Parameter Estimation Using a CLPSO Strategy

H. Tang, W. Zhang, C. Fan, and S. Xue

Abstract—**As a novel evolutionary computation technique, particle swarm optimization (PSO) has attracted much attention and wide applications for solving complex optimization problems in different fields mainly for various continuous optimization problems. However, it may easily get trapped in a local optimum when solving complex multimodal problems. This paper utilizes an improved PSO by incorporating a comprehensive learning strategy into original PSO to discourage premature convergence, namely CLPSO strategy to estimate parameters of structural systems, which could be formulated as a multi-modal optimization problem with high dimension. Simulation results for identifying the parameters of a structural system under conditions including limited output data and no prior knowledge of mass, damping, or stiffness are presented to demonstrate the effectiveness of the proposed method.**

I. Introduction

In system identification, considerable effort have been invested in developing methods for identification of system models and their parameters. Currently, a wide range of analytical techniques exists for linear or nonlinear systems. For civil-engineering systems, limited progress have been made with analytical methods for complexity and incomplete prior information. Instead, some success have been achieved with various traditional optimization algorithms. Instead, some successes have been achieved with various intelligent optimization algorithms. Evolution strategy (ES) algorithms have been presented for the identification of multiple degree of freedom (DOF) systems [1]. Perry *et al.* [2] have presented a modified GA to identify structural systems. GAs have been used to solve the global system identification problem in shear-type building structures [3][4][5] .

As a novel evolutionary computation technique, Particle swarm optimization (PSO) has attracted much attention and wide applications, owing to its simple concept, easy implementation and quick convergence [6]. PSO has been successfully applied in many areas, such as function optimization, artificial neural network training, fuzzy system control, simulation and identification, structural reliability assessment, automatic target detection, optimal design and parameters estimation [7]-[14], to name a few. Most of the engineering applications that make use of swarm intelligence computation techniques for optimization purposes employ PSO, which is probably the most powerful alternative among the techniques available.

The original PSO algorithm is easy to implement and has been shown to perform well on many optimization problems. However, it may easily get trapped in a local optimum when solving complex multimodal problems. In order to improve PSO's performance, an improved PSO by incorporating a comprehensive learning strategy into the original PSO is presented to enhance the convergence properties [15]. The new strategy makes the particles have more exemplars to learn from and a larger potential space to fly. This strategy enables the diversity of the swarm to be preserved to discourage premature convergence. Another attractive property of the CLPSO is that it does not introduce any complex operations to the original simple PSO framework. The only difference from the original PSO is the velocity update equation. Huang *et al.* [16] have presented a comprehensive learning optimizer to handle multiple objective optimization problems. More detailed literature reviews on different variants PSO can be found in [12].

The problem of system identification is an inverse problem of difficult solution. Currently, difficulties lie in the development of algorithms that use incomplete measured data from the system to characterize it without significant a priori knowledge of the system. In this study, a parameter estimation technique based on CLPSO is presented to overcome some of the difficulties encountered in the field, which could be formulated as multimodal numerical optimization problems with high dimension. Some numerical examples are presented from which the effectiveness and efficiency of the CLPSO is investigated.

II. Problem Formulation

The basic idea in system identification is to compare the time dependent response of the system and a parameterized model by a norm or some performance criterion giving a measure to how well the model response fits the system response. Hence, the objective is to find a set of parameters that minimize the prediction error between system output $y(t)$, i.e., the measured data, and model output $\hat{y}(\boldsymbol{\theta},t)$ at each time-step t.

Therefore, our interest lies in minimizing the predefined error norm of the outputs, e.g., the following mean square

This work was supported in part by the Nature Science Foundation of China under Grant 50708076 and SRF for ROCS, SEM of China.

H. Tang is with the Research Institute of Structural Engineering and Disaster Reduction, Tongji University, Shanghai 200092 China (phone: 86-21-65982390; fax: 86-21-65983410; e-mail: thstj@mail.tongji.edu.cn).

W. Zhang is with the Research Institute of Structural Engineering and Disaster Reduction, Tongji University, Shanghai 200092 China (e-mail: zw.myid@gmail.com).

C. Fan is with the Department of civil engineering, Suzhou University of science and technology, Suzhou 215011 China (e-mail: fcxzf@pub.sz.jsinfo.net).

S. Xue is with the Department of Architecture, Kinki University, Osaka 577-0056 Japan (e-mail: xue@arc.kindai.ac.jp).

978-1-4244-1823-7/08/$25.00©2008 IEEE

error (MSE) function.

$$f(\boldsymbol{\theta}) = \frac{1}{T}\sum_{t=1}^{T}\left\|\boldsymbol{y}(t) - \hat{\boldsymbol{y}}(t)\right\|^2 \quad (1)$$

where $\|\cdot\|$ represents the Euclidean norm of vectors. Formally, the optimization problem requires finding a vector $\boldsymbol{\theta}^* \in R^n$, so that a certain quality criterion is satisfied, namely that the error norm *f*(*) is minimized. The function *f*(*) is commonly called a fitness function or objective function. In CLPSO, typically an objective function is used which reflects the goodness of solution. The identification problem thus is treated as a linearly constrained multi-dimensional optimization problem, namely

$$\text{Minimize } f(\boldsymbol{\theta}), \boldsymbol{\theta} = (\theta_1, \theta_2, \cdots, \theta_n)^T$$

$$s.t.\ \ \boldsymbol{\theta} \in S,\quad S = \{\boldsymbol{\theta} : \theta_{\min,i} \le \theta_i \le \theta_{\max,i},\quad \forall i = 1,2,\cdots,n\} \quad (2)$$

where $f(\boldsymbol{\theta})$ =objective function which maps decision variable $\boldsymbol{\theta}$ into the objective space $f = R^n \to R$, S is the *n*-dimensional feasible search space, $\theta_{\max}$ and $\theta_{\min}$ denote the upper bounds and the lower bounds of the *n* parameters respectively.

III. CLPSO STRATEGY

PSO algorithms draw inspiration from the social behavior of groups of individuals as observed in nature, like insects or birds. These animals are able to explore the space in search of a common goal (e.g. food) by moving in a coordinate and competitive manner [6]-[9]. In PSO, candidate solutions of a population, called particles, coexist and evolve simultaneously based on knowledge sharing with neighboring particles. While flying through the problem search space, each particle generates a solution using directed velocity vector. Each particle modifies its velocity to find a better solution (position) by applying its own flying experience (i.e. memory having best position found in the earlier flights) and experience of neighboring particles (i.e. best-found solution of the population). The *d*th dimension of the *i*th particles update their positions θ_i^d and velocities v_i^d as shown below:

$$v_i^d \leftarrow w * v_i^d + ac_1 * r_1 * (pbest_i^d - \theta_i^d) + ac_2 * r_2 * (gbest^d - \theta_i^d) \quad (3)$$

$$\theta_i^d \leftarrow \theta_i^d + v_i^d \quad (4)$$

where $\boldsymbol{\theta}_i = (\theta_i^1, \theta_i^2, \cdots, \theta_i^D)$ is the position of the *i*th particle; $\boldsymbol{v}_i = (v_i^1, v_i^2, \cdots, v_i^D)$ represents velocity of particle *i*. $\boldsymbol{pbest}_i = (pbest_i^1, pbest_i^2, \cdots, pbest_i^D)$ is the best previous position yielding the best fitness value for the *i*th particle; and $\boldsymbol{gbest} = (gbest^1, gbest^2, \cdots, gbest^D)$ is the best position discovered by the whole population. ac_1 and ac_2 are the acceleration constants reflecting the weighting of stochastic acceleration terms that pull each particle toward *pbest* and *gbest* positions, respectively. r_1 and r_2 are two independently uniformly distributed random numbers in the range [0, 1]. w is the particle inertia weight. Our approach incorporates a linearly time-decreasing inertia weight over the course of search.

In the original PSO, each particle learns from its *pbest* and *gbest* simultaneously. Restricting the social learning aspect to only the *gbest* makes the original PSO converge fast. However, because all particles in the swarm learn from the *gbest* even if the current *gbest* is far from the global optimum, particles may easily be attracted to the region and get trapped in a local optimum if the search environment is complex with numerous local solutions. Mendes *et al*. [17] and Liang *et al.* [12][15] proposed a novel learning strategy where all particles' historical best information is used to update a particle's velocity. This strategy ensures that the diversity of the swarm is preserved to discourage premature convergence. The following velocity updating equation is used in this learning strategy:

$$v_i^d \leftarrow w * v_i^d + c * r * (pbest_{f_i(d)}^d - \theta_i^d) \quad (5)$$

where $\boldsymbol{f}_i = (f_i(1), f_i(2), \cdots, f_i(D))$ defines which particles' *pbests* the particle *i* should follow. $pbest_{f_i(d)}^d$ can be the corresponding dimension of any particle's including its own *pbest*, and the decision depends on probability *Pc*, referred to as the learning probability[18], which can take different values for different particles. The learning proportion, *Pc*, determines how many dimensions are chosen to learn from other particles' *pbests*. For each dimension of particle *i*, a random number is generated. If this random number is larger than Pc_i, the corresponding dimension will learn from its own *pbest*; otherwise it will learn from another particle's *pbest*. In the strategy, *m* dimensions are randomly chosen to learn from the *gbest*. Some of the remaining *D-m* dimensions are randomly chosen to learn from some randomly chosen particles' *pbest* and the remaining dimensions learn from its *pbest*. These operations increase the particles' initial diversity and enable the swarm to overcome premature convergence problem. The refreshing gap parameter needs be tuned. In our experiences, better results were obtained when the refreshing gap is set at 0.6-0.8 times of the dimension. A tournament selection procedure of the CLPSO is shown in Table 1.

TABLE I
THE PSEUDO CODE FOR TOURNAMENT SELECTION

```
For i =1 to NP   /NP is the population size/
  rc = randperm(D)  /random permutation of D integers/
  ai(rc (1:m)) = 1
  fi1 = ceil(NP*rand(1,D))  /ceiling operate/
  fi2 = ceil(NP *rand(1,D))  /ceiling operate/
  fi = (pbest (fi1)<pbest(fi2)) *fi1+(pbest(fi1)>=pbest(fi2))*fi2
                                                  /tournament selection/
  bi = ceil(rand(1,D)-1+Pci)  /ceiling operate/
  fi=bi*fi+(1-bi)*fi
EndFor i
```

IV. SIMULATION RESULTS

For structural identification problems it is common that the mass of the structure is assumed to be known and the identification aims to identify structural stiffness properties. In this paper this common problem is considered and extension is made to the much more difficult case of unknown mass systems, where the mass, stiffness and damping of the structure are to be identified.

In order to assess the effectiveness of the parameter estimation technique with the CLPSO presented above, numerical simulations of a five degree of freedom (DOF) structure system are carried out. The properties of the structural systems are given in Table 2. It is assumed that the system is excited by known forces and that the response of the structure, in terms of accelerations, is recorded at some given points. In the "full output" scenario, measurements at all DOFs are available, whereas in the second "partial output" scenario, only DOFs 2, 4, and 5 are available.

TABLE II
SYSTEM PARAMETERS

Stiffness(kN/m)		
k_1	k_2	k_3-k_5
2.485e5	1.921e5	1.522e5
Mass(kg)		
m_1-m_2	m_3-m_5	
2.762e3	2.300e3	
Damping(kN·s/m)		
c_1	c_2	c_3-c_5
3.129e3	2.536e3	2.030e3

The dynamic equation of motion of a structural system can be written as

$$\boldsymbol{M}\ddot{\boldsymbol{x}}(t)+\boldsymbol{C}\dot{\boldsymbol{x}}(t)+\boldsymbol{K}\boldsymbol{x}(t)=\boldsymbol{u}(t) \quad (6)$$

where **M**, **C** and **K** are the mass ($m_1,\ldots,m_5$), damping($c_1,\ldots,c_5$) and stiffness ($k_1,\ldots,k_5$) matrices, $\boldsymbol{x}$ is the displacement vector and $\boldsymbol{u}$ is the input force vector.

Therefore, the system is fully described by the set of parameters

$$\boldsymbol{\theta}=(m_1,\cdots,m_5,k_1,\cdots,k_5,c_1,\cdots,c_5) \quad (7)$$

The influence of limited availability of measurements on the performance of CLPSO for parameters estimation is discussed. In this experimental study, the following cases of data availability will be treated here as:

• Case 1: Full set of the accelerations (outputs) available

$$\boldsymbol{y}(t)=(\ddot{x}_1(t),\ddot{x}_2(t),\ddot{x}_3(t),\ddot{x}_4(t),\ddot{x}_5(t)) \quad (8)$$

• Case 2: Partial set of the outputs available

$$\boldsymbol{y}(t)=(\ddot{x}_2,\ddot{x}_4,\ddot{x}_5) \quad (9)$$

In simulation test, parameters are set as follows: swarm size n = 20, maximum evolution generation = 500 (stopping condition), ac_1 = 2.8, ac_2 = 1.3, c=1.494, m=0.8D, w = linearly time-decreasing values from 0.9 to 0.7. The search space is defined 0.5-2 times actual values. We empirically developed the following expression to set a Pc_i value for each particle:

$$Pc_i=0.05+0.45\frac{\exp\left(\frac{5(i-1)}{NP-1}\right)-1}{\exp(5)-1} \quad (10)$$

where NP is the population size and i is the particle's ID.

The statistical simulation results of 20 independent runs for the example with the usage of the PSO and CLPSO methods are shown in Tables 3-4. In addition, we present typical simulation results (including the convergent processes of objective value and all parameters) for the example with Figs. 1-8.

From Table 3-4, in the "full output" case, it can be seen that the results obtained by both PSO and the CLPSO are very close to the true values. Nevertheless, the average and variance results obtained by CLPSO greatly outperform those obtained by the PSO. In the second "partial output" case, the values of estimated parameters obtained by CLPSO are still very close to the true values of original parameters. The CLPSO seems to be more powerful in escaping local optima and in search for the global optimum on more complex problems.

From Figs. 1-6, we observe that for the "partial output" case, the PSO seemed to have more difficulty locating the solution than the CLPSO. This is because that the CLPSO has a large potential search space using a comprehensive learning strategy. This new strategy ensures that the diversity of the swarm is preserved to discourage premature convergence.

TABLE III
STATISTICAL RESULTS OF PARTIAL OUTPUT SCENARIO

Par.	True value	Estimation by PSO		Estimation by CLPSO	
		Mean	Var.	Mean	Var.
k_1	2.485e5	2.410e5	4.478e3	2.488e5	1.983e2
k_2	1.921e5	1.936e5	0.526e3	1.922e5	0.373e2
k_3-k_5	1.522e5	1.511e5	0.755e3	1.525e5	2.043e2
m_1-m_2	2.762e3	2.777e3	1.485e1	2.761e3	1.273e0
m_3-m_5	2.300e3	2.300e3	1.919e1	2.301e3	0.523e0
c_1	3.129e3	3.291e3	4.969e1	3.126e3	6.743e0
c_2	2.536e3	2.575e3	0.553e2	2.536e3	0.392e0
c_3-c_5	2.030e3	2.035e3	2.192e1	2.031e3	2.161e0

Par.=parameter, Var.=variance

TABLE IV
STATISTICAL RESULTS OF FULL OUTPUT SCENARIO

Par.	True value	Estimation by PSO		Estimation by CLPSO	
		Mean	Var.	Mean	Var.
k_1	2.485e5	2.316e5	8.754e3	2.528e5	2.653e2
k_2	1.921e5	2.063e5	2.764e3	1.924e5	2.322e2
k_3-k_5	1.522e5	1.541e5	1.443e3	1.517e5	3.795e2
m_1-m_2	2.762e3	2.710e3	3.461e1	2.774e3	1.745e1
m_3-m_5	2.300e3	2.272e3	3.772e1	2.305e3	1.206e1
c_1	3.129e3	2.939e3	9.654e1	3.148e3	9.950e0
c_2	2.536e3	2.880e3	2.838e1	2.541e3	2.850e0
c_3-c_5	2.030e3	2.051e3	2.013e1	2.026e3	8.441e0

Par.=parameter, Var.=variance

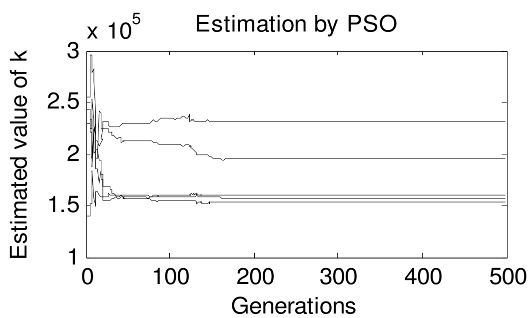

Fig. 1. One typical simulation result of *k* (kN/m) by PSO

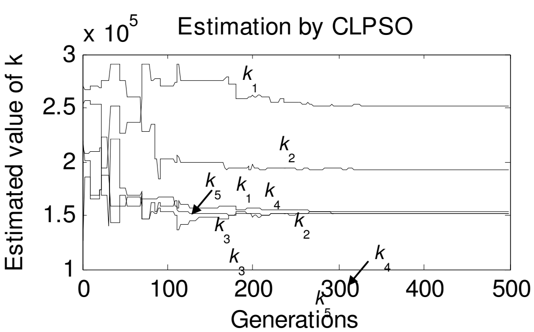

Fig. 2. One typical simulation result of *k* (kN/m) by CLPSO

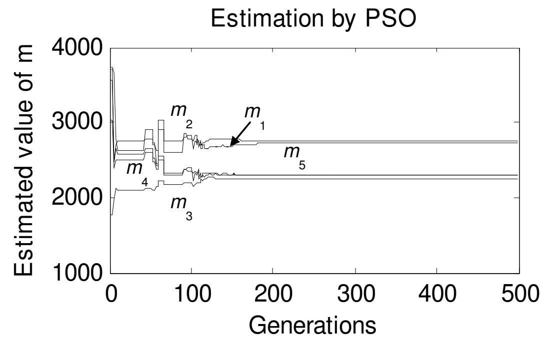

Fig. 3. One typical simulation result of *m* (kg) by PSO

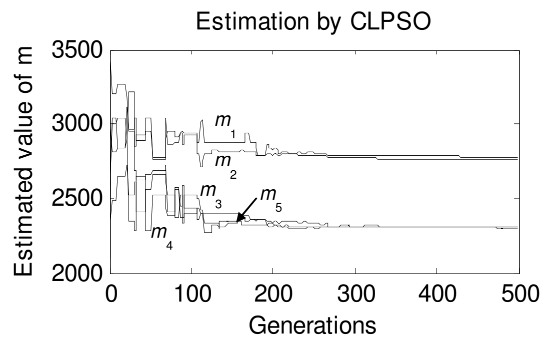

Fig. 4. One typical simulation result of *m* (kg) by CLPSO

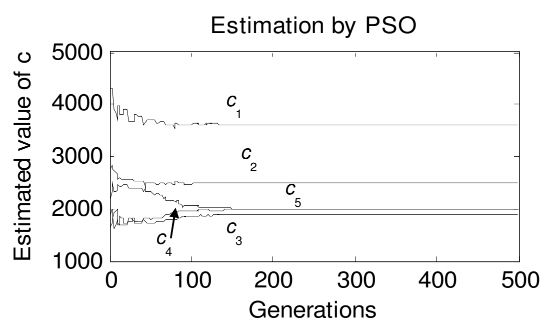

Fig. 5. One typical simulation result of *c* (kN·s/m) by PSO

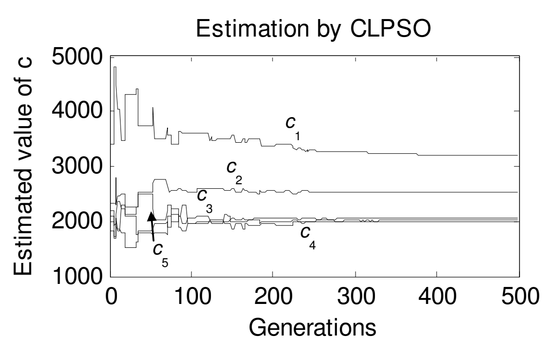

Fig. 6. One typical simulation result of *c* (kN·s/m) by CL PSO

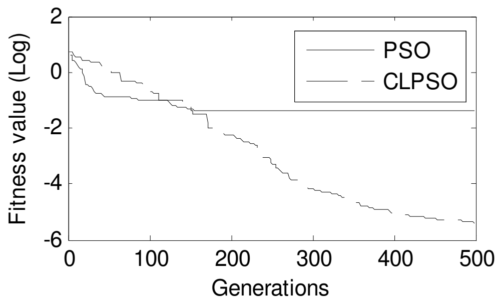

Fig. 7. One typical convergence characteristics of estimation (full measurements case)

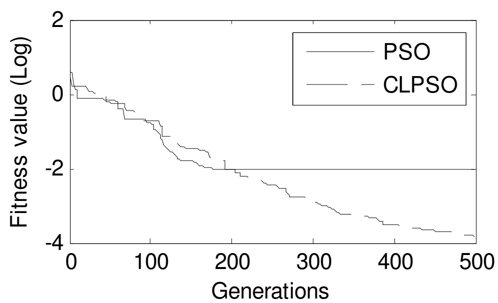

Fig. 8. One typical convergence characteristics of estimation (partial measurements case)

It can be seen in Figs.7-8 that the cost function value reached in CLPSO is very low (therefore, close to the global minimum), whereas in PSO a somewhat higher cost has been reached (further away from the global minimum). This implies that the CLPSO is more effective in solving problems with less linkage. This property is due to the PSO's dimension-wise updating rule, as well as CLPSO's learning of different dimensions from different exemplars. With the new updating rule, different dimensions may learn from different exemplars. Due to this, the CLPSO explores a larger search space than the standard PSO. Because of this, the CLPSO performs comparably to or better than original PSO on the insufficient data case in this paper.

V. CONCLUSION

This paper presents a CLPSO strategy for structural parameters estimation. This novel strategy ensures that the diversity of the swarm is preserved to discourage premature convergence. Comparative studies have been investigated to assess the applicability of the CLPSO for structural parameters estimation. From the analysis results, we observe that the CLPSO outperforms the standard PSO algorithms on no prior knowledge case, and especially significantly improves the results on partial output scenario.

ACKNOWLEDGMENT

The authors thank Prof. Suganthan of Nanyang Technological University for providing matlab codes to improve this paper.

REFERENCES

[1] G. Franco, R. Betti, and H. Lus, "Identification of Structural Systems Using an Evolutionary Strategy," *Journal of Engineering Mechanics*, vol.130, no.10, pp.1125-1139, Oct. 2004.

[2] M.J. Perry, C.G. Koh, and Y.S. Choo, "Modified genetic algorithm strategy for structural identification," *Computers & Structures*, vol.84, no.8-9, pp.529-540, Jan. 2006.

[3] C.G. Koh, B. Hong, and C.Y. Liaw, "Parameter identification of large structural systems in time domain," *J. Struct. Eng.*, vol.126, no.8, pp.957–963, Aug. 2000.

[4] C.G. Koh, B. Hong, and C.Y. Liaw, "Substructural system identification by genetic algorithms," In: *3rd US–Japan Workshop on Nonlinear System Identification and Structural Health Monitoring*, Los Angeles, CA., 2000.

[5] C.G. Koh, Y.F. Chen, and C.Y. Liaw, "A hybrid computational strategy for identification of structural parameters," *Comput. Struct.*, vol.81 , no.2, pp.107–117, Feb. 2003.

[6] J. Kennedy, and R.C. Eberhart, "Particle swarm optimization.," in *Proc. IEEE Int. Conf. Neural Networks*, Perth, Australia, 1995.

[7] R.C. Eberhart, Y. Shi, "Particle swarm optimization: developments, applications and resources," in *Proc. 2001 Congress Evolutionary Computation*. Piscataway, NJ: IEEE Press, 2001.

[8] J. Kennedy, R.C., Eberhart, and Y. Shi, "Swarm intelligence," San Francisco: Morgan Kaufmann Publishers, 2001.

[9] X.H. Hu, Y.H. Shi, R. Eberhart, "Recent advances in particle swarm," in *2004 IEEE Congress on Evolutionary Computation*, pp.90-97.

[10] C. Elegbede, "Structural reliability assessment based on swarm optimization," *Structural Safety*, vol.27, pp.171-186, April, 2005.

[11] M. Meissner, M. Schmuker, and G. Schneider, "Optimized Particle Swarm Optimization (OPSO) and its application to artificial neural network training," BMC *Bioinformatics*, 7:125 doi: 10.1186/1471-2105-7-125, 2006.

[12] J.J. Liang, A.K.Qin, P.N. Suganthan, and S. Baskar, "Comprehensive Learning Particle Swarm Optimizer for Global Optimization of Multimodal Functions," IEEE *Transactions on Evolutionary Computation*, vol.10, no.3, pp.281-295, June, 2006.

[13] Q. He, and L. Wang, "An effective co-evolutionary particle swarm optimization for constrained engineering design problems," *Engineering Applications of Artificial Intelligence*, vol.20, pp.89–99, Feb. 2007.

[14] H. Tang, M. Fukuda, and S. Xue, "Particle Swarm Optimization for Structural System Identification," in *6th International Workshop on Structural Health Monitoring*, Stanford University, Stanford, CA, 2007.

[15] J.J. Liang, A.K. Qin, P.N. Suganthan, and S. Baskar, "Evaluation of comprehensive learning particle swarm optimizer," Lecture Notes in Computer Science, vol. 3316. Berlin: Springer, pp. 230–235, 2004.

[16] V.L. Huang, P.N. Suganthan, J.J. Liang, "Comprehensive Learning Particle Swarm Optimizer for Solving Multiobjective Optimization Problems," *In. J. of Int. Sys.*, vol.21, no.2, pp.209–226, Feb. 2006.

[17] R. Mendes, J. Kennedy, and J. Neves, "The fully informed particle swarm: Simpler, maybe better," *IEEE Transactions on Evolutionary Computation*, vol.8, no. 3, pp. 204–210, June, 2004.

[18] V. Miranda and N. W. Oo, "New Experiments with EPSO - Evolutionary Particle Swarm Optimization," in *IEEE Symposium on Swarm Optimization*, Indianapolis, Indiana, 2006.