Step Size Optimization of LMS Algorithm Using Aunt Colony Optimization & Its comparison with Particle Swarm optimization Algorithm in System Identification

Shikha Tripathi, Mohammad Asif Ikbal

Assistant professor, Department of Electronics & Communications Engineering, JIT,Barabanki, India

---***---

Abstract - ***In order to transform and analyze signals that have been sampled from analogue sources, digital signal processing (DSP) algorithms are employed. The advantages of DSP are based on the fact that the performance of the applied algorithm is always predictable. Further, the digital techniques are error prone due to quantization, free of device errors which are usually caused by the device noises, mismatches due statistical nature of the fabrication processes, and the defects that may arise during fabrication. DSP algorithms are usually described as a digital filter. Digital filters can be broadly divided into two-sub classes: finite impulse-response filters and infinite impulse-response (IIR) filters. Because the error surface of IIR filters is generally multi-modal, global optimization techniques are required in order to avoid local minima and design efficient digital IIR filters. In this work, a new method based on the Ant Colony Optimization algorithm is proposed. This algorithm is found efficient with global optimization ability, therefore it is proposed for the design of digital filters.***

The algorithm is implemented using MATLAB, and the simulation results obtained show that the proposed approach is quite efficient, accurate and has a fast convergence rate. The results obtained also demonstrate that the proposed method can be efficiently used in digital filter designing.

1. INTRODUCTION

Digital signal processing systems has become an active area of research in recent decades due to their low cost, reliability, accuracy, small physical sizes and flexibility. One of the most common digital signal processing systems is the digital filters. Non-adaptive filters cannot process time-varying nonstationary signals and they require *a* priori knowledge of the statistics of the signal to be processed. Contrary to non-adaptive structures, if the input data varies with respect to time or there is no prescribed specification for the variation, adaptive filters are needed.

Adaptive filters are associated to one of two major groups by their impulse response: finite impulse response (FIR) and infinite impulse response (IIR) filters. Due to their uni-modal error surfaces and intrinsic stable behavior, gradient based algorithms, such as least-mean-square (LMS) and normalized least-mean square (NLMS), are very effective in the design of adaptive FIR filters.

However, since the gradient based algorithms try to find the global minimum of the error surface by moving in the direction of the negative gradient, approaches based on these algorithms may lead the filter to a local minimum when the error surface is multi-modal as such in IIR filters. IIR filters can provide a much better performance than the FIR filters having the same number of coefficients.

1.1 Problem Definition:

Despite this important advantage of IIR filters over FIR filters, there are two main problems encountered in the design of IIR filters: they might have a multi-modal error surface and the filter might become unstable during the adaptation process. The stability of the filter can be handled by limiting the parameter space in a suitable value range. The multi-modal error surface of IIR filter causes the gradient based algorithms to be stuck at local minima and not converge to the global optimum

1.2 Analysis of Previous Research Works and Scope of Work:

In order to overcome this problem and find the global optimum solution, approaches based on global optimization algorithms such as genetic algorithm(GA), simulated annealing (SA), tabu search (TS), differential evolution (DE), particle swarm optimization (PSO) and artificial bee colony (ABC) algorithms can be used for designing adaptive IIR filters. Among these algorithms, DE, PSO, TS, ABC, SA and GA based design methods have been described and applied to the adaptive filter. One weakness of conventional PSO is that its local search is not guaranteed convergent; its local search capability lies primarily in the swarm size and search parameters. On the other hand, the problem with simply running a brute-force population of independent LMS algorithms is that there is no collective information exchange between population members, which makes the algorithm inefficient and prone to the local minimum problem of standard LMS. Therefore, it is desirable to combine the convergent local search capabilities of the LMS algorithm with the global search of ACO

1.3 Formulation of Problem:

In this work, a new approach based on ACO algorithm is presented for both FIR and IIR adaptive filter structures and an improved version of ACO algorithm is applied for adaptive noise cancellation. Also, a new approach for finding the optimal step size value of LMS-type algorithms is proposed in the design of adaptive FIR filters.

When initialized in the global optimum valley, the LMS algorithm can be tuned to provide an optimal rate of convergence without apprehension of encountering a local minimum. Therefore, by using a structured stochastic search, such as ACO, to quickly focus the population on regions of interest, an optimally tuned LMS algorithm can take over and provide better results than standard LMS.

An important step in System identification procedure is the estimation of parameters. When an input is applied to both the system and model, and the difference between the target system's output and model's output is used in appropriate manner to update a parameter vector to reduce this difference. To apply the parameter vector, we use LMS algorithm. Because of the computational simplicity of the LMS algorithm, this algorithm is widely used. But it suffers from a slow rate of convergence. Further, for implementation of LMS algorithm, we need to select appropriate value of the step size, which affects the stability and performance. We have search algorithm, Aunt Colony Optimization (ACO) to control the value of the step size in accordance with the input adaptively. This work introduces a novel algorithm named Aunt Colony Optimization (ACO) to optimize the step size of LMS algorithm and then LMS algorithm calculate system identification parameters adaptively.
After that a comparison is carried out between the results of PSO& ACO algorithms.

2. What is ACO?

Ant colony algorithm is the probabilistic technique to compute the computational problem. This algorithm is based on the food finding technique of ant. Ant is seeking for path between their colony and food. So ant searching for the food and search which food is nearer to their colony then establish shortest path between colony and food With the help of pheromones trial. The inspiring source of ACO is the foraging behavior of real ants.

ACO technique:

- probabilistic technique
- Searching for the optimal path in the graph based on behavior of ants seeking a path between their colony and source of food.
- Meta-heuristics optimization

2.1 Application of ACO:

- Routing in telecommunication networks
- Traveling Salesman
- Graph Coloring
- Scheduling
- Constraint Satisfaction
- Optimizing a PID controller parameters
- CNC tool path selection algorithm

2.1 Basic configuration of ACO:

Ant moves random in path because they are blind. They select shortest path between colony and food source via pheromone trails. Each ant moves randomly and pheromone is deposited on the path .Ants move towards the maximum pheromones on path .more the pheromone on the path increases the probability of the path followed by the ants. Pheromones get evaporated time to time so where ,less no of ant are there pheromones is less to each ant attracted time to time strongest pheromones .

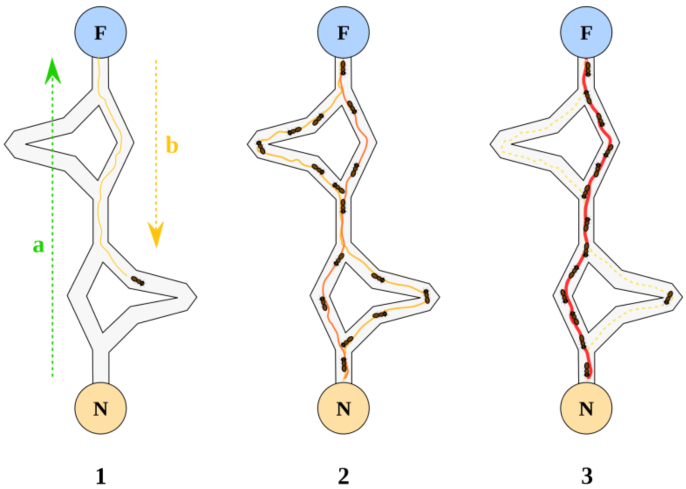


Figure. 2.1 understanding the movement of ant and understand the ant colony optimization.

2.2 Implementation of ACO:

Implementation of algorithm that adapts the behavior of real ants to optimize real ant problems on graphs. The arterial ants make the solution to certain optimization problem. The information about these solutions making allusion to the communication system of real ants.

$$p_{ij}(t) = \frac{\tau_{ij}(t)^{\alpha}\left(\frac{1}{dij}\right)^{\beta}}{\sum_{j \ni nodes} \tau_{ij}(t)^{\alpha}\left(\frac{1}{dij}\right)^{\beta}}$$

...eq.(2.1)

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \sum_{\substack{k \in colonythat \\ usededge \\ (i,j)}} \frac{Q}{L_m}$$

……eq.(2.2)

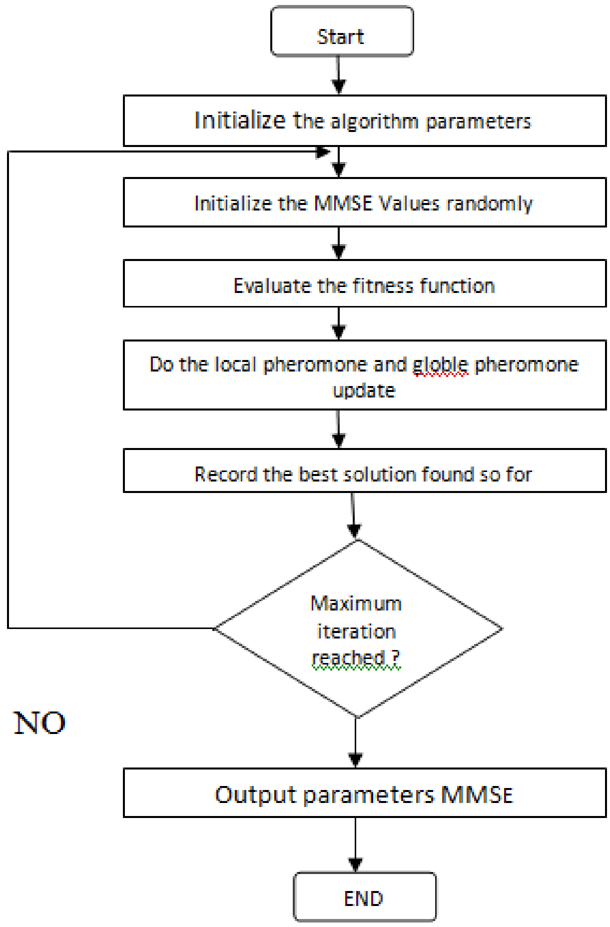
Equation 2.1 represents the probability of ant to move between the two nodes I and j.
Equation 2.2 represents the local updates of pheromone after travelling from node to node.

- Set a concentration *of* pheromone τij to each link (i, j).
- Assigned a number k=1, 2… n in the nest.
- Iteratively build a path to the food source, using Eq. (2.1) for every ant.

Remove cycles and compute each route weight $f(x^k(t))$. A cycle could be generated when there are no feasible candidates' nodes then the predecessor of that node is included as a former node of the path.

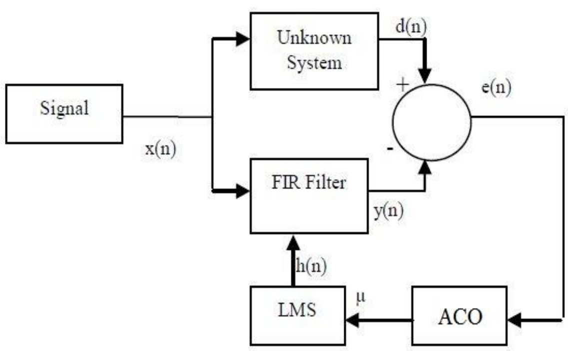
- Update of the pheromone concentration using Eq. (2.2).
- Finally, finish the algorithm in any of the three different ways:
  - When a maximum number of approaches has been reached.
  - When it has been found an acceptable solution, with $f(x^k(t))<\varepsilon$.
  - When all ants follow the same path.

2.3 Flow chart representation of ACO:



3. Unknown System Model

The unknown system will be modeled by a FIR system of length N. Both the unknown system and the FIR model are connected in parallel and exited by the same input sequence {x(n)}. If {y(n)} denotes the output of the model and {d(n)} denotes the output of the unknown system, the error sequence is {e(n)=d(n)-y(n)}.The unknown system is FIR system.



In this paper a new procedure to estimate the coefficients of an adaptive filter is proposed. It combines the Ant Colony Optimization Algorithm with Least-Mean-Square (LMS) method, i.e. in each iteration of ACO, after the calculation of gbest, an LMS algorithm will be applied based on the previous gbest. In our structure, error signal is adjusted to the ACO block and ACO decides an appropriate step-size with less error value. Then selected step-size value is adjusted to the LMS block, and LMS block updates the coefficients simultaneously. The main advantage of ACO is that it can escape local minima, but is a slow process. LMS algorithm is a faster algorithm but may diverge in some cases or may remain in local minima and its results are not as accurate as ACO-based procedures. Our proposed approach combines the benefits of both algorithms while accelerates the very slow rate of ACO and escapes from the local minima which may result from LMS. [1]

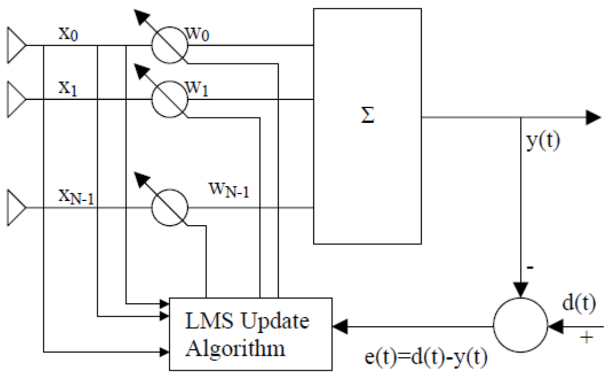
4 Least Mean Square (LMS) algorithm:

The Least Mean Square (LMS) algorithm, introduced by Widrow and Hoff in 1959 is an adaptive algorithm, which uses a gradient-based method of steepest decent. LMS algorithm uses the estimates of the gradient vector from the available data. LMS incorporates an iterative procedure that makes successive corrections to the weight vector in the direction of the negative of the gradient vector which eventually leads to the minimum mean square error. Compared to other algorithms LMS algorithm is relatively simple; it does not require correlation function calculation nor does it require matrix inversions.

Consider a Uniform Linear Array (ULA) with N isotropic elements, which forms the integral part of the adaptive beam forming system as shown in the figure below.

The output of the antenna arrays given by,

$$x(t) = s(t)a(\theta_0) + \sum_{i=1}^{N_u} u_i(t)a(\theta_i) + n(t)$$



s(t) denotes the desired signal arriving at angle θ_0 and $u_i(t)$ denotes interfering signals arriving at angle of incidences θ_i respectively. $a(\theta_0)$ and $a(\theta_i)$ represents the steering vectors for the desired signal and interfering signals respectively. Therefore it is required to construct the desired signal from the received signal amid the interfering signal and additional noise n(t).

As shown above the outputs of the individual sensors are linearly combined after being scaled using corresponding weights such that the antenna array pattern is optimized to have maximum possible gain in the direction of the desired signal and nulls in the direction of the interferers. The weights here will be computed using LMS algorithm based on Minimum Squared Error (MSE) criterion. Therefore the spatial filtering problem involves estimation of signal from the received signal x(t) (i.e. the array output) by minimizing the error between the reference signal d(t) , which closely matches or has some extent of correlation with the desired signal estimate and the beam former output y(t) (equal to wx(t)). This is a classical Weiner filtering problem for which the solution can be iteratively found using the LMS algorithm.

From the method of steepest descent, the weight vector equation is given by,

$$w(n+1) = w(n) + \frac{1}{2}\mu[-\nabla(E\{e^2(n)\})]$$

Where μ is the step-size parameter and controls the convergence characteristics of the LMS algorithm; $e^2(n)$ is the mean square error between the beam former output y(n) and the reference signal which is given by,

$$e^2(n) = [d^*(n) - w^h x(n)]^2$$

The gradient vector in the above weight update equation can be computed as

$$\nabla_w(E\{e^2(n)\}) = -2r + 2Rw(n)$$

In the method of steepest descent the biggest problem is the computation involved in finding the values r and R matrices in real time. The LMS algorithm on the other hand simplifies this by using the instantaneous values of covariance matrices r and R instead of their actual values i.e.

$$R(n) = x(n)x^h(n)$$

$$r(n) = d^*(n)x(n)$$

Therefore the weight update can be given by the following equation,

$$w(n+1) = w(n) + \mu x(n)[d^*(n) - x^h(n)w(n)]$$
$$= w(n) + \mu x(n)e^*(n)$$

The LMS algorithm is initiated with an arbitrary value w(0) for the weight vector at n=0. The successive corrections of the weight vector eventually leads to the minimum value of the mean squared error.

Therefore the LMS algorithm can be summarized in following equations;

$$Output,\ y(n) = w^h x(n)$$

$$Error,\ e(n) = d^*(n) - y(n)$$

$$Weight,\ w(n+1) = w(n) + \mu x(n)e^*(n)$$

5 Simulation Results:

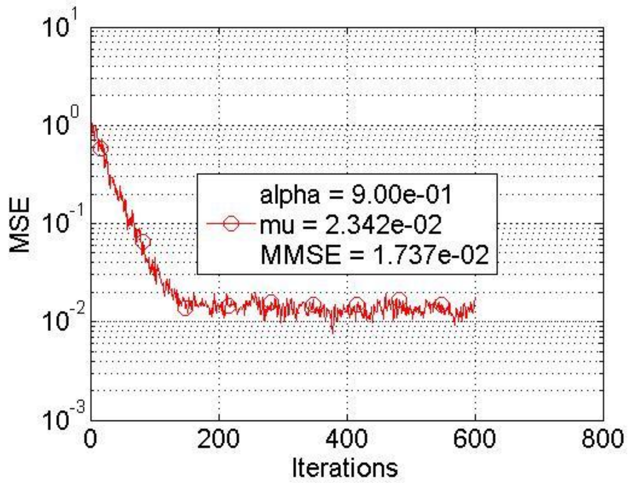


Figure 5.1

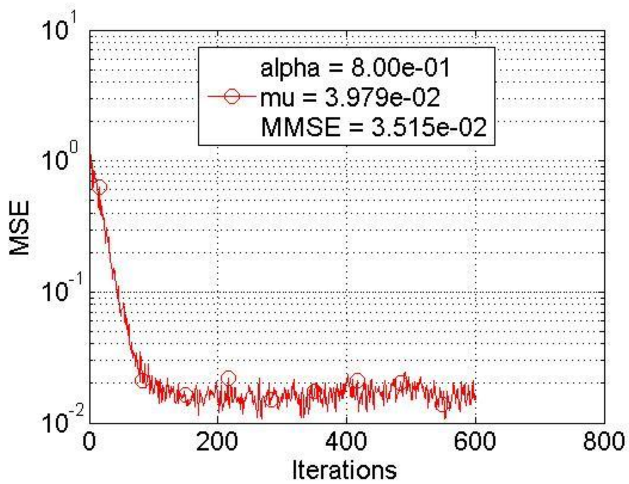


Figure 5.2

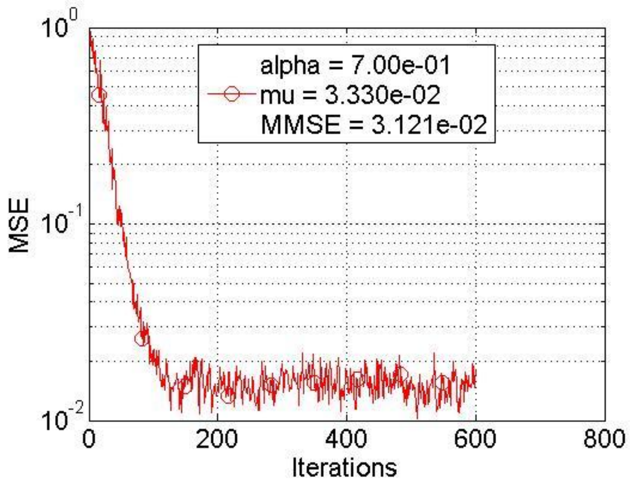


Figure 5.3

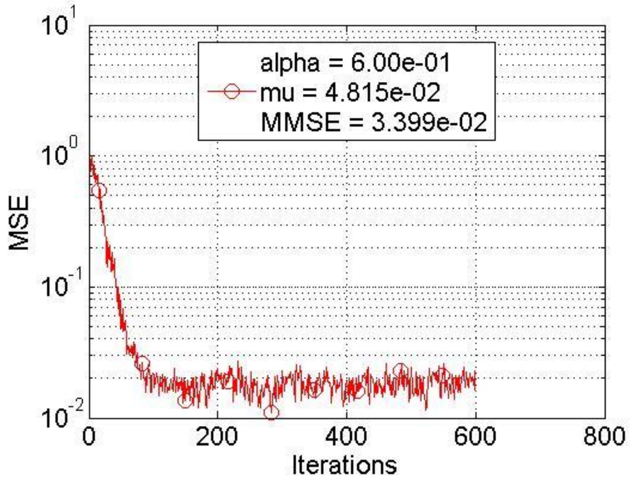


Figure 5.4

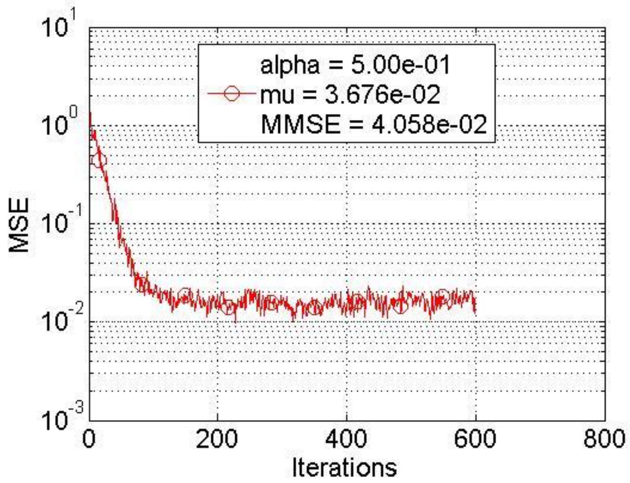


Figure 5.5

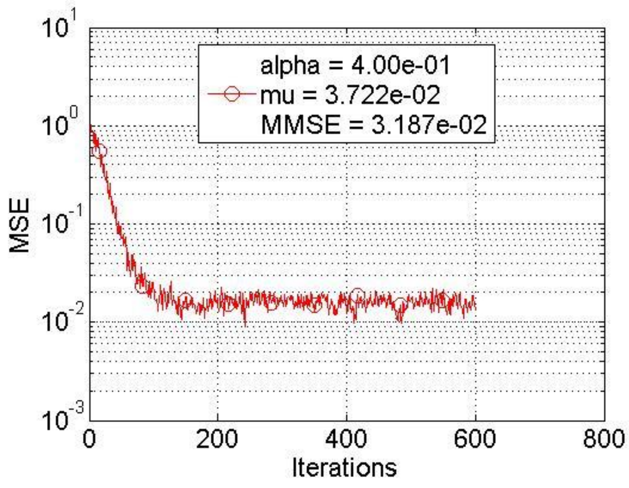


Figure 5.6

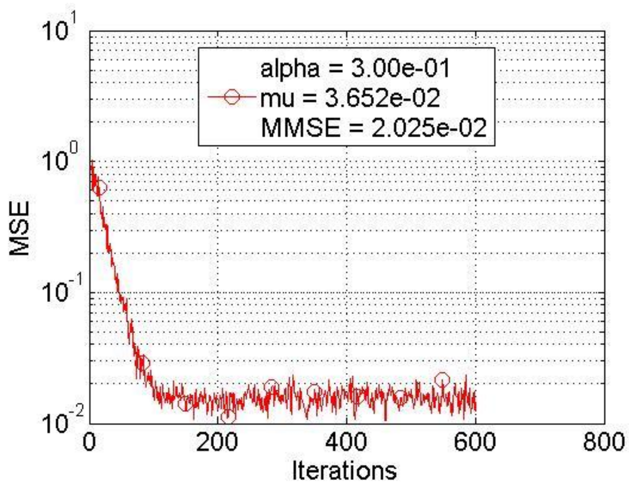


Figure 5.7

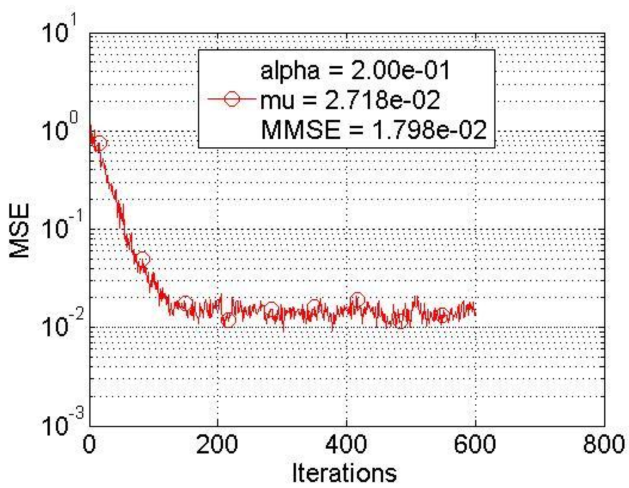


Figure 5.8

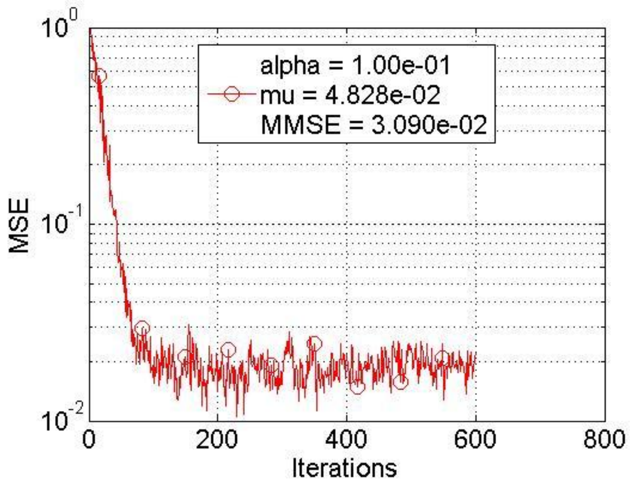


Figure 5.9

Figure 5.1-5.9 showing the simulation result of MSE for the different values of alpha and mu.

Table -1: Analysis chart for ACO

S.N.	Alpha	mu	MMSE
ACO1	0.9	0.02342	0.01737
ACO2	0.8	0.03979	0.03515
ACO3	0.7	0.0333	0.03121
ACO4	0.6	0.04815	0.03399
ACO5	0.5	0.03676	0.04058
ACO6	0.4	0.03722	0.03187
ACO7	0.3	0.03652	0.02025
ACO8	0.2	0.02718	0.01798
ACO9	0.1	0.04828	0.03090

Table -2: Analysis chart for PSO

S.N.	Alpha	mu	MMSE
PSO1	0.9	0.02787	0.01471
PSO2	0.8	0.04588	0.04027
PSO3	0.7	0.02634	0.02277
PSO4	0.6	0.02797	0.01716
PSO5	0.5	0.04708	0.0889
PSO6	0.4	0.04148	0.0341
PSO7	0.3	0.04743	0.02126
PSO8	0.2	0.03675	0.02288
PSO9	0.1	0.0468	0.02767

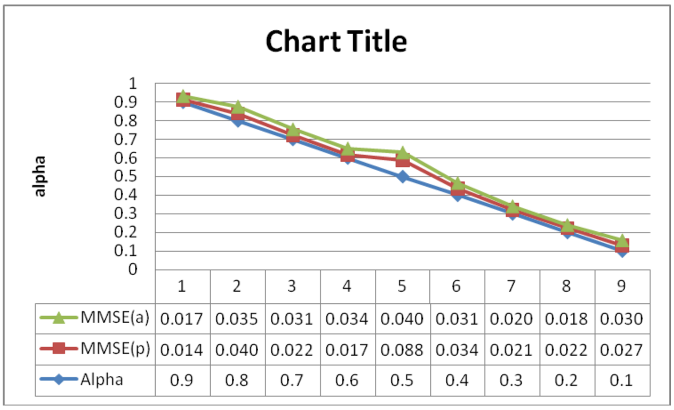


Figure 6.1

6. Conclusion & Future Work

In this paper, a powerful and robust algorithm which is based on ant colonies is proposed to find global minimum. When compared with previous methods to find global minimum such as PSO, results shows that proposed algorithm has a noticeable performance. The proposed ACO based algorithm found exactly the best values.

REFERENCES

[1] Rajni and AkashTayal, "Step Size Optimization of LMS Algorithm Using Particle Swarm Optimization Algorithm in System Identification" International Journal of Computer Science and Network Security, VOL.13 No.6, June 2013.

[2] P. S. R. Diniz, Adaptive Filtering: Algorithms and Practical Implementations, Kluwer Academic Publishers, Boston, 1997.

[3] Kennedy J and Eberhart R C (1995) Particle Swarm Optimization. In Proceedings of the IEEE International Conference Neural Networks, pp 1942-1948.

[4] A.Abraham et al: Swarm intelligence: Foundations, perspectives and applications. Studies in Computational Intelligence(SCT)26,3-25(2006).