Adaptive Nonlinear System Identification using Comprehensive Learning PSO

Venkatesh Katari*, Satish Malireddi*, Satya Kanth S.Bendapudi*, G. Panda**, *Senior Member, IEEE,*
Venky.kv@gmail.com satish.malireddi@gmail.com bssatyakanth@gmail.com ganapati.panda@gmail.com
*GITAM University, **NIT Rourkela

***Abstract*-In this paper we introduce the Comprehensive Learning Particle Swarm Optimization (CLEPSO) technique for identification of nonlinear systems. System identification in noisy environment has been a matter of concern for researchers in control theory for nonlinear analysis and optimization. In the recent past the Least Mean Square Algorithm (LMS), Genetic Algorithm (GA), Particle Swarm Optimization (PSO) etc. have been employed for developing mathematical archetype of an anonymous system. LMS performs inversely with nonlinearity. Although PSO performs better than GA in terms of convergence rate, it suffers from premature convergence. To alleviate the problem we propose a novel CLEPSO technique for updating the parameters of the Functional Link Artificial Neural Network (FLANN) model. The CLEPSO is a variant of PSO which ascertains the convergence of the model parameters to the global optimum with a faster speed and better accuracy. Comprehensive computer simulations corroborate that CLEPSO is a better parameter updating algorithm than PSO even in noisy conditions, both in terms of accuracy and convergence speed.**

I. INTRODUCTION

System Identification plays an important role in uncertain control systems. The traditional Least Mean Square algorithm [1,2] is well suited for identification of linear static systems. However in practice most of the systems are nonlinear and dynamic. The conventional linear approaches do not achieve very satisfying performance which leads to the development of nonlinear approaches and advanced methods

Artificial neural networks are universal approximators and have strong mapping capability which is well suited for non-linear classification problems. Many neural networks have been applied to deal with nonlinear system identification such problems, such as multilayer perceptron (MLP), Radial Basis Function (RBF) networks and Recurrent neural networks etc. To improve the identification performance of nonlinear systems various techniques such as ANN [3,4], evolutionary algorithms (EA) (such as GA, PSO) [5,6] have been reported in the literature. Unlike other optimization techniques, the EAs are a population-based search algorithms, which work with a population of chromosomes or particles that represent different potential solutions. Therefore, EAs have inherent parallelism that improves their exploration and the optima can be located more precisely. Some researchers have applied PSO technique [7,8] to identify the nonlinear systems with higher convergence rate. PSO is also a population based algorithm which ensures the convergence of model parameters to the global optimum. PSO offers faster convergence during training and computationally involves low complexity as compared to GA. There exists tradeoff between model accuracy and complexity in the identification problem. The system model optimized under the specific criterion is not always the optimal model because there are usually several demands to a system model. For example, it required that the model should be easy to handle and well explainable for the modeling data set contaminated by observation noise, but these properties are mutually exclusive. So our model should be adaptive to various noise ranges and hence the weight updating algorithm should be able to handle high nonlinearity and should also have high convergence rate. For this purpose the comprehensive learning PSO (CLEPSO)[10] algorithm is proposed to be employed . Performance of CLEPSO based model is compared with its PSO counterpart using standard nonlinear systems. Simulation results exhibit that the CLEPSO perform better than that of PSO in terms of speed and accuracy.

The organization of the present work is embodied in the following sections. Section II introduces the identification problem. Section III deals with the FLANN model. The basic principles of PSO and CLEPSO have been dealt in Section IV and V respectively. The algorithm required for identification of the models using CLEPSO is developed and presented in Section VI. To validate the performance of the model the simulation study of different nonlinear systems is carried out in Section VII. Finally the conclusion of the proposed investigation is outlined in Section VIII

II. ADAPTIVE SYSTEM IDENTIFICATION

The essential and principal property of adaptive system is its time-varying, self-adjusting performance. An adaptive automaton is a system whose structure is alterable or adjustable in such way that its behavior or performance (according to some desired condition) improves through contact with its environment.

A system identification structure is shown in Fig.1. The system impulse response is represented by h(n). The block labeled N.L. represents nonlinearity associated with the system. White Gaussian noise q(n) is added with nonlinear output. The desired output d(n) is compared with the estimated output y(n) of the identifier. The adaptive algorithm uses the error e(n) for updating the weights of the

978-1-4244-1688-2/08/$25.00 ©2008 IEEE

identifier model. The model is trained until the error becomes minimum. At this stage the correlation between input signal and error signal is minimum.

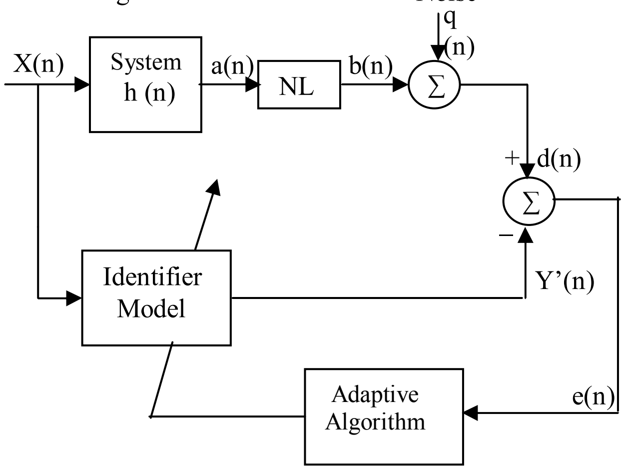

Fig.1. System Identification structure

III.FUNCTIONAL LINK ANN

The block diagram of FLANN structure[9] is shown in Fig.2.The functional link of these structure maps the input signal vector $\mathbf{X_n}$ into N linearly independent functions $\mathbf{\Phi}(\mathbf{X_n})=[\ \varphi_1(\mathbf{X_n})\ \varphi_2(\mathbf{X_n})\ \varphi_3(\mathbf{X_n})\ldots\ldots\ \varphi_N(\mathbf{X_n})]^T$. In this case we have chosen $\varphi_1,\ \varphi_2,\ \varphi_3 \ldots\ldots \varphi_N$ as trigonometric functions such as $\sin(\pi x),\cos(\pi x)$, $\sin(2\pi x),\cos(2\pi x)\ldots\ldots \sin(p\pi x),\cos(p\pi x)$ where p is an integer. The major difference between the hardware structures of MLP and FLANN is that FLANN has only input and output layers and the hidden layers are completely replaced by the nonlinear mappings. In fact, the task performed by the hidden layers in an MLP is carried out by functional expansions in FLANN. The trigonometric expansion transforms the linearly non separable problems in the original low-dimensional signal space into separable one in a high-dimensional space.

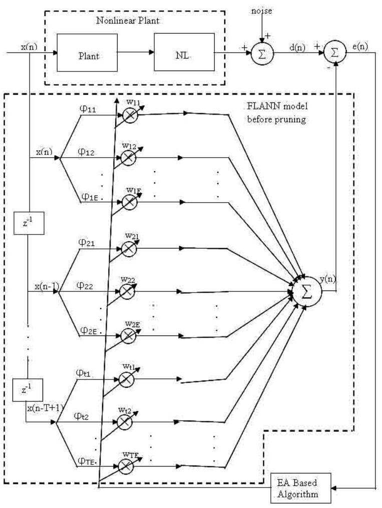

Fig 2: FLANN Structure

IV PARTICLE SWARM OPTIMIZATION

Particle swarm optimization (PSO) technique involves simulating social behavior among individuals (particles) "flying" through a multidimensional search space, each particle representing a single intersection of all search dimensions. The particles evaluate their positions relative to a goal (fitness) at every iteration, and particles in a local neighborhood share memories of their "best" positions, use those memories to adjust their own velocities, and thus subsequent positions. The original PSO formulae (1),(2) define each particle as a potential solution to a problem in D-dimensional space, with particle i represented $Xi=(x_{i1},x_{i2},\ldots,x_{iD})$.Each particle also maintains a memory of its previous best position, $Pi=(p_{i1},p_{i2},\ldots,p_{iD})$, and a velocity along each dimension, represented as $Vi=(v_{i1},v_{i2},\ldots,v_{iD})$. At each iteration t, the P vector of the particle with the best fitness in the local neighborhood, designated $P_{gbest}(t)$, and the P vector of the current particle($P_i(t)$) are combined to adjust the velocity along each dimension, and that velocity is then used to compute a new position for the particle(i.e. $X_i(t+1)$). The portion of the adjustment to the velocity influenced by the individual's previous best position is considered the cognition component, and the portion influenced by the best in the neighborhood is the social component. w is the inertia constant.

$$v_i(t+1)=w*v_i(t)+c_1 rand\left(P_i(t)-x_i(t)\right)+c_2 rand\left(P_{gbest_i}(t)-x_i(t)\right) \quad \ldots\ldots\ldots\ldots\ldots (1)$$

$$x_i(t+1)=x_i(t)+v_i(t+1) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

where the constants c1 and c2 determine the relative influence of the social and cognitive components, and are usually both set the same to give each component equal weight as the cognitive and social learning rate.

V. COMPREHENSIVE LEARNING PARTICLE SWARM ALGORITHM

CLEPSO has been introduced by J.J.Liang, A. K. Qin [10] and is specifically aimed at overcoming the problem of premature convergence. Here, the new velocity of each particle can be updated on the basis of the pbest information of any particle within the swarm, chosen according to a specific logic, which helps in preserving diversity within the swarm that should potentially discourage premature convergence. In this algorithm, the particle's velocity is updated according to the followingequation:

$$V_i^d = w*V_i^d + c*rand_i^d*(pbest^d_{f_i(d)} - x^d_i) \ldots\ldots\ldots (3)$$

Where $f_i=[f_i(1), f_i(2),\ldots\ldots f_i(D)]$ defines which particles' pbest that the particle i should follow. $pbest^d_{f_i(d)}$ is the corresponding dimension of any particle's pbest including its own pbest, and the decision depends on probability Pc referring to the learning probability which can take different values for different particles. For each dimension of particle i we generate a random number .If this random number is larger than Pc, the corresponding dimension will learn from its own *pbest*; otherwise it will learn from another particle's *pbest* .We employ the tournament selection procedure when the particle's dimension learns from another particle's as follows:

1) Randomly choose two particles out of the population which excludes the particle whose velocity is updated.
2) Compare the fitness values of these two particles' *pbest*s and select the better one.
3) Use the winner's *pbest* as the exemplar to learn from for that dimension. If all exemplars of a particle are its own *pbest*, we will randomly choose one dimension to learn from another particle's *pbest*'s corresponding dimension.

All these *pbest* can generate new positions in the search space using the information derived from different particles' historical best positions.

VI.CLEPSO & PSO BASED LEARNING (PARAMETER UPDATES)

Learning is an iterative process in which the parameters of the model are updated to interpolate or approximate a continuous multivariate function in accordance with some algorithms like LMS, PSO etc. The model is trained using the following steps

(1)Generate K number of input(x)-output(d) training patterns which are required to learn the network uniformly distributed between -0.5 to 0.5 with variance of 1/12 and mean 0.

(2)Each input pattern (x) is functionally expanded by choosing a set of specific basis functions (φ). In our present research we have used X_k, $\sin(X_k)$, $\cos(X_k)$ (so N=3) and the expanded samples are connected to the single summation unit through multiplying weights (w). Trigonometric functions are continuous and they properly distribute the non-linearity.

(3) Each of the input samples is passed through the original plant(b(n)) and the output is added with the measurement noise of known strength. The resultant signal acts like the desired signal. In this way K number of desired signals is produced by feeding all the K input samples.

(4) Each of the input samples is also passed through the model using each particle as model parameters Thus in each case M sets of K estimated outputs are obtained.

(5) Each of the desired output is compared with corresponding estimated output and K errors are produced. The mean square error (MSE) for a set of parameters (corresponding to mth particle) is determined by using the relation.

$$MSE(n) = \frac{\sum_{i=1}^{K} e^2{}_i}{K} \quad \text{...} \quad (4)$$

This is repeated for M times.

(6) Equation (4) represents the fitness function for the particles. The particles are updated using equations (1), (2) for PSO and (3) for CLEPSO respectively and respective algorithms is executed.

(7)In each generation the minimum MSE is plotted against generation to obtain the learning characteristics. Learning is stopped when minimum MSE levels are reached and no further decrease is observed.

VII. SIMULATION STUDY

In this section we carry out the simulation study of CLEPSO based identification system. The block diagram of Fig.1 is simulated where the coefficients of the FLANN model is updated using LMS, PSO and CLEPSO. For this, the algorithm proposed in section-VI is used in the simulation. While training, the additive noises used in the channel are -30dB (low noise),-20 db and -10dB (high noise) to assess the performance of the three different algorithms in different noise conditions. Finally the performance of proposed model is obtained by comparing their responses when they are provided with freshly generated random input.

The following nonlinear channel models are used in the simulation study:

Example-1: The impulse response of the linear system of the plant is [0.2600, 0.9300, 0.2600] and nonlinearity associated is

$$y_n(k) = Tanh(y(k))$$

Example-2: Parameters of the linear system of the plant [0.2600, 0.9300, 0.2600] and nonlinearity associated is

$$y_n(k) = y(k) + 0.2y(k)^2 - 0.1y(k)^3$$

Example-3: Parameters of the linear system of the plant [0.341, 0.8760, 0.3410] and nonlinearity associated is

$$y_n(k) = y(k) + 0.2y(k)^2 - 0.1y(k)^3 + 0.5Cos(y(k))$$

Where y(k) is the output of the linear part of the plant and $y_n(k)$is the output of the overall system.

The convergence characteristics for CLEPSO and PSO obtained from simulation study are shown in Figs.4 (a, b, c), 4(d, e, f), 4(g, h, i) for Example-1, Example-2 and Example-3 respectively. Finally the responses of CLEPSO identifier and PSO identifier are compared during testing phase and shown in Figs.3 (a, b, c), 3(d, e, f), 3(g, h, i) for Example-1, Example-2 and Example-3 respectively .It is evident from these plots that the proposed CLEPSO based model converges faster and to the lower noise floor level than those obtained by PSO and LMS based models. Table.1shows the minimum of the mean squared errors for the algorithms at various noise levels.Table.2 provides the number of error evaluations and CPU time taken by the algorithms when they are implemented in the same computer under similar conditions.

Table 1: Mean Squared Errors

	NOISE	LMS	PSO	CLEPSO
Example 1	10 db	0.0124	0.0066	0.0038
	20 db	0.0012	0.0007	0.0006
	30 db	0.0004	0.0002	0.0001
Example 2	10 db	0.0153	0.0067	0.0045
	20 db	0.0037	0.0024	0.0011
	30 db	0.0006	0.0004	0.0002
Example 3	10 db	0.0359	0.0165	0.0098
	20 db	0.0201	0.0048	0.0044
	30 db	0.0174	0.0032	0.0030

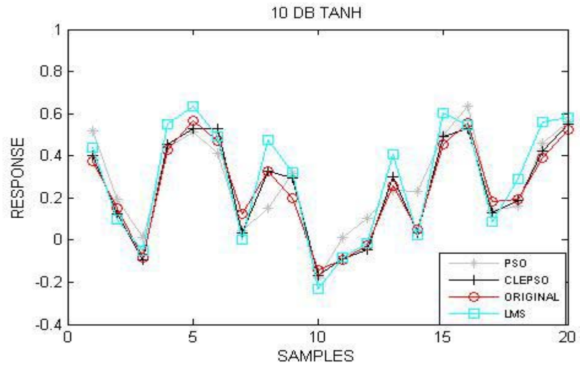

Fig 3(a): Response Curve for example 1 at 10 db Noise

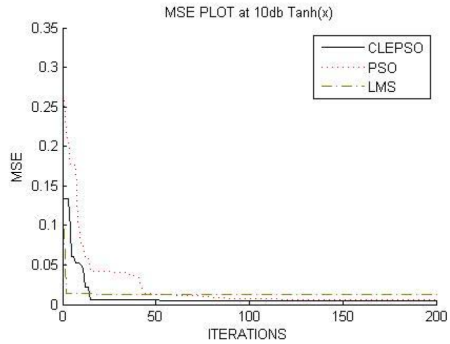

Fig 4(a): Convergence of example 1 at 10 db Noise

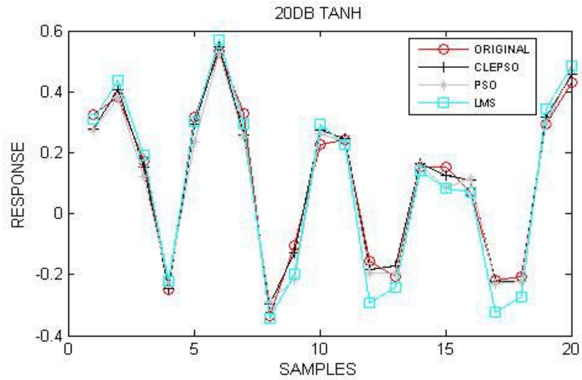

Fig 3(b): Response Curve for example 1 at 20 db Noise

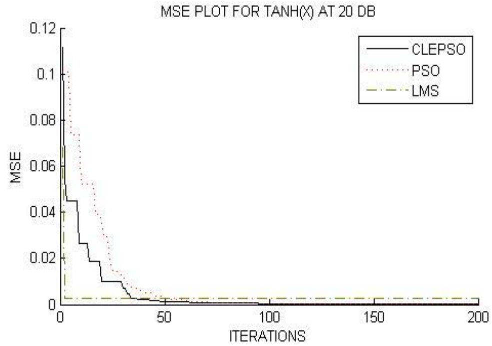

Fig 4(b): Convergence of example 1 at 20 db Noise

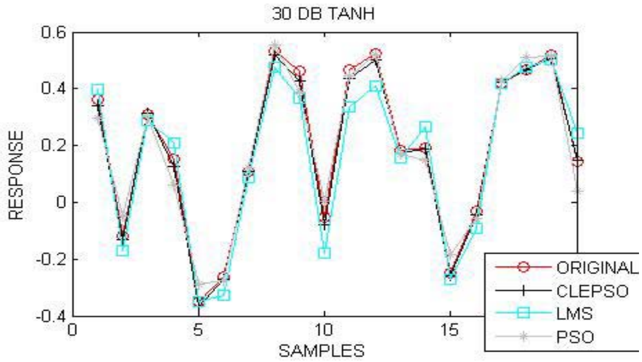

Fig 3(c): Response Curve for example 1 at 30 db Noise

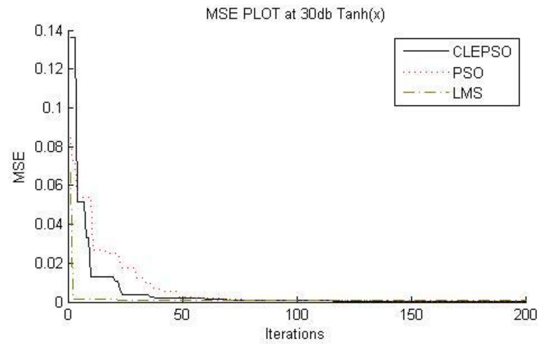

Fig 4(c): Convergence of example 1 at 30 db Noise

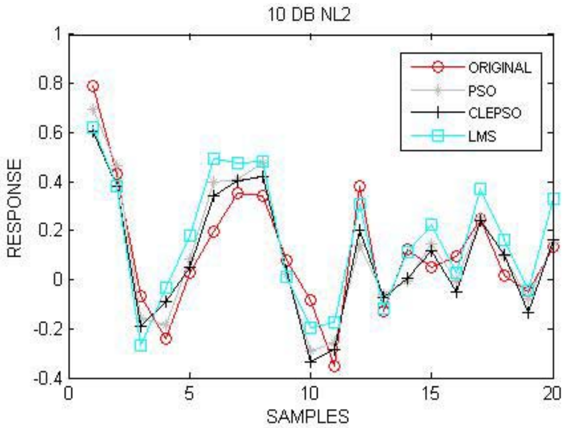

Fig 3(d): Response Curve for example 2 at 10 db Noise

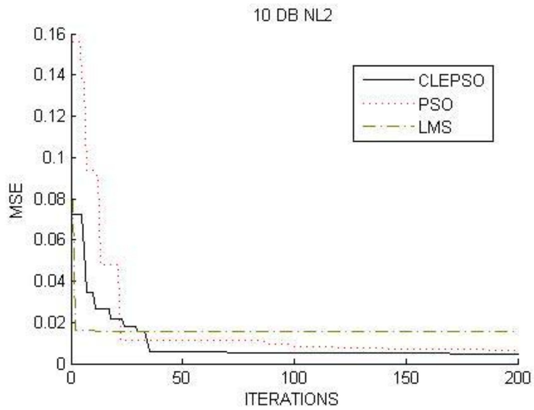

Fig 4(d): Convergence of example 2 at 10 db Noise

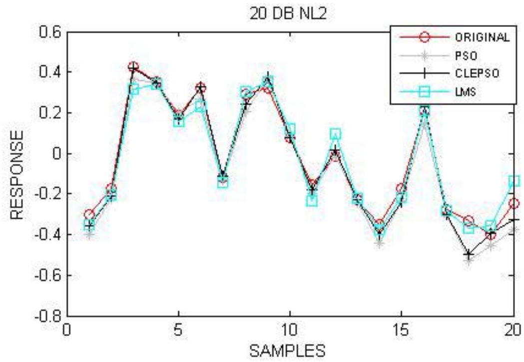

Fig 3(e): Response Curve for example 2 at 20 db Noise

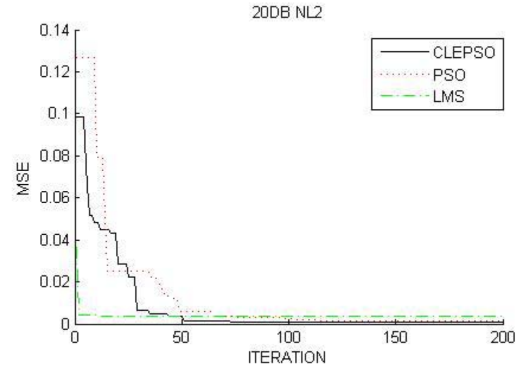

Fig 4(e): Convergence of example 2 at 20 db Noise

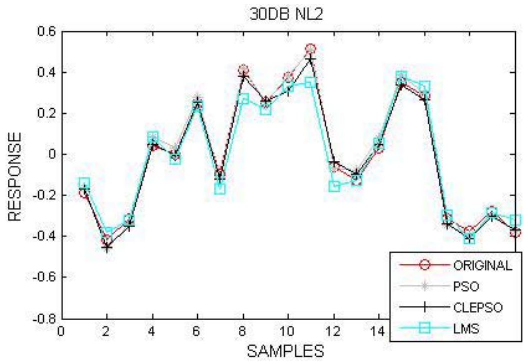

Fig 3(f): Response Curve for example 2 at 30 db Noise

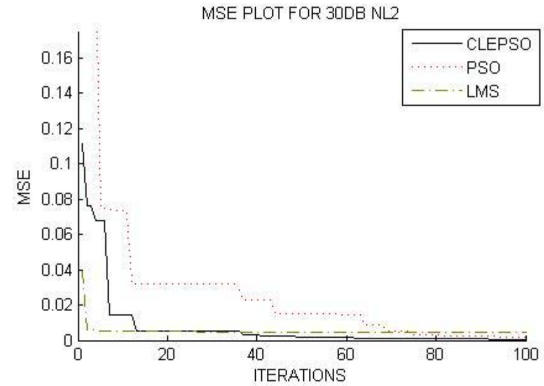

Fig 4(f): Convergence of example 2 at 30 db Noise

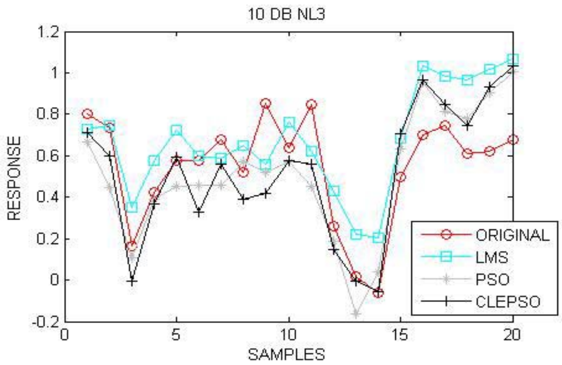

Fig 3(g): Response Curve for example 3 at 10 db Noise

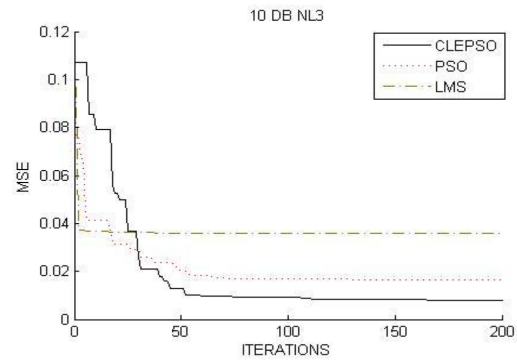

Fig4 (g): Convergence of example 3 at 10 db Noise

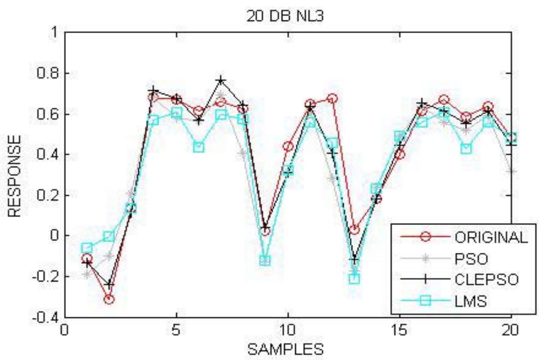

Fig 3(h): Response Curve for example 3 at 20 db Noise

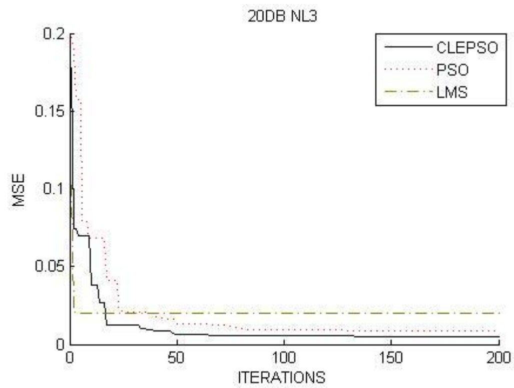

Fig 4(h): Convergence of example 3 at 20 db Noise

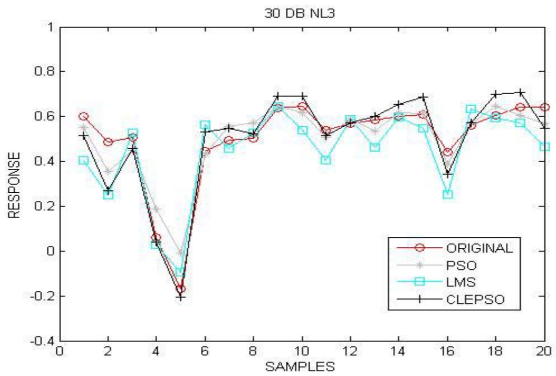

Fig 3(i): Response Curve for example 3 at 30 db Noise

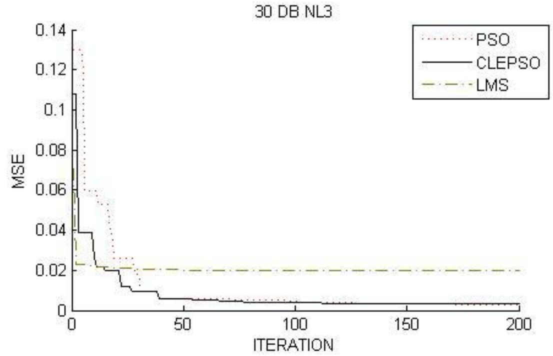

Fig 4(i): Convergence of example 3 at 30 db Noise

Table 2: Comparision of CPU times

Non Linear System	PSO (in sec)	PSO error evaluations	CLEPSO (in sec)	CLEPSO error evaluations
Ex-1 at 10 db	1.175	45000	1.31	37500
Ex-1 at 20 db	0.94	36000	1.05	30000
Ex-1 at 30 db	1.41	54000	0.84	24375
Ex-2 at 10 db	0.94	36000	0.568	16875
Ex-2 at 20 db	1.2925	49500	0.909	26250
Ex-2 at 30 db	1.0575	40500	0.985	28215
Ex-3 at 10 db	1.645	63000	0.647	18750
Ex-3 at 20 db	0.94	36000	0.979	28125
Ex-3 at 30 db	1.175	45000	0.945	28125

Careful observations of Table-1 infers that at low noise levels all the 3 algorithms PSO,CLEPSO and LMS are performing equally, but not so at high noise condition. The PSO is able to perform better than LMS but not better than CLEPSO. So one can clearly observe from the above graphs and Table-1 that CLEPSO out performs the PSO and LMS. Tables -2 indicates that the CPU time taken by CLEPSO is better compared to PSO in most of the cases except for Ex-1 at 10db,20db and Ex-3 20db ,but in these cases CLEPSO is able to achieve a better MSE value than PSO. However in all cases the number of error calculations of CLEPSO based model is less compare to that offered by standard PSO model.

VIII. Conclusion

This paper has proposed the use of a new learning algorithm(CLEPSO) for updating the weights of a FLANN model meant for nonlinear system identification. Use of this learning tool has resulted an efficient identification of nonlinear plants. Simulation study clearly that the CLEPSO based method achieves better results than its PSO and LMS counterparts, both in terms of speed and minimum MSE

References

[1] S. Haykin, *Neural Networks*, Ottawa, ON Canada: Maxwell Macmillan, 1994.

[2] B. Widrow and S. D. Sterns, "*Adaptive Signal Processing*", Pearson Education, pp. 22, Inc. 1985.

[3] Brouwn , G.G ; Krijgsman , A.J, "Hybrid Neural networks for nonlinear system identification, IEEE International conference On Control, volume 1, 21-24 March 94

[4] Chen , J.R ; Mars , P , 'The feasibility of using MLP networks for system identification' IEE colloquium on Neural Networks for systems: Principles and applications, Pages: 3/1 – 3/3 , 25 Jan 1991.

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

[6] Kumon, T. Iwasaki, M., Suzuki, T., Hashiyama, T.; Matsui, N., Okuma, S., "Nonlinear system identification using Genetic algorithm" Industrial Electronics Society, IECON 2000. 26th Annual Conference of the IEEE - Volume 4, 22-28 Oct. 2000 Page(s):2485 - 2491 vol.4, 2000.

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

[8] Ge Hongwei, Liang Yanchun, "Identification for non-linear systems based on particle swarm optimization and recurrent neural network [ultrasonic motor control applications] " , Proc. International conference on communications, circuits and systems, Vol. 2, 27-30 May 2005 Page(s).

[9] Y.H. Pao,Adaptive Pattern Recognition and neural network, Reading,MA,Addison Wesley,1989,chapter 8,pp.197-222

[10] J. J. Liang, A. K. Qin, Ponnuthurai Nagaratnam Suganthan,S. Baskar , "Comprehensive Learning Particle Swarm Optimizer for Global Optimization of Multimodal Functions'', IEEE Transactions on evolutionary computation

[11] J. Sjoberg, Q. Zhang et.al. : Nonlinear black-box modeling in system identification : a unified overview, *Automatica*, Vol.31, No.12, pp.1691-1724, 1995.

[12] T. A. Johansen : Multi-objective identification of FIR models, *Proceedings of IFAC Symposium on System Identification SYSID2000*, Vol.3, pp.917-922, 2000.

[13] J. Holland : *Adaption in Natural and Artificial Systems*, The University of Michigan Press, MI, 1975.

[14]Hayakawa T, Haddad WM, Bailey JM, Hovakimyan N. Passivity-based neural network adaptive output feedback control for nonlinear nonnegative dynamical systems. IEEE Transactions on Neural Networks 2005; 16(2):387-398.