# The Granular Computing in Uncertain Optimization Problems

Piotr Orantek[1] and Tadeusz Burczyński[1,2]

[1] Silesian University of Technology, Department for Strength of Materials and Computational Mechanics, Konarskiego 18A, 44-100 Gliwice, Poland, e-mail: Piotr.Orantek@polsl.pl
[2] Cracow Univerity of Technology, Artificial Intelligence Department, Warszawska 24, 31-455 Cracow, Poland, e-mail: Tadeusz.Burczynski@polsl.pl

**Abstract.** The paper is devoted to applications of evolutionary algorithms in identification of structures being under the uncertain conditions. Uncertainties can occur in boundary conditions, in material or geometrical parameters of structures and are modelled by three kinds of granularity: interval mathematics, fuzzy sets and theory of probability. In order to formulate the optimization problem for such a class of problems by means of evolutionary algorithms the chromosomes are considered as interval, fuzzy and random vectors whose genes are represented by: (i) interval numbers, (ii) fuzzy numbers and (iii) random variables, respectively. Description of evolutionary algorithms with granular representation of data is presented in this paper. Various concepts of evolutionary operator such as a crossover and a mutation and methods of selections are described. In order to evaluate the fitness functions the interval, fuzzy and stochastic finite element methods are applied. Several numerical tests and examples of identification of uncertain parameters are presented.

## 1 Introduction

In the majority engineering cases it is not possible to determine exactly all parameters of the physical systems. It is necessary to introduce some uncertain parameters which describe the granular character of data. Representation of uncertain values may have different forms. It depends of the physical meaning of the considered problem and the assumed model of uncertainty. There are several formal frameworks in which information granules can be built [7] among them interval analysis [19], fuzzy sets [21] and random variables [3] can be considered.
The aim of an optimization problem is to find some unknown parameters of a fitness function. In this paper the parameters of the function have the granular character. The fitness function is also the granular value.
The evolutionary algorithms [1], as the global optimization technique for searching uncertain values, can be applied in finding the interval parameter [7], fuzzy models [8], fuzzy controllers [11], fuzzy rules [2], random parameters and others. In such algorithms, the chromosome consists of uncertain genes. Therefore, the evolutionary operators are modified for uncertain types of data.
This paper describes a new conception of application of the granular evolutionary algorithm in optimization problems with uncertain parameters. The following systems are considered as the granular models (i) interval numbers, (ii) fuzzy numbers and (iii) random variables. The

proposed granular evolutionary algorithm is examined for a testing bench-mark, due to the optimal parameters of the algorithms (population size, probability of mutation and crossover) are found.

## 2 Granular Evolutionary Algorithm

The paper concerns the granular evolutionary algorithm with granular operators and granular representation of the data. The chromosomes contain granular genes. Each gene decides about the heredity of one or a few characteristics. The individuals can be modified by means of the granular operators. The evolutionary operators generate new chromosomes. The next step is the operator of the selection. It creates a new generation, which contains better chromosomes. All steps are repeated until the stop condition is fulfilled (Fig. 1).

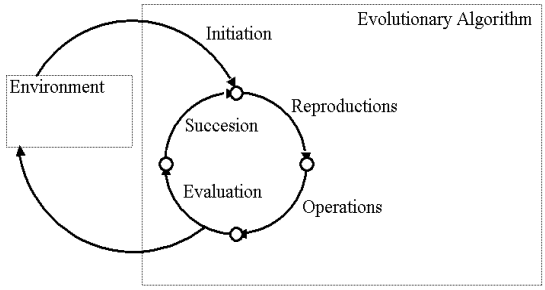


**Figure 1.** The flow chart of the granular evolutionary algorithm.

In the granular evolutionary algorithm an individual expresses a granular solution. In each generation the granular evolutionary algorithm contains a population of solutions. Each solution is evaluated, and as the result a granular value of the fitness function is obtained.

### 2.1 The granular representation of chromosomes

#### 2.1.1 The interval chromosome

In most cases the evolutionary algorithm has the genes as the real values. The granular algorithm works on the granular data, so the gene should be modified to granular data. In the paper the following cases are considered: (i) interval genes, (ii) fuzzy genes and (iii) random genes.
In the interval case the gen $[x]=[\underline{x},\overline{x}]$ is described by the central value $cv([x])=(\underline{x}+\overline{x})/2$ and the radius $r([x])=(\underline{x}-\overline{x})/2$.
Therefore the interval chromosome expressed by:

$$\left[[x_1],[x_2],...,[x_i],...,[x_n]\right] \tag{1}$$

can be replaced by the real-coded chromosome:

$$\left[(cv_1,r_1),(cv_2,r_2),...,(cv_i,r_i),...,(cv_n,r_n)\right] \tag{2}$$

where: $[x_i]=(cv_i,r_i)$.

### 2.1.2 The fuzzy chromosome

In the fuzzy case the gene $x$ can be considered as a fuzzy set. The fuzzy set is considered as a set of pairs of the $x$ and the density function $\mu(x)$. When the fuzzy set is convex and normal and the density function is continuos, the fuzzy set is the fuzzy number. The concept of α–cuts plays the important role in the theory of fuzzy sets. An α-cut of a fuzzy number $A$ is a interval that contains all the numbers of $A$ that have the membership value of $A$ greater than or equal to α. In this case the fuzzy number can be replaced by a set of the interval values, which are stretched on the adequate levels (α–cuts) of the fuzzy value. This approach has some advantages. For each α–cut the very good known interval arithmetic operators are used. It is possible to obtain different forms of the fuzzy values due to the generation of a few α–cuts and corresponding them interval values $[\underline{x}; \overline{x}]$. The forms can be symmetric or not symmetric. They describe some characteristic forms of the fuzzy values, and permit to build a new form of the fuzzy value too. Finally, each gene $x$ is expressed as the real value: the *central value* $cv(x)$ and a set of parameters $a^i(x)$ and $b^i(x)$, ($i$=1,…,$M$, where $M$ is a number of α–cuts) which define distances between $cv(x)$ and the boundaries of the intervals. It is possible to introduce the constraints on the $cv(x)$ and non-symmetric constraints on the widths of the intervals using the parameters $a^i(x)$ and $b^i(x)$.
Therefore, the fuzzy chromosome expressed by:

$$\left[\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_i, ..., \mathbf{x}_n\right] \tag{3}$$

can be replaced by the real-coded chromosome (for $M$=2):

$$\left[\left(a_1^1, a_1^2, cv_1, b_1^2, b_1^1\right), \left(a_2^1, a_2^2, cv_2, b_2^2, b_2^1\right), ..., \left(a_i^1, a_i^2, cv_i, b_i^2, b_i^1\right), ..., \left(a_n^1, a_n^2, cv_n, b_n^2, b_n^1\right)\right] \tag{4}$$

where: $\mathbf{x}_i = \left(a_i^1, a_i^2, cv_i, b_i^2, b_i^1\right)$.

### 2.1.3 The stochastic chromosome

In the theoretical model of random phenomena the basic role is played by the probability space ($\boldsymbol{\Gamma}$, $\mathcal{F}$, $P$). The set $\boldsymbol{\Gamma}$, called the *space of elementary events* represents all the possible simplest outcomes of a trial associated with a given random phenomenon. $\mathcal{F}$ is a $\sigma$ - algebra of subset of $\boldsymbol{\Gamma}$. The elements of the $\mathcal{F}$ are called *random events*. $P$ is a *probability* defined on $\mathcal{F}$ [20].
In the random case the gene is represented by a random variable, which is a real function $X_i = X_i(\gamma)$, $\gamma \in \boldsymbol{\Gamma}$, defined on a sample space $\boldsymbol{\Gamma}$ and measurable with respect to $P$: i.e., for every real number $x_i$, the set $\{\gamma : X_i(\gamma) < x_i\}$ is an event in $\mathcal{F}$. The chromosome $\mathbf{X}(\gamma)$ is a function (measurable respect to $P$) which takes every element $\gamma \in \boldsymbol{\Gamma}$ into a point $\mathbf{x} \in R^n$ [14].
The chromosome is expressed as random vector:

$$\mathbf{X}(\gamma) = [X_1(\gamma), X_2(\gamma), ..., X_i(\gamma), ..., X_n(\gamma)] \tag{5}$$

which has an $n$-dimensional Gaussian distribution of the probability density function, given as follows:

$$p(x_1, x_2, ..., x_i, ..., x_n) = \frac{1}{(2\pi)^{n/2}\sqrt{|\mathbf{K}|}}\left[-\frac{1}{|\mathbf{K}|}\sum_{i,j=1}^{n}\left|K_{ij}\right|(x_i - m_i)(x_j - m_j)\right] \tag{6}$$

where $|\mathbf{K}| \neq 0$ is the determinant of the matrix covariances, $\mathbf{K} = [k_{ij}]$, $i, j = 1,2,\ldots,n$, where $k_{ij} = \mathbf{E}[(X_i - m_i)(X_j - m_j)]$, $|K_{ij}|$ is the co-factor of the element $k_{ij}$ the matrix $\mathbf{K}$ and $m_i = \mathbf{E}[X_i(\gamma)]$ is the mean value of $X_i(\gamma)$.
It is assumed that random genes are independent random variables. The joint probability density function is expressed by the probability density functions of single random genes as follows:

$$p(x_1, x_2, ..., x_i, ..., x_n) = p_1(x_1)p_2(x_2)....p_i(x_i)....p_n(x_n) \tag{7}$$

where:

$$p_i(x_i) = N(m_i, \sigma_i) = \frac{1}{\sigma_i\sqrt{2\pi}} \exp\left[-\frac{(x_i - m_i)^2}{2\sigma_i^2}\right] \tag{8}$$

is the probability density function of the random gene $X_i(\gamma)$, where $\sigma_i$ denotes the standard deviation of $X_i(\gamma)$.
It can be seen that if the random genes $X_i(\gamma)$, i=1,2,…,*n*, are random independent Gaussian variables, two moments describe the probability density function of the random variable $X_i(\gamma)$.
The stochastic chromosome (5) can be repleaced by:

$$[(m_1, \sigma_1), (m_2, \sigma_2), ..., (m_i, \sigma_i), ..., (m_n, \sigma_n)] \tag{9}$$

where: $m_i$ - the mean value and $\sigma_i$ - standard deviation.

## 2.1 The granular mutation

In the interval case two types of the mutation operators are applied. In both cases the modified gene $x_j$ is randomly selected from the chromosome $x$=[ $x_1$, $x_2$, ..., $x_j$ , ..., $x_n$].
In the first type of the mutation (mutation I) the central value $cv(x_j)$ of the *j*-th interval value $x_j$ is modified. The operator is expressed by the following equation:

$$h(x_j^*) = h(x_j) + G_h \tag{10}$$

where: $h$=$cv$ for each gene, $G_h$ – random value (with Gaussian distribution), $j$=1..$n$ is the number of the gene.
The second type of the mutation operators (mutation II) concentrates/deconcentrates the interval value. The mutation changes the radius $r(x_j)$ according to the equation (10), where $h$=$r$ for each gene. Therefore, two types of the mutation operator are introduced, both can work together or independently.
In the fuzzy case two types of the mutation operators are also applied. In both cases the modified gene $x_j$ is randomly selected from the chromosome $x$=[ $x_1$, $x_2$, ..., $x_j$ , ..., $x_n$].
In the first type of the mutation (mutation I) the central value $cv(x_j)$ of the *j*-th fuzzy value $x_j$ is modified. The operator is expressed by the equation (10), where $h$=$cv$.
The second type of the mutation operators (mutation II) concentrates/deconcentrates the fuzzy value $x_j$. The mutation changes the distances $a_i(x_j)$ or $b_i(x_j)$ by equation (10) where: $h$= $a_i$, $b_i$.
This operator is considered as symmetric ($a_i(x^j)$ and $b_i(x^j)$ are changed by means of the same value), and non-symmetric ones. The operator can change only the selected α–cut. Therefore, two types of the mutation operator is introduced, both can work together or independently.

In the random variables case, two types of the mutation operators are also applied. In both cases the modified gene $x_j$ is randomly selected from the chromosome $x$=[ $x_1$, $x_2$, ..., $x_j$ , ..., $x_n$]. The first type changes the fisrt normal moment $m$ using formula (10), where: $h=m$. The second type changes the standard deviation of the $j$-th random variable using formula (10), where: $h=\sigma$.

## 2.2 The granular crossover

The granular arithmetic crossover operator is proposed in the granular evolutionary algorithm. The crossover creates two offspring individuals $x^* = \left[ x_1^*, x_2^*, ..., x_j^*, ..., x_n^* \right]$ and $y^* = \left[ y_1^*, y_2^*, ..., y_j^*, ..., y_n^* \right]$ on the basis of two parent chromosomes $x = \left[ x_1, x_2, ..., x_j, ..., x_n \right]$ and $y = \left[ y_1, y_2, ..., y_j, ..., y_n \right]$.
The selected parameters of the $j$-th genes of the offspring chromosomes are expressed by the following equations (interval cases):

$$h(x_j) = \lambda h(x_j) + (1-\lambda) h(y_j) \tag{11}$$

$$h(y_j^*) = \lambda h(y_j^*) + (1-\lambda) h(x_j^*) \tag{12}$$

where: $h=cv, r,$ and $\lambda \in [0,1]$ is a random value with the uniform distribution.
In the fuzzy case the selected parameters of the $j$-th genes of the offspring chromosomes are expressed by the equations (11 and 12), where $h=cv,a_i,b_i$.
In the random variables case the offspring chromosomes are expressed by equations (11) and (12), where $h=m,\sigma$.

## 2.3 The granular selection

The last modified operator for the interval, fuzzy values and random variables is the selection operator. This operator is constructed on the basis of a well known tournament selection. In this selection the fitness function values $f$ are compared, and the better chromosome wins more often. Therefore the special strategy of comparison of two granular values $f_1$ and $f_2$ is proposed.
In interval and fuzzy cases the special conditions are constructed:

$$h_{f_1} < h_{f_2} \tag{13}$$

where: $h=cv,r$.
In the fuzzy case the condition (13) is checked, where: $h=a_1$, $b_1$ ,$r$. In the stochastic case the condition (13) is checked, where: $h=m$, $\sigma$ .

## 2.4 The granular fitness function

One of the most important steps of the evolutionary algorithm is the evaluation of the fitness function. If the design variables are deterministic, the fitness function result is also deterministic. In the case of solving the granular optimization problems, the problem of evaluating the fitness function is much more complicated. A few ways to estimate the results are possible in this case. In the case of simple mathematical functions the basic arithmetic operators {+;-;*;/} for granular representation are used.

Unfortunately, in many cases the fitness function can be examinated after solving the interval/fuzzy/stochastic boundary-value problem. The boundary-value problems can be solved by means of the interval/fuzzy/stochastic boundary element method or the interval/fuzzy/stochastic finite element method.

## 3 Testing the granular evolutionary algorithm

The aim of the test is to find the granular vector **x** which minimizes the function:

$$f = f(\mathbf{x}) = \sum_{i=1}^{n}\left[\frac{1}{\pi}\left|x_i - 0.7\right|^{\frac{\pi}{3}}\left(\frac{\pi}{2} - \cos\left(2\pi p\left|x_i - 0.7\right|^{\frac{\pi}{3}}\right)\right)\right] \tag{14}$$

where: n – the number of granular design decision variables $x_i$ , p – the number of the optimum.
In the first step of examination, the best (optimal) probabilities of mutation (*pm*) and crossover (*pc*) operators were searched. In the second stage the best population size (*ps*) was searched. For each combination (*n*=1..5, *p*=1..5) the 10000 independend experiments were run. The optimal probabilities and population size of granular evolutionary algorithms (interval, fuzzy and stochastic) are included in Tables 1-4.

**Table 1.** The optimal probabilities *pm*, *pc* and population size *ps* of the granular evolutionary algorithm (interval case)

| *n* | *p* | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | | | 2 | | | 3 | | | 4 | | | 5 | | |
| | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* |
| 1 | 0.4 | 0.2 | 4 | 0.4 | 0.2 | 3 | 0.3 | 0.1 | 3 | 0.3 | 0.1 | 4 | 0.3 | 0.1 | 3 |
| 2 | 0.4 | 0.1 | 4 | 0.4 | 0.1 | 4 | 0.4 | 0.1 | 4 | 0.3 | 0.1 | 4 | 0.3 | 0.1 | 4 |
| 3 | 0.3 | 0.1 | 4 | 0.3 | 0.1 | 5 | 0.3 | 0.1 | 4 | 0.2 | 0.1 | 4 | 0.2 | 0.1 | 5 |
| 4 | 0.3 | 0.1 | 5 | 0.2 | 0.1 | 4 | 0.2 | 0.1 | 4 | 0.2 | 0.1 | 4 | 0.2 | 0.1 | 4 |
| 5 | 0.3 | 0.1 | 5 | 0.2 | 0.1 | 5 | 0.2 | 0.1 | 4 | 0.3 | 0.1 | 4 | 0.2 | 0.1 | 5 |

**Table 2.** The optimal probabilities *pm*, *pc* and population size *ps* of the granular evolutionary algorithm (fuzzy case, 2 alfa-cuts)

| *n* | *p* | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | | | 2 | | | 3 | | | 4 | | | 5 | | |
| | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* |
| 1 | 0.4 | 0.1 | 5 | 0.4 | 0.1 | 4 | 0.3 | 0.1 | 5 | 0.3 | 0.1 | 5 | 0.3 | 0.1 | 6 |
| 2 | 0.4 | 0.1 | 4 | 0.4 | 0.1 | 5 | 0.4 | 0.1 | 5 | 0.4 | 0.1 | 7 | 0.4 | 0.1 | 8 |
| 3 | 0.5 | 0.2 | 11 | 0.5 | 0.2 | 15 | 0.5 | 0.2 | 15 | 0.5 | 0.2 | 18 | 0.5 | 0.2 | 19 |
| 4 | 0.5 | 0.2 | 10 | 0.5 | 0.2 | 15 | 0.5 | 0.2 | 18 | 0.5 | 0.2 | 18 | 0.5 | 0.2 | 20 |
| 5 | 0.5 | 0.2 | 11 | 0.5 | 0.2 | 16 | 0.5 | 0.2 | 19 | 0.5 | 0.2 | 21 | 0.5 | 0.2 | 23 |

**Table 3.** The optimal probabilities *pm*, *pc* and population size *ps* of the granular evolutionary algorithm (fuzzy case, 3 alfa-cuts)

| *n* | *p* | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | | | 2 | | | 3 | | | 4 | | | 5 | | |
| | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* |
| 1 | 0.4 | 0.1 | 4 | 0.3 | 0.1 | 6 | 0.3 | 0.1 | 8 | 0.3 | 0.1 | 9 | 0.3 | 0.1 | 11 |
| 2 | 0.4 | 0.1 | 4 | 0.4 | 0.1 | 6 | 0.4 | 0.1 | 6 | 0.4 | 0.1 | 9 | 0.4 | 0.1 | 11 |
| 3 | 0.4 | 0.2 | 8 | 0.4 | 0.2 | 14 | 0.4 | 0.2 | 13 | 0.4 | 0.2 | 13 | 0.4 | 0.2 | 15 |
| 4 | 0.5 | 0.2 | 10 | 0.5 | 0.2 | 19 | 0.4 | 0.2 | 17 | 0.4 | 0.2 | 21 | 0.5 | 0.2 | 22 |
| 5 | 0.5 | 0.2 | 11 | 0.5 | 0.2 | 16 | 0.5 | 0.2 | 19 | 0.5 | 0.2 | 24 | 0.5 | 0.2 | 27 |

**Table 4.** The optimal probabilities *pm*, *pc* and population size *ps* of the granular evolutionary algorithm (stochastic case)

| *n* | *p* | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | | | 2 | | | 3 | | | 4 | | | 5 | | |
| | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* | *pm* | *pc* | *ps* |
| 1 | 0.4 | 0.1 | 4 | 0.4 | 0.1 | 5 | 0.3 | 0.1 | 4 | 0.3 | 0.1 | 9 | 0.3 | 0.1 | 11 |
| 2 | 0.3 | 0.1 | 4 | 0.3 | 0.1 | 6 | 0.3 | 0.1 | 9 | 0.4 | 0.1 | 9 | 0.4 | 0.1 | 11 |
| 3 | 0.2 | 0.1 | 4 | 0.2 | 0.1 | 5 | 0.2 | 0.1 | 5 | 0.4 | 0.2 | 13 | 0.4 | 0.2 | 15 |
| 4 | 0.2 | 0.1 | 4 | 0.2 | 0.1 | 5 | 0.2 | 0.1 | 5 | 0.4 | 0.2 | 21 | 0.5 | 0.2 | 22 |
| 5 | 0.2 | 0.1 | 4 | 0.2 | 0.1 | 5 | 0.2 | 0.1 | 4 | 0.5 | 0.2 | 24 | 0.5 | 0.2 | 27 |

## 4 Conclusions

An effective intelligent technique based on the granular evolutionary algorithm has been presented. This approach can be applied in the optimization and the identification of systems that are in the uncertain conditions. This approach is very promising for reliability optimization in which the safety of a system is estimated and represented by the probability of its failure, i.e. the occurrence of an ultimate limit state manifesting itself.

The more general approaches will be developed in future. The other types of probability density function (PDF) with a greater number of moments will be examined. The dependences between random numbers will be also examined.

The presented algorithm for uncertain identification problems in mechanical structures was used. The results will be presented on the conference.

In the general case uncertain conditions have the granular form [3]. The models based on the interval and fuzzy numbers were used instead of stochastic approach presented in this paper. The models based on the perturbation numbers will be presented in the future.

## Acknowledgement

The research is financed from the budget resources as the research project PBU-68/RMT-4/2007.

## Bibliography


[1] Arabas, J., *Wykłady z algorytmów ewolucyjnych*. WNT, 2001.
[2] Arslan, A., Kaya, M., Determination of fuzzy logic membership functions using genetic algorithms. *Fuzzy Sets and Systems* 118 (2001) Elsevier 2001.
[3] Bargiela, A., Pedrycz, W., *Granular Computing: An introduction*. Kluwer Academic Publishers Boston/Dordrecht/London 2002.
[4] Bui, H.D., *Inverse Problems in the Mechanics of Materials: An Instroduction.* CRC Pres, Bocca Raton 1994.
[5] Burczyński, T., Orantek, P., *Application of neural networks in controlling of evolutionary algorithms*. First Asian-Pacific Congress on Computational Mechanics APCOM 01. Sydney, Australia 2001.
[6] Burczyński, T., Skrzypczyk, J., Fuzzy aspects of the boundary element method. *Engineering Analysis with Boundary Elements* 19, 1997, pp. 209-216.
[7] Chen, L., Rao, S.S., Fuzzy finite element approach for vibrating analysis of imprecisely defined systems. *Finite Elements in Analysis and Design*. 1977, vol. 27, pp. 69-83.
[8] Cordon, O., Gomide, F., Herrera, F., Homann, F., Magdalena, I., Ten years of genetic fuzzy systems: current framework and new trends. *Fuzzy Sets and Systems* 141 (2004).
[9] Czogała, E., Pedrycz, W., *Elementy i metody teorii zbiorów rozmytych.* PWN, Warszawa 1985.
[10] Kacprzyk, J., *Zbiory rozmyte w analizie systemowej*. PWN Warszawa 1986.
[11] Karr, C.L., *Design of an adaptive fuzzy logic controller using a genetic algorithm*, Proc. 4th Int. Conf. on Genetic Algorithms, San Diego, July 13-16, pp. 450-457.
[12] Kleiber, M.(ed), 1998, *Handbook of Computational Solid Mechanics*, Springer-Verlag, Berlin 1998
[13] Orantek, P., *The optimization and identification problems of structures with fuzzy parameters.* Proc. of 3rd European conference on computational mechanics ECCM-2006, Lizbona, Portugal 2006.
[14] Papoulis, A., *Probability, Random Variables, and Stochastic Processes*. McGraw Hill, New York 1991
[15] Pedrycz, W., Fuzzy evolutionary computing. *Soft Computing 2* (1998), Springer-Verlag 1998.
[16] Piegat, A., *Modelowanie i sterowanie rozmyte*. Akademicka Oficyna Wydawnicza EXIT, Warszawa 2003.
[17] Rutkowska, D., Piliński, M., Rutkowski, L., *Sieci neuronowe, algorytmy genetyczne i systemy rozmyte*. PWN Warszawa-Łódź 1997.
[18] Schaefer, R., *Podstawy genetycznej optymalizacji globalnej*. Wydawnictwo Uniwersytetu Jagiellońskiego, Kraków, 2002.
[19] Skalna, I., *Zastosowanie metod algebry przedziałowej w jakościowej analizie układów mechanicznych*. Praca doktorska, Politechnika Śląska, Gliwice 2002.
[20] Sobczyk, K*., Stochastic wave propagation*. PWN Warszawa 1984.
[21] Zadeh, L.A., Fuzzy sets, *Information and Control*, Vol.8, 1965.