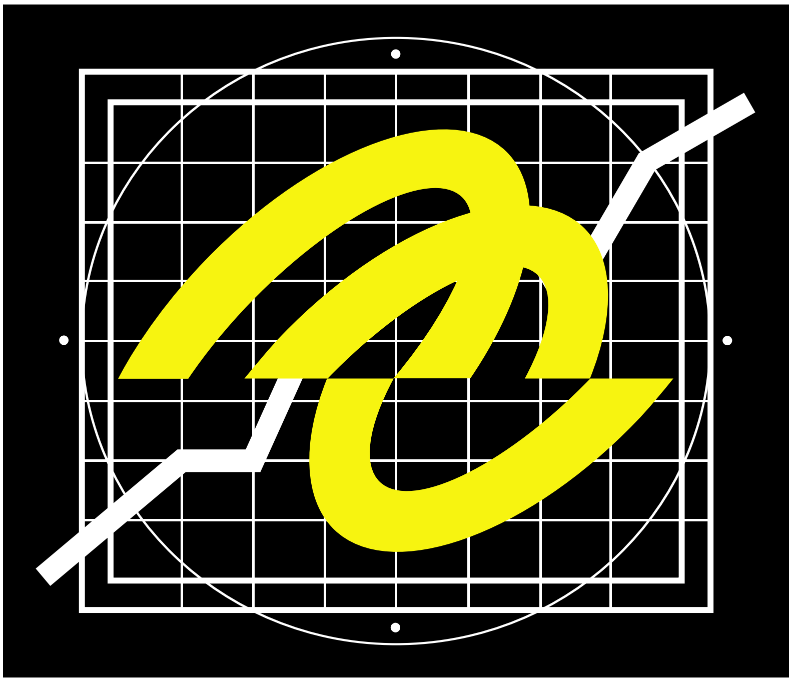
Organizacija



Letnik 39, številka 10, december 2006, ISSN 1318-5454

Poštnina plačana pri pošti 4101 Kranj

Simulation Based Decision Support

Guest Editor: Miroljub Kljajić

REVIJA ZA MANAGEMENT, INFORMATIKO IN KADRE
Journal of Management, Informatics and Human Resources

Založba Moderna organizacija

Organizacija

Organizacija je interdisciplinarna strokovna revija ki objavlja prispevke s področja organizacije, informatike in kadrovskega managementa. Primeri tematskih sklopov, ki jih pokriva revija, so:

- *teoretične osnove organizacijskega razvoja ter spreminjanja organizacijskih struktur in procesov*
- *novi organizacijski pristopi ter njihova uporaba*
- *organizacijski ukrepi za doseganje večje produktivnosti, ekonomičnosti in rentabilnosti poslovanja in proizvodnje*
- *management kakovosti*
- *kadrovanje in izobraževanje kadrov pri prestrukturiranju podjetij*
- *stimulativnost nagrajevanja v spremenjenih lastninskih razmerah*
- *prestrukturiranje organizacijskih in informacijskih sistemov*
- *načrtovanje, razvoj in uporaba informacijske tehnologije in informacijskih sistemov*
- *medorganizacijski sistemi, elektronsko poslovanje*
- *odločanje, podpora odločanju, direktorski informacijski sistemi*

Vsebina ni omejena na navedene tematske sklope. Še posebej želimo objavljati prispevke, ki obravnavajo nove in aktualne teme in dosežke razvoja na predmetnem področju revije, ter njihovo uvajanje in uporabo v organizacijski praksi.

Organizacija, letnik 39, številka 10, december 2006

Kazalo/Content
10/2006

POVZETKI	622		
UVODNIK	624		
RECENZENTI	625		
RAZPRAVE	626	HOSSEIN ARSHAM MIROLJUB KLJAJIĆ	Simulation System Design
	635	MARKO ŽIBERT MIROLJUB KLJAJIĆ	Modelling Parallel Service Systems in GPSS
	652	ROBERT RAUCH MIROLJUB KLJAJIĆ	Discrete Event Passenger Flow Simulation Model for an Airport Terminal Capacity Analysis
	661	DAVORIN KOFJAČ MIROLJUB KLJAJIĆ	Simulation Approach to Warehouse Cost Minimization in Stochastic Environment
	669	ANDREJ ŠKRABA DAVORIN KOFJAČ ČRTOMIR ROZMAN	The Periodicity of the Anticipative Discrete Demand-Supply Model
	680	UROŠ RAJKOVIČ OLGA ŠUŠTERŠIČ JOŽE ZUPANČIČ	A Model of E-documentation of Community Nursing
DONATORJI	688		
IMENSKO KAZALO LETNIK (39) 2006	690		

Hossein Arsham, Miroljub Kljajić

Simulation System Design

The use of simulation as a tool to design complex stochastic systems is often inhibited by cost. Extensive computer processing is needed to find a design parameter value given a desired target for the performance measure of a given system. The designer simulates the process numerically and obtains an approximation for that same output. The goal is to match the numerical and experimental results as closely as possible by varying the values of input parameters in the numerical simulation.

The most obvious difficulty in solving the design problem is that one cannot simply calculate a straightforward solution and be done. Since the output has to be matched by varying the input, an iterative method of solution is implied. This paper proposes a “stochastic approximation” algorithm to estimate the necessary controllable input parameters within a desired accuracy given a target value for the performance function. The proposed solution algorithm is based on Newton’s methods using a single-run simulation approach to estimate the needed derivative. The proposed approach may be viewed as an optimization scheme, where a loss function must be minimized. The solution algorithm properties and the validity of the estimates are examined by applying it to some reliability and queueing systems with known analytical solutions.

Key words: system design and simulation, local response surface, goal seeking problem, parameter setting design, gradient estimation, discrete-event systems.

Marko Žibert, Miroljub Kljajić

Modelling Parallel Service Systems in GPSS

This article treats parallel service system modelling in the GPSS simulation language. The transactions entering such systems select between numerous different servers and we can mostly detect two rules in the selecting of the appropriate server. The first rule always gives the first few (regarding its position in the system) entities (either servers or queues) precedence over the others, while the second rule always treats all the equal entities evenly and selects among them quite randomly. Since GPSS normally operates by the first rule, we frequently come up against difficulties when modelling systems that serve by another rule. The present article offers a methodology how to solve this problem within GPSS.

Key words: discrete simulation, modelling, GPSS, parallel service systems, queuing theory

Robert Rauch, Miroljub Kljajić

Discrete Event Passenger Flow Simulation Model for an Airport Terminal Capacity Analysis

This paper describes analysis of departure passenger flow in an airport terminal, from the passenger entrance to boarding; involving the development of simulation model. The basis for the simulation model was a snapshot of the passenger flow data taken during the summer periods from 2003 to 2004. Various data were collected and used to define the inputs to a simulation model. The next step was a statistical analysis of the collected data. A discrete-event simulation model using simulation programming language General Purpose Simulation System (GPSS) was constructed. The performance of the system in the present and expected future was studied. The simulation model helped us to evaluate the passenger flow, identify the system bottlenecks as well as the system capacities. Critical aspects in the passenger flow through the airport terminal have been explored and studied. Recommendations based on the results of models runs were made.

Key words: modeling, discrete event simulation, airport, passenger flow, GPSS

Davorin Kofjač, Miroljub Kljajić

Simulation Approach to Warehouse Cost Minimization in Stochastic Environment

The objective of inventory management is to balance conflicting goals like keeping stock levels down to have cash available for other purposes and having high stock levels for the continuity of the production. Simulation approach is used to minimize total warehousing cost while no stock-outs occur and warehouse capacity is not exceeded. A case study of replenishment process optimization is presented on several representative materials of an automotive company using two replenishment algorithms: fix review period and full capacity ordering. The presented simulation results indicate considerable cost reduction without violating the mentioned constraints. The fuzzy logic evaluator, a decision support system used for simulation results assessment, is presented and discussed.

Key words: inventory control, simulation, optimization, stochastic models, fuzzy sets, decision support system

Andrej Škraba, Davorin Kofjač, Črtomir Rozman

The Periodicity of the Anticipative Discrete Demand-Supply Model

This paper presents an analysis of the periodicity solutions to the discrete anticipative cobweb model. Dubois' anticipative principle was applied in the modification of Kaldor's cobweb model. Characteristic solutions are gained through the application of a simulation, which determines the cyclical behaviour of supply and demand. *Z-transform* was applied in the determination of the solutions. The interconnection between the anticipative definition of the cobweb model and Hicks model is addressed.

Keywords: cobweb, hyperincursivity, system dynamics, anticipative system, nonlinear system, Farey tree, chaos

Uroš Rajkovič, Olga Šušteršič, Jože Zupančič

A Model of E-documentation of Community Nursing

This article presents the development of electronic documentation for community nursing using a system approach. Documentation is viewed as an information model for organizing and managing processes. The community nurse plans the nursing process after gathering and evaluating information on the patient's health and his/her family status. Documentation is thus considered to be a basis for the successful work of the health team and as a way of ensuring quality in nursing. The article describes a prototype software model for e-documentation in community nursing together with its evaluation in practice.

Key words: nursing, community nursing, modelling, documentation, software solution

Editorial 10/2006

The aim of this special issue is to continue presenting the research activity of the Cybernetics and Decision Support Laboratory at the University of Maribor, the Faculty of Organizational Sciences in the field of the modelling and simulation of complex systems. The special issue includes papers dealing with the development of simulation methodology, modelling tools and practice for decision assessment in parallel processing, service systems, control and optimization, social dynamics research and living laboratory development.

The first paper, entitled "Simulation System Design", addressed the use of computer simulation as a tool for designing complex stochastic systems and some technical problems regarding computing time. This paper proposes a "stochastic approximation" algorithm to estimate the necessary controllable input parameters within a desired accuracy given a target value for the performance function. The solution algorithm is based on Newton's methods, using a single-run simulation approach to estimate the needed derivative. The approach proposed may be viewed as an optimization scheme, where the loss function must be minimized. The solution algorithm properties and the validity of the estimates are examined by applying it to some reliability and queuing systems with known analytical solutions.

The paper entitled "Modelling Parallel Service Systems in GPSS" deals with parallel service system modelling in the GPSS simulation language. The problem can be solved using a variety of theoretical approaches. In this article, the simulation method carried out by a digital computer is being used. The transactions entering the systems select between numerous different servers and we can mostly detect two rules in the selection of the appropriate server. The first rule always gives the first few (regarding its position in the system) entities (either servers or queues) precedence over the others, while the second rule always treats all the equal entities evenly and selects among them quite randomly. Since GPSS normally operates by the first rule, we frequently come up against difficulties when modelling systems that serve by another rule. The present article offers a methodology how to solve this problem within GPSS.

The paper entitled "Discrete Event Passenger Flow Simulation Model for an Airport Terminal Capacity Analysis" describes an analysis of passenger flow in an airport departure terminal, from the passenger entrance to boarding, involving the development of a simulation model. Data was collected and used to define the inputs to the simulation model and that was followed by a statistical analysis of the collected data. A discrete-event simulation model was constructed using the General Purpose Simulation System (GPSS) simulation programming language. The performance of the system in the present and the expected future was studied. The simulation model helped us to evaluate the passenger flow, identify system bottlenecks as well as the system capacities. Critical aspects in the passenger flow through the airport terminal have also been explored and studied.

The paper entitled "A Simulation Approach to Warehouse Cost Minimization in a Stochastic Environment" describes the simulation method used to solve replenishment strategy problems in a medium-sized company in order to improve its warehousing processes. The simulation approach is used to minimize total warehousing cost while ensuring no stock-outs occur and that the warehouse capacity is not exceeded. A case study of optimization to the replenishment process is presented on several representative materials of an automotive company using two replenishment algorithms: fixed review period and full capacity ordering. The simulation results presented indicate a considerable cost reduction without violating the constraints mentioned.

The paper entitled "The Periodicity of Discrete Demand-Supply Model Solutions using the Anticipative Principle" presents analysis of the periodicity solutions of the discrete cobweb model. Dubois' anticipative principle was applied in the modification of Kaldor's cobweb model. By the application of simulation, characteristic solutions are gained that determine the cyclical behaviour of demand and supply. Z-transform was applied when the solutions were determined. The interconnection between the anticipative definition of the cobweb model and the Hicks model is addressed.

The last paper of this issue, "A Model of the E-Documentation of Community Nursing", presents the development of electronic documentation for community nursing using a system approach. Documentation is viewed as an information model for the organization and management of processes. The community nurse plans the nursing process after gathering and evaluating information on the patient's health and his/her family status. The article describes a prototype software model for e-documentation in community nursing together with its evaluation in practice.

Guest editor
Miroljub Kljajić

Recenzenti v letu 2006 / Reviewers in 2006

Bašič Hrvatin Sandra, Univerza v Ljubljani
Belič Igor, Univerza v Mariboru
Berk Aleš, Univerza v Ljubljani
Bernik Igor, Univerza v Mariboru
Bohanec Marko, Inštitut Jožef Stefan
Bračun Franc, Abanka Vipa d.d.
Bratina Danijel, Univerza na Primorskem
Bren Matevž, Univerza v Mariboru
Bukovec Boris, VŠUP Novo mesto
Černetič Metod, Univerza v Mariboru
Čižman Anton, Univerza v Mariboru
Devetak Gabrijel, Univerza v Mariboru
Dinevski Dejan, Univerza v Mariboru
Dolinšek Slavko, Univerza na Primorskem
Duh Mojca, Univerza v Mariboru
Ferjan Marko, Univerza v Mariboru
Geršak Suzana, OŠ France Prešeren Kranj
Gomišček Boštjan, Univerza v Mariboru
Gorenak Vinko, Vlada RS, Ministrstvo za notranje zadeve
Gradišar Mirko, Univerza v Ljubljani
Gričar Jože, Univerza v Mariboru
Hočevar Marko, Univerza v Mariboru
Hudoklin Božič Alenka, Univerza v Mariboru
Ivankovič Gordana, Univerza na Primorskem
Jakomin Igor, Univerza v Ljubljani
Jereb Eva, Univerza v Mariboru
Jesenko Jože, Univerza v Mariboru
Jurše Milan, Univerza v Mariboru
Kavčič Slavka, Univerza v Ljubljani
Kljajić Miroljub, Univerza v Mariboru
Klopčič Sonja, Oria computers
Knez Rajko, Univerza v Mariboru
Koletnik Franc, Univerza v Ljubljani
Koletnik Mihaela, Univerza v Mariboru
Konrad Edvard, Univerza v Ljubljani
Kos Drago, Vlada Republike Slovenije
Kos Marko, Univerza v Ljubljani
Kovač Jurij, Univerza v Mariboru
Kovačič Andrej, Univerza v Ljubljani
Krašna Tadeja, Višja strokovna šola za gostinstvo in turizem Bled
Lahovnik Matej, Univerza v Ljubljani
Lenart Gregor, Univerza v Mariboru
Lipičnik Bogdan, Univerza v Ljubljani
Ljubič Anton, Univerza v Mariboru
Makovec - Brenčič Maja, Univerza v Ljubljani
Marn Ferdinand, Univerza v Mariboru
Mayer Janez, Univerza v Mariboru
Meglič Jure, Univerza v Mariboru
Mihelčič Peter, Višja strokovna šola za gostinstvo in turizem Bled
Milost Franko, Univerza na Primorskem
Mrkaić Mićo, Univerza v Mariboru
Mulej Matjaž, Univerza v Mariboru
Nećak Lük Albina, Univerza v Ljubljani
Novak Vesna, Univerza v Mariboru
Odar Marjan, Univerza v Ljubljani
Ovsenik Marija, Univerza na Primorskem
Pavlovič Zoran, Pedagoški inštitut Ljubljana
Podlogar Mateja, Univerza v Mariboru
Pucihar Andreja, Univerza v Mariboru
Rajkovič Vladislav, Univerza v Mariboru
Rant Živa, Ribenska cesta 30, Bled
Rejc Buhovac Adriana, Univerza v Ljubljani
Roblek Matjaž, , Univerza v Mariboru
Rozman Rudi, Univerza v Ljubljani
Sedej Marjan, Modest d.o.o., Preserje
Senegačnik Marjan, Univerza v Mariboru
Škraba Andrej, Univerza v Mariboru
Šmitek Branislav, Univerza v Mariboru
Tavčar Mitja, Univerza v Ljubljani
Treven Sonja, Univerza v Mariboru
Twrdy Elen, Univerza v Ljubljani
Uhan Stane, Univerza v Mariboru
Vehovar Vasja, Univerza v Ljubljani
Vida Irena, Univerza v Ljubljani
Vovk Aleš, GOinfo d.o.o.
Vukovič Goran, Univerza v Mariboru
Werber Borut, Univerza v Mariboru
Završnik Bruno, Univerza v Mariboru
Zidanski Majda, Albatros Bled
Zupan Nada, Univerza v Ljubljani
Zupan Neja, Univerza v Mariboru
Zupančič Jožef Mihael, Univerza v Mariboru
Žabkar Vesna, Univerza v Ljubljani

Simulation System Design

Hossein Arsham[1], **Miroljub Kljajić**[2]

[1]University of Baltimore, Baltimore, MD, 21201, USA, harsham@ubalt.edu
[2]University of Maribor, Faculty of Organizational Sciences, Kidričeva cesta 55a, SI-4000 Kranj, Slovenia,
miroljub.kljajic@fov.uni-mb.si

The use of simulation as a tool to design complex stochastic systems is often inhibited by cost. Extensive computer processing is needed to find a design parameter value given a desired target for the performance measure of a given system. The designer simulates the process numerically and obtains an approximation for that same output. The goal is to match the numerical and experimental results as closely as possible by varying the values of input parameters in the numerical simulation.
The most obvious difficulty in solving the design problem is that one cannot simply calculate a straightforward solution and be done. Since the output has to be matched by varying the input, an iterative method of solution is implied. This paper proposes a "stochastic approximation" algorithm to estimate the necessary controllable input parameters within a desired accuracy given a target value for the performance function. The proposed solution algorithm is based on Newton's methods using a single-run simulation approach to estimate the needed derivative. The proposed approach may be viewed as an optimization scheme, where a loss function must be minimized. The solution algorithm properties and the validity of the estimates are examined by applying it to some reliability and queueing systems with known analytical solutions.

Key words: system design and simulation, local response surface, goal seeking problem, parameter setting design, gradient estimation, discrete-event systems

Načrtovanje in planiranje z metodo simulacije

Uporaba simulacije kot orodja za načrtovanje kompleksnih stohastičnih sistemov je pogosto časovno zahtevna naloga. Potereben je izdaten računalniški čas da se najde vrednost vhodnih parametrov ki ustrezajo željenim performansam sistema. Načrtovalec simulira proces numerično za izbrane vhodne parametre da dobije oceno želene vrednosti izhoda. Cilj je da dobimo kar se da slične vrednosti experimentalnih in simulacijskih rezultatov z variranjem vhodnih parametrov simulacijskega modela. Pproblem je da ne obstaja enostaven način računanja da direkto dobimo zahtevanno rešitev problema. Ker izhod (rešitev) mora odgovarati enoj od možnih vrednosti vhodnih parametrov metoda reševanja je nujno iterativna kar zahteva veliko računalniškega časa. V tem članku predlagava postopek "stohastičnega približka" za oceno potrebnih controlabinih vhodnih parametrov za določitev željene vrednosti sistema v mejah predpisane zanesljivosti. Predlagani algoritam temelji na Newtonovi metodi, kjer spomočjo (enega) simulacijskega teka ocenimo prvi odvod potreban za optimizacijo kriterijske funkcije. Predlagani postopek lahko razumemo kot optimizacijsko shemo, kjer funkcijo izgube je treba minimizirati. Predlagani postopek je preizkušen in ovrednoten na nekaj primerih zanesljivosti in sistemov strežbe z znanimi analitičnimi rešitvami.

Ključne besede: načrtovanje in simulacija, metoda lokalnega odziva, ciljno optimiziranje, parametrska optimizacija, gradientne ocene, dogodkovna simulacija

1 Introduction

Simulation continues to be the primary method by which engineers obtain information about analysis of complex stochastic systems, such as production assembly lines, flexible manufacturing systems, and reliability systems. Almost all stochastic system performance evaluation can be formulated as an estimation of an expected value. Consider a system with continuous parameter
$v \in V \subseteq R$, where V is an open interval. Let

$$J(v) = E_{Y|v}\,[Z\,(Y)], \qquad (1)$$

be the steady-state expected performance measure, where Y is a random vector with known probability density function (pdf), f(y;v) depends on v, and Z is the performance measure. For example, in a reliability system, J(v) might be the mean time to failure; Z is the lifetime of a system; Y is the lifetime of the components; and v might be the components' mean lifetimes. In general, v is the shape or scale parameter of the underlying pdf. Another example could be a queueing system, where Y is the sequence of a two-dimensional vector of inter-arrivals and service times, Z is the delay in the system, and v is the arrival rate.

Before proceeding further, we distinguish between discrete event static systems (DESS) and discrete event dynamic systems (DEDS). Dynamic systems evolve over time; static systems do not evolve over time. Examples of dynamic systems are the queuing systems; examples of static systems are reliability systems. Note that while in DESS, Y is a multidimensional vector; in DEDS, Y represents a stochastic process. This paper deals with perturbation analysis of both DESS and DEDS.

In systems analysis, we resort to simulation when Z is either unknown or is too complicated to calculate analytically. Simulation is needed to estimate J(v) for most DESS and DEDS. The principal strength of simulation is its flexibility as a systems analysis tool for highly complex systems.

In discrete event systems, Monte Carlo simulation is usually needed to estimate J(v) for a given value $v = v_0$. By the law of large numbers

$$\hat{J}(v_0) = 1/n \sum_{i=1}^{n} Z(y_i), \qquad (2)$$

converges to the true value, where y_i, i = 1, 2, ..., n are independent, identically distributed random vector realizations of Y from f (y; v_0), and n is the number of independent replications. The numerical result based on (2) is only a point estimate for J(v) at $v = v_0$. The numerical result based on (2) is a solution to a ***system analysis***: Given the underlying pdf with a particular parameter value v_0, estimate the output function $J(v_0)$. The direct problem is widely used in stochastic systems analysis. Now we pose the ***system design problem***: Given a target output value of the system and a parameterized pdf family, find an input value for the parameter which generates such an output. The solution to the design problem has potential application in stochastic systems analysis and design. Mathematical formulation of the design problem is as follow:

Given τ, find $v \in V \subseteq R$ subject to $J(v) = \tau$, where

$$J(v) = E_{Y|v} [Z(Y)] = \int Z(y)\, f(y; v)dy, \qquad (3)$$

$Z: R^m \rightarrow R$ is a system performance measure
$Y \in R^m$ is a random vector (or a truncated stochastic process) with pdf f (y; v)

The design problem is essentially backwards. The output is given, but the input must be determined. This is easiest to appreciate when a designer wants to match experimental data in order to obtain some basic parameters. The designer simulates the process numerically and obtains an approximation for that same output. The goal is to match the numerical and experimental results as closely as possible by varying the values of input parameters in the numerical simulation. Analyzing this, clearly, the output is there, and it is the input quantity that needs to be determined. The most obvious difficulty in solving the design problem is that one cannot simply calculate a straightforward solution and be done. Since the output must be set by varying the input, an iterative method of solution is implied. Our approach may be viewed as an optimization scheme where a loss function must be minimized. Therefore, the process of solving a design problem often comes down to finding the best method of minimizing the loss function. The key part of optimization is to compute the derivative of the output with respect to an input parameter.

There are strong motivations for both problems. In the case when v is any controllable or uncontrollable parameter, the designer is interested in estimating J(v) for a *small* change in $v = v_0$ to $v = v_0 + \delta v_0$. This is the so-called what-if problem which is a direct problem. However, when v is a controllable input the decision maker may be interested in the goal-seeking problem; i.e., "What perturbation of the input parameter will achieve a desired change in the output value?". Another application of the design problem is where we may want to adapt a model to satisfy a new constraint with stochastic function. While the what-if problem has been extensively studied, the goal-seeking simulation problem is relatively new. Design interpolation based on regression models provides an indirect approach to solve the design problem. In this treatment, one simulates the system for many different values of $v = v_0$ and then one approximates the response surface function J(v), see e. g. (Kleijnen, 1979). Finally, the fitted function is used to interpolate to obtain the unknown parameter v. Since the shape of J(v) function is unknown, this approach is tedious, time-consuming and costly. Moreover, in random environments, the fitted model might have unstable estimates for the coefficients. The only information available about J(v) is general in nature, for example, continuity, differentiability, invertability, and so on.

The simulation models based on (2), although simpler than the real-world system, are still a very complex way of relating input (v) to output J(v). Sometimes a simpler analytic model may be used as an auxiliary to the simulation model. This auxiliary model is often referred to as a local response surface model (known also as a metamodel (Friedman, 1996). Local response surface models may have different goals: model simplification and interpretation (Yu & Popplewell, 1994), optimization (Arsham, 1996), what-if analysis (Arsham, 1996a), and generalization to models of the same type. The following polynomial model can be used as an auxiliary model.

$$J(v) = J(v_0) + \delta v.J'(v_0) + (\delta v)^2 J''(v_0) / 2 + ..., \qquad (4)$$

where $\delta v = v - v_0$ and the primes denote derivatives. This local response surface model approximates J(v) for small δv. To estimate J(v) in the neighborhood of v_0 by a linear function, we need to estimate the nominal J(v) based on (2) and its first derivative. Traditionally, this derivative is estimated by crude Monte Carlo; i.e., finite difference which requires rerunning the simulation model. Methods which yield enhanced efficiency and accuracy in estimating, at little additional cost, are of great value.

There are few ways to obtain efficiently the derivatives of the output with respect to an input parameter (Arsham, 1998). The most straightforward method is the Score Function (SF). The SF approach (Arsham et al., 1989) is the major method for estimating the performance measure and its derivative, while observing *only a single* sample path from the underlying system (Rubinstein & Melamed, 1998). The basic idea of SF is that the derivative of the performance function, J'(v), is expressed as expectation with respect to the *same* distribution as the performance measure itself.

This paper treats the design problem as a simulation (as opposed to regression) problem. By this approach, we are able to apply variance reduction techniques (VRT) used in the direct problem. Specifically, we embed a stochastic version of Newton's method in a recursive algorithm to solve the stochastic equation J(v) = J for v, given J at a nominal value v_0.

The explicit use of a linear local response surface model is the target parameter design: Given a desired value J = J(v), find the prerequisite input parameter v.

Most engineering design methods essentially involve a framework for arriving at a target value for product, process, and service attributes through a set of experiments which include Monte Carlo experiments. To solve the product design problem, we will restrict our model to the first order expansion. For a given J(v) the estimated δv using (4) is

$$\hat{\delta}v = [J(v)-\hat{J}(v_0)] \,/\, \hat{J}'(v_0), \qquad (5)$$

provided that the denominator in (5) does not vanish for all v_0 in interval V.

The remainder of this article is divided into eight sections. The next section contains the construction of a polynomial local response surface model using estimated derivatives of J(v). Section 3 deals with the target setting problem in design of a system. This is followed by construction of an accuracy measure. Section 5 develops an iterative solution algorithm for the parameter selection problem. Sections 6 and 7 illustrate the proposed method for reliability and queueing systems, respectively. Finally, Section 8 provides some concluding remarks and ideas for further research and extensions.

2 Polynomial Local Response Surface Model Construction by Single-Run Simulation

Simulation models, although simpler than real-world systems, are still a very complex way of relating input parameters (v) to performance measures J(v). Sometimes a simple analytical model may be used as an auxiliary to the simulation model. This auxiliary local response surface model is often referred to as a metamodel (Friedman, 1996). In this treatment, we have to simulate the system for some different values of (v) and then use a "goodness-of-fit" regression. We fit a response surface to these data (Kleijnen, 1979). Clearly, coupling the simulation model with the Score Function method enhances the efficiency of local response surface model construction. A local response surface model can also be constructed by using sensitivities in a Taylor expansion of J(v) in the neighborhood of v = v_0. The resulting local response surface model can be used for characterization (such as increasing/decreasing, and convexity/concavity) of the response surface.

Let

$$J(v) = E_{Y|v}\,[Z(Y)] = \int Z(y)\, f(y; v)dy, \qquad (6)$$

Z is a system performance measure

$Y \in R^m$ is a random vector (or a truncated stochastic process) with pdf f (y; v)

be the steady state performance measure, then

$$J'(v) = \int [\, Z(y).f(y;v)]'\, dy, \qquad (7)$$

where the prime (') denotes the derivative with respect to v. Note that despite the fact that y depends on v, only the function Z.f is subject to differentiation with respect to v. From (7) it follows that

$$J'(v) = \int Z(y)\, f'(y; v)dy = E_{Y|v}\,[Z(Y)\, S], \qquad (8)$$

where S = f'(y;v) / f(y;v) is the Score Function. Differentiation is with respect to v. This is subject to the assumptions that the differentiation and the integration operators are interchangeable, f'(y;v) exists, and f(y;v) is positive for all vεV, where V is an open interval. A necessary and sufficient condition for the interchangeability used above is that there must be no discontinuity in the distribution with position depending on the parameter v (Arsham, 1996a). Similarly, the second derivative is

$$J''(v) = \int [\, Z(Y)\, S'\, f(y; v) + Z(Y)\, S\, f'(y; v)]dy = E_{Y|v}\,[Z(Y)\, H] \qquad (9)$$

where

$$H = S' + S^2. \qquad (10)$$

In the multidimensional case, the gradient and Hessian of J(v) could be obtained in a straightforward manner by generalizing these results (Arsham, 1998). The estimator for the first and second derivatives based on (8) and (9) are given by:

$$\hat{J}'(v_0) = \sum_{i=1}^{n} Z(y_i).S(y_i; v)/n \qquad (11)$$

$$\hat{J}''(v_0) = \sum_{i=1}^{n} Z(y_i).H(y_i; v_0)/n \qquad (12)$$

where

$$S(y_i; v_0) = f'(y_i; v_0) / f(y_i; v_0) \qquad (13)$$

and

$$H(y_i; v_0) = f''(y_i; v_0) / f(y_i; v_0). \qquad (14)$$

Notice that both (11) and (12) estimators are evaluated at v = v_0, and y_i 's are the same n independent replications used in (2) for estimating the nominal performance $J(v_0)$; therefore they are quite efficient in terms of CPU cost. Estimates obtained by using (11) and (12) are unbiased, consistent, and they converge to the true values in the sense of the mean squared error (Arsham, 1998). The estimated gradient can also be used in solving optimization problems by simulation using the stochastic version of the classical nonlinear programming algorithms (Arsham, 1996). Other applications of sensitivity information include stability analysis (Arsham, 1996a).

3 Design-setting Problem

Most engineering system designs such as product, process, and service design, involve a framework for arriving at a target value for a set of experiments, which may include Monte Carlo experiments. A random quality loss function $L(Z_i)$ for a given system can be expanded in the neighborhood of the target value τ as follows:

$$L(Z_i) = L(\tau) + (Z_i-\tau)L'(\tau) + (Z_i-\tau)^2 L''(\tau)/2 + ... \qquad (15)$$

It can be shown that $L(Z_i)$ converges in *mean squared error* if $*Z_i-\tau*<1$ and derivatives are finite. Since the *optimal* loss is zero at τ, equation (15) reduces to the following quadratic approximation

$$L(Z_i) = K (Z_i-\tau)^2 \qquad (16)$$

In (16), K is some constant which can be determined in terms of the customer's tolerance limit
(τ - δ), which suggests that the product performs unsatisfactorily when Z_i slips below this limit. Given that the cost to customer is A dollars, then $K = A/\delta^2$. Without loss of generality, for simplicity let K=1.

The goal of parameter design is to choose the setting of the design parameter v that minimizes the average loss (the risk function). The risk function R(τ) is the expected value of the loss function, which can be shown as:

$$R(\tau) = E \{L(Z_i)\} = (J - \tau)^2 + \mathrm{Var}\ (Z_i), \qquad (17)$$

This risk function measures the average loss due to a product performance which is proportional to the square of the deviation from the target value τ .

The non-adjustable variational noise; i.e.;

$$\mathrm{Var}\ (Z_i |v) = \mathrm{Var}\ (Z_i), \qquad (18)$$

is a measure of variation among products. However, the role of product design is to reduce the $(J - \tau)^2$ part of risk, which is our interest in this paper. Note that all estimates involved in computing δv based on (5); i.e., in

$$\hat{\delta} v = [J(v)-\hat{J}(v_0)] / \hat{J}' (v_0) \qquad (19)$$

are computed *simultaneously* from *a single-run simulation* of the nominal system ($v = v_0$). This was achieved by transforming all probability space to the nominal one. Note that to estimate the derivative we do not need to rerun the simulation. To estimate the derivatives adds only moderate computational cost to the base simulation.

4 Accuracy of the Estimate

In the design problem, input parameter is random, while the output is fixed and given as a target value. Upon estimating the input parameter, we must provide a measure, such as a confidence interval, to reflect the precision of the estimate. To construct a confidence interval for δv using the estimator (19), let

$$A_i = J(v) - Z(y_i; v_0), \qquad (20)$$
$$B_i = Z(yi ; v_0) S(y_i; v_0) \qquad (21)$$

and denote

$$A = \Sigma A_i /n , \text{ and } B = \Sigma B_i /n, \qquad (22)$$

then

$$S^2 = S_{11}^2 - 2\hat{}v S_{12} + (\hat{}v)^2 S_{22} \qquad (23)$$

where

$$S_{11} = \Sigma (A_i - A)^2/(n-1), S_{22} = \Sigma (B_i - B)^2/(n-1), \qquad (24)$$

and

$$S_{12} = \Sigma (A_i -A)(B_i - B) /(n-1), \qquad (25)$$

An exact 100 (1- α) % confidence interval for δv is given by

$$P [n^{½} \frac{|\delta v -v |}{S/B} \leq t_{n-1, 1- \alpha / 2}] \geq 1-\alpha, \qquad (26)$$

where $t_{n-1,1- \alpha / 2}$ is the 100 (1- α / 2) percentile of Student's t distribution with (n-1) degrees of freedom (Kleijnen & Van Groenendaal, 1992).

5 A Recursive Solution Algorithm

The solution to the design problem is a solution of the stochastic equation J(v) = J, which we assume lies in some bounded open interval V. The problem is to solve this stochastic equation by a suitable experimental design to ensure convergence as δv approaches zero. The following algorithm involves placing experiment j+1 according to the outcome of experiment j immediately preceding it. That is,

$$v_{j+1} = v_j+d_j [\tau - \hat{J}(v_j)] / \hat{J}'(v_j), \qquad (27)$$

where d_j is any sequence of positive numbers satisfying the following conditions:

$$\sum_{j=1}^{\infty} d_j = \infty, \qquad (28)$$

and

$$\sum_{j=1}^{\infty} d_j^2 < \infty, \qquad (29)$$

The first condition is a necessary condition for the convergence δv to approach zero, while the second condition asymptotically dampens the effect of the simulation random errors. These conditions are satisfied, for example, by the harmonic sequence $d_j = 1/j$. With this choice, the rate of reduction of d_i is very high initially but may reduce to very

small steps as we approach the root. Therefore, a better choice is, for example
$d_j = 9 / (9 + j)$.

Since the adjustments are made in proportion to the recent value, we must be sure that the results remain finite. This requires that J'(v) does not vanish for $v \in V$, where V is an open interval. To prevent excessive over-correction, we assume further that the solution lies in some finite interval V. Under these not unreasonable conditions, this algorithm will converge in mean square; moreover, it is an almost sure convergence. For some generalizations and studies concerning speed of convergence and acceleration techniques, see (Dippon & Renz, 1997). Finally, as in Newton's root-finding method, it is impossible to assert that the method converges for just any initial $v = v_0$, even though J'(v) may satisfy the Lipschits condition over V. Indeed, if the initial value v_0 is sufficiently close to the solution, which is usually the case, then this algorithm requires only a few iterations to obtain a solution with very high accuracy.

ALGORITHM

```
Step 0. INPUTS
        $\tau$ = Desired output
        j = Iteration number
        $v_j$ = Controllable input parameter v
        n = Sample size
        U = Desired upper limit for absolute increment u = $v_{j+1} - v_j$
        $\alpha$ = A desired significance level

Step 1. INITIALIZATION
        Set j=1
        Set $v_j = v_0$

Step 2. ESTIMATIONS
        $J(v_j)$ using (2)
        $J'(v_j)$ using (9)

Step 3. COMPUTATIONS
        $u = 9[\tau - \hat{J}(v_j)] / [(9+j)\hat{J}'(v_j)]$
        If | u | < U
        Construct 100( 1- $\alpha$ )% confidence interval for v
        using (20)
        Stop.
        Otherwise
        set $v_{j+1} = v_j + u$ and $j \rightarrow j+1$

Step 4. RESET: Reset the seeds of random number generators
to their initial values. Go to step 2.
```

Note that, by resetting the seeds to their initial values, we are using the Common Random Variate approach as a variance reduction technique.

6 Design of a Reliability System

For most complex reliability systems, the performance measures such as mean time to failure (MTTF) are not available in analytical form. We resort to Monte Carlo Simulation (MCS) to estimate MTTF function from a family of single-parameter density functions of the components life with specific value for the parameter. The purpose of this section is to solve the design problem which deals with the calculation of the components' life parameters (such as MTTF) of a homogeneous subsystem, given a desired target MTTF for the system. A stochastic approximation algorithm is used to estimate the necessary controllable input parameter within a desired range of accuracy. The potential effectiveness is demonstrated by simulating a reliability system with a known analytical solution.

Consider a coherent reliability sub-system consists of 4 homogeneous elements; i.e., manufactured by an identical process, components having independent random lifetimes Y_1, Y_2, Y_3, and Y_4, which are distributed exponentially with rates $v = v_0 = 0.5$.

The first 2, and the last two elements are in series, while these tow series each with two components are in parallel. The system lifetime is $Z(Y_1,Y_2,Y_3,Y_4; v_0) = \max[\min(Y_3,Y_4), \min(Y_1,Y_2)]$. It is readily seen that the theoretical expected lifetime of this system is $J(v_0) = 3/(4 v_0)$, (Barlow & Proschan, (1975). Now we apply our results to compute a necessary value for v to obtain a particular value for J(v), say J(v) = 2. For this reliability system, the underlying probability density function is:

$$f(y;v) = v^4 \exp(-v \Sigma y_i),\ i = 1, 2, 3, 4. \qquad (30)$$

The Score Function is

$$S(y) = f'(y; v) / f(y; v) = 4/v - \Sigma y_i,\ i = 1, 2, 3, 4. \qquad (31)$$

$$H(y) = f''(y; v)/ f(y; v) = [v^2 (\Sigma y_i)^2 - 8v (\Sigma y_i) + 12]/v^2,\ i = 1, 2, 3, 4. \qquad (32)$$

The estimated average lifetime and its derivative for the nominal system ($v = v_0 = 0.5$) based on (2) and (9) are

$$J(v_0) = \Sigma \max[\min(Y_{3,j},Y_{4,j}), \min(Y_{1,j},Y_{2,j})] / n, \qquad (33)$$

and

$$J'(v_0) = \Sigma \max[\min(Y_{3,j},Y_{4,j}), \min(Y_{1,j},Y_{2,j})] . S(Y_{i,j}) / n, \qquad (34)$$

$$J''(v_0) = \Sigma \max[\min(Y_{3,j},Y_{4,j}),\min(Y_{1,j},Y_{2,j})] . H(Y_{i,j})/n, \qquad (35)$$

respectively where $Y_{i,j}$ is the jth observation for the ith component (i = 1, 2, 3, 4). We have performed a Monte Carlo experiment for this system by generating n = 10000 independent replications using SIMSCRIPT II.5 random number streams 1 through 4 to generate exponential variates Y_1, Y_2, Y_3, Y_4 , respectively, on a VAX system. The estimated performance is J(0.5) = 1.5024, with a standard error of 0.0348. The first and second derivative estimates are -3.0933 and 12.1177 with standard errors of 0.1126 and 1.3321, respectively.

The response surface approximation in the neighborhood v = 0.5 is:

$$J(v) = 1.5024 + (v - 0.5)(-3.0933) + (v - 0.5)^2 (12.1177)/2 + \ldots . \ 6.0589v^2 - 9.1522v + 4.5638 \qquad (36)$$

A numerical comparison based on direct simulation and local response surface model (36) is given in Table 1.

Notice that the largest error in Table 1 is 0.33% which could be reduced by either more accurate estimates of the derivatives and/or using a higher order Taylor expansion. A comparison of the errors indicates that the errors are smaller and more stable in the direction of increasing v. This behavior is partly due to the fact that lifetimes are exponentially distributed with variance 1/v. Therefore, increasing v causes less variance than the nominal system (with v = 0.50).

TABLE 1: A second order polynomial local response surface model and direct simulation

v	Analytic	Simulutation	Metamodel	Abs.error(%)
0.40	1.8750	1.8780	1.8723	0.14
0.42	1.7857	1.7885	1.7887	0.17
0.44	1.7045	1.7072	1.7098	0.31
0.46	1.6304	1.6330	1.6359	0.33
0.48	1.5625	1.5650	1.5667	0.27
0.50	***1.5000***	***1.5024***	***1.5024***	***0.16***
0.52	1.4423	1.4446	1.4430	0.05
0.54	1.3889	1.3911	1.3884	0.04
0.56	1.3393	1.3414	1.3386	0.05
0.58	1.2931	1.2951	1.2937	0.05
0.60	1.2500	1.2520	1.2537	0.29

Now assume that the manufacturer wants to improve the average lifetime of the system to
$J(v) = \tau = 2$. To achieve this goal, we have set $v_0 = 0.5$ and $U = 0.0001$ in the proposed algorithm. The numerical results are tabulated in Table 2.

TABLE 2: Iterative decision parameter estimate for the reliability system

(1) Iteration number j
(2) Fixed input v_j
(3) Estimated MTTF
(4) Estimated derivative
(5) Change in v_j
(6) New input parameter v_{j+1}

(1)	(2)	(3)	(4)	(5)	(6)
1	0.5000	1.5024	-2.9598	-0.1513	0.3487
2	0.3487	2.1544	-6.0862	-0.0208	0.3694
3	0.3694	2.0333	-5.4217	+0.0046	0.3740
4	0.3740	2.0083	-5.2888	+0.0011	0.3751
5	0.3751	2.0025	-5.2583	+0.0003	0.3754
6	0.3754	2.0009	-5.2498	+0.0001	0.3755
7	0.3755	2.0003	-5.2471	+0.0000	0.3756*

The estimated input parameter to achieve the output $J(v) = \tau = 2$ is 0.3756. A 90% confidence interval based on this estimate using (20) is:

$$P[0.3739 \leq v \leq 0.3773] \geq 0.90. \qquad (37)$$

Comparing the theoretical value $v_0 = 0.3750$, obtained from $J(v) = 3/4v_0 = 2$, with our computational value suggests that the results based on the proposed algorithm are quite satisfactory. In fact, running this system with v = 0.3756, and n = 10000, we obtained an estimated MTTF of J(v) = 2.0000. Hence the discrepancy in the estimated input parameter by this algorithm must be considered as a pure random error which can be reduced by increasing n. The metamodel (36) could also be applied to J(v) = 2 to estimate the desirable v. Solving the resulting quadratic equation, the relevant root is v = 0.3725. This result is an inferior estimate for v compared with the iterative method, although the accuracy of the latter comes with greater computational cost.

7 Design of a Service System

This section presents implementation details and some statistical results on the efficiency of the proposed technique for a discrete event dynamic system. To evaluate the proposed single-run technique to solve the design problem, we have chosen to implement it on an M/G/1 queueing system with a *known* analytical solution. Consider, a single-server, first-come-first-served, Poisson input queue with arrival rate of 1 customer per unit of time. The server works according to a Gamma density

$$f(y;v) = y\, e^{-y/v} / v^2, \; v > 0, \; y \geq 0. \qquad (38)$$

The analytic solution for the expected *steady-state* waiting time as a performance measure, in this system is:

$$J(v) = \rho + (\rho^2 + \sigma^2)/[2(1-\rho)] \qquad (39)$$

which is obtained by using the Pollaczek-Khintchin formula (Gross & Harris, 1998), where $\sigma^2 = Var(y) = 2v^2$ and ρ = traffic intensity =1/service rate = 2v. If we set the nominal value v = 0.25 for the nominal system, then we have $\sigma^2 = 0.125$ and $\rho = 0.5$ resulting in J(0.25) = 0.875.

To estimate J' (v) for the nominal system, we will use the method of Batch Means. Other methods, such as Independent Replications or Regenerative Method could also be used.

Batch Means is a method of estimating the steady-state characteristic from a single-run simulation. The single run is partitioned into equal size batches large enough for estimates obtained from different batches to be approximately independent. In the method of Batch Means; it is important to ensure that the bias due to initial conditions is removed to achieve at least a covariance stationary waiting time process. An obvious remedy is to run the simulation for a period (say R customers) large enough to remove the effect of the initial bias. During this warm-up period, no attempt is made to record the output of the simulation. The results are thrown away. At the end of this warm-up period, the waiting time of customers are collected for analysis. The practical question is "How long should the warm-up period be?" Abate and Whitt (Abate & Whitt, 1987) provided a relatively simple and nice expression for the time required (t_p) for an M/M/1/4 queue system (with traffic intensity ρ) starting at the origin (empty) to reach and remain within 100(1-p)% of the steady-state limit as follows:

$$t_p(\rho) = 2C(\rho)\ \mathrm{Ln}\ \{1/[(1-p)(1+2C(\rho))]\}/(1-\rho)^2 \qquad (40)$$

where

$$C(\rho)=[2+\rho + (\rho^2 + 4\rho)^{½}] / 4. \qquad (41)$$

Some notions of $t_p(\rho)$ as a function of r and p, are given in Table 3.

TABLE 3: Time (tp) required for an M/M/1 queue to reach and remain with 100(1-p)% limits of the steady-state value

Traffic Intensity ρ	100(1-p)% 95.0	99.0	99.9	99.99
0.10	3.61	6.33	10.23	14.12
0.20	5.01	8.93	14.53	20.14
0.30	7.00	12.64	20.71	28.79
0.40	10.06	18.39	30.31	42.23
0.50	15.18	28.05	46.47	64.89
0.60	24.70	46.13	76.79	107.45
0.70	45.51	85.87	143.61	201.36
0.80	105.78	201.53	338.52	475.51
0.90	435.74	838.10	1413.7	1989.4

Although this result is developed for M/M/1 queues, it has already been established that it can serve as an approximation for more general; i.e., GI/G/1 queues (Whitt, 1989). To compute the Score Function S, we need the density function of the steady-state process.

Clearly, for computational implementation, we need a truncated (say m-truncated) version of this process. The waiting time of customer t at steady state depends on values of the (m - 1) previous customers interarrival and service times. The dependency order m must be chosen so that the correlation between the waiting time of customer t and (t-m) is negligible. Notice that the order of dependency m is equivalent to the "Batch Size" widely discussed in simulation literature in connection with the method of Batch Means. We have chosen m = R large enough to ensure independency and not too large to create the singularity problem.

Let X_k and Y_k be the interarrival and service times of the k^th^ customer at steady state, k ≥ R+1. The underlying density function for the j^th^ customer, j ≥ 2R+1, in batch number i is:

$$f(v) = \sum_{k=j-m+1}^{j} f(y_k)\, f(x_k), \qquad j = (i+1)R+1, (i+1)R+2,\ldots, (i+2)R \quad (42)$$

where

$$f(x_k) = \exp(-x_k)$$

and

$$f(y_k) = [y_k \exp(-y_k / v)]/ v^2.$$

The expected waiting time for the nominal system is:

$$\hat{J}(v) = \sum_{i=1}^{n} \sum_{j=(i+1)R+1}^{(i+2)R} L_{i,j} / (Rn) \qquad (43)$$

where $L_{i,j}$ is the waiting time of the j^th^ customer in the i^th^ batch. The Score Function S is:

$$S_{j,i} = -2m / v + \Sigma\, x_{j,k} / v^2 \qquad (44)$$

For the nominal system ($v = v_0 = 2$), we have used n = 500 independent replications. In each run, we set k = m = T = 100. The estimated delay in the system and its derivative based on these simulation parameters are 1.007 and -0.951 with computed variance 0.001 and 0.012, respectively. Clearly, derivative estimators discussed in this paper work much better for terminating models for which only small number of observations are generated.

Consider the system described above. Assume we want to find a value for the controllable input parameter, service rate v, such that J(v) = J = 0.8. We have set $v_0 = 2$ and U = 0.0001 in the proposed algorithm. The simulation results are contained in Table 4. Our computations are performed on a PC computer using streams 1 and 2 of SIMSCRIPT II.5 to generate the inter-arrival and service times, respectively.

Table 4: Estimated service rate to achieve a desirable steady state average delay in an M/G/1/∞ queue.

Iteration Number	Fixed Input Parameter v_0	Estimated δv_0	Updated v_0
1	2.000	0.236	2.236
2	2.236	0.001	2.237
3	2.237	0.001	2.238
4	2.238	0.001	2.239
5	2.239	0.000	2.239

The estimated input parameter to achieve the output J(v) = 0.8, is v = 2.239 with standard error 0.128. A 95% confidence interval for δv at the fifth iteration, based on the usual t-statistic is:

$$P[\ -0.016 \le \delta v \le 0.016] \ge 0.95 \qquad (45)$$

A comparison of the analytical value v = 2.25, obtained from (39) with our estimated value suggests that the results based on the proposed algorithm are quite satisfactory. In fact, solving the direct problem using the same simulation parameters with $v_0 = 2.239$, the estimated expected waiting time turned out to be 0.800 with variance equal to 0.001. Hence the discrepancy in the estimated input parameter by this algorithm must be considered as a random error which can be reduced by increasing n.

The method of Independent Replication has lower efficiency than the method of Batch Means for the steady-state perturbation analysis. In the Independent Replication method, the output data are collected over a period of length T in a simulation run over a period of length R + m + T; (T could be as small as 1). The ratio T/(R+m+T), which is the fraction of CPU time generating useful data, would be very small. Clearly, the method of Batch Means is more efficient.

8 Conclusions

Almost all discrete event systems simulation computation can be formulated as an estimation of an expected value of the system performance measure, which is a function of an input parameter of the underlying probability density function. In the ordinary system simulation, this input parameter must be known in advance to estimate the output of the system. From the designer's point of view, the input parameters can be classified as controllable and uncontrollable (Morrice & Bardhan, 1995). The influential controllable input can be recognized by factor screening methods (Ruppert, 1985). In this paper, we considered the design problem: "What must be the perturbation of the current controllable input parameter value to achieve a desired output value?" The approach used in this study was:

- To estimate the derivative of the output function with respect to the input parameter for the nominal system by a single-run, and on-line simulation;
- To use this estimated derivative in a Taylor's expansion of the output function in the neighborhood of the parameter; and finally,
- To use a recursive algorithm based on the Taylor's expansion to estimate the necessary controllable input parameter value within a desired accuracy.

Under some mild and reasonable conditions, the algorithm converges to the desired solution with probability 1. The efficiency of the proposed algorithm in terms of accuracy is tested using an $M/G/1/\infty$ queuing service, as well as a reliability product designs with satisfactory results. The approach may have major implications for simulation modelers and practitioners in terms of time and cost savings. As always, since this experiment was done on these specific numerical examples, one should be careful in making any other generalizations.

In the course of future research:

1. We expect to introduce other efficient variance reduction techniques (VRT). The Common Random Variates as a VRT is already embedded in the algorithm. Notice that since

   $$E[S] = E\,[\mathrm{Ln}\, f\,]' = \int [\mathrm{Ln}\, f\,]'\, f\, dx = \int f\,'\, dx = [\int f\, dx\,]' = 0. \quad (46)$$

   We can express the gradient in terms of covariance between Z and S

   $$J'(v) = \mathrm{Cov}\,[Z(Y), S\,] = E[Z.S] + E[Z].E[S]. \quad (48)$$

   and

   $$J'(v) = E[Z(Y).S] + \alpha\, E[S] \quad (49)$$

   where α could be the optimal linear control. Note also that (6) can be written as:

   $$J'(v) = \int Z\,(y)\, f\,'\,(y; v)\, dy$$

   $$= \int Z\,(y)[\, f\,'\,(y; v) \,/\, \varphi(y; v)]\, \varphi(y; v)\, dy. \quad (50)$$

   The best choice for φ is the one proportional to Z(y).f ' (y; v). This minimizes the variance of J'(v); however, this optimal φ depends on the performance function Z(y), which is not known in advance for most cases. One may use the empirical version of Z(y).f ' (y; v). We recommend a pilot run to study the effectiveness of these and other variance reduction techniques before implementing them.
2. We expect to extend our methodology to higher order Taylor's expansion. We believe that there is a tradeoff between number of iterations, sample size in each iteration; and the order of Taylor's expansion. Clearly, estimating the second derivative requires a larger sample size n, but a fewer iterations to achieve the same accuracy.
3. We also expect to extend our methodology to the design problems with two or more unknown parameters by considering two or more relevant outputs to ensure uniqueness. By this generalization, we could construct a linear system of stochastic equations to be solved simultaneously by multidimensional versions of the stochastic approximation proposed in (Ruppert, 1985; Wei, 1987) as well as the Newton method in (Polak, 1997; Tyrtyshnikov, 1997) using the second order derivatives (e.g., Hessian).
4. The algorithms in this paper are presented in English-like step-by-step format to facilitate implementation in a variety of operating systems and computers, thus improving portability. However, there is a need to develop an expert system that makes the algorithms more practically applicable to stimulation in system design (Clymer, 1995).

Acknowledgment: This work is supported by The National Science Foundation Grant CCR-9505732.

Literature

Abate, J. & Whitt, W. (1987). Transient behavior of M/M/1 queue: Starting at origin, Queueing *Systems*, **2**: 41-65.

Arsham, H. (1996). Stochastic optimization of discrete event systems simulation, *Microelectronics and Reliability*, **36**: 1357-1368.

Arsham, H. (1996a). Performance extrapolation in discrete-event systems simulation, *International Journal of Systems Science*, **27**: 863-869.

Arsham, H. (1998). Algorithms for sensitivity information in discrete-event systems simulation, *Simulation Practice and Theory*, **6**: 1-22.

Arsham, H., Feuerverger, A., McLeish D., Kreimer J. & Rubinstein, R. (1989). Sensitivity analysis and the 'what-if' problem in simulation analysis, Mathematical *and Computer Modelling*, **12**: 193-219.

Barlow, R. & Proschan F. (1975). *Statistical Theory of Reliability and Life Testing Probability Models*, Holt Rinehart and Winston, New York.

Clymer, J. (1995). System design and evaluation using discrete event simulation with AI, European *Journal of Operational Research*, **84**, 213-225.

Dippon, J. & Renz J. (1997). Weighted means in stochastic approximation of minima, *SIAM Journal of Control and Optimization*, **35**: 1811-1827.

Friedman, L. (1996). *The Simulation Metamodel*, Kluwer Academic Publishers, Norwell, Massachusetts.

Gross, D. & Harris C. (1998). *Fundamentals of Queueing Theory*, John Wiley and Sons, New York.

Kleijnen, J. & Van Groenendaal, W. (1992). *Simulation: A Statistical Perspective*, Wiley, Chichester, U.K.

Kleijnen, J. (1979). Regression metamodel for generalizing simulation results, *IEEE Transactions on Systems, Man and Cybernetics*, *SMC*, **9:** 93-96.

Morrice, D. & Bardhan, I. (1995). A weighted least squares approach to computer simulation factor screening. *Operations Research*, **43**(5): 792-806.

Polak, E. (1997). *Optimization: Algorithms and Consistent Approximations*, Springer, New York.

Rubinstein, R., & Melamed, B. (1998). *Modern Simulation and Modeling*, Wiley, New York.

Ruppert, D. (1985). A Newton-Raphson version of the multivariate Robbins-Monro procedure, *Annals of Statistics*, **13**: 236-245.

Tyrtyshnikov, E. (1997). *A Brief Introduction to Numerical Analysis*, Birkhauser, Berlin.

Wei, C. (1987). Multivariate adaptive stochastic approximation, *Annals of Statistics*, **15**: 1115-1130.

Whitt, W. (1989). Planning queueing simulation, *Management Science*, **35**: 1341-1366.

Yu, B. & Popplewell, K. (1994). Metamodels in manufacturing: A review, *International Journal of Production Research*, **32**: 787-796.

Hossein Arsham is the Harry Wright Distinguished Professor of Decision Science and Statistics at the Merrick School of Business, University of Baltimore, USA. Education: D.Sc. in Operations Research, M.Sc. in Management Science, B.Sc. in Physics. His teaching, research, and consulting activities are multidisciplinary, interdisciplinary, and transdisciplinary which include: General Optimization, Systems Simulation, and Statistical Data Analysis. His aim in constructing model-based decision support systems is to provide models for the managers which are easy to understand, easy to use, while providing useful managerial information. He has published over 40 articles in recognized world journal. For more information about Dr. Arsham, visit his home page at: http://home.ubalt.edu/ntsbarsh/index.html.

Miroljub Kljajić is Professor at the Faculty of Organizational Sciences, University of Maribor in the field of System Theory, Decision Theory and Computer Simulation. He completed his Dipl. Eng., M.Sc, and D.Sc. at the Faculty of Electrical Engineering in Ljubljana. He developed a method of quantitative gait evaluation and a simulation system for decision making support in the business systems. He has been the principal investigator of many national and international modeling and simulation projects. As author and co-author he has published 26 scientific articles recognized in SCI.

Modelling Parallel Service Systems in GPSS

Marko Žibert[1], Miroljub Kljajić[2]

[1]ZZZS, Miklošičeva 24, 1000 Ljubljana, Slovenia, marko.zibert@zzzs.si
[2]Univerza v Mariboru, Fakulteta za organizacijske vede, 4000 Kranj, Kidričeva 55a, Slovenia, miroljub.kljajic@fov.uni-mb.si

This article treats parallel service system modelling in the GPSS simulation language. The transactions entering such systems select between numerous different servers and we can mostly detect two rules in the selecting of the appropriate server. The first rule always gives the first few (regarding its position in the system) entities (either servers or queues) precedence over the others, while the second rule always treats all the equal entities evenly and selects among them quite randomly. Since GPSS normally operates by the first rule, we frequently come up against difficulties when modelling systems that serve by another rule. The present article offers a methodology how to solve this problem within GPSS.

Key words: discrete simulation, modelling, GPSS, parallel service systems, queuing theory

Modeliranje sistemov paralelne strežbe v GPSS-u

Članek obravnava modeliranje paralelnih strežnih sistemov v simulacijskem jeziku GPSS. Transakcije, ki vstopajo v takšne sisteme, izbirajo med večjim številom strežnih mest. Pri zasedanju teh mest pa lahko v grobem zasledimo dva različna pravila. Prvo pravilo daje prednost zasedanju prvih (po svoji poziciji v sistemu) entitet (bodisi strežnikov, bodisi čakalnih vrst), medtem ko drugo pravilo obravnava te entitete enakovredno in izbira med njimi povsem naključno. Ker GPSS v svojem delovanju privzema prvo pravilo, lahko pri modeliranju sistemov, ki strežejo po drugem pravilu, pogosto naletimo na določene težave. Pričujoči prispevek ponuja metodologijo, kako znotraj tega jezika reševati omenjeni problem.

Ključne besede: diskretna simulacija, modeliranje, GPSS, sistemi paralelne strežbe, teorija vrst

1 Definition of the Problem

The parallel service of complex systems is currently an increasingly important research area. This area is gaining in significance as computer science progresses and a lot of scientific periodicals and reviews are now occupied with this field of studies (Katwijk and Zalewski, 1999). Namely the most common problem in service systems recently is the increasing demand for processing a large volume of transactions in real time. These requests could be normally complied with by simply decomposing the original system and its base activity into more dependent subsystems, each with its own activity. But by doing this, new problems can turn up. One of them is the distribution or allocation of the incoming transaction evenly to all the subsystems (the problem of load balancing). The problem can be solved using a variety of theoretical approaches, for instance by intelligent agents (Wooldridge and Jennings, 1995), by Markov chains (Rosenthal, 2000; Song et al., 2004), by Petri nets (Murata, 1989) and by the simulation method (Guariso et al., 1996). In this article, the simulation method, carried out by digital computer, is being used (the computer simulation method).

For the necessity of the simulation and the modelling, a lot of simulation languages (compilers as well as interpreters) have now been developed (Sang et al., 1994). They are being executed on various computers and on the different types of operation systems. One of the first of these languages, and at the same time also the most common, is the GPSS language (General Purpose Simulation System), which was developed in the early sixties for analyzing the responses of the IBM mainframe systems (Blake and Gordon, 1964). At that time it was called General Purpose Systems Simulator (Gordon, 1962). The main GPSS emphasized characteristics (Crain, 1997; Crain, 1998; Crain and Henriksen, 1999; Henriksen and Crain, 2000) that made it very popular among the end-users, such as:

- It was developed for different computer environments (IBM 370 mainframes, personal computers, etc)
- Different versions of GPSS are executable under different operation systems (Multiple Virtual Storage – GPSSSV, Disk Operating System – GPSS/PC)
- The base components of the simulation language (blocks) represent the constituents of the system very well, so we can quickly and easily model any service system taken from reality.
- It creates precise default statistics and reports during the execution of the simulation.
- It is able to perform additional statistics and reports on request.
- Through the **HELP** block it can access an external user-written program (in FORTRAN).

One of the most important characteristics listed above is certainly the structure of the simulation language.

Its main components, semantically meaningful model building blocks, are trying to functionally imitate a particular constituent part of the serving system. So the block names, such as **ADVANCE**, **ASSEMBLE**, **ENTER**, **LEAVE**, **RELEASE**, **SEIZE**, **TEST**, **TRANSFER**, **QUEUE** etc., allow even the uninitiated user to follow the logical flow of a model, at least roughly (Chisman, 1992). In fact, these blocks are just more or less adequate computer projections of the functioning constituents. Thus, without much knowledge of programming and by simply arranging these blocks as they can be seen in reality, we can quickly and easily build precise computer model of the real world system.

In spite of the fact that GPSS is a very user friendly simulation tool, users are not always successful in their modelling of reality. Although in some cases the simulation model is properly built according to the modelling methodology rules (and is also submitted to the syntax rules of GPSS) some considerable discrepancies between the behaviour of the model and the real system can be noticed during the phase of the model evaluation and validation.

The discrepancies described are particularly visible when the simulated system has more equal parallel servers and each of them has the same service characteristics. This means that the service times of each server have the same mean, the same variance and the same statistical distribution. In most cases, as we can also expect, the workload in such systems is evenly distributed among all of the servers. However, the GPSS simulation model that ought to represent such a system, contrary to our expectation, shows unequally loaded servers. In other words, the results of the simulation always indicates that the utilization is the highest at the first server and then it gradually decreases. If the occupation rate per server in the model – the utilization rate that is defined as the fraction of the time the server is working (Adan and Resing, 2001) – increases then the differences in the workloads among particular servers lessen, but the declining trend of the server utilization (from the first server to the last) still exists.

Considering this declining trend, it can be concluded that the discrepancy (the deviation from reality) is especially notable when the modelled systems have more parallel servers than they really need on behalf of system reliability and availability. Under normal circumstances most of these servers would simply be redundant, but in the area of informatics we are frequently dealing with automatic server systems that must be firmly reliable and continuously available, sometimes even under conditions of emergency and under minimum control by the operator. These requirements can be easily complied with some additional parallel servers that could normally be spared.

In this way (by adding additional parallel servers to the system) we are, of course, decreasing the occupation rate per server and, as was said before, we are also increasing the unsuitability of the GPSS model by contrast with the real system. Such a model usually shows that only first few servers are somewhat utilized while the others are completely free and standing idle.

The reasons for the problem described are in special GPSS blocks – the **TRANSFER** and **SELECT** blocks – designed for routing transactions to the target server. Various attributes of some sequential permanent entities, such as **facility** and **queue,** are compared in these blocks. The compared attribute of the **facility** entity is its current state of occupation (whether it is busy or not) and the compared attribute of the **queue** entity is the current length of the queue (the number of transactions waiting in the queue). If these compared attributes are equal then the current transaction always picks out the first positioned feasible entity (in GPSS programme code). For example, if the first n parallel servers in a model are occupied and if the next servers from n+1 to n+k are free, in this case the GPSS simulation always chooses the (n+1)-th server to execute the current transaction.

In reality the server systems more often than not behave quite differently under these circumstances. When the attributes of the compared entities are equal then one of the suitable entity is chosen by transaction clearly at random in most cases. We can experience this especially in the area of informatics where the randomness is even coded into the programmes, subroutines, macros, distribution modules etc. (Cicsplex SM Concepts and Planning; Žibert, 2005). So in the above case the transaction wouldn't precisely pick out the (n+1)-th server but, on the contrary, it would select any among the free k servers (from n+1 to n+k).

Although the server systems with the characteristics described are not very numerous, they can still be found in the real world. Mostly they are connected with the single queue that leads to the very first service facility. All the other service facilities are arranged in a row, one after another at some proper physical distance to each other (this discipline can be often carried out in banks where the customers join a single queue and the first person in line physically engages the nearest free bank-teller), so the transaction (the client, customer, etc.), after leaving the single queue, always seizes its nearest server.

Regarding our brief outline of the activities in the parallel server systems, we can conclude that the real issue is the order in which transactions seize one entity among all the equivalent entities (in case that the entity is a server facility), or enter one entity among all the equivalent entities (in case that the entity is a queue). Although there are also some other possibilities from the real world – especially where people (customers), with their characteristic behaviour, represent the transactions in a system (Azar et al., 1994; Mitzenmacher, 1997) – we would stress that in both cases the transaction serving could be:

- in random order, or in disorder (which is more frequent, even standard in some cases – and we could name it as service in random order);
- in an order of precedence (which is not very common in the real world but it is always used in the modelling with the GPSS programming language – which we could name the service in order of precedence).

As a result of the approaches explained, we can state that GPSS modelling of the parallel server system with the service in order of precedence is very easy and uncomplicated. Namely, both the system itself and the GPSS model use the service in order of precedence, so the simulation results are usually in accordance with what happens in the real system.

We always come up to against difficulties, on the

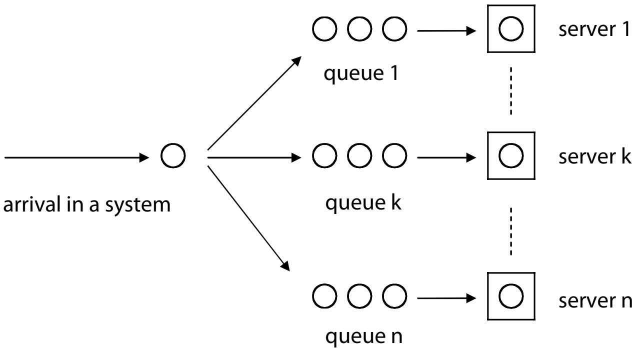


Figure 1: The scheme of the multiple servers system, each server with its own waiting queue

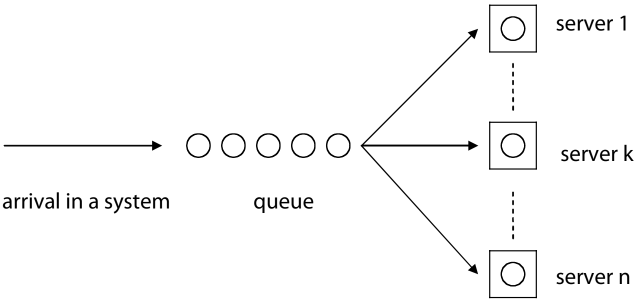


Figure 2: The scheme of the multiple server system with a single waiting queue

other hand, when we try to build a GPSS model of a parallel server system with the service in random order. The reason is obvious because the system and its model use different types of service order. As we said earlier the modelling problems are even bigger when the occupation rate per server in the system is low. In this case the simulated utilization of the parallel servers in the GPSS model would be completely inadequate.

That is why, in the following chapters, we are trying to develop a new GPSS methodology of modelling and simulating parallel server systems with the service in random order.

2 The Two Main Types of Parallel Service

The existing GPSS methodology of modelling parallel server systems depends on the general type of the parallel service we want to simulate. We can distinguish two main types of parallel service and each of them has its special solution within the normal (classical) usage of the GPSS programming language. That means that it has its own sequence of various GPSS blocks that should illustrate the functioning of the system.

The term "sequence of blocks" is not something that is fixed and defined once for all by the GPSS developers. We should consider it just as one variation among the many possibilities that GPSS programmer can normally use. The stress here is not just on the "sequence of blocks" but also on the normal or classical usage of the GPSS language. Generally speaking we have two main "sequences of blocks" in classical GPSS programming for depicting parallel server systems. Although there are certainly many individual variations in recording these blocks, they are almost always based on either the **SELECT MIN** or the **TRANSFER ALL** structure.

These two main types are:

- multiple servers, each with its own waiting queue (Figure 1)
- multiple servers, all with one single waiting queue (Figure 2)

At first sight it seems that both systems are very complex and thus hard to model in the GPSS language. It seems that by attempting this we couldn't avoid many pages of long block sequences contributing to almost completely unclear, complicated and messy programme code. But the boot is on other foot. Lots of tough work and tiresome coding can be saved by simply using indirect addressing (Chisman, 1992). But on the other hand, by using indirect addressing we also loose something. The visual flow of transactions through the system becomes clouded and confused.

The following programme codes show us the possibilities of how to use indirect addressing in classical programming for such parallel systems. Figure 3 represents the GPSS model of a multiple server system where each server has its own waiting queue, while Figure 4 shows a model of a similar system with a single waiting queue.

3 The Graphic Representation of the Problem

If we tried to persuasively demonstrate the functioning of the GPSS models presented in Figure 3 and Figure 4 we would have to establish some requirements first, namely:

- the number of parallel servers in the system
- the service time distribution of each server
- the distribution of transaction time between arrivals into the system.

We can also use data from the real world for the purpose of our research, especially from the computer world. So for the time between the arrivals function and for the service time function we can use the tables published in (Žibert, 1999). For the sake of simplicity let us also assume that all the servers are equivalent in our model (meaning that the service time function is the same for all the parallel servers in the whole system). In this way Figure 5 and Figure 6 show us complemented and developed programmes (based originally on Figure 3 and Figure 4).

```
    REALLOCATE COM,32720
    SIMULATE
*
*   Parallel server system - n servers each with its own waiting queue.
*
SERVER1                 EQU         1,F
SERVER.                 EQU         .,F
SERVERN                 EQU         N,F    n servers
QUEUE1                  EQU         1,Q
QUEUE.                  EQU         .,Q
QUEUEN                  EQU         N,Q    n waiting queues
PROCES1                 EQU         1,Z
PROCES.                 EQU         .,Z
PROCESN                 EQU         N,Z    n process functions
PROCES1                 FUNCTION    RNx,Cx Service time distribution 1
          0,../1,..
PROCES.                 FUNCTION    RNx,Cx Service time distribution .
          0,../1,..
PROCESN                 FUNCTION    RNx,Cx Service time distribution N
          0,../1,..
PRIHOD                  FUNCTION    RNx,Cx Time between arrivals distribution
          0,../1,..
                        GENERATE    FN$PRIHOD
*                       The TRANSFER block determines which entity in the range
                        from
*                       QUEUE1 to QUEUEN has the minimum content and then places
                        the
*                       number of this entity into parameter 1.

                        SELECT MIN  1,QUEUE1,QUEUEN,,Q
                        QUEUE       P1
                        SEIZE       P1
                        DEPART      P1
                        ADVANCE     FN*P1
                        RELEASE     P1
                        TERMINATE   1
*
                        START       xxx    The number of processed transactions
                        END
```

Figure 3: Classical usage of GPSS blocks for modelling a system of n servers, each with its own waiting queue

```
    REALLOCATE COM,32720
    SIMULATE
*
*   Parallel server system - n servers with one single waiting queue.
*
SERVER1                 EQU         1,F
SERVER.                 EQU         .,F
SERVERN                 EQU         N,F
QUEUE1                  EQU         1,Q
PROCES1                 EQU         1,Z
PROCES.                 EQU         .,Z
PROCESN                 EQU         N,Z
PROCES1                 FUNCTION    RNx,Cx Service time distribution 1
          0,../1,..
PROCES.                 FUNCTION    RNx,Cx Service time distribution .
          0,../1,..
PROCESN                 FUNCTION    RNx,Cx Service time distribution N
          0,../1,..
PRIHOD                  FUNCTION    RNx,Cx Time between arrivals distribution
          0,../1,..
*
                        GENERATE    FN$PRIHOD
                        QUEUE       QUEUE1
*The TRANSFER block will see if the engaging transaction can go to the first
*location (BCPU1); if not, it will try to go to the next (BCPU2);if not, then to
*(BCPU3), until it tries the last location (BCPUN). If it cannot send it anywhere,
*it starts all over again, until it can finally move transaction to one of these
*locations.
                        TRANSFER    ALL,BCPU1,BCPUN,3
BCPU1                   SEIZE       1
                        ASSIGN      1,1
                        TRANSFER    ,DALJE
BCPU2                   SEIZE       2
                        ASSIGN      1,2
                        TRANSFER    ,DALJE
BCPU.                   SEIZE       .
                        ASSIGN      1,.
                        TRANSFER    ,DALJE
BCPUN                   SEIZE       N
                        ASSIGN      1,N
DALJE                   DEPART      QUEUE1
                        ADVANCE     FN*P1
                        RELEASE     P1
                        TERMINATE   1
*
                        START       xxx    The number of processed transactions
                        END
```

Figure 4: Classical usage of GPSS blocks for modelling a system of n servers with one single waiting queue

```
    REALLOCATE COM,32720
    SIMULATE
*
*   Parallel server system - n servers each with its own waiting queue.
*
SERVER1            EQU              1,F
SERVER.            EQU              .,F
SERVERN            EQU              N,F
QUEUE1             EQU              1,Q
QUEUE.             EQU              .,Q
QUEUEN             EQU              N,Q
PROCES1            EQU              1,Z
PROCES.            EQU              .,Z
PROCESN            EQU              N,Z
PROCES1            FUNCTION         RN1,C16         Service time distribution 1
                                                    (in seconds)
     0.0000,0.0/0.3867,0.1/0.5693,0.2/0.6829,0.3/0.7604,0.4/0.8117,0.5/
     0.8463,0.6/0.8702,0.7/0.8887,0.8/0.9036,0.9/0.9150,1.0/0.9319,1.2/
     0.9476,1.5/0.9648,2.0/0.9795,3.0/1.0000,5.0

PROCES.            FUNCTION         RN1,C16         Service time distribution .
                                                    (in seconds)
* The same data as above in PROCES1

PROCESN            FUNCTION         RN1,C16         Service time distribution N
                                                    (in seconds)
* The same data as above in PROCES1

FPRIHOD            FUNCTION         AC1,C62         Time between arrivals
                                                    distribution (10 hours)
* The same data as above in PROCES1

VPRIHOD            FVARIABLE        FN$FPRIHOD*(ABS(LOG(1-(RN2/1000))))
                   GENERATE         V$VPRIHOD,,ST   The simulation begins
                                                    at time = ST
                   SELECT MIN       1,QUEUE1,QUEUEN,,Q
                   QUEUE            P1
                   SEIZE            P1
                   DEPART           P1
                   ADVANCE          FN*P1
                   RELEASE          P1
                   TERMINATE
*
                   GENERATE         DT                The simulation lasts DT
                                                      seconds
                   TERMINATE        1
*
                   START            1
                   END
```

Figure 5: The classical model of a system with n servers and n waiting queues that processes statistical data from (Žibert, 1999)

```
    REALLOCATE COM,32720
    SIMULATE
*
*   Parallel server system - n servers with one single waiting queue.
*
SERVER1       EQU          1,F
SERVER.       EQU          .,F
SERVERN       EQU          N,F
QUEUE1        EQU          1,Q
PROCES1       EQU          1,Z
PROCES.       EQU          .,Z
PROCESN       EQU          N,Z
PROCES1       FUNCTION     RN1,C16              Service time distribution 1
                                                (in seconds)
     0.0000,0.0/0.3867,0.1/0.5693,0.2/0.6829,0.3/0.7604,0.4/0.8117,0.5/
     0.8463,0.6/0.8702,0.7/0.8887,0.8/0.9036,0.9/0.9150,1.0/0.9319,1.2/
     0.9476,1.5/0.9648,2.0/0.9795,3.0/1.0000,5.0
PROCES.       FUNCTION     RN1,C16              Service time distribution .
                                                (in seconds)
* The same data as above in PROCES1
PROCESN       FUNCTION     RN1,C16              Service time distribution N
                                                (in seconds)
* The same data as above in PROCES1
FPRIHOD  FUNCTION  AC1,C62 Time between arrivals distribution (10 hours)
* Data for this function are defined in FPRIHOD in Figure 3
VPRIHOD       FVARIABLE    FN$FPRIHOD*(ABS(LOG(1-(RN2/1000))))
              GENERATE     V$VPRIHOD,,ST        The simulation begins at time = ST
              QUEUE        QUEUE1
              TRANSFER     ALL,BCPU1,BCPUN,3
BCPU1         SEIZE        1
              ASSIGN       1,1
              TRANSFER     ,DALJE
BCPU2         SEIZE        2
              ASSIGN       1,2
              TRANSFER     ,DALJE
BCPU.         SEIZE        .
              ASSIGN       1,.
              TRANSFER     ,DALJE
BCPUN         SEIZE        N
              ASSIGN       1,N
DALJE         DEPART       QUEUE1
              ADVANCE      FN*P1
              RELEASE      P1
              TERMINATE
*
              GENERATE     DT                   The simulation lasts DT seconds
              TERMINATE    1
*
              START        1
              END
```

Figure 6: The classical model of a system with n servers and a single waiting queue that processes statistical data from (Žibert, 1999)

Table 1: Average server utilization (column 3) and average queue content (column 5) depending on the number of servers in the model (column 1)

Number of servers	**Average server utilization**		**Average queue content**	
3	SERVER1	0.924	QUEUE1	2.712
	SERVER2	0.832	QUEUE2	2.388
	SERVER3	0.697	QUEUE3	2.143
4	SERVER1	0.874	QUEUE1	0.932
	SERVER2	0.740	QUEUE2	0.738
	SERVER3	0.529	QUEUE3	0.506
	SERVER4	0.310	QUEUE4	0.307
5	SERVER1	0.865	QUEUE1	0.705
	SERVER2	0.722	QUEUE2	0.527
	SERVER3	0.497	QUEUE3	0.325
	SERVER4	0.259	QUEUE4	0.158
	SERVER5	0.110	QUEUE5	0.065
6	SERVER1	0.865	QUEUE1	0.657
	SERVER2	0.713	QUEUE2	0.495
	SERVER3	0.495	QUEUE3	0.308
	SERVER4	0.254	QUEUE4	0.144
	SERVER5	0.097	QUEUE5	0.044
	SERVER6	0.030	QUEUE6	0.012

Table 2: The number of transactions (column 3) and their percentages (column 4) passed through the individual servers (column 2) in the model with N (column 1) parallel servers

Number of servers	**Server**	**Number of transactions**	**Percentage [%]**
3	SERVER1	8.587	0.378
	SERVER2	7.729	0.341
	SERVER3	6.366	0.281
	SUM	22.682	1.000
4	SERVER1	8.189	0.361
	SERVER2	6.658	0.294
	SERVER3	4.887	0.215
	SERVER4	2.949	0.130
	SUM	22.683	1.000
5	SERVER1	8.120	0.358
	SERVER2	6.372	0.281
	SERVER3	4.625	0.204
	SERVER4	2.501	0.110
	SERVER5	1.065	0.047
	SUM	22.683	1.000
6	SERVER1	8.032	0.355
	SERVER2	6.430	0.283
	SERVER3	4.516	0.199
	SERVER4	2.436	0.107
	SERVER5	999	0.044
	SERVER6	270	0.012
	SUM	22.683	1.000

Having enhanced our models with real data, we are now able to carry out the series of simulations. In each simulation we can apply some modifications, such as the number of concurrent servers (N), the starting time (defined by letter Z in our programme), the duration of the simulation (defined by letter Y in our programme) etc.

First we can try using the model from Figure 5. For testing purposes we can accept the following parameters:

- the starting time is zero (ST = 0)
- the duration time is one hour (DT = 3600)
- the number of servers is increasing from the minimum to the maximum reasonable number (meaning that there are at least certain number of servers with the attention of avoiding queues that are too long in the model and that all N servers in our model have some traffic -min. <= N =< max.).

The results are represented in Table 1.

As we can see from the table above, the utilization of the servers in the first few positions in the programme code (SERVER1 and SERVER2) is quite high – considerably higher in comparison with the servers positioned at the end of the code (SERVER4, SERVER5 and SERVER6). Furthermore, we can't fail to observe that the utilization of these same servers (SERVER1 and SERVER2) is not changed much by adding additional servers in the model. This can also be seen by looking at the number of transactions (and their percentages) passed through the individual servers (Table 2).

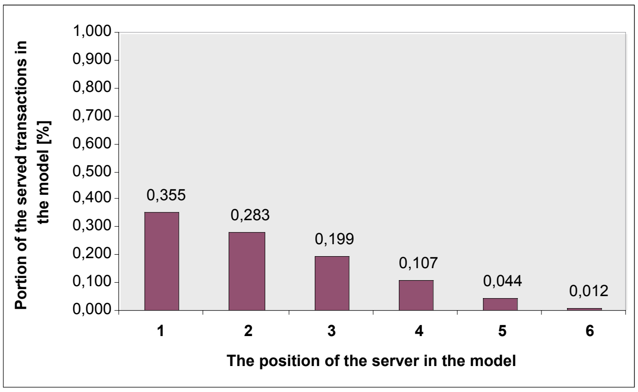


Figure 7: The distribution of the service in the classical GPSS model with six parallel servers

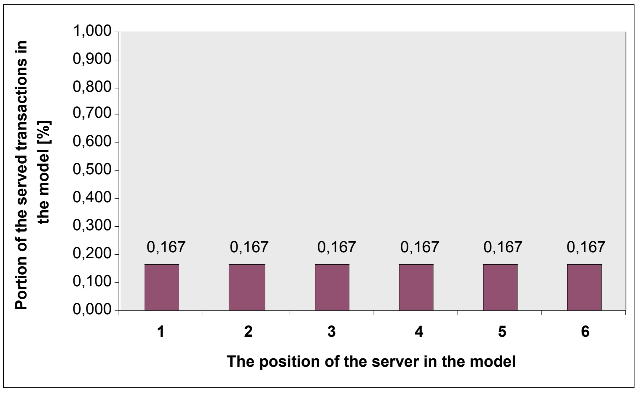


Figure 8: The expected distribution of the service in the classical GPSS model with 6 servers

The same can be seen more obviously in a graphic way, especially for the model with six parallel servers (Figure 7). Here we can clearly observe the declining trend of the server utilization. Thus, the first server in the model executes almost 36 percent of all the completed transactions and the sixth server executes only one and if we expanded our model by adding some new servers then they would be completely idle.

Of course, considering the service in random order, which was our presumption, we would expect that the above graph would be quite different and similar to Figure 8.

But what would we get if the traffic in the model (the number of entering transactions) diminished rapidly? Let's now change our GPSS programme from Figure 6 (the model with n parallel servers and with a single waiting queue) as follows:

- the starting time ST = 33000 (though the density of the transaction arrivals is much lower)
- the duration time is again one hour, DT = 3600
- the number of server is increased from 2 to 6 (2 <= N =< 6).

Table 3: The number of transactions (column 3) and their percentages (column 4) that passed through the individual servers (column 2) in the model with N parallel servers (column 1) during conditions of low traffic density

Number of servers	Server	Number of transactions	Percentage [%]
2	SERVER1	449	0.947
	SERVER	025	0.053
	SUM	474	1.000
3	SERVER1	449	0.947
	SERVER2	025	0.053
	SERVER3	000	0.000
	SUM	474	1.000
4	SERVER1	449	0.947
	SERVER2	025	0.053
	SERVER3	000	0.000
	SERVER4	000	0.000
	SUM	474	1.000
5	SERVER1	449	0.947
	SERVER2	025	0.053
	SERVER3	000	0.000
	SERVER4	000	0.000
	SERVER5	000	0.000
	SUM	474	1.000
6	SERVER1	449	0.947
	SERVER2	025	0.053
	SERVER3	000	0.000
	SERVER4	000	0.000
	SERVER5	000	0.000
	SERVER6	000	0.000
	SUM	474	1.000

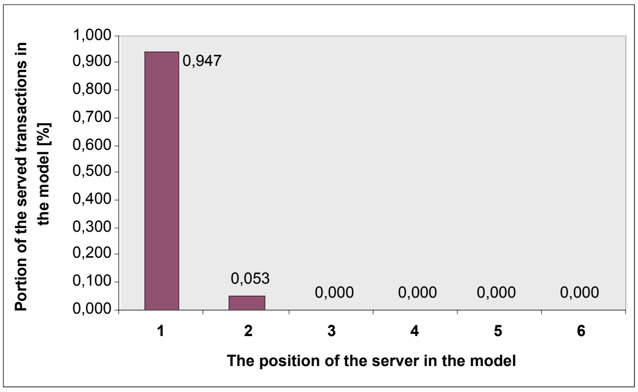


Figure 9: The distribution of the service in the classical GPSS model (with six parallel servers) on condition of low traffic density

The results are presented in Table 3 and in Figure 9 for the model with six servers.

Straight away we can see that the model shows quite unrealistic situation during conditions of low traffic density (and by that also a low occupation rate per server). Barely more than one server is utilized in the model (i. e. the first one). In our case it is (only by chance) fully loaded while all the other servers are practically unattached. This example clearly demonstrates the discrepancy of the model and its dependence on the occupation rate per server explained earlier. Hence it follows that each and every possible solution should be proved under the same conditions – i.e. models with a low occupation rate per server.

4 The Solution of the Problem

In the previous chapters we established and proved that classical (ordinary) usage of GPSS blocks in modelling doesn't take into consideration the principle of service in random order. So whenever we model a system in GPSS that operates in this way we always come up against difficulties. However, in spite of everything, this principle can be achieved. Taking into account that there are two main types of parallel service (described in Figure 1 and Figure 2) we will also offer two different solutions for each type.

For the first type (the systems containing n servers each with its own queue) this difficult task could be tackled in the following way. At first the GPSS programme determines the length of the shortest queue in the system (**SELECT MIN**). Then it randomly (variable **VAR1**) chooses one of the feasible queues (**ASSIGN**) and compares its length with the length of the shortest one (**TEST E**). If both lengths are equal then the transaction is normally sent to the randomly chosen queue (**QUEUE**). Otherwise the programme picks out another waiting queue (the execution of the programme returns to label "**PONOVNO**"). Figure 10 shows the principle part of the GPSS programme explained above.

Our sample programme as a whole, upgraded using the method described, would look like that shown in Figure 11.

```
VAR1          VARIABLE       ((RN3*N/1000)+1)
              GENERATE       ....
              SELECT MIN     1,QUEUE1,QUEUEN„Q
PONOVNO       ASSIGN         2,V$VAR1
              TEST           E                    Q*P1,Q*P2,PONOVNO
              QUEUE                               P2
```

Figure 10: The section of the GPSS programme that solves the problem in a model with n servers and n queues

```
    REALLOCATE COM,32720
    SIMULATE
*
*   Parallel server system - n servers each with its own waiting queue.
*   The upgraded variant
*
SERVER1       EQU            1,F
SERVER.       EQU            .,F
SERVERN       EQU            N,F
QUEUE1        EQU            1,Q
QUEUE.        EQU            .,Q
QUEUEN        EQU            N,Q
PROCES1       EQU            1,Z
PROCES.       EQU            .,Z
PROCESN       EQU            N,Z
PROCES1       FUNCTION       RN1,C16              Service time distribution 1 (in seconds)
  0.0000,0.0/0.3867,0.1/0.5693,0.2/0.6829,0.3/0.7604,0.4/0.8117,0.5/
  0.8463,0.6/0.8702,0.7/0.8887,0.8/0.9036,0.9/0.9150,1.0/0.9319,1.2/
  0.9476,1.5/0.9648,2.0/0.9795,3.0/1.0000,5.0
PROCES.       FUNCTION       RN1,C16              Service time distribution . (in seconds)
* The same data as above in PROCES1
PROCESN       FUNCTION       RN1,C16              Service time distribution N (in seconds)
* The same data as above in PROCES1
FPRIHOD       FUNCTION       AC1,C62              Time between arrivals distribution (10 hours)
* Data for this function are defined in FPRIHOD in Figure 3
VPRIHOD       FVARIABLE      FN$FPRIHOD*(ABS(LOG(1-(RN2/1000))))
              INITIAL        X$SERVNUM,N          Number of servers = N
VAR1          VARIABLE       ((RN3*X$SERVNUM/1000)+1)  A randomly chosen queue
              GENERATE       V$VPRIHOD,,ST        The simulation begins at time = ST
              SELECT MIN     1,QUEUE1,QUEUEN,,Q
PONOVNO       ASSIGN         2,V$VAR1
              TEST  E        Q*P1,Q*P2,PONOVNO
              QUEUE          P2
              SEIZE          P2
              DEPART         P2
              ADVANCE        FN*P2
              RELEASE        P2
              TERMINATE
*
              GENERATE       DT                   The simulation lasts DT seconds
              TERMINATE      1
*
              START          1
              END
```

Figure 11: Our upgraded model of a system with n servers and n waiting queues that processes the same statistical data as the programme in Figure 5

Table 4: The number of transactions (column 3) and their percentages (column 4) passed through the individual servers (column 2) in the upgraded model with N (column 1) parallel servers

Number of servers	Server	Number of transactions	Percentage [%]
2	SERVER1	225	0.475
	SERVER	249	0.525
	SUM	474	1.000
3	SERVER1	152	0.321
	SERVER2	154	0.325
	SERVER3	168	0.354
	SUM	474	1.000
4	SERVER1	114	0.241
	SERVER2	111	0.234
	SERVER3	121	0.255
	SERVER4	128	0.270
	SUM	474	1.000
5	SERVER1	84	0.177
	SERVER2	96	0.203
	SERVER3	95	0.200
	SERVER4	93	0.196
	SERVER5	106	0.224
	SUM	474	1.000
6	SERVER1	71	0.150
	SERVER2	81	0.171
	SERVER3	73	0.154
	SERVER4	81	0.171
	SERVER5	78	0.164
	SERVER6	90	0.190
	SUM	474	1.000

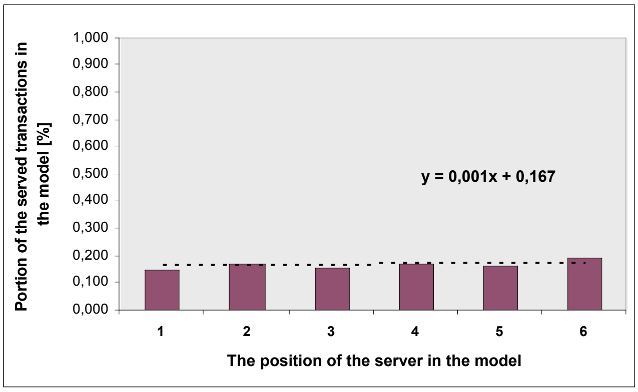


Figure 12: The distribution of the service in the upgraded GPSS model with a dotted trend line

When the upgraded GPSS programme is executed under the same conditions as before (the starting time ST = 33000, the duration time DT = 3600 and the number of servers ranging between 2 and 6) we get the following simulation results (Table 4) and the following graph for a model with six servers (Figure 12).

Right away we can recognize that the behaviour of the model is quite different to that in all the earlier cases. As we can see, each server now seems to complete approximately the same percentage of the incoming transactions so the workload in our upgraded model quite realistically seems to be evenly distributed among all of the servers in the system. That hypothesis was even statistically confirmed using the chi-squared test in (Žibert, 2005).

For the parallel service model using a single waiting queue (earlier defined as the second type) it is much harder to find a solution. The classic GPSS system uses a long sequence of blocks for this purpose. Thus in our sample we controlled the flow of a transaction within the blocks declared in the transfer block (TRANSFER ALL, BCPU1, BCPUN, 3). That means that we controlled it all the way from the **TRANSFER ALL** block to the block labelled **BCPUN** plus three additional subsequent blocks. It seems that all the blocks in between form an indivisible entity where randomness of any kind can not be taken into account.

```
INITIAL        X$CPUNUM,N                          Number of servers = N
VAR1           VARIABLE       ((RN3*X$CPUNUM/1000)+1)
               GENERATE       ...
               QUEUE          QUEUE1
               ASSIGN         1,X$CPUNUM-1
               ASSIGN         2,V$VAR1
PONOVNO        TEST L         P(X$CPUNUM-P1+1),X$CPUNUM,ZACETEK
               ASSIGN         (X$CPUNUM-P1+2),P(X$CPUNUM-P1+1)+1
               TRANSFER       ,ZANKA
ZACETEK        ASSIGN         (X$CPUNUM-P1+2),1
               TRANSFER       ,ZANKA
ZANKA          LOOP           1,PONOVNO
```

Figure 13: The complex section of the GPSS programme that solves the problem in a model with n servers and a single queue

However, the problem here can be also grappled with. By applying indirect addressing in the **SEIZE** blocks we could always use one of the transaction parameters (**P1**, **P2**, **P3**, etc). That means that the transaction occupies the facility that is coded in that parameter. Thus, if we changed the contents of all those parameters belonging to the transaction at the time of its birth (generation), we would

```
REALLOCATE    COM,32720
    SIMULATE
*
*   Parallel server system - n servers with a single waiting queue.
*   The upgraded variant
*
SERVER1       EQU           1,F
SERVER.       EQU           .,F
SERVERN       EQU           N,F
QUEUE1        EQU           1,Q
PROCES1       EQU           1,Z
PROCES.       EQU           .,Z
PROCESN       EQU           N,Z
PROCES1       FUNCTION      RN1,C16    Service time distribution 1 (in seconds)
     0.0000,0.0/0.3867,0.1/0.5693,0.2/0.6829,0.3/0.7604,0.4/0.8117,0.5/
     0.8463,0.6/0.8702,0.7/0.8887,0.8/0.9036,0.9/0.9150,1.0/0.9319,1.2/
     0.9476,1.5/0.9648,2.0/0.9795,3.0/1.0000,5.0
PROCES.       FUNCTION      RN1,C16    Service time distribution . (in seconds)
* The same data as above in PROCES1
PROCESN       FUNCTION      RN1,C16    Service time distribution N (in seconds)
* The same data as above in PROCES1
FPRIHOD       FUNCTION      AC1,C62  Time between arrivals distribution (10 hours)
* Data for this function are defined in FPRIHOD in Figure 3
VPRIHOD       FVARIABLE     FN$FPRIHOD*(ABS(LOG(1-(RN2/1000))))
              INITIAL       X$CPUNUM,N Number of servers = N
VAR1          VARIABLE      ((RN3*X$CPUNUM/1000)+1)   A random number from 1 to N
              GENERATE      V$VPRIHOD,,ST The simulation begins at time = ST
              QUEUE         QUEUE1
*   The start of filling our parameter table
              ASSIGN        1,X$CPUNUM-1
              ASSIGN        2,V$VAR1
PONOVNO       TEST     L    P(X$CPUNUM-P1+1),X$CPUNUM,ZACETEK
              ASSIGN        (X$CPUNUM-P1+2),P(X$CPUNUM-P1+1)+1
              TRANSFER      ,ZANKA
ZACETEK       ASSIGN        (X$CPUNUM-P1+2),1
              TRANSFER      ,ZANKA
ZANKA         LOOP          1,PONOVNO
*   The end of filling our parameter table
              TRANSFER      ALL,BCPU1,BCPUN,5
BCPU1         SEIZE         P2
              DEPART        QUEUE1
              ADVANCE       FN*P2
              RELEASE       P2
              TRANSFER      ,DALJE
BCPU.         SEIZE         P.
              DEPART        QUEUE1
              ADVANCE       FN*P.
              RELEASE       P.
              TRANSFER      ,DALJE
BCPUN         SEIZE         P(N+1)
              DEPART        QUEUE1
              ADVANCE       FN*P(N+1)
              RELEASE       P(N+1)
DALJE                       TERMINATE
*
              GENERATE      DT         The simulation lasts DT seconds
              TERMINATE     1
*
              START         1
              END
```

Figure 14: Our upgraded model of a system with n servers and a single waiting queue that processes the same statistical data as the programme from Figure 6

Table 5: The number of transactions (column 3) and their percentages (column 4) passed through the individual servers (column 2) in the upgraded model with N (column 1) parallel servers

Number of servers	**Server**	**Number of transactions**	**Percentage [%]**
2	SERVER1	226	0.477
	SERVER	248	0.523
	SUM	474	1.000
3	SERVER1	155	0.327
	SERVER2	155	0.327
	SERVER3	164	0.346
	SUM	474	1.000
4	SERVER1	117	0.247
	SERVER2	111	0.234
	SERVER3	122	0.257
	SERVER4	124	0.262
	SUM	474	1.000
5	SERVER1	87	0.183
	SERVER2	96	0.203
	SERVER3	96	0.203
	SERVER4	93	0.196
	SERVER5	102	0.215
	SUM	474	1.000
6	SERVER1	73	0.154
	SERVER2	82	0.173
	SERVER3	73	0.154
	SERVER4	81	0.171
	SERVER5	78	0.165
	SERVER6	87	0.183
	SUM	474	1.000

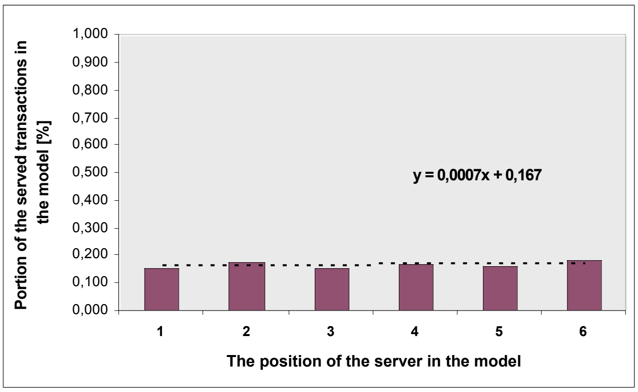


Figure 15: The distribution of the services in the upgraded GPSS model with a dotted trend line

also change all the facilities at hand in the **SEIZE** blocks. All we must do is to create a table of parameters for each generated transaction and fill it randomly with the numbers that represent the appointed facility. Figure 13 shows us how to do it.

This time, our complete sample programme, upgraded using the method described, would look like the that in Figure 14.

The results of the above GPSS simulation model (under the same condition as earlier, with the starting time ST = 33000, the duration time DT = 3600 and the number of servers ranging between 2 and 6) are once again presented as a table (Table 5) and as a graph for a model with six servers (Figure 15).

As before, we can perceive that the servers in the model are treated approximately much the same. So the workload in this upgraded programme could also be considered as evenly distributed among all of the servers (Žibert, 2005).

```
*
*   The definitions of macro IZBIRAQ for systems with n parallel
*   servers and n waiting queues
*
*   Macro parameters:
*   #A - the number of parallel servers - N
*   #B - random number (1 - 9)
*
*   The exit is parameter 2.
*
IZBIRAQ     STARTMACRO  #A,#B
VAR1        VARIABLE    ((#B*(#A+1-P1)/1000)+P1)
            SELECT MIN  1,QUEUE1,#A,,Q
PONOVNO     ASSIGN      2,V$VAR1
            TEST  E     Q*P1,Q*P2,PONOVNO
            ENDMACRO
*
*   The end of macro IZBIRAQ
```

Figure 16: The IZBIRAQ macro for models with n parallel servers and n waiting queues

5 Practical Forms of our Solution

Perhaps it would be helpful for many of us if we also introduced our solution in a rather different form – using so-called macros. This would make the solution more common, even user friendly and (we hope) more applicable. Macros are not just used to be easily and repeatedly called from every possible point inside the programme. They are also applied to shorten large source codes and to make them much easier to understand.

In this way, we present the GPSS macros for the models in Figure 16 and Figure 17.

In next figures (Figure 18 and Figure 19), there are two simple samples of how the above two macros can be used, so the readers can learn by examples.

```
*
*   The definition of macro IZBIRAF for systems with n parallel
*   servers and a single waiting queue
*
*   Macro parameters:
*   #A - the number of parallel servers - N
*   #B - random number (1 - 9)
*
*   The exit is a table of parameters from P2 to PN.
*
IZBIRAF       STARTMACRO     #A,#B
VAR1          VARIABLE       ((#B*#A/1000)+1)
              ASSIGN         1,#A-1
              ASSIGN         2,V$VAR1
PONOVNO       TEST    L      P(#A-P1+1),#A,ZACETEK
              ASSIGN         (#A-P1+2),P(#A-P1+1)+1
              TRANSFER       ,ZANKA
ZACETEK       ASSIGN         (#A-P1+2),1
              TRANSFER       ,ZANKA
ZANKA         LOOP            1,PONOVNO
              ENDMACRO
*
*             The end of macro IZBIRAF
```

Figure 17: The IZBIRAF macro for models with n parallel servers and a single waiting queue

```
    SIMULATE
*
*   System with 5 servers, each with its own waiting queue.
*   The programme calls macro IZBIRAQ.
*
CPU1          EQU           1,F
CPU2          EQU           2,F
CPU3          EQU           3,F
CPU4          EQU           4,F
CPU5          EQU           5,F
QUEUE1        EQU           1,Q
QUEUE2        EQU           2,Q
QUEUE3        EQU           3,Q
QUEUE4        EQU           4,Q
QUEUE5        EQU           5,Q
PROCES1       EQU           1,Z
PROCES2       EQU           2,Z
PROCES3       EQU           3,Z
PROCES4       EQU           4,Z
PROCES5       EQU           5,Z
PROCES1       FUNCTION      RN1,C2
       0,1/1,1
PROCES2       FUNCTION      RN2,C2
       0,1/1,1
PROCES3       FUNCTION      RN3,C2
       0,1/1,1
PROCES4       FUNCTION      RN4,C2
       0,1/1,1
PROCES5       FUNCTION      RN5,C2
       0,1/1,1
              INITIAL       X$CPUNUM,5
*
*             The definition of macro IZBIRAQ
*
IZBIRAQ       STARTMACRO    #A,#B
VAR1          VARIABLE      ((#B*(#A+1-P1)/1000)+P1)
              SELECT MIN    1,QUEUE1,#A,,Q
PONOVNO       ASSIGN        2,V$VAR1
              TEST   E      Q*P1,Q*P2,PONOVNO
              ENDMACRO
*
*             The end of the macro
*
*
*             The main programme
*
              GENERATE      0.6,0.5,,,,27
IZBIRAQ       MACRO         X$CPUNUM,RN6
              QUEUE         P2
              SEIZE         P2
              DEPART        P2
              ADVANCE       FN*P2
              RELEASE       P2
              TERMINATE     1
*
              START         10000
              END
```

Figure 18: A simple programme showing how to use the IZBIRAQ macro

```
    SIMULATE
*
*   System with 3 servers and one waiting queue.
*   The programme calls macro IZBIRAF.
*
SERVER1         EQU             1,F
SERVER2         EQU             2,F
SERVER3         EQU             3,F
QUEUE1          EQU             1,Q
PROCES1         EQU             1,Z
PROCES2         EQU             2,Z
PROCES3         EQU             3,Z
PROCES1         FUNCTION        RN1,C2
       0,1/1,1
PROCES2         FUNCTION        RN2,C2
       0,1/1,1
PROCES3         FUNCTION        RN3,C2
       0,1/1,1
FPRIHOD         FUNCTION   AC1,C7
     00000,1.2/00600,1.4/01200,0.9/01800,0.8/02400,0.75/03000,0.9/03600,1.2
VPRIHOD         FVARIABLE       FN$FPRIHOD*(ABS(LOG(1-(RN2/1000))))
                INITIAL         X$CPUNUM,3
*
*               The start of macro IZBIRAF
*
IZBIRAF         STARTMACRO      #A,#B
VAR1VARIABLE    ((#B*#A/1000)+1)
                ASSIGN          1,#A-1
                ASSIGN          2,V$VAR1
PONOVNO         TEST   L        P(#A-P1+1),#A,ZACETEK
                ASSIGN          (#A-P1+2),P(#A-P1+1)+1
                TRANSFER        ,ZANKA
ZACETEK         ASSIGN          (#A-P1+2),1
                TRANSFER        ,ZANKA
ZANKA           LOOP              1,PONOVNO
                ENDMACRO
*
*               The end of the macro
*
*               The main programme
*
                GENERATE        V$VPRIHOD
                QUEUE           QUEUE1
IZBIRAF         MACRO           X$CPUNUM,RN4
                TRANSFER        ALL,BCPU1,BCPU3,5
BCPU1           SEIZE           P2
                DEPART          QUEUE1
                ADVANCE         FN*P2
                RELEASE         P2
                TRANSFER        ,DALJE
BCPU2           SEIZE           P3
                DEPART          QUEUE1
                ADVANCE         FN*P3
                RELEASE         P3
                TRANSFER        ,DALJE
BCPU3           SEIZE           P4
                DEPART          QUEUE1
                ADVANCE         FN*P4
                RELEASE         P4
DALJE           TERMINATE
*
                GENERATE        3600
                TERMINATE       1
*
                START           1
                END
```

Figure 19: A simple programme showing how to use the IZBIRAF macro

6 Conclusions

As we said in chapter one when describing the problem, the GPSS modelling of parallel server systems with the service in random order always causes some problems. The final effect of these troubles (and our problem) is more or less presented as a discrepancy between the model and the real system. In some extreme cases the discrepancy could be so large that we are talking about an inadequate model.

In this article we tried to show a methodology of how to surmount the obstacles represented in this simulation language and how to correctly model some prevailing types of parallel service systems. The suggested solution is mainly composed of some additional control statements (as declarations at the beginning of the GPSS programme) and an extra section of block sequence that randomly chooses one of the suitable entities in the model. To make the solution more applicable among users we also went a step further. In this respect, we made it easily available as a macro called from the main programme with some additional parameters.

As presented in the article (especially in the graphs), the solution successfully simulates the behaviour of parallel service systems with the service in random order. Without using it, the model would describe the same system but with different type of service order. In this case it would represent a system with the service in order of precedence, which is immanent to GPSS.

Literature

Adan, I. & Resing, J. (2001). *Queueing Theory*, Department of Mathematics and Computing Science, Eindhoven University of Technology.

Azar, Y., Broder, A., Karlin, A. & Upfal, E. (1999). Balanced Allocations, *SIAM Journal on Computing,* 29(1): 180-200.

Blake, K. & Gordon G. (1964). System simulation with digital computers, *IBM Systems Journal*, **3**(1):14 – 20.

Chisman, A. J. (1992). *Introduction to Simulation Modeling Using GPSS/PC, Prentice-Hall International, Englewood Cliffs.*

Cicsplex SM Concepts and Planning (1999), International Business Machines Corporation, New York.

Crain, C. R. (1997). Simulation using GPSS/H, Proceedings of the 1997 Winter Simulation Conference, Uredili: Andradbttir, S., Healy, K. J., Withers, D. H. & Nelson, B. L. Atlanta 7-10 dec. 1997. Piscataway: Institute of Electronics and Electrical Engineers.

Crain, C. R. (1998). Simulation using GPSS/H, Proceedings of the 1998 Winter Simulation Conference, Edited by Medeiros, D. J., Watson, E. F., Carson, J. S., Manivannan, M. S. Washington 13-16 dec. 1998. Piscataway: Institute of Electronics and Electrical Engineers.

Crain, C. R. & Henriksen, J. O. (1999). Simulation using GPSS/H, *Proceedings of the 1999 Winter Simulation Conference, Edited by Farrington, P. A., Nembhard, H. B., Sturrock, D. T. & Evans, G. W. Phoenix 5-8 dec. 1999. Phoenix: Institute of Electronics and Electrical Engineers.*

Gordon G. (1962). A general purpose systems simulator, IBM Systems Journal, **3(1):18-32.**

Guariso, G., Hitz, M. & Werthner, H. (1996). An integrated simulation and optimization modelling environment for decision support, *Decision Support Systems*, **16**(2):103 – 117.

Henriksen, J. O. & Crain, C. R. (2000). GPSS/H: A 23 – year retrospective view, *Proceedings of the 2000 Winter Simulation Conference*, Edited by Joines, J. A., Barton, R. R., Kang, K. & Fishwick, P. A. Orlando 10-13 dec. 2000, Piscataway: Institute of Electronics and Electrical Engineers.

Katwijk, J. & Zalewski, J. (1999). Parallel and Distributed Real-Time Systems: An Introduction, *Parallel and Distributed Computing Practices*, **2**(1):1-6.

Mitzenmacher, M. (1999). On the Analysis of Randomized Load Balancing Schemes, *Theory of Computing Systems*, **32**(3): 361-386.

Murata, T. (1989). Petri nets: Properties, analysis and applications, *Proceedings of the IEEE*, **77**(4): 540– 581.

Rosenthal, J. S. (2000). Parallel computing and Monte Carlo algorithms, *Far East Journal of Theoretical Statistics*, **4**(2): 207-236.

Sang, J., Chung, K. & Rego, V. (1994). A Simulation Testbed based on Lightweight Processes, *Software: Practice and Experience*, **24**(5): 485-505.

Song, B., Ernemann, C. & Yahyapour, R. (2004). Parallel Computer Workload Modeling with Markov Chains, *Proceedings of Job Scheduling Strategies for Parallel Processing: 10th International Workshop*, edited by Feitelson, U. D., Rudolph, L. & Schwiegelshohn, U. New York 13 jun. 2004. Berlin: Springer Verlag.

Wooldridge, M. & Jennings, N. (1995). Intelligent Agents: Theory and Practice, *Knowledge Engineering Review*, 10(2): 115-147.

Žibert, M. (1999). Simulacija delovanja CICS-a na osrednjem računalniku na Zavodu za zdravstveno zavarovanje Slovenije (in Slovenian), Diploma Assignment, University of Maribor, Faculty of Organizational Science.

Žibert, M. (2005). Simulacija delovanja paralelnih enojnih strežnikov v GPSS-u, Master Thesis, University of Maribor, Faculty of Organizational Science.

Marko Žibert received his Master degree in information systems management at the University of Maribor, Faculty of Organizational Science, in 2005. His major research interest includes discrete simulation for the optimization of various information systems. He is employed as a system programmer in the System Department of the Health Insurance Institute of Slovenia (ZZZS).

Miroljub Kljajić is Professor at the Faculty of Organizational Sciences, University of Maribor. His brief biography is published on page 634 of this issue.

Discrete Event Passenger Flow Simulation Model for an Airport Terminal Capacity Analysis

Robert Rauch[1], **Miroljub Kljajić**[2]

[1]Aerodrom Ljubljana, d.d., Zg. Brnik 130 a, 4210 Brnik, Slovenija, robert.rauch@lju-airport.si
[2]Faculty of Organizational Sciences, University of Maribor, Kidričeva cesta 55, 4000 Kranj, Slovenia
miroljub.kljajic@fov.uni-mb.si

This paper describes analysis of departure passenger flow in an airport terminal, from the passenger entrance to boarding; involving the development of simulation model. The basis for the simulation model was a snapshot of the passenger flow data taken during the summer periods from 2003 to 2004. Various data were collected and used to define the inputs to a simulation model. The next step was a statistical analysis of the collected data. A discrete-event simulation model using simulation programming language General Purpose Simulation System (GPSS) was constructed. The performance of the system in the present and expected future was studied. The simulation model helped us to evaluate the passenger flow, identify the system bottlenecks as well as the system capacities. Critical aspects in the passenger

Key words: modeling, discrete event simulation, airport, passenger flow, GPSS

Dogodkovni model dinamičnih potniških tokov za potrebe analize kapacitet v letališkem terminalu

V prispevku smo analizirali karakteristike dinamičnih potniških tokov na letališču Ljubljana. Z ozirom na kompleksnost problema smo se osredotočili na študij funkcionalnosti strežbe odhodnega potniškega toka in sicer od prihoda potnikov na letališče do vkrcanja potnikov v letalo. Osnova za izgradnjo prototipnega simulacijskega modela so bili zbrani podatki v času posameznih dnevnih prometnih konic v poletnih mesecih v letih 2003 in 2004. Na osnovi statističnih analiz zbranih podatkov smo dobili statistične porazdelitve medprihodnih časov prihodov in časov strežbe potnikov, ki so vstopili v letališki sistem. Dogodkovni simulacijski model smo razvili z uporabo simulacijskega jezika General Purpose Simulation System (GPSS). Z eksperimentiranjem na simulacijskem modelu smo proučevali dinamiko obstoječih in bodočih potniških tokov na letališču, identificirali ozka grla ter pomanjkanje tako obstoječih kot tudi bodočih kapacitet. Na podlagi dobljenih simulacijskih rezultatov so podani konkretni predlogi za izboljšave.

Ključne besede: modeliranje, dogodkovna simulacija, letališče, potniški tokovi, GPSS

---

1 Introduction

An airport is an operational system comprising of a framework of infrastructure, facilities, equipment, systems and personnel which collectively provide a service to a customer. Fast growth in airline passenger traffic and, on the other hand, slow expansion of airport capacity, is straining the ability of airports to maintain satisfactory customer service. The complexity of the logistical process of an airport is huge and many airports are faced with operational efficiency problems. Airports are nowadays struggling with increasing numbers of passengers with strong variations in processing time, shorter transfer connection times, environmental and noise limitation, increased security (baggage screening, biometrics, etc.), pressuring them to become more efficient. Widespread increases in queuing and processing times are well-documented frustration for airports, airlines and passengers.

Although passengers acknowledge the need for increased security (especially post 9-11 event), delayed boarding, cancelled flights, long waits have created an environment of passenger dissatisfaction. The urgent need to better use assets, handle more flights, coordinate schedule and quickly respond to delays confronts many airports which are trying to meet passenger needs by improving overall operational flexibility and increasing customer services. Interested study of Airport Operational Efficiency principles, characteristics, regulating and measuring it, was done by International Civil Aviation Organization (Vreedenburgh, 1996). Operational efficiency at an airport can have a direct impact on safety, user and customer satisfaction and also at the financial performance of the airport (Durante, 2003). The level of communication between airports, airlines and handlers can also be a problem to solve (Verougstraete, 2001). Airports do all possible to get the passengers from the entrance point near check-in to the gate and the aircraft as smoothly as can be. Several studies have focused on the most important customers of the airport, the passengers. Aviation industry is focused on simple processes that bring value to the passenger; the same passenger will feel as a welcome guest.

Deciding what mix of resources, processes and technologies will deliver the best combination in improved customer service is very difficult. There are studies, which are dealing with the airport capacity problem from the mathematical point of view (Usenik and Radačič, 1997). Using analytical methods, queuing time can be hardly accurately predicted. The most important reason is the peaks in arrival patterns. Better approach is simulation, which offers additional advantages like flexible modeling and animation. Simulation is a process of building a model of a system and conducting experiments on that system to determine the performance of the system under varying conditions. Some international (big) airports (e.g. Amsterdam Schiphol, Vienna Schwechat, Istanbul Ataturk, Washington Dulles, etc.) have done some extensive simulation studies in this area. Regardless of all contemporary (and not so cheap) software tools in the market (Arena, Incontrol ED, Simul8, ProModel, Simmod, etc.) and some very high specialized companies such as Incontrol Enterprise Dynamics, Kiran Consulting Group and others, there is still place for improvements, especially from the »small« airports perspective (such as Airport Ljubljana), considering the »cost«.

In 2004, Ljubljana Airport has, for the first time in the airport's history, served its millionth passenger in a single year. This is a round number that has great symbolic value for the airport and obligates it to undertake new development projects in the future. In 2005, more than 1.2 million passengers used the airport. Annual growth in the past few years is above 10 percent (Aerodrom Ljubljana, 2006).
With the Slovenia's accession to the EU in 2004 and consecutive application of the Schengen standards, five new categories of passenger traffic will be introduced:

- International departures (existing passenger flow),
- International arrivals (existing passenger flow),
- Transfer: International arrivals - International departures (existing passenger flow),
- Departures: Schengen traffic (new passenger flow),
- Arrivals: Schengen traffic (new passenger flow),
- Transfer: Schengen arrivals - Non Schengen departures (new passenger flow),
- Transfer: Non Schengen arrivals - Schengen departures (new passenger flow),
- Transfer: Schengen arrivals - Schengen departures (new passenger flow).

A Schengen standard means providing an appropriate airport infrastructure for inspecting passengers from the »Schengen« area to the »Non Schengen« area. In order to provide an analysis of existing and future airport facilities, concerning that real life testing are expensive or not (yet) possible, we decided to investigate computer simulation modeling techniques as a possible solution to the problem. Main reason, comparing the mathematical models, is flexibility and versatility. Simulation has been proven very valuable in airport passenger logistic to study bottlenecks and test potential solutions (Kukulich & Leone, 2002; de Ruiter, 2002; Park & Ahn, 2003).

2 Simulation objectives

The primarily objectives of the simulation modeling were to identify the bottlenecks and evaluate the airport system alternatives during peak day operations. Other objectives were: a dynamic analysis of the existing system under current and projected levels of activity, find solutions from which can we benefit most from, support for making decisions about future airport developments and hopefully, continuous use of the model for »what-if« scenarios.

3 Simulation model

3.1 Data collection

Correct data is essential to get valid and valuable results about bottlenecks and to define relevant scenarios. Simulation studies require an exact description of processes and representative data. Airport processes involved with passenger handling have been analyzed and the numbers of resources estimated. A large amount of information had to be collected and laid down. Process times, waiting times, and queue lengths have been measured and collected statistics has been analyzed. Annual growth in the airport Ljubljana in the past few years is above 10 percent (Fig. 1). The basis for the simulation model was a snapshot of the passenger flow data, taken during the summer periods between 2003 and 2004 (Fig. 2).

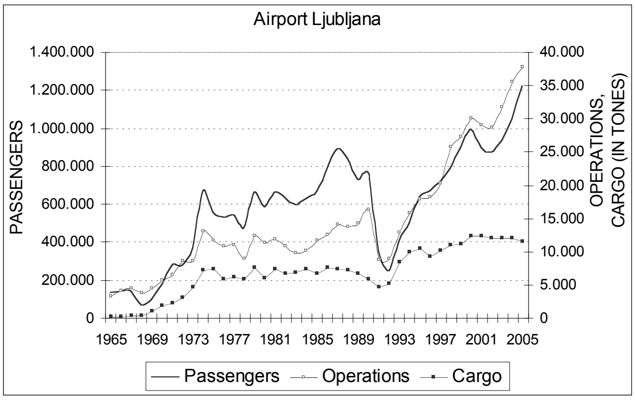


Figure 1: Annual traffic figures (1964-2005)

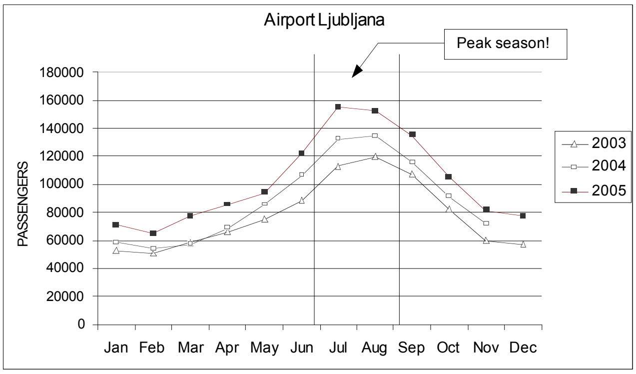


Figure 2: Passengers departures monthly traffic figures (comparision 2003-2005)

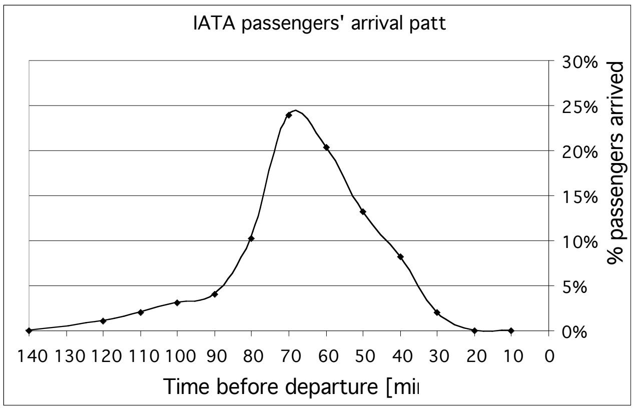


Figure 3: IATA passengers' arrival pattern (IATA, 1995)

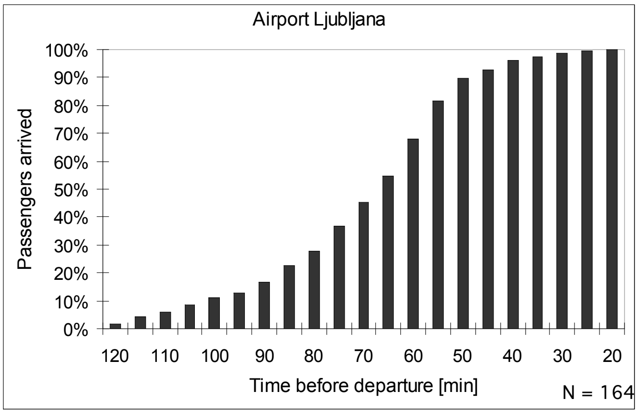


Figure 4: Passengers' arrival pattern

The typical peaked ness of the IATA arrival pattern (IATA, 1995) is shown on Figure 3. United States Federal Aviation Authority for example, uses typical peak hour passenger (TPHP) concept (FAA, 2004). The TPHP measure depends on total annual count of passengers (Ahyudanari, 2001). In some previous relevant work in other airports was shown that more than 60% of the passengers arrive approximately two hours before scheduled time of departure (de Ruiter, 2002). But patterns and peaks of passengers varies for each airport. The arrival pattern in the simulation model we used is an actual observed data in Airport Ljubljana (Fig. 4). Data samples were collected for two-hour periods before the flights departure time. The passenger arrival distributions per flight were averaged to get an aggregated arrival pattern.

Distribution of arriving passengers might be different. The average of different daily patterns is applied to represent the »Airport Ljubljana pattern« (Fig. 5). In the existing simulation model the different distribution type can be also quite easily applied.

We analyzed the statistical data and compared it to the most useful analytical distributions. We used several statistical software for curve fitting (Stat:Fit, Curve Expert and Expert Fit) and Chi-Square goodness-of-fit tests. After thorough analysis, comparing collected statistical data to the useful »standard« distributions (and not so perfect fitting results), we decided to use empirical data in the simulation model. That was, considering the amount of the gathered data, quite reasonable.

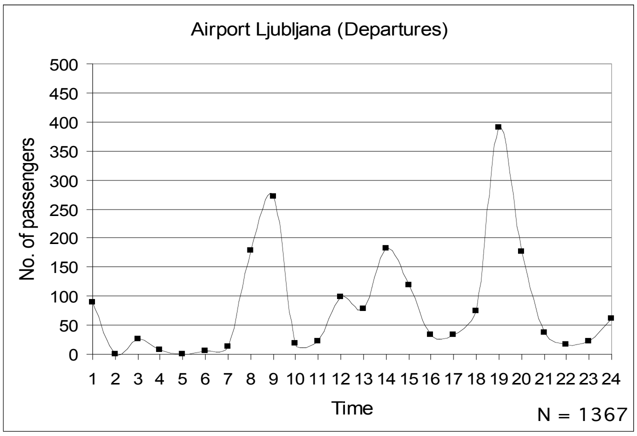


Figure 5: Daily average departures' pattern

3.2 Discrete event simulation model

A discrete-event simulation model can be described as one in which the state variables change only at those discrete points in time at which events occur (Kljajić, Škraba & Bernik, 1999). Airport traffic (passenger) flow is discrete stochastic process. Discrete event simulation is often used to model system where complex processes are combined with a limited infrastructure of capacity (Verbraeek & Valentin, 2002). Airports are quite an ideal application area for discrete event simulation.

A simulation model was constructed using simulation programming language GPSS (General Purpose Simulation System). There are multiple versions of GPSS available (Chisman, 1992) and the one used in the model was GPSS/H, specifically designed for the personal computer. GPSS was felt to possess the versatility, reliability and ease of programming necessary to produce a model of sufficient detail and sensitivity.

3.3 Simulation model development

The first step in the simulation model development was to understand and describe the current situation completely. Simulation model, which is a representation of an actual system, does not solve a problem but tells us how a system will operate under a given set of parameters. The key entities are passengers that move through a set of processes and activities that consume resources (Fig. 6).

We detaily analyzed departure passenger flow, as more complex than the arrival one, from the passenger entrance to boarding. Simulation starts with passengers entering the airport terminal (departure hall) at a certain time before departure and walking to the check in desks. Hereafter passengers have to pass the immigration to get into the international hall area. Finally, the passengers go to boarding and leave the airport.

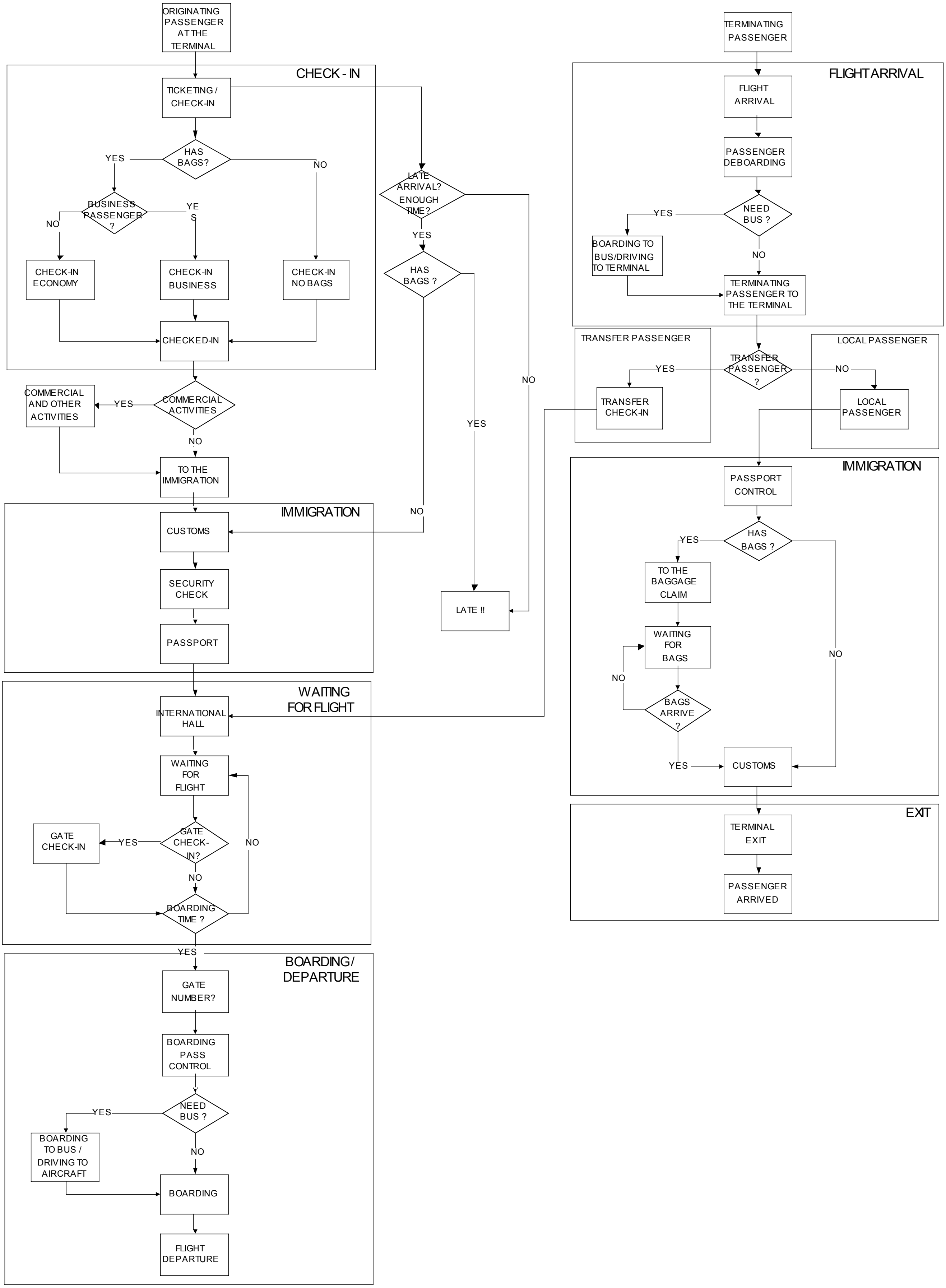


Figure 6: Passenger processing flowcharts (Rauch, 2005)

Ljubljana Airport has a special feature. Even though it is not a large airport, airport has substantial transfer traffic. This distinguishes it from many regional European airports. The airport is an especially important connection point between the southern Balkans and Central Europe. These traffic flows are subsiding somewhat, but transfer traffic still represent more than 20 percent of all traffic at Ljubljana Airport (Aerodrom Ljubljana, 2006). For the simulation purpose it is very important to accurately represent the passengers structure. Considering the gathered statistical data, we have 74 percent Economy passengers, 16 percent Business passengers and 10 percent passengers with no bags.

Check-in area is the main component of passenger service areas in airports. The queuing time passengers spend at check-in in one of the most important criteria of passengers satisfaction and service performance. Check-in queues are generally dependent of: flight departures, dynamic arrival pattern, available capacity and also of the alternative such as self service and internet/WAP check-in. The process of check-in is stochastic and the number of required check-in counters varies with time since the total number of passengers per flight is different (Chun, 1999). The queue discipline at check-in counters is FIFO, which means passengers are served in a »first-in, first-out« fashion.

Check-in capacity in Airport Ljubljana is generally sufficient to meet the total daily demands but because of strong fluctuations and peaks over the day in the number of arriving passengers, queuing takes place much stronger than it should on average. Check-in process is modeled using statistics collected on check-in servicing time. For the service rate, sample data were randomly collected at different times using stopwatch methodology. The statistics include check-in servicing time for different destinations and times of the day. The graph of the passenger check-in data is known in the industry as a »check-in curve« (Fig. 7).

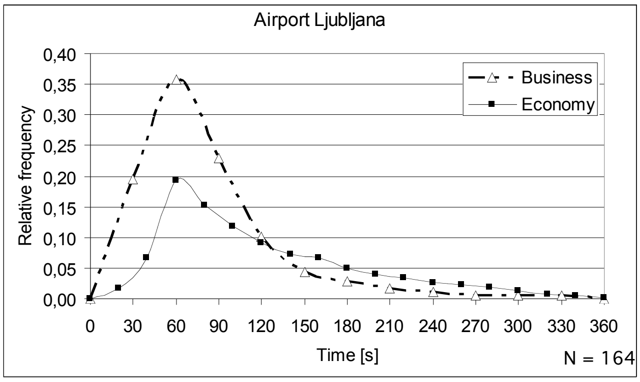


Figure 7: Check-in service distribution

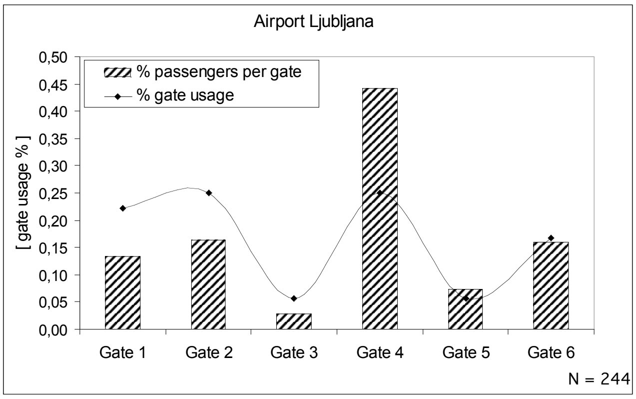


Figure 8: Existing gate usage

The immigration processing time (customs, security check, passport control) are dependent on factors such as passenger immigration arrival pattern, available capacity (number of security check points) and factors related to the flight. For example, flights to certain destinations may require more processing time because certain countries may require a more stringent passport and visa check etc. Airport, in the present situation has 6 gates (in the time of the study the seventh one was still constructed). The most used one is Gate 4, with almost half of the daily boarded passengers, in average (Fig. 8).

Constructing a suitable model of an airport passenger service process presents a number of challenges (Kyle, 1998). The passenger arrival rates are dependent on many factors such as the time of departure, destination of the flight, etc. For example, if the flight is scheduled to depart early in the morning, passengers will usually arrive a bit later than the statistical average. The arrival process is modeled using passenger arrival statistics which showed that the arrival rate of passengers is a function of time. The passenger arrival rate is non-stationary distribution with more passengers arriving during the middle part of the check-in counter opening period (de Ruiter, 2002).

The normal congestion condition is based on the flight operations for a busy operating day, that is, the number of departing flights is 60 (in the summer period) and the boarding ratio is 70 percent. On busy days more than six thousand people arrive and depart from the airport. Approximately half of them are departures and 20 percent of all are transfer passengers. For the simulation model purpose a »company« is selected to be analyzed (more than 70 percent of all flights are on this company). Passengers are generated in the simulation model (Fig. 9) using the arrival pattern and the passenger service time considering pas-

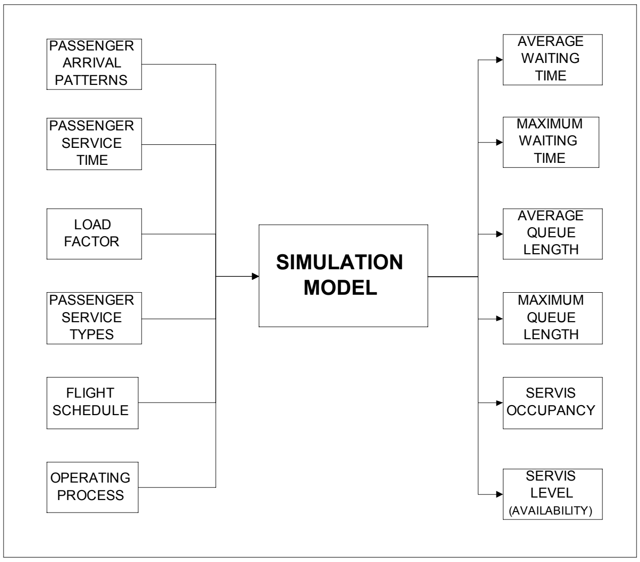


Figure 9: Simulation model (inputs and outputs summary)

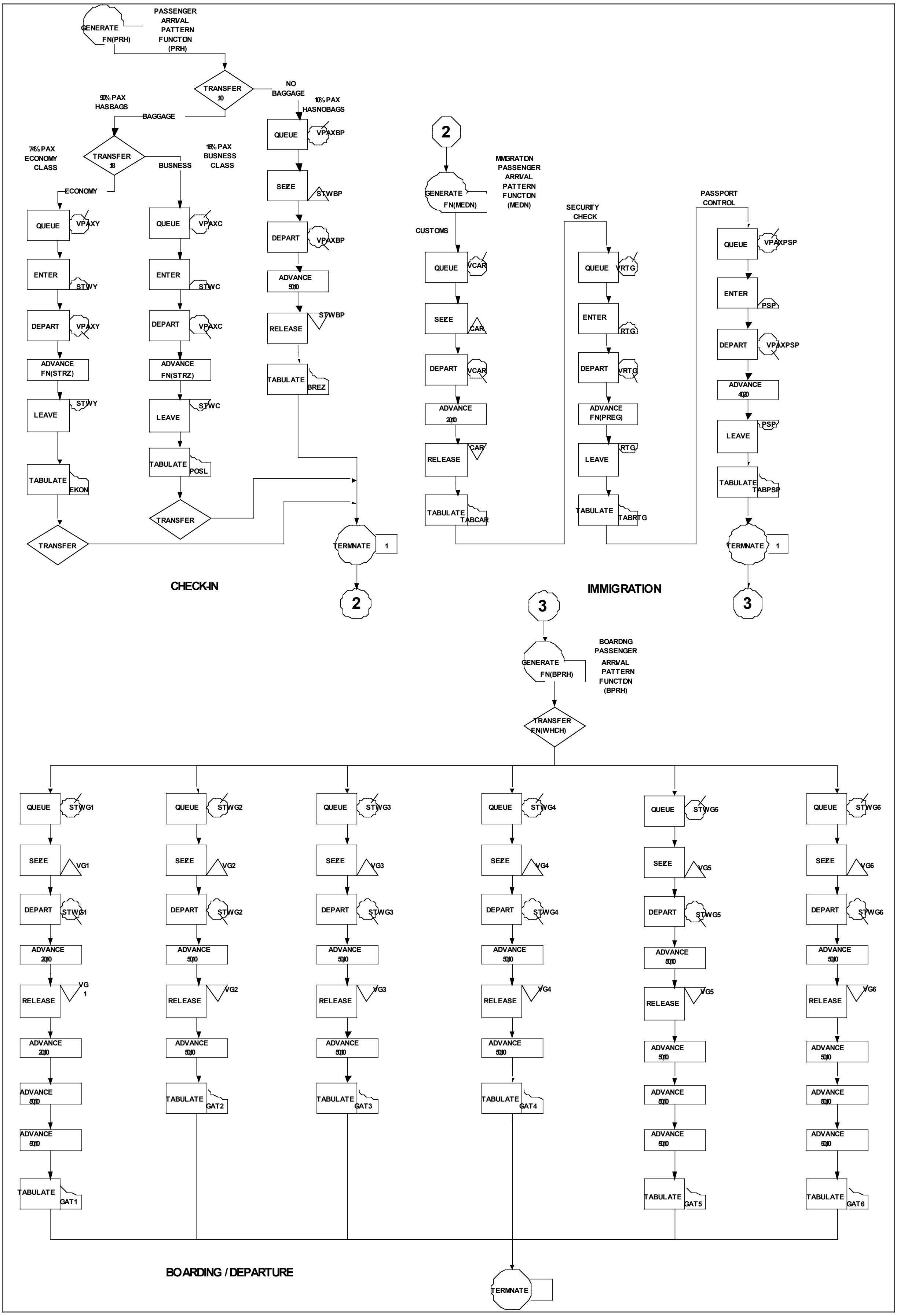


Figure 10: Simulation model block scheme

]

senger service type, load factor and the scheduled time of departure of a flight (passenger load factors are the number of passengers who flew on the aircraft divided by the total available seats). Simulation model block scheme is shown in Figure 10.

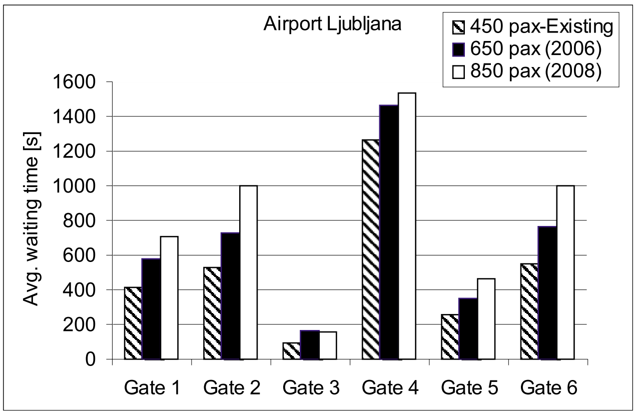
3.4 Model validation

Model validation is critical in the development of a simulation model. Validation is the process of ensuring that the model is an accurate representation of the real system. Input data for the model was the first to be validated. The second step of the model validation involved verification of the model to make sure that reflected the provided data accurately. The next step involved validation of the simulation model. Simplified version of the model development process and validation is shown in Figure 11 (Sargent, 2003).

Data that we collected on a system for building the test model were partly used to build the model and partly to determine weather the model behaves as the system does. Several replications (runs) were made to determine the stochastic variability of the model. The results were then compared with real conditions. Luckily, because we already have an existing »system«, we can easily compare simulated results with »real word data«. We also asked knowledgeable (independent) colleague about the system, whether the logic (input-output) of the model is reasonable to them. Further validation included a structured »walk through« to verify the model logic and compared the model output with the actual system key performances (by observing actual passenger arrivals at the check-in and immigration lines).

Comparison of the model results with the actual system showed that the model quite accurately represents the actual system. Once the model is calibrated, input require-

CHECK-IN
AVERAGE WAITING TIME



BOARDING
AVERAGE WAITING TIME

Figure 12: Extracts of some simulation results

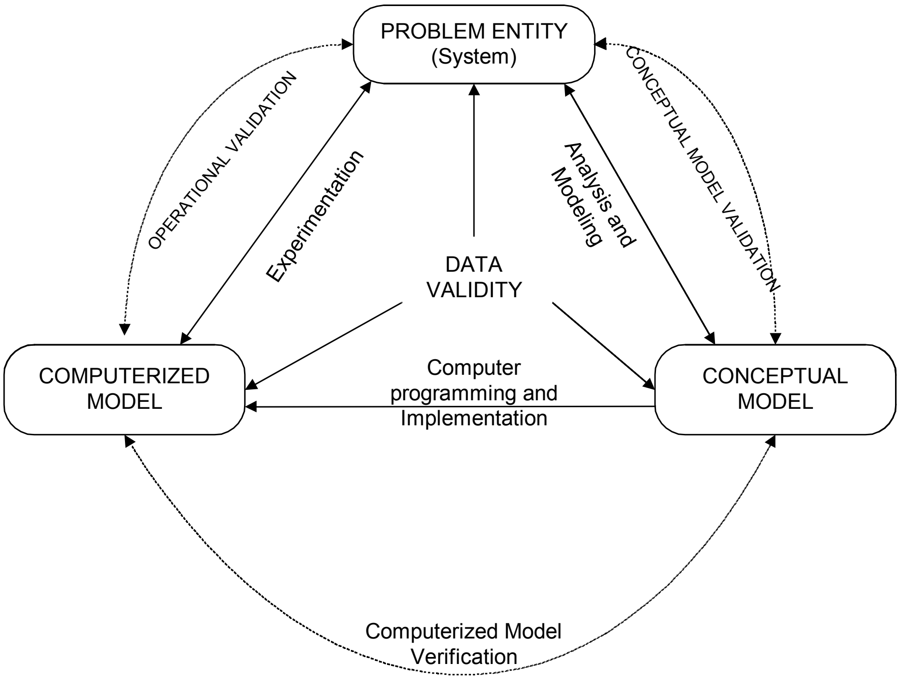


Figure 11: Modeling process and validation (Sargent, 2003)

ments can be changed for evaluating the »What-if« scenarios. We are aware of the fact determining that a model has sufficient accuracy does not guarantee that a model is valid everywhere in its applicable domain (Sargent, 2003). There is no set of specific tests that can be easily applied to determine the »correctness« of a model. Since the airport is exposed to air traffic developments and external influences, it is desirable to validate and repeat the study results frequently.

4 Simulation results

After analyzing all collected simulation statistics, we focused on investigating conditions where passenger waiting time exceeded. Bottleneck resources for the peak hours were found to be in the check-in process and in the immigration control process (especially in the security control). Several simulations were performed; simulated statistics were then compared with the tolerable waiting time and queue length (IATA, 1995). The overview of bottlenecks and the comparison of measurements formed the required results of the simulation study. Recommendations based on the results of models runs were made (Fig.12).

Some proposed solutions turned out to be quite very effective, whereas others showed no effect. However, some other effects of the study turned out to be just as useful such as exact analyzing of processes and representative data.

5 Conclusion

A model was developed as much as flexible, allowing modifying the different parameters of the system easily. Critical aspects in the passenger flow through the airport terminal have been explored and studied. The advantage of using simulation models (software) is that you can explore »what-if« scenarios or new methods without the expense of experimenting with the real system.

One result of growing air traffic at Ljubljana Airport and Slovenia's entry into the European Union, which requires the separation of traffic into Schengen and non-Schengen, is the planned construction of a new airport terminal with planned capacity of 850 departing passenger and 850 arriving passengers per hour. The current terminal will be renovated and in the next years expanded to 32,000 square meters, where 40 check-in counters will be set up, including some automatic ones (Aerodrom Ljubljana, 2006). The complete separation of Schengen and non-Schengen traffic will be ensured, as well as the separation of arriving and departing passengers. An IATA level C standard of services (IATA, 1995) is envisaged to ensure the necessary quality for passenger arrivals and departures. Passenger volumes will double in the next decade, airport space will remain relatively unchanged. New technologies such as e-ticketing and check-in from remote locations are major opportunities (passengers can bypass counters and proceed directly to boarding area), clearly, considering the increased security in the recent years. Wireless communications are rapidly infiltrating in all sectors of airport operations and instituting new services to improve service on a real time basis with their customers.

One of the greatest advantages of using simulation software in airports is that once you have developed a valid simulation model, you can explore new policies, operating procedures or methods without the expense and disruption of experimenting with the real system. Modifications are incorporated in the model, and you observe the effects of those changes on the computer rather than the real system. Simulation lets you »Test Drive« your Airport before you build or change it. Nevertheless, frequent maintenance of the simulation model is necessary, because airports are in a continuous state of change. Future works that may improve this study include the development of a friendly user interface to the model to allow users to change variables. Using simulation it is also possible to test alternative check-in methods, e.g. dynamic opening and closing of check-in counters depending on the number of queuing passengers.

Hopefully, this paper can provide some insights on how computer simulation can help to find bottlenecks and solutions insight in peak flows, optimize quality of service to our customers and at the same time reduce costs.

Literature

Aerodrom Ljubljana (2006). Available from: http://www.lju-airport.si (14.07.2006).

Ahyudanari, E. (2001). Space Requirement at Airport Check-In area, *Conference of Australian Institutes of Transport research (CAITR 2001),* 10th -12th Dec. 2001. Available from: http://civil.eng.monash.edu.au/its/caitrhome (15.06.2004).

Chisman, J. (1992). *Introduction to simulation modeling using GPSS/PC*, Prentice-Hall International, Inc.

Chun, H.W. (1999). Intelligent Resource Simulation for an Airport Check-in Counter Allocation System, *IEEE Transaction on Control Systems Technology*, **29**(3): 325-335.

de Ruiter, G. (2002). Airport Optimization by Simulation, *Air Transport and Travel Information Systems Conference*, ATTIS 2002, 25th - 27th Feb. 2002. Amsterdam.

Durante, J. (2003): Simplifying Passenger Travel (SPT)…the way travel should be, *International Airport Review*, 2003 (2): 30-35.

Federal Aviation Administration - FAA. (2004). Available from: http://www.faa.gov (12.01.2004).

International Air Transport Association - IATA (1995). *Airport Development Reference Manual*, 8th Edition.

Kljajić, M., Škraba, A., Bernik, I. (1999). *Dogodkovna simulacija sistemov* (*Discrete event simulation)*, University of Maribor, Faculty of Organizational Sciences.

Kukulich, M., Leone, K. (2002). Modeling: addressing the checkpoint challenge, *Airport Technology International*, 2002: 68-74.

Kyle, R. (1998). Washington Dulles International Airport Passenger Conveyance Study, *Proceedings of the 1998 Winter Simulation Conference (WSC 1998).* Available from: http://www.informs-cs.org/wscpapers.html (14.03.2004).

Park, Y., Ahn, S. (2003). Optimal assignment for check-in counters based on passenger arrival behavior at an airport, *Korea Transport Institute*, 2003. Available from: http://ekoti.koti.re.kr/project/Epubreports.nsf (27.02.2004).

Rauch, R. (2005). Discrete event passenger flow simulation model for an airport terminal capacity analysis, Master thesis, University of Maribor, Faculty of Organizational Sciences.

Sargent, R. (2003). Verification and validation of simulations models, *Proceedings of the 2003 Winter Simulation Conference (WSC 2003)*. Available from: http://www.informs-cs.org/wscpapers.html (14.03.2004).

Usenik, J., Radačić, Ž. (1997). An airport system control-a theoretical mathematical model with discrete inputs, *Proceedings of the 6th International Symposium on Electronics in Traffic, ISEP 97,* Edited by Vidmar, H., Ljubljana, 09-10 Oct. 1997. Ljubljana, Elektrotehniška zveza Slovenije.

Verbraeek, A., Valentin, E. (2002). Simulation building blocks for airport terminal modeling, *Proceedings of the 2002 Winter Simulation Conference (WSC 2002)*. Available from: http://www.informs-cs.org/wscpapers.html (14.03.2004).

Verougstraete, A. (2001). Effective communication is essential to deliver an optimal end product. *3rd ACI Ground Handling Conference*, ACI 2001, 9th-11th Apr. 2001. Juan les Pins, France.

Vredenburg, M. (1996). Airport Operational Efficiency, *Airport privatization seminar forum*, International Civil Aviation Organization (ICAO), Guatemala City, 13th -16th Dec. 1996. Available from: http://icao.int/en/m_publications.html (17.01.2004).

Robert Rauch graduated from the Faculty of Mechanical Engineering, University of Ljubljana with graduation thesis »Aircraft handling planning«. He has received his B.Sc and M.Sc. degree in information systems management from the University of Maribor, Faculty of Organizational Sciences. He has been working at the Airport Ljubljana as an airport technology expert, where he has been involved in several projects concerning airport development. His major research interests cover fields of airport information systems development, airport simulations and airport technology. He has over 10 years of experience in the related areas.

Miroljub Kljajić is Professoor at the Faculty of Organizational Sciences, University of Maribor. His brief biography is published on page 643 of this issue.

Simulation Approach to Warehouse Cost Minimization in Stochastic Environment

Davorin Kofjač, Miroljub Kljajić

University of Maribor, Faculty of Organizational Sciences, Kidričeva cesta 55a, SI-4000 Kranj, Slovenia
davorin.kofjac@fov.uni-mb.si, miroljub.kljajic@fov.uni-mb.si

The objective of inventory management is to balance conflicting goals like keeping stock levels down to have cash available for other purposes and having high stock levels for the continuity of the production. Simulation approach is used to minimize total warehousing cost while no stock-outs occur and warehouse capacity is not exceeded. A case study of replenishment process optimization is presented on several representative materials of an automotive company using two replenishment algorithms: fix review period and full capacity ordering. The presented simulation results indicate considerable cost reduction without violating the mentioned constraints. The fuzzy logic evaluator, a decision support system used for simulation results assessment, is presented and discussed.

Key words: inventory control, simulation, optimization, stochastic models, fuzzy sets, decision support system

Simulacijski pristop k minimiziranju stroškov skladišča v stohastičnem okolju

Cilj upravljanja z zalogami je uravnotežiti nasprotujoče si kriterije, npr. imeti na zalogi dovolj za kontinuirano proizvodnjo, hkrati pa imeti kar najnižje zaloge, da so finančna sredstva na voljo za druge namene. Predstavljen je simulacijski pristop za minimizacijo skupnih stroškov poslovanja skladišča ob dveh omejitvah: ne sme biti izpadov proizvodnje in kapaciteta skladišča ne sme biti presežena. Raziskava optimizacije procesa naročanja se je vršila na izbranih artiklih podjetja. Uporabljena sta bila dva algoritma naročanja: naročanje na stalne intervale in naročanje na polno kapaciteto skladišča. Predstavljeni simulacijski rezultati kažejo na pomembno zmanjšanje stroškov brez kršitev zastavljenih omejitev. Opisan je sistem za podporo odločanju, t.j. ocenjevalec simulacijskih rezultatov na podlagi mehke logike.

Ključne besede: kontrola zalog, simulacija, optimizacija, stohastični modeli, mehke množice, sistem za podporo odločanju

1 Introduction

In an organization even with moderate size, there may be thousands of inventory stock keeping units and the main warehouse task is to enable the undisturbed production by assuring the right amount of materials. Several different principles of warehouse optimization are described in (Silver et al., 1998; Tompkins and Smith, 1998; Ljubič, 2000). One way of optimizing the warehouse process is to find the right replenishment strategy, while reducing the cost of the warehousing processes to a minimum without stock-outs occurring and warehouse capacity being exceeded. Therefore, the operator is dealing with the decision problem – when to order and how many? This decision problem is vast, especially if we consider the fact, that he has to find the right replenishment strategy for more than 10.000 components stored in the warehouse. Here the help of decision support system (DSS) is crucial as warehouse operators mainly use only their experience and intuition.

Decision making inherently involves consideration of multiple objectives and uncertain outcomes; and in many situations, we have to take into account both the outcomes of current decisions and future decision opportunities. Decision processes under uncertainty deal with the optimization of decision making under uncertainty over time. Problems of this type have found applications in a variety of decision contexts in different industries, including manufacturing, R&D management, finance, transportation, power systems, and water management. Manufacturing firms operate in an environment in which such factors as product demand and technology evolution inevitably involve uncertainty. Production planning and inventory control are operational level decisions that firms must make on a regular basis. Effective inventory control is important to managing cost by properly balancing various costs such as inventory carrying costs and transportation costs. Capacity planning is also the crucial part of strategic level decision making in the manufacturing and service industry. Complications arise in decisions on timings and sizes of investments in capacity due to the uncertain demand for capacity, e.g., customer demand, and availability of capacity, e.g., technology development. All of these factors, along with the significant and long-term impact of capacity decisions, make capacity planning one of the most important yet complex decisions for most industries (Cheng et al., 2005).

In supply chain organization, the main difficulty relates to system complexity. Many different viewpoints have to be considered, from legal agreements to technical constraints. To deal with such a complexity, most decision support systems are based on simulation tools (Arda and Hennet, 2004). Among specialists, it is widely accepted that mathematical or analytical modeling techniques are not sufficient if a detailed analysis is required of complex systems. The major weaknesses in using mathematical or analytical methodologies are (Wang and Chatwin, 2004):

- When analyzing a complex system, stochastic elements cannot be accurately described by a mathematical model and cannot be evaluated analytically as modern systems consist of many operations that occur randomly and nonlinearly. Therefore, the objective function may not be expressible as an explicit function of the input parameters; hence, mathematical models or other methods are impractical.
- Dynamic systems involve randomness that changes with time, such as an assembly line, where the components being assembled change with time. The modeling of complex dynamic systems theoretically requires too many simplifications, and the emerging models may not, therefore, be valid.
- Purely analytical methods are often insufficient for optimization because a mathematical model can only be built based on simplifying assumptions; therefore, accuracy often becomes a major problem for system optimization.

In some cases, one must resort to simulation even though in principle some systems are analytically tractable; that is because some performance measures of the system have values that can be found only by running a simulation model or by observing an actual system. Consequently, the analytical effort required to evaluate the solution may be so formidable that computer simulation is the only realistic option. Instead of using experts to build an extensive mathematical model by using the analytical approach, computer-based simulation is used where the method of analyzing the system is purely theoretical. Computer-based simulation is seen as an integral business tool giving flexibility and convenience in designing, planning and analyzing complex processes and/or systems. This is because computer-based modeling and simulation methods have the capability of representing the complex static structure as well as the dynamic behavior of systems (Wang and Chatwin, 2004; Kljajić et al., 2000).

Clearly, the imaginative and disciplined application of dynamic modeling and simulation provides a potentially useful mechanism through which managers can gain a comprehensive understanding of system behavior, concentrating on core business processes such as order fulfillment, product development as well as customer acquisition, satisfaction, and retention. However, the means by which management in general and senior management in particular make decisions can, in itself, also be regarded as a core value-adding process that impacts fundamentally upon the overall effectiveness of the organization (Fowler, 2003).

This paper presents the simulation model based on system dynamies methodology (Forester, 1961), used to solve replenishment strategy problems (when to place an order and how many products to order) in a medium-sized company in order to improve its warehousing processes. Previous research is described in (Kofjač and Kljajić, 2004).

2 Warehouse model

2.1 The warehouse problem formulation

Dealing with problems of warehousing, we encounter several contradictory criteria. An overly large warehouse means a greater amount of stock, greater capital cost and more staff. The space itself is very valuable today. An overly small warehouse can represent possible stock-outs, it demands a reliable supplier etc.

Products stored in a warehouse also play an important role in a process of optimization of warehousing processes. They belong to different categories according to ABC and XYZ classification. ABC classification divides products into three categories according to their value, while XYZ classification divides products into three categories according to the dynamics of their consumption (Silver et al., 1998; Ljubič, 2000). The dynamics of product consumption and the products value must be taken into consideration in order to improve the warehousing processes. We believe that there is a lack of optimization technology in use and that there are a number of possibilities of how to improve the warehousing processes. The warehouse personnel solve the complex problems mostly by using their experience, without the use of optimization techniques.

Our goal is to rationalize the warehouse replenishment process, this means determining the interval between orders and the quantity to be ordered, so that the warehouse will operate with minimal common costs. Cost function includes:

- fixed ordering costs,
- transportation costs,
- costs of taking over the products,
- costs of physical storage,
- cost of capital.

The following limitations have to be taken into consideration:

- maximal warehouse capacity for a specific product must not be exceeded,
- no stock-outs may occur.

In this case we are dealing with a warehouse used for storing components for further build-in. The lead time for some products delivered into the warehouse is stochastic within an interval [t_{dmin}, t_{dmax}]. The problem occurs in defining ordering quantity, because past orders must be considered as well as the average consumption of a specific product. Long lead times also represent a problem, because they are usually much longer than the time period in which the consumption plan can be predicted with a certainty. Therefore, the variability of a production plan has to be considered. Unlike deterministic models, stochastic models do not necessarily give the same output for the same input. Within a stochastic model there will be at least one variable that is not known with certainty (Oakshott, 1997). In this case the variables are the consumption plan and lead times.

A consumption plan is planned for 24 weeks and can be predicted with a certainty, e.g. for six weeks. After this period, a consumption plan uncertainty factor (e.g. 3%) must be considered every two weeks. Therefore, a safety factor, which increases the ordering quantity, must be considered when placing an order (e.g. 10%).

2.2 CLD of the warehouse process and its simulation model implementation

Figure 1 represents the causal loop diagram (CLD) from which the influences of the warehouse model elements can be observed. The arrow represents the direction of the influence and the + or – sign its polarity.

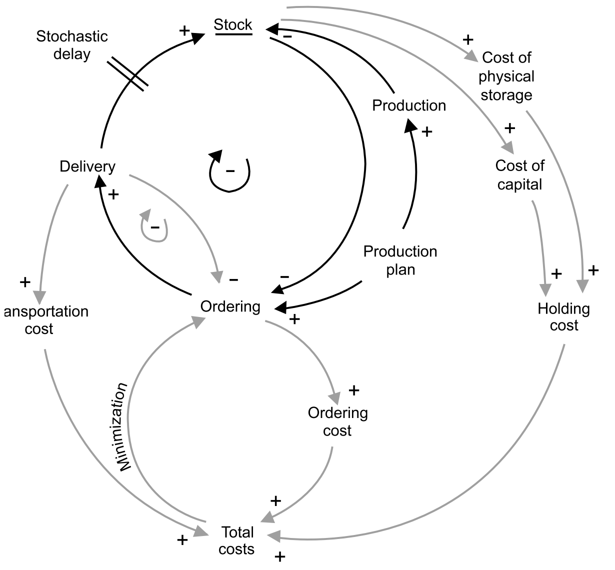


Figure 1: Causal loop diagram of the warehouse model

Delivery impacts *Stock* and *Transportation Costs*. If the amount of *Delivery* increases above what it would have been, the *Stock* and *Transportation Costs* are increased above the initial value. The increased value of *Stock*, increases *Cost of physical storage* and *Cost of capital*, but it decreases *Ordering quantity*. If the quantity of *Production plan*, which represents the reference value, is increased, *Consumption* and *Ordering quantity* are both increased. The increased value of *Consumption* decreases *Stock*. If the *Ordering quantity* is increased, the *Delivery* and *Fixed ordering cost* are both increased. The increased values of *Cost of physical storage* and *Cost of capital* increase the value of *Holding cost*, which increases the value of *Total cost* together with *Fixed ordering cost* and *Transportation cost*.

There are two negative feedback loops in the causal loop diagram. The first interconnects *Stock*, *Ordering* and *Delivery* and it represents the fact that we order less if the stock level is high. The second interconnects *Delivery* and *Ordering* and represents the concept that we order less if we have ordered more before. This loop takes into account orders which haven't been delivered yet and will have impact on the stock level later on.

Figure 2 shows the warehouse simulation model built with Matlab (submodels are excluded). Matlab was chosen because it supports simulation with Simulink and offers a powerful computational engine, which provides a quick execution of the simulation runs.

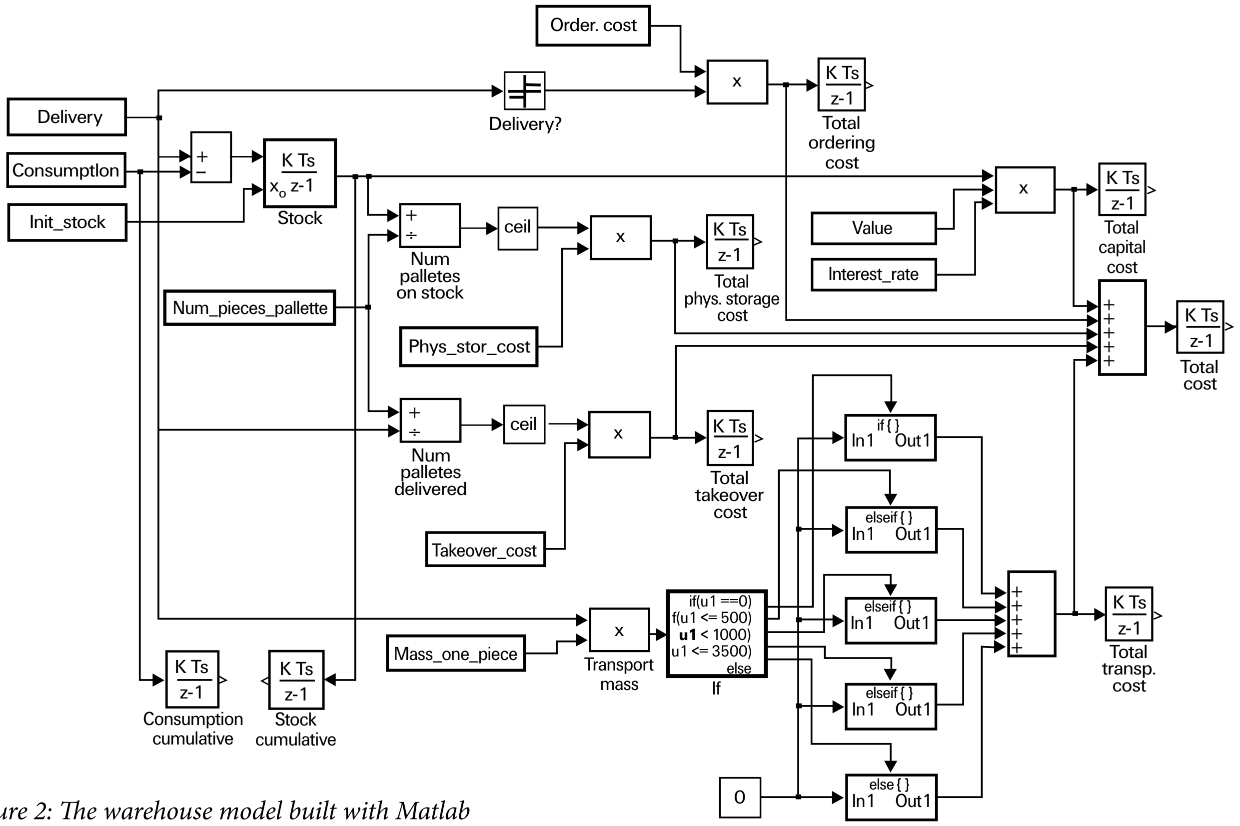


Figure 2: The warehouse model built with Matlab

3 Replenishment algorithms

Two replenishment algorithms were applied in the simulation model in order to find an ordering strategy which would produce less common costs: a model with fix review period and a full capacity ordering model.

3.1 Fixed review period algorithm

The Fixed review period algorithm (FRP) is similar to the (R, S) system (described in Silver et al., 1998), where, every R units of time, an order is made to adjust the stock level to the order-up-to-level S. In contrast to the (R, S) system, S is not a fixed value in the FRP algorithm. The FRP is based on making a sum of consumption for a specific material over a specific period (fixed) of time. The quantity of this sum is used in order quantity calculation together with the past orders and stock-on-hand. This model is appropriate for products with great warehouse capacity.

3.2 Full capacity ordering algorithm

The Full capacity ordering algorithm (FCO) is similar to the FRP and the (s, S) system (described in Silver et al., 1998). Replenishment in the (s, S) system is made whenever the stock level drops to the order point s or lower and variable replenishment quantity is used, ordering enough to raise the stock level to the order-up-to-level S. In comparison to the (s, S) system, the s is omitted in FCO. The consumption for a specific material is not summed for a fixed review period (like in the FRP algorithm); instead the consumption is summed until we reach the maximum warehouse capacity (S) for a specific material. This algorithm is appropriate for materials with very limited warehouse capacity. This algorithm is not used if the capacity is unknown.

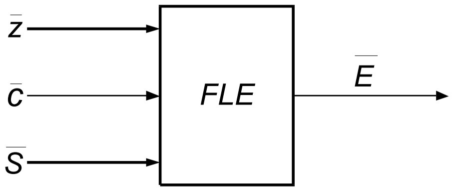


Figure 3: Fuzzy logic evaluator

4 Fuzzy logic evaluator

Several factors in the supply chain are hard to evaluate, e.g. shortage cost, supplier's reliability etc. In some cases, it is possible to evaluate e.g. supplier's reliability only linguistically by saying that supplier is "very reliable" or "sometimes reliable". Since those are only soft descriptions, fuzzy logic (Zadeh, 1965) is often used to assess such linguistic descriptions. In our case, the shortage cost is unknown; only the number of stock-outs is known. The fuzzy number of stock-outs, together with fuzzy highest stock level and fuzzy total cost, is used to assess the results of each simulation scenario.

The fuzzy logic evaluator (FLE) is presented in Figure 3. The inputs of the FLE are the crisp arrays of number of stock-outs ($\overline{C}$), total costs ($\overline{Z}$) and highest stock levels ($\overline{S}$). The array contains e.g. the number of stock-outs for each simulation scenario. The arrays are normalized in the interval [0, 1] and then fuzzified using equally spaced gaussian membership functions (MFs) as shown in Table 1.

The FSA inference system contains the expert's rule base consisting of 125 (= 5^3) rules. Estimation of the i^{th} replenishment strategy is calculated according to the eqn. 1:

If Z_i **and** C_i **and** S_i **then** E_i, $i = 1, 2, \ldots n$ (1)

where *n* is a number of simulation scenarios. The output of the FLE is the array of scenario assessments, $\overline{E}$ which is defuzzified using a *Som* (smallest value of minimum) function in the range [0, 1]. The scenario, i.e. replenishment algorithm, with the lowest grade is suggested as the most suitable.

5 Results

The experiment was performed with the historic data provided by the observed company. Altogether, nine materials were examined in this study and their details are presented in Table 2.

The simulation of the actual warehouse process (RP – Real Process) was using real data of delivery and demand, while the simulation with replenishment algorithms (VP – Virtual Process) was using only real demand data. The ordering and delivery process in VP were controlled by replenishment algorithms. The RP simulation was run only once, whereas ten simulation runs were executed

Table 1: FLE membership functions

Variable	Membership functions				
Z	none	few	some	many	a lot
S	very low	low	middle	high	very high
C	very low	low	middle	high	very high
E	excellent	very good	good	poor	very poor

Table 2: The materials details – classification, capacity and lead-times

Case	Classification	Capacity (piece)	Lead-time (week)
1	BY	200000	6-8
2	AX	-	5-6
3	BY	-	5-6
4	AZ	-	5-6
5	AY	-	6-7
6	BZ	-	6-7
7	AY	120000	5-6
8	BX	-	6-7
9	AZ	70000	14-16

for every VP replenishment algorithm. Based on these simulation runs, average costs and average stock-outs were calculated. With several simulation runs and a calculation of average values, we have tried to minimize the influences of the random generator, which represents the stochastic environment. Out of all simulation runs the maximum stock level was considered and the strategy with minimum highest stock level is favored. A Monte Carlo simulation was used for variation of production plan unreliability. The production plan variability was simulated by perturbations of its quantity every 2 weeks for a certain amount, e.g. 5%. Variable lead times are simulated by uniform random generator. If stock-outs occurred during the simulation, the missing quantity was transferred into the next period. The safety stock was also considered; it was equal to the average weekly demand.

Table 3: The replenishment algorithms simulation results for stock-outs (Z), highest stock level (S) and total cost (C) for the observed cases

Case		Real	FRP 2 week	FRP 3 week	FRP 4 week	FRP 5 week	FRP 6 week	FRP 7 week	FRP 7 week	FRP 9 week	FRP 10 week	FCO
1	$Z*e^{0}$	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	***0,00***
	$S*e^{5}$	1,44	2,17	2,09	2,28	2,31	2,37	2,59	2,53	2,56	2,55	***2,00***
	$C*e^{6}$	1,03	1,63	1,57	1,63	1,62	1,60	1,65	1,64	1,66	1,64	***1,38***
2	$Z*e^{1}$	0,00	0,00	0,00	0,00	0,00	***0,00***	0,00	0,00	0,00	0,00	
	$S*e^{3}$	4,32	1,30	1,58	1,52	1,74	***1,85***	1,89	2,05	2,51	2,86	
	$C*e^{6}$	2,50	1,40	1,34	1,28	1,24	***1,23***	1,24	1,26	1,29	1,29	
3	$Z*e^{1}$	0,00	0,00	0,00	0,00	0,00	***0,00***	0,00	0,00	0,00	0,00	
	$S*e^{3}$	4,95	0,89	1,27	1,44	1,46	***1,48***	1,47	2,20	1,62	1,93	
	$C*e^{6}$	3,21	1,14	1,06	1,02	0,98	***0,94***	0,96	1,03	1,04	1,03	
4	$Z*e^{1}$	0,00	***0,00***	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
	$S*e^{3}$	4,48	***4,02***	4,29	5,41	5,48	5,99	6,71	7,67	7,78	8,97	
	$C*e^{6}$	2,32	***1,69***	1,71	1,78	1,81	1,90	1,92	2,17	2,16	2,21	
5	$Z*e^{0}$	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	***0,00***	
	$S*e^{3}$	6,25	4,75	4,95	5,99	5,91	6,15	5,76	6,61	6,81	***7,52***	
	$C*e^{6}$	8,59	9,65	9,69	8,56	7,73	8,37	7,97	7,42	7,41	***7,23***	
6	$Z*e^{0}$	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	***0,00***	
	$S*e^{3}$	1,21	1,10	1,25	1,88	1,83	1,48	1,87	2,76	2,77	***2,03***	
	$C*e^{5}$	2,65	3,02	3,10	2,94	2,88	2,75	2,89	2,86	2,76	***2,71***	
7	$Z*e^{0}$	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	***0,00***
	$S*e^{5}$	1,04	2,06	2,06	2,30	2,21	2,23	2,26	2,33	2,43	2,38	***1,20***
	$C*e^{6}$	2,75	1,83	1,87	1,95	1,88	1,94	1,94	2,00	2,00	1,97	***1,83***
8	$Z*e^{0}$	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	***0,00***	
	$S*e^{3}$	1,26	0,82	0,97	0,92	0,99	1,24	1,32	1,40	1,42	***1,51***	
	$C*e^{6}$	1,67	1,76	1,56	1,45	1,33	1,22	1,16	1,13	1,12	***1,05***	
9	$Z*e^{1}$	0,00	0,00	0,00	0,00	0,00	0,00	***0,00***	0,00	0,00	0,00	0,00
	$S*e^{4}$	2,07	4,72	5,13	5,41	5,69	6,34	***6,17***	7,30	7,37	7,81	7,00
	$C*e^{6}$	6,77	5,81	5,58	5,21	4,78	4,62	***4,59***	4,73	4,55	4,64	4,89

Table 4: The achieved savings of the VP regarding the RP

Case	RP total cost	VP total cost	Savings	The best VP strategy
1	1,03*e^{6}	1,38*e^{6}	33,9	FCO
2	2,50*e^{6}	1,23*e^{6}	50,8	FRP - 6
3	3,21*e^{6}	0,94*e^{6}	70,7	FRP - 6
4	2,32*e^{6}	1,69*e^{6}	27,2	FRP - 2
5	8,59*e^{6}	7,23*e^{6}	15,8	FRP - 10
6	2,65*e^{5}	2,71*e^{5}	2,3	FRP - 10
7	2,75*e^{6}	1,83*e^{6}	33,5	FCO
8	1,67*e^{6}	1,05*e^{6}	37,1	FRP - 10
9	6,77*e^{6}	4,59*e^{6}	32,2	FRP - 7[1]

Simulation results are shown in Table 3, where the "Real" column represents simulation results of the RP, while the other columns represent results of the VP. The VP strategy producing the best results, as assessed by FLE, is indicated in bold and italics for each case. The research team verified the FLE assessments.

The summary of replenishment process optimization is presented in Table 4. No stock-outs occurred in any case and significant total cost savings were achieved with almost all cases, except for Cases 1 and 6, where VP replenishment algorithms could not achieve any savings; on the contrary, these algorithms yielded higher total costs – 33,9% for Case 1 and 2,3% for Case 6. Obviously, the warehouse operator has been using a better replenishment method than the one proposed by VP replenishment algorithms.

The FCO algorithm yielded the best results for the Cases 1 and 7, where capacity is known; all other VP replenishment strategies exceeded the warehouse capacity for these two cases, thus violating the capacity restriction. Capacity is also known for Case 9, but in this case the "FRP 7 weeks" algorithm was chosen as the one producing the best results. Although "FRP 9 weeks" produced lower

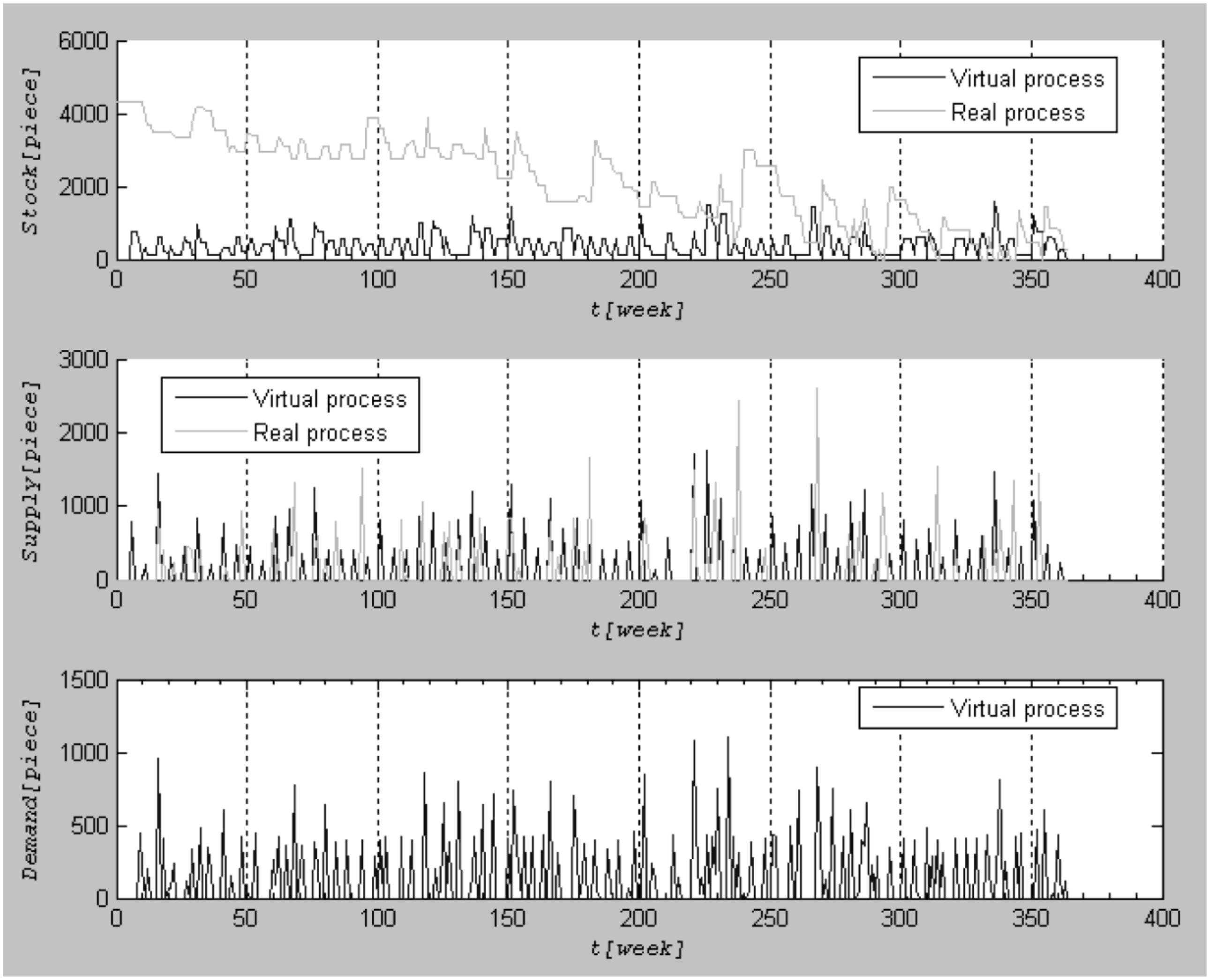


Figure 4: Comparison of stock-on-hand, supply and demand dynamics for the RP and the VP for a specific case

total cost than "FRP 7 weeks", it also violated the capacity restriction.

Figure 4 presents simulation results for the RP and VP for a specific case. The RP is represented by a brighter line and the VP by a darker line. The first graph presents stock level dynamics, the second delivery dynamics and the third the consumption dynamics throughout simulation time. The results shown can be used to indicate similarities or differences between the two processes. The supply dynamics graph indicates some similarities in the ordering strategy – some peaks (representing order quantity) are very similar but with some time delay. The simulator also allows us to compare two methodologies used in the ordering process: heuristics of the warehouse operator and algorithm based on simulation. From the Figure 4 one can observe, from the stock level dynamics, the operators' "learning by experience". Namely, starting from high stock level, the operators' replenishment strategy slowly improves over time approaching optimal strategy obtained by simulation. From the obtained results we can deduce about the usefulness of simulation method for the operator training for the new replenishment strategy.

Figure 5 shows frequency distribution comparison for the ordering quantities and the intervals between two consecutive orders. In the upper diagram, one can notice ordering quantities similarities between the RP and the VP with majority around 500 pieces per order. The lower diagram points towards a big difference between ordering intervals; the RP's intervals are scattered throughout the diagram, while the VP's intervals have a predominant value at five weeks.

The analysis presented in Figure 4 and Figure 5 allows in-depth insight into the reasons for the total cost reductions. From Figure 4, one can assume that cost savings of the VP are mainly due to the much lower cost of capital, because of much lower cumulative stock-on-hand. On the other hand, Figure 5 might point towards the non-optimal reorder timing of the RP. Obviously, the observed case needs a strict replenishment policy with a fixed review period instead of the frequently changing policy used by the warehouse operator.

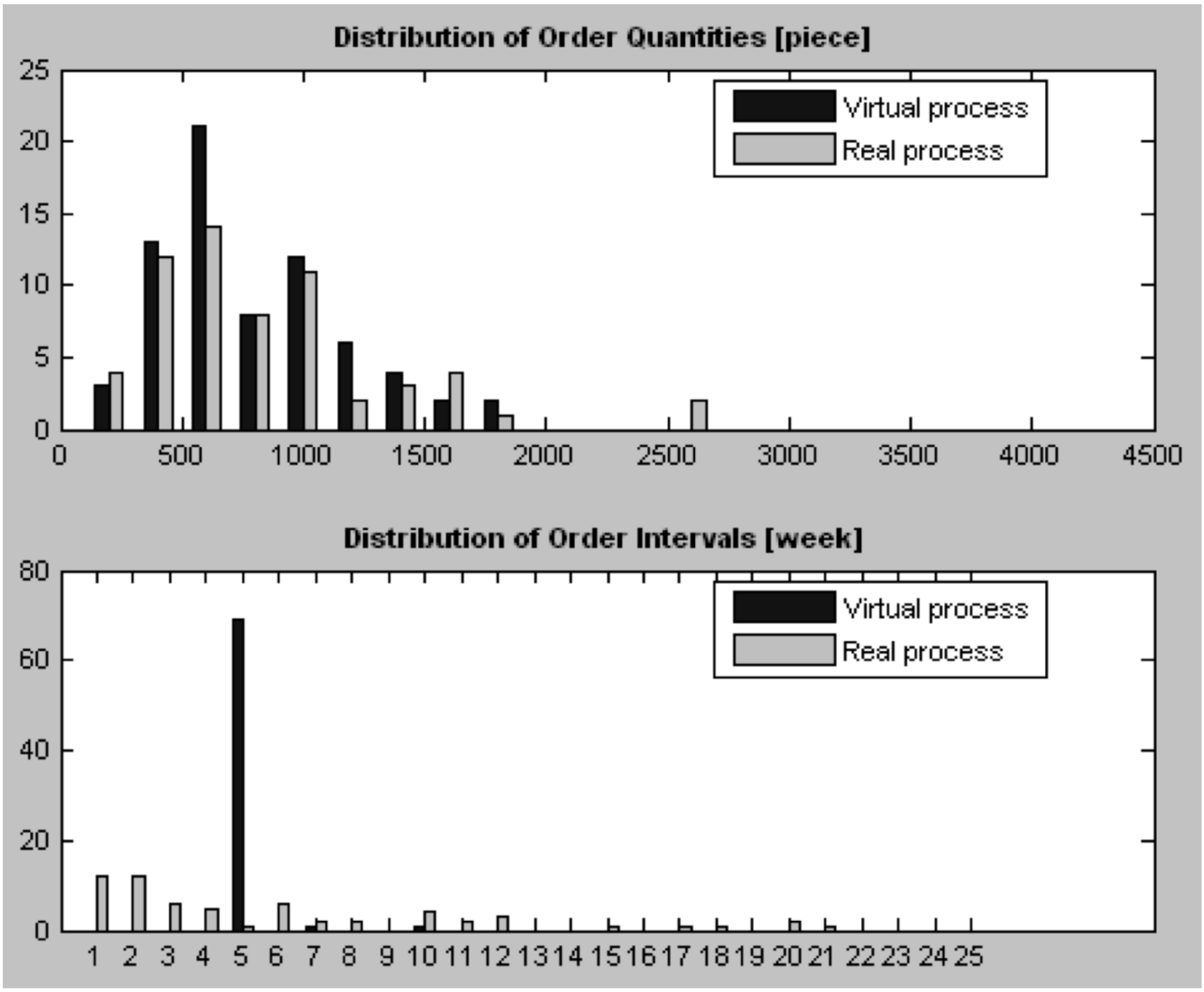


Figure 5: Comparison of ordering quantity and ordering intervals frequency distribution for a specific case

6 Conclusion

This paper researched the warehouse stock optimization using two optimization algorithms. The SD approach was used in modeling and validation of the warehouse model. The simulation model was built using Matlab – Simulink. The observed company's representative cases were analyzed and the simulation results show potential since a cost reduction of a few percents is usually considered a success. Nevertheless, the simulation model still needs to be validated meticulously, although the logical validation was already accepted by the research team.

The advantage of optimization with presented methods was in its speed with which we have achieved the results. The solution is available almost instantly and this is very important as the staff is usually under severe pressure and that makes the decision process more difficult, result in them making more mistakes. The FLE proved as a reliable decision support system in suggesting the proper replenishment algorithm, thus taking some more pressure off the staff. The consequence is the near-optimal material quantity in the warehouse, which assures undisturbed production and minimal holding costs.

Management today is faced with more decision factors than they feel they can cope with. Managers face great uncertainty about the operating environment and what could happen as a consequence of various decisions. The simulator presented in this paper can simulate various scenarios, undertake "what-if" analyses and help to determine potential outcomes and strategies, so as to reveal the truly best options.

Since the simulation results are promising, the research is still in progress in a way to fully implement inventory control decision support system in the company. Nevertheless, the presented simulator is useful as a learning tool for the warehouse operator to learn about new replenishment policies.

Acknowledgements

This research was supported by Ministry of Higher Education, Science and Technology of Republic of Slovenia (Program No. UNI-MB-0586-P5-0018). Our sincere thanks goes also to Mr. Valter Rejec and his team from Iskra Avtoelektrika d.d.

Literature

Arda, Y. & Hennet, J.C. (2004). Optimizing the ordering policy in a supply chain. *11th IFAC Symposium on Information Control Problem in Manufacturing (INCOM'2004)*, Salvador da Bahia (Brésil), 5-7 April 2004.

Cheng, L., Subrahmanian, E. & Westerberg, A.W. (2005). Multiobjective Decision Processes under Uncertainty: Applications, Problem Formulations, and Solution Strategies. *Industrial & Engineering Chemistry Research*, **44**: 2405-2415.

Forrester, J. W. (1961). *Industrial Dynamics*. MIT Press, Cambridge, MA.

Fowler, A. (2003). Systems modelling, simulation, and the dynamics of strategy. *Journal of Business Research*, **56**(2): 135-144.

Kljajić, M., Bernik, I. & Škraba, A. (2000). Simulation Approach to Decision Assessment in Enterprises. *Simulation,* **74**(4): 199-210.

Kofjač, D. & Kljajić, M. (2004). Simulation approach to the stochastic optimization of the warehouse. *16th International Conference on Systems Research, Informatics and Cybernetics,* Baden-Baden, Germany, July 29 - August 4, 2004.

Ljubič, T. (2000). *Planiranje in vodenje proizvodnje*. Moderna organizacija, Kranj.

Oakshott, L. (1997). *Business modelling and simulation*. Pitman Publishing.

Silver, E.A, Pyke, D.F. & Peterson, R. (1998). *Inventory management and production planning and scheduling*. John Wiley & Sons.

Tompkins, A. J. & Smith, J.D. (1998). *The warehouse management handbook*. Thompkins Press, 2nd edition.

Wang, Q. & Chatwin, C.R. (2004). Key issues and developments in modelling and simulation-based methodologies for manufacturing systems analysis, design and performance evaluation. *The International Journal of Advanced Manufacturing Technology*, **22**(8): 720-729.

Zadeh, L.A. (1995). Fuzzysets. *Information and Control,* **8**: 338-353.

Davorin Kofjač received his Associate's degree in computer science at the Faculty of Electrical Engineering and Computer Science in Maribor in 1999. He achieved his B.Sc. degree in information systems management at the Faculty of Organizational Sciences in Kranj in 2002. Currently he is working as a young researcher at the Faculty of Organizational Sciences. His research interests include modeling and simulation, artificial intelligence and operational research.

Miroljub Kljajić is Professoor at the Faculty of Organizational Sciences, University of Maribor. His brief biography is published on page 643 of this issue.

The Periodicity of the Anticipative Discrete Demand-Supply Model

Andrej Škraba[1], Davorin Kofjač[1], Črtomir Rozman[2]

[1]Univerza v Mariboru, Fakulteta za organizacijske vede, Kidričeva cesta 55a, SI-4000 Kranj
[2]Univerza v Mariboru, Fakulteta za kmetijstvo, Vrbanska 30, SI-2000 Maribor
{andrej.skraba, davorin.kofjac}@fov.uni-mb.si, crt.rozman@uni-mb.si

This paper presents an analysis of the periodicity solutions to the discrete anticipative cobweb model. Dubois' anticipative principle was applied in the modification of Kaldor's cobweb model. Characteristic solutions are gained through the application of a simulation, which determines the cyclical behaviour of supply and demand. Z-transform was applied in the determination of the solutions. The interconnection between the anticipative definition of the cobweb model and Hicks model is addressed.

Keywords: cobweb, hyperincursivity, system dynamics, anticipative system, nonlinear system, Farey tree, chaos

Obravnava periodičnosti rešitev anticipativnega diskretnega modela ponudbe in povpraševanja

V prispevku je izvedena analiza periodičnosti v diskretnem anticipativnem cobweb modelu. Pri modifikaciji Kaldorjevega cobweb modela je upoštevan Duboisov anticipativni princip. S pomočjo simulacije so prikazane karakteristične rešitve, ki opredeljujejo ciklična nihanja med ponudbo in povpraševanjem. Pri določitvi rešitev je bila uporabljena z-transformacija. V prispevku je naslovljena povezava med anticipativno definicijo cobweb modela in Hicksovim modelom.

Ključne besede: cobweb, hiperinkurzivnost, sistemska dinamika, anticipativni sistem, nelinearni sistem, Fareyevo drevo, kaos

1 Introduction

The cobweb model presents the adjustment to market supply and demand. It is typically viewed as the model of the agricultural pricing mechanism. The story behind the model might be briefly explained as follows: "The quantity offered for sale this year depends on what was planted at the start of the growing season, which in turn depends on last year's price. Consumers look at the current prices, though, when deciding to buy. The cobweb model also assumes that the market is perfectly competitive and that supply and demand are both linear schedules." For a clear and extensive introduction to the topic, see Rosser, 2003. The fact, that the cobweb model is in the field of discrete dynamics is rather an advantage, since the systems of difference equations are often easier to grasp. For example, in his enduringly valuable scholarly work on the studies of Dynamic Systems, Luenberger (1979) firstly addresses difference and differential equations later on. The model in question has all the characteristics of classical System Dynamics (SD) models: equilibrium, competitiveness, human perception, delay and adjustment, but somehow it avoids being included settled in the common SD model bank of each SD modeller. The main reason for the elusiveness of the cobweb model lies in its original form, which is not suitable for direct transformation to the common elements such as Level (L) and Rate (R). The functions of supply $Q_s(k)$ and demand $Q_d(k)$ can be specified in the form:

$$Q_d(k) = a + bP(k) \tag{1}$$

$$Q_s(k) = c + dP(k-1) \tag{2}$$

where *a*, *b*, *c* and *d* are parameters specific to the individual markets. $P(k)$ and $Q_s(k)$ should be restricted to the positive values. In the cobweb model it is assumed that producers supply a given amount in any one time period (determined by the previous time period's price) and then the price adjusts so that all the products supplied are bought by customers. If we write this in the form of an equation, then $Q_d(k) = Q_s(k)$, which enables us to state that the price is:

$$P(k) = \tfrac{d}{b}P(k-1) + \tfrac{c-a}{b} \tag{3}$$

Eqs. 1, 2 and 3 are not quite in the proper form for performing the transformation to the SD model. One of the things is the time argument (*k*-1). The other is the missing Rate (R) elements and the corresponding Δt. One should expect that the transformation will provide the known equations in the familiar form for the structure shown in Fig. 1. The model developed should enable us to examine the properties of the cobweb model and also to consider its structural and incursive perspective.

As the Wiener's cybernetics principle (Wiener, 1961) stands firm in the system's theory, the cobweb model principle stands as the basic linearized principle construct for the systems interaction dependence and will probably remain the basic starting tool for the quantitative analysis of complex systems.

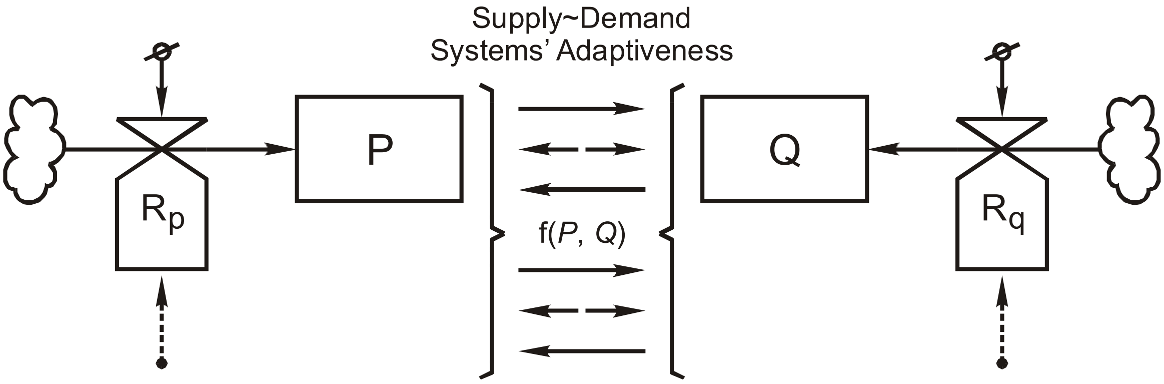


Figure 1. The main elements of the proposed cobweb SD structure

2 Transformation to the SD Form

Fig. 1 shows, that the price *P* and quantity *Q* of the goods should be stated as the Level elements depending on the Rates that determine the change in price and quantity supplied. The theory behind the cobweb model states that the quantity supplied at the present depends on the price in the past. Therefore the price and the quantity supplied should be dependant variables as illustrated in Fig. 1. Restating the above Equations, while eliminating the time argument *k*-1, gives us the following set of equations:

$$Q_d(k+1) = a + bP(k+1) \tag{4}$$

$$Q_s(k+1) = c + dP(k) \tag{5}$$

$$P(k+1) = \tfrac{d}{b}P(k) + \tfrac{c-a}{b} \tag{6}$$

Eqs. 4, 5 and 6 will enable the determination of the Rates elements. Let us determine the Rate element for the change of Price $R_p(k)$. As the equations are in the different form the Rate will be determined as *R*(*k*)=*L*(*k*+1)-*L*(*k*):

$$R_P(k) = P(k+1) - P(k) = \tfrac{c+dP(k)-Q_s(k}{b} \tag{7}$$

A little more work will be needed for the $R_Q(k)$ since special time considerations had to be taken. We will apply the time arguments of *k*+1 and *k*+2 in order to loose the *k*-1 argument that is present in Eq. 2:

$$R_Q(k) = Q_s(k+2) - Q_s(k+1) = d\tfrac{c+dP(k^*)-Q_s(k^*)}{b} \tag{8}$$

Since the time *k** argument with the consideration of Eq. 1 and 2 actually represents the past, i.e. the *k*-1 argument, we should state the equations for *P*(*k*-1) and Qs(*k*-1). Eq. 3 will enable us to state *P*(*k*-1):

$$P(k-1) = \tfrac{bP(k)-c+a}{d} \tag{9}$$

$$Q_s(k-1) = a + P(k-1)\ b \tag{10}$$

Eq. 10 is set by the fact that Qd(k) = Qs(k) and Eq. 4. The consideration of the k-1 time argument is necessary in order to perform calculations in the model. At each time step the previous values are needed in order to perform the calculation. By inserting

$$P(k-1) = \tfrac{bP(k)-c+a}{d}$$

in Eq. 10 we get:

$$Q_s(k-1) = a + \tfrac{b^2P(k)-bc+ab}{d} \tag{11}$$

By inserting Eq. 11 and 9 into Eq. 8 we get a simplified form of the rate equation:

$$R_Q(k) = -a + c - (b-d)P(k) \tag{12}$$

As the result of the above algebraic manipulation, the cobweb model could be stated in the standard SD form:

$$P(k+1) = P(k) + \Delta t R_P(k) \tag{13}$$

$$R_P(k) = \tfrac{c+dP(k)-Q_s(k)}{b} \tag{14}$$

$$Q_s(k+1) = Q(k) + \Delta t R_Q(k) \tag{15}$$

$$R_{Q_s}(k) = -a + c - (b-d)P(k) \tag{16}$$

with the starting conditions $P(0) = \tfrac{x-a}{b}$ and Qs(0)=*x* where *x* represents the starting perturbation of the model. In the above set of equations the Δ*t* is introduced which is not present in the formulation of the classical cobweb model. If Δ*t*=1 then the model is equivalent to the classical cobweb.

Fig. 2 shows the SD structure of the cobweb model corresponding to Eqs. 13, 14, 15 and 16. There are two levels represented, *P* and *Qs*, and two rate elements, R_P and R_{Qs}. The model's behaviour is determined by the input parameters *a*, *b*, *c* and *d* as well as by the perturbation parameter *p*. The element P_0 represents the initial value of the level element *P*. The initial value of the level element *Qs* is equal to the arbitrary value of perturbation *p*.

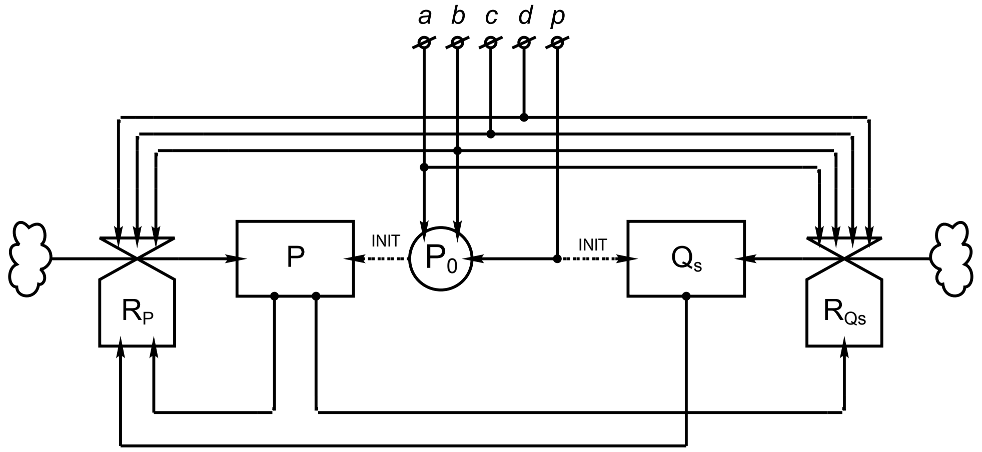


Figure 2. System Dynamics structure of the cobweb model

The response of the classic cobweb model developed by SD methodology is shown in Fig. 3 and Fig. 4. The parameter values applied and the description of the system's response are shown in Table 1.

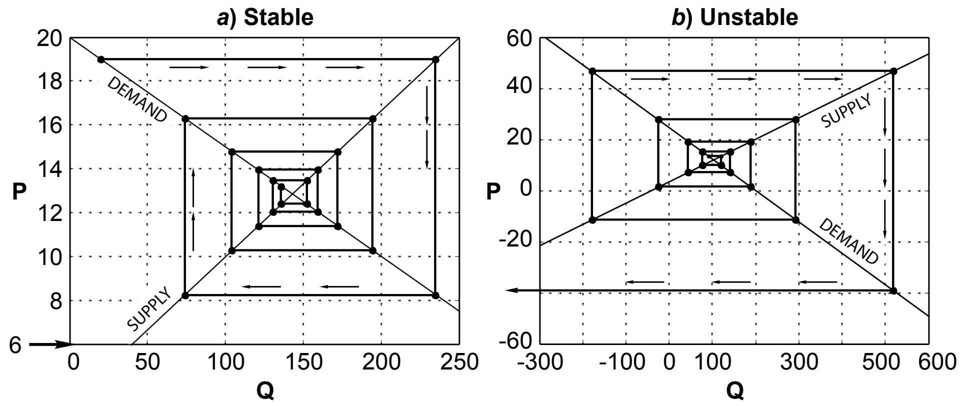


Figure 3. Response of the SD cobweb model: a) Stable, b) Unstable

Table 1. Parameter values

a	b	c	d	p	Response
400	-20	-50	15	20	Stable
200	-8.1	-43	12	90	Unstable
160	-2	-20	2	55	Dyn. Stable

There are three possibilities: a) a Stable system, where the supply and demand converge, b) an Unstable system where the supply and demand diverge and c) the Dynamically stable system shown in Fig. 4, where the price and demand neither converge nor diverge.

A dynamically stable response indicates the periodical solution that will be of interest in further examination of the model. In general a solution y_n is periodic if $y_{n+m} = y_n$ for some fixed integer of *m* and all of *n*. The smallest integer for *m* is called the period of the solution. In our case the solution in Fig. 4 is a two-cycle solution and in general the following definition will be applied (Shone, 1997):

Definition 1. If a sequence $\{y_t\}$ has, for example, two repeating values y_1 and y_2, then y_1 and y_2 are called period points and the set $\{ y_1, y_2 \}$ is called a periodic orbit.

This periodical response of the system is important because real agricultural systems depend on the cyclic behaviour and could be controlled only by regarding the periodicity of such systems. Examples from real cases could easily be found in crops as well as in the stock.

1 Separation of the Structural Elements

The structure of the model in Fig. 2 shows that the Price and Quantity are related. However the structure can be repre-

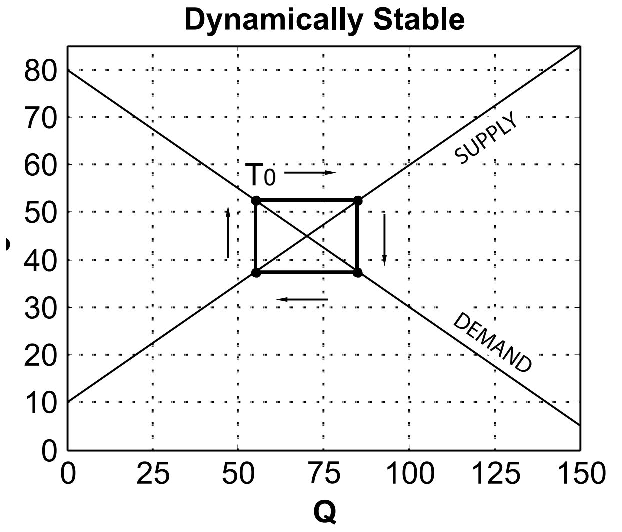


Figure 4. Response of the SD cobweb model: Dynamically stable

sented in a different way. By transforming the cobweb model into an SD form, the model could become non-autonomous depending on the variable Δ*t*. The following two equations represent the different formulation of the cobweb model:

$$Q_s(k+1) = c + d\frac{Q_s(k)-a}{b} \tag{17}$$

$$P(k+1) = \frac{c+dP(k)-a}{b} \tag{18}$$

This reformulation represents Qs and *P* as the non-related quantities. The only boundary that exists are the coefficients. The rate elements should be determined in order to formulate the complete SD model:

$$R_P(k) = P(k+2) - P(k+1) = \frac{c+dP(k+1)-a}{b} - \frac{c+dP(k)-a}{b} = \frac{d}{b}(P(k+1)-P(k)) \tag{19}$$

$$R_{Q_s}(k) = Q_s(k+2) - Q_s(k+1) = \frac{d}{b}(Q_s(k+1) - Q_s(k)) \tag{20}$$

In order to meet the initial conditions of the model, the Q_S(*k*-1) should be determined:

$$Q_s(k-1) = a + \frac{b}{d}(Q(k) - c) \tag{21}$$

Equations for *P* and Q_S in standard SD form are as follows:

$$P(k+1) = P(k) + \Delta t R_P(k) \tag{22}$$

$$R_P(k) = \frac{d}{b}\left(P(k) - \frac{bP(k)-c+a}{d}\right) \tag{23}$$

$$Q_s(k+1) = Q_s(k) + \Delta t R_{Q_s}(k) \tag{24}$$

$$R_{Q_s}(k) = \frac{d}{b}\left(Q_s(k) - \left(a + \frac{b}{d}(Q_s(k) - c)\right)\right) \tag{25}$$

Eqs. 22, 23, 24 and 25 represent the cobweb model in a separated SD form as shown in Fig. 5. Note that the terms for *P* and Q_S are related only to the coefficients *a*, *b*, *c*, *d* and *p*. *P*(*k*+1) is dependent only on the value of *P*(*k*) and the coefficients *a*, *b*, *c*, *d* and *p*, but not on Q_S. Respectively for the Q_S(*k*+1).

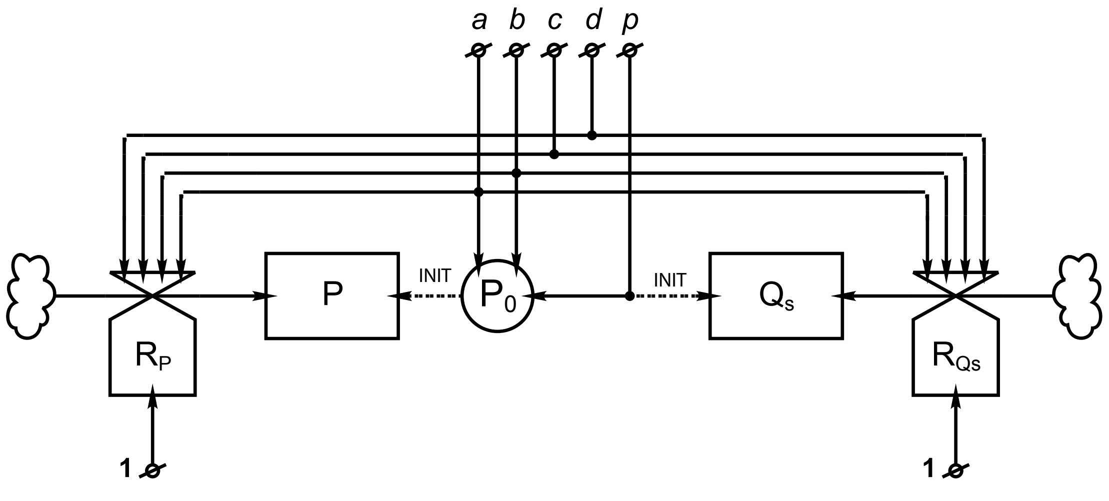


Figure 5. A cobweb model in SD form ~ separated elements

2 Anticipative Formulation

Comparison of the structures shown in Fig. 2 and Fig. 5 indicates that *P* and Q_S depend only on the parameter values of *a*, *b*, *c*, *d* and *p*, i.e. on the initial conditions. Eqs. 22, 23, 24 and 25 enable the determination of the entire anticipative (future event) chain, while the equation:

$$P(k-1) = \frac{bP(k)-c+a}{d} \tag{26}$$

and Eq. 21 enable the determination of the feedback (past event) chain. The representation of the Feedback ~ Anticipative chain is shown in Fig. 6. The dynamics of interest are therefore the chain's dynamics that are dependant on the parameters *a*, *b*, *c*, *d* and *p*. Both chains are actually dependant on strategy dynamics which could be formulated as f(*a*, *b*, *c*, *d*, *p*, *t*).

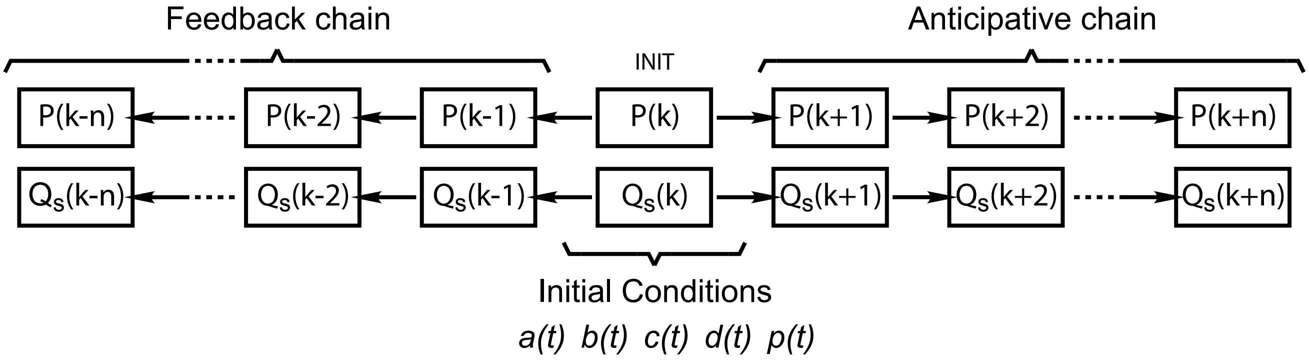


Figure 6. Feedback ~ Anticipative chain

Application of the hyperincursive algorithm and inspection of the equations gained with Dubois' (Dubois and Resconi, 1992) formulation of logistic growth and previous research (Kljajić, 1998; Kljajić, 2001, Kljajić et al., 2005), yields the following set of equations for the hyperincursive cobweb model:

$$P(k+2)=\frac{d}{b}\left(A-\left(\frac{bB-c+a}{d}\right)\right) \tag{27}$$

$$Q_s(k+2)=\frac{d}{b}\left(C-a-\frac{b}{d}\left(D-c\right)\right) \tag{28}$$

with the initial conditions:

$$P_0(k+1)=\frac{p-a}{b} \tag{29}$$

$$P_0(k)=\frac{bP_0(k+1)+a-c}{d} \tag{30}$$

$$Q_{s0}(k+1)=p \tag{31}$$

$$Q_{s0}(k)=a+\frac{b}{d}\left(Q_{s0}(k+1)-c\right) \tag{32}$$

The coefficients *A* and *B* in Eq. 27 could be replaced by the terms *P*(*k*+1) or *P*(*k*) while the coefficients *C* and *D* in Eq. 28 can be replaced by $Q_S(k+1)$ or $Q_S(k)$. This yields 16 different combinations of system defined by Eqs. 27 and 28 that should be studied. The appropriate explanation of the modified system structure should follow the techniques of graphical solutions for homogenous difference equations (Puu, 1963; Azariadis, 1993).

The system combination further examined will have the following terms: *A*=*P*(*k*+1), *B*=*P*(*k*) *C*= $Q_S(k+1)$ and *D*= $Q_S(k)$. This yields the following set of equations:

$$P(k+2)=\frac{d}{b}\left(P(k+1)-\left(\frac{bP(k)-c+a}{d}\right)\right) \tag{33}$$

$$Q_s(k+2)=\frac{d}{b}\left(Q_s(k+1)-a-\frac{b}{d}\left(Q_s(k)-c\right)\right) \tag{34}$$

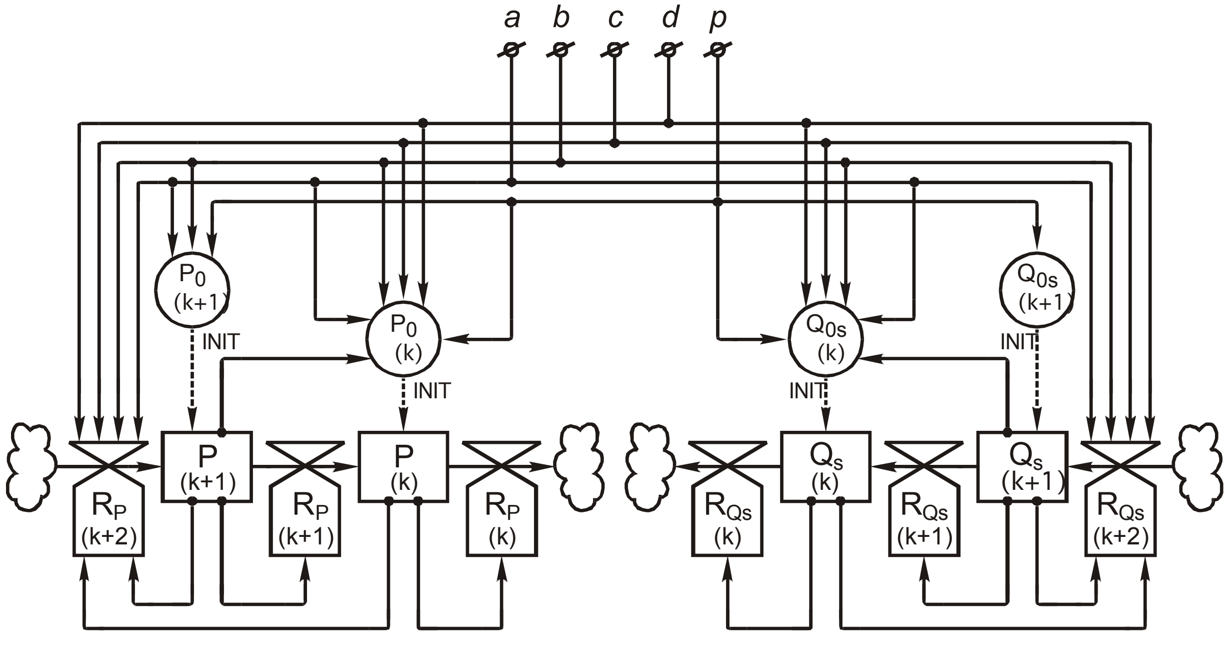


Figure 7. Structure of the hyperincursive Cobweb model; Euler integration, Δt=1

Eqs. 33 and 34 with the initial conditions stated by Eqs. 29 ~ 32 could be modelled as shown in Fig. 7. The structure represents the cobweb model in a hyperincursive form modelled using classic SD elements. The Euler integration method is applied with the time-step Δt=1.

Eqs. 33 and 34 could be reformulated in order to show the dependency of the future-present-past events as follows:

$$P(k) = \frac{bP(k-1)+a-c}{d} + \frac{b}{d}P(k+1) \tag{35}$$

$$Q_s(k) = \frac{b}{d}Q_s(k+1) + \frac{b}{d}Q_s(k-1) + a - \frac{bc}{d} \tag{36}$$

Eqs. 35 and 36 state that the value of the present is dependent on the past as well as on the future. This paradoxical statement is realizable since the formulation of a feedback ~ anticipative chain could be stated. Fig. 7 has two delay chains, one for *P* and one for Q_S. One might note that the level and rate elements are dependant only on the coefficients and initialization values.

3 The Periodicity of the System

The *z*-transform is the basis of an effective method for the solution of linear constant-coefficient difference equations. It essentially automates the process of determining the coefficients of the various geometric sequences that comprise a solution (Luenberger, 1979). The application of the *z*-transform on Eqs. 33 and 34, with initial conditions stated by Eqs. 29 ~ 32, gives:

$$Y(z) = \frac{-y_1 z + y_0 dz - y_0 z^2}{-1+dz-z^2} \tag{37}$$

An inverse *z*-transform yields the following solution:

$$\begin{aligned} Y^{-1}(z) = {} & 2^{-1-n}\, y_0 \left(d - \sqrt{-4+d^2}\right)^n - \frac{y_1\left(d-\sqrt{-4+d^2}\right)^n}{2^n\sqrt{-4+d^2}} + \\ & + \frac{2^{-1-n}\, y_0\, d\left(d-\sqrt{-4+d^2}\right)^n}{\sqrt{-4+d^2}} + 2^{-1-n}\, y_0 \left(d + \sqrt{-4+d^2}\right)^n + \\ & + \frac{y_1\left(d+\sqrt{-4+d^2}\right)^n}{2^n\sqrt{-4+d^2}} - \frac{2^{-1-n}\, y_0\, d\left(d+\sqrt{-4+d^2}\right)^n}{\sqrt{-4+d^2}} \end{aligned} \tag{38}$$

The following equation should be solved in order to acquire the conditions for the periodic response of the system:

$$Y^{-1}(z) = y_0 \tag{39}$$

Let us compute a numerical example of a periodic solution by applying the *z*-transform. The period examined will be the period of 9 i.e. *n*=9. One should insert the condition *n*=9 into Eq. 39. The following possible solution for the initial condition is worth examining:

$$d = \frac{1}{\left(\frac{1}{2}(-1+i\sqrt{3})\right)^{\frac{1}{3}}} + \left(\frac{1}{2}(-1+i\sqrt{3})\right)^{\frac{1}{3}} \tag{40}$$

The term $(-1+i\sqrt{3})^{\frac{1}{3}}$ (let us denote the term as z^*) could be expressed in the following way using three different imaginary values in polar form:

$$z_1^* = \sqrt[3]{2}\left(\cos\frac{2\pi}{9} + i\sin\frac{2\pi}{9}\right) \tag{41}$$

$$z_2^* = \sqrt[3]{2}\left(\cos\frac{8\pi}{9} + i\sin\frac{8\pi}{9}\right) \tag{42}$$

$$z_3^* = \sqrt[3]{2}\left(\cos\frac{14\pi}{9} + i\sin\frac{14\pi}{9}\right) \tag{43}$$

By inserting Eqs. 41, 42 and 43 into Eq. 40 and performing a trigonometric reduction, one gets the following solutions:

$$d_1 = 2\cos\frac{2\pi}{9} \quad d_2 = 2\cos\frac{4\pi}{9} \quad d_3 = 2\cos\frac{8\pi}{9} \tag{44}$$

By inspecting Eq. 40 and considering the equation for the roots of complex numbers (Kreyszig, 1997):

$$\sqrt[n]{z} = \sqrt[n]{r}\left(\cos\frac{\theta+2k\pi}{n} + i\sin\frac{\theta+2k\pi}{n}\right) \tag{45}$$

the general form of the solution for the parameter *d* could be assumed:

$$d = 2\cos\frac{2\pi m}{n} \tag{46}$$

where *n* is the period and *m* = 1, 2, 3, …, *n*-1. A similar procedure could be performed for the arbitrary period of *n*. More general solutions, which also regard the parameter, *b* that was fixed for the purpose of determining the solutions, is:

$$d = 2b\cos\frac{2\pi m}{n} \tag{47}$$

In some cases the solutions could be expressed in an alternative algebraic or trigonometric form. Table 2 below shows the solutions for the parameter *d* up to the period of *n*=12. Alternative solutions could be expressed as the roots of the polynomial. The table incorporates the Shape symbols, which are important in the study of the response of dynamical systems. This is especially the case in the examination of complex nonlinear dynamical systems (Sonis, 1999; Matsumoto, 1997; Hommes, 1998). Mapping of the system and the visualization of the periodic solution is therefore important for the analysis of periodic or chaotic solutions of differential and discrete difference equations. The description of the shape is taken from the vocabulary of proper shapes although the response of the system is primarily not proper. The numerical values of the solutions for parameter *d* are important since these values also confirm the findings of Sonis (1999) on the domain of attraction for 2D dynamics by *n*-dimensional linear bifurcation analysis. One of the important conditions gained by the proposed inspection is the value of the period *n*=10, which is in close relation to the period *n*=5. The value of the parameter *d* is

$d = \frac{\sqrt{5-1}}{2}$ with the numerical value being *d*=1.61803…

This solution represents the "*Golden Ratio*" (Φ). Some of the different representations of the solution for the value of parameter *d* with period of *n*=10 are:

$$d_{10} = \phi = 2\cos\frac{\pi}{5} = \frac{1}{2}(1+\sqrt{5}) = 1.61803398874 \tag{48}$$

The first solution of the parameter *d* with the period of *n*=10 connects the discrete system considered with the Fibonacci numbers given by the infinite series:

$$d_{10} = \phi = 1 + \sum_{u=1}^{\infty} \frac{(-1)^{u+1}}{F_u F_{u+1}} \tag{49}$$

The fact that the periodicity conditions of the discrete system examined incorporates the golden ratio number Φ, could be observed in other studies (Brock and Hommes, 1997) of complex nonlinear expansions of the basic cob-web systems, e.g. in Brock and Hommes "Almost Homoclinic Tangency Lemma". One should expect that the symmetrical response in *n*-mapping should follow the pattern with a synchronization match, e.g. in a certain point of the solution. The source of the condition mentioned is presented using the above procedure. (The value of parameter *d* for the period of *n*=5 is $d = \frac{\sqrt{5-1}}{2}$). The periodicity conditions are similar to the parameter values gained for the domain of *2-d* dynamic attraction by Sonis (1999). This set of parameters is augmented with two values for the periods 8 and 12, which are not stated in (Sonis, 1999).

4 Nonlinear Setup and Results

The system's responses were gained according to the parameter values gathered in Table 2. The changes were made to parameter *d,* which yielded the synchronization patterns as shown in the shape column. The parameter values were gained from the simulation, where the range of parameter *d* was set at [-40, 40] with Δ*d*=0.001. The condition for the determination of the parameter values was set by the rule of acceptable error between steps of the simulation and definition 1 of the synchronization.

Table 2. Parameter values at sync.

desc.	a	b	c	d	p
Triangle	400	-20	-50	20.0000	160
Quadrangle	400	-20	-50	-0.0010	160
Pentagon	400	-20	-50	-12.3671	160
Pentagram	400	-20	-50	32.3620	160
Hexagon	400	-20	-50	-20.0000	160
Nonagram	400	-20	-50	-6.9450	160
Hexagram	400	-20	-50	15.3070	160

Let us consider the following expansion of the model (Škraba et al., 2005, Škraba et al., 2006) and let us define the adaptive nonlinear rule *R* as:

$$R = \begin{cases} \frac{P_{k+1} - P_k}{P_k} & if \quad -1 < \frac{P_{k+1} - P_k}{P_k} < 1 \\ 1 & if \quad \frac{P_{k+1} - P_k}{P_k} \geq 1 \\ -1 & if \quad \frac{P_{k+1} - P_k}{P_k} \leq -1 \end{cases} \tag{50}$$

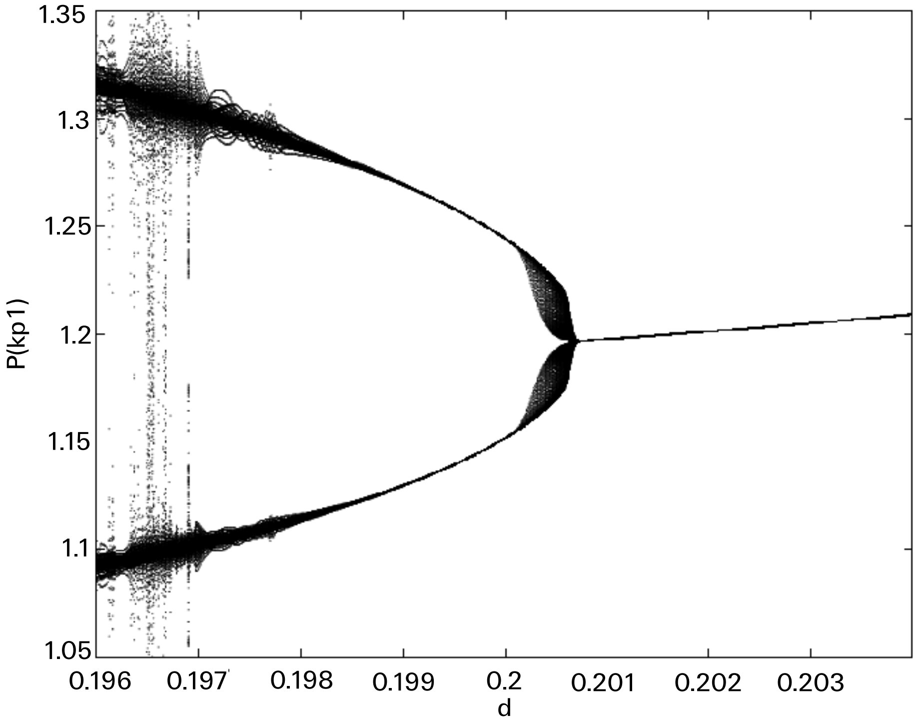


Figure 8. Bifurcation diagram in the range d □ [0.196, 0.204]

Since the forced nonlinear rule has been applied, we have arrived at the characteristic nonlinear bifurcation diagram. Let us observe the response of the system at the period of *p*=6 as one of the polygon rules, which should provide the periodicity of the system considered. The beginning of the bifurcation corresponds to the value of the parameter *d*=1, which has been indicated in the analysis of the initial 2-d discrete map. Period six is followed by *p*=7 and *p*=8. However, the analytical proof of the periodicity would be hard since the underlying Farey sequence defines the adapted nonlinear *2-d* discrete map. Such evidence is also found in other works on nonlinear system analysis, for example (Brock and Hommes, 1997; Gallas and Nusse, 1996) or in the recent works of Puu (Puu and Shushko, 2004; Puu, 2005).

Consider another generic alteration of the initial anticipative cobweb model:

$$\begin{aligned}
&P_K(k+1) = P_K + P_{KP1}(k) - \left(P_K(k) + \tfrac{1}{P_Z(k)P_K(k)}\right)\\
&P_{KP1}(k+1) = P_{KP2}(k)\\
&P_{KP2}(k+1) = \tfrac{d}{b}\left(P_{KP1} - \tfrac{bP_K(k)-c+a}{d}\right)\\
&P_Z(k+1) = P_Z(k) + P_K(k)P_{KP1}(k) - vP_Z(k) \qquad (51)
\end{aligned}$$

Slight modification of initial Hicks' model (Puu, 1963) gives this interesting response. The system can be represented in three dimensions, which reveals the periodicity of the system for which the previously determined conditions of the Farey tree generally still hold. Fig. 9 shows the *3d* bifurcation diagram for the altered model. You can see the four attractors, which are simultaneous and represent the four possible equilibrium states for the trade dynamics. This 4-cycle characteristic is preserved during the alteration of the parameter *d,* which can be observed in the Fig. 10. The four dots on the centre-right side of the figure represent the four-cycle characteristic of the response. The larger orbits indicate the steep change in the modus of the system.

In order to analyze the preservation of the periodic solutions, the most significant periodic solution, i.e. the period of 6, has been applied to the system in Eq. 50, which is restated in the following form:

$$\begin{aligned}
&P_{KP1} = \tfrac{d}{b}P_{KP1} - P_K + c - a\\
&P_K = P_{KP1} - \tfrac{1}{P_Z P_K}\\
&P_Z = P_Z(1-v) + P_{KP1}P_K \qquad (52)
\end{aligned}$$

This proposed model, with certain limitations, yields the periodicity solutions that are related to the system attractor. For example, for period 10 the initial values are: P_{KP1}=-1, P_{KP1}=-1.61803, P_Z=1.61803, *d*=1.61803 and *v*=1. Fig. 11 represents the response of the system for period 6, where the parameter *d*=1. The starting points of the attractor are from the interval (-2, 2). The simulation for the determination of the attractor was performed using 30,000 random starting points. Fig. 12 shows the period 6 attractor with 6 attractive regions, which are doubled, actually making 12 beams of periodicity. The centre of the attractor reveals the distinguishing *6-sided* polygon shape. Fig. 12 shows a magnification of the centre of the period 6 attractor.

5 Conclusion

The story revealed from the hyperincursive model developed here raises the following questions: a) Does a change in the strategy change the structure or does it only change the relations between the elements of that structure? b) Does a change in the strategy change the future as well as the past?

A change in the strategy would mean a new and differ-

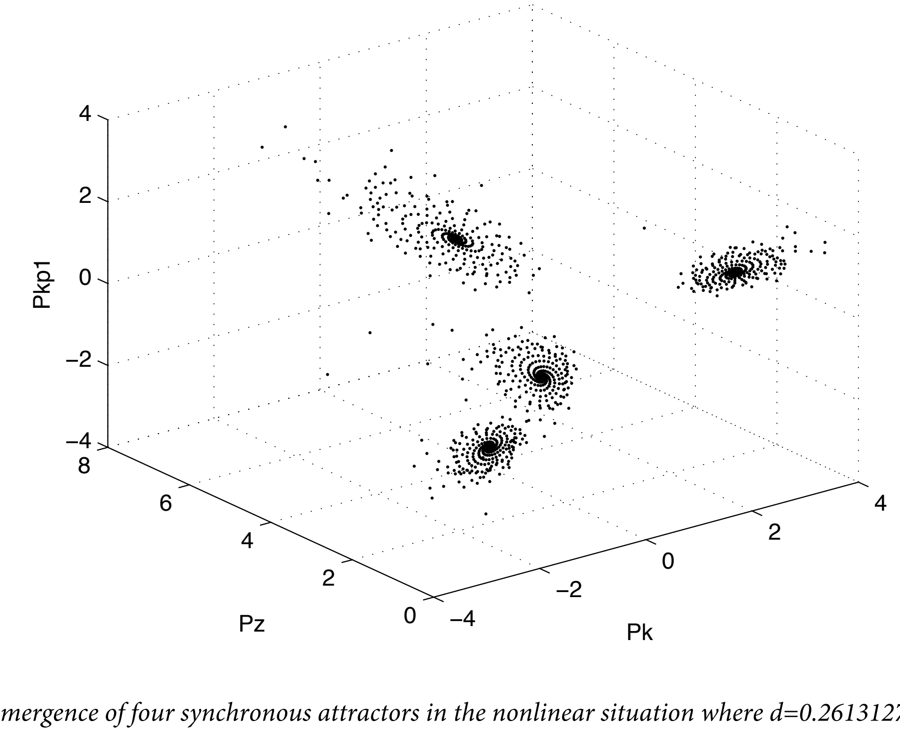


Figure 9. The emergence of four synchronous attractors in the nonlinear situation where d=0.26131278 and b=0.33

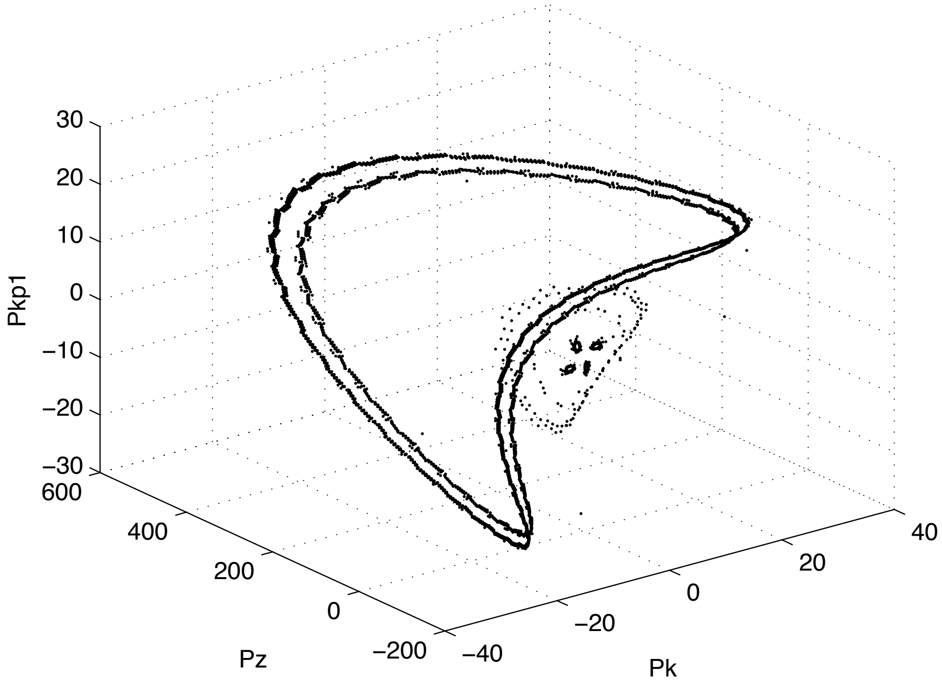


Figure 10. The preservation of four synchronous attractors in the nonlinear situation where d=0.26151152 and b=0.33

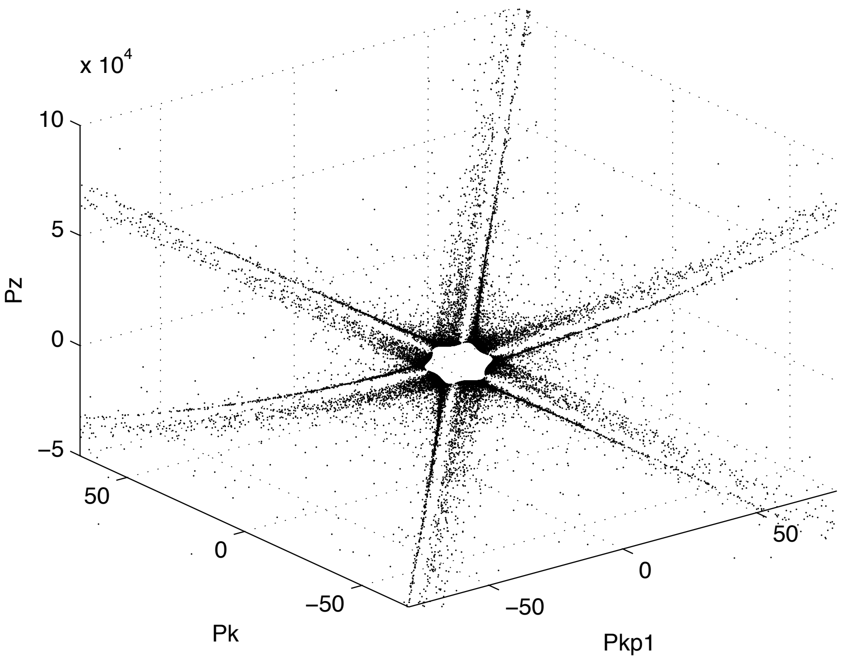


Figure 11. The period 6 attractor with the parameters: d=1, b=1 and v=1

ent future and should also mean a different past if the strategy change occurred earlier. The hyperincursive cobweb model consideration enables us to change the future as well as the past chain of events. However, a different examination of the system dynamics is proposed where change in the key parameters is performed while observing the change in a complete future and past chain rather than observing the classical time response of the system.

The following procedure proposition emerges, which enables the anticipative formulation of the classical dynamic system. Since the hyperincursive systems are hard to determine (Dubois and Resconi, 1992; Rosen, 1985), the anticipatory mechanisms developed should be applied. Therefore, the model should a) be transformed into the separated form, b) provide the property of the past-future chain and c) apply the hyperincursive structure to the model studied.

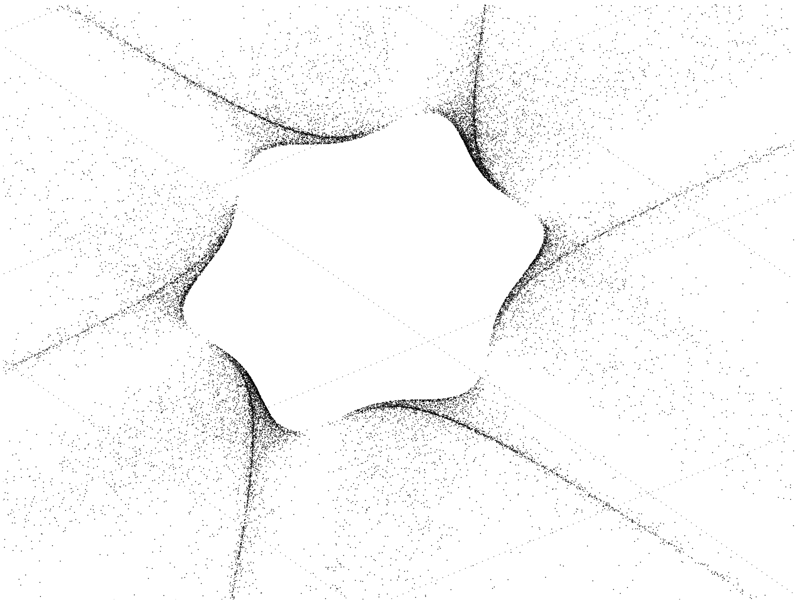


Figure 12. A magnification of the centre of the period 6 attractor from Fig. 11 with the parameters: d=1, b=1 and v=1

The model developed shows that, by the statement of the general rule of the system, the synchronization of the entire feedback-anticipative chain could be gained by setting the appropriate strategy in the form of the value of the parameters set, which should be time dependant.

The idea for the simulation proposed in this paper is quite different from the common paradigm. The structure of the model should yield the entire feedback-anticipative chain and observation should be made of the entire system response. This provides new and quite challenging responses that should initiate further interest and examination of the proposed model.

One of the interesting responses from the model is the helix like shape that is synchronized at certain time steps. The entire feedback-anticipative chain, i.e. all the point set, is synchronized according to the period of the system. The solution of the periodicity conditions for the *2-d* discrete linear cobweb map provided the means to determine these periodicity conditions and an analytical approach using *z-transformation* provides the proper way to determine the periodic solutions. The emergence of a Farey tree as the rational fraction representation yields the organization of the periodicity solutions. The model developed shows that, by the statement of the general rule of the system, the synchronization of the entire feedback-anticipative chain could be achieved by setting the appropriate strategy in the form of the value of the parameters set, which should be time dependant. The bifurcation experiment with the nonlinear mapping provided the example of periodicity transposition to systems of higher complexities. Period 6 has been determined as one of the most stable periodic solutions, as has been explicitly shown by the analysis of system's attractors.

Literature

Azariadis C. (1993). *Intertemporal Macroeconomics*. Blackwell Publishers. Cambridge MA.

Brock, W. A. & Hommes, C. H. (1997) A rational route to randomness. *Econometrica* **65**(5): 1059 – 1095.

Dubois D. & Resconi G. (1992). *Hyperincurisvity - a new mathematical theory*, 1st ed. Presses Universitaires de Liège.

Gallas, J. & Nusse, H. E. (1996). Periodicity versus chaos in the dynamics of cobweb models. *Journal Of Economic Behavior & Organization*, **29**: 447 – 464.

Hommes, C. H. (1998). On the consistency of backward-looking expectations: The case of the cobweb. *Journal Of Economic Behavior & Organization*, **33**: 333 – 362.

Kljajić M. (1998). Modeling and Understanding the Complex System Within Cybernetics. Ramaekers M. J. (Ed.), *15th International Congress on Cybernetics.* Association International de Cybernetique. Namur.pp. 864 – 869.

Kljajić M. (2001). Contribution to the meaning and understanding of anticipatory systems. Dubois D. M. (Ed.). *Computing anticipatory systems*, (AIP conference proceedings, 573). Melville (New York): American Institute of Physics, pp. 400 – 411.

Kljajić, M., Škraba, A., Kofjač, D. &. Bren, M., (2005). Discrete Cobweb Model Dynamics with Anticipative Structure. *WSEAS Transactions on Systems.* **4**(5): pp. 196-203.

Kreyszig, E. (1993) *Advanced Engineering Mathematics.* 7th ed. John & Wiley Sons Inc.

Luenberger, D. G. (1979). *Introduction to Dynamics Systems - Theory, Models & Applications.* John & Wiley Sons Inc.

Matsumoto, A. (1997). Ergodic cobweb chaos. *Discrete Dynamics in Nature and Society,* **1**: 135 – 146.

Puu T. (1963). A Graphical Solution of Second-Order Homogeneous Difference Equations. *Oxford Economic Papers.* Mar 1963; **15**(1): 53-58.

Puu T. (2005). On the Genesis of Hexagonal Shapes Tönu Puu. *Networks and Spatial Economics.* **5**(1): 5-20.

Puu T. & Sushko I. (2004). A business cycle model with cubic nonlinearity. *Chaos, Solitons and Fractals,* **19**: 597–612.

Rosen R. (1985). *Anticipatory Systems.* Pergamon Press.

Rosser. M. (2003). *Basic Mathematics for Economists*, 2^{nd} ed. Routledge.

Shone. R. (1997). *Economic Dynamics*. Cambridge University Press.

Škraba, A., Kljajić, M., Kofjač, D. Bren, M. & Mrkaić M. (2005). Periodic Cycles in Discrete Cobweb Model. *WSEAS Transactions on Mathematics*, **3**(4): 196-203.

Škraba, A., Kljajić, M., Kofjač, D. Bren, M. & Mrkaić M. (2006). Anticipative Cobweb Oscillatory Agents and Determination of Stability Regions by Lyapunov Exponents. *WSEAS Transactions on Mathematics*, **5**(12): 1282-1289.

Sonis, M. (1999). Critical Bifurcation Surfaces of 3D Discrete Dynamics. *Discrete Dynamics in Nature and Society*, **4** (3). 333-343.

Wiener N. (1961). *Cybernetics or Control and Communication in the Animal*. MIT Press: Cambridge, MA.

Acknowledgement

This research was supported by the Ministry of Higher Education, Science and Technology of the Republic of Slovenia (Program No. UNI-MB-0586-P5-0018).

Andrej Škraba obtained his Ph.D. in the field of Organizational Sciences from the University of Maribor. He works as a researcher, teaching assistant and assistant professor at the University of Maribor, Faculty of Organizational Sciences in Cybernetics and DSS Laboratory. His research interest covers modelling and simulation, system dynamics, decision processes and multiple criteria decision-making. Recently his interests have been focused on the analysis of nonlinear chaotic systems.

Davorin Kofjač received his Associate's degree in computer science at the Faculty of Electrical Engineering and Computer Science in Maribor in 1999. He achieved his B.Sc. degree in information systems management at the Faculty of Organizational Sciences in Kranj in 2002. Currently he is working as a young researcher at the Faculty of Organizational Sciences. His research interests include modeling and simulation, artificial intelligence and operational research.

Črtomir Rozman achieved his PhD. at University of Maribor, Faculty of Agriculture. He is active as Assistant Professor for Farm management and Agronomic information systems in the Department for Agriculture Economics and Rural Development (Faculty of Agriculture, University of Maribor). His research includes development of DSS for farm management (simulation modelling, multi criteria decison analysis) and economics of agricultural production with the emphasis on organic agriculture. He is also involved in teaching activities as Head of Department and as thesis supervisor at post graduate study programmes. He is author or coauthor of 121 papers including 6 papers in journals with impact factor.

A Model of E-documentation of Community Nursing

Uroš Rajkovič[1], Olga Šušteršič[2], Jože Zupančič[1]

[1]University of Maribor, Faculty of Organisational Sciences,
Kidričeva cesta 55a, SI-4000 Kranj, Slovenia, uros.rajkovic@fov.uni-mb.si, joze.zupancic@fov.uni-mb.si
[2]University of Ljubljana, College of Health Studies, Poljanska cesta 26a, SI-1000 Ljubljana, Slovenia, olga.sustersic@vsz.uni-lj.si

This article presents the development of electronic documentation for community nursing using a system approach. Documentation is viewed as an information model for organizing and managing processes. The community nurse plans the nursing process after gathering and evaluating information on the patient's health and his/her family status. Documentation is thus considered to be a basis for the successful work of the health team and as a way of ensuring quality in nursing. The article describes a prototype software model for e-documentation in community nursing together with its evaluation in practice.

Key words: nursing, community nursing, modelling, documentation, software solution

Model e-dokumentacije zdravstvene nege v patronažnem varstvu

Prispevek predstavlja sistemski pristop k oblikovanju elektronske dokumentacije zdravstvene nege v patronažnem varstvu. Dokumentacija predstavlja informacijski model, ki je namenjen organiziranju in upravljanju procesov. Na osnovi informacij, zbranih z ocenjevanjem zdravstvenega stanja pacienta oz. stanja razmer v družini, medicinska sestra načrtuje proces zdravstvene nege. Dokumentacija je temelj za uspešno delo zdravstvenega tima pa tudi element zagotavljanja kakovosti zdravstvene nege. V prispevku bo opisan prototipni model informacijske rešitve za izvajanje e-dokumentacije v patronažnem varstvu.

Ključne besede: zdravstvena nega, patronažno varstvo, modeliranje, dokumentacija, programska rešitev

1 Introduction

A system approach in the organizational and informational context brings specific challenges in terms of the complete management of complex systems (Kaplan, 1997). Systems in healthcare, of which nursing is a part, similarly belong within this framework (Taylor, 2001; Van Bemmel and Musen, 1997). Along with the system approach, it makes sense to use the potentials of contemporary information and communications technology (ICT) and to study the possibility of adding value in managing complex systems, especially in terms of the effective use of resources and quality assurance.

E-documentation of any process is an information model, which uses ICT for organizing and managing the process according to established goals. Nursing documentation consists of patient and family data. Nurses use these data to plan the nursing process, which in short covers assessing patient's nursing problems, making nursing diagnoses, implementing nursing interventions and evaluating the work (Gordon, 1994; Taylor et al., 2001). The documentation of the nursing process is the basis for the successful work of a nurse, and also represents an element of quality assurance in nursing (Ball et al., 2000; Rajkovič et al., 2000; Saba and McCormick, 2000; Potter and Griffin Perry, 2003). Existing nursing documentation mainly consists of words, and only rarely includes graphs and pictures. It provides a data set that serves as a base for a software solution (Šušteršič et al., 2002; Klein, 2003). As long as such documentation is kept manually, ICT possibilities are not exploited. It makes sense to use object-oriented approach to the reengineering of documentation into electronic form (McFadden and Hoffer, 1994; Barry, 1996; Kroell & Birthe Garde, 2005), which enables more suitable structuring and processing of data in electronic form. It is thus a matter of structuring the documentation in terms of an object orientation, whereby classical data are combined with models and procedures for their implementation, e.g., graphic presentation of numerical data (Kaplan 1997). At the same time, when reengineering the documentation, possibilities and needs appear for the reengineering of basic processes (Jacobson et al., 1994; Ferioli, Migliarese, 1997; Chang, 1999; Meystel and Albus, 2002) in the organisational sense, in this case in the field of nursing.

We wish to propose a model that will serve for the reengineering of classical documentation into e-documentation. With a suitable object-oriented organisation and use of contemporary ICT it is thereby possible to achieve a higher level of quality especially in regard to integral treatment of the patient. The active computer model itself supports the work of the nurse and, at the same time, reduces the possibility of mistakes at work.

This article is based on findings and models developed within the framework of a *Project for preparation of a model tool for establishing quality with the aid of documentation in nursing* at the Ministry of Health of the Republic of Slovenia. Below are presented the elaboration and implementation of the proposed model, as well as testing of the prototype software for community nursing.

2 Analysis of existing documentation in nursing

Using the survey research we first analysed the current state of documentation in nursing in selected health organisations in Slovenia. The sample included three old people's homes, Ljubljana Health Centre with five units, Clinical Centre Ljubljana and Maribor General Hospital. We distributed 386 questionnaires, of which 286 (response rate of 74.1%) were returned.

From the results of the survey on the use and suitability of nursing documentation we can conclude both the actual state of the documentation itself and the process of documenting, and also the perception (opinions, considerations) of existing problems and possible solutions on the part of those surveyed.

The majority of documents (86%) are prescribed on the level of the institution. The only exceptions are community nursing and old people's homes. Documentation for community nursing is prescribed and unified throughout the country, while old people's homes have a uniform computer supported information system. Rather less than 13% of documents are computer supported. Among the types of documents, the following were most frequently listed: nursing care plan, referral/discharge document, continuation notes and variance report, admission document and report on an undesired event.

It can be concluded that those five most frequently used documents should be unified firstly, taking into account the specificities of individual services. Given that with contemporary ICT we can generally provide effective support to documentation and increase the use of computers.

From the perspective of content, a process method of work is only used in 32% of nursing documentation. Over 52% use only a fragmented process approach. It appears that existing documentation is to a large extent at fault for this, since the majority uses only those elements that the documentation enables. It is therefore sensible to reengineer the documentation in a way that will enable documenting all phases of the nursing process.

Minimal data set on patient are recorded by three quarters of survey participants. One of the reasons that the percentage is not higher is unsuitability of existing documentation.

Discussions with the patient, observation of the patient and measurements are sources of data for completing documentation in more than half of cases. Slightly less than half have also stated nursing documentation as a regularly used source of data. It is sensible to consider links between other health documentations and nursing documentation.

In the opinion of the surveyed nurses, they see the purpose of documentation or documenting mainly in the continuity of nursing, security for members of the nursing team and patient and an account of the work of individual members of the nursing team. The content thus supports the work, with emphasis on the legal security of members of the nursing team and the patient.

Among reasons for the non-use of nursing documentation, according to a quarter of nurses, are understaffing and insufficient knowledge of the nursing process, and among unspecified reasons, the fact that existing documentation is unsuitable was most often noted.

In terms of the influence of nursing documentation, the following are highlighted: the quality of nursing, uniform doctrine of work and reducing the possibilities of mistakes. With improved documentation we expect most changes in the quality of collaboration inside the health team and in the distribution of work and responsibilities among nurses and other health team members.

The results have shown that reengineering documentation using ICT can and should positively influence on the quality of nursing care. Because of unified documentation in the community nursing we have decided to begin the reengineering of documentation in community nursing.

3 Process method of work in nursing

The basis for developing e-documentation is the nursing process. Figure 1 shows a schematic presentation of the process method of work in the IDEF1 standard. The division in the figure differ from the literature (Taylor et al., 2001), and Table 1 shows the link between the two models. A major difference is in the stage of evaluation due to standardisation restriction. IDEF1 standard does not allow any process to appear in the scheme more than once. There is also a difference due to the cybernetic feedback loops, which are of crucial importance from a systemic point of view for system management, in this case of nursing.

The user interface of the prototype supports this process method in the nurse's job sequence. Only a few elements must be added, which are specific for community nursing (Rajkovič and Šušteršič, 2000). These are elements such as entering referrals for community care visits to patients or families and for planning dates of home visits. Home visits can only be planned on the basis of referrals received from the general practitioner and on the basis of instructions for implementing community nursing. Later on the same steps apply for each home visit as in the already mentioned process scheme.

4 Database model

The base for a software solution is a database that enables data archiving and accessing data. Critical analysis of nursing documentation was a starting point in the database design process (Handler and Hieb, 2003).

In paper form, the documentation is often mainly unstructured. Thus words in sentences can be entered. A problem occurs when seeking data in a longer text. The legibility of the writing often presents additional difficulty. Similarly, the statistical processing of data for research,

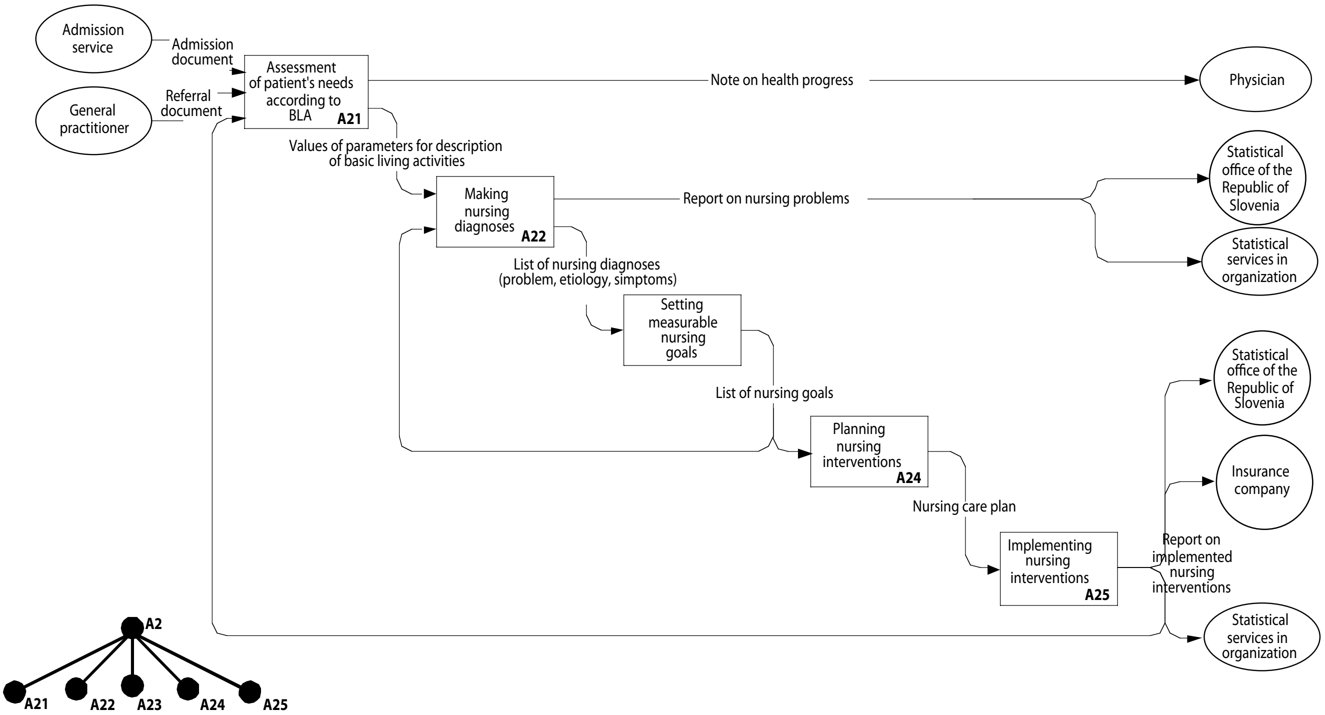


Figure 1: Schematic presentation of the process method of work in nursing according to the IDEF1 standard

Table 1: Comparison of phases of the nursing process and the schematic presentation in Figure 1

Stages of nursing process	Sub-processes of the process method of work in Figure 1	
1. Assessment	A21	Assessment of patient's needs according to basic living activities
2. Diagnosis	A22	Making nursing diagnoses
3. Planning	A23	Setting measurable nursing goals
	A24	Planning nursing interventions
4. Implementation	A25	Implementing nursing interventions
5. Evaluation	A21	Assessment of patient's needs according to basic living activities
	A22	Final evaluation; making changes in the list of nursing diagnoses

education and other purposes can become unreasonably difficult (International Council of Nurses, 1999).

The higher level of data structuring in electronic form enables a higher usability of acquired data. The nurse is reminded with the entry fields, of all data that are desired in the documentation. The nurse is also forced to gather and record important data in the compulsory fields. This results in electronic form of data that enable electronic processing.

Additional fields for entering comments and data that were not envisaged in the original structure are also important. These fields serve in the prototype solution also as information for further development – which data must be additionally structured for electronic processing.

For ease of overview, we have grouped similar data according to their semantic relations. Table 2 shows a tree structure of data for describing a patient, and Table 3 the structure of data for describing a family.

Table 2: Tree structure of data groups on a patient

Patient
- Basic data
  - Basic data (name, surname, sex etc.)
  - Addresses
  - Admission data
  - Selected physicians
  - Education and employment
  - Health insurance
  - Nursing attributes
    - Chronic non-contagious diseases
    - Genetic diseases
    - Disability
    - Other
  - Notes
- Other data
- Family
  - Patient's data in regard to the family
  - Family data
    - List of family members
- Visits
  - Home visits;
  possible viewing of data from an individual visit
- Childbirth and development
  - Newborn infant
    - Data on birth
    - Measurements at time of birth
    - Data at time of discharge from hospital
    - Other data
  - Infant and pre-school child
  - School pupil and teenage years
  - Age independent child's data
- Pregnancy/postnatal
  - List of pregnancies
  - Data on selected pregnancy
  - Data on selected delivery
    - List of born children
- Document of nursing care

Table 3: Tree structure of data groups on a family

Family
- General data
  - Demographic data
  - List of family members
- Home visits
- Document of nursing care

A relational database suitable for storing data in electronic form was developed. It enables simple entry of data, reduces duplication of data and provides fast extraction (Madsen, 2005). The entity-relationship diagram of the proposed database is shown in Figure 2.

It is worth highlighting some particularities in the database diagram. The nursing diagnosis is directly bound to the subject of nursing. In the nursing diagnosis we record to which basic living activity it is bound, and we are aware that after the evaluation of the nursing care plan, it may remains in the care plane throughout one or more of the following visits.

The subject of nursing can be a patient or a family. In the case of a visit to a family or a visit to a patient living with other family members, we also see a list of all family members who live together. From this list, we can access data on an individual member or on the whole family.

We see elements of the diagnostic therapeutic programme as a list of previously determined nursing interventions, which must be carried out independently in relation to the established nursing problems, nursing diagnoses or nursing goals.

In electronic documentation a user uses a password protected log-in. While it was necessary to sign some of documents in paper form, in electronic version data on the user are automatically recorded. For example, when the user records that an individual nursing intervention has been carried out, it is automatically recorded who entered the data for the individual activity and when.

5 Software solution

The steps that comprise the desired course of work of the community nurse (CN) required for each home visit are in accordance with the process method of work in nursing. When the CN selects a patient or a family and one of the planned home visits, a screen picture is shown for the individual home visit. In the upper part are shown data on the selected patient or family, and below the individual steps of the nursing process supported through four tabs: nursing anamnesis, assessment of patient's/family's need, planning and implementation. We will describe later how the evaluation phase is supported.

We have grouped criteria for an overall assessment of patient's need in a tree structure based on the fourteen basic living activities (Henderson, 1997; Bohanec et al., 2000; Šušteršič et al., 2003). It is a professionally accepted and well-known division as it has been confirmed in our survey. A list of parameters opens for each basic living activity of which we wish to remind the CN for gathering relevant data. These parameters are taken from the profession, and in nomenclature we followed the Slovenian translation of the International Classification of Nursing Practice (Cibic et al., 2000).

With each parameter there is a field with free text for the entry of values, e.g., with the parameter of excessive body weight we can insert the body mass index. In addition, with each parameter we can also determine the degree of a problem according to a five-point scale (no problem, minor problem, medium problem, major problem, very severe problem).

The CN chooses the values in relation to the assessed state with individual nursing subjects. From the values describing degrees of problems for parameters under the same basic living activity, the degree of problem for an individual basic living activity is calculated. These calculated values are then shown in the phase of planning. We will later show how we have supported evaluation with these grades.

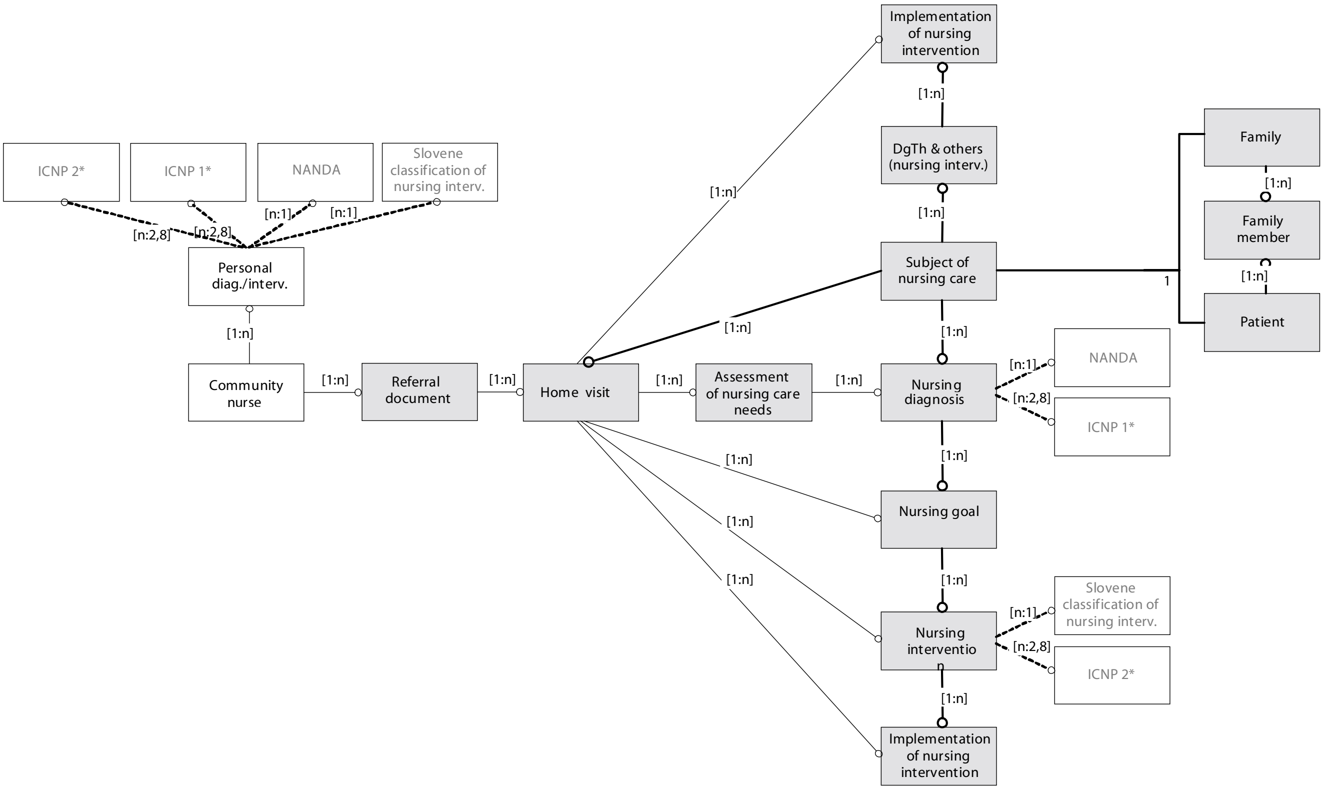
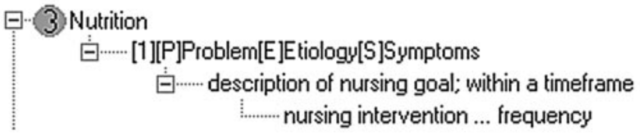


Figure 2: Entity-relationship diagram of the proposed database

Thus, e.g., the value of the parameter appetite has an impact on diet, this on the basic living activity diet and drinking which in return affects the physical basic living activity and, consequently, the overall assessment of the patient. Values of the higher level parameters on the tree structure of parameters are calculated. The CN records values only for final parameters in the tree structure that is parameters on the tree leaves. After a simple calculation, we then obtain the grade of nursing problem for, e.g., an individual basic living activity, or total overall assessment.

Under the tab planning we compose a nursing care plan in a tree structure. At the first level we see a list of basic living activities and with each a calculated degree of a problem. On the second level we can add to each basic living activity an arbitrary number of nursing diagnoses. To each nursing diagnosis we must further add at least one nursing goal, and to each nursing goal at least one nursing intervention (Figure 3).



3 Nutrition
[1][P]Problem[E]Etiology[S]Symptoms
description of nursing goal; within a timeframe
nursing intervention ... frequency

Degree of a problem, basic living activity
Nursing diagnosis
Nursing goal; with timeframe
Nursing intervention … frequency

Figure 3: Schematic presentation of nursing care plan in tree structure with explanation of individual levels

The nursing diagnosis is made according to the PES system (Problem, Etiology, Symptoms). In denoting a problem, the nurse can get help from the International Classification of Nursing Practice (ICNP beta 2) or the classification of the North American Nursing Diagnosis Association (NANDA) (Gordon, 1994; International Council of Nurses, 1999; Šušteršič et al., 1999; Rajkovič et al., 2003).

Nursing interventions are described by name and frequency of performing them. Nurses denote interventions with the aid of the Slovenian classification of nursing interventions and the ICNP beta 2.

The nurse can store the most often used nursing diagnoses and interventions in her personal directory. In the prototype this is a planned solution, which can be simply supplemented with the catalogues that International Council of Nurses propagates as lists of the most often used nursing diagnoses and interventions for individual fields of nursing.

Under the implementation tab are shown all planned nursing interventions, those carried out need only to be marked.

For the needs of evaluation, we can record comments on individual nursing interventions, e.g., ongoing evaluation or values of measurements, materials and time used.

After carrying out nursing interventions it is necessary to reassess the condition at the end of the visits. Assessments of the condition between two visits can also be compared. The evaluation phase is supported with the following visualisation elements: comparison of overall assessments that shows progress for every parameter, a progress graph for a selected parameter throughout all previous visits. CN uses these measurements of changes in patient's needs to evaluate nursing goals and other elements in the nursing care plan.

When comparing assessments, we can compare the grade of nursing problem for the recorded and calculated criteria for two entries of overall assessments. We can thus compare two home visits, analyse the condition in the time between visits, or compare the overall assessment before and after visits are made, and analyse the impact of the intervention carried out on the change of condition.

Where we have a number of assessments, it is also possible to show a time series of levels of nursing problem for an individual parameter – what sort of level of nursing problem was shown through all assessments of the condition e.g., appetite.

With these elements we wish to support the evaluation phase. The result of the evaluation phase is reflected in the changed nursing care plan. This means in practice to seek inappropriate elements in the nursing care plan, supplement them, exchange or remove them, and plan a new part of the nursing care plan for the new focus of problems.

The tabs are similar for home visits to the family, differing only in the parameters by which we describe the overall assessment. Instead of basic living activities, we divided the parameters of families into the following groups: socio-economic state, health anamnesis, relations within the family and with the wider environment and functions of the family.

Computer support is provided by a completed prototype software solution. The solution used is a type of client-server and enables the use of laptop computers, which the CN can use directly in the field.

The software is accompanied by a user's manual. It contains instructions for installing the programme and organisational and informational instructions for the direct use of the prototype software. It is basically intended as an aid to the work of the CN in using the software.

6 Testing the model in practice

We wanted to check the following categories in testing the proposed solutions:

- Success in implementing the nursing process in accordance with individual phases,
- Strengths and weaknesses of the use of hierarchical models of basic living activities in the process,
- Suitability of the structure of data in nursing documents with an emphasis on the nursing care plan,
- Accordance of the data model and links among data with the current method of work or existing documentation,
- Interface with other processes.

In the alpha phase of testing, after completing the writing of the manual, we checked the operation of the programme in compliance with the manual using simulation of real data. This was first carried out by the programmer and then by two working groups.

Beta testing of the software took place in the community nursing units of Ajdovščina Health Centre and Ljubljana-Bežigrad Health Centre. At both locations we placed software of a client-server type. Each participant in the testing, a CN, thus had available her own computer supported worksite. Data inserted at locations was gathered on the server.

We began with an introductory seminar at each location, which covered:

- Presentation of the programme in accordance with the process method of work,
- Presentation of the manual and annex and
- Test entry of data.

The CNs then had a month to become familiar with the programme and to try some test entries. During this time we solved some open questions in relation to terminology, the new model and software solution.

In order for the CN to become accustomed to work with the programme, during the month they entered test data daily (1-2 entries daily). This was the introductory period, which was intended to make a significant contribution to CNs being subsequently able to carry out the extensive plan of testing.

As a last step, we presented a detailed plan of testing. More than 80 entries in the period of one month provide the framework for testing various subjects with various needs and difficulty of work.

At a new meeting, we then discussed possible difficulties and proposals of improvements and examined the entered records.

A SWOT analysis was carried out, which we performed with the help of the participants of testing at a final meeting.

Strenghts (advantages):

- Providing users with integral nursing of high quality;
- Timely recognition of some dangers that threaten the patient;
- Systemically arranged data of a relatively large quantity, which provides an easily viewed information picture;
- Encouragement to the CNs own professional development.

Weaknesses:

- Insufficient ICT equipment;
- In dealing with patients with existing solutions we do not cover some administrative needs (e.g. reading health insurance cards);
- Too many patients per day prevent concurrent insertion of data into the computer (work norms are frequently exceeded);
- Lack of professional knowledge;
- The question is raised as to whether we know and can suitably use the available data.

Opportunities:

- To be more attuned to the user by means of the available data and be able to offer a higher quality of nursing;
- Users can be better informed and educated;
- Timely recognition of conditions;
- Production of guidelines for professional treatment and higher quality services for users;
- Including family and others in the nursing process;
- Motivation of staff.

Threats (dangers):

- Insufficient ICT equipment of community nursing services could hold back the use of the system;
- Lack of permanent professional training and willingness of nurses to change could negatively affect on the use of such system;
- Changes in existing methods of work often trigger resistance in staff;
- Commitment to the computer rather than the patient.

Although the model and its prototype are already suitable for use in practice, we will continue to take certain comments into account and make the necessary changes. Extended testing will follow, which means monitoring use of the model in practice in a larger number of community nursing institutions throughout longer period of time.

7 Conclusion

The presented model of e-documentation covers the treatment of patients and families both from the aspects of processes and data. On this basis, a prototype organisational and informational solution of nursing documentation for the community nursing segment was developed and has been tested in practice and critically evaluated.

The added value that contemporary ICT can contribute to nursing was presented, deriving primarily from a structured information picture, which monitors the patient and the nurse in the nursing process. It is worth highlighting in particular the use of hierarchical models in the treatment of basic living activities. The model of calculating the grade of nursing problem, which the computer carries out concurrently in relation to the condition of the patient, thus enables an integral overview of the patient and systemically links apparently separate problems. It is thus a direct contribution to reducing the possibility of overlooking something important. E-documentation relies on the nursing record of the patient as a part of the overall health record of the patient (Hammer et al., 2003). This way we avoid duplication of data and the associated excessive work and obtain an overall information picture, which significantly contributes to greater security for the patient and members of the nursing team.

We will continue the work not merely by extending testing and analysis of this model, but also by developing a similar model for documenting nursing in hospitals and dispensaries.

Acknowledgements

The authors thank the Ministry of Health of the Republic of Slovenia for financing the research work. We are also sincerely grateful to all institutions and individuals who took part in the project.

Literature

Ball, M.J., Hannah, K.J., Newbold, S.K. & Douglas, J.V. (eds.) (2000). *Nursing Informatics. Where Caring and Technology Meet, (3th edition)*, Springer, Berlin.

Barry, D.K. (1996). *The Object Database Handbook*, John Wiley & Sons Inc, New York.

Bohanec, M., Zupan, B. & Rajkovič, V. (2000). Applications of Qualitative Multi-Attribute Decision Models in Health Care. *Int J Med Inf*, **58-59**: 191-205.

Cibic, D., Dogša, I., Filej, B., Šlajmer-Japelj, M. & Šušteršič, O. (2000). *International Classification of Nursing Practice (in Slovenian language: Mednarodna klasifikacija prakse zdravstvene nege)*, Kolaborativni center SZO za primarno zdravstveno nego, Maribor.

Chang, R.Y. (1999.) *Process Reengineering in Action: A Practical Guide to Achieving Breakthrough Results*, Jossey-Bass/Pfeiffer, San Francisco.

Ferioli, C. & Migliarese, P. (1997). The Organizational Relational Model: Proposal and Result, *Decision Support in Organizational Transformation, Edited by* Humphreys, P., Ayestaran, S., McCosh, A. & Mayon-White, B., Chapman & Hill, London.

Gordon, M. (1994). *Nursing Diagnosis. Process and Aplication. (3th ed.)*, Mosby, St Louis.

Hammer, S.V., Moen, A., Børmark, S.R. & Husby, E.H. (2003). A Hospital Wide Approach to Integration of Nursing Documentation in the Electronic Patient Record, *Proceedings of the 8th International Congress in Nursing Informatics,* Edited by de Fatima Marin, H.M., Marques, E.P., Hovenga, E. & Goossen, W., Rio de Janeiro 20.-25. June 2003, pp. 212-216, International Council of Nurses, Rio de Janeiro.

Handler, T.J. & Hieb, B.R. (2003). *The Gartner 2004 Criteria for the Enterprise Computer-Based Patient Record*, Gartner Group. Available from http://www4.gartner.com/resource/118300/118364/118364.pdf

Henderson, V. (1997). *Basic Principles of Nursing Care*, International Council of Nurses, Geneve.

International Council of Nurses. (1999). *International Classification of Nursing Practice. Beta*, International Council of Nurses, Geneve.

Jacobson, I., Ericson, M. & Jacobson, A. (1994). *The Object Advantage: Business Process Reengineering with Object Technology*, Addison-Wesley Pub Co, Reading, Massachusetts.

Kaplan, R.M. (1997). Intelligent Multimedia Systems, John Wiley & Sons Inc, New York.

Klein, J. (2003). *Predictions for 2004 in Healthcare and Life Sciences*, Gartner Group. Available from http://www4.gartner.com/resources/119000/119010/119010.pdf

Kroell, V. & Birthe Garde, A. (2005). Strategy for Documentation in Nursing at a National and at a Local Level in Denmark, *ACENDIO 2005: Documenting Nursing Care*, Edited by Oud, N., Sermeus, W. & Ehnfors, M., Bern, 2005, pp. 139-142, Verlag Hans Huber, Bern.

Madsen, I. (2005). How to Avoid Redundant Data in an Interdisciplinary Electronic Patient Record, *ACENDIO 2005: Documenting Nursing Care,* Edited by Oud, N., Sermeus, W. & Ehnfors, M., Bern, 2005, pp. 139-142, Verlag Hans Huber, Bern.

McFadden, F.R. & Hoffer, J.A. (1994). *Modern Database Management, Fourth Edition*, The Benjamin/Cummings Pub Co Inc, Redwood City.

Meystel, A.M. & Albus, J.S. (2002). *Intelligent Systems: Architecture, Design and Control*, John Wiley & Sons Inc, New York.

Potter, P.A. & Griffin Perry, A. (2003). *Basic Nursing, Fifth Edition*, Mosby Inc, St Louis.

Rajkovič, V. & Šušteršič, O., (eds.) (2000). *Information System for Community Nursing (*in Slovenian language: *Informacijski sistem patronažne zdravstvene nege)*, Založba Moderna organizacija, Kranj.

Rajkovič, V., Šušteršič, O., Leskovar, R., Bitenc, I. & Zelič, I. (2000). Increasing Quality of Nurses' Work by an Information System: Community System Case, *Nursing Informatics 2000.* Edited by Saba, V., Carr, R., Sermeus, W. & Rocha, P., Auckland, 2000, pp. 529-536, Adis International Ltd, Auckland.

Rajkovič, V., Šušteršič, O., Rajkovič, U., Porenta, A. & Zupančič, M.J. (2003). How E-Representation brings International Classification of Nursing Practice Closer to User, *ACENDIO 2003. 4th European Conference of ACENDIO. Making nursing visible.* Edited by Oud, N., pp. 166-170, Verlag Hans Huber, Bern.

Saba, V.K. & McCormick, K.A. (2000). *Essentials of Computers for Nurses: Informatics for the New Millennium*, McGraw-Hill, New York.

Šušteršič, O., Rajkovič, V. & Kljajić, M. (1999). An Evaluation of Community Nursing Process in the Frame of the International Classification for Nursing Practice, *ICNP and Telematic Applications for Nurses in Europe. The Telenurse Experience.* Edited by Mortensen, R.A., pp. 243-249, IOS Press OHM Ohmska, Amsterdam.

Šušteršič, O., Rajkovič, V., Kljajić, M., Rajkovič, U. (2003). Improving Nursing Care Documentation by Computerised Hierarchical Structures, *Proceedings of the 8th International Congress in Nursing Informatics.* Edited by de Fatima Marin, H.M., Marques, E.P., Hovenga, E. & Goossen, W., Rio de Janeiro 20.-25. June 2003, pp. 212-216, International Council of Nurses, Rio de Janeiro.

Šušteršič, O., Rajkovič, V., Leskovar, R., Bitenc, I., Bernik, M. & Rajkovič, U. (2002). An Information system for Community Nursing, *Public Health Nurs*, **19**(2002): 184-90.

Taylor, C., Lillis, C. & LeMone, P. (2001) *Fundamentals of Nursing, The Art and Science of Nursing Care (4th)*, Lippincott, Philadelphia.

Van Bemmel, J.H. & Musen, M.A. (eds.) (1997). *Handbook of Medical Informatics*, Springer-Verlag, Berlin.

Uroš Rajkovič has a BSc degree in information system management from University of Maribor, Faculty of Organizational Sciences, where he is employed as a teaching assistant. His main research field are information systems in nursing care.

Olga Šušteršič has a PhD degree in information system management from University of Maribor, Faculty of Organizational Sciences. She is associate professor of nursing care, community and dispensary nursing care and computer science in nursing at the University of Ljubljana, College of Health Studies. She participates in numerous national and international research projects regarding nursing care.

Jože Zupančič has a PhD from University of Ljubljana, Faculty of Electrical and Electronic Engineering. He is professor of information systems at University of Maribor, Faculty of Organizational Science. His primary research interests are information systems development methods and tools, information systems management, and user acceptance of information systems.

Donatorji novogradnje Fakulteta za organizacijske vede

ADRIA AIRWAYS - Slovenski letalski prevoznik d.d.
Kuzmičeva 7, 1000 LJUBLJANA

ARBORETUM
Volčji potok 3, 1235 RADOMLJE

ALPETOUR REMONT d.d.
Ljubljanska 22, 4000 KRANJ

ATOTECH - Kemična tovarna Podnart d.d.
Podnart 24, 4244 PODNART

AVTOTEHNA d.d.
Slovenska cesta 54, 1000 LJUBLJANA

BIGRAD
Kolodvorska 37d, 2310 SLOVENSKA BISTRICA

BRDO - PROTOKOLARNI SERVIS
Predoslje 39, 4000 KRANJ

DEDALUS d.o.o.
Dunajska 156, 1000 LJUBLJANA

DELO - Časopisno in založniško podjetje d.d.
Dunajska 5, 1509 LJUBLJANA

DOMEL d.d. - Elektromotorji in gospodinjski aparati
Otoki 21, 4228 ŽELEZNIKI
DOMPLAN d.d.
Bleiweisova cesta 14, 4000 KRANJ

ELEKTRO GORENJSKA
Javno podjetje za distribucijo električne energije, d.d.
Bleiweisova cesta 6, 4000 KRANJ

ELEKTROTEHNIŠKO PODJETJE d.d.
Ulica Mirka Vadnova 11, 4000 KRANJ

EL - VER, Elektroinstalacije Zvonko Verlič s.p.
Streliška 150, 2000 MARIBOR

ETIKETA Tiskarna d.d.
Industrijska ulica 6, 4226 ŽIRI

EXOTERM Kemična tovarna, d.d.
Struževo 66, 4000 KRANJ

FOTO TIVOLI d.o.o.
Cankarjeva 7, 1000 LJUBLJANA

GORENJSKA BANKA d.d.
Bleiweisova 1, 4000 KRANJ

GORENJSKA PREDILNICA d.d.
Kidričeva cesta 75, 4220 ŠKOFJA LOKA

GORENJSKI TISK d.d.
Ul. Mirka Vadnova 6, 4000 KRANJ

GRADBINEC GIP d.o.o.
Nazorjeva 1, 4000 Kranj

GRATEX d.o.o.
Spodnja Rečica 81, 3270 LAŠKO

HIT d.d. Nova Gorica - Hoteli igralnica turizem
Delpinova 7a, 5000 NOVA GORICA

HTG - Hoteli Turizem Gostinstvo d.d.
Partizanska cesta 1, 6210 SEŽANA

IBM Slovenija d.o.o.
Trg Republike 3, 1000 LJUBLJANA

IBI Kranj - Proizvodnja žakarskih tkanin d.d.
Jelenčeva ulica 1, 4000 KRANJ

ISA Anton Mernik s.p. - Izvajanje sanacij v gradbeništvu
Kolodvorska ulica 35c, 2310 SLOVENSKA BISTRICA

ISKRAEMECO, d.d.
Savska Loka 4, 4000 KRANJ

ISKRA - Iskra avtoelektrika d.d.
Polje 15, 5290 ŠEMPETER PRI GORICI

ISKRA - Industrija sestavnih delov d.d.
Savska loka 4, 4000 KRANJ

ISKRA INSTRUMENTI d.d.
Otoče 5a, 4244 PODNART

ISKRATEL - Telekomunikacijski sistemi d.o.o., Kranj
Ljubljanska cesta 24/a, 4000 KRANJ

ISKRA TRANSMISSION d.d.
Stegne 11, 1000 LJUBLJANA

Izredni študenti FOV

JELOVICA d.d.
Kidričeva 58, 4220 ŠKOFJA LOKA

JEROVŠEK COMPUTERS, d.o.o.
Breznikova 17, 1230 DOMŽALE

KOGRAD GRADNJE d.o.o.
Miklavška cesta 82, 2311 SPODNJE HOČE

KOMUNALNO PODJETJE GORNJA RADGONA p.o.
Trate 7, 9250 GORNJA RADGONA

KOPIRNICA DEU s.p.
Kidričeva 55a, 4000 KRANJ

KOVINAR d.o.o. Vitanje
Kovaška cesta 12, 3205 VELENJE

KRKA, d.d., Novo mesto
Šmarješka cesta 6, 8501 NOVO MESTO

KRKA ZDRAVILIŠČA - Zdraviliške, turistične in gostinske storitve d.o.o.
Germova ulica 4, 8501 NOVO MESTO

LESNA Lesnoindustrijsko podjetje d.d.
Pod gradom 2, 2380 SLOVENJ GRADEC

LETNIK SAUBERMACHER d.o.o.
Sp. Porčič 49, 2230 LENART V SLOVENSKIH GORICAH

LINIJA - Rajko Flerin, s.p., Slikopleskar in črkoslikar
Britof 284, 4000 KRANJ

LJUBLJANSKE MLEKARNE d.d.
Tolstojeva 63, 1000 LJUBLJANA

LUKA KOPER d.d.
Vojkovo nabrežje 38, 6000 KOPER

MAGNETOMEDICINA d.o.o.
Tržaška cesta 468, 1351 BREZOVICA PRI LJUBLJANI

MARMOR HOTAVLJE d.d.
Hotavlje 40, 4224 GORENJA VAS

MAT d. o. o.
Orlova 12 a, 1000 LJUBLJANA

MEHANIZMI - Iskra Mehanizmi d.d. Lipnica
Lipnica 8, 4245 KROPA

MERCATOR - TRGOAVTO d.d. - Trgovina, servis
Pristaniška 43/a, 6000 KOPER

MERCATOR - PC GRADIŠČE d.d.
Golijev trg 11, 8210 TREBNJE

MERCATOR-OPTIMA - Inženiring d.o.o.
Breg 14, 1000 LJUBLJANA

MERKUR - Trgovina in storitve d.d.
Cesta na Okroglo 7, 4202 NAKLO

MESNA INDUSTRIJA PRIMORSKE d.d.
Panovška 1, 5000 NOVA GORICA

MICROSOFT d.o.o.
Šmartinska cesta 140, 1000 LJUBLJANA

MOBITEL d.d.
Vilharjeva 23, 1537 LJUBLJANA

OBČINA RADOVLJICA
Gorenjska cesta 19, 4240 RADOVLJICA

Opravljanje del z gradbeno mehanizacijo MARJAN RAZPOTNIK s.p.
Krače 8, 1411 IZLAKE

OPTIMA - Podjetje za inženiring in trgovino d.o.o.
Ulica 15. maja 21, 6000 KOPER

PALOMA SLADKOGORSKA - Tovarna papirja d.d.
Sladki vrh 1, 2214 SLADKI VRH

PIVOVARNA UNION d.d.
Pivovarniška ulica 2, 1001 LJUBLJANA

POSLOVNI SISTEM MERCATOR d.d.
Dunajska cesta 107, 1000 LJUBLJANA

POSLOVNI SISTEM - ŽITO LJUBLJANA d.d.
Šmartinska cesta 154, 1000 LJUBLJANA

POSLOVNO PRIREDITVENI CENTER - GORENJSKI SEJEM Kranj d.d.
Stara cesta 25, 4000 KRANJ

POŠTA SLOVENIJE d.o.o.
Slomškov trg 10, 2000 MARIBOR

PRIMORJE d.d.
Vipavska cesta 3, 5270 AJDOVŠČINA

REGIONALNI CENTER ZA RAZVOJ d.o.o.
Cesta zmage 35, 1410 ZAGORJE OB SAVI

SATURNUS - AVTOOPREMA d.d.
Letališka c. 17, 1001 LJUBLJANA

SAVA - Gumarska in kemična industrija d.d.
Škofjeloška 6, 4502 KRANJ

SIEMENS d.o.o.
Dunajska cesta 22, 1000 LJUBLJANA

SLOBODNIK JOŽE
Generalni častni konzul RS v Kanadi

SLOVENIJALES PRODAJNI CENTRI
Dunajska cesta 22, 1000 LJUBLJANA

SLOVENSKE ŽELEZNICE d.d
Kolodvorska ulica 11, 1000 LJUBLJANA

SVEA LESNA INDUSTRIJA d.d.
Cesta 20. julij 23, 1410 ZAGORJE OB SAVI

SUROVINA d.d. MARIBOR
Pobreška cesta 20, 2000 MARIBOR

TELEKOM SLOVENIJE d.d
Cigaletova 15, 1000 LJUBLJANA

TERME MARIBOR Zdravstvo, turizem, rekreacija d.d.
Ul. heroja Šlandra 10, 2000 MARIBOR

TERMO d.d. - Industrija termičnih izolacij
Trata 32, 4220 ŠKOFJA LOKA

TERMOELEKTRARNA TOPLARNA Ljubljana d.o.o.
Toplarniška 19, 1000 LJUBLJANA

TOVARNA KLOBUKOV ŠEŠIR d.d.
Kidričeva 57, 4220 ŠKOFJA LOKA

TRIMO Inženiring in proizvodnja montažnih objektov d.d.
Prijateljeva 12, 8210 TREBNJE

UNITAS - Tovarna armatur d.d.
Celovška cesta 224, 1107 LJUBLJANA

USTANOVA SLOVENSKA ZNANSTVENA FUNDACIJA
Štefanova 15, 1000 LJUBLJANA

ZAVAROVALNICA TRIGLAV, d.d.
Miklošičeva cesta 19, 1000 LJUBLJANA

ZAVAROVALNICA TRIGLAV, d.d. Območna enota Kranj
Bleiweisova cesta 1, 4000 KRANJ

ZVEZA RAČUNOVODIJ, FINANČNIKOV IN REVIZORJEV SLOVENIJE
Dunajska cesta 106, 1000 LJUBLJANA

ŽIVILA KRANJ - Trgovina in gostinstvo d.d.
Cesta na Okroglo 3, 4202 NAKLO

ŽITO GORENJKA d.d.
Rožna dolina 8, 4248 LESCE

Imensko kazalo letnika (39) 2006

A
AHČAN Aleš, št. 4, str. 233. The Effects of Business Uncertainity on Investment Policies of Finacial Intermediaries.
ALADIN – Alpe Adria Initiative Universities' Network
ALADIN – Alpe Adria Initiative Universities' Network : Cooperation in e-Integration Research & Teaching in the Region.
ARH Tanja, KOVAČIČ Matjaž, JERMAN-BLAŽIČ Borka, št. 6, str. 393. Struktura ponudbe e-izobraževanja v Sloveniji.
ARSHAM Hossein, KLJAJIĆ Miroljub, št. 10, str. 626. Simulation System Design.

B
BAČA Miroslav, MALEKOVIĆ Mirko, RABUZIN Kornelije, št. 7, str. 425. A Multimodal Biometric System Based upon the Active Database paradigm.
BAČANOVIĆ Danica, JOŠANOV Borislav, št. 3, str. 205. Contributions of ALADIN to eRegion Development: The Novi Sad Business School Perspective, Department for Informatics.
BALANTIČ Zvone, št. 8, str. 546. Spiralni razvoj programske opreme kot stalen proces v e-zdravstvu.
BASL Josef, PODLOGAR Mateja, št. 3, str. 184. SAP ERP Case Study at University of Maribor, Slovenia and at University of Economics, Prague, Czech Republic.
BAVEC Cene, št. 3, str. 209. Policy Vision of eRegions – the Case of EU and non-EU Countries.
BENČIČ Margareta, št. 2, str. 141. Prepoznavanje pomena strategije turizma v lokalni skupnosti.
BERNIK Igor, JEREB Eva, št. 4, str. 273. Elektronsko preverjanje znanja: pripravljenost študentov.
BERNIK Igor, JEREB Eva, št. 8, str. 526. Mnenja študentov o e-preverjanju znanja pred in po e-testiranju.
BEŠKOVNIK Bojan, št. 7, str. 446. Spremembe pri organizaciji dobave blaga v nabavni logistiki.
BEŠTER Janez, GRIČAR, Jože, št. 1, str. 75. Pospešitev vključevanja Slovenije v e-regijo bližnjih držav
BOBEK Samo, DEŽELAK Zdenko, STERNAD Simona, št. 3, str. 169. Comparative Analysis of E-Business Implementation Critical Success Factors.
BOBEK Samo, PERKO Igor, št. 5, str. 306. Usmeritve raziskav in področja razvoja sistemov za podporo managementa.
BOROTA Bogdana, BRODNIK Andrej, št. 8, str. 532. Učenje glasbe podprto z IKT tehnologijo.
BREZAVŠČEK Alenka, ZUPAN Lucija, št. 1, str. 58. Novosti, ki jih prinašajo spremembe standarda BS 7799.
BRODNIK Andrej, BOROTA Bogdana, št. 8, str. 532. Učenje glasbe podprto z IKT tehnologijo.
BUKOVEC Boris, št. 2, str. 117. Management človeških virov in obvladovanje organizacijskih sprememb.
BUNETO Andreja, MUMEL Damijan, VIRT Tina, št. 6, str. 361. Uporaba instrumentov internega komuniciranja ter cilji internega komuniciranja v velikih in malih podjetjih v Sloveniji.
BURGAR Bojan, št. 9, str. 606. Osebnostna izraznost ravnatelja v funkciji managementa človeških virov na področju vzgoje in izobraževanja.

Č
ČATER Tomaž, št. 6 str. 386. Pomen osnov konkurenčne prednosti podjetja v skladu s šolo na temelju znanja
ČEH Živa, št. 7, str. 441. Jezik stroke za povečanje delovne učinkovitosti človeških virov.
ČERNETIČ Metod, DEČMAN DOBRNJIČ Olga, št. 8, str. 475. Planiranje izobraževanja in menedžment sprememb.

D
DEČMAN DOBRNJIČ Olga, ČERNETIČ Metod, št. 8, str. 475. Planiranje izobraževanja in menedžment sprememb.
DEVETAK Gabrijel, št. 8, str. 482. Prizadevanja za izboljšanje računalniške pismenosti.
DEŽELAK Zdenko, STERNAD Simona, BOBEK Samo, št. 3, str. 169. Comparative Analysis of E-Business Implementation Critical Success Factors.
DINEVSKI Dejan, JAKONČIČ FAGANEL Janja, LOKAR Matija, ŽNIDARŠIČ Boštjan, št. 8, str. 498. Model ocenjevanja kakovosti elektronskih učnih gradiv.
DOUCEK Peter, št. 3, str. 206. Contributions of ALADIN to eRegion Development: The University of Economics – Prague Perspectives.
DOUCEK Peter, NEDOMOVÁ Lea, KLAS Jan, št. 1, str. 16.Integrated Management System in Information Society.

F
FLERIN Zdene, ZUPANČIČ Jože, št. 1, str. 67. Uporabniška rešitev za zagotavljanje sledljivosti in učinkovitosti rezalnikov kartona.
FINK BABIČ Sonja, št. 5, str. 314. Vloga inovativnosti pri trajnostnem razvoju.
FRIEDL Peter, št. 5, str. 331. Težave in priložnosti podatkovnih povezav med gospodarstvom in javno upravo.

G
GÁBOR András, št. 3, str. 201. The ALADIN Initiative is a Framework for the Strategic Renewal of Corvinus University of Budapest.
GERLIČ Ivan, št. 8, str. 472. Konceptualno učenje in interaktivna učna gradiva.
GOMIŠČEK Boštjan, PODLIPNIK, Katja, št. 9, str. 565. Celovito obvladovanje tveganj v slovenski kemični industriji.
GORENAK Irena, PAGON Milan, št. 4, str. 247. Vpliv organizacijskega komuniciranja na zadovoljstvo policistov pri delu.
GORENAK Irena, št. 7, str. 416. Correlation between Communications Styles of Detectives and their Cooperation with Representatives of other Organizations.

GRIČAR, Jože, BEŠTER Janez, št. 1, str. 75. Pospešitev vključevanja Slovenije v e-regijo bližnjih držav.

GRIČAR Jože, KLJAJIĆ Miroljub, št. 3, str. 202. Contributions of ALADIN to eRegion Development: The Faculty of Organizational Sciences, University of Maribor Perspective.

GRIČAR Jože, VOGEL Douglas R., št. 3, str. 214. eSilk&Amber Road Regions Meeting: Business & Government Executives & Professors

H

HAREJ, Janko, št. 8, str. 486. Uvajanje novih storitev v vzgojno-izobraževalne zavode

HOROVČÁK Pavel, LAVRIN Anton, OROSZ Marián, št 4, str. 278. Training Needs Analysis Tool.

HUBER Tatjana, SVETINA Marko, št. 4, str. 284. Integracija spletnega portala za e-poslovanje s podjetji z ostalimi informacijskimi sistemi v trgovski organizaciji.

HUTINSKI Željko, SAJKO Mario, RABUZIN Kornelije, št. 2, str. 97. System development for the e-Learning in an Institution of Higher Education.

I

INCHINGOLO Paolo, UKOVICH Walter, št. 3, str. 207. Contributions of ALADIN to eRegion Development: The Perspective of the University of Trieste.

J

JANKOVIČ Patricija, št. 1, str. 54. Strokovni vodja in poslovodni organ sta lahko tudi dve ločeni funkciji.

JAKONČIČ FAGANEL Janja, DINEVSKI Dejan, LOKAR Matija, ŽNIDARŠIČ Boštjan, št. 8, str. 498. Model ocenjevanja kakovosti elektronskih učnih gradiv.

JEREB Eva, BERNIK Igor, št. 4, str. 273. Elektronsko preverjanje znanja: pripravljenost študentov.

JEREB Eva, BERNIK Igor, št. 8, str. 526. Mnenja študentov o e-preverjanju znanja pred in po e-testiranju.

JERMAN Damjana, ZAVRŠNIK Bruno, št. 6, str. 373.Vpliv marketinških sposobnosti na porast inovativnosti v organizacijah.

JOŠANOV Borislav, BAČANOVIĆ Danica, št. 3, str. 205. Contributions of ALADIN to eRegion Development: The Novi Sad Business School Perspective, Department for Informatics.

JOVANOVIĆ; Petar, št. 4, str. 240. The Serbian Companies Organizational Restructuring Proceures.

JUKIĆ Tina, VINTAR Mirko, št. 3, str. 176. E-government: the State in Slovenian Local Self-government.

K

KAMPOS Tjaša, št. 8, str. 518. Poskus kot sredstvo vizualizacije za aktivno in kvalitetno učenje.

KATZY Bernard R., št. 3, str. 203. Contributions of ALADIN to eRegion Development: Research and Innovation in a Regional »Living Laboratory«.

KEŠELJEVIĆ Aleksandar, št. 7, str. 432. (Ne)razumevanje znanja kot spoznavnega procesa v okviru ekonomske teorije organizacije.

KLAS Jan, NEDOMOVÁ Lea, DOUCEK Peter, KLAS Jan, št. 1, str. 15. Integrated Management System in Information Society.

KLJAJIĆ Miroljub, GRIČAR Jože, št. 3, str. 202. Contribution of ALADIN to eRegion Development: The Faculty of Organizational Sciences, University of Maribor Perspective.

KLJAJIĆ Miroljub, ARSHAM Hossein, št. 10, str. 626. Simulation System Design.

KLJAJIĆ Miroljub, RAUCH Robert, št. 10, str. Discrete event passenger flow simulation model for an airport terminal capacity analysis.

KLJAJIĆ Miroljub, KOFJAČ Davorin, št. 10, str. 667. Simulacijski pristop k minimiziranju stroškov skladišča v stohastičnem okolju.

KLJAJIĆ Miroljub, ŽIBERT Marko, št. 10, str. 635. Modelling Parallel Service Systems in 6PSS.

KOCIJANČIČ Saša F., ŠORGO Andrej, št. 8, str. 513. Naravoslovni eksperiment : most med šolskim znanjem in vsakdanjimi izkušnjami.

KOFJAČ Davorin, KLJAJIĆ Miroljub, št. 10, str. 667. Simulacijski pristop k minimiziranju stroškov skladišča v stohastičnem okolju.

KOFJAČ Davorin, ŠKRABA Andrej, ROZMAN Črtomir, št. 10, str. 676. The Periodicity of the Anticipative Discrete Demand - Supply Model.

KOLAR Tomaž, št. 7, str. 409. Advancing Market Orientation of Service Companies.

KOLENC Igor, ŠUMANSKI MIKLAVČIČ Martina, MARKIČ Mirko, št. 9, str. 597. Socialno emocionalni odnosi kot spodbuda za oblikovanje novih znanj.

KONEČNIK Maja, št. 4, str. 265. Izzivi proučevanja kompleksne narave blagovne znamke s pomočjo uravnoteženega pogleda nanjo.

KRANJC Gabrijela, MAHNIČ Viljan, št. 8, str. 539. Programiranje v parih v srednjih šolah.

L

LAHOVNIK Matej, št. 6, str. 368. Problematika razkrivanja spornih poslovnih praks v organizacijah.

LAVRIN Anton, ZELKO Miroslav, št. 3, str. 191. Knowledge Sharing in Regional Digital Ecosystems.

LAVRIN Anton, HOROVČÁK Pavel, OROSZ Marián, št 4, str. 278. Training Needs Analysis Tool.

LIKAR Borut, PUCELJ Mitja, št. 2, str. 132. Vodenje kot ključni dejavnik za ustvarjanje inovacijske kulture.

LOKAR Matija, DINEVSKI Dejan, JAKONČIČ FAGANEL Janja, ŽNIDARŠIČ Boštjan, št. 8, str. 498. Model ocenjevanja kakovosti elektronskih učnih gradiv.

M

MAHNIČ Viljan, KRANJC Gabrijela, št. 8, str. 539. Programiranje v parih v srednjih šolah.

MALEKOVIĆ Mirko, RABUZIN Kornelije, BAČA Miroslav, št. 7, str. 425. A Multimodal Biometric System Based upon the Active Database paradigm.

MARKIČ Mirko, ŠUMANSKI MIKLAVČIČ Martina, KOLENC Igor, št. 9, str. 597. Socialno emocionalni odnosi kot spodbuda za oblikovanje novih znanj.

MAYER Janez, št. 5, str. 301. Preraščanje spontane konfliktnosti v organizaciji.

MAYER, Janez, št. 8, str. 467. Raztrgani žamet bolonjske reforme.

MAYER Janez, ŠAKANOVIČ Zlatko, št. 4, str. 254. Nekateri vidiki vodenja in njihov vpliv na organizacijsko klimo in zadovoljstvo zaposlenih v slovenski policiji.
MULEJ Matjaž, POTOČAN Vojko, št. 1, str. 44. Teorija kompleksnosti spada v več tokov teorije sistemov.
MULEJ Matjaž, ROSI Bojan, št. 1, str. 35. Kako celoviteje prepoznavati, preprečevati in obvladati probleme.
MUMEL Damijan, BUNETO Andreja, VIRT Tina, št. 6, str. 361. Uporaba instrumentov internega komuniciranja ter cilji internega komuniciranja v velikih in malih podjetjih v Sloveniji.

N
NEDOMOVÁ Lea, KLAS Jan, DOUCEK Peter, št. 1, str. 15. Integrated Management System in Information Society.
NEKREP Andreja, SLANA Jožica, št. 8, str. 489. Perspektiva uvajanja e-izobraževanja v programe stalnega strokovnega izpopolnjevanja učiteljev.
NOVAK Polona, št. 2, str. 124. Strategija marketinga v visokošolskih zavodih.

O
OROSZ Marián, LAVRIN Anton, HOROVČÁK Pavel, št 4, str. 278. Training Needs Analysis Tool.

P
PAGON, Milan, GORENAK Irena št. 4, str. 247. Vpliv organizacijskega komuniciranja na zadovoljstvo policistov pri delu.
PERKO Igor, BOBEK Samo, št. 5, str. 306. Usmeritve raziskav in področja razvoja sistemov za podporo managementa.
PETROVIC Otto, št. 3, str. 201. Contributions of ALADIN to eRegion Development: The University of Graz Perspective.
PODJED Dan, št. 9, str. 585. Antropološki vidiki postmoderne organizacije.
PODLIPNIK, Katja, GOMIŠČEK Boštjan, št. 9, str. 565. Celovito obvladovanje tveganj v slovenski kemični industriji.
PODLOGAR Mateja, BASL Josef, št. 3, str. 184. SAP ERP Case Study at University of Maribor, Slovenia and at University of Economics, Prague, Czech Republic.
POTOČAN Vojko, MULEJ Matjaž, št. 1, str. 44. Teorija kompleksnosti spada v več tokov teorije sistemov.
PUCELJ Mitja, LIKAR Borut, št. 2, str. 132. Vodenje kot ključni dejavnik za ustvarjanje inovacijske kulture.

R
RABUZIN Kornelije, HUTINSKI Željko, SAJKO Mario, št. 2, str. 97. System development for the e-Learning in an Institution of Higher Education.
RABUZIN Kornelije, MALEKOVIĆ Mirko, BAČA Miroslav, št. 7, str. 425. A Multimodal Biometric System Based upon the Active Database paradigm.
RAJKOVIČ Uroš, ŠUŠTERŠIČ Olga, ZUPANČIČ Jože, št. 10, str. 687. A model of e-documentation of community nursing.
RAJKOVIČ Vladislav, RODMAN Marjan, št. 8, str. 504. Poučevanje odločitvenih znanj v osnovni šoli.
RASPOR Andrej, VOLK ROŽIČ Nevenka, št. 6, str. 378. Posredovanje dela.
RAUCH Robert, KLJAJIĆ Miroljub, št. 10, str. 656. Discrete event passenger flow simulation model for an airport terminal capacity analysis.
REBOLJ Ivan, št. 9, str. 575. Analiziranje in oblikovanje upravljanja obstoječih cestninskih sistemov.
REK Mateja, št. 9, str. 591. Upravljanje z organizacijskim znanjem v kibernetskem prostoru.
REK Mateja, TOMŠIČ Matevž, št. 1, str. 28. Kulturni vidiki ekonomskega in političnega razvoja : stare in nove članice EU v primerjalni perspektivi.
REPONEN Tapio, št. 3, str. 213. eUniversity Forum – eIntegration Challenges for Rectors & Deans : Cross-border Cooperation in e-Region & Ambassadors Involvement
RODMAN Marjan, RAJKOVIČ Vladislav, št. 8, str. 504. Poučevanje odločitvenih znanj v osnovni šoli.
ROSI Bojan, MULEJ Matjaž, št. 1, str. 35. Kako celoviteje prepoznavati, preprečevati in obvladati probleme.
ROZMAN Črtomir, ŠKRABA Andrej, KOFJAČ Davorin, št.10, str. 676. The Periodicity of the Anticipative Discrete Demand - Supply Model.
RUDEŽ Valter, št. 9, str. 561. The MICO Model of Intellectual capital.

S
SAJKO Mario, RABUZIN Kornelije, HUTINSKI Željko, št. 2, str. 97. System development for the e-Learning in an Institution of Higher Education.
SEGETLIJA Zdenko, št. 2, str. 108. Organizational and Structural Changes in Retail Trade in Croatia.
SLANA Jožica, NEKREP Andreja, št. 8, str. 489. Perspektiva uvajanja e-izobraževanja v programe stalnega strokovnega izpopolnjevanja učiteljev.
STERNAD Simona, DEŽELAK Zdenko, BOBEK Samo, št. 3, str. 169. Comparative Analysis of E-Business Implementation Critical Success Factors.
SVETINA Marko, HUBER Tatjana, št. 4, str. 284. Integracija spletnega portala za e-poslovanje s podjetji z ostalimi informacijskimi sistemi v trgovski organizaciji.

Š
ŠAKANOVIČ Zlatko, MAYER Janez, št. 4, str. 254. Nekateri vidiki vodenja in njihov vpliv na organizacijsko klimo in zadovoljstvo zaposlenih v slovenski policiji.
ŠKRABA Andrej, KOFJAČ Davorin, ROZMAN Črtomir, št. 10, str. 676. The Periodicity of the Anticipative Discrete Demand - Supply Model.
ŠOLAR Slavko V., ŽIBRET Gorazd, št. 5, str. 339. Evropski kazalci trajnosti za področje mineralnih surovin na nacionalni ravni kot orodje za podporo pri odločanju in mednarodno primerjavo.
ŠORGO Andrej, KOCIJANČIČ Saša F., št. 8, str. 513. Naravoslovni eksperiment : most med šolskim znanjem in vsakdanjimi izkušnjami.
ŠTEMPELJ Boštjan, št. 5, str. 321. Vpeljava mentorstva v proces prodaje storitev: primer zavarovalnice.
ŠULIGOJ Metod, št. 2, str. 146. Delovni pogoji v gostinski dejavnosti.
ŠUMANSKI MIKLAVČIČ Martina, KOLENC Igor,

MARKIČ Mirko, št. 9, str. 597. Socialno emocionalni odnosi kot spodbuda za oblikovanje novih znanj.
ŠUŠTERŠIČ Olga, RAJKOVIČ Uroš, ZUPANČIČ Jože, št. 10, str. 687. A model of e-documentation of community nursing.

T
TOMŠIČ Matevž, REK Mateja, št. 1, str. 28. Kulturni vidiki ekonomskega in političnega razvoja : stare in nove članice EU v primerjalni perspektivi.

U
UKOVICH Walter , INCHINGOLO Paolo, št. 3, str. 207. Contributions of ALADIN to eRegion Development: The Perspective of the University of Trieste.

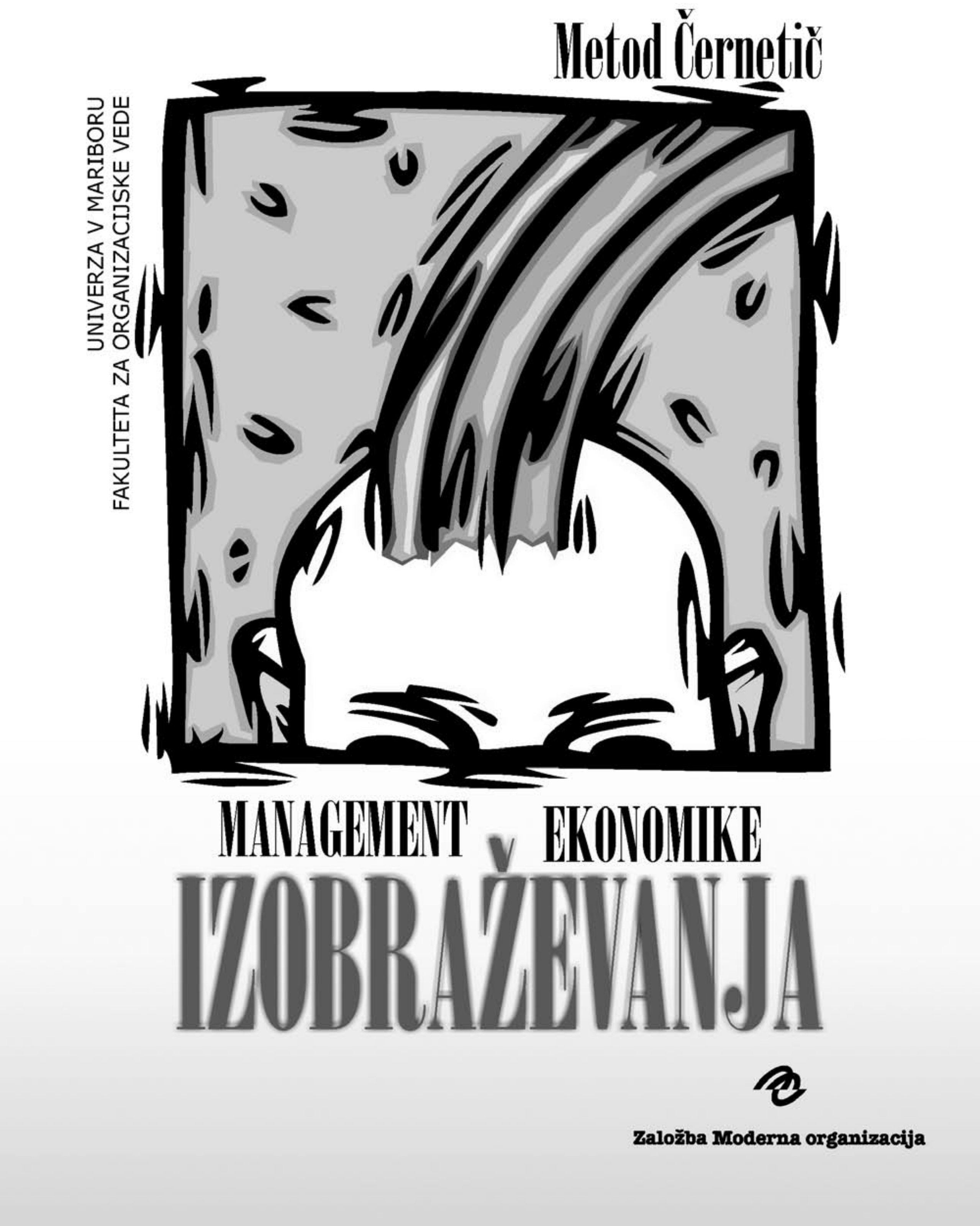
V
VESELKO Gregor, št. 5, str.346. Management oskrbovalne verige v elektronskem poslovnem okolju.
VIDULIN, Vedrana, št. 2, str. 154. Constructivist Learning Theory as a Link between Artificial Neural Networks and Intelligent Tutoring Systems.
VINTAR Mirko, JUKIĆ Tina, št. 3, str. 176. E-government: the State in Slovenian Local Self-government.
VIRT Tina, BUNETO Andreja, MUMEL Damijan, št.6, str. 361. Uporaba instrumentov internega komuniciranja ter cilji internega komuniciranja v velikih in malih podjetjih v Sloveniji.
VOGEL Douglas R., GRIČAR Jože, št. 3, str. 214. eSilk&Amber Road Regions Meeting: Business & Government Executives & Professors
VOLK ROŽIČ Nevenka, RASPOR Andrej, št. 6, str. 378. Posredovanje dela.

W
WECHTERSBACH Rado, št. 8, str. 469. Informacijska revolucija v izobraževanju.

Y
YOLLES Maurice, št. 1, str. 5. Understanding the Intelligent Organization.

Z
ZAVRŠNIK Bruno, JERMAN Damjana, št. 6, str. 373.Vpliv marketinških sposobnosti na porast inovativnosti v organizacijah.
ZELKO Miroslav, LAVRIN Anton, št. 3, str. 191. Knowledge Sharing in Regional Digital Ecosystems.
ZUPAN Lucija, BREZAVŠČEK Alenka, št. 1, str. 58. Novosti, ki jih prinašajo spremembe standarda BS 7799.
ZUPANČIČ Jože, FLERIN Zdene, št. 1, str. 67. Uporabniška rešitev za zagotavljanje sledljivosti in učinkovitosti rezalnikov kartona.
ZUPANČIČ Jože, RAJKOVIČ Uroš, ŠUŠTERŠIČ Olga, št. 10, str. 687. A model of e-documentation of community nursing.

Ž
ŽIBERT Marko, KLJAJIĆ Miroljub, št. 10, str. 626. Modelling Parallel Service Systems in GPSS.
ŽIBRET Gorazd, ŠOLAR Slavko V., št. 5, str. 339. Evropski kazalci trajnosti za področje mineralnih surovin na nacionalni ravni kot orodje za podporo pri odločanju in mednarodno primerjavo.
ŽNIDARŠIČ Boštjan, DINEVSKI Dejan, JAKONČIČ FAGANEL Janja, LOKAR Matija, št. 8, str. 498. Model ocenjevanja kakovosti elektronskih učnih gradiv.



Navodila avtorjem prispevkov

V reviji Organizacija praviloma objavljamo dela s predmetnega področja revije, ki še niso bila objavljena in niso bila poslana v objavo v kakšni drugi reviji ali zborniku. Pisec je odgovoren za vse morebitne kršitve avtorskih pravic. Če je bil prispevek že natisnjen drugje, poslan v objavo ali predstavljen na strokovni konferenci, mora avtor to sporočiti, pridobiti soglasje založnika, če je potrebno, in navesti razloge za ponovno objavo.

V Organizaciji objavljamo **razprave** (znanstvene članke, rezultate raziskovalnega dela avtorjev, ali pregledne članke), **predloge za prakso** (strokovne članke, na primer prikaze in ocene pristopov in metod in njihove uporabe v praksi), **razmišljanja** (krajši prispevki), **informacije** in **knjižne ocene**. Občasno vključujemo tudi odmeve na objavljene prispevke, enciklopedične razlage, intervjuje s strokovnjaki s predmetnega področja revije in druga besedila. Približne omejitve dolžine prispevkov so naslednje:

- razprave in predlogi za prakso: največ **45.000 znakov,** vključno s presledki
- razmišljanja, informacije: do **10.000** znakov
- knjižne ocene, odmevi: do **5.000** znakov.

Znanstvene članke (razprave) objavljamo praviloma v angleščini, druge prispevka pa v angleščini ali slovenščini. Razprave in predloge za prakso **ocenita vsaj dva recenzenta,** druge prispevke pa uredniški odbor ali urednik. Na osnovi mnenj recenzentov in sourednikov uredniški odbor ali urednik sprejmejo prispevek, zahtevajo manjše ali večje popravke, dopolnitve ali ga zavrnejo. Če urednik oziroma recenzenti predlagajo večje popravke, se prispevek praviloma ponovno pošlje v recenzijo. Urednik lahko sprejeti prispevek pošlje v lektoriranje. Lektorirana besedila se lahko vrnejo avtorju v pregled.

Besedilo naj bo oblikovano za tiskanje na papirju formata A4 levo poravnano, z razmakom med vrsticami **vsaj 1,5**. Razpravam in predlogom za prakso naj bo dodan **povzetek** (izvleček) dolg 10-20 vrstic, **ključne besede** in kratek **strokovni življenjepis** vsakega od avtorjev (do 10 vrstic) in **letnica rojstva** (zaradi vnosa podatkov v knjižnični informacijski sistem COBISS, v reviji letnica ne bo objavljena). Na prvi strani besedila naj bodo napisani le naslov prispevka, imena ter poštni in elektronski naslovi avtorjev članka, po možnosti tudi telefonska številka enega od avtorjev. Da bi zagotovili anonimnost recenziranja, naj se imena avtorjev ne pojavljajo v besedilu prispevka.

Članek naj bo razčlenjen v oštevilčena poglavja. Naslovi članka, poglavij in podpoglavij naj bodo napisani z malimi črkami, da so razvidne kratice. **Povzetek**, **naslov članka** in **ključne besede** naj bodo tudi prevedene slovenščino, pri prispevkih, ki so napisani v slovenščini, pa v angleščino.

Slike in tabele v elektronski obliki vključite ~~kar~~ v besedilo. Oštevilčite jih z zaporednimi številkami (Slika 1, slika 2, …Tabela 1, Tabela 2, …). Vsaka slika mora imeti podnapis, tabela pa nadnapis (naslov). Vsaka tabela oz. slika mora biti vsaj enkrat omenjena v besedilu. Besedilu so lahko priložene slike in/ali tabele na papirju v obliki pripravljeni za preslikavo. V tem primeru naj bo vsaka slika na posebnem listu, oštevilčene naj bodo z arabskimi številkami, v besedilu naj bo označeno, kam približno je treba uvrstiti sliko: na tem mestu naj bo številka slike/ tabele in njen podnapis. Slike bomo praviloma pomanjšali in jih vstavili v članek. Upoštevajte, da morajo biti oznake in besedila na vseh slikah naj bodo dovolj velika, da bo bodo čitljiva tudi pri velikosti slike, kot bo objavljena v reviji.Vse slike naj bodo črno-bele z belim ozadjem; barvnih slik ne moremo objaviti.

Pri sklicevanju na literaturo med besedilom navedite le priimek prvega avtorja, oziroma prvega in drugega (glej vzorec), letnico izdaje, lahko tudi stran. Popolni bibliografski podatki naj bodo v seznamu literature in / ali virov na koncu prispevka, urejeni po abecednem redu (prvih) avtorjev, literatura istega avtorja pa po kronološkem redu izida; če navajate dve ali več del nekega avtorja oziroma avtorjev, ki so izšla v istem letu, uporabite črkovno oznako pri letnici, na primer 2003a, 2003b, ... V seznamu literature in/ali virov ne navajte del, ki jih ne omenjate v besedilu članka. Ne uporabljajte opomb za citiranje; eventualne opombe, ki naj bodo kratke, navedite na dnu strani. Označite jih z arabskimi številkami.

V seznamu lahko ločite literaturo (članki v revijah, knjige, zborniki konferenc, doktorske disertacije, …) in vire (dokumenti, zakoni, standardi, interni viri, …). Pri citiranju literature uporabite način, ki je prikazan na naslednjih primerih:

- "... v nasprotju z (Novak in Vajda, 1996:123) raziskava (Wilkinson et al., 2001: 234) nakazuje, da ..."
- "... kot poročata Smith (2003) in Jankowski (2004) metodo uporabljajo za ..."
- "... kot ugotavljajo nekateri drugi avtorji (Zima 1999; Novak in Vajda, 1996; Wilkinson et al., 1993), številna podjetja ....".

Bibliografske podatke v seznamu literature navajte na »harvardski način«, kot to kažejo vzorci v nadaljevanju. Podroben opis tega načina najdete na http:// …

Članek v reviji:

Novak, A. & Vajda, B.M. (1996). Effect of surface runoff water on quality easurement, *European Journal of Information Systems*, **31**(4): 31 - 39.

Zraven letnika v oklepaju navedite številko v letniku le, če se vsaka številka začne s stranjo 1. Če revija nima letnika, lahko navedete mesec ali drugo ustrezno oznako, na primer **Poletje 1999.**

Članek v elektronski reviji:

Lynch T. & Szorenyi Z. (2005). Dilemmas surrounding information technology education in developing countries, *The Electronic Journal of Information Systems in Developing Countries*, **21**(4): 1-16, dosegljivo na: http:// www.ejisdc.org (22.8.2005).

Knjiga:

Smith, S.I. (2003). *Interpreting Information Systems in Organizations*, Elsevier Publishing, New York.

Poglavje v knjigi:

Zupan, N. & Leskovar, R. (2002). Pričakovanja v zvezi z elektronskim poslovanjem v malih organizacijah. *Organizacija in management – izbrana poglavja.* Uredila: Florjančič J., & Paape, B. Kranj: Založba Moderna organizacija.

Referat objavljen v zborniku konference:

Wilkinson, K.J., Kumar, R. & Kumar, S. (2001). We can do better: integrating theories of novel organizations, *Proceedings of the Twelfth European Conference on Information Systems.* Uredil: Johnson, M. Bled 12-14 jun. 2001. Berlin: Springer Verlag.

Diploma, magisterij ali doktorat:

Zima, B. (1999). Analiza potrebnih znanj diplomiranih informatikov v Sloveniji, magistrsko delo, Univerza v Mariboru, Fakulteta za organizacijske vede.

Poročila, interni dokumenti, zakoni, …:

ACM (1994). ACM SIGCHI Curricula for Human-Computer Interaction, The Association for Computing Machinery, New York.

Zakon o elektronskem poslovanju in elektronskem podpisu (ZEPEP), Ur.l. RS, št. 57/2000, 30/2001.

Pri **internetnih virih / literaturi** naj bo poleg (eventualnega avtorja in) naslova besedila naveden tudi internetni naslov vira (URL) in datum dostopa do dokumenta.

Banka Slovenije, Basel II – Nov kapitalski sporazum, dosegljivo na: http://www.bsi.si/html/basel2/default.htm (6.4.2006).

V literaturi ne navajajte internetnih naslovov (URL) brez drugih podatkov. Lahko pa se nanje sklicujete v besedilu ali v opombah na dnu strani.

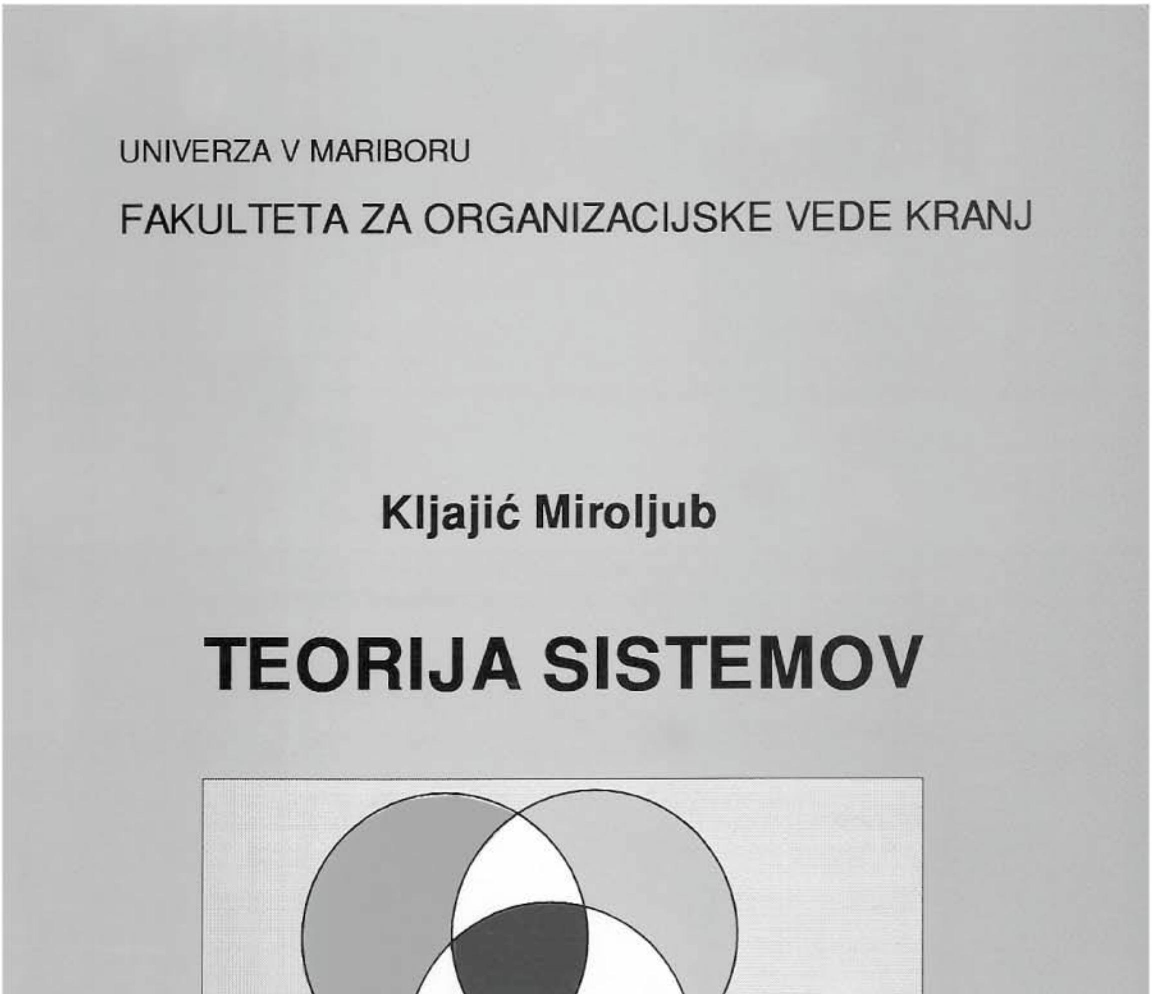
Prispevek v elektronski obliki (po možnosti kot eno Word-ovo datoteko) predložite preko spletne strani **http://organizacija.fov.uni-mb.si.** Vprašanja v zvezi s predložitvijo in objavo člankov lahko pošljete na organizacija@fov.uni-mb.si (uredništvo). Lahko tudi vprašate po telefonu 04-2374-226 (Dunja Škofic). Datoteko poimenujte z imenom (prvega) avtorja ali avtorice, na primer *Kopac.doc.*

Naslov uredništva je:

Univerza v Mariboru
Fakulteta za organizacijske vede
Uredništvo revije Organizacija
Kidričeva cesta 55a
4000 KRANJ
e-pošta: Organizacija@fov.uni-mb.si
tel.: 04 2374-226
faks: 04 2374-299
URL: http://organizacija.fov.uni-mb.si/

Prva slovenska revija za organizacijska in kadrovska raziskovanja in prakso. Revijo sofinancira Javna agencija za raziskovalno dejavnost Republike Slovenije.
Ponatis in razmnoževanje deloma ali v celoti brez pisnega dovoljenja nista dovoljena. Izdajatelj: Univerza v Mariboru, Fakulteta za organizacijske vede Kranj,
Založba MODERNA ORGANIZACIJA, Kidričeva cesta 55a, KRANJ, telefon: 04 23 74 374,
telefax: 04 23 74 299, E-pošta: OMIK@FOV.UNI-MB.SI. Uredništvo revije: Kidričeva cesta 55a, 4000 Kranj, naročniški oddelek: 04 23 74 295.
Letna naročnina: za pravne osebe za prvi naročeni izvod 16.900 SIT (70,52 EUR), drugi naročeni izvod 13.700 SIT (57,17 EUR),
vsak nadaljnji 12.100 SIT (50,49 EUR), za posameznike 8.300 SIT (34.64 EUR). Cena posamezne številke je 1.865 SIT (7,51 EUR).
Na leto izide 10 številk. Grafično oblikovanje: L.M. Ljubljana. Priprava in tisk: GRAFIKA SOČA, Nova Gorica. Naklada 3000 izvodov.
Revija Organizacija je indeksirana v naslednjih bazah: INSPEC, ERGONOMIC ABSTRACT in CSA SOCIOLOGICAL ABSTRACTS.

Založba Moderna organizacija



UNIVERZA V MARIBORU

FAKULTETA ZA ORGANIZACIJSKE VEDE KRANJ

Kljajić Miroljub

TEORIJA SISTEMOV

moderna organizacija

Knjigo lahko naročite na naslov:
Univerza v Mariboru, Fakulteta za organizacijske vede, Založba Moderna organizacija,
Kidričeva cesta 55a, 4000 Kranj, tel.: 04 2374 390, fax: o4 2374 299