# Step Size Optimization of LMS Algorithm Using Particle Swarm Optimization Algorithm in System Identification

**Rajni and AkashTayal,**

Guru Gobind Singh Indrapastha University, Delhi,India

**Summary**
System identification is the art and science of building mathematical models of dynamic systems from observed input-output data .This paper combines Particle Swarm Optimization Algorithm and LMS algorithm to describe the application of a Particle swarm Optimization (PSO) to the problem of parameter optimization for an adaptive Finite Impulse Response (FIR) filter. LMS algorithm computes the filter coefficients and PSO search the optimal step-size adaptively. Because step-size influences on the stability and performance, so it is necessary to apply method that can control it.. However, the statistical Least Mean Squares method is faster than the genetic algorithm. For this reason we suggest using the genetic algorithm for off-line applications, and the statistical method for on-line adaptation. A hybrid method combining the advantages of both methods is proposed for real world applications.

**Keywords**
*FIR, LMS, Particle Swarm Optimization, System Identification*

## 1. Introduction

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 [1]. Therefore, it is desirable to combine the convergent local search capabilities of the LMS algorithm with the global search of PSO.

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 [2]. Therefore, by using a structured stochastic search, such as PSO, 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, Particle Swarm Optimization Algorithm (PSO) to control the value of the step size in accordance with the input adaptively. . This paper introduces a novel algorithm named particle swarm optimization (PSO) to optimize the step size of LMS algorithm and then LMS algorithm calculate system identification parameters adaptively . PSO is a population based search similar to the genetic algorithm (GA) [3]. In adaptive filtering, the mean squared error (MSE) between the output of the unknown system and the output of the AF is the typical cost function, and will hence be used for the fitness evaluation of each particle in the on-line form of PSO. In this we also have taken MSE as cost function.

## 2. Swarm Intelligence

Swarm intelligence describes the collective behavior of decentralized, self organized natural or artificial systems. Swam intelligence model were employed in artificial intelligence. The expression was introduced in the year 1989 by Jing wang and Gerardo Beni in cellular robotic systems. Swarm Intelligence (SI) was a innovative pattern for solving optimizing problems. SI systems are typically made up of populations of simple agents interacting locally with one another and with their environment [4]. The agent follows simple rules and the interactions between agents lead to the emergence of “intelligent” global behavior, unknown to the individual agents.

### 2.1 PSO an Optimization Tool

Particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Dr. Ebehart and Dr. Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling [3]. PSO shares many similarities with evolutionary computation

Manuscript received June 5, 2013
Manuscript revised June 20, 2013

techniques such as Genetic Algorithms (GA). The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles. The detailed information will be given in following sections. Compared to GA, the advantages of PSO are that PSO is easy to implement and there are few parameters to adjust. PSO has been successfully applied in many areas: function optimization, artificial neural network training, fuzzy system control, and other areas where GA can be applied.

## 2.2 Particle Swarm Optimization Algorithm

PSO simulates the behaviors of bird flocking. Suppose the following scenario: a group of birds are randomly searching food in an area. There is only one piece of food in the area being searched. All the birds do not know where the food is. But they know how far the food is in each iteration. So what's the best strategy to find the food? The effective one is to follow the bird, which is nearest to the food. PSO learned from the scenario and used it to solve the optimization problems. In PSO, each single solution is a "bird" in the search space. We call it "particle". All of particles have fitness values, which are evaluated by the fitness function to be optimized, and have velocities, which direct the flying of the particles. The particles fly through the problem space by following the current optimum particles [3].

PSO is initialized with a group of random particles (solutions) and then searches for optima by updating generations. In each iteration, each particle is updated by following two "best" values. The first one is the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called pbest. Another "best" value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the population. This best value is a global best and called g-best. When a particle takes part of the population as its topological neighbors, the best value is a local best and is called p-best.

After finding the two best values, the particle updates its velocity and positions with following equation:

$$\mathbf{V_i^{(u+1)} = \mathit{w}V_i^{(u)} + C_1\, rand_1(\ldots) * (pbest_i - P_i^{(u)}) + C_2\, rand_2(\ldots) * (gbest - P_i^{(u)})} \quad (1)$$

$$\mathbf{P_i^{(u+1)} = P_i^{(u)} + V_i^{(u+1)}} \quad (2)$$

In the above equation,

The term rand ( )*(pbest$_i$ -P$_i^{(u)}$) is called particle memory influence

The term rand ( )*( gbest$_i$ -P$_i^{(u)}$) is called swarm influence.

$V_i^{(u)}$ which is the velocity of I$_{th}$ particle at iteration 'u' must lie in the range

$Vmin \le Vi^{(u)} \le Vmax$

The parameter Vmax determines the resolution, or fitness, with which regions are to be searched between the present position and the target position

• .If Vmax is too high, particles may fly past good solutions. If Vmin is too small, particles may not explore sufficiently beyond local solutions.

• In many experiences with PSO, Vmax was often set at 10-20% of the dynamic range on each dimension.

• The constants C1and C2 pull each particle towards pbest and gbest positions.

• Low values allow particles to roam far from the target regions before being tugged back. On the other hand, high values result in abrupt movement towards, or past, target regions.

• The acceleration constants C1 and C2 are often set to be 2.0 according to past experiences

• Suitable selection of inertia weight 'w' provides a balance between

global and local explorations, thus requiring less iteration on average to find a sufficiently optimal solution.

• In general, the inertia weight w is set according to the following equation

$$\mathbf{W = W_{max} - \left[\frac{W_{max} - W_{min}}{ITER_{max}}\right] x\ ITER} \quad (3)$$

Where w -is the inertia weighting factor

Wmax - maximum value of weighting factor

Wmin - minimum value of weighting factor

ITERmax - maximum number of iterations

ITER - current number of iteration

## 2.3 PSO Flowchart

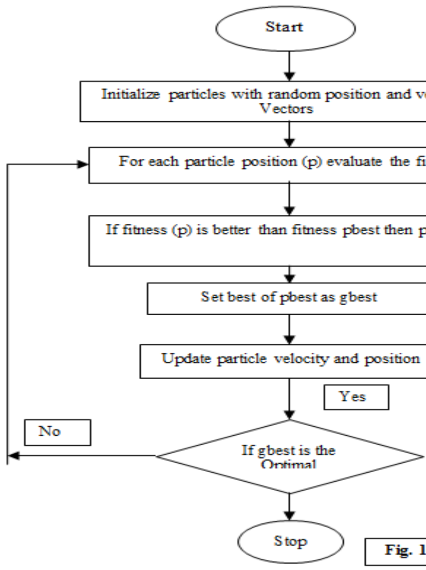



**Fig. 1 PSO Flowchart**

## 2.4 PSO for Adaptive Filtering

In adaptive filtering, the mean squared error (MSE) between the output of the unknown system and the output of the adaptive filter is the typical cost function, and will hence be used for the fitness evaluation of each particle in the on-line form of PSO. For an adaptive system identification configuration the windowed MSE cost function is as follows:

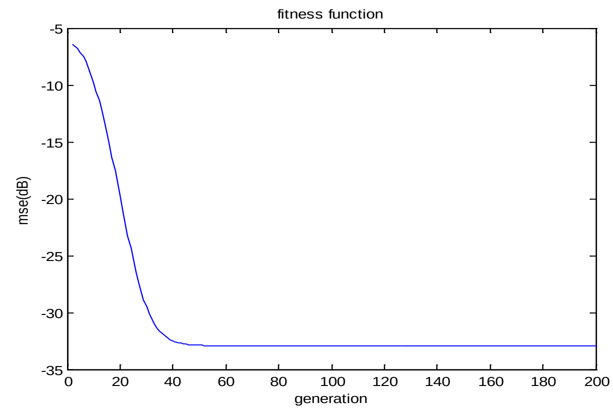
$$J(n) = \min\left(\frac{1}{N}\left[\sum_{k=0}^{N-1}\left(d(n-k) - y_{k,i}(n)\right)^2\right]\right) \quad \textbf{(4)}$$

$$y_{k,i}(n)=f[x(n-k-1), x(n-k-2),\ldots\ldots\ldots\ldots\ldots x(n-k-L)] \quad \textbf{(5)}$$

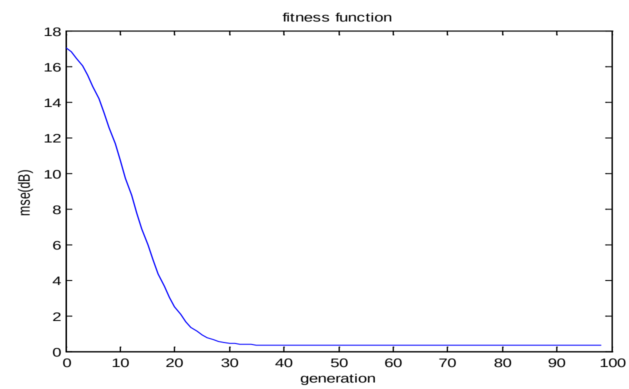
where $f(\cdot)$ is a nonlinear operator, N is the length of the window that the error is averaged and L is the amount of delay in the filter. The adaptive filter output yk(n) may also be a function of past values of itself if it contains feedback, or also a function of intermediate variables if the adaptive filter has a cascaded structure. When J(n) is minimized, the adaptive filter parameters provide the best possible representation of the unknown system.

## 2.5 Population Size

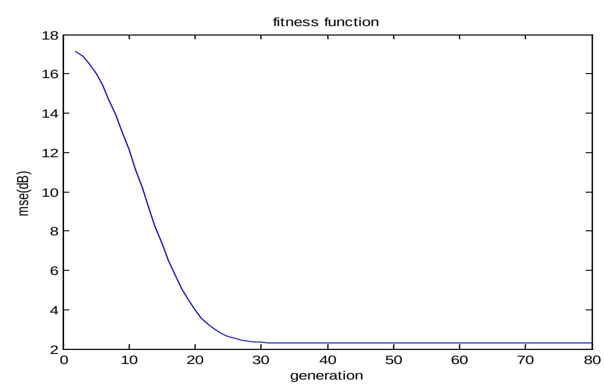
It is obvious that a larger population will always provide a better search and faster convergence on average, regardless of the complexity of the error surface, due to the increased number of estimates evaluated at each epoch. The result given in Figure indicates that an increased search capacity enables the algorithm to converge to a more precise solution. But the computational complexity of the algorithm increases linearly with population size, which is a motivation for other algorithm variants that are robust with smaller populations.



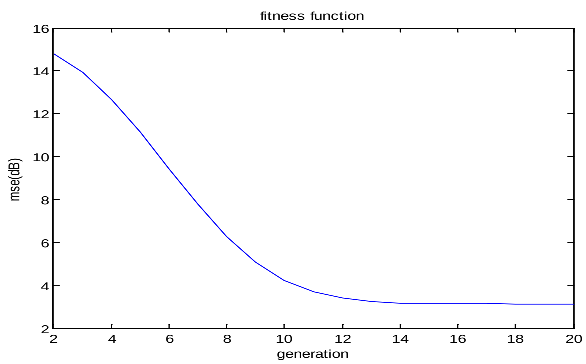

(a) Population 200




(b) Population 100




(c) Population 80




(d) Population 20

Fig. 2 Effect of population Size in PSO

# 3. LMS Algorithm

## 3.1 Basic Principle of LMS Algorithm

The LMS algorithm is a type of adaptive filter known as stochastic gradient-based algorithms as it utilizes the gradient vector of the filter tap weights to converge on the optimal wiener solution [6]. It is well known and widely used due to its computational simplicity. It is this simplicity that has made it the benchmark against which all other adaptive filtering algorithms are judged.

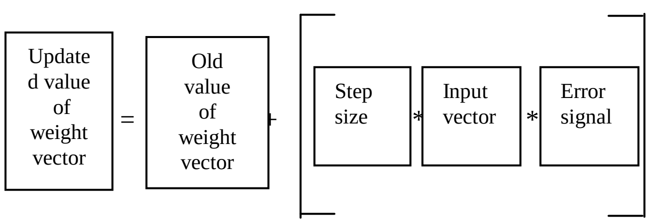



Fig. 3 LMS Algorithm

With each iteration of the LMS algorithm, the filter tap weights of the adaptive filter are updated according to the following formula-

$$W(n+1) = W(n) + 2\mu\, e(n)\, x(n) \quad \textbf{(6)}$$

Here **x** (n) is the input vector of time delayed input values,

$$\mathbf{x}(n) = [x(n)x(n-1) \;\ldots\ldots\ldots\; x(n-N+1)]^T \quad \textbf{(7)}$$

The vector $w(n) = [w_0(n)\ w_1(n)\ w_2(n)\ ..\ w_{N-1}(n)]^T$ represents the coefficients of the adaptive FIR filter tap weight vector at time n. The parameter μ is known as the step size parameter and is a small positive constant. This step size parameter controls the influence of the updating factor. Selection of a suitable value for μ is imperative to the performance of the LMS algorithm, if the value is too small the time the adaptive filter takes to converge on the optimal solution will be too long; if μ is too large the adaptive filter becomes unstable and its output diverges.

# 4. System Model & Fir Filter

## 4.1 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 [7].

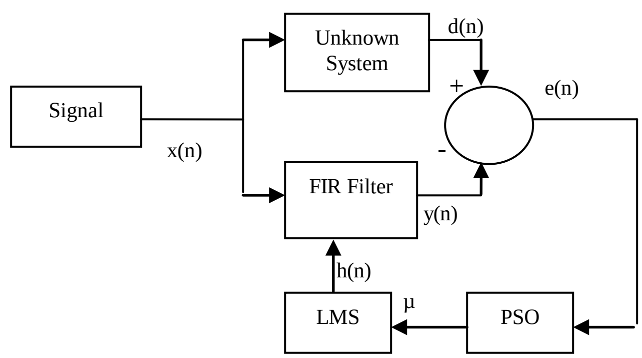



Fig. 4 Proposed Structure

In this paper a new procedure to estimate the coefficients of an adaptive filter is proposed. It combines the Particle Swarm Optimization Algorithm with Least-Mean-Square (LMS) method, i.e. in each iteration of PSO, 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 PSO block and PSO 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 PSO 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 PSO-based procedures. Our proposed approach combines the benefits of both algorithms while accelerates the very slow rate of PSO and escapes from the local minima which may result from LMS [8].

## 5. Simulation Results

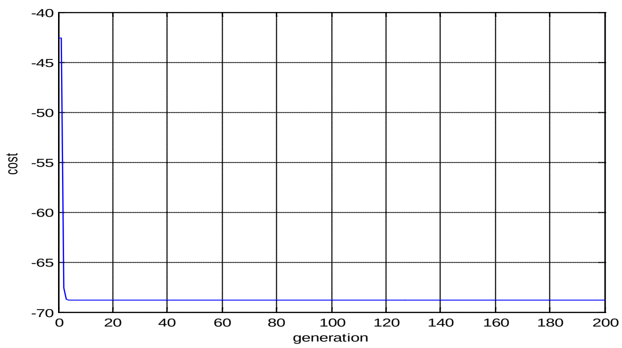



Fig. 5 Fitness function

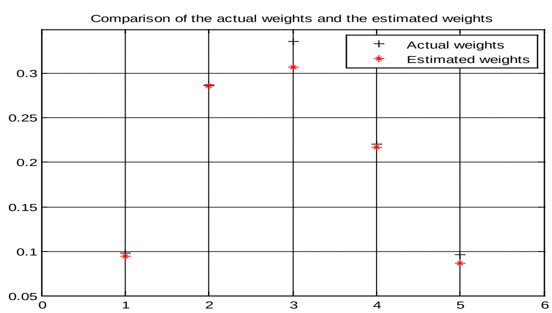



Fig. 6 Comparison of actual weights and the estimated weights using fixed step size LMS& PSO variable step size LMS

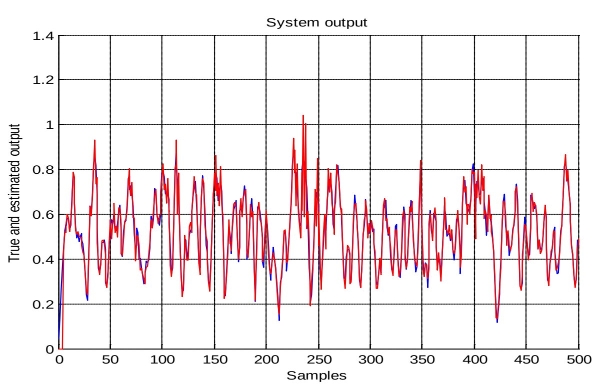



Fig. 7 True & estimated outputs using PSO variable step size LMS

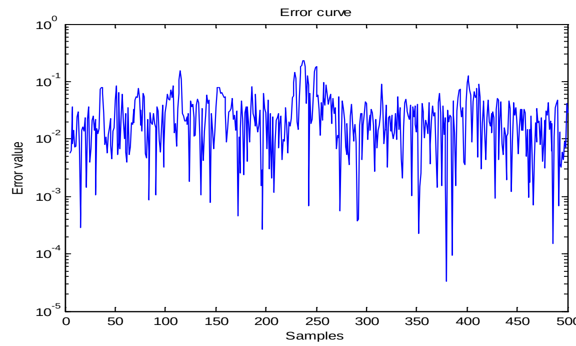



Fig. 8 Error Curve for PSO Variable Step Size LMS

## 6. Conclusion &Future Work

The goal of this work is to expose PSO as a viable approach adaptive filtering problems, as well as to introduce and explore enhancements to the convergence properties of structured stochastic search algorithms in general.

It becomes apparent that PSO is not only competitive with the conventional global search techniques, but are superior in many instances. As for structured stochastic techniques in general, in addition to avoiding local minima, they demonstrate convergence rates that are an order of magnitude faster (per input sample) than existing gradient based techniques.

A topic that warrants further research is an assessment of the performance of these algorithms on modified error surfaces. It is evident that structured stochastic search algorithms perform better on surfaces exhibiting relatively few local minima, because the local attractors offer little interference to the search. By incorporating alternate formulations of the adaptive filter structures or performing data preprocessing, such as input orthognalization and power normalization, the error surface can be smoothed dramatically. These additions will likely result in improved performance of stochastic search algorithms. Related to this notion is the assertion that structured stochastic algorithms, particularly versions of PSO, have a tendency to excel on lower order parameter spaces. This is due partly to the fact that lower dimensional parameter spaces tend to exhibit fewer local minima in general, and because the volume of the hyperspace increases exponentially with each additional parameter to be estimated. Therefore it is hypothesized that dimensionality reduction techniques, such as implementing alternate formulations of adaptive filter structures that are able to accurately model unknown systems with minimum number of parameters, could benefit the overall performance.

In addition, further work is needed to extend and optimize the relationship and coordination of the various algorithm parameters and convergence mechanisms. Finally, the parallel hardware implementation of these algorithms needs to be investigated, which is essential in order for these algorithms to become more widely accepted in practice.

## References


[1] Nambiar, R. and Mars, P., “Genetic and Annealing Approaches to Adaptive Digital Filtering,” Proc. 26th Asilomar Conf. on Signals, Systems, and Computers, vol. 2, pp. 871-875, Oct. 1992.

[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).

[5] Z.Banjac, B.Kovacevic, M.Veinovic and M.Milosavljevic ,“Robust least mean square adaptive FIR filter algorithm” IEEE 2001.

[6] Crina Grosan and Ajith Abraham “Intelligent Systems” Springerliink Reference library,2011, Volume 17,345-386,DOI: 10.1007/978-3-642-21004-4_14

[7] Xiaochun Guan “Qx-LMS Adaptive FIR Filters For System Identification”Image and Siignal Processing,209. CISP ‘9, 2nd IEEE International Conference

[8] Oscar Castillo, Oscar Montie1, Roberto Sepulveda, and Patricia Melin. “Application of a Breeder Genetic Algorithm for System Identification in an Adaptive Finite Impulse Response Filter” .IEEE 2001


**Rajni** was born in 1987 in Rohtak,Haryana. She received her B.Tech degree in 2008 in Electronics and Communication Engineering from Sh. Baba Mastnath College of Engineering,Maharishi Dayanand University ,Rohtak,Haryana. She is currently a student of Master’s degree program at Indira Gandhi Institute of Technology, Guru Gobind Singh Indraprastha University, Delhi, India. She has 4 years of teaching experience.

Her research interest includes Metaheuristics,, Signal Processing, Image Processing.

**Akash Tayal** received his B.Tech degree from Jamia Millia Islamia, Delhi, India and M.Tech degree from Netaji Subhash Institue of Technology, Delhi University, Delhi, India in 2002.

He has over 9 years teaching experience. He is presently working with Department of Electronics and Communication Engineering, Indira Gandhi Institute of Technology, Guru Gobind Singh Indrapastha University, Delhi, India as Assistant Professor.

He has published around 20 research papers in reputed Journals and International Conference. His research interest includes Digital Image Processing, Optimization, Nonlinear Statistical processing and Game Theory.