Paper 277

Proceedings of the Eleventh International Conference
on Computational Structures Technology,
B.H.V. Topping, (Editor),
Civil-Comp Press, Stirlingshire, Scotland

A Differential Evolution Algorithm for Fuzzy Control of Smart Structures

M. Marinaki, Y. Marinakis and G.E. Stavroulakis
Department of Production Engineering and Management
Technical University of Crete, Chania, Greece

Abstract

Smart structures include elements of active, passive or hybrid control. For complicated structures, mainly the ones including nonlinearities, or nonlinear control laws, the theoretical results from the area of control are not very helpful. In this paper, the fuzzy control is considered, which is a suitable tool for the systematic development of active control strategies, and the Differential Evolution algorithm is proposed and used for the calculation of the continuous and discrete free parameters in the fuzzy control system. Numerical applications for smart piezoelastic beams are presented. The results obtained are compared with the ones obtained with the fuzzy controller optimized using Particle Swarm Optimization and with the fuzzy controller optimized using a Genetic Algorithm.

Keywords: differential evolution, particle swarm optimization, genetic algorithms, active control of structures, fuzzy control, smart structures.

1 Introduction

During the last years, there has been an increasing application of nature inspired approaches and evolutionary approaches to many fields, and, especially when the task is optimization within complex domains of data or information. Nature inspired approaches represent successful animal and micro-organism team behaviour. Swarm or flocking intelligence inspired Particle Swarm Optimization [1], ants foraging behaviors gave rise to Ant Colony Optimization [2], the mimesis of biological immune systems led to Artificial Immune Systems [3, 4], the simulation of the foraging behaviour of bees led to many approaches like the Artificial Bee Colony (ABC) Algorithm [6, 5], the Virtual Bee Algorithm [7], etc., while the mating behaviour of bees led to the Honey Bees Mating Optimization Algorithm [8, 9]. Evolutionary algorithms

(EAs) [10] are search methods inspired from natural selection and survival of the fittest in the biological world. They simulate the evolution of individual structures via processes of selection and pertubation and have been successfully applied to a variety of optimization problems.

Differential Evolution (DE) is a stochastic, population-based algorithm that was proposed by Storn and Price [11, 12, 13]. Recent books for the DE can be found in [14, 15]. DE has the basic characteristics of the evolutionary algorithms as it is an evolutionary algorithm. It focuses in the distance and the direction information of the other solutions. In the differential evolution algorithms [16], initially, a mutation is applied to generate a trial vector and, afterwards, a crossover operator is used to produce one offspring. The mutation step sizes are not sampled from an a priori known probability distribution function as in other evolutionary algorithms but they are influenced by differences between individuals of the current population.

In this paper, the Differential Evolution is used in order to optimize the parameters of a fuzzy control system that is used for the vibration control problem of a flexible structure (smart beam). The results obtained are compared with the results of the fuzzy control system when its parameters are not optimized by any method, with a fuzzy control system that its parameters are optimized by Particle Swarm Optimization [17] and with a fuzzy control system that its parameters are optimized by a Genetic Algorithm [17].

Beams are fundamental elements in many mechanisms and structures [18, 19, 20, 21, 22, 23]. Therefore, the modeling of the dynamic behavior as well as of the control of beams is a significant problem. A smart structure with bonded sensors and actuators as well as an associated control system, which enable the structure to respond to external excitations in such a way that it suppresses undesired effects, is considered. The choice of the control technique is important for the design of controllers which ensures the performance of the flexible structure under required conditions and at the same time can be easily applied. It should be mentioned here that the proposed method is quite general and can be used for the design of other smart structures like plates, shells, etc. Results in this direction will be reported in the future.

The rest of the paper is organized as follows: In section 2, the modelling of the smart beams is outlined. In section 3, an analytical description of the proposed fuzzy control system optimized by DE is given while in section 4 the numerical results are presented. In the last section, the conclusions and some proposals for further research are given.

2 Models of Smart Beams and Structures

A smart laminated composite beam with rectangular cross-section having length L, width b, and thickness h is considered (Figure 1) where the control actuators (thickness h_A) and the sensors (thickness h_S) are piezoelectric patches symmetrically bonded on the top and the bottom surfaces of the host beam. Both piezoelectric layers are

positioned with identical poling directions and can be used as sensors or actuators.

The mathematical formulation of the model is based on the shear formulation beam theory (Timosenko theory) and the linear theory of piezoelectricity. Furthermore, quasi-static motion is assumed, which means that the mechanical and electrical forces are balanced at any given instant. The numerical solution of the model is based on the development of superconvergent finite elements using the form of exact solution of the Timoshenko beam theory and Hamilon's principle (see [19, 24, 25, 26, 27]).

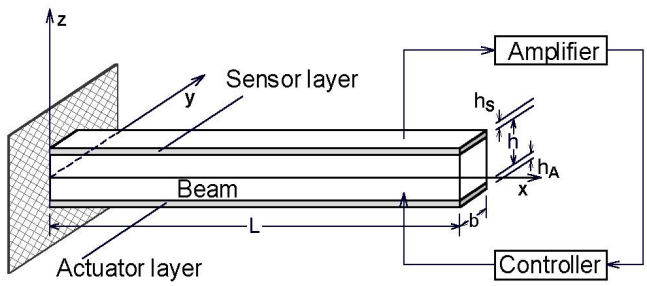


Figure 1: Laminated beam with piezoelectric sensors, actuators and the schematic control system

The linear constitutive equations of the two coupled fields read:

$$\{\sigma\} = [Q]\left(\{\varepsilon\} - [d]^T\{E\}\right) \tag{1}$$

$$\{D\} = [d][Q]\{\varepsilon\} + [\xi]\{E\} \tag{2}$$

where $\{\sigma\}_{6\times1}$ is the stress vector, $\{\varepsilon\}_{6\times1}$ is the strain vector, $\{D\}_{3\times1}$ is the electric displacement, $\{E\}_{3\times1}$ is the strength of applied electric field acting on the surface of the piezoelectric layer, $[Q]_{6\times6}$ is the elastic stiffness matrix, $[d]_{3\times6}$ is the piezoelectric matrix and $[\xi]_{3\times3}$ is the permittivity matrix. Equation (1) describes the inverse piezoelectric effect (which is exploited for the design of the actuator). Equation (2) describes the direct piezoelectric effect (which is used for the sensor). Additional assumptions are used for the construction of the simplified model: (a) Sensor and actuator (S/A) layers are thin compared with the beam thickness. (b) The polarization direction of the S/A is the thickness direction (z axis). (c) The electric field loading of the S/A is uniform uni-axial in the x-direction. (d) Piezoelectric material is homogeneous, transverse isotropic and elastic. Therefore, the set of equations (1) and (2) is reduced as follows

$$\left\{ \begin{array}{c} \sigma_x \\ \tau_{xz} \end{array} \right\} = \left[\begin{array}{cc} Q_{11} & 0 \\ 0 & Q_{55} \end{array} \right] \left(\left\{ \begin{array}{c} \varepsilon_x \\ \gamma_{xz} \end{array} \right\} - \left[\begin{array}{c} d_{31} \\ 0 \end{array} \right] E_z \right) \tag{3}$$

$$D_z = Q_{11} d_{31} \varepsilon_x + \xi_{33} E_z \tag{4}$$

The electric field intensity E_z can be expressed as

$$E_z = \frac{V}{h_A} \tag{5}$$

where V is the applied voltage across the thickness direction of the actuator and h_A is the thickness of the actuator layer.

Since only strains produced by the host beam act on the sensor layer and no electric field is applied to it, the output charge from the sensor can be calculated using Equation (4). The charge measured through the electrodes of the sensor is given by

$$q(t) = \frac{1}{2} \left\{ \left(\int_{S_{ef}} D_z dS \right)_{z=\frac{h}{2}} + \left(\int_{S_{ef}} D_z dS \right)_{z=\frac{h}{2}+h_S} \right\} \tag{6}$$

where S_{ef} is the effective surface of the electrode placed on the sensor layer.

The current on the surface of the sensor is given by

$$i(t) = \frac{dq(t)}{dt} \tag{7}$$

The current is converted into open-circuit sensor voltage output by

$$V^S = G_S i(t) \tag{8}$$

where G_S is the gain of the current amplifier.

Furthermore, it is supposed that the bending-torsion coupling and the axial vibration of the beam centerline are negligible and that the components of the displacement field $\{u\}$ of the beam are based on the Timoshenko beam theory which, in turn, means that the axial displacement is proportional to z and to the rotation $\psi(x,t)$ of the beam cross section about the positive y-axis and that the transverse displacement is equal to the transverse displacement $w(x,t)$ of the point of the centroidal axis $(y = z = 0)$. The strain-displacement relationships read

$$\varepsilon_x = z \frac{\partial \psi}{\partial x}, \varepsilon_{xz} = \psi + \frac{\partial w}{\partial x} \tag{9}$$

The kinetic energy of the beam with the layers can be expressed as

$$T = \frac{1}{2} \int_V \rho \{\dot{u}\}^T \{\dot{u}\} dV = \frac{b}{2} \int_0^L \int_{-\frac{h}{2}-h_A}^{\frac{h}{2}+h_S} \rho [(z\dot{\psi})^2 + \dot{w}^2] dz dx \tag{10}$$

on the assumption that the host beam and the piezoelectric patches have identical densities. The strain (potential) energy is given by

$$U = \frac{1}{2}\int_V \{\varepsilon\}^T\{\sigma\}dV =$$

$$\frac{b}{2}\int_0^L \int_{-\frac{h}{2}-h_A}^{\frac{h}{2}+h_S} \left[Q_{11}\left(z\frac{\partial\psi}{\partial x}\right)^2 + Q_{55}\left(\psi + \frac{\partial w}{\partial x}\right)^2 \right] dzdx \tag{11}$$

If the only loading consists of moments induced by the piezoelectric actuators and since the structure has no bending-twisting couple then the first variation of the work has the form

$$\delta W = b\int_0^L M^A \delta\left(\frac{\partial\psi}{\partial x}\right) dx \tag{12}$$

where δ is the first variation operator and M^A is the moment per unit length induced by the actuator layer and is given by

$$M^A = \int_{-\frac{h}{2}-h_A}^{-\frac{h}{2}} z\sigma_x^A dz = \int_{-\frac{h}{2}-h_A}^{-\frac{h}{2}} zQ_{11}d_{31}E_z^A dz \quad (E_z^A = \frac{V_A}{h_A}) \tag{13}$$

Using Hamilton's principle the equations of motion of the beam are derived.

For the finite element discretization beam finite elements are used, with two degrees of freedom (d.o.fs) at each node: the transversal deflection w_i and the rotation ψ_i. They are gathered to form the degrees of freedom vector $X_i = [w_i \quad \psi_i]$. After assembling the mass and stiffness matrices for all elements, it is obtained the equation of motion in the form

$$M\ddot{X} + \Lambda\dot{X} + KX = F_m + F_e \tag{14}$$

where M and K are the generalized mass and stiffness matrices, F_e is the generalized control force vector produced by electromechanical coupling effects, Λ is the viscous damping matrix and F_m is the external loading vector. It should be mentioned here that technical bending theories for plates can be constructed analogously.

The main objective is to design control laws for the smart beam bonded with piezoelectric S/A subjected to external induced vibrations. For this purpose, a fuzzy control system optimized by DE is applied in this paper. A two-input, single-output fuzzy inference controller is tested in this paper. This configuration is suitable for a feedback control force based on the displacement and velocity at a given point (collocated configuration of the local controller). If needed, several independent and decentralized (local) fuzzy controllers, in general with different characteristics, can be installed at the various points of a larger structure.

3 Fuzzy Control System optimized by Differential Evolution

3.1 Fuzzy Control

Fuzzy systems [28] can be applied in many fields and can solve different kinds of problems in various application domains. Fuzzy inference rules systematize existing experience and can be used for the mathematical formulation of nonlinear controllers. The feedback is based on fuzzy inference and may be nonlinear and complicated. Knowledge or experience on the controlled system is required for the application of this technique. Less knowledge of the logic and availability of more observations (experimental data) requires the use of some hybrid technique in the form of a neuro-fuzzy controller. In the area of smart systems the application of neuro-fuzzy control has been adopted by many authors. In particular, it seems to be suitable for the control of structures with complicated or nonlinear characteristics, like a fuzzy uncertainty modellization adopted for structural engineering in [29], the tuned mass damper systems for aseismic design [30, 31] or the semi-active control using active friction devices or electrorheological fluid dampers [32].

In order to design a fuzzy control system (construct the control rules, state the membership functions, tune the parameters, ...) various approaches have been proposed. In the last years, there is an increasing interest to optimize the fuzzy control systems with heuristic, metaheuristic, and nature inspired approaches. Mainly, genetic algorithms have been used for these purposes while Particle Swarm Optimization and Differential Evolution have not been used so extensively. For example in [33] a method for designing fuzzy logic controllers with symmetric partitioning of the universe of discourse is presented where both the rule base and membership functions of the input and output variables are designed optimally using a genetic algorithm. In [34] a genetic algorithm based optimal fuzzy controller design is proposed where the design procedure is accomplished by establishing an index function as the consequent part of the fuzzy control rule. An efficient genetic reinforcement learning algorithm for designing fuzzy controllers is proposed in [35]. In [36] the proposed algorithm combines the genetic algorithm (GA) and the least-squares estimate (LSE) method to construct the genetic algorithm-based neural fuzzy system for temperature control. An extensive analysis of the genetic fuzzy systems can be found in [37]. [38] presents a particle swarm optimization method for optimizing a fuzzy logic controller for a photovoltaic grid independent system consisting of a PV collector array, a storage battery, and loads (critical and non-critical loads). A learning approach based on Particle Swarm Optimization for the determination of the consequent parameters and premise parameters of a Takagi and Sugeno type fuzzy model is formulated and explained in [39]. [40] introduces the use of the adaptive particle swarm optimization (APSO) for adapting the weights of fuzzy neural networks. In [41] proposes a zero-order Takagi-Sugeno-Kang (TSK)-type fuzzy system learning using a two-phase swarm intelligence algorithm (TPSIA) where the first phase of TPSIA learns fuzzy system structure and parameters

	Displacement				
Velocity	Far_{Up}	$Close_{Up}$	$Equilibrium$	$Close_{Dn}$	Far_{Dn}
Up	Max	Med+	Low-	Null	High-
$Null$	Med+	Low+	Null	Low-	Med-
$Down$	High+	Null	Low+	Med-	Min

Table 1: Fuzzy Inference System Rules, e.g. If displacement is far up and velocity is up then control force is max

by on-line clustering-aided ant colony optimization and phase two aims to further optimize all of the free parameters in the fuzzy system using particle swarm optimization. In [42] a new technique for eliciting a fuzzy inference system (FIS) from data for nonlinear systems is proposed and the parameters of the refined fuzzy model are tuned by means of differential evolution. In [43] a method is presented for the automatic design of a hierarchical fuzzy logic controllers (HFLC) using differential evolution and the feasibility of the method is demonstrated by developing a two-stage HFLC for controlling a cart-pole with four state variables. A hybrid method to train Fuzzy Cognitive Maps, based on a two stage learning approach, first stage the nonlinear Hebbian learning algorithm and second stage the differential evolution algorithm, has been developed, tested and applied in three problems with different complexity levels in [44].

3.2 Fuzzy control system for vibration control of smart structures

In order to reduce the displacement field of the cantilever beam system, a non-linear fuzzy controller [27, 45] was constructed by using the Fuzzy Toolbox of Matlab. More specifically, a Mamdani-type Fuzzy Inference System, consisted of two inputs and one output, was developed. The system receives as inputs the displacement (u) and the velocity ($\dot{u}$), while gives as output the increment of the control force (z). Triangular and trapezoidal shape membership functions were chosen both for inputs and output.

In order to describe the present system-controller 15 rules were used. All rules have weights equal to 1 and use the AND-type logical operator. These rules are presented in the Table 1. The implication method was set to minimum (min), while the aggregation method was set to maximum (max). The defuzzified output value has been created by using the MOM (Mean of Maximum) defuzzification method.

3.3 Structural dynamics and fuzzy control

The Houmbolt numerical integration method was chosen [27, 45] in order to integrate the equations of motion (Equation 14).

According to this method, when acceleration is constant, and the Houmbolt factors are set to

$$\beta = 0.25, \gamma = 0.5 \tag{15}$$

The total integration time was chosen equal to 1 sec, while the time step (Δt) equal to 0.001 sec.

The integration constants are

$$c_1 = \frac{1}{\beta(\Delta t)^2}, c_2 = \frac{1}{\beta \Delta t}, c_3 = \frac{1}{2\beta}, c_4 = \frac{\gamma}{\beta \Delta t}, c_5 = \frac{\gamma}{\beta}, c_6 = \Delta t(\frac{\gamma}{2\beta} - 1) \tag{16}$$

In each step (t) of the numerical integration, the fuzzy controller provides a control force (z), according to the given input values (displacement u and velocity $\dot{u}$). Both the control force and the external loads provide the next step's ($t + \Delta t$) values of displacement and velocity.

3.4 Fuzzy control system optimized by Differential Evolution for the vibration control of smart structures

Some parameters of the fuzzy control system described in section 3.2 were optimized by Differential Evolution. More precisely, the parameters (the break points) of the triangular and trapezoidal membership functions, the weights of the rules, and the logical operator are considered as variables. Firstly, a number of individuals is randomly initialized, where an individual is a solution to the problem. In each individual, the first values correspond to the parameters of the membership functions and can take continuous values, the next 15 values correspond to the weights of the rules and can take continuous values in the range [0,1] while the last 15 values correspond to the AND/OR - type logical operator and can take discrete values equal to 1 for AND and 2 for OR. It should be noted that in DE implementations usually only continuous or discrete values exist. In our case, there is a combination of continuous and discrete values in each individual and in order to solve the problem a combination of the equations used for finding the new solution for the individual in DE for continuous and discrete problems is used.

More precisely, the position of an individual is represented by a d-dimensional vector in problem space $s_i = (s_{i1}, s_{i2}, ..., s_{id})$, $i = 1, 2, ..., N$ (N is the population size, d is set equal to the values parameters plus the values of weights plus the values of the logical type operator).

The Fitness function that has to be minimized is the following error function

$$err = ||X - \bar{X}|| \tag{17}$$

where X is the L^2 norm in R^m (m is the total number of the assumed d.o.fs), $X = [X_1, ..., X_m]$ is the nodal displacements and rotations array and $\bar{X} = [\bar{X_1}, ..., \bar{X_m}]$ is the wished value of the nodal displacements and rotations array (it is set equal to zero).

In Differential Evolution, each individual is randomly placed in the d-dimensional space as a candidate solution. The mutation operator produces a trial vector for each individual of the current population by mutating a target vector with a weighted differential. This trial vector will, then, be used by the crossover operator to produce offspring. For each parent, $s_i(t)$, the trial vector, $u_i(t)$, is generated as follows: a target vector, $s_{i_1}(t)$, is selected from the population, such that $i \neq i_1$. Then, two individuals, s_{i_2} and s_{i_3}, are selected randomly from the population such that i, i_1, i_2 and i_3 are all different. Using these individuals, the trial vector is calculated by perturbing the target vector as follows (in the continuous case)

$$u_i(t) = s_{i_1}(t) + \beta(s_{i_2}(t) - s_{i_3}(t)) \tag{18}$$

where $\beta \in (0, \infty)$ is the scale factor. The upper bound of β is usually the value 1 because as it has been proved if the $\beta > 1$ there is no improvement in the solutions [16, 15] and the most usually utilized value is $\beta = 0.5$.

For the discrete values of the individuals, after the calculation of the target vector, the following equations are used in order to transform the continuous values calculated by Equation 18 into discrete values

$$sig(u_i) = \frac{1}{1 + exp(-u_i)} \tag{19}$$

$$u_i(t) = \begin{cases} 1, & \text{if } rand3 < sig(u_i) \\ 0, & \text{if } rand3 \geq sig(u_i) \end{cases} \tag{20}$$

The base vector s_{i_1} can be determined in a variety of ways and the two most known ways are by selecting a random member of the population or by selecting the best member of the population. In this paper, we use a random member of the population because after a number of tests the choise of a random member produced the best results. The differences vector s_{i_2} and s_{i_3} are selected randomly.

After the completion of the mutation phase of the algorithm, a ***binomial crossover operator*** [16] or ***uniform crossover operator*** [15] is applied. In this crossover operator, the points are selected randomly for the trial vector and for the parent. Initially, a crossover operator number (Cr) is selected [15] that controls the fraction of parameters that are selected from the trial vector. The Cr value is compared with the output of a random number generator, $rand_i(0, 1)$. If the random number is less or equal to the Cr the corresponding value is inherited from the trial vector, otherwise it is selected from the parent

$$s'_i(t) = \begin{cases} u_i(t), & \text{if } rand_i(0,1) \leq Cr \\ s_i(t), & \text{otherwise.} \end{cases} \tag{21}$$

Thus, the choise of the Cr is very significant because if the value is close or equal to 1, then, most of the values in the offspring are inherited from the trial vector (the mutant) but if the value is close to 0, then, the values are inherited from the parent [15].

After the crossover operator, the fitness function of the offspring $s_i'(t)$ is calculated and if it is better than the fitness function of the parent, then, the trial vector is selected for the next generation, otherwise the parent survives for at least one more generation. The algorithm stops after a prespecified number of generations.

4 Numerical Results

The problem of a cantilever beam was studied in the present paper. The beam has a total length equal to 0.8 m and the structure has been discretized with four finite elements resulting in a model with eight degrees of freedom. A dynamic loading is used as disturbance that simulates a strong wind and is a periodic sinusoidal loading pressure ($sin20t$) that influences the displacement of the fourth (free end) finite elements. The purpose of the fuzzy controller is to reduce the oscillation. The controller is collocated, takes as input the displacement and the velocity of the free end and gives back the control force to be applied at the same point. The results obtained with the fuzzy controller optimized by DE are compared with those of the classical fuzzy controller (without any optimization procedure), with the results of fuzzy controller optimized by Particle Swarm Optimization (PSO) [17] and with the results of fuzzy controller optimized by a Genetic Algorithm (GA). The results of the classic fuzzy controller have been compared with the results that arose from classical control (LQR) in [27, 45] giving, with suitably chosen parameters for the fuzzy controller, comparable reduction of the vibrations with the LQR control by using fewer inputs (two instead of sixteen measurements in this example) and less effective reduction of the velocities. The whole algorithmic approach was implemented in Matlab on a Centrino Mobile Intel Pentium M 750 at 1.86 GHz. The parameters of the DE algorithm are selected after thorough testing and are:

- the number of individuals is equal to 15,
- the number of generations is equal to 20,
- the parameter β is set equal to 0.5,
- the crossover rate Cr is set equal to 0.8.

Particle Swarm Optimization (PSO) is a population-based swarm intelligence algorithm that was originally proposed by Kennedy and Eberhart as a simulation of the social behavior of social organisms such as bird flocking and fish schooling [1]. PSO uses the physical movements of the individuals in the swarm and has a flexible and well-balanced mechanism to enhance and adapt to the global and local exploration

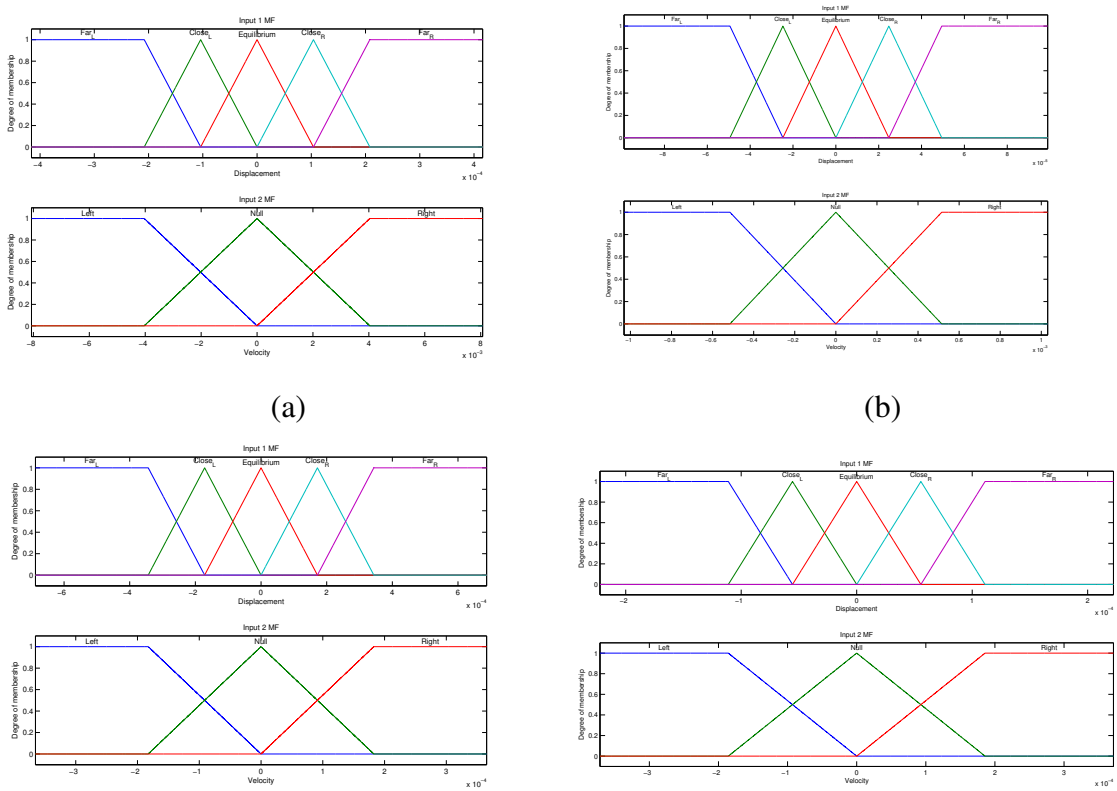
abilities. The wide use of PSO, mainly during the last years, is due to the number of advantages that this method has, compared to other optimization methods. Some of the key advantages are that this optimization method does not need the calculation of derivatives, that the knowledge of good solutions is retained by all particles and that particles in the swarm share information between them. Furthermore, PSO is less sensitive to the nature of the objective function, can be used for stochastic objective functions and can easily escape from local minima. Concerning its implementation, PSO can easily be programmed, has few parameters to regulate and the assessment of the optimum is independent of the initial solution. In Particle Swarm Optimization implementation, the parameters were chosen in such a way that in all algorithms the same number of function evaluations are needed. The results of the Particle Swarm Optimization fuzzy controller are analyzed in more details in [17]. Thus, the parameters are:

- the number of swarms is set equal to 1,
- the number of particles is equal to 15,
- the number of iterations is equal to 20,
- the parameters e_1 and e_2 are both set equal to 2.

Genetic Algorithms (GAs) [46] are search procedures based on the mechanics of natural selection and natural genetics. A GA is a stochastic iterative procedure that maintains the population size constant in each iteration, called a generation. Their basic operation is the mating of two solutions in order to form a new solution. To form a new population, a binary operator called crossover, and a unary operator, called mutation are used [16, 47]. In Genetic Algorithm implementation, the parameters were chosen in such a way that in all algorithms the same number of function evaluations are needed. Thus, the parameters for the GA are:

- the number of individuals is equal to 15,
- the number of generations is equal to 20,
- the mutation rate is set equal to 0.5,
- 1-point crossover was selected with crossover rate equal to 0.8.

The load, the displacement, the displacement velocity, the rotation and the rotation velocity, for the four kinds of control (classic fuzzy control, fuzzy control optimized by DE, fuzzy control optimized by PSO and fuzzy control optimized by GA), are presented in the graphs below. The values without control are shown with a dashed line. The membership functions obtained by the fuzzy controller without any optimization procedure (Figures 2(a), 3(a)), when optimized by DE (Figures 2(b), 3(b)), when optimized by PSO (Figures 2(c), 3(c)) and when optimized by GA (Figures 2(d), 3(d))



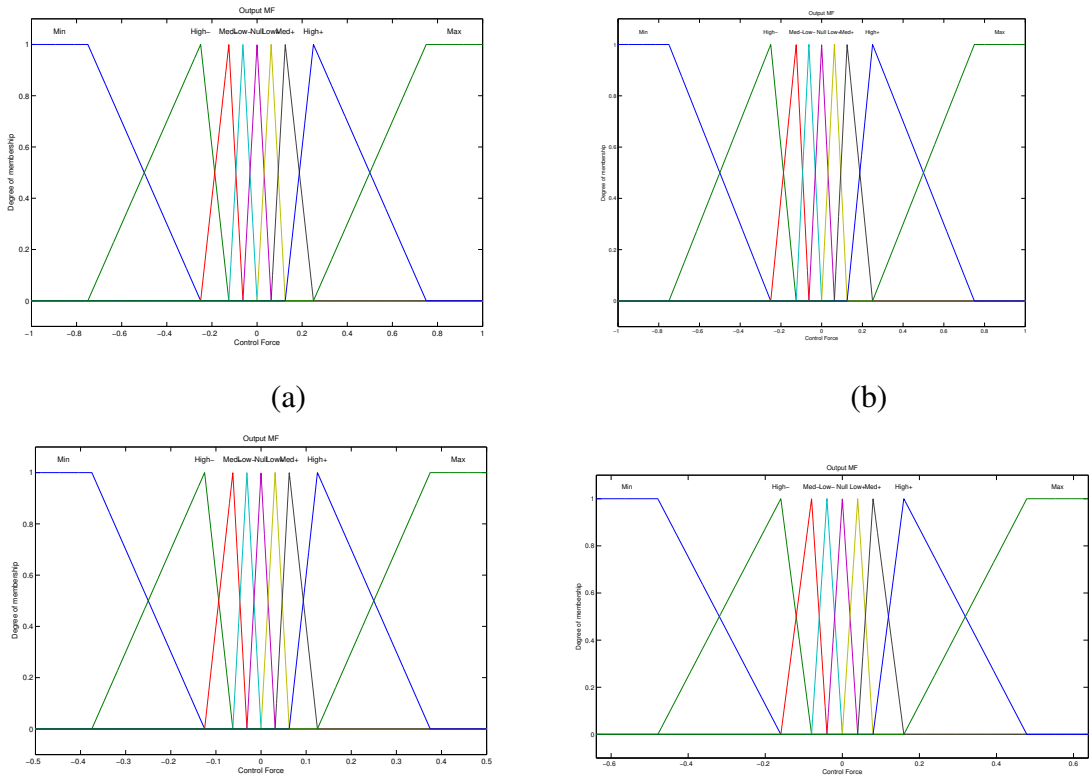
(a) (b)

(c) (d)

Figure 2: Membership functions for the inputs of the classical fuzzy controller (a), the fuzzy controller optimized by DE (b), the fuzzy controller optimized by PSO (c) and the fuzzy controller optimized by GA (d)

are presented. Figure 4 presents the control force at the free end for classic fuzzy controller, for the fuzzy controller optimized by DE, for the fuzzy controller optimized by PSO and for the fuzzy controller optimized by GA.

As it can be seen, the fuzzy controllers optimized by DE, by PSO and by GA give superior control results compared to the classical fuzzy controller. More precisely, the displacements (Figures 5) and the rotations (Figures 7) are almost zero in the case of the fuzzy controller optimized by DE, in the case of fuzzy controller optimized by PSO and in the case of the fuzzy controller optimized by GA and the displacement (Figure 6) and rotation (Figure 8) velocities have significantly smaller values for the fuzzy controller optimized by DE, for the fuzzy controller optimized by PSO and for fuzzy controller optimized by GA compared with the ones obtained by the classical fuzzy controller. It should be noted that the fuzzy controller optimized by DE gives better control results than the fuzzy controller optimized by PSO and the fuzzy controller optimized by GA, as the displacements (Figures 5(b)), the rotations (Figures 7(b)), the displacement velocities (Figure 6(b)) and the rotation velocities (Figure 8(b)) obtained by the fuzzy controller optimized by DE have smaller values compared with the ones



(a) (b)

(c) (d)

Figure 3: Membership functions for the outputs of the classical fuzzy controller (a), the fuzzy controller optimized by DE (b), the fuzzy controller optimized by PSO (c) and the fuzzy controller optimized by GA (d)

of the fuzzy controller optimized by PSO and of the fuzzy controller optimized by GA. Of course, as the differences in the results are small we can say that all methods, DE, PSO and GA, can be applied with very good results in order to optimize the fuzzy controller used for the vibration control of beams.

5 Conclusion

In this paper, a fuzzy control system optimized by Differential Evolution (DE) was used for the vibration control of beams with piezoelectric sensors and actuators. The parameters of the fuzzy controller was selected optimally by using the DE algorithm. The results obtained for a sinusoidal loading pressure using the fuzzy controller system optimized by DE are very efficient. A comparison of the proposed method with a classic fuzzy controller, a fuzzy controller optimized by PSO and a fuzzy controller optimized by GA for this specific problem revealed the high performance of the proposed method. Also, a comparison between them revealed the fact that the fuzzy controller optimized by DE gives superior results compared to the fuzzy controller

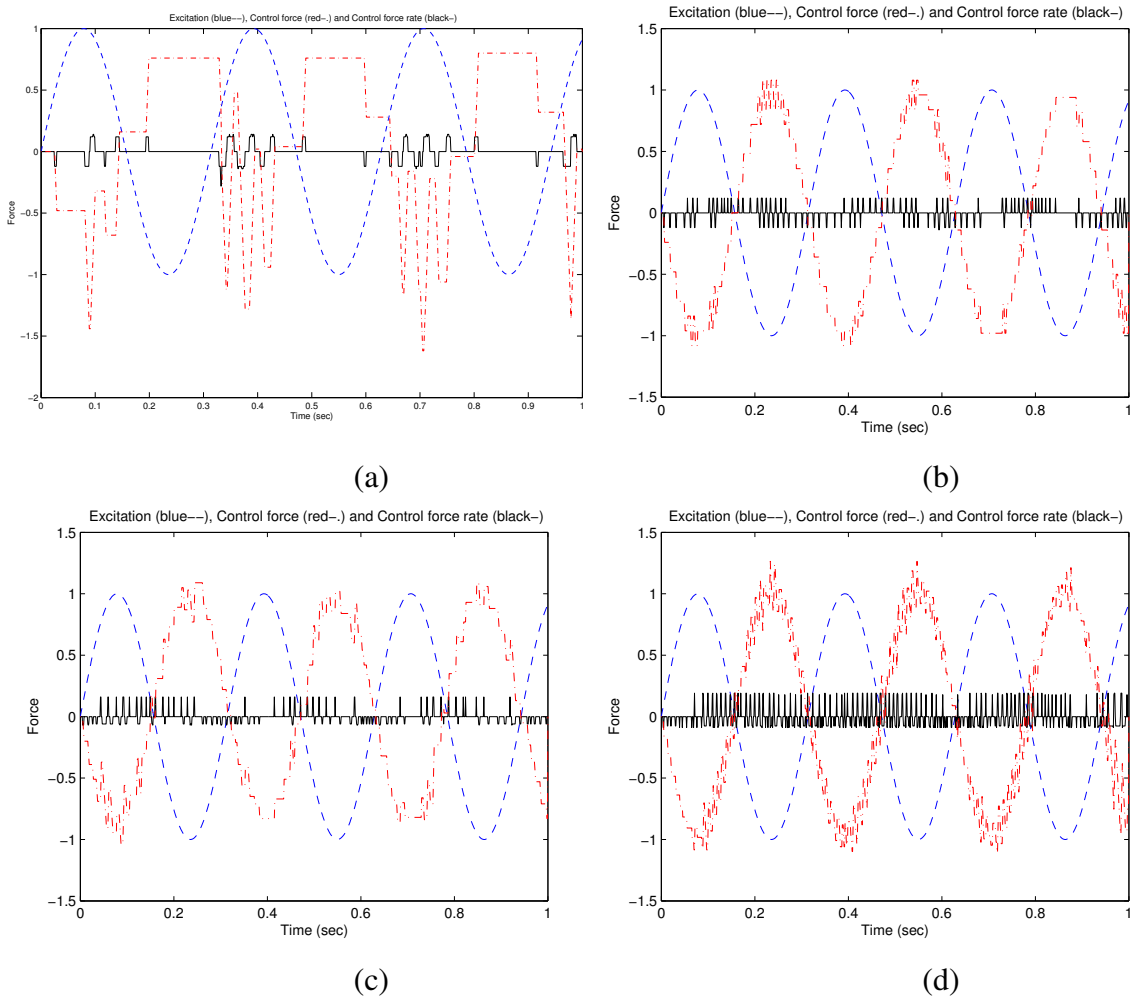


Figure 4: Control force at the free end for classical fuzzy controller (a), for the fuzzy controller optimized by DE (b), for the fuzzy controller optimized by PSO (c) and for the fuzzy controller optimized by GA (d)

optimized by PSO and the fuzzy controller optimized by GA. Future research is intended to be focused in the application of the proposed algorithms for the design of other smart structures like plates, shells and in the modification of the methods in order to be used for optimally design other parts of the fuzzy system like the definition of the control rules.

References

[1] J. Kennedy, R. Eberhart, “Particle Swarm Optimization”, in “Proceedings of 1995 IEEE International Conference on Neural Networks”, 4, 1942-1948, 1995.

[2] M. Dorigo, T. Stutzle, “Ant Colony Optimization”, A Bradford Book, The MIT Press Cambridge, Massachusetts, London, England, 2004.

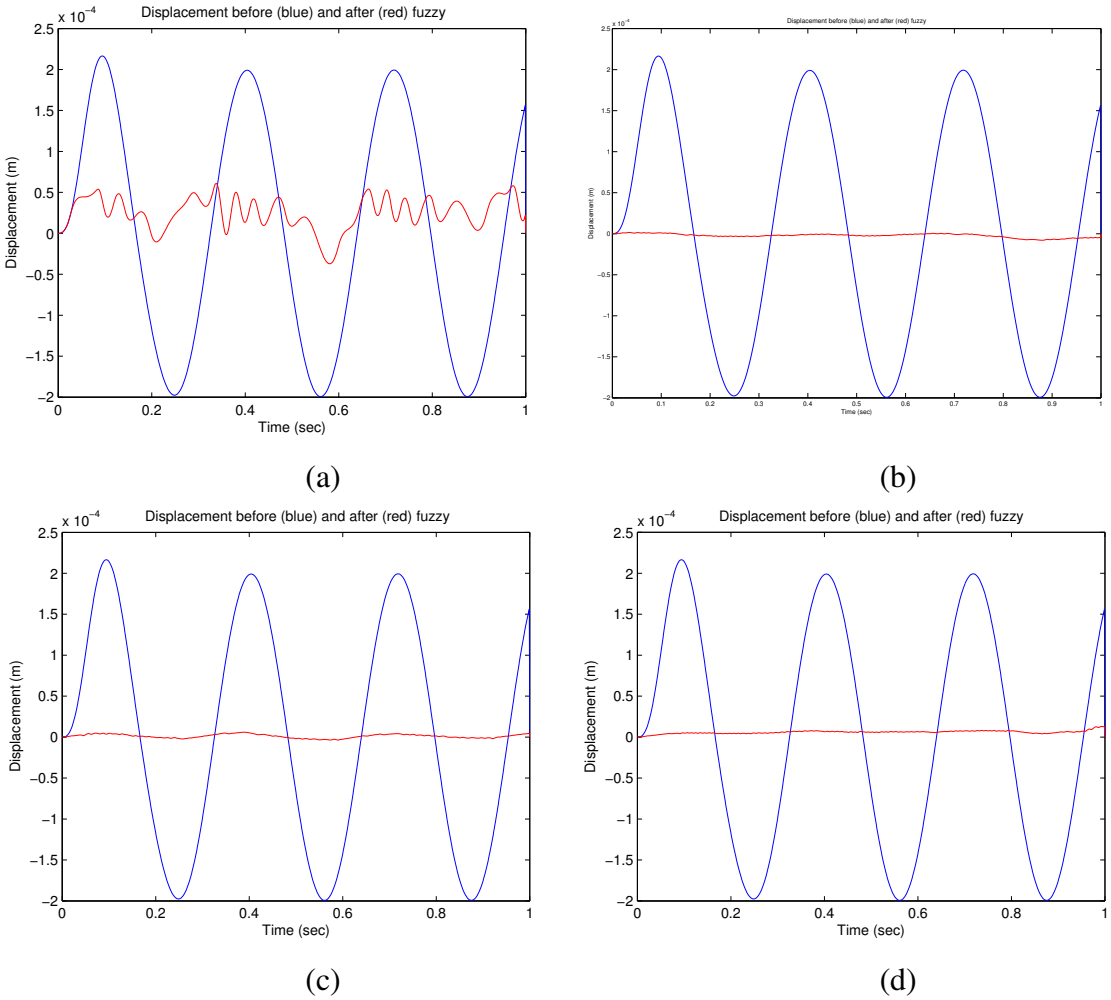


Figure 5: Displacement at the free end for classical fuzzy controller (a), for the fuzzy controller optimized by DE (b), for the fuzzy controller optimized by PSO (c) and for the fuzzy controller optimized by GA (d)

[3] D. Dasgupta (Editor), "Artificial Immune Systems and their Application", Springer, Heidelberg, 1998.

[4] L.D. De Castro, J. Timmis, "Artificial Immune Systems: A New Computational Intelligence Approach", Springer, Heidelberg, 2002.

[5] D. Karaboga, B. Basturk, "On the Performance of Artificial Bee Colony (ABC) Algorithm", Applied Soft Computing, 8, 687-697, 2008.

[6] D. Karaboga, B. Basturk, "A Powerful and Efficient Algorithm for Numerical Function Optimization: Artificial Bee Colony (ABC) Algorithm", Journal of Global Optimization, 39, 459-471, 2007.

[7] X.S. Yang, "Engineering Optimizations via Nature-inspired Virtual Bee Algorithms", in "IWINAC 2005, LNCS 3562", J.M. Yang, J.R. Alvarez, (Editors), Springer-Verlag, Berlin, Heidelberg, 317-323, 2005.

[8] H.A. Abbass, "A Monogenous MBO Approach to Satisfiability", in "Proceeding

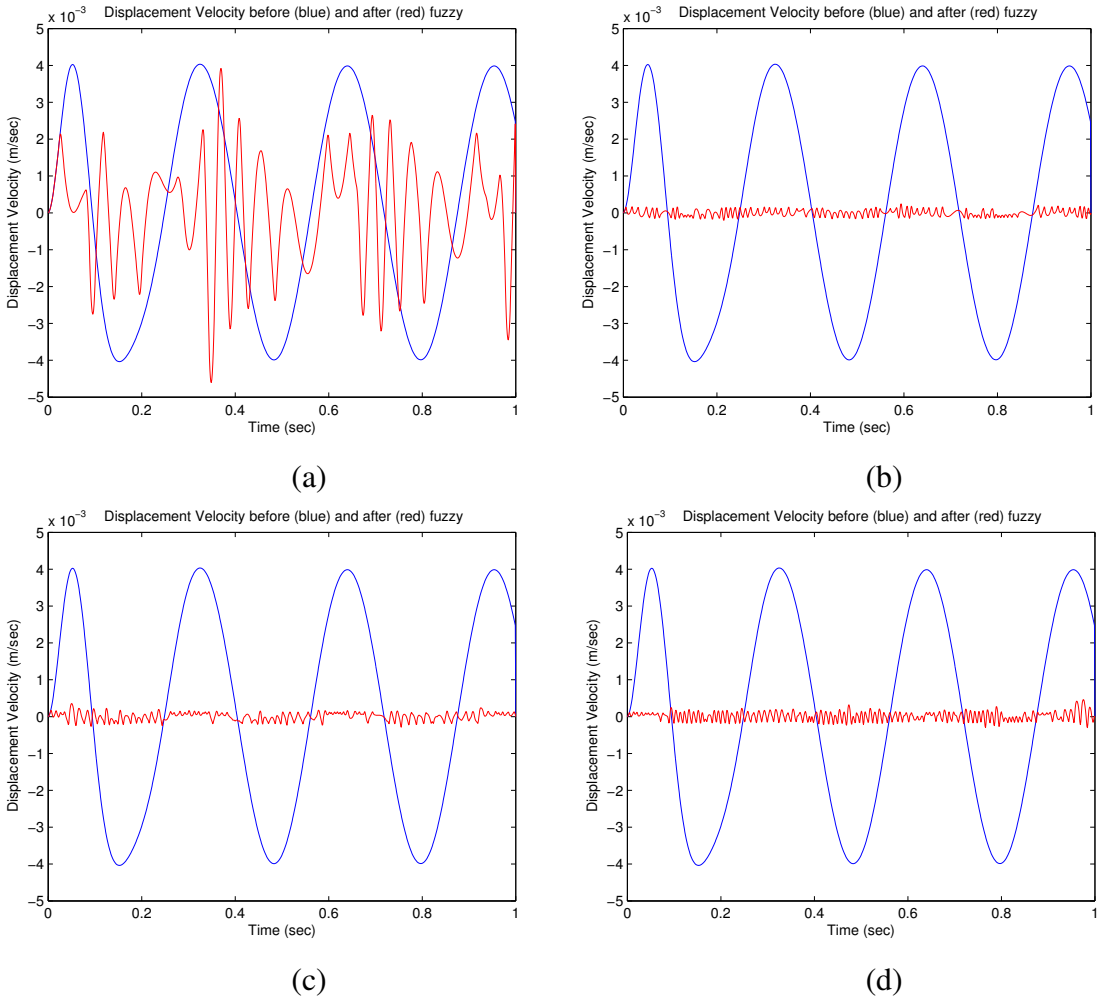


Figure 6: Displacement velocity at the free end for classical fuzzy controller (a), for the fuzzy controller optimized by DE (b), for the fuzzy controller optimized by PSO (c) and for the fuzzy controller optimized by GA (d)

of the International Conference on Computational, Intelligence for Modelling, Control and Automation, CIMCA'2001", Las Vegas, NV, USA, 2001.

[9] H.A. Abbass, "Marriage in Honey-bee Optimization (MBO): A Haplometrosis Polygynous Swarming Approach", The Congress on Evolutionary Computation (CEC2001), Seoul, Korea, May 2001, 207-214, 2001.

[10] W.M. Spears, "Evolutionary Algorithms: The Role of Mutation and Recombination", Natural Computing Series, Springer-Verlag, Berlin, Heidelberg, 2000.

[11] R. Storn, "On the Usage of Differential Evolution for Function Optimization", in "Proceedings of the Biennial Conference of the North American Fuzzy Information Processing Society", 519-523, 1996.

[12] R. Storn, "System Design by Constraint Adaptation and Differential Evolution", IEEE Transactions on Evolutionary Computation, 3(1), 22-34, 1999.

[13] R. Storn, K. Price, "Differential Evolution - A Simple and Efficient Heuristic for

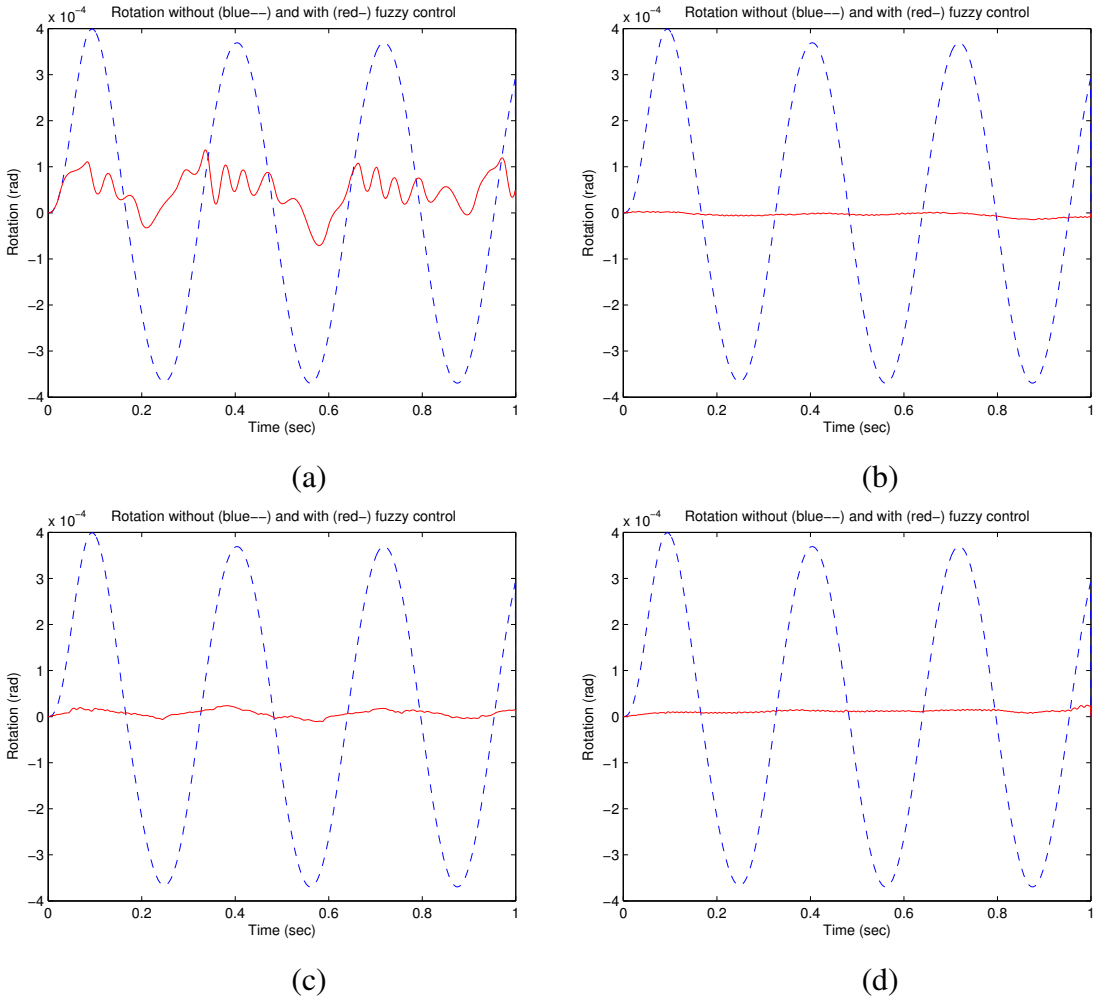


Figure 7: Rotation at the free end for classical fuzzy controller (a), for the fuzzy controller optimized by DE (b), for the fuzzy controller optimized by PSO (c) and for the fuzzy controller optimized by GA (d).

Global Optimization over Continuous Spaces", Journal of Global Optimization, 11(4), 341-359, 1997.

[14] V. Feoktistov, "Differential Evolution - In Search of Solutions", Springer, NY, 2006.

[15] K.V. Price, R.M. Storn, J.A. Lampinen, "Differential Evolution: A Practical Approach to Global Optimization", Springer, Berlin, 2005.

[16] A.P. Engelbrecht, "Computational Intelligence: An Introduction", John Wiley and Sons, England, 2007.

[17] M. Marinaki, Y. Marinakis, G.E. Stavroulakis, "Fuzzy control optimized by PSO for vibration suppression of beams", Control Engineering and Practice, 18, 618-629, 2010.

[18] C.N. Della, D. Shu, "Vibration of Beams with Piezoelectric Inclusions", International Journal of Solids and Structures, 44, 25092522, 2007.

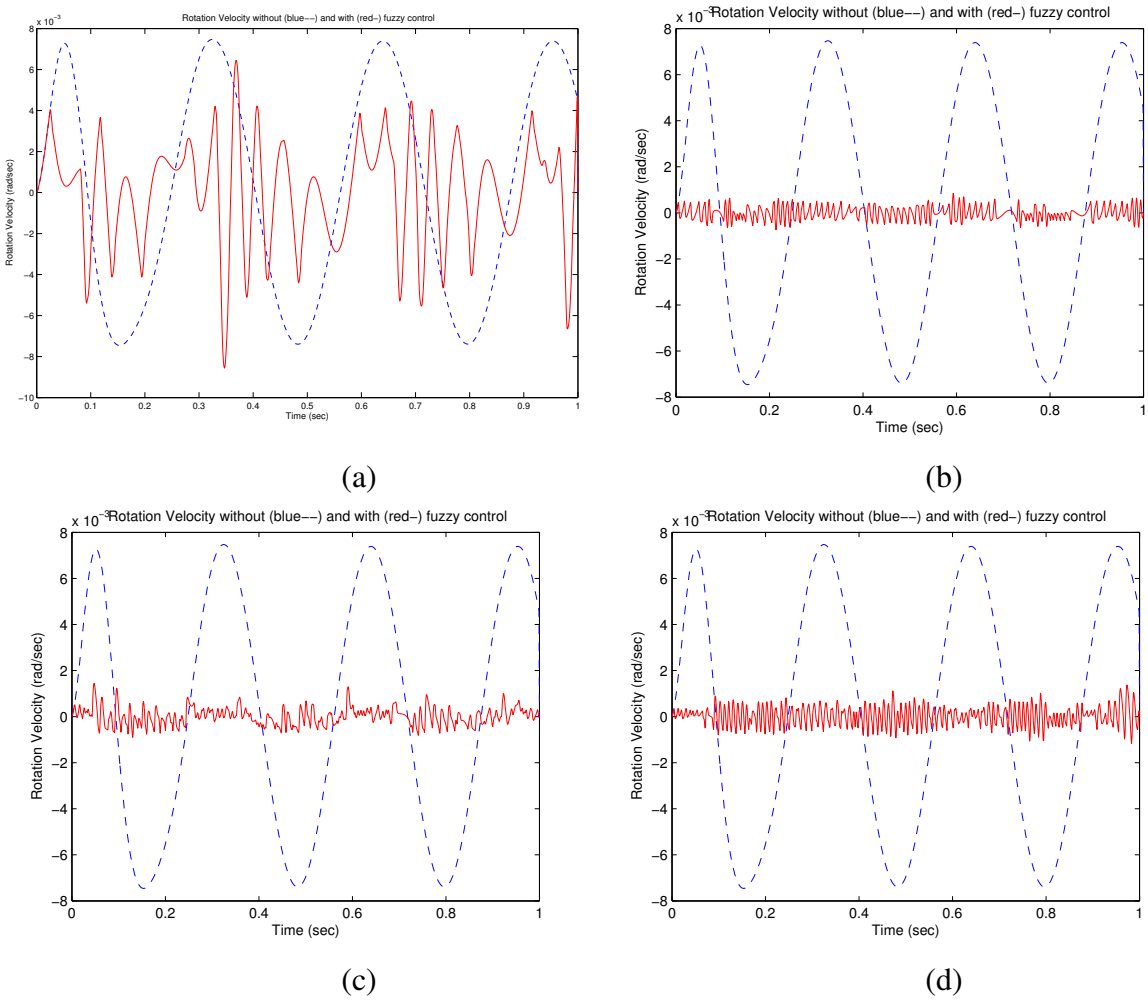


Figure 8: Rotation velocity at the free end for classical fuzzy controller (a), for the fuzzy controller optimized by DE (b), for the fuzzy controller optimized by PSO (c) and for the fuzzy controller optimized by GA (d).

[19] G. Foutsitzi, D.G. Marinova, E. Hadjigeorgiou, G.E. Stavroulakis, "Robust H2 Vibration Control of Beams with Piezoelectric Sensors and Actuators", International Conference, Physics and Control, 1, 157-162, 2003.

[20] S. Kapuria, "An Efficient Coupled Theory for Multilayered Beams with Embedded Piezoelectric Sensory and Active Layers", International Journal of Solids and Structures, 38, 9179-9199, 2001.

[21] D. Sun, L. Tong, S.N. Atluri, "Effects of Piezoelectric Sensor/Actuator Debonding on Vibration Control of Smart Beams", International Journal of Solids and Structures, 38, 9033-9051, 2001.

[22] M.A. Trindade, A. Benjeddou, R. Ohayon, "Piezoelectric Active Vibration Control of Damped Sandwich Beams", Journal of Sound and Vibration 246(4), 653-677, 2001.

[23] M.T. Valoor, K. Chandrashekhara, S. Agarwal, "Self-adaptive Vibration Control

of Smart Composite Beams using Recurrent Neural Architecture", International Journal of Solids and Structures, 38, 7857-7874, 2001.

[24] O.J. Aldraihem, R.C. Wetherhold, T. Sigh, "Distributed Control of Laminated Beams: Timoshenko Theory vs. Euler-Bernoulli Theory", Journal of Intelligent Mathematical Systems and Structures, 8, 149-157, 1997.

[25] E.P. Hadjigeorgiou, G.E. Stavroulakis, C.V. Massalas, "Shape Control and Damage Identification of Beams using Piezoelectric Actuation and Genetic Optimization", International Journal of Engineering Science, 44, 409-421, 2006.

[26] G.E. Stavroulakis, G. Foutsitzi, E. Hadjigeorgiou, D.G. Marinova, C.C. Baniotopoulos "Design and Robust Optimal Control of Smart Beams with Application on Vibrations Suppression", Advances in Engineering Software, 36, 806-813, 2005.

[27] G.K. Tairidis, G.E. Stavroulakis, D.G. Marinova, E.C. Zacharenakis, "Classical and Soft Robust Active Control of Smart Beams", ECCOMAS Thematic Conference on Computational Methods in Structural Dynamics and Earthquake Engineering, M. Papadrakakis, D.C. Charmpis, N.D. Lagaros, Y. Tsompanakis, (Editors), Rethymno, Crete, Greece, 2007.

[28] D. Driankov, H. Hellendoorn, M. Reinfrak, "An Introduction to Fuzzy Control", 2nd edition, Springer Verlag, New York, 1996.

[29] G.C. Marano, G. Quaranta, "Fuzzy-based Robust Structural Optimization", International Journal of Solids and Structures, 45, 3544-3557, 2008.

[30] S. Pourzeynali, H.H. Lavasani, A.H. Modarayi, "Active Control of High Rise Building Structures using Fuzzy Logic and Genetic Algorithms", Engineering Structures, 29, 346-357, 2007.

[31] A.-P. Wang, Y.-H. Lin, "Vibration Control of a Tall Building Subjected to Earthquake Excitation", Journal of Sound and Vibration, 299, 757-773, 2007.

[32] D.G. Reigles, M.D. Symans, "Supervisory Fuzzy Control of a Base Isolated Benchmark Building utilizing a Neuro-Fuzzy Model of Controllable Fluid Viscous Dampers", Journal of Structural Control and Health Monitoring, 13, 724-747, 2006.

[33] K. Belarbi, F. Titel, "Genetic Algorithm for the Design of a Class of Fuzzy Controllers: An Alternative Approach", IEEE Transactions on Fuzzy Systems, 8(4), 398-405, 2000.

[34] C.H. Chou, "Genetic Algorithm-based Optimal Fuzzy Controller Design in the Linguistic Space", IEEE Transactions on Fuzzy Systems, 14(3), 372-385, 2006.

[35] C.F. Juang, J.Y. Lin, C.T. Lin, "Genetic Reinforcement Learning Through Symbiotic Evolution for Fuzzy Controller Design", IEEE Transactions on Systems Man and Cybernetics, Part B: Cybernetics, 30(2), 290-302, 2000.

[36] C.J. Lin, "A GA-based Neural Fuzzy System for Temperature Control", Fuzzy Sets and Systems, 134, 311-333, 2004.

[37] O. Cordón, F. Gomide, F. Herrera, F. Hoffmann, L. Magdalena, "Ten Years of Genetic Fuzzy Systems: Current Framework and New Trends", Fuzzy Sets and Systems, 141(1), 5-31, 2004.

[38] R. Welch, G.K. Venayagamoorthy, "A Fuzzy-PSO Based Controller for a Grid

Independent Photovoltaic System", in "Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007)", 2007.

[39] B. Niu, Y. Zhu, X. He, H. Shen, "A Multi-swarm Optimizer Based Fuzzy Modeling Approach for Dynamic Systems Processing", Neurocomputing, 71, 1436-1448, 2008.

[40] K.H. Youssef, H.A. Yousef, O.A. Sebakhy, M.A. Wahba, "Adaptive Fuzzy APSO based Inverse Tracking-controller for DC Motors", Expert Systems with Applications, 36(2), 3454-3458, 2009.

[41] C.F. Juang, C. Lo, "Zero-order TSK-type Fuzzy System Learning using a Two-phase Swarm Intelligence Algorithm", Fuzzy Sets and Systems, 159, (21), 2910-2926, 2008.

[42] M. Eftekhari, S.D. Katebi, M. Karimi, A.H. Jahanmiri, "Eliciting Transparent Fuzzy Model using Differential Evolution", Applied Soft Computing, 8, 466-476, 2008.

[43] F. Cheong, R. Lai, "Designing a Hierarchical Fuzzy Logic Controller using the Differential Evolution Approach", Applied Soft Computing, 7, 481491, 2007.

[44] E.I. Papageorgiou, P.P. Groumpos, "A New Hybrid Method using Evolutionary Algorithms to Train Fuzzy Cognitive Maps", Applied Soft Computing, 5, 409431, 2005.

[45] G.K. Tairidis, G.E. Stavroulakis, D.G. Marinova, E.C. Zacharenakis, "Classical and Soft Robust Active Control of Smart Beams. Computational Structural Dynamics and Earthquake Engineering", M. Papadrakakis, D.C. Charmpis, N.D. Lagaros, Y. Tsompanakis, (Editors), Chapter 11, 165-178, CRC Press/Balkema and Taylor & Francis Group, London, U.K., 2009.

[46] J.H. Holland, "Adaptation in Natural and Artificial Systems", University of Michigan Press, Ann Arbor, MI, 1975.

[47] D.E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley Publishing Company, INC, Massachussets, 1989.