The Pennsylvania State University

The Graduate School

Department of Electrical Engineering

COMPARISON OF CAT SWARM OPTIMIZATION WITH PARTICLE SWARM OPTIMIZATION FOR IIR SYSTEM IDENTIFICATION

A Thesis in

Electrical Engineering

by

Jinhyun So

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science

August 2013

The thesis of Jinhyun So was reviewed and approved* by the following:

Kenneth Jenkins
Professor of Electrical Engineering
Thesis Advisor

Julio Urbina
Professor of Electrical Engineering

Victor Pasko
Professor of Electrical Engineering
Graduate Program Coordinator

*Signatures are on file in the Graduate School

ABSTRACT

Infinite impulse response (IIR) systems are widely used in modeling systems such as communications, bio-systems, acoustics, etc. Many algorithms in computational intelligence area have been developed to identify the systems with a novel search technique, but system identification is challenging due to non-unimodality of the error surface and the non-linear relationship between the error signal and the system parameters. Cat swarm optimization (CSO) was recently introduced to solve optimization problem with a new learning rule based on swarm intelligence to show better performance than particle swarm optimization (PSO). Also, it has been tried to be used for infinite impulse response (IIR) system identification. Optimum parameters are proposed to solve optimization problem. However, parameters adapted for the IIR system identification have not been investigated enough. This thesis examines the parameters of CSO in order to optimize them for IIR system identification with a few benchmarked IIR plants. Inertia-weighted PSO is used as a comparison for performance issue. The results demonstrate the better performance of the CSO than PSO.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

System identification has been considered a challenging research area due to the nonlinear structure of the system caused by noise from the natural environment. Finite impulse response (FIR) and infinite impulse response (IIR) methods are competent models for unknown system identification. In general, the IIR method is better than the FIR method for system identification since FIR method may not give a reasonable length of a filter; on the other hand, IIR has recursive characteristics and requires less amount of memory space to design complex systems. Also, most nonlinear systems have a recursive characteristic in nature. So, IIR systems can represent the models for real world system better than FIR systems. Gradient-based search and stochastic search are also competent methods in terms of optimization techniques in order to search for the global minimum in an error surface. Gradient-based optimization technique is used to estimate the gradient of an error surface. So, it has the possibility of getting trapped in a local minimum before encountering the global minimum. On the other hand, the stochastic search is used to focus on increasing the probability of encountering the global minimum. The stochastic search technique can be considered a better method since it can easily overcome the multimodal error surface problem although computational complexity increases. Also, the search method can be easily applied to cascade and parallel structure in modeling system. In the current thesis, the IIR method and stochastic search technique are employed for system identification.

In the past, many optimization algorithms in the computational intelligence area such as the genetic algorithm (GA), simulated annealing (SA), particle swarm optimization (PSO), etc., have been used for system identification. PSO adopts a population-based stochastic search with a certain learning rule for balancing the global search and local search. Cat swarm optimization

(CSO) algorithm was recently developed to show better optimization technique than PSO. The algorithm also adopts a population-based stochastic search but has a different learning rule. It has been used for various applications such as clustering and neural network optimization. The CSO has been adopted for IIR system identification but not investigated enough for its parameters. Thus, the current thesis focuses on the optimization of the parameters and suggestions for the best parameter choices for IIR system identification. The PSO algorithm is used as a comparison.

1.1 IIR System Identification

System identification shown in Figure 1.1 can be thought of as the mathematical modeling of an unknown system by monitoring its output data. A set of input data is given and the output of y(n) is mixed with a noise signal v(n). Infinite impulse response (IIR) system identification has been considered a challenging area and requires research since various applications in the area such as signal processing and communication, have adopted IIR structures. The adaptive algorithm implanted in the block diagram of IIR system identification iteratively tunes the IIR filter coefficients in order to minimize the error between the output of the unknown system and the estimated output of the IIR filter. When the function of error reaches the minimum, the adaptive filter with updated its coefficients mimics the function of unknown IIR system so an optimal model of the IIR filter is attained.

1.1.1 IIR Adaptive Filters

The relationship between input and output of the IIR adaptive filters is shown in the following Eq. 1.1.

$$y(n) = \sum_{i=0}^{n} a_i(n)x(n-i) - \sum_{i=1}^{m} b_i(n)y(n-i) \tag{1.1}$$

where $a_i(n)$ and $b_i(n)$ are the tap weights. The output $y(n)$ is dependent on weighted inputs and outputs. With the assumption of $a_0(0) = 1$, the transfer function, $H(z)$,of the adaptive IIR filter can be expressed as given in Eq. 1.2.

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_m z^{-m}}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}} \tag{1.2}$$

In this thesis, $H_{PLANT}(z)$ and $H_{AF}(z)$ denote the transfer function of the unknown plant to be identified and the transfer function of adaptive filter used, respectively.

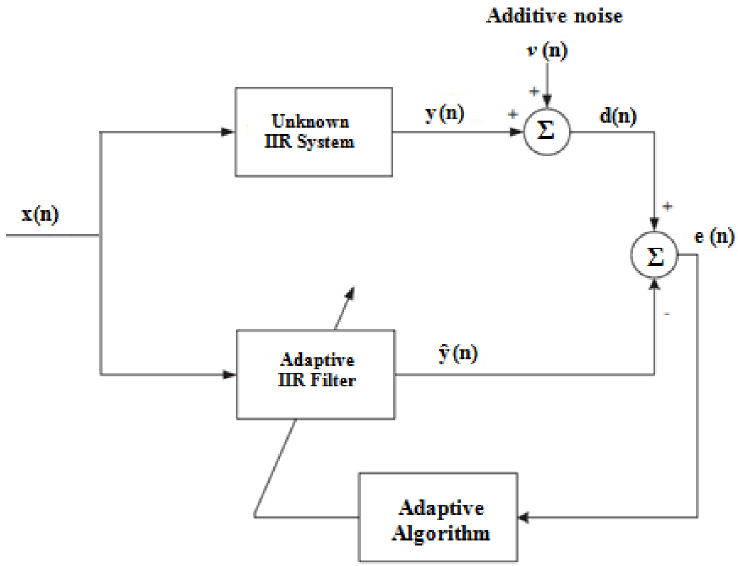

Figure 1-1 IIR System Identification Block Diagram

The main task of the system identification is to vary coefficients of $H_{AF}(z)$ by using algorithms employed in this thesis until they get as close as possible to the coefficient values of $H_{PLANT}(z)$. As a result, the mean square error (MSE), also called the cost function, converges to the minimum value. Eq. 1.5 is used as the cost function.

$$d(n) = y(n) + v(n) \tag{1.3}$$

$$e(n) = d(n) - \tilde{y}(n) \tag{1.4}$$

$$J = \mathrm{E}\left[\left(\mathrm{d}(n) - \tilde{y}(n)\right)^2\right] = \frac{1}{\mathrm{N}}\sum_{n=1}^{N} \mathrm{e}^2(n) \tag{1.5}$$

where N and $E(\cdot)$ operator denote the number of input samples and the statistical expectation, respectively. The output error method shown in Eq. 1.4 is used by the adaptive algorithm shown in Figure1.1 to adjust the coefficients of the IIR filter. The cost function, J, results in the best possible parameters of a filter to represent unknown system when it is minimized. The drawback of this method is that the coefficients are supposed to be controlled and monitored properly. Otherwise, it may be possible to cause a slow learning rate, large MSE, and the instability of the IIR filter.

1.2 Particle Swarm Optimization

Particle swarm optimization (PSO), developed by Eberhart and Kennedy [3] in 1995, is a population-based stochastic optimization technique. The optimization algorithm is inspired by the notion of swarm intelligence of birds in order to reach an optimal solution. The algorithm tries to mimic the group communication of a flock of birds. Furthermore, the algorithm has been introduced in the adaptive filtering literature [1] [4]. In the current thesis, the implementation of PSO is compared to cat swarm optimization discussed in Chapter 2 in IIR system identification. Similar to the evolutionary algorithm (EA), PSO is based on a population of individuals, termed a swarm of particles. Each particle has a different possible set of parameters that are optimized by a certain learning rule. The set of parameters of a particle are evaluated by a fitness function, resulting in fitness value, which is the optimality of the particle. The main task is to efficiently search for the single optimum solution in the search space by moving the particles. In each generation, particles are supposed to move toward the best fit solution evaluated in the previous generation, trying to find the current best fit solution that would be better than the previous one. Eventually, all the particles converge to the neighborhood of a single point that has the minimum error.

1.2.1 Conventional PSO

The conventional PSO algorithm starts with an initialization of the swarm of particles by randomly distributing them in the solution space of certain number of dimensions. The number of particles is dependent on the user's choice regarding their performance and the complexity of the target system. Each particle, also called candidate solution, has unknown parameters to be optimized. Those parameters are evaluated by the selected fitness function, resulting in the fitness of each particle in every generation. The algorithm stores the fitness values of all particles and compares them to find a single particle that has the highest fitness in every generation. The particle is called the local best particle, *pbest*, in the current generation. Also, the algorithm compares the fitness values of *pbest* with the best fitness of a particle stored in the previous generation, also called global best particle, *gbest*. The global best particle, *gbest*, is a single particle that has the best fitness value up to the current generation. Thus, the fitness value and location of global best particle is updated when the particle of a current generation has a better fitness value than the *gbest* of the previous generation. The parameters of each particle at every generation are updated according to the rule in the following Eq. 1.6 and Eq. 1.7:

$$v_i(t+1) = \omega * v_i(t) + acc_1 * rand * (gbest - p_i(t)) + acc_2 * rand * (pbest_i - p_i(t)) \tag{1.6}$$

$$p_i(t+1) = p_i(t) + v_i(t+1) \tag{1.7}$$

where t is current generation, v_i is the velocity vector of i^{th} particle and *rand* is a vector of random value between 0 and 1, $p_i(t)$ is the location vector of i^{th} particle of the current generation,

acc_1 and acc_2 are the acceleration constants toward *gbest* and $pbest_i$, respectively, and ω is the inertia weight.

Eq. 1.6 and Eq. 1.7 show that future velocity of a particle is affected by current velocity and location of *gbest* and *pbest*. Two acceleration coefficients determine the step size of a particle, meaning the amount of influence of direction toward *gbest* and *pbest*. Those coefficient values are usually chosen between 0 and 1 since the value chosen to be greater than 1 possibly causes over-step *gbest* or *pbest* and the value chosen to be greater than 2 may cause the algorithm to be unstable. Also, setting the acceleration coefficients between 0 and 1 enables the particle to have constrained step size in the region bounded by *gbest* and *pbest* as shown in Figure 1.2. Typically, the inertia weight is set to decrease as the particle moves closer to *gbest*. Otherwise, the algorithm will be unstable and particles will not converge to *gbest*. Inertia weight, ω, decides the influence of the previous velocity of a particle. In Figure 1.2, *gbest*, *pbest*, and velocity vectors consists of the boundary in which the next generation particle is. As all the particles get closer to *gbest* during the update, the search regions continue to shrink and finally all of the particles converge to *gbest*, of which parameters have the minimum error.

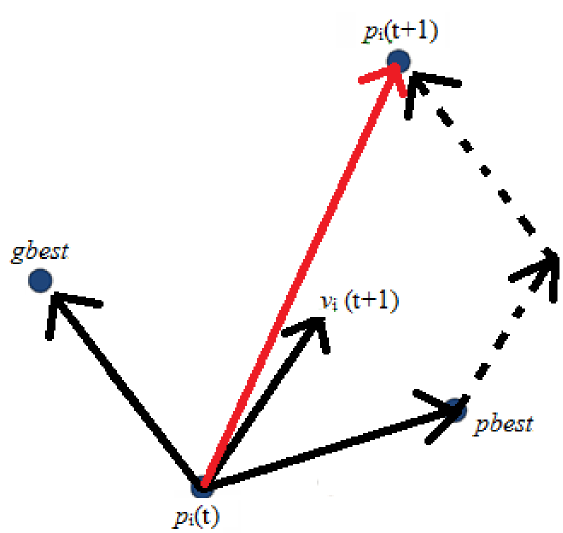

Figure 1-2. Possible Search of a Particle in 2 Dimensions

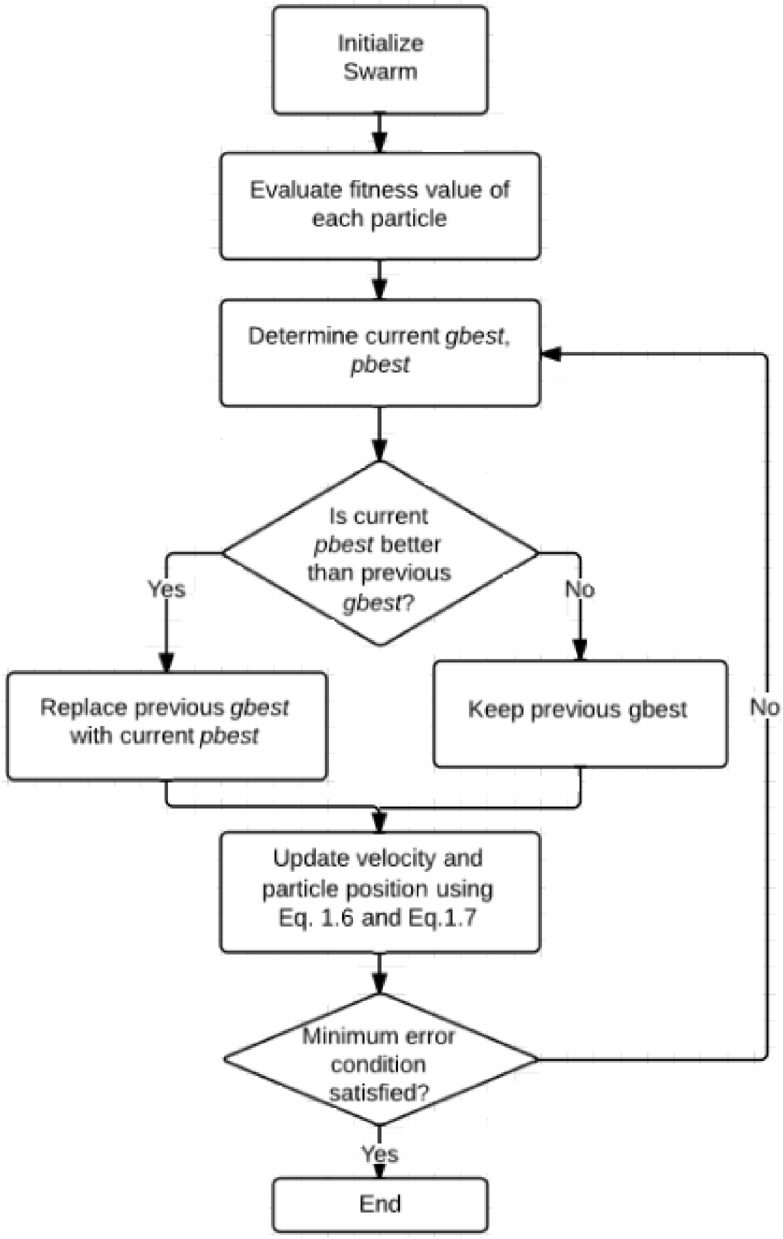

Figure 1-3. Flow Chart of PSO

1.2.2 Parameters of PSO

Figure 1.3 shows the conventional PSO algorithm that is implemented. As a conventional PSO, both acceleration coefficients are chosen to be 1.2, equally and inertia weight linearly decreases from 0.5 to zero by the last iteration. Figure 1.4 shows the effect of acceleration coefficients with the values between 1 and 2 showing the best performance [1]. The Inertia weight affects the global search and the local search. A large inertia weight increases the randomness of the search since it increases the influence of previous velocity. An increase in the randomness of the search decreases the efficient search of the local neighborhood so particles are likely to focus more on the global search. On the other hand, a small value of inertia weight causes particles to focus more on the local search, reducing the influence of previous velocity. This may result in slow convergence. The initial inertia value of 0.5 decaying to zero is chosen from experimental results in Figure 1.5 [1]. The enhancement of convergence speed is conducted by using a Gaussian distribution that has a more peaky distribution rather than a uniform distribution for choosing the random direction vector components.

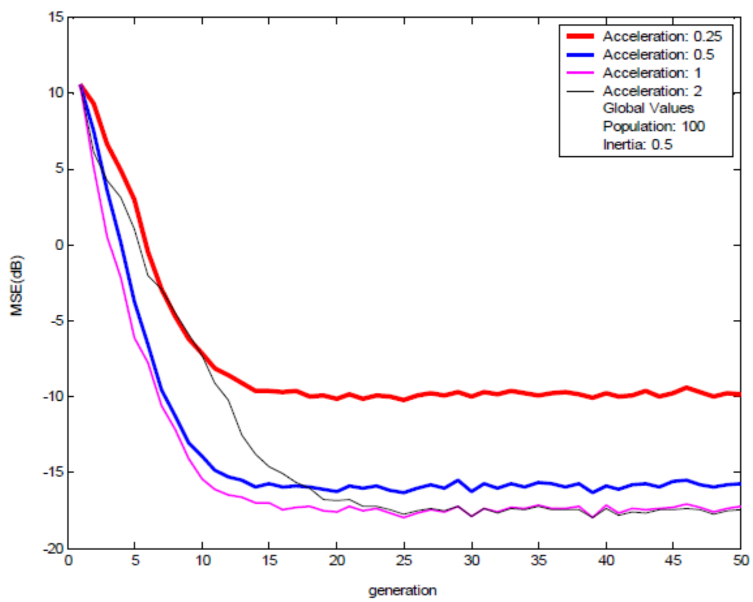

Figure 1-4. The Effect of Acceleration Coefficients [1]

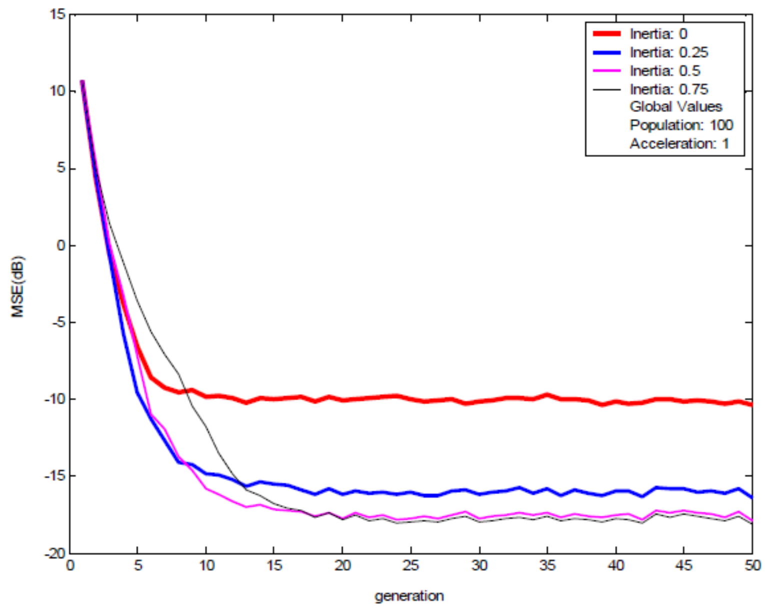

Figure 1-5. The Effect of Inertia Weight [1]

1.3 Simulation Examples

The conventional PSO algorithm is tested with different order IIR systems. 2nd, 3rd, 4th, and 5th order IIR systems are tested with different population of 20, 40, and 100. The parameters for PSO are chosen based on the results from the previous section. Inertia value is chosen to linearly decrease to zero from 0.5 and acceleration coefficients are set to 1.2, equally. For the simulation, MSE is averaged over 50 independent Monte Carlo trials. All the used transfer functions are modeled using the same order IIR filter.

1.3.1 Example 1: 2nd order system

For this example, the transfer function of the 2nd order system shown in Eq. 1.8 is taken from [2] [3] [10].

$$H(z) = \frac{0.05-0.4z^{-1}}{1-1.1314z^{-1}+0.25z^{-2}} \quad (1.8)$$

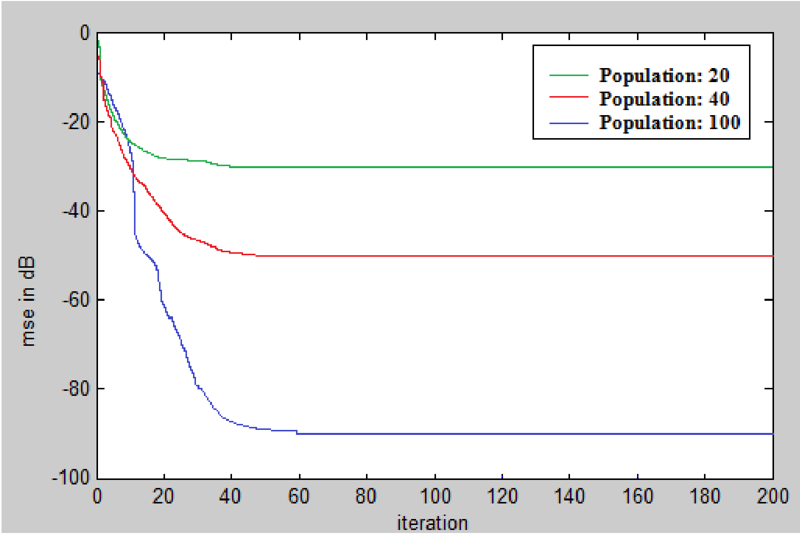

Figure 1-6 Example 1.3.1, PSO with Population Variation

1.3.2 Example 2: 3rd order system

For this example, the transfer function of the 3rd order system shown in Eq. 1.9 is taken from [3] [10] [11].

$$H(z) = \frac{-0.2-0.4z^{-1}+0.5z^{-2}}{1-0.6z^{-1}+0.25z^{-2}-0.2z^{-3}} \tag{1.9}$$

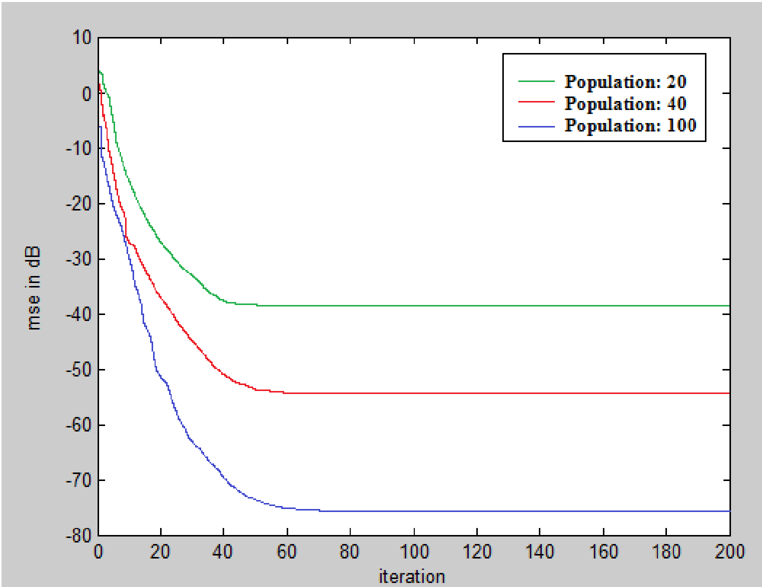

Figure 1-7. Example 1.3.2, PSO with Population Variation

1.3.3 Example 3: 4^{th} order system

For this example, the transfer function of the 4^{th} order system shown in Eq. 1.10 is taken from [3] [12] [13].

$$H(z) = \frac{1-0.9z^{-1}+0.81z^{-2}-0.729z^{-3}}{1+0.04z^{-1}+0.2775z^{-2}-0.2101z^{-3}+0.14z^{-4}} \quad (1.10)$$

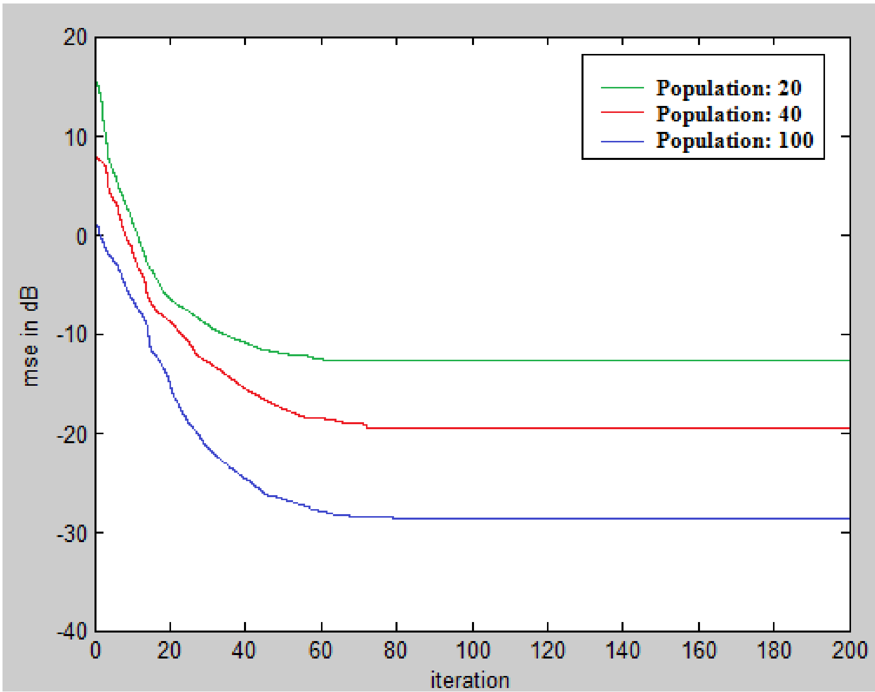

Figure 1-8. Example 1.3.3, PSO with Population Variation

1.3.4 Example 4: 5^{th} order system

For this example, the transfer function of 5^{th} order system shown in Eq. 1.11 is taken from [1] [3] [4].

$$H(z) = \frac{0.1084-0.5419z^{-1}+1.0837z^{-2}-1.0837z^{-3}+0.5419z^{-4}+0.1084z^{-5}}{1+0.9853z^{-1}+0.9738z^{-2}+0.3864z^{-3}+0.1112z^{-4}+0.1113z^{-5}} \quad (1.11)$$

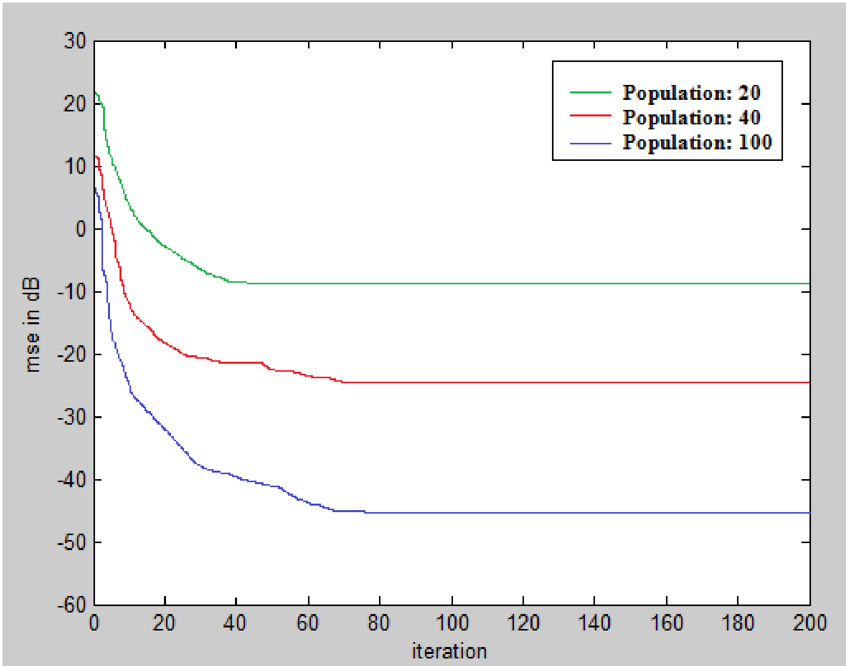

Figure 1-9. Example 1.3.4, PSO with population variation

Population	20	40	100
2^{nd} order system	-31.809	-49.568	-90.623
3^{rd} order system	-38.192	-54.408	-76.649
4^{th} order system	-12.193	-19.462	-28.04
5^{th} order system	-8.545	-24.727	-46.094

Table 1-1 Data Table of PSO with Population Variation

1.3.5 Analysis of the Results

From the figure 1.6 through 1.9, increase in population size causes better MSE value since search agents are more in the solution space. Especially, the MSE value of the lower order system is more affected by the increase of population. However, the 4^{th} order system shows worse MSE value than it of 5^{th} order system in spite of the fact that it is simpler system. Also, simpler system tends to have better MSE value

Chapter 2

Cat Swarm Optimization

Cat swarm optimization algorithm (CSO) was first proposed in 2007 by Chu and Tsai to achieve better performance than PSO [5]. Similar to PSO, CSO also belongs to the swarm intelligence based on population. The cat corresponding to a particle used in PSO is used as an agent. The behavior of a cat is modeled to solve the optimization problem with employing a new learning rule. Two kinds of behavior are the seeking mode and tracing mode. The detail of them is explained in the following section 2.1. The CSO algorithm has been adopted to be used for IIR system identification since it can be to be tuned in order to produce a better search with securing stability for IIR system than PSO. Also, CSO algorithm has been proven as an effective method in solving the optimization problem. Some values for a couple of parameters have been suggested as optimal for employment in CSO for IIR system identification [3]. However, an investigation of all the parameters has to be conducted in order to figure out the optimal values for them. This chapter explains the CSO algorithm in detail and shows the experimental results of optimal values for all the parameters used in CSO.

2.1 Cat Swarm Optimization Algorithm

Chu and Tsai have proposed a new optimization algorithm that adopts a population-based learning rule rooted on the behavior of cats. Cats look lazy since they spend most of their time in resting and move slowly. However, they have a strong curiosity toward moving objects around them even when they are in resting, which means that they always remain alert. On the other hand, when they are aware of a prey, they spend large amount of energy to chase it with high speed. Those two characteristics of slow moving with looking around their surrounding area and chasing the objective with high speed are represented as seeking mode and tracing mode, respectively. So, searching cats are classified into these two modes. Each cat has its' own position to be decided by a fitness function, velocity for each dimension, and a fitness value evaluated by a fitness function in the solution space. The solution space has a certain number of dimensions. Also, every cat has its flag indicating it is in either seeking mode or tracing mode. The number of seeking flags and tracing flags is decided by mixture ratio (MR). MR defines the ratio of the number of tracing mode cats to the number of seeking mode cats.

2.1.1 Seeking Mode

The idea of the seeking mode comes from the slow movement of a cat. This mode corresponds to a local search since the search area is the surrounding area of each cat. Some of the terms used in this mode are seeking memory pool (SMP), seeking range of the selected dimension (SRD), and counts of dimension to change (CDC). SMP is the number of copies of each cat. All the copies of a cat are spread around their original cat based on a SRD range. SRD defines the distance between the original cat and its copies as shown in Figure 2.1. CDC is the percent of dimensions to be changed to the total number of dimensions that each cat has. The details of these parameters are explained in the following section 2.2. This mode has following steps:

1. Create C number of copies of ith cat if SMP = C
2. Based on CDC percents, spread all the copied cats by randomly adding or subtracting SRD percents to or from the present position of original cat.
3. Evaluate the fitness value of all copies and original cat.
4. Select the one having highest fitness and replace the original cat with it.

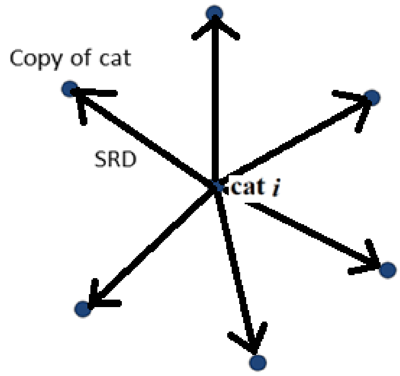

Figure 2-1 Seeking Mode Cat and Its Copies

2.1.2 Tracing Mode

The tracing mode comes from the quick movements of a cat. This mode corresponds to a global search. The tracing mode cat has a velocity and position toward the global best cat, *gbest*. The velocity and position are updated according to the following Eq. 2.1 and Eq. 2.2:

$$v_i\,(t+1) = \omega * v_i(t) + acc * rand * (gbest - p_i(t)) \quad (2.1)$$

$$p_i(t+1) = p_i(t) + v_i(t) \quad (2.2)$$

where ω is the inertia weight, acc is acceleration coefficient, and *rand* is a random number uniformly distributed in the range between 0 and 1.

2.1.3 CSO Algorithm Flow

As mentioned previously, the CSO algorithm has two different cat modes to reach the optimal solution: seeking mode and tracing mode. These two groups of cats join together in solving the optimization problem. The CSO algorithm has following steps:

1. Randomly initialize the position and velocity of cats in D-dimensional space.

2. Evaluate the fitness of the initialized cats and store the cat having the best fitness as global best, *gbest*.

3. According to MR, cats are randomly set to either seeking mode or tracing mode. Once the cat is set to either mode, it does not transfer to the other mode. It keeps its mode until simulation is finished.

4. If a cat is set to seeking mode, conduct seeking mode process. Otherwise, conduct tracing mode process.

5. Evaluate the fitness of all the cats and store the cat having the best fitness as current global best, *gbest*.

6. Compare the fitness of previous global best cat with the cat stored in step 5 and update global best cat.

7. Check the termination condition. If it is not satisfied, repeat steps 4-6.

The flowchart of the algorithm is shown in Figure 2.2.

2.1.4 CSO for IIR Weights Update Algorithm

1. Input signal generation

The fitness value of cat is evaluated at each generation using the entire input data. Random signal x(n), where n = 1,2,…N, is used as an input.

2. Population and velocity matrix construction

The position of the cat, which consists of the weights of the adaptive IIR filter, is initialized with a random value in the range between -1 and 1. If the adaptive filter in Figure 2.3 has D number of weights, the cats should be in D dimensions. A population matrix P for M cats in D dimensions is constructed as M by D matrix. The elements of each row represent the weights of the adaptive filter. Also, corresponding velocity matrix V, M by D matrix, is constructed.

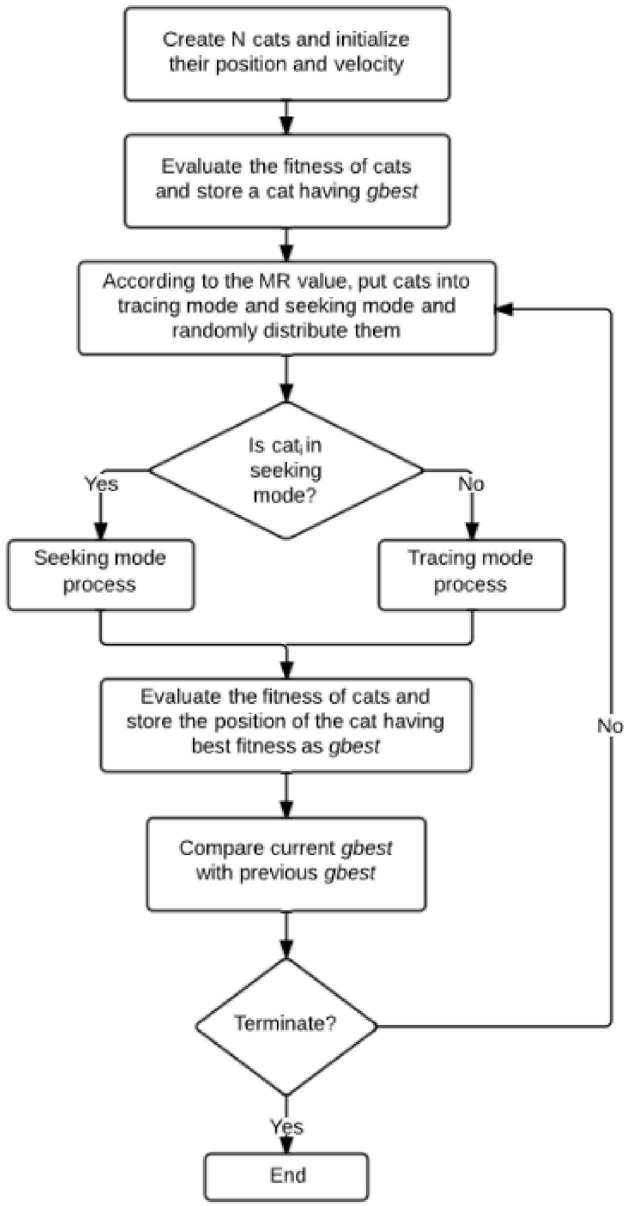

Figure 2-2 Flowchart of Cat Swarm Optimization

3. Setting tracing mode cat and seeking mode cat

Arbitrary integer I_{tm} is chosen in the range between 1 and M, where tm = 1, 2,…, LxMR. The elements of I represent the row indices of population matrix P that are set to go through the tracing mode process. The number of elements of I represents the number of tracing mode cats that the user wants to test. All the other rows are set to go through the seeking mode process.

4. Let the tracing mode cats and seeking mode cats undergo their process.

5. Evaluate the fitness of each cat

i. Pass all N inputs through the benchmarked system as well as the adaptive filter model using the elements of each row in the population matrix as its weights.

ii. Evaluate the fitness of each cat, $J = E[e^2(n)]=\frac{1}{N}\sum_{n=1}^{N} e^2(n)$, where n =1,2,…,N. This represents the MSE for each cat.

6. Store minimum error and the position of cat

The minimum error is stored as J_{min} and the position of cat, which consists of the weights of a cat, is stored as *gbest*.

7. Comparison for minimum error and update of *gbest*

At every generation, J_{min} is calculated and compared to the J_{min} of *gbest* stored so far. Update process is that smaller J_{min} is stored and the position of corresponding cat is also stored.

8. Repeat step 4 through 7 until the maximum generation is reached.

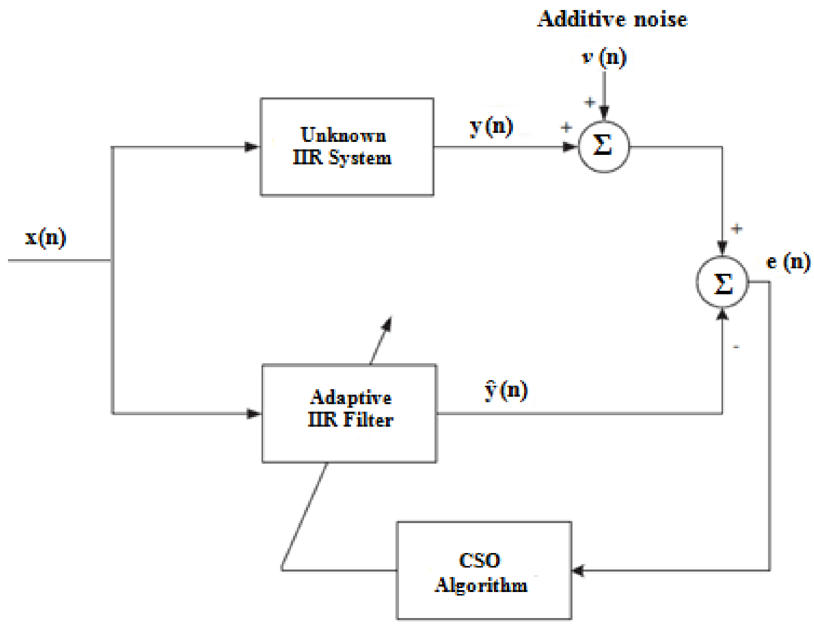

Figure 2-3 Block Diagram of CSO based IIR System Identification

2.2 Parameter Investigation for CSO

There are several parameters such as the MR, CDC, SMP, and SRD in CSO. Optimum values for these parameters have been proposed but not in the context of the IIR system. The following sections provide details for each parameter, show experimental results from several values of each parameter, and propose the combination of their optimum values for the IIR system. Test transfer functions are the same as the four transfer functions of Eq. 1.8, 1.9, 1.10 and 1.11 used in section 1.3.

2.2.1 Mixture Ratio (MR)

As the MR defines the ratio of the number of tracing mode cats to the number of seeking mode, it is set in the range between zero and 1. An MR of 0.9 is chosen in paper [3], showing experimental results but the value between 0.5 and 0.9 was not tested. The objective of the MR test is to find the optimum ratio between the tracing mode cat and seeking mode cat in order to achieve better performance. The tracing mode cat represents the global search and seeking mode cat represents the local search. From the experiments with different IIR plants, the MR value tends to be proportional to the convergence speed due to an increase in the number of tracing mode cats with velocity. A low value of MR increases the proportion of seeking mode cats, reducing the convergence speed. However, from the experiments, it is observed that the local search of the seeking mode cat and the global search of the tracing mode cat should be combined to achieve better search quality. From the test of four different plants, the MR ratio in the range between 0.7 and 0.5 shows the better performance as shown in Figure 2.5, 2.9, 2.13, and 2.17.

2.2.2 Counts of Dimension to Change (CDC)

CDC is defined as the ratio of dimensions to be changed to the total number of dimensions that each cat has. For example, if a cat has 4 dimensions and CDC is set to 75%, randomly chosen 3 out of 4 dimensions are to be mutated. CDC can be thought of as the shape of search direction of seeking mode cats. Figure 2.4 and 2.5, for example, shows the case in which a cat has 3 dimensions. If the CDC is set to 100%, the distribution of copies will be ball-shaped. So, the possible candidates to be searched around the cat are 8 vertices of a cube. If CDC is set to 66%, the distribution of copies will be 4 vertices of 3 plain surfaces that have the original cat in the center. In general, a larger CDC value gives many possible candidates, generating lots of mutation. However, experimental results suggest that a CDC value should be chosen in the range between 50% and 75%. The results mean that a larger value does not always give a better search since lots of mutation may disturb the search for a better solution.

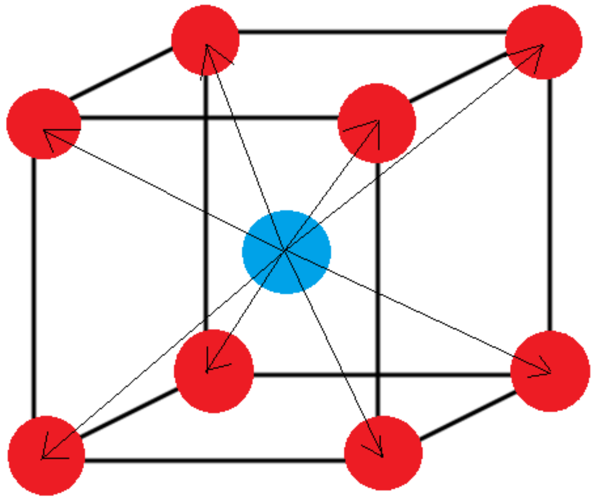

Figure 2-4 Searching Direction based on CDC of 100%

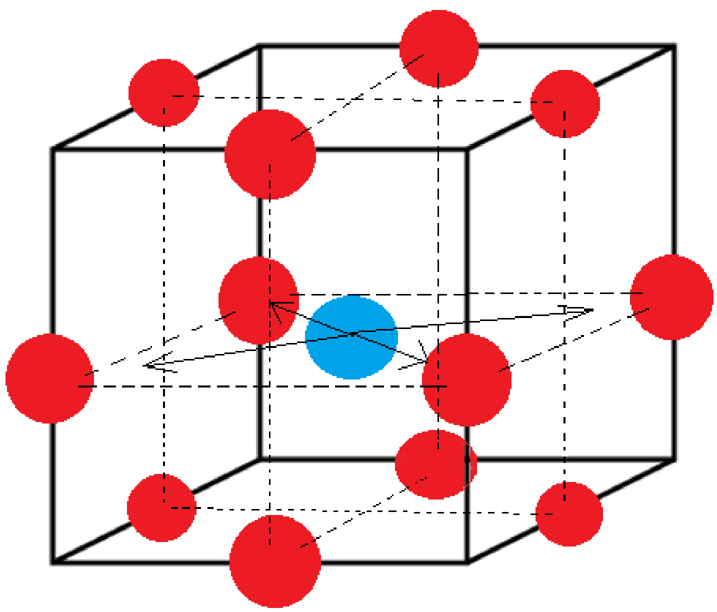

Figure 2-5 Searching Direction based on CDC of 66%

2.2.3 Seeking Memory Pool (SMP)

Seeking Memory Pool is the number of copies of a cat to be spread around its original cat for a local search. The test values for following experiments are 5, 10, 13, 15, and 20. A value in the range between 10 and 15 shows better performance than other values as shown in Figure 2.7, 2.11, 2.15, and 2.19.

2.2.4 Seeking Range of Selected Dimension (SRD)

SRD is defined as a mutative ratio of the selected dimension. SRD is set in the range between 0 and 1. If SRD is set to be 0.1, the 10% of value of the dimension will be either adding to the value of the dimension or subtracting from it. SRD can be thought of as a step size of a cat as well as the distance between the original cat and its copies. A smaller SRD means the copies of cats are spread close to their original one, resulting in narrower local search. A large value of SRD may cause overstep and slow convergence speed. From the experimental results as shown in Figure 2.8, 2.12, 2.16, and 2.20, a SRD value in the range of 0.1 to 0.2 shows better performance.

2.3 Simulation Results

The objective of a test for parameters such as MR, CDC, SMP, and SRD is to find the optimum combination that produces lower errors. Tests of them are conducted, sequentially. First, MR values of 0.1, 0.3, 0.5, 0.6, 0.7, and 1 are tested. For the MR test, CDC, SMP, and SRD are initially set to be 100%, 5, and 0.2, respectively. The value of MR showing the best performance is kept for the CDC test. With a certain MR value, the CDC test is conducted with values of 100%, 75%, 50%, and 25%. For the higher-order system such as 5th order system, the CDC of 66% is also tested. The value of CDC having the best performance is kept for the SMP test. SMP values of 5, 13, 15, and 20 are tested. Finally, the SRD values of 0.1, 0.15, 0.2, and 0.3 are tested.

2.3.1 Example 1: 2nd order system

The 2nd order transfer function of the plant shown in Eq. 1.8 is tested for performance comparison with PSO. The test starts with a CDC of 100%, SMP of 5, and SRD of.0.2. It is suggested that MR of 0.5, CDC of 50%, SMP of 13, and SRD of 0.1 shown in Figure 2.9 is the optimum combination of those parameters for this system. The MR value of 0.5 shows the MSE value of -40.885dB in the MR variation test. The CDC value of 75% lowers the MSE value down to -44.041dB shown in Figure 2.7. By choosing a SMP value of 13, the MSE value decreases down to -46.462 dB. The SRD value of 0.1 also decreases the MSE value of -46.462 down to -50.99 dB shown in Figure 2.9. The improvement of MSE value in Figure 2.6 through 2.9 is about -10.1dB. The final minimum MSE value is -50.99 dB, which is much better than the PSO performance of -30.435 dB.

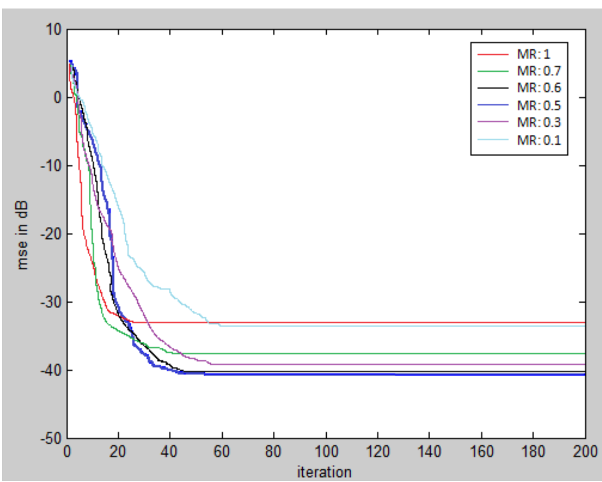

Figure 2-6 Example 1: MR variation

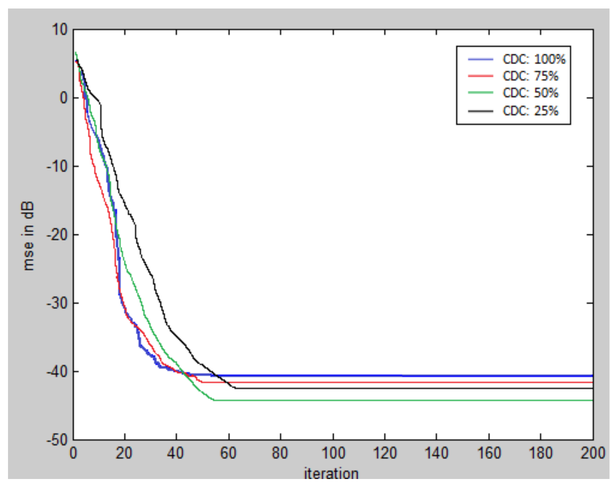

Figure 2-7 Example 1: CDC variation

MR	1	0.7	0.6	**0.5**	0.3	0.1
MSE (dB)	-33.3910	-38.6827	-40.3247	**-40.8845**	-39.3056	-32.317

Table 2-1 Example 1: MR variation

CDC	100%	75%	**50%**	25%
MSE (dB)	-40.8845	-41.1165	**-44.0419**	-41.8297

Table 2-2 Example 1: CDC variation

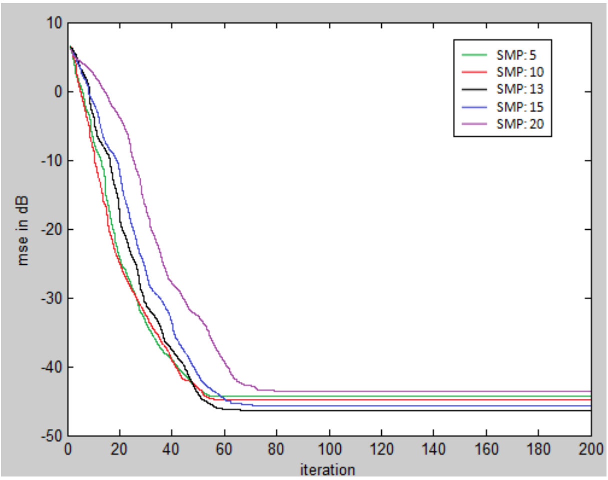

Figure 2-8 Example 1: SMP variation

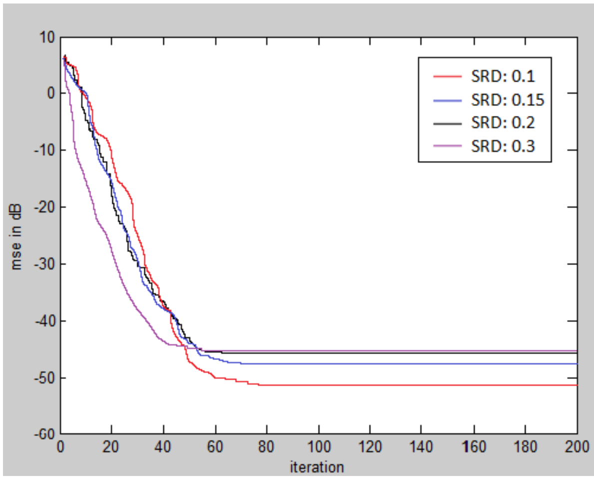

Figure 2-9 Example 1: SRD variation

SMP	5	10	**13**	15	20
MSE (dB)	-44.0419	-44.7738	**-46.4627**	-45.6247	-43.8297

Table 2-3 Example 1: SMP variation

SRD	**0.1**	0.15	0.2	0.3
MSE (dB)	**-50.99**	-48.1041	-46.4627	-46.4458

Table 2-4 Example 1: SRD variation

2.3.2 Example 2: 3rd order system

The 3rd order transfer function of the plant is shown in Eq. 1.9. The test starts with a CDC of 100%, SMP of 5, and SRD of.0.2 and ends up with a MR of 0.7, CDC of 50%, SMP of 15, and SRD of 0.2 as an optimum combination shown in Figure 2.13. The MR value of 0.7 shows the MSE value of -44.207dB. The CDC value of 50% lowers the MSE value down to -45.41 dB. By choosing a SMP value of 15, the MSE value decreases down to -47.987 dB. The SRD value of 0.2 shows the MSE value of -47.987, which is better than other values shown in Figure 2.13. The improvement of MSE value in Figure 2.10 through 2.13 is about -5.78 dB. The final minimum MSE value is -47.987 dB, which is better than the PSO performance of -39.379 dB.

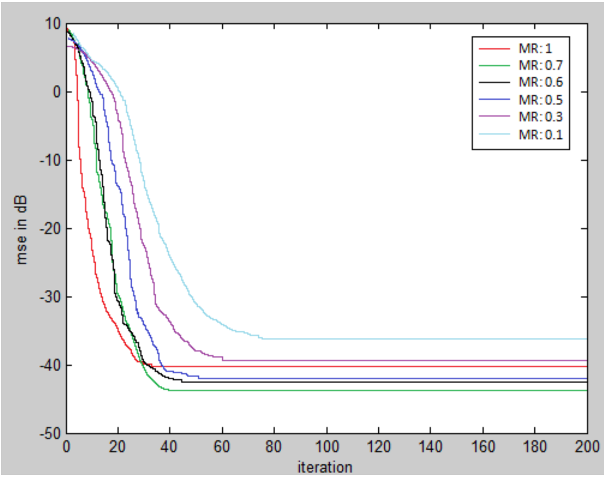

Figure 2-10 Example 2: MR variation

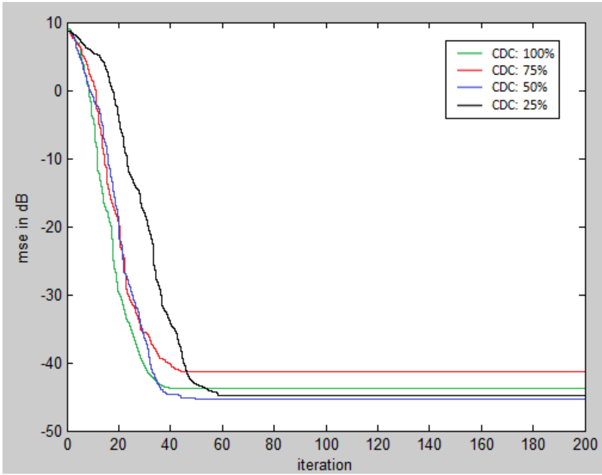

Figure 2-11 Example 2: CDC variation

MR	1	**0.7**	0.6	0.5	0.3	0.1
MSE (dB)	-40.444	**-44.2074**	-42.9373	-42.2728	-39.2286	-36.5863

Table 2-5 Example 2: MR variation

CDC	100%	75%	**50%**	25%
MSE (dB)	-44.2074	-41.7319	**-45.4104**	-44.5759

Table 2-6 Example 2: CDC variation

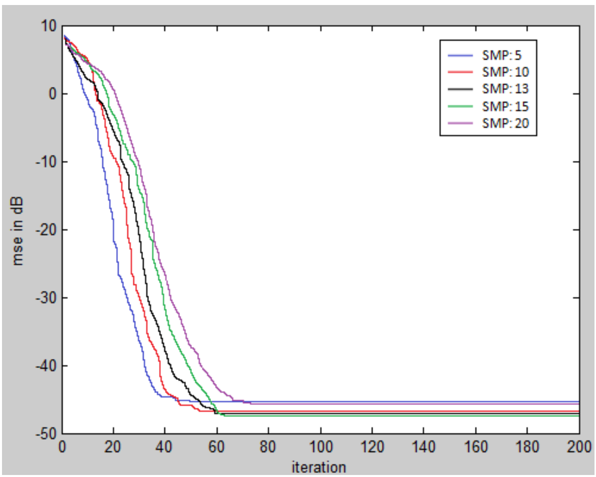

Figure 2-12 Example 2: SMP variation

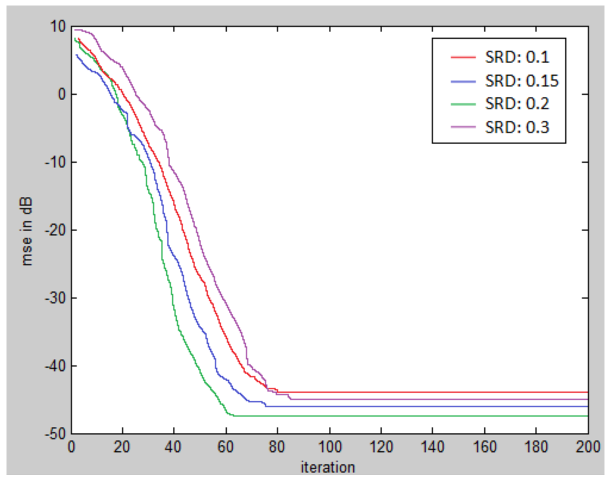

Figure 2-13 Example 2: SRD variation

SMP	5	10	13	**15**	20
MSE (dB)	-45.4104	-47.8392	-47.88	**-47.987**	-45.6913

Table 2-7 Example 2: SMP variation

SRD	0.1	0.15	**0.2**	0.3
MSE (dB)	-43.157	-45.671	**-47.987**	-44.507

Table 2-8 Example 2: SRD variation

2.3.3 Example 3: 4th order system

The 4th order transfer function of the plant shown in Eq. 1.10 is tested for performance in comparison with PSO. The test suggests that MR of 0.7, CDC of 75%, SMP of 10, and SRD of 0.1 is the optimum combination of those parameters for this system. The MR value of 0.7 shows the MSE value of -16.818 dB shown. The CDC value of 75% lowers the MSE value down to -17.537 dB. By applying the SMP value of 10, the MSE value decreases down to -19.073 dB. The SRD value of 0.1 also helps decrease the MSE value of -19.073 down to -21.038 dB shown in Figure 2.17. The improvement of the MSE value in Figure 2.14 through 2.17 is about -4.2 dB. The final minimum MSE value is -21.038 dB, which is better than the PSO performance of -12.192 dB.

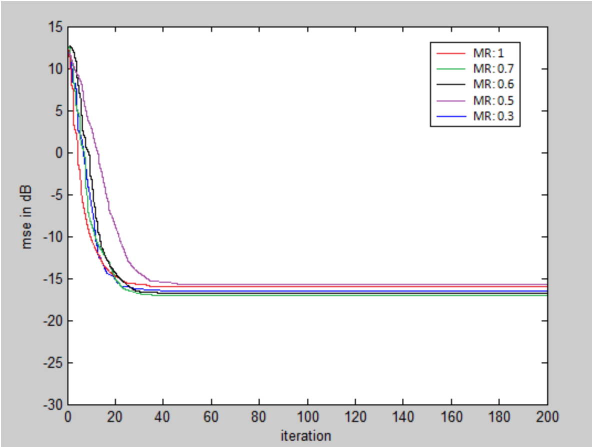

Figure 2-14 Example 3: MR variation

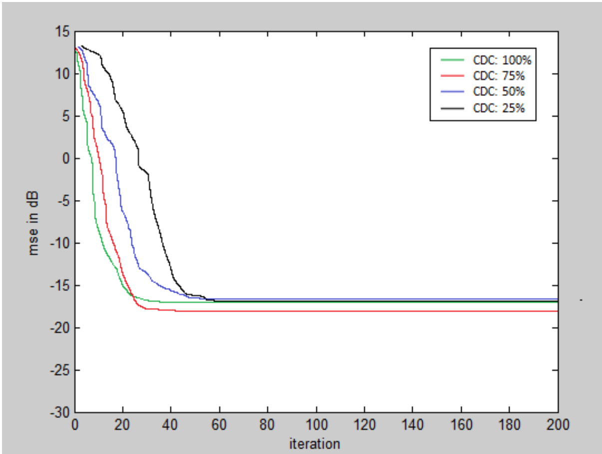

Figure 2-15 Example 3: CDC variation

MR	1	**0.7**	0.6	0.5	0.3	0.1
MSE (dB)	-15.7873	**-16.8187**	-16.4161	-15.7518	-16.5209	-16.1445

Table 2-9 Example 3: MR variation

CDC	100%	**75%**	50%	25%
MSE (dB)	-16.8187	**-17.5375**	-16.3226	-16.7511

Table 2-10 Example 3: CDC variation

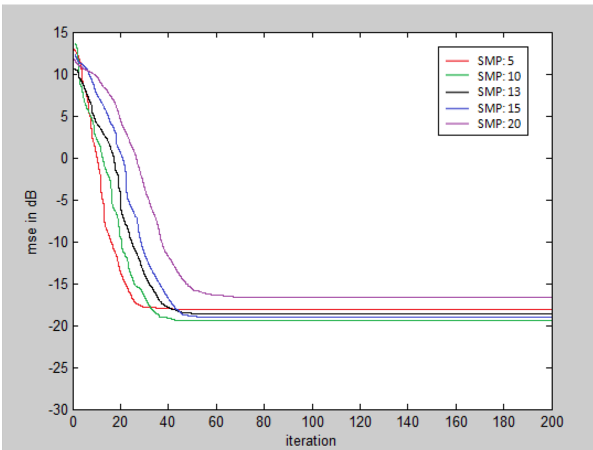

Figure 2-16 Example 3: SMP variation

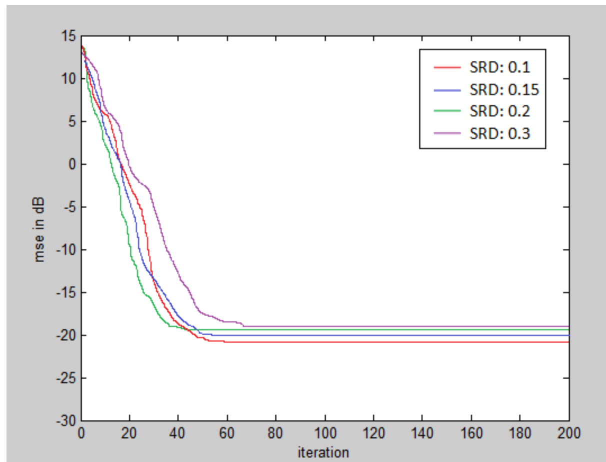

Figure 2-17 Example 3: SRD variation

SMP	5	**10**	13	15	20
MSE (dB)	-17.5375	**-19.0736**	-18.0155	-18.6638	-17.1953

Table 2-11 Example 3: SMP variation

SRD	**0.1**	0.15	0.2	0.3
MSE (dB)	**-21.0358**	-20.2507	-19.0736	-19.0462

Table 2-12 Example 3: SRD variation

2.3.4 Example 4: 5th order system

The 5th order transfer function of the plant shown in Eq. 1.11 is tested. The parameter test for this system suggests that a MR of 0.6, CDC of 50%, SMP of 13, and SRD of 0.1 shown in

Figure 2.21 is the optimum combination of these parameters. The MR value of 0.6 represents the MSE value of -32.468 dB. The CDC value of 50% lowers the MSE value down to -33.566 dB. By selecting the SMP value of 13, the MSE value decreases down to -35.908 dB. The SRD value of 0.1 also decreases the MSE value of -35.908 down to -37.03 dB shown in Figure 2.21. The improvement of the MSE value in Figure 2.18 through 2.21 is about -4.47 dB. The final minimum MSE value is -37.03 dB, which is much better than the PSO performance of -8.545 dB.

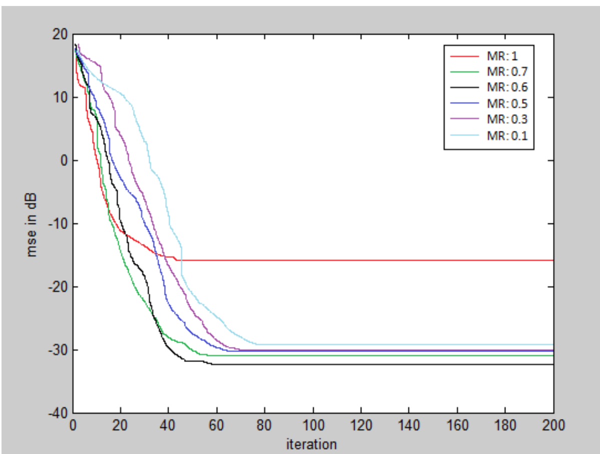

Figure 2-18 Example 4: MR variation

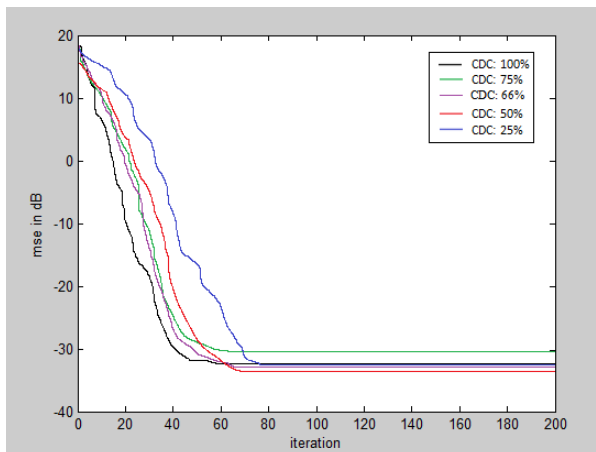

Figure 2-19 Example 4: CDC variation

MR	1	0.7	**0.6**	0.5	0.3	0.1
MSE (dB)	-16.1445	-31.1052	**-32.4683**	-30.2604	-30.1209	-29.3789

Table 2-13 Example 4: MR variation

CDC	100%	75%	66%	**50%**	25%
MSE (dB)	-32.1647	-30.3204	-32.4683	**-33.5663**	-31.738

Table 2-14 Example 4: CDC variation

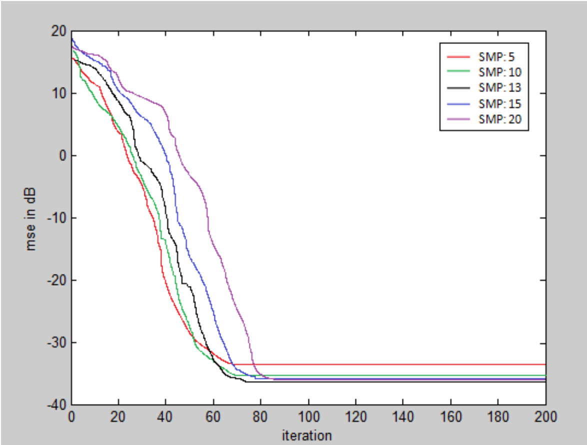

Figure 2-20 Example 4: SMP variation

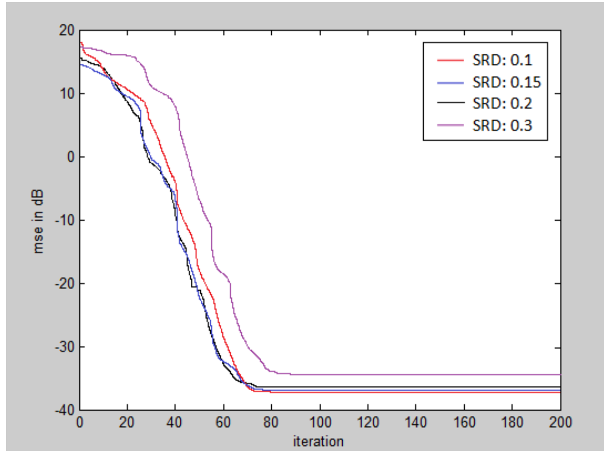

Figure 2-21 Example 4: SRD variation

SMP	5	10	**13**	15	20
MSE (dB)	-33.5663	-35.4721	**-35.9081**	-34.7242	-34.7977

Table 2-15 Example 4: SMP variation

SRD	**0.1**	0.15	0.2	0.3
MSE (dB)	**-37.0302**	-36.8376	-35.9081	-34.6345

Table 2-16 Example 4: SRD variation

2.4 Data Comparison

This section compares data results from a couple of papers that apply different algorithm to the same IIR systems tested in the current thesis.

2.4.1 Example 1: 2^{nd} order system

The algorithms proposed from the paper [3] and [10] are evaluated by the 2^{nd} order transfer function shown in Eq. 1.8 with the population size of 50. The paper [3] chooses different values of the parameters for CSO and the paper [10] proposes the harmony search (HS) algorithm. The table 2.17 shows the data results from the current thesis and these papers. The CSO algorithm in the current thesis shows the minimum MSE value.

Algorithm	CSO	CSO [3]	HS [10]
MSE (dB)	**-67.3592**	-41.948	-49.629

Table 2-17 Example 1: Data Comparison

2.4.2 Example 2: 3rd order system

The algorithms proposed from the paper [3] and [10] are evaluated by the 2nd order transfer function shown in Eq. 1.9 with the population size of 50. The table 2.18 shows the data results from the current thesis and these papers. The CSO algorithm in the current thesis shows the minimum MSE value.

Algorithm	CSO	CSO [3]	HS [10]
MSE (dB)	**-63.431**	-41.97	-45.411

Table 2-18 Example 2: Data Comparison

2.4.3 Example 3: 4th order system

The algorithms suggested in the paper [3] and [13] are evaluated by the 4th order transfer function shown in Eq. 1.10 with the population size of 50. The table 2.19 shows the data results from the current thesis and these papers. The CSO algorithm used in paper [3] shows the minimum MSE value.

Algorithm	CSO	CSO [3]	PSO-QI [13]
MSE (dB)	-26.742	**-42.26**	-23.098

Table 2-19 Example 3: Data Comparison

2.4.4 Example 4: 5th order system

The suggested algorithms in the paper [1] and [3] are evaluated by the 5th order transfer function shown in Eq. 1.11 with the population size of 50. Paper [1] proposes modified particle swarm optimization (MPSO) algorithm. The table 2.20 shows the data results from the current thesis and these papers. The MPSO algorithm used in paper [1] shows the minimum MSE value.

Algorithm	CSO	MPSO [1]	CSO[3]
MSE (dB)	-46.129	**-69.5**	-41.94

Table 2-20 Example 4: Data Comparison

2.4.5 Example 4: 5th order system with Population Variation

The MPSO suggested in the paper [1] is evaluated by the 5th order transfer function with different population size of 20, 40 and 50. The result from the paper [1] is compared to the CSO in the current thesis in the table 2.21. The CSO tends to be more effective with smaller number of population; on the other hand, MPSO tends to be more effective with larger number of population sizes.

Population size	CSO	MPSO [1]
20	**-37.03**	-19.5
40	**-44.71**	-17.5
50	-46.129	**-69.5**

Table 2-21 Example 4: Data Comparison with population variation

2.5 Summary

The main difference in the learning rule between the PSO and CSO is that the local and global searches for *pbest* and *gbest*, respectively, in PSO affect all particles' velocity and position that decide the step size of the particle in the next generation. On the other hand, the CSO has different strategies for global and local searches. The update of velocity and position is used only for the global search cats in tracing mode, not for the local search cats in seeking mode. For the cats in seeking mode, the step size of them is selected by the SRD value chosen, not affected by the velocity and position. Without the large change in the velocity and position of seeking mode cats, they are possible to focus on an exhaustive search for the local area. Randomly-picked dimensions to be mutated based on the CDC value provide a diverse movement of local cats for a better solution. The SMP also provides a proper number of copies of a cat with regard to convergence speed as well as MSE value. From the experimental tests, the results of this work suggests that a MR value in the range between 0.7 and 0.5, CDC percents in the range 75% and 50%, a SMP value in the range between 10 and 15, and a SRD value in the range between 0.1 and 0.2 are optimal for the IIR system identification.

One possible future work is to test more complex IIR systems with parallel and cascade structures to improve the MSE value since parallel and cascade structures may possibly provide a lower MSE value by reorganizing the higher-order system into a lower-order system. Also, it can be observed that how well the CSO algorithm lowers the MSE value with these structures.

Another possible direction for future work is to implement PSO-CSO hybrids by combining the global search technique of the PSO with the local search technique of the CSO. From the data in the table 2.21, the PSO-CSO hybrids can perform effectively both small and large number of population. The local search technique of CSO using the CDC, SRD, and SMP parameters may provide a better search for the optimum solution with an exhaustive search. The drawback of the local search is a low convergence speed but the global search technique of the PSO can redeem the low convergence speed.

References

1. Krusienski, D.J., “Enhanced structured stochastic global optimization algorithms for IIR and nonlinear adaptive filtering,” Ph.D. Thesis, Dept. of Electrical Eng., The Pennsylvania State University, University Park, PA, 2004.
2. Shynk, J.J., "Adaptive iir filtering," IEEE Transactions on Signal Processing, 4-21, 1989a.
3. Panda, G., Pradhan, P.M. and Majhi, B, "IIR System Identification Using Cat Swarm Optimization″, Expert Systems with Applications, (Elsevier), vol. 38, no. 10, p.12671-12683, 2011.
4. Mars, P., Chen, J.R. and Nambiar, R., "Learning Algorithms: Theory and Applications in Signal Processing, Control, and Communication," CRC Press, Inc., 1996
5. Chu, S.-C. and P.-W. Tsai, "Computational Intelligence Based On The Behavior Of Cats", International Journal of Innovative Computing, Information and Control, vol.3, no.1, pp.163-173, 2007
6. Eberhart, R.C. and Kennedy, J., "A new optimizer using particle swarm theory," Proceedings of the Sixth International Symposium on Micromachine and Human Science, nagoya, Japan. pp. 29-43, 1995.
7. Merkle, D and Middendorf, M., “Swarm Intelligence and Signal Processing,” IEEE Signal Processing Magazine, p. 152-158, November 2008.
8. Krusienski, D. J. and Jenkins, W. K., “A particle swarm optimization- LMS hybrid algorithm for adaptive filtering,” Proc. of the 38th Asilomar Conf. on Signals, Systems, and Computers, Nov. 2004.
9. Krusienski, D.J. and Jenkins, W.K., "Design and Performance of Adaptive Systems Based on Structured Stochastic Optimization Strategies," IEEE Circuits Syst. Magazine, p. 8-20, 2005.
10. Shafaati, M., Ahmadi, M. and Mojallai, H, “ Identification of IIR Systems Using Harmony Search Algorithm,” 2nd International Conference on Control, Instrumentation and Automation, 2011.

11. Ng, S.C., Leung, S.H., Chung, C.Y., Luk, A., Lau, W.H., " The genetic search approach. A new learning algorithm for adaptive IIR filtering", IEEE Signal Processing Magazine, p38-46, 1996.

12. Shynk, J.J., "Adaptive iir filtering," IEEE Transactions on Signal Processing, 4-21, 1989b.

13 Luitel, B. and Venayagamoorthy, G.K., "Particle Swarm Optimization with Quantum Infusion for System Identification", Engineering Application of Artificial Intelligence, vol. 23, issue 5, p.635-649, 2010.