of cooperation between the teacher and a group of students* (Fig. 3). The aim of the simulation is to assess if the expected repository development plan is possible to realise with the teacher working in a distance mode, with a specified distribution of student arrival, and a specified distribution of grading.

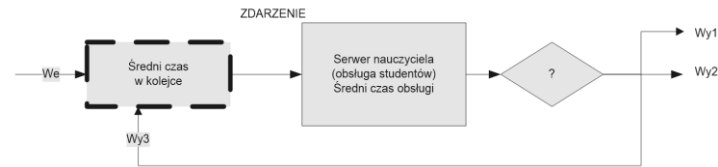


Fig.3. General form of the simulation model. Source: own study

5 Conclusion

1. Simulation is a tool that the teacher directly uses. The teacher, thanks to the results obtained from experiments conducted on a simulation model, can change the strategy of working with students and modify their motivation function, e.g. by changing the difficulty of the tasks underlying the acquisition of competencies. The simulation model allows to evaluate the different options of the teacher's work, which must take into account the constraints characteristic of the educational situation (time, type of student group, student involvement, etc.).

2. For the purposes of collecting statistics and conducting simulation experiments, the process of teacher-student collaboration can be interpreted as a queuing system in which:

- with specific content and portions of teaching materials, it can be assumed that the teacher's job is to check the task;

- with a specific course, time and group, the teaching job can be treated as a server with a specific input, output, average rating time;
- the average assessment time is based on the teacher's experience (specificity of each course and subject, difficulty of tasks, type of student group, time of completion of classes) and is further supported by the mechanism of extracting motivation from the linguistic knowledge base;
- the flow of students is stochastic (in the selected time period, there is no certainty of how many students will arrive, no knowledge of how many will pass the task, no knowledge of whether a new student will arrive, etc.);

References

1. Buzacott, J., Shanthikumar, J.: *Modelling and analysis of manufacturing systems*. Wiley & Sons, New York (1993).
2. Kleinrock, L.: *Queuing Systems*. John Wiley, New York (1975).
3. Banks, J., Carson, J., Nelson, B., Nivola D.: *Discrete-event system simulation*. Prentice Hall, New York (2001).
4. Chung, C.: *Simulation modelling handbook-A practical approach*. CRS Press, Boca Raton (2004).
5. Robinson, S.: *Simulation: The practice of model development and use*. John Wiley@Sons, Chichester (2004).
6. Kushtina, E., Dolgui A., Malachowski, B.: Organisation of the modelling and simulation of the discrete processes. In: Saeed K., Pejas J., (eds), *Information Processing and Security system*, pp. 443-452. Springer, Heidelberg (2005).
7. Kushtina, E., Rózewski, P.: Analiza systemowe idei otwartego nauczania zdalnego, in: Straszaka A., Owsinińskiego J., (Red.), *Badania operacyjne i systemowe 2004: Na drodze do społeczeństwa wiedzy*, pp. 231-245. Akademicka Oficyna Wydawnicza EXIT, Warsaw, (2004).
8. Zaikin, O., Korytkowski, P., Kushtina, E., Malachowski, B.: Modelling of the Supply Chain for a Distributed Publishing Enterprise, A. Dolgui et al. (eds.), *Supply Chain Optimisation*, pp. 101-116, Springer, Berlin, (2005).
9. Kelton, W., Sadowski, R., Sadowski, D.: *Simulation with Arena*. McGraw-Hill, New York (1997).
10. Zaikin, O. and Dolgui, A.: Simulation model for optimization of resources allocation in the queuing networks In: *Proceed. of IMACS Symposium on Mathematical Modelling 3rd MATHMOD, Argesim Report #15, Vienna*. (2000).
11. Gordon W., Newell G.: *Closed Queuing Systems with Exponential Servers*. *Operation Research* 15, 254-265 (1967).