Nonlinear System Identification using Evolutionary Computing based Training Schemes

Swati Swayamsiddha
School of Electronics Engineering
KIIT University, Bhubaneswar, India

H.Pal Thethi
School of Electronics Engineering
KIIT University, Bhubaneswar, India

ABSTRACT

The present work deals with application of recently developed evolutionary computing based training methods for non-linear system identification problem. Generally, most of the systems are nonlinear in nature. The conventionally used standard derivative based identification scheme does not work satisfactorily for nonlinear systems, which is due to premature settling of the model parameters. To prevent the premature settling of the weights, evolutionary computing based update algorithms have been proposed. In this paper we have compared three popular derivative free evolutionary computing based update algorithms namely Genetic Algorithm (GA), Differential Evolution (DE) and Particle Swarm Optimization (PSO) for identification of nonlinear systems, in terms of convergence graph of cost function over number of iterations. It has been demonstrated that the derivative free population based schemes provided excellent performance for identification of nonlinear systems and they are not trapped in problem of local minima as well.

Keywords

Nonlinear System Identification, Particle Swarm Optimization, Genetic Algorithm, Differential Evolution.

1. INTRODUCTION

The system identification problem can be considered as an optimization task where the idea is to estimate the parameters of the model and minimize the error between the output of known plant and the output of adaptive model [8],[9]. This error is minimized iteratively over a period of time by using update algorithm. Conventionally, derivative based adaptive algorithms such as normalized Least Mean Square (nLMS) and Recursive Least Squares (RLS) have been used to minimize this error signal. But in derivative based algorithms there are chances of error being trapped into local minima. This paper intends to use evolutionary approach for nonlinear system identification and attempts to show how Genetic Algorithm (GA), Differential Evolution (DE) and Particle Swarm Optimization (PSO) can be formulated in minimizing the error signal [1],[2],[3],[4],[5]. The performance analysis of these evolutionary based techniques is also carried out to show the effectiveness of the proposed methodology.

2. MODEL OF NONLINEAR SYSTEM IDENTIFICATION

In system identification a mathematical model of a known plant is created. Earlier system identification has been used in modern communication and control [7],[8],[9]. Generally the system identification problem involves the following considerations: the structure realization and a method of updating the weights of the model. Fig.1 shows the schematic representation of an adaptive system identification process. A random signal $x(k)$ is applied as input of the nonlinear system, the output of this block is $y(k)$. To mimic an ideal system, noise of known strength is added. The output of this block is $y_n(k)$, which is considered to be the output of known plant. Same random input $x(k)$ is also applied to the adaptive system model, thereby producing the output of model $\hat{y}(k)$. The prediction error is then calculated which is mathematically given as $e(k)=y_n(k)-\hat{y}(k)$. This error signal is then used to calculate the mean square error(MSE) function, which serves as the cost function for the evolutionary based update algorithm.

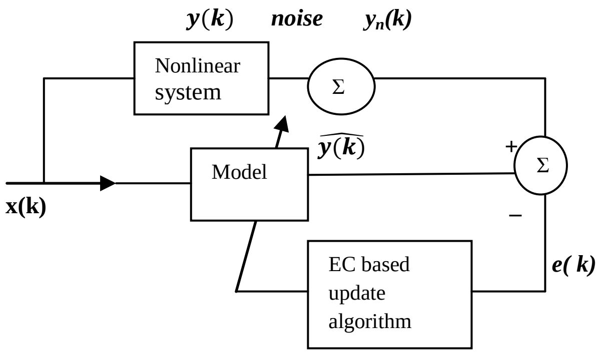

Fig. 1: Adaptive Nonlinear System Identification Scheme

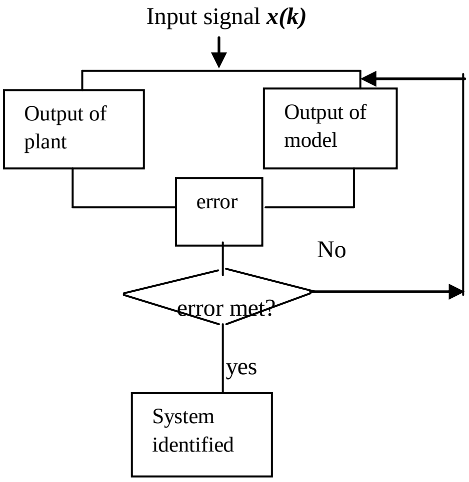

Fig. 2: Flowchart of Adaptive System Identification

3. EVOLUTIONARY COMPUTING BASED UPDATE ALGORITHMS

3.1 Genetic Algorithm

GA is based upon the process of natural selection of "the survival of the fittest". It is a population based search mechanism and it involves the following four major steps: initialization of population, fitness evaluation, selection and reproduction. Initial population is evolved iteratively over and over again till the stopping criterion is met, which is the minimum value of the cost function in identification problems. The crucial parameters involved are population size, crossover and mutation probability and the selection procedure involved [1],[9].

3.2 Differential Evolution

Differential Evolution is an evolutionary method which has been recently used for global optimization and it is efficient and effective optimization tool much like GA, but it is faster as compared to GA [3]. Steps involved in DE are as follows:

1. Initialization: The target vector is randomly initialized and evaluated using the fitness function, which is mean squared error (MSE) in the present work.
2. Differential Mutation: For each target vector a mutant vector is generated, which is achieved by adding the weighted difference between any two randomly chosen target vectors to a third target vector.
3. Crossover Operation: To introduce diversity, which prevents the solution being trapped in local minima, the parameters of mutated vector are interchanged with a predetermined target vector to generate a trial vector.
4. Selection: The child (trial vector) produced replaces the parent (target vector) if it is having lower cost function (MSE), else the parent is passed on to the next generation of the algorithm.

3.3 Particle Swarm Optimization

In particle swarm optimization method the flocking behavior of birds is formulated as an optimization problem. One of the primary advantage of this method is that it has memory, i.e., every particle remembers its best solution which is termed as local best, as well as the group's best solution termed as global best [4],[5].The modifications of the velocity and position of the particle is carried out iteratively by using following equation:

$$V_i^{k+1} = wV_i^k + C_1 \times rand_1 \times (pbest_i - S_i^k) + C_2 \times rand_2(gbest_i - S_i^K) \qquad (1)$$

c_j = weighting factor, w = weighting function, $rand$ is a random number between [0,1], S_i^k = current position of agent i at k^{th} iteration, $pbest_i$ = personal best of agent i, $gbest$ = global best of the group.

$$w = w_{max} - \frac{(w_{max} - w_{min})}{iter_{max}} \times iter \qquad (2)$$

where, w_{max} = initial weight, w_{min} = final weight, $iter_{max}$ = maximum iteration number, $iter$ = current iteration number

Using the above equation, a certain velocity, which progressively gets close to ***pbest*** and ***gbest*** can be calculated [14]. The current position which is equal to searching point in the solution space, which can be modified by the following equation:

$$S_i^{k+1} = S_i^k + V_i^{k+1} \qquad (3)$$

4. EVOLUTIONARY COMPUTING BASED TRAINING SCHEMES FOR WEIGHT UPDATE OF THE MODEL

The updating of the parameters of the model is carried out using evolutionary computing based rule as outlined in the following steps:

- The coefficients of the model are initially chosen from a population of ***M*** chromosomes in GA/DE or from a swarm of ***M*** particles (birds) in PSO. Each chromosome constitutes ***NL*** number of random binary bits where each continuous group of L-bits represents one coefficient of the adaptive model, where ***N*** is the number of parameters of the model. In PSO, each particle constitutes '***p***' number of parameters and each parameter represents one coefficient of adaptive filter.
- Generate ***K*** number of input signal samples each of which is having zero mean and uniformly distributed between-2√3 to +2√3, having unit variance.
- Each of the input samples is passed through the nonlinear system and then contaminated with the white Gaussian noise of known strength. The resultant signal acts like the desired signal. In this way ***K*** numbers of desired signals are produced by feeding all the ***K*** input samples.
- Each of the input samples is also passed through the model using each chromosome (GA/DE) or each particle (PSO) as model parameters and ***M*** sets of ***K*** estimated outputs are obtained.
- 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 is determined by using the relation: $\mathrm{MSE(n)} = \frac{\sum_{i=1}^{k} e_i^2}{k}$. This is repeated for ***M*** times.

- Since the objective is to minimize ***MSE (m), m=1,2,3…, M***, the GA/DE/PSO based optimization is used.
- In GA the cycle of encoding, crossover, mutation and selection operator are serially carried out. The DE is similar to GA except that the differential mutation operation is carried out first followed by crossover. Similarly in PSO the position and velocity updating is done using the equations (1) and (3).
- In each generation the minimum MSE is obtained and plotted against generation to show the learning characteristics.
- The learning process is stopped when MSE reaches minimum level. At this step all the chromosomes attend almost identical genes and particles in GA/DE and PSO respectively and these values represent the estimated parameters of the unknown model.

5. SIMULATIONS AND RESULTS

The simulation study is carried out to demonstrate the identification performance of GA, DE and PSO based training schemes, which is shown in terms of MSE floor. The block diagram shown in Fig.1 is used for simulation study. A unit variance and zero mean random uniform signal is applied to the known system having following transfer function:

$$P(z) = 0.0976 + 0.2873z^{-1} + 0.3360z^{-2} + 0.2210z^{-3} + 0.0964z^{-4} \quad (4)$$

The output of the known plant is then passed through three nonlinear functions and the output of this block is added with white Gaussian noise of known strength of –30dB and -40dB.

NLF1 : $y_n(k) = \tanh(y(k))$

NLF2: $y_n(k) = y(k) + 0.2y^2(k) - 0.1y^3(k)$

NLF3 : $\boldsymbol{y_n(k) = y(k) + 0.2(y(k).^2) - 0.2(y(k).^3) + 0.5cos(\pi.y(k))}$ (5)

where ***y(k)*** is the output of the nonlinear block and ***$y_n(k)$*** is the output after adding noise. The convergence graph of MSE over number of iterations is shown below for all three kinds of update algorithms.

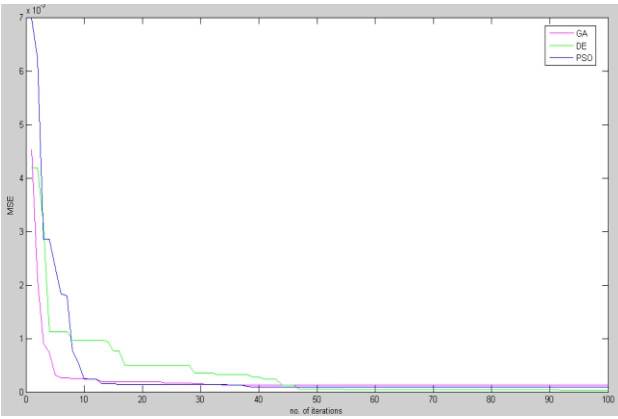

Fig.3 MSE floor for NLF1 at -30dB noise

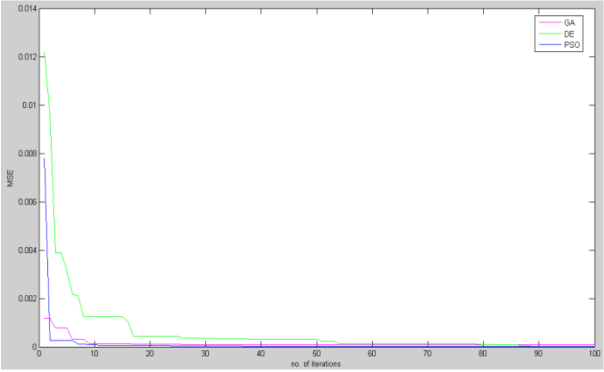

Fig.4 MSE floor for NLF1 at -40dB noise

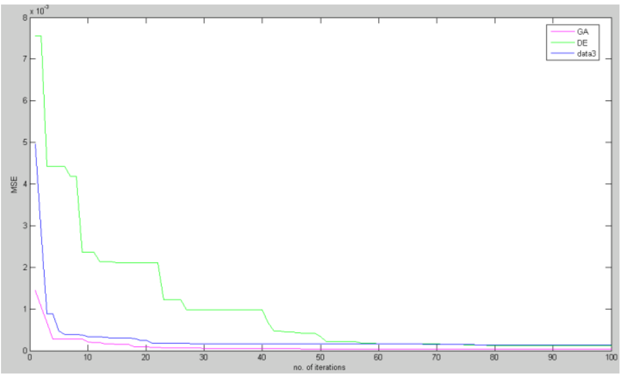

Fig. 5 MSE floor for NLF2 at -30dB noise

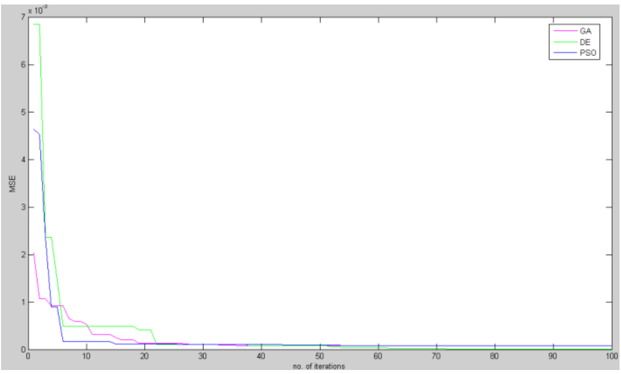

Fig. 6 MSE floor for NLF2 at -40dB noise

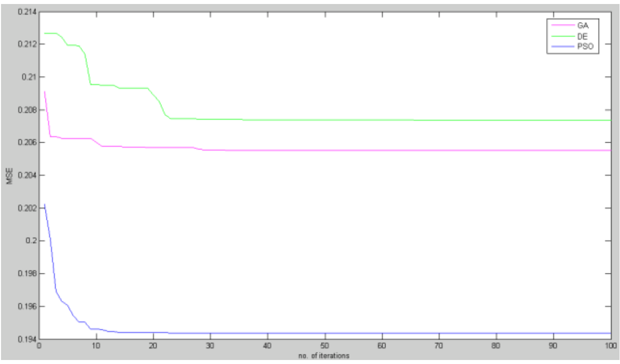

Fig.7 MSE floor for NLF3 at -30dB noise

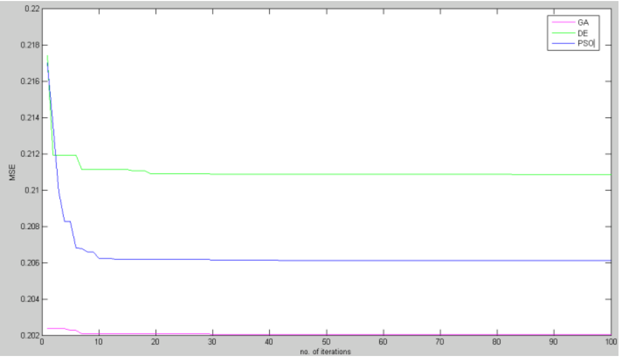

Fig.8 MSE floor for NLF3 at -40dB noise

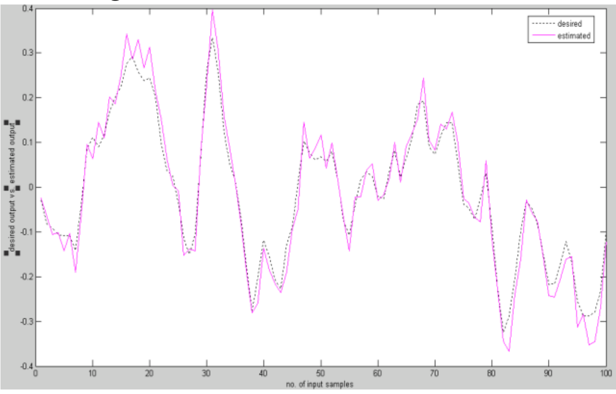

Fig.9 Desired output vs. estimated output using GA based learning algorithm for NLF1 at noise of -30dB

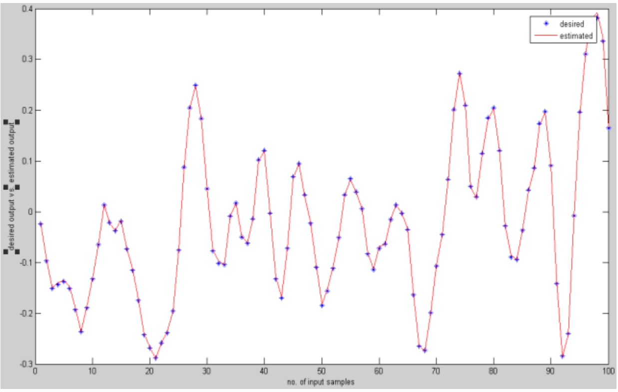

Fig.10 Desired output vs. estimated output using DE based learning algorithm for NLF1 at noise of -30dB

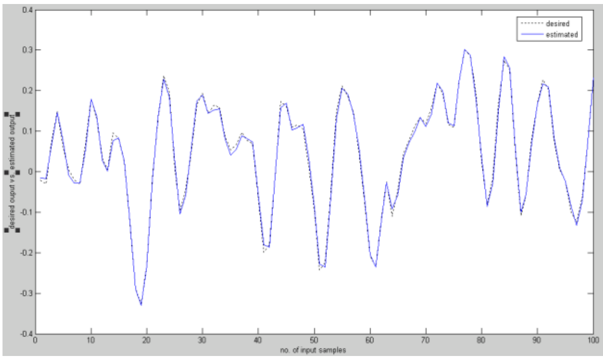

Fig. 11 Desired output vs. estimated output using PSO based learning algorithm for NLF1 at noise of -30dB

The MSE floor in Fig.3 for NLF1 at -30dB noise shows that the GA based learning algorithm is the fastest, but subsequently the DE based learning algorithm shows the best results. For NLF1 at -40dB noise, at Fig. 4 the MSE floor, for all three algorithms is almost identical. For NLF2 at -30dB and -40dB at Fig.5 and 6 respectively confirms that GA based learning algorithm is fastest, but subsequenty DE based learning provides the best identification of plant weights. Fig. 7 shows the MSE floor of NLF3 at -30dB noise and here PSO based learning algorithm provides best results and for -40dB noise GA based learning algorithm provides best results, which is shown in Fig. 8. Fig. 9-11 shows that all these evolutionary based learning algorithms provide satisfactory mapping between desired and estimated signal, though the result provided by DE based learning algorithm is by far the best.

6. CONCLUSION

This paper confirms that the population based stochastic algorithms can be used for nonlinear system identification problem. Mostly, DE based learning algorithm provided the best results of identification, but it takes more time to settle the weights. All these learning schemes can be applied to various other fields such as pattern recognition, forecasting as well as classification tasks.

7. REFERENCES

[1] D.E.Goldberg, "Genetic algorithms in search, optimization and machine learning",AdditionWesley,1989.

[2] A. E. Eiben and J. E. Smith, "***Introduction to Evolutionary Computing***", Springer, ISBN 3-540-40184-9, 2003.

[3] R. Storn and K. Price, "*Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces*", Journal of Global Optimization, vol.11, pp.341-359, 1997.

[4] J. Vesterstroem and R. Thomsen, "A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems," Proc. Congr. Evol. Comput., vol. 2, pp. 1980–1987, 2004.

[5] Particle Swarm Optimization and Differential Evolution Algorithms: Technical Analysis, Applications and Hybridization Perspectives , Swagatam Das, Ajith Abraham, and Amit Konar.

[6] Swagatam Das, Ajit Abraham & Amit Konar, "Particle Swarm Optimization and Differential algorithms: technical analysis, applications & hybridization perspectives" Advances in Computational intelligence in industrial systems, study in computational intelligence, 2008, Vol 116/2008, 1-38, DOI:10.1007/978-3-540-78297-1-1

[7] S. Chen, S. A. Billings and P. M. Grant, :Nonlinear system identification using neural networks", Int. J. Contr., vol. 51, no. 6, pp. 1191-1214, June 1990.

[8] J. Abonyi, J. Madar, L. Nagy, and F. Szeifert, "Interactive Evolutionary Computation in Identification of Dynamical Systems", *Comp.& Chem. Engineering, Carnegie Mellon University*, IF: 0.784, 2004

[9] Kumon, T. Iwasaki, M. Suzuki, T. Hashiyama, T. Matsui, N. Okuma, "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.

[10] S. Haykin, *Adaptive Filter Theory*. Upper Saddle River, NJ: Prentice-Hall Inc., 1996.

[11] N. P. Padhy, *Artificial Intelligence and Intelligent Systems*, chapter-9,10, pp-459-578, Oxford Publications, Second Edition

[12] 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.

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

[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.