Structural Engineering and Mechanics, *Vol. 61, No. 3 (2017) 347-357*
DOI: https://doi.org/10.12989/sem.2017.61.3.347

Chaotic particle swarm optimization in optimal active control of shear buildings

Saeed Asil Gharebaghi* and Ehsan Zangooei[a]

Civil Engineering Faculty, K. N. Toosi University of Technology, Tehran, Iran

(Received November 25, 2015, Revised October 20, 2016, Accepted November 3, 2016)

Abstract. The applications of active control is being more popular nowadays. Several control algorithms have been developed to determine optimum control force. In this paper, a Chaotic Particle Swarm Optimization (CPSO) technique, based on Logistic map, is used to compute the optimum control force of active tendon system. A chaotic exploration is used to search the solution space for optimum control force. The response control of Multi-Degree of Freedom (MDOF) shear buildings, equipped with active tendons, is introduced as an optimization problem, based on Instantaneous Optimal Active Control algorithm. Three MDOFs are simulated in this paper. Two examples out of three, which have been previously controlled using Lattice type Probabilistic Neural Network (LPNN) and Block Pulse Functions (BPFs), are taken from prior works in order to compare the efficiency of the current method. In the present study, a maximum allowable value of control force is added to the original problem. Later, a twenty-story shear building, as the third and more realistic example, is considered and controlled. Besides, the required Central Processing Unit (CPU) time of CPSO control algorithm is investigated. Although the CPU time of LPNN and BPFs methods of prior works is not available, the results show that a full state measurement is necessary, especially when there are more than three control devices. The results show that CPSO algorithm has a good performance, especially in the presence of the cut-off limit of tendon force; therefore, can widely be used in the field of optimum active control of actual buildings.

Keywords: Chaotic Particle Swarm Optimization; logistic map; instantaneous optimal active control; active tendon system; shear buildings; LPNN; BPFs

1. Introduction

During past decades, the ineffectiveness of the traditional designs has been proven, and led to the application of smart structures in civil engineering. Although the optimal design of structures against damages, caused by earthquakes, strong winds, or other disasters may be the major challenge in structural engineering, the concept of controlling such structures may be considered as another alternative. In the field of structural engineering, the concept of controlling the seismic structural response originally developed by Kobori and Minai (1960). Beside the other control methods, particularly semi-active control in engineering structures (Kerbouaa *et al.* 2014), the idea of active control has been utilized in various areas of engineering such as aerospace, mechanical, and electrical engineering (Block *et al.* 1997, Wu *et al.* 2004, Oliveira *et al.* 2009).

An active-controlled structure may be considered as a smart structure that can sense the external dynamic loads via sensors and adjust its controlling efforts immediately, in such a way that it would be able to resist against external dynamic loads such as earthquake, strong winds, and impact loads (Fisco and Adeli 2011). More precisely, the control unit of an active control system senses and generates necessary controlling signals, upon a predetermined control algorithm to drive the actuator, and limit structural responses to desired values. Therefore, it is clear that some control algorithms are necessary to be used in designing of the controller in order to compute the best control signal or force, based on a control objective or criteria. Typical control algorithms are H_2 (Dyke *et al.* 1996), H_∞ (Jabbari *et al.* 1995), intelligent control based on fuzzy logic (Akinori *et al.* 1998), neural networks (Cho *et al.* 2005) and genetic algorithms (Kim and Ghaboussi 2007, Li *et al.* 2002). In order to design an active control system, the control algorithm must satisfy the control objective. The control objective may be chosen to minimize the structural response with minimum required energy or force. Besides, a performance index is used in such situations to achieve a balance between the need of reducing structural response and minimizing control forces simultaneously. In this paper, the Instantaneous time-dependent performance index $J_p(t)$ is considered as a performance index of optimization criteria.

As it was mentioned before, an optimization algorithm is required to find the optimum signal or force, which will be applied to structure. Stochastic global search methods like Genetic Algorithm (GA) have been employed in structural active control (Ghanbarpor and Mohebbi 2006). However, when the system has a highly epistemic objective function (i.e., where parameters being optimized are highly correlated), and there are many parameters to be optimized,

*Corresponding author, Assistant Professor
E-mail: asil@kntu.ac.ir
[a]M.Sc. Graduate
E-mail: ehsan.zangooei@gmail.com

ISSN: 1225-4568 (Print), 1598-6217 (Online)

the efficiency of these methods in obtaining global optimum solution decreases. Beside this drawback, in the above-mentioned situations, the simulation process uses a lot of computing time and other resources (Shayeghi *et al.* 2009). Thus, it will be necessary to perform a full state measurement. Kalman filter could be considered as an effective solution; however, an inappropriate selection of the noise parameters of this filter may leave a negative impact on its performance. Yuen *et al.* (2013) have proposed an online estimation method for aforementioned parameters to resolve the issue. Furthermore, the extended Kalman filter have been adopted to identify the nonlinearity of structures (Lei *et al.* 2015). The method is capable of being used in structural health monitoring. Full state measurement has not been performed in the current study. Indeed, the authors are focusing on the efficiency of Chaotic Particle Swarm Optimization method (CPSO) optimal control in finding the best control force. Nevertheless, the time history of Central Processing Unit (CPU) time has been presented for each example.

The Particle Swarm Optimization (PSO) algorithm demonstrates an excellent performance under complicated conditions (Altinoz *et al.* 2010). It has been found to be appropriate and reliable in solving discrete, nonlinear, non-differentiable and multi-modal problems (Cheng *et al.* 2007, Shi and Eberhart 1998). Various chaotic versions of PSO have been proposed (Liu *et al.* 2005, Chuanwen and Bompard 2005, Chuang *et al.* 2011, Yang *et al.* 2012). Logistic version of CPSO is a fast method for solving optimization problems. It inherits its characteristics from original PSO, but adds new properties of a chaotic exploration. In the present paper, CPSO (Logistic map) is used as a tool for structural vibration control for minimizing the structural responses. The chaotic behavior of the exploration could increase the search domain by the passage of time, whereas using random variables especially pseudo-random variables generated by common computer programs cannot ensure optimal ergodicity in the search space (Chuanwen and Bompard 2005). The problem of structural control of a building with active tendon is formulated as an optimization problem, and the control force is obtained by minimizing the IOAC-based fitness function using CPSO technique to get the maximum reduction in the displacement and velocity responses of the floors, where the cut-off limit of tendon forces are considered simultaneously. The efficiency of the simulation is evaluated for a three-story shear building under California (1952), El Centro (1940) and Northridge (1994) earthquakes. This three-story shear building has been used as a benchmark in the literature of active control (Kim *et al.* 2008); therefore, it is possible to compare the results of reduced structural response. Another example, a ten-story shear building, is provided from the literature. The building has been previously controlled using ten active tendons. In this example, the control algorithm is based on Block Pulse Functions (BPFs) (Ghaffarzadeh and Younespour 2014). Although the above-mentioned researches do not provide the time history of control force, in the present paper, the amount of the force is assumed to be limited, and its time history is presented. It is obvious that the response of a structure will dramatically decrease if the amount of the cut-off limit of control force is increased. Thus, the control process of a structure is more challenging when the amount of the allowable control force is limited to a specific value. Despite of the absence of the time history of CPU time in the aforementioned examples, CPU time have been depicted in all presented examples. In the last section of the numerical examples, a twenty-story shear building is controlled using three active tendons. All results demonstrate the efficiency of CPSO-based control algorithm.

2. Active control of MDOF shear building

In this paper, the proposed algorithm has been applied to shear buildings. A typical shear building could be seen in Fig. 3. The governing equations of an MDOF shear building with N degrees of freedom (or N story), can be expressed as

$$[M]\{\ddot{x}_{(t)}\} + [C]\{\dot{x}_{(t)}\} + [K]\{x_{(t)}\} = -[M]\{l\}\ddot{x}_{g(t)} + [\gamma]\{u_{(t)}\} \tag{1}$$

where

$$[M] = \begin{bmatrix} m_1 & 0 & \cdots & 0 \\ 0 & m_2 & \cdots & 0 \\ \vdots & \vdots & \vdots\vdots\vdots & \vdots \\ 0 & 0 & \dots & m_N \end{bmatrix} \tag{2}$$

$$[C] = \begin{bmatrix} c_1 + c_2 & -c_2 & 0 & \cdots & 0 \\ -c_2 & c_2 + c_3 & -c_3 & \cdots & 0 \\ 0 & -c_3 & c_3 + c_4 & \dots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & c_N \end{bmatrix} \tag{2}$$

$$[K] = \begin{bmatrix} k_1 + k_2 & -k_2 & 0 & \cdots & 0 \\ -k_2 & k_2 + k_3 & -k_3 & \cdots & 0 \\ 0 & -k_3 & k_3 + k_4 & \dots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & k_N \end{bmatrix} \tag{3}$$

In Eqs. (2)-(4), $[M$, $[C]$ and $[K]$ are $N \times N$ matrices of mass, damping and stiffness of the structure, respectively. Besides, $\{\iota\}$ of $N \times 1$ is the impact vector for earthquake ground acceleration $\ddot{x}_{g(t)}$, and $\{x_{(t)}\}$ of $N \times 1$ and $\{u_{(t)}\}$ of $r \times 1$ are the vectors of floor displacements and control forces of active tendons, respectively. The matrix $[\gamma]$ of $N \times r$ is the location matrix of control forces of active tendons. Eq. (1) includes N differential equations, which governs the relative displacements of a linear elastic MDOF system, subjected to an earthquake ground motion $\ddot{x}_{g(t)}$. In the case of a structure with active tendons, installed at each floor, the controller location matrix $[\gamma]$ of $N \times N$ takes the following form

$$[\gamma] = \begin{bmatrix} -1 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & -1 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \dots & \dots & \dots & \vdots & \vdots \\ \vdots & \vdots & \dots & \dots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -1 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & -1 \end{bmatrix}_{\mathrm{N \times N}} \tag{4}$$

Moreover, the control force $\{u_{(t)}\}$ of $N \times 1$ is as

$$\{u_{(t)}\} = [u_{1(t)}, u_{2(t)}, \ldots, u_{i(t)}, \ldots, u_{N(t)}]^T \quad (5)$$

As it can be seen from Eq. (1), in each time step, the Instantaneous Optimal Active Control (IOAC) algorithm is used to compute the amount of control force. In fact, an optimization problem should be solved in each step to find the best control force. The optimization phase, needs time to be done; therefore, when the optimization is accomplished, the structure will be in a new state, in which not only the computed control force may lose its efficiency, but it may harm the structure. This is an example of uncertainty. In real world, a structure may have thousands of DOFs. As a result, there would be a considerable time lag, which is inevitable because of solving the governing equations, optimizing the control force, *etc*. This time delay increases the aforementioned uncertainty. In the theory of control, it is a challenging filed of research, to reduce such uncertainties. Although using modern computers and applying high performance techniques may reduce the amount of time delay, which is elapsed in computing the best control force, the problem remains unsolved. In the present study, the amount of time, used by CPU of computer will be presented, and compared with the sampling time step of excitation.

In a realistic control process, it is necessary to perform full state measurement. Kalman filter is a well-known algorithm, which is recursively executed to estimate the state of a dynamical system (Kalman, 1960). It has two essential steps: prediction, and filtering/updating. It assumes that the external force has two components. The first one is deterministic, and the second is a stochastic component, which is modeled as a Gaussian process with zero mean and covariance matrix of the external force $\{f_{(t)}\}$. The external force could be considered as the right hand side of Eq. (1). Now, the governing equation could be rewritten as

$$\{\dot{Z}_{(t)}\} = [A]\{Z_{(t)}\} + [B]\{f_{(t)}\} \quad (6)$$

in which, $\{Z_{(t)}\}$ is the $2N \times 1$ state vector

$$\{Z_{(t)}\} = \begin{Bmatrix} \{x_{(t)}\} \\ \{\dot{x}_{(t)}\} \end{Bmatrix} \quad (7)$$

$[A]$, and $[B]$ are computed as the following

$$[A] = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix} \quad (8)$$

$$[B] = \begin{bmatrix} 0 \\ -M^{-1}T_0 \end{bmatrix} \quad (9)$$

Eq. (7) could be discretized as

$$Z_{k+1} = A_d Z_k + B_d f_k \quad (10)$$

Eq. (11) is the first equation of the state of a linear, time-invariant dynamical discrete-time system. In Eq. (11), k denotes the iteration number. The output of this state vector would be

$$y_{k+1} = C_d Z_{k+1} + D_d F_{D,k+1} + n_{k+1} \quad (11)$$

In which, C_d is the observation matrix, and D_d is the input distributing matrix of the deterministic component of external force F_D. C_d and D_d matrices could be easily obtained by sensor configuration. Besides, n_{k+1} represents the measurement noise, which is a Gaussian process with zero mean and covariance matrix $\Sigma_{n,k+1}(\theta_{n,k+1})$ at the k^{th} step. $\theta_{n,k+1}$ is the characteristic parameter vector of the covariance matrix $\Sigma_{n,k+1}$. The measurement noise and the noise in the stochastic component of the external force are assumed to be statistically independent (Kalman 1960). Eq. (12) is the second equation of the state of a linear, time-invariant dynamical discrete-time system. Assuming that the measurements of the output y_i are given up to k^{th} time step, it would be possible to predict the state vector at $(k+1)^{th}$ step.

Although the Kalman filter provides a well-posed formulation, the accuracy of computed state vector depends on the prior selection of covariance matrices. In fact, this highlights the importance of the reliable time-varying noise covariance, which in general is not available. Thus, the original formulation could result in inappropriate estimation. Yuen *et al*. (2013) have presented a new approach to implement online estimation of process and measurement noises. They have utilized Bayesian method to estimate optimal noise parameters. Besides, their method is capable of being used for nonstationary conditions. The state estimation is more challenging in nonlinear systems. In most practical situations, it is necessary to identify nonlinear structure, especially when the location of nonlinearities are not clear. For instance, the identification of structural nonlinearity could be useful in structural health monitoring. Lei *et al.* (2015) have used partial measurements of structural response and extended Kalman filter to identify the structural nonlinearities.

In the current study, the main attempt is to implement CPSO-based control algorithm, as well as optimizing the program to reduce the CPU time.

3. Optimization method for active control

3.1 Chaotic particle swarm optimization algorithm

Iterative optimization is as old as human life, even very primitive beings activities has been formed according to the motivation of “To improve the situation”. Many strategies that researchers encounter every day in nature and prove their effectiveness in human’s various actions, already may offer a broad range of new solution methods. Therefore, it is not unexpected that the origin of several mathematical models of optimization has been taken from biological behaviors. Among these models, those corresponding to social behavior can be distinguished from the methods using individual behavior. For instance, Particle Swarm Optimization (PSO) is a population-based evolutionary algorithm developed by Kennedy and Eberhart (1995), and has been motivated by the social behavior of organisms. In 1927, Karl Von Frisch had discovered that bees brought back to the hive not only nectar and pollen, but also information. Unfortunately, a well-posed model explaining the search strategy was not proposed at that time (Maurice 2006).

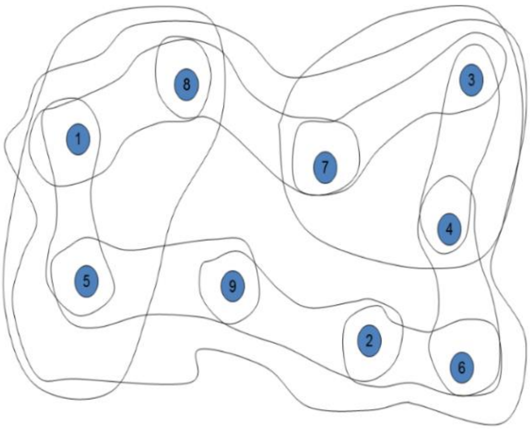
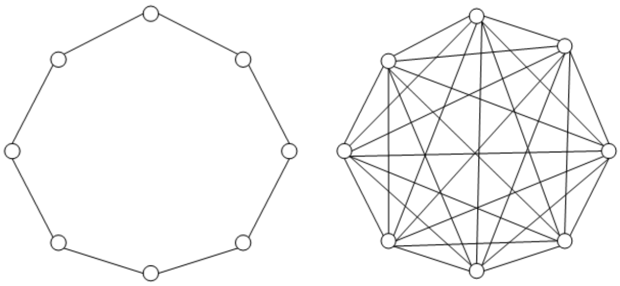


Fig. 1 The neighborhood principle



(a) Ring neighborhood structure ($pbest$)

(b) Star neighborhood structure ($gbest$)

Fig. 2 Social network structure

PSO is a heuristic global technique that is broadly used for discrete and continuous optimization problems. It exhibits evolutionary computation features as the followings (Hu and Eberhart 2002):

i. It is started with a population of random solutions.
ii. Optimum solution is searched by updating generations.
iii. Previous generations are updated in subsequent steps.

In this method, potential solutions which are called "particles", in a continuous D-dimensional space travel and search the global optima. Each particle in PSO keeps track of its coordinates in the solution space, associated with the best previous solution. This value is called $pbest$. Additionally, the overall best value that is called $gbest$ and its location is obtained by all particles in the population. Each particle has all the other particles as neighbors, as is illustrated in Fig. 1. This implies that the global best particle position for all particles is unique, as is shown in Fig. 2. Each particle adjusts its movement, based on its own experience and its companions' experiences.

Thus the previous particle's best position ($pbest$) and population's overall best position ($gbest$) are used to update the particle velocities. As was mentioned before, the original PSO algorithm uses some random variables in order to contribute the acceleration of all particles from one iteration to the next iteration.

Even the best Random Number Generators (RNGs), use some arithmetic operations. For example, a Linear Congruential Generator (LCG) uses a recursive equation containing some integers

$$x_{n+1} = (ax_n + c)\ mod\ m \tag{12}$$

In which, x_n is the previously generated random number. As it can be seen, Eq. (13) only generates integers numbers. The initial value of x_0 is called "seed" and should be known to initiate the process. The parameters a, c, and m are the constants of the method. Eq. (13) shows that the longest period of all computed integers will be equal to m, and they will fall in $[0, m-1]$. Furthermore, it is obvious that using a predefined x_0, all values of x_n will be deterministic and there will be no random behavior. In other word, the first condition of an ideal RNG will be ignored if the LCG formula is utilized as an RNG with a constant x_0. The first characteristic of a True RNG states that an ideal RNG is a discrete memoryless information source (Stojanovski and Kocarev 2001). One solution to this obstacle is using another different x_0 at the first of each attempt of random number generation process. Most of RNGs use CPU time in order to initial different seeds.

Moreover, according to the capacity of a computer in storing an integer, which is limited to a specific number of bits, a computer is only able to generate random integer numbers in a predefined limited period. For instance, in a 64-bit computer, the maximum value of m would be equal to 2^{64}. Although the period of such RNGs is long, they could not be assumed as a True RNG. In fact, the generated integer numbers will be repeated by the passage of time. From a technical point of view, they are called Pseudo-Random Number Generators (PRNGs) ("Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin", John von Neumann) (Stojanovski and Kocarev 2001). The random variables, in original version of PSO, cannot ensure optimal ergodicity in the search space (Chuanwen and Bompard 2005).

Instead of using LCGs, some Chaos-based RNGs have been proposed, which utilize chaotic maps. Some methods use bit operations (François *et al.* 2014), whereas other methods directly use the chaotic maps in PSO (Chuang *et al.* 2011, Yang *et al*. 2012). Although such Chaos-based RNGs use arithmetic operations, and they are not true RNGs, there are no fixed points, periodic orbits, or quasi-periodic orbits in the behavior of a chaotic system (Kuo 2005). More precisely, when a chaotic map is used in PSO, the numbers generated using the map are in floating-point format. Owing to the fact that a computer is capable of storing floating-point numbers in a wider range, the period of the generated numbers increases drastically. The chaotic version of PSO, in which the random variables are replaced with chaotic counterparts, can resist being trapped in local minima in comparison to original PSO (Liu *et al*. 2005). A detailed statistical comparison among several versions of PSO has been done by Yang *et al*. (2012). They have indicated the rank sums of CPSO (Logistic map), CPSO (Sinusoidal map), CPSO (Tent map), CPSO (Gauss map), CPSO (Circle map), CPSO (Arnold map), CPSO (Sinai map), CPSO (Zaslavskii map), and PSO as 239, 215, 153, 244, 266, 100, 150, 184, and 203 respectively. According to their research, any two methods which are more than 30.29 units apart (=0.05), may be considered as having unequal performances. It is notable that CPSO (Logistic map) has a rank of 239, while the original PSO has a rank of 203. This indicates that not only they have unequal performances, but CPSO is slightly better than PSO. Chuang *et al.* (2011) have presented Chaotic CatfishPSO, a modified chaotic version of PSO, and have compared its performance to original PSO, CPSO, and CatfishPSO. They have concluded that CPSO solves multimodal optimization problems very fast. In their work, C-CatfishPSO outperformed CPSO. However, even in the worst cases of their tests, CPSO performed the optimization slightly better than PSO (Chuang *et al.* 2011). Although in Rastrigin, Griewank, and Sphere benchmarks both CPSO and PSO were trapped in local minima, in Rosenbrock, Ackley, and Schwefel benchmarks CPSO outperformed PSO, and had a considerable performance.

Because of the simplicity and less computational effort of using Logistic map, and the above-mentioned advantages of CPSO (Logistic map) in comparison to PSO, in the present study, chaotic numbers have been utilized instead of random numbers. More precisely, in each time step, Chaotic PSO algorithm (CPSO) accelerates the velocity of each particle toward its $pbest$ and $gbest$ positions according to Eq. (14). The acceleration is weighted by chaotic terms, with separate chaotic numbers r_1 and r_2 for accelerating toward $pbest$ and $gbest$ locations, respectively.

$$v_i^{t+1} = w.v_i^t + c_1.r_1.(P_i^t - x_i^t) + c_2.r_2.(P_g^t - x_i^t) \quad (13)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (14)$$

Where $i = 1, 2, \ldots, N$ indicates the number of particles in the population, $t = 1, 2, \ldots, t_{max}$ indicates the number of iterations, w is the inertia, $v_i = [v_{i1}, v_{i2}, \ldots, v_{in}]^T$ stands for the velocity of the i^{th} particle, $x_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T$ represents the position of the i^{th} particle of the population, and $P_i = [P_{i1}, P_{i2}, \ldots, P_{in}]^T$ indicates the best previous position of the i^{th} particle. The positive constants c_1 and c_2 are the cognitive and social components, respectively. As it was mentioned before, c_1 and c_2 have acceleration effects on the particle velocity toward $pbest$ and $gbest$, respectively. The index g represents the index of the best particle among all the particles in the swarm. The variables r_1 and r_2 are chaotic numbers, which are separately generated using the logistic map

$$r_k^{t+1} = 4.r_k^t.(1 - r_k^t) \quad (15)$$

For $k = 1, 2$ the logistic map generates independent chaotic values in [0, 1] for i^{th} particle. The idea of using chaotic parameters comes from their characteristics: they are not periodic, and they cover the phase space by the passage of time.

CPSO is similar to continuous Genetic Algorithm in initialization phase because both of them use random population. Unlike GA, CPSO does not have evolution operators such as crossover and mutation. In fact, the CPSO algorithm is composed of certain number of particles, which are randomly initialized. Each particle is represented by two vectors in order to characterize its position and velocity in a multi-dimensional space. These particles simultaneously move in the D-dimensional space of the optimization problem iteratively, and search for new possible locations, then compute their fitness value as measurement criteria. The dimension of the optimization problem space is equal to the number of parameters of the function to be optimized, which is called the objective function. Two important memories are used in order to save the best results. The first one is used for saving particle's best previous position, and the second for the best position among all particles. Using each experience, obtained from each iteration and the best results, each particle decides about its movement trajectory. According to the above-mentioned approach, the CPSO implementation is presented as follows:

i. Create a random population of particles, also called agents, which are uniformly distributed in the D-dimensional space. In this paper, D is the number of active tendons. In fact, each particle represents a unique combination of control force(s). If there are D number of tendons, the best combination of control force(s) is desired, and a D-dimensional optimization problem should be solved.

ii. Evaluate the fitness value of each particle according to the objective function. From a mathematical point of view, it means to analyze the structure under earthquake and tendon force(s), and to compute the corresponding index, which results in a fitness value for each particle.

iii. Compare each particle's fitness value with that particle's $pbest$, which has been evaluated so far. If the current position of a particle is better than its previous best position, then replace the $pbest$ of that particle with its current location, and save the corresponding information.

iv. Determine the current global best particle according to the particles' best locations. If the current value is better than $gbest$, then reset $gbest$ to the current global best value, and save the corresponding information.

v. Update the velocity and position of each particle according to Eqs. (14)-(15) using chaotic parameters to insure that no repeated velocity and location is generated.

vi. Go to the step (ii) until a termination criterion is satisfied.

vii. The particle, which has been made the $gbest$ value, will be assumed as the best particle. Therefore, the components of that particle will be the amount of the control force(s), acting on the corresponding degree of freedom.

The termination criterion is satisfied, if:

a. A suitable tolerance is achieved, or

b. The smallest value of objective function remains unchanged over a certain number of generations, or

c. The number of iterations exceeds a certain maximum value.

3.1 Cost or objective function

Objective function is used to present a measurement of the performance of particles: how particles have acted in the optimization problem space. The best particle, for example in minimization problems, will have the lowest numerical value of the objective function. Moreover, in the optimization process, various functions could be used to achieve a certain performance level. However, it is obvious that selecting different objective functions leads to different results. It should be noted that the choice of suitable objective function is very important in defining the procedure because a proper selection may increase the performance of achieving a desired level of optimization (Shayeghi *et al.* 2008). There are various optimization criteria in the field of active control of structures. In this paper, the performance index of Instantaneous Optimal Active Control is used as the objective function to determine the best control force. Yang *et al.* adopted this idea in order to control structures under earthquake excitation (Yang *et al.* 1987). IOAC algorithm computes optimal control force $\{u_{(t)}\}$ by minimizing an instantaneous time-dependent performance index expressed as

$$Jp_{(t)} = \{Z_{(t)}\}^T[Q]\{Z_{(t)}\} + \{u_{(t)}\}^T [R]\{u_{(t)}\} \quad (17)$$

Where $\{u_{(t)}\}$ of $r \times 1$ is the vector of the control forces of active tendons. $\{Z_{(t)}\}$ is the state vector as defined in Eq. (8). The matrix $[Q]$ of $2n \times 2n$ is a positive semi-definite symmetrical matrix, in which n is the number of degrees of freedom. The matrix $[R]$ of $r \times r$ is a positive definite symmetrical matrix so that all control forces affect the value of the performance index. r represents the number of tendon force(s). The performance index establishes a weighted balance between structural response and control energy. In one hand, when the value of $\{Z_{(t)}\}^T[Q]\{Z_{(t)}\}$ is big, in comparison to the elements of $\{u_{(t)}\}^T [R]\{u_{(t)}\}$, the responses of the structure are significantly reduced at the expense of increased control energy. On the other hand, when the elements of $\{u_{(t)}\}^T [R]\{u_{(t)}\}$ are big, in comparison to the elements of $\{Z_{(t)}\}^T[Q]\{Z_{(t)}\}$, the control energy is reduced, but there will be no guarantee that the algorithm can significantly reduce structural responses. Although the values of the elements of $[Q]$ and $[R]$ are important, the overall product of the first and second terms of the right hand side of Eq. (17) do determine the value of reduction in responses or control force(s). For instance, when a problem is solved in SI units, the desired values of displacements and velocities of structure are small numbers (in m and m/s),whereas the value of control force(s) are big numbers in N. This results in big elements in $[Q]$ and small values in $[R]$ to make a balanced index. Nevertheless, it does not affect the numerical optimization except for very big or small floating-point numbers, which may cause overflow, underflow, or cut-off errors. As a result, the active control problem is formulated considering the following constrained optimization problem, where the constraint is the cut-off limit of control force.

$$\begin{gathered} Minimize\ Jp(t)\ wih \\ u_{c,min} < u_{(t)} < u_{c,max} \end{gathered} \quad (18)$$

Eventually, CPSO is used to solve the above-mentioned optimization problem. The best control force, which minimizes Eq. (17), is repeatedly computed and applied to the structure in order to reduce structural responses.

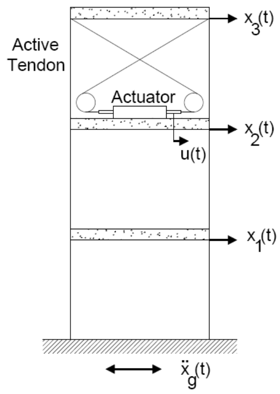


Fig. 3 Three-story shear building with active tendon (Kim *et al.* 2008)

Table 1 Modeling Parameters for Three-Story Building (Kim *et al.* 2008)

Parameter	Value
Mass matrix *M* (kg)	$\begin{bmatrix} 98.3 & 0 & 0 \\ 0 & 98.3 & 0 \\ 0 & 0 & 98.3 \end{bmatrix}$
Stiffness matrix *K* (N/m)	$10^5 \times \begin{bmatrix} 13.68 & -6.84 & 0 \\ -6.84 & 12 & -5.16 \\ 0 & -5.16 & 5.16 \end{bmatrix}$
Damping matrix *C* (N.s/m)	$\begin{bmatrix} 100 & -50 & 0 \\ -50 & 175 & -125 \\ 0 & -125 & 125 \end{bmatrix}$

4. Numerical simulations

4.1 Three-story shear building

A three-story shear building is considered to compare CPSO controlling algorithm with previous works in the literature. The model, shown in Fig. 3, has been studied and controlled, using Lattice type Probabilistic Neural Network (LPNN) method (Kim *et al.* 2008). It is notable that there is no evidence of the time history and magnitude of the control force in their work; however, in the present study, the control force is limited to a specific range, which is closer to the reality of structural control. In other word, it is possible to reduce the responses of this three-story building by increasing the amount of control force unlimitedly, as it was mentioned in the interpretation of $[Q]$ and $[R]$ in Eq. (17). Nevertheless, the amount of external control force should be actual and applicable according to the capacity of actuators or tendons. In the present study, the maximum allowable value of the control force is limited. For instance, in the first example, the control force should not exceed 200N.

The structural properties are given in Table 1. The control force applied to the structure is determined by minimizing the objective function, expressed in Eq. (17). In this simulation, the number of particles, the number of

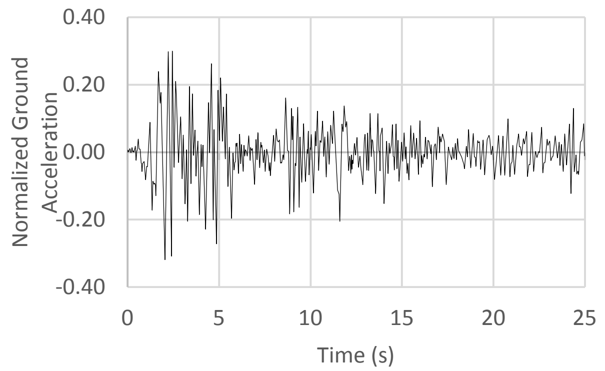


Fig. 4 North-South component of horizontal ground acceleration, El Centro, California during the Imperial Valley earthquake of May 18, 1940

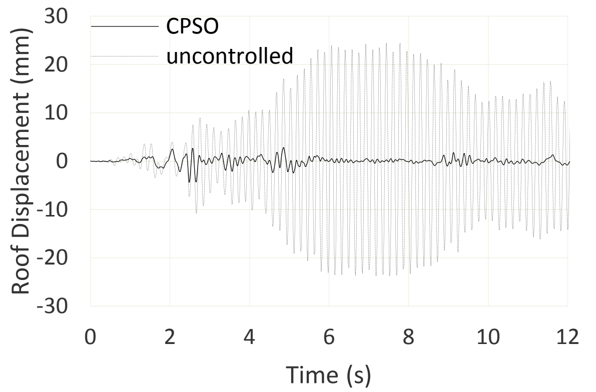


Fig. 5 CPSO-controlled vs. uncontrolled displacement for 3rd floor under El Centro (1940) earthquake ($[R] = 10^{-2}, [Q] = 10^5 \times I_{6\times6}$)

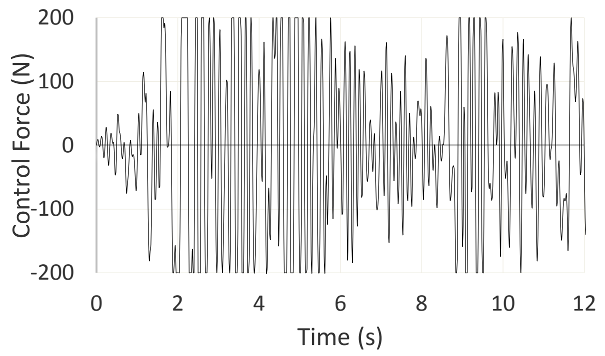


Fig. 6 CPSO Control force applied to the structure under El Centro (1940) earthquake ($[R] = 10^{-2}, [Q] = 10^5 \times I_{6\times6}$).

Table 2 Reduction factor of responses of a three-story building by CPSO under earthquake excitation

Earthquake excitation	Reduction factor of response (percent)	
	Displacement	Velocity
California (1952)	81.95	85.00
Northridge (1994)	71.33	77.27
El Centro (1940)	77.46	82.20

Table 3 Reduction factor of the responses of a three-story building by LPNN under earthquake excitation (Kim *et al.* 2008)

Earthquake excitation	Reduction factor of response (percent)	
	Displacement	Velocity
California (1952)	68.03	66.74
Northridge (1994)	65.44	59.47
El Centro (1940)	60.36	59.18

iterations, the parameters c_1 and c_2, and the weight coefficient w are assumed to be equal to 8, 20, 2, 2, 1, respectively. As aforementioned, the minimum and maximum allowable force of tendon system, which is applied to the structure is limited 200 N. All dynamical characteristics and controller properties are assumed to be identical to the literature (Kim *et al.* 2008).

The controlled and uncontrolled responses of the 3rd floor, under El Centro (1940) earthquake (Fig. 4) are shown in Fig. 5. It can be seen that the response of the third floor is significantly reduced. The reduction factor of the displacement of the 3rd floor is approximately equal to 77%, using CPSO method.

Moreover, the reduction factor of the structural responses for California (1952) and Northridge (1994) earthquakes are shown in Table 2. The table illustrates that CPSO algorithm is able to control the structural responses more effectively: not only reduces the responses up to a specific amount, but limits the amount of the control force in a range of [-200, 200]. It is easy to increase the amount of the reduction in all responses by ignoring the limitations in control force, which may lead to an unreal maximum control force. However, all results are produced using the previously mentioned limitation of the control force. Fig. 6 shows the time history of the control force under El Centro (1940) earthquake.

PSO-controlled displacement force of the roof under Northridge (1994) earthquake is presented in Fig. 7. The reduction factors of the responses, using LPNN (Kim *et al.* 2008) are also shown in Table 3. Besides, the time history of control force, which is computed by CPSO, is presented in Fig. 8. As it was mentioned before, there is no data regarding the control force history of LPNN method. As a result, it seems to be a good idea to compare the history of control forces of the two methods.

Central Processing Unit (CPU) of computer needs time to execute CPSO algorithm and compute the best controlling signal. Therefore, there will be a time delay in each step. In the current study, the simulation has been done in an eight-physical-core computer using automatic parallel configuration. The normalized CPU times (with respect to sampling time step Δt) for both El Centro (1940) and Northridge (1994) earthquakes are depicted in Figs. 9 and 10. The figures show a considerable delay in the first step of the execution. Besides, there are some regions, where the time delay increases temporarily. The first time delay is clear because of the time, which is required to load the computer program and its modules in Random Access Memory (RAM). However, the other delays may be caused

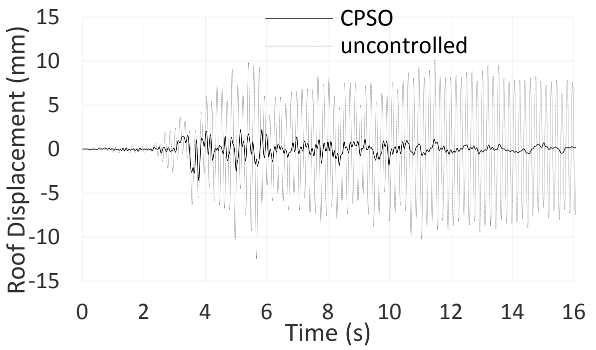


Fig. 7 CPSO-Controlled vs. uncontrolled displacement of a three-story building under Northridge (1994) earthquake ($[R]=10^{-2}, [Q]=10^{5}\times I_{6\times 6}$)

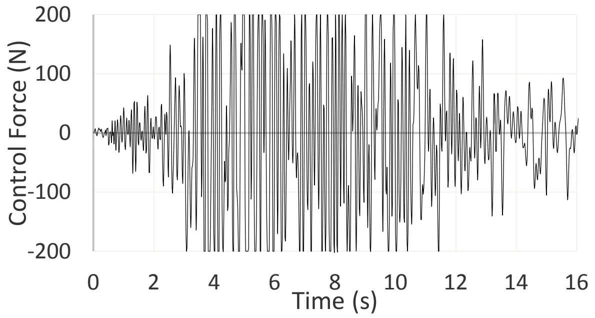


Fig. 8 CPSO Control force applied to the structure under Northridge (1994) earthquake ($[R]=10^{-2}, [Q]=10^{5}\times I_{6\times 6}$)

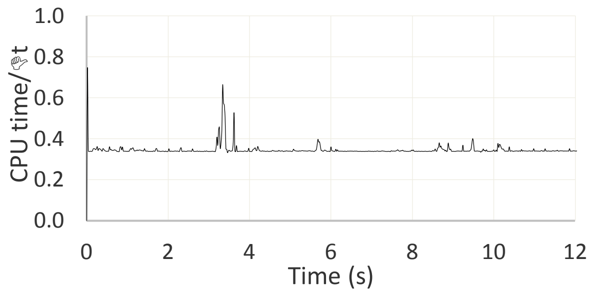


Fig. 9 CPU time (normalized to sampling time step Δt=0.02) under El Centro (1940) earthquake. Average CPU time: 0.007s

by interrupt signals sent by other services, running at the background. As a result, it is recommended to disable any redundant service before executing the simulation. Nevertheless, the results have been presented untouched to demonstrate the effect of background services and interrupts.

4.2 Ten-story shear building

The second example is also selected from the literature of active control to compare the CPSO algorithm with Block Pulse Functions method (Ghaffarzadeh and Younespour 2014). A ten-story shear building has been

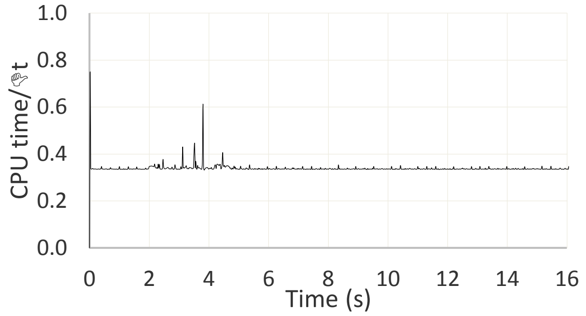


Fig. 10 CPU time (normalized to sampling time step Δt=0.02) under Northridge (1994) earthquake. Average CPU time: 0.006s

Table 4 Structural characteristics of 10-storyshear building (Ghaffarzadeh and Younespour 2014)

Story	Mass $(\times 10^{3}\ Kg)$	Stiffness $(\times 10^{5}\ N/m)$
1-3	105	1700
4-6	95	1600
7-9	90	1400
10	85	1100

controlled using active tendon controllers under Loma Prieta (1989) earthquake, which has a time sampling of 0.005s. The structural characteristics of the example are presented in Table 4. The $[C]$ matrix is computed utilizing Rayleigh damping assumption.

It is valuable to note that Ghaffarzadeh and Younespour (2014) have equipped all ten stories with active tendons. Therefore, CPSO must solve a 10-dimensional optimization problem as fast as possible. Again, in this study, the control forces have been assumed to fall in $[-85KN, +85KN]$. The displacement response of the roof and the corresponding control force are depicted in Figs. 11 and 12. The figure shows that CPSO algorithm has an efficiency of 65% that is

$$\frac{Uncontrolled\ Response - Controlled\ Response}{Uncontrolled\ Response}\times 100 = \frac{16.8-5.8}{16.8} \cong 65\% \tag{16}$$

It seems that CPSO algorithm is comparable to Block Pulse Functions (BPFs) method, regarding the amount of reduction to its counterpart, which is approximately equal to 55% for Loma Prieta (1989) earthquake. In a mathematical point of view, the ten-story example, which is equipped with ten active tendons, is a ten-dimensional optimization problem, where the best forces of each tendon should be computed so that the function, defined in Eq. (17), is minimized. As a result, it seems that CPSO control algorithm has performed better in this example.

The time history of CPU time, which is required for finding the optimal control forces, is depicted in Fig. 12. Owing to the fact that the sampling time step of Loma Prieta (1989) earthquake is equal to 0.005s, it can be seen that the CPSO algorithm requires approximately 0.077s to compute the best control force, by utilizing an eight-

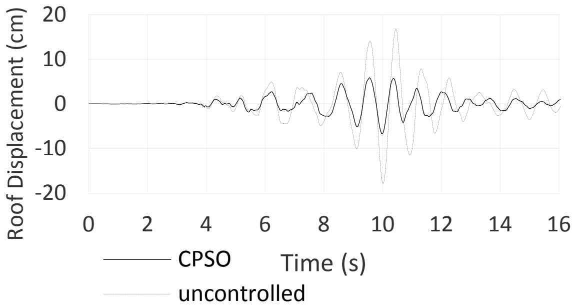


Fig. 11 CPSO-Controlled vs. uncontrolled displacement of a 10-story building under Loma Prieta (1989) earthquake ($[R]=10^{-3}\times I_{10\times10}, [Q]=10^{5}\times I_{20\times20}$)

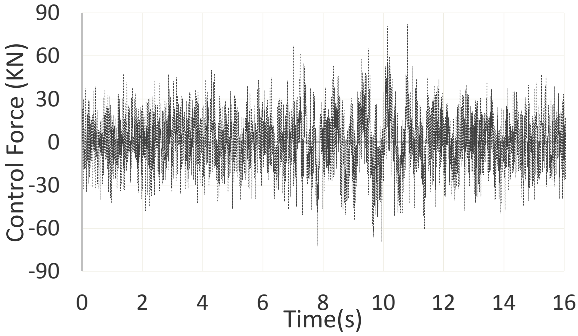


Fig. 12 CPSO Control force applied to the structure under Loma Prieta (1989) earthquake ($[R]=10^{-3}\times I_{10\times10}$, $[Q]=10^{5}\times I_{20\times20}$)

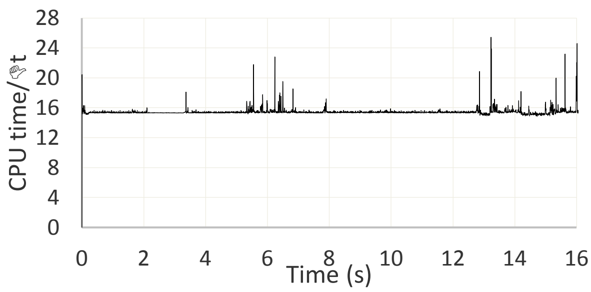


Fig. 13 CPU time (normalized to sampling time step Δt=0.005) under Loma Prieta (1989) earthquake. Average CPU time: 0.077s

physical-core CPU and parallel configuration. In a technical point of view, because of a considerable lag, the control algorithm could not be considered as a real-time method. The main reasons of high normalized delay, shown in Fig. 12, are:

1) the dimension of the search space, which is equal to ten, and

2) the time sampling of the earthquake, which is very small (0.005s).

It is important to emphasize that Ghaffarzadeh and Younespour (2014) have not provided the CPU time of their BPFs method; therefore, the time delays of both method could not be compared. Although in this example, each of 10 stories has one active tendon, in practical situations, only some stories of a building are equipped with such devices. Consequently, the required CPU time may decrease because of the reduction in the number of control forces, which are being optimized.

The aforementioned issue may be resolved by utilizing Kalman filter as presented in Eqs. (7)-(12). In such situations, a full state measurement is needed. More technical details have been provided in (Yuen *et al*. 2013, Lei *et al*. 2015).

Table 5 Structural characteristics of 20-story shear building

Story	Mass ($\times 10^3\ Kg$)	Stiffness ($\times 10^8\ N/m$)	Damping ($\times 10^5\ N.s/m$)
1-5	120	1.35	12.42
6-11	100	1.25	11.50
12-20	90	1.15	10.58

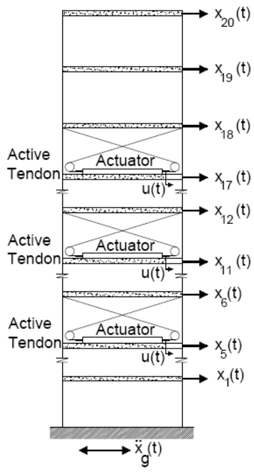


Fig. 14 Twenty-story shear building equipped with three active tendon in 6th, 12nd and 18th stories

4.3. Twenty-story shear building

In the previous sections, the results of CPSO control algorithm were compared with two examples from prior researches. In the current section, a twenty-story shear building is considered as the third example, which is shown in Fig. 14. Although the best location of active tendons is important, it may be defined as another optimization problem. Therefore, for simplicity, it has been assumed that the building is equipped with three active tendons in 6th, 12nd and 18th stories. The maximum value of each tendon force is limited to 200KN. The structural characteristics of the example are given in Table 5. To excite the structure, El Centro (1940) earthquake is applied to the building. The time history of displacement of the 20th story and corresponding control force are depicted in Figs. 15 and 16. The amount of the reduction factor of the displacement response of 20th floor is presented in Eq. (20).

$$Reduction\ Factor=\frac{24.7-7.7}{24.7}\cong 68\% \qquad (17)$$

Fig. 17 shows the time history of the CPU time during

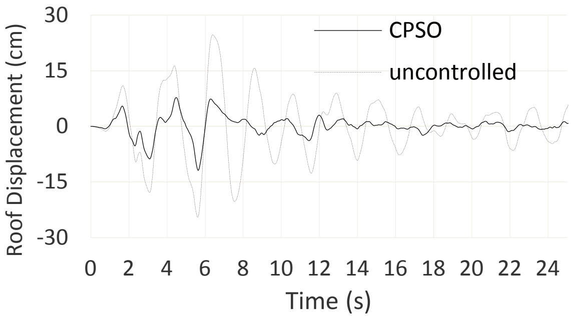


Fig. 15 CPSO-Controlled vs. uncontrolled displacement of a 20-story building under El Centro (1940) earthquake ($[R] = 10^{-3} \times I_{3\times3}, [Q] = 10^{4} \times I_{40\times40}$)

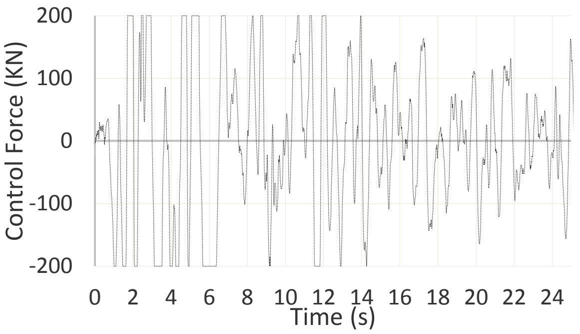


Fig. 16 CPSO Control force applied to the structure under El Centro (1940) earthquake ($[R] = 10^{-3} \times I_{3\times3}, [Q] = 10^{4} \times I_{40\times40}$)

the control process. The figure shows that the control algorithm approximately has a 1.1Δ*t* delay. The comparison between the second and third examples shows that the optimization phase has a profound impact on the time delay. As a result, it seems to be a good idea to reduce the number of controller devices in the second example from ten to two, and optimize the location of each device to achieve the best results.

5. Conclusions

In this paper, some comparisons are represented from the literature of numerical optimization to show the efficiency of Chaotic-PSO (Logistic map) in comparison to PSO. Owing to the fact that CPSO slightly beats original PSO, it is used to compute optimal control force, and reduce the response of shear buildings under earthquake excitation. Each combination of control forces, which is called "particle", is randomly generated in a predetermined interval $[u_{c,min}, u_{c,max}]$, where $u_{c,min}$ and $u_{c,max}$ are the minimum and maximum of control force. Consequently, by using an iterative procedure and a chaotic map (Logistic), the best values among all particles, which minimizes the fitness function, is found and applied to the corresponding degree of freedom of shear building as a control force at each time step.

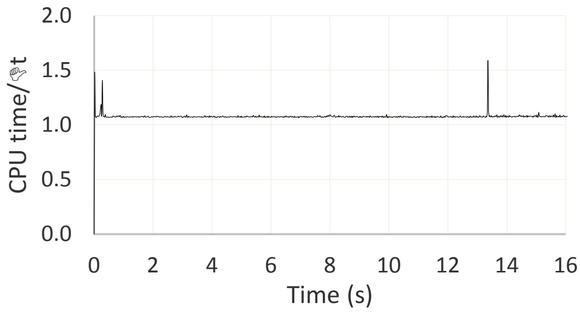


Fig. 17 CPU time (normalized to sampling time step Δ*t*=0.02) under El Centro (1940) earthquake. Average CPU time: 0.022s

Unlike the Genetic Algorithm (GA), CPSO does not have many parameters or operators. Thus, it is simple and convenient especially when the reduction of computation time is important. Three examples are presented and controlled using CPSO algorithm. Two examples out of three are taken from previous works to compare the efficiency of the method. The last example demonstrates the performance of the control algorithm in a more realistic situation. It is shown that CPSO algorithm can effectively find the best control force and be used in controlling the structural response under earthquake excitation. It is also shown that CPSO-based optimal active control probably have better overall performance in comparison to Lattice type Probabilistic Neural Network (LPNN) and Block Pulse Functions (BPFs) methods, especially when the amount of control force is important and must be limited to a certain value. It may be a good challenge to compare the reduction factor, the time history of control force, and the CPU time of CPSO, LPNN and BPFs methods under the same conditions. Although all above-mentioned algorithms could be consider as an alternative of optimal active control, the CPU time and the delay caused by computational effort should be carefully considered to find out whether or not the method is capable of being used in a real-time control scenario. A full state measurement would be necessary in case of considerable time delay.

References

Altinoz, O.T., Yilmaz, A.E. and Weber, G.W. (2010), "Chaos Particle Swarm Optimized PID Controller for the Inverted Pendulum System", *2nd International Conference on Engineering Optimization*, Lisbon.

Block, J. and Gilliatt, H. (1997), "Active control of an aeroelastic structure", *AIAA Meeting Papers on Disc*, 1-11.

Cheng, Y.M., Li, L., Chi, S. and Wei, W.B. (2007), "Particle swarm optimization algorithm for the location of the critical non-circular failure surface in two-dimensional slope stability analysis", *Comput. Geotech.*, **34**, 92-103.

Cho, H.C., Fadali, M.S., Saiidi, M.S. and Lee, K.S. (2005), "Neural network active control of structures", *Int. J. Control, Autom. Syst.*, **3**(2), 202-210.

Chuang, L.Y., Tsai, S.W. and Yang, C.H. (2011), "Chaotic catfish particle swarm optimization for solving global numerical optimization problems", *Appl. Math. Comput.*, **217**(16), 6900-

6916.

Chuanwen, J. and Bompard, E. (2005), “A self-adaptive chaotic particle swarm algorithm for short term hydroelectric system scheduling in deregulated environment”, Energy Conversion and Management, **46**, 2689-2696.

de Oliveira, L.P.R., Janssens, K., Gajdatsy, P., Van der Auweraer, H., Varoto, P. S., Sas, P. and Desmet, W. (2009), “Active sound quality control of engine induced cavity noise”, *Mech. Syst. Signal Pr.*, **23**, 476-488.

Dyke, S., Spencer, B., Jr., Quast, P., Sain, M., Kaspari, D. Jr. and Soong, T. (1996), “Acceleration feedback control of MDOF structures”, *J. Eng. Mech.*, ASCE, **122**(9), 907-918.

François, M., Defourm, D. and Negre, C. (2014), “A fast chaos-based Pseudo-Random bit generator using binary64 floating-point arithmetic”, *Informatica*, **38**, 115-124.

Ghaffarzadeh, H. and Younespour, A. (2014), “Active tendons control of structures using block pulse functions”, *Struct. Control Hlth. Monit*., **21**(12), 1453-1464.

Ghanbarpor, Y.R. and Mohebbi, M. (2006), “Distributed genetic algorithms based control method for reducing structural responses under seismic exitation”, *International Conference on Computing and Decision Making in Civil and Building Engineering*, Montreal.

Hu, X. and Eberhart, R.C. (2002) “Adaptive particle swarm optimization: detection and response to dynamic systems”, *IEEE*, **2**, 1666-1670.

Jabbari, F., Schmitendorf, W.E. and Yang, J.N. (1995), “H∞ control for seismic-excited buildings with acceleration feedback”, *J. Eng. Mech.*, ASCE, **121**(9), 994-1002.

Kalman, R.E. (1960), “A new approach to linear filtering and prediction problems”, *J. Basic Eng.-T.*, ASME, **82**(1), 35-45.

Kennedy, J. and Eberhart, R.C. (1995), “Particle swarm optimization”, *IEEE Neural Networks Council, Proceedings of IEEE International Conference on Neural Networks*, **1**, 1942-1948.

Kerboua, M., Benguediab, M., Megnounif, A., Benrahou, K.H. and Kaoulala, F. (2014), “Semi active control of civil structures, analytical and numerical studies”, *Phys. Procedia*, **55**, 301-306.

Kim, D.H., Kim, D., Chang, S. and Jung, H. (2008), “Active control strategy of structures based on lattice type probabilistic neural network”, *Probab. Eng. Mech.*, **23**, 45-50.

Kim, Y.J. and Ghaboussi, J. (2007), “A new genetic algorithm based control method using state space reconstruction”, *Proc. Second World Conf. Struct. Control*, Kyoto, Japan.

Kobori, T. and Minai, R. (1960), “Analytical study on active seismic response control”, *Arch. Inst. JPN*, **66**, 257-260.

Kuo, D. (2005), “Chaos and its computing paradigm”, *IEEE Poten. Mag.*, **24**(2), 13-15.

Lei, Y., Hua, W., Luo, S. and He, M. (2015), “Detection and parametric identification of structural nonlinear restoring forces from partial measurements of structural responses”, *Struct. Eng. Mech.*, **54**(2), 291-304.

Li, Q.S., Liu, D.K., Leung, A.Y.T., Zhang, N. and Luo, Q.Z. (2002), “A multilevel genetic algorithm for the optimum design of structural control systems”, *Int. J. Numer. Meth. Eng.*, **55**(7), 817-834.

Liu, B., Wang, L., Jin, Y.H., Tang, F. and Huang, D.X. (2005), “Improved particle swarm optimization combined with chaos”, *Chaos Solit. Fract.*, **25**, 1261-1271.

Maurice, C. (2006), *Particle Swarm Optimization*, Chippenham, Antony Rowe Ltd.

Fisco, N.R. and Adeli, H. (2011), “Smart structures: Part I-active and semi-active control”, *Scientia Iranica*, **18**(3), 275-284.

Shayeghi, A., Shayeghi, H. and Kalasar, H.E. (2009), “Application of PSO technique for seismic control of tall building”, *Int. J. Intel. Syst. Technol.*, **3**(4), 1116-1123.

Shayeghi, H., Jalili, A. and Shayanfar, H.A. (2008), “Multi-stage fuzzy load frequency control using PSO”, *Energy Convers. Manage.*, **49**, 2570-2580.

Shi, Y. and Eberhart, R. (1998), “A modified particle swarm optimizer”, *1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on*, IEEE.

Stojanovski, T. and Kocarev, L. (2001), “Chaos-based random number generators-Part I: analysis”, *IEEE Tran. Circuit. Syst.-I: Fundam. Theo. Appl.*, **48**(3), 281-288

Tani, A., Kawamura, H. and Ryu, S. (1998), “Intelligent fuzzy optimal control of building structures”, *Eng. Struct.*, **20**(3), 184-192.

Wu, B. and Bodson, M. (2004), “Multi-channel active noise control for periodic sources-indirect approach”, *Automatica*, **40**(2), 203-212.

Yang, C.H., Tsai, S.W., Chuang, L.Y. and Yang, C.H. (2012), “An improved particle swarm optimization with double-bottom chaotic maps for numerical optimization”, *Appl. Math. Comput.*, **219**(1), 260-279.

Yang, J.N., Akbarpur, A. and Ghaemmaghami, P. (1987), “Instantaneous optimal control laws for tall buildings under seismic excitations”, NCEER Technical Report 87-0007, National Center for Earthquake Engineering, Buffalo.

Yuen, K.V., Liang, P.F. and Kuok, S.C. (2013). “Online estimation of noise parameters for Kalman filter”, *Struct. Eng. Mech.*, **47**(3), 361-381.

PL