271/2004

Raport Badawczy
Research Report

RB/17/2004

Application of uncertain variables and learning process to decision making in a class of complex systems

Z. Bubnicki

Instytut Badań Systemowych
Polska Akademia Nauk

Systems Research Institute
Polish Academy of Sciences

POLSKA AKADEMIA NAUK

Instytut Badań Systemowych

ul. Newelska 6

01-447 Warszawa

tel.: (+48) (22) 8373578

fax: (+48) (22) 8372772

Kierownik Pracowni zgłaszający pracę:
Prof. dr hab. inż. Zdzisław Bubnicki

Warszawa 2004

APPLICATION OF UNCERTAIN VARIABLES AND LEARNING PROCESS TO DECISION MAKING IN A CLASS OF COMPLEX SYSTEMS

Z. Bubnicki[1), 2)]

[1)]Institute of Control and Systems Engineering, Wroclaw University of Technology

Wyb. Wyspianskiego 27, 50-370 Wroclaw, POLAND,

e-mail: zdzislaw.bubnicki@pwr.wroc.pl, tel: +48 71 320 33 28, fax: +48 71 320 38 84

[2)]Systems Research Institute of Polish Academy of Sciences, Laboratory of Knowledge Systems and Artificial Intelligence, Podwale St. 75, 50–449 Wroclaw, POLAND

ABSTRACT

In recent years two new concepts have been introduced and developed as a tool for decision making in a class of uncertain systems (in particular, control systems) described by a relational knowledge representation: uncertain variables characterized by an expert and a learning process consisting in *step by step* knowledge validation and updating. The purpose of this paper is to show how these two approaches may be combined and used for decision making in a class of complex systems with a distributed knowledge.

In the first part of the paper the decision problem based on the uncertain variables and the learning process are shortly described. Then the combination of these approaches in one learning system in which the expert's knowledge is modified according to current results of the learning is described and the algorithm of the decision making in the learning system is presented. In the second part it is shown how to apply these concepts in two cases of complex systems: two-level system with the distributed knowledge and a complex of parallel operations. A simple example and a result of simulations illustrate the presented approach.

KEY WORDS

intelligent control systems, learning systems, uncertain variables, uncertain systems

1. Introduction

The paper is concerned with a class of uncertain systems described by a set of relations between the variables characterizing the decisions and their effects (a relational knowledge representation). There exists a great variety of definitions and formal descriptions of uncertainties and uncertain systems (see e.g. [1, 2, 3]). In recent years two new concepts have been introduced and developed for the decision making in uncertain relational systems with unknown parameters [4, 5, 6]:

1. Uncertain variables described by certainty distributions given by an expert.
2. Learning processes consisting in *step by step* knowledge validation and updating.

It has been shown how these two approaches may be applied in a class of complex knowledge systems [7, 8, 9]. The purpose of this paper is to present a new idea based on the combination of the learning process and the uncertain variables for the same class of systems. In the system under consideration, at each step of the learning process an expert's knowledge is modified according to the current result of the learning.

Short descriptions of the uncertain variables and the learning process without the additional expert's knowledge are presented in Secs 2 and 3. The details may be found in [5, 6]. The concept of the learning system based on the current expert's knowledge is described in Sections 4 and 5. Sections 6 and 7 present the application of this concept in two cases of a complex uncertain system with the distributed knowledge: a two-level system and a group of parallel processors [7, 8, 10]. In the second case, the application of the uncertain variables may be compared with the application of probabilistic descriptions of allocation and scheduling problems in the systems with uncertain execution times (see e.g. [11]).

2. Uncertain Variables

In the definition of the uncertain variable $\bar{x}$ we consider two soft properties (i.e. such properties $\varphi(x)$ that for the fixed x the logic value $v[\varphi(x)] \in [0,1]$): "$\bar{x} \cong x$" which means "$\bar{x}$ is approximately equal to x" or "x is the approximate value of $\bar{x}$", and "$\bar{x} \,\tilde{\in}\, D_x$" which means "$\bar{x}$ approximately belongs to the set D_x" or "the approximate value of $\bar{x}$ belongs to D_x". The ***uncertain variable*** $\bar{x}$ is defined by a set of values X (real number vector space), the function $h(x) = v(\bar{x} \cong x)$ (i.e. the certainty index that $\bar{x} \cong x$, given by an expert) and the following definitions for $D_x, D_1, D_2 \subseteq X$:

$$v(\bar{x} \,\tilde{\in}\, D_x) = \max_{x \in D_x} h(x),$$

$$v(\bar{x} \,\tilde{\notin}\, D_x) = 1 - v(\bar{x} \,\tilde{\in}\, D_x),$$

$$v(\bar{x} \,\tilde{\in}\, D_1 \vee \bar{x} \,\tilde{\in}\, D_2) = \max\{v(\bar{x} \,\tilde{\in}\, D_1), v(\bar{x} \,\tilde{\in}\, D_2)\},$$

$$v(\bar{x} \,\tilde{\in}\, D_1 \wedge \bar{x} \,\tilde{\in}\, D_2) = \min\{v(\bar{x} \,\tilde{\in}\, D_1), v(\bar{x} \,\tilde{\in}\, D_2)\}.$$

The function $h(x)$ is called a *certainty distribution*.

Let us consider a static plant with the input vector $u \in U$ and the output vector $y \in Y$, described by a relation $R(u,y;x) \subset U \times Y$ where the vector of unknown parameters $x \in X$ is assumed to be a value of an uncertain variable described by the certainty distribution $h(x)$ given by an expert. If the relation R is not a function then the value u determines a set of possible outputs

$$D_y(u;x) = \{y \in Y : (u,y) \in R(u,y;x)\}.$$

For the requirement $y \in D_y \subset Y$ given by a user, we can formulate the following **decision problem**: For the given $R(u,y;x)$, $h(x)$ and D_y one should find the decision u^* maximizing the certainty index that the set of possible outputs approximately belongs to D_y (i.e. belongs to D_y for an approximate value of $\bar{x}$). Then

$$u^* = \arg\max_{u \in U} v[D_y(u;\bar{x}) \mathrel{\tilde{\subseteq}} D_y] = \arg\max_{u \in U} \max_{x \in D_x(u)} h(x) \qquad (1)$$

where $D_x(u) = \{x \in X : D_y(u;x) \subseteq D_y\}$. It is easy to see that u^* maximizes $v[u \mathrel{\tilde{\in}} D_u(\bar{x})]$ where $D_u(x)$ is a set of all u such that the implication $u \in D_u(x) \rightarrow y \in D_y$ is satisfied.

3. Learning Process

Assume now that the parameter x in the relation R has the value $x = a$ and a is unknown. Then, for the fixed value u it is not known if u is a correct decision, i.e. if $u \in D_u(a)$ and consequently $y \in D_y$. Our problem may be considered as a classification problem with two classes. The point u should be classified to class $j = 1$ if $u \in D_u(a)$ and to class $j = 2$ if $u \notin D_u(a)$. Assume that we can use the learning sequence $(u_1, j_1), (u_2, j_2), \ldots, (u_n, j_n) \stackrel{\Delta}{=} S_n$ where $j_i \in \{1, 2\}$ are the results of the correct classification given by an external trainer.

Let us denote by $\bar{u}_i$ the subsequence for which $j_i = 1$, i.e. $\bar{u}_i \in D_u(a)$ and by $\hat{u}_i$ the subsequence for which $j_i = 2$, and introduce the following sets in X:

$$\bar{D}_x(n) = \{x \in X : \bar{u}_i \in D_u(x) \text{ for every } \bar{u}_i \text{ in } S_n\}, \quad (2)$$

$$\hat{D}_x(n) = \{x \in X : \hat{u}_i \in U - D_u(x) \text{ for every } \hat{u}_i \text{ in } S_n\}. \quad (3)$$

The set

$$\bar{D}_x(n) \cap \hat{D}_x(n) \stackrel{\Delta}{=} \Delta_x(n)$$

may be proposed as the estimation of a. The determination of $\Delta_x(n)$ may be presented in the form of the following recursive algorithm:

If $j_n = 1$ $(u_n = \bar{u}_n)$.

1. **Knowledge validation** for $\bar{u}_n$. Prove if

$$\bigwedge_{x \in \bar{D}_x(n-1)} u_n \in D_u(x) .$$

If yes then $\bar{D}_x(n) = \bar{D}_x(n-1)$. If not then one should determine the new $\bar{D}_x(n)$, i.e. update the knowledge.

2. **Knowledge updating** for $\bar{u}_n$

$$\bar{D}_x(n) = \{x \in \bar{D}_x(n-1) : u_n \in D_u(x)\} .$$

Put $\hat{D}_x(n) = \hat{D}_x(n-1)$.

If $j_n = 2$ $(u_n = \hat{u}_n)$.

3. **Knowledge validation** for $\hat{u}_n$. Prove if

$$\bigwedge_{x \in \hat{D}_x(n-1)} [u_n \in U - D_u(x)] .$$

If yes then $\hat{D}_x(n) = \hat{D}_x(n-1)$. If not then one should determine the new $\hat{D}_x(n)$, i.e. update the knowledge.

4. **Knowledge updating** for $\hat{u}_n$

$$\hat{D}_x(n) = \{x \in \hat{D}_x(n-1) : u_n \in U - D_u(x)\} .$$

Put $\overline{D}_x(n) = \overline{D}_x(n-1)$ and $\Delta_x(n) = \overline{D}_x(n) \cap \hat{D}_x(n)$.

The successive estimation of a may be performed in a closed-loop learning system where u_i is the sequence of the decisions. The decision making algorithm is as follows:

1. Put u_n at the input of the plant and introduce y_n.

2. Determine $\Delta_x(n)$ using the estimation algorithm with knowledge validation and updating.

3. Choose randomly x_n from $\Delta_x(n)$, put x_n into $D_u(x)$ and choose randomly u_{n+1} from $D_u(x_n)$.

4. Learning System Based on Current Expert's Knowledge

At the n-th step, the result of the learning process in the form of a set $\Delta_x(n)$ may be used to present an expert's knowledge in the form of a certainty distribution $h_n(x)$ such that $h_n(x) = 0$ for every $x \notin \Delta_x(n)$. Thus, the expert formulates his / her current knowledge, using his / her experience and the current result of the learning process based on the knowledge of the external trainer. In particular, $h_n(x) = h(x, b_n)$, i.e. the form of the certainty distribution is fixed, but the parameter b_n (in general, b_n is a vector of parameters) is currently adjusted. For example, if in one-dimensional case $\Delta_x(n) = [x_{\min,n}, x_{\max,n})$ and $h_n(x) = h(x, x_n^*, d_n)$ has a triangular form presented in Fig. 1, then $b_n = (x_n^*, d_n)$ and

$$x_n^* = \frac{x_{\min,n} + x_{\max,n}}{2}, \quad d_n = \frac{x_{\max,n} - x_{\min,n}}{2}. \qquad (4)$$

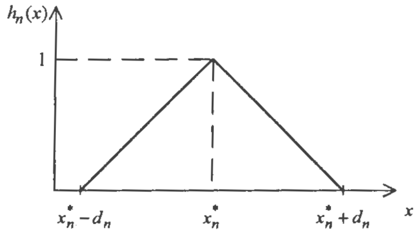

Figure 1. Example of certainty distribution.

For $h_n(x)$ the next decision u_{n+1} may be determined in the way presented in Sec. 2, i.e.

$$u_{n+1} = \arg\max v_n(u)$$

where

$$v_n(u) = v[u \mathrel{\tilde{\in}} D_u(\bar{x})] = v[\bar{x} \mathrel{\tilde{\in}} D_x(u)] = \max_{x \in D_x(u)} h_n(x), \qquad (5)$$

and $D_x(u) = \{x \in X : u \in D_u(x)\}$. In general, as a result of the maximization of $v_n(u)$ one may obtain a set of decisions $D_{u,n+1}$. For $h(x, b_n)$ we obtain the fixed form of the function $v(u, b_n)$:

$$v_n(u) = \max_{x \in D_x(u)} h(x, b_n) \stackrel{\Delta}{=} v(u, b_n)$$

and consequently, the fixed form of the final result, i.e. one decision $u_{n+1} = u(b_n)$ or the set of decisions $D_{u,n+1} = D_u(b_n)$.

The block scheme of the learning decision making system under consideration is presented in Fig. 2 where G is a generator of random variables for the random choosing of u_{n+1} from $D_{u,n+1}$. The blocks in the figure illustrate parts of the computer decision system or parts of the program which may be used for the computer simulations.

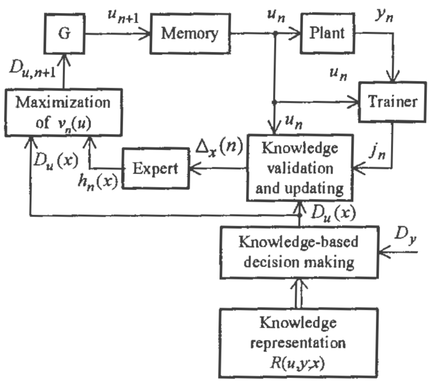

Figure 2. Block scheme of the knowledge-based decision system.

5. Example

Consider a decision plant with the input vector $\tilde{u}$, one-dimensional output y and one-dimensional unknown parameter x, described by the relation

$$y \le (\widetilde{u}^{\mathrm{T}}\widetilde{u})x^{-1}, \ x > 0.$$

The components $u^{(l)}$ of u may denote some features of a raw material in a manufacturing process, which may be chosen as decisions, and y may denote a cost of the process. For the requirement $y \le \bar{y}^2$ we obtain the set $D_u(x)$ described by inequality

$$u^{\mathrm{T}}u \le x$$

where $u^{(l)} = \widetilde{u}^{(l)} \cdot (\bar{y})^{-1}$. In this case, according to (2) and (3)

$$\overline{D}_x(n) = [x_{\min,n}, \infty), \qquad \hat{D}_x(n) = [0, x_{\max,n}),$$

$$\Delta_x(n) = [x_{\min,n}, x_{\max,n})$$

where

$$x_{\min,n} = \max_i \bar{u}_i^{\mathrm{T}} \bar{u}_i, \quad x_{\max,n} = \min_i \hat{u}_i^{\mathrm{T}} \hat{u}_i.$$

The estimation algorithm with the knowledge validation and updating is then as follows:

1. Put u_n at the input and introduce j_n.

2. For $j_n = 1 \quad (u_n = \bar{u}_n)$, prove if

$$u_n^{\mathrm{T}} u_n \le x_{\min,n-1}.$$

If yes then $x_{\min,n} = x_{\min,n-1}$. If not, $x_{\min,n} = u_n^{\mathrm{T}} u_n$.

Put $x_{\max,n} = x_{\max,n-1}$.

3. For $j_n = 2 \quad (u_n = \hat{u}_n)$, prove if

$$u_n^{\mathrm{T}} u_n \ge x_{\max,n-1}.$$

If yes then $x_{\max,n} = x_{\max,n-1}$. If not, $x_{\max,n} = u_n^{\mathrm{T}} u_n$.

Put $x_{\min,n} = x_{\min,n-1}$, $\Delta_x(n) = [x_{\min,n}, x_{\max,n})$.

Let us assume that $h_n(x)$ has a triangular form presented in Fig. 1 where x_n^* and d_n are determined by (4). Using (5) it is easy to obtain the certainty index that u "approximately" belongs to the set $D_u(x)$:

$$v_n(\alpha) = \begin{cases} 1 & \text{for} \quad \alpha \le x_n^* \\ -d_n^{-1}(d_n - x_n^*) + 1 & \text{for} \quad x_n^* \le \alpha \le x_n^* + d_n \\ 0 & \text{for} \quad \alpha \ge x_n^* + d_n \end{cases}$$

where $\alpha = u^{\mathrm{T}} u$. Then

$$D_{u,n+1} = \{u_{n+1} \in U : u_{n+1}^{\mathrm{T}} u_{n+1} \le x_n^*\} \qquad (6)$$

and for every u_{n+1} from this set $v_n(u_{n+1}) = 1$. Consequently, the decision making algorithm in the learning system is the following:

1. Put u_n at the input and introduce j_n.

2. Determine $x_{\min,n}$ and $x_{\max,n}$ using the estimation algorithm with the knowledge validation and updating.

3. Choose randomly u_{n+1} from the set (6).

Assume that in $u^{\mathrm{T}} u < x$ $x = a$, i.e. a is the unknown value of x. From (6) it is easy to note that x_n^* may be considered as an estimation of a. Under some conditions, in the same way as in [6], it may be proved that x_n^* converges to a with probability 1. Figure 3 presents the result of simulations for the following data: $a = 5$, $\alpha_0 = u_0^{\mathrm{T}} u_0 = 20$, $\alpha_{n+1} = u_{n+1}^{\mathrm{T}} u_{n+1}$ is chosen randomly from the interval $[0, x_n^*]$ with the rectangular probability density.

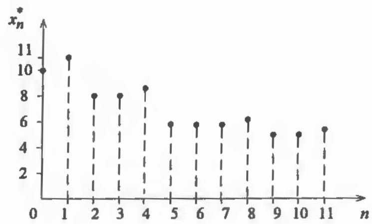

Figure 3. Results of simulation.

6. Two-Level System

Let us consider a distributed system with a structure presented in Fig. 4 where u_j, x_j, y are real number vectors. The upper and lower level subsystems are described by relations $R(w,y;c)$, $R(u_j,y_j;x_j)$, $j=1,...,k$ where $w=(w_1,...,w_k)$. This set of relations forms a **distributed knowledge representation**. The unknown parameters c and x_j are assumed to be values of uncertain variables described by the certainty distributions $h_c(c)$ and $h_{xj}(x_j)$, respectively. The decision problem consists in finding the decisions u which maximize the certainty index that the set of all possible y approximately belongs to the set D_y given by a user. The problem may be decomposed into two levels [6, 7, 8]. On the upper level one should find w maximizing the certainty index that the set of possible y approximately belongs to D_y, and on the lower level one should find u_j $(j=\overline{1,k})$ maximizing the certainty index that the set of possible w_j

approximately belongs to the set obtained as a result of the upper level. Then to each level one can apply the approach presented in Sec. 4.

In [7] it has been shown how the learning process may be decomposed into two levels. At the n-th step we obtain $\Delta_c(n)$ as the estimation of c on the upper level and the sets $\Delta_{xj}(n)$ as the estimations of x_j on the lower level. Consequently, we can determine w_{n+1} for $\Delta_c(n)$ and $u_{j,n+1}$ for $\Delta_{xj}(n)$ in the same way as u_n for $\Delta_x(n)$ in Sec. 4.

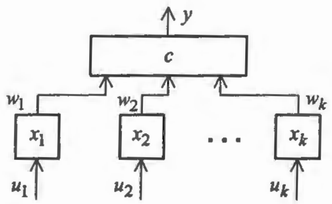

Figure 4. Two-level system.

7. Task Allocation in the Complex of Parallel Operations

The uncertain variables may be applied to allocation problems consisting in the proper task or resource distribution in a complex of operations described by a relational knowledge representation with unknown parameters [6]. The parts of the complex may denote manufacturing operations [10], computational operations in a computer system [7] or operations in a project to be managed. Let us consider a complex of k parallel operations described by a set of inequalities

$$T_i \le \varphi_i(u_i, x_i), \quad i = 1, 2, \ldots, k \qquad (7)$$

where T_i is the execution time of the i-th operation, u_i is the size of a task in the problem of task allocation or the amount of a resource in the problem of resource allocation, an unknown parameter $x_i \in R^1$ is a value of an uncertain variable $\bar{x}_i$ described by a certainty distribution $h_i(x_i)$ given by an expert, and $\bar{x}_1, ..., \bar{x}_k$ are independent variables. The complex may be considered as a decision plant described in Sec. 2 where y is the execution time of the whole complex $T = \max\{T_1, ..., T_k\}$, $x = (x_1, ..., x_k)$, $u = (u_1, ..., u_k) \in \overline{U}$. The set $\overline{U} \subset R^k$ is determined by the constraints: $u_i \geq 0$ for each i and $u_1 + ... + u_k = U$ where U is the total size of the task or the total amount of the resource to be distributed among the operations (Fig. 5).

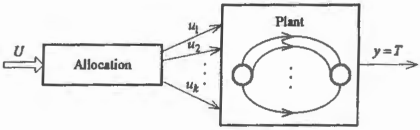

Figure 5. Complex of parallel operations as a decision plant.

According to the general formulation of the decision problem presented in Sec. 2, the allocation problem may be formulated as an optimization problem consisting in finding the optimal allocation u^* that maximizes the certainty index of the soft property: "the set of possible values T approximately belongs to $[0, \alpha]$" (i.e. belongs to $[0, \alpha]$ for an approximate value of $\bar{x}$). **Optimal allocation problem**: For the given φ_i, h_i $(i \in \overline{1,k})$, U and α find

$$u^* = \arg\max_{u \in \overline{U}} v(u)$$

where

$$v(u) = v\{D_T(u;\bar{x}) \tilde{\subseteq} [0, \alpha]\} = v(T(u, \bar{x}) \tilde{\leq} \alpha).$$

The soft property "$D_T(u;\bar{x}) \tilde{\subseteq} [0,\alpha]$" is denoted here by "$T(u,\bar{x}) \tilde{\leq} \alpha$", and $D_T(u;x)$ denotes the set of possible values T for the fixed u, determined by the inequality

$$T \leq \max_i \varphi_i(u_i, x_i) .$$

According to (7)

$$v(u) = v\{[T_1(u_1,\bar{x}_1) \tilde{\leq} \alpha)] \wedge ... \wedge [T_k(u_k,\bar{x}_k) \tilde{\leq} \alpha)]\} .$$

Then

$$u^* = \arg \max_{u \in \overline{U}} \min_i v_i(u_i) \qquad (8)$$

where

$$v_i(u_i) = v[T_i(u_i,\bar{x}_i) \tilde{\leq} \alpha)] = v[\varphi_i(u_i,\bar{x}_i) \tilde{\leq} \alpha)] = v[\bar{x}_i \tilde{\in} D_i(u_i)],$$

$$D_i(u_i) = \{x_i \in R^1 : \varphi_i(u_i, x_i) \leq \alpha\} .$$

Finally

$$v_i(u_i) = \max_{x_i \in D_i(u_i)} h_i(x_i) \qquad (9)$$

and

$$u^* = \arg\max_{u \in \overline{U}} \min_i \max_{x_i \in D_i(u_i)} h_i(x_i) .$$

In [7] it has been shown that the learning algorithm described in Sec.3 may be executed separately for the executors $1, 2, ..., k$. As a result one obtains the sets $\Delta_{xj}(n)$ as the estimations of x_j in the j-th operation. Consequently, one can determine $u_{j,n+1}$ for $\Delta_{xj}(n)$ in the same way as u_n for $\Delta_x(n)$ in Sec.4.

8. Conclusions

The uncertain variables have been proved to be a convenient tool for the analysis and design of the distributed knowledge systems under consideration. For the class of distributed systems considered in the paper, at each step of the learning process it is possible to use the current expert's knowledge based on the uncertain variables. The computer system illustrated in Fig. 2 has been implemented and used for simulations, for two cases of the distributed plant described in Sections 6 and 7. The simulations showed that the using of the expert's knowledge during the learning process may improve the quality of the decisions. The presented approach may be applied to computer supported manufacturing systems with the distributed knowledge [10], to control of a transportation system based on uncertain variables description [12, 13] and to systems with uncertain and random parameters [14].

9. Acknowledgement

This work was supported by the Polish State Committee for Scientific Research under Grant No. 4 T11C 001 22.

References

[1] K. C. Ranze and H. Stuckenschmidt, Modelling uncertainty in expertise, *Proc. of XV IFIP World Computer Congress. Information Technologies and Knowledge Systems,* Vienna, Austria, 1998, 105–116.

[2] G. J. Klir and T. A. Folger, *Fuzzy sets, uncertainty, and information* (Englewood Cliffs, NJ: Prentice-Hall, 1988).

[3] R. R. Yager, Uncertainty representation using fuzzy measures, *IEEE Trans. SMC, Part B: Cybernetics, 32*, 2002, 13–20.

[4] Z. Bubnicki, Uncertain variables and their application to decision making, *IEEE Trans. SMC A: Systems and Humans, 31*(6), 2001, 587–596.

[5] Z. Bubnicki, *Uncertain logics, variables and systems* (London, Berlin, N. York: Springer-Verlag, 2002).

[6] Z. Bubnicki, *Analysis and decision making in uncertain systems* (London, Berlin, N. York: Springer-Verlag, 2004).

[7] Z. Bubnicki, Knowledge validation and updating in a class of uncertain distributed knowledge systems, *Proc. of 16th IFIP World Computer Congress. Intelligent Information Processing*, Beijing, China, 2000, 516–523.

[8] Z. Bubnicki, Application of uncertain variables to decision making in a class of distributed computer systems, *Proc. of 17th IFIP World Computer Congress. Intelligent Information Processing*, Montreal, Canada, 2002, 261–264.

[9] Z. Bubnicki, Quality of an operation system control based on uncertain variables, *Modelling, Identification and Control*, M. H. Hamza, ed., (Zurich: Acta Press, 2004) 148–153.

[10] Z. Bubnicki, Learning process in a class of computer integrated manufacturing systems with parametric uncertainties, *Journal of Intelligent Manufacturing, 13*(6), 2002, 409–415.

[11] S. Tongsima et al., Probabilistic loop scheduling for applications with uncertain execution time, *IEEE Trans. Computers, 49*(1), 2000, 65-79.

[12] Z. Bubnicki, Uncertain variables and learning process in an intelligent transportation system with production units, *Proc. of 5th IFAC/EURON Symposium on Intelligent Autonomous Vehicles IAV2004*, Lisbon, Portugal, 2004.

[13] R. Toledo, A study of a simple transportation problem applying uncertain variables, *Proc. of 15th International Conference on Systems Science*, Wroclaw, Poland, 2004, Vol. II, 339–345.

[14] Z. Bubnicki, Application of uncertain variables in a class of control systems with uncertain and random parameters, *Proc. of the European Control Conference* Cambridge, UK, 2003.