Research Paper
Volume : 3 | Issue : 4 | April 2014 • ISSN No 2277 - 8179

Adaptive Inertia Weight Particle Swarm Optimization For Linear and Nonlinear Channel Equalization

Engineering

KEYWORDS : Inter Symbol Interference (ISI), Particle Swarm Optimization (PSO), Evolutionary Algorithm, Mean Square Error (MSE).

Diana D.C	Research Scholar, Department of Electronics Engineering, MIT Campus, Anna University, Chennai – 600044, India
Joy Vasantha Rani.S.P	Assistant Professor (Sr.Gr.), Department of Electronics Engineering, MIT Campus, Anna University, Chennai – 600044, India
Nithya.T.R	B.E. Student, Department of Electronics Engineering, MIT Campus, Anna University, Chennai – 600044, India
Srimukhee.B	B.E. Student, Department of Electronics Engineering, MIT Campus, Anna University, Chennai – 600044, India

ABSTRACT

The paper presents the application of the Particle Swarm Optimization (PSO) technique based on a fitness function of the particles in the field of channel equalization. PSO is a derivative free algorithm that can be used for optimization problems. However, it has the problem of getting trapped in local minima. To overcome this, many modifications have been suggested based on inertia weight, swarm initialization and so on. Modifications based on inertia weight that have already been proposed are reviewed and then the technique based on the fitness of particles is discussed for the application of adaptive equalization. It is found that the technique suggested exhibits superior performance in both linear and nonlinear channels, compared to modifications and existing methods.

I. Introduction

From mobile phones to data transmission through the internet, every form of communication that we use today involves digital communication. A major limiting factor when it comes to high speed transmission through limited bandwidth available is the Inter Symbol Interference (ISI) which causes distortions in the transmitted symbols. In order to remove the effect of such ISI and recover the transmitted signal, we make use of an equalizer at the front end of the receiver. An equalizer is essentially a filter whose transfer function is the inverse of the channel's transfer function. Adaptive equalization (Qureshi, 1985) of the channel can be done using various gradient based algorithms (Haykin, 1996) like Least Mean Square (LMS), Recursive Least Square (RLS) etc. However, since they are gradient based algorithms, while training the equalizer, the weights may not be optimum because there is a possibility of the mean square error (MSE) getting trapped in their local minima. The performance of these algorithms may further degrade in a nonlinear channel. In order to overcome these problems, various evolutionary algorithms have been proposed which are basically derivative free algorithms. Here equalization is an iterative process of reducing mean square error. Particle Swarm Optimization is one such algorithm which can be used to adjust the weights of an equalizer to minimize the MSE.

A new application has been proposed in this paper by using the inertia weights based on particle fitness to channel equalization. PSO with such modified inertia weight efficiently removes the effects due to multipath propagation, co-channel interference and channel noise. Also, the time for error reduction is greatly reduced thereby making channel with dynamic characteristics suitable for reliable transmission

PSO is an evolutionary algorithm, developed by Eberhart and Kennedy (1995), inspired by a pool of fishes or a swarm of birds moving as a group, in a particular pattern, towards their destination. Here every bird will try to move towards the center of the swarm, maintaining a particular distance from its neighbors. This is called 'Swarm Intelligence'. Using this analogy, we use Particle Swarm Optimization to train the equalizer weights. We aim to use PSO to move the received signal towards a minimum value of MSE by adjusting the weights of the equalizer accordingly.

PSO provides a better performance compared to its counterparts because of many reasons. It consumes lesser CPU time since PSO uses the past history of the population and each other's experience to solve the problem. MSE doesn't get trapped in its local minima since PSO searches the solution space for a globally optimal solution. It provides an efficient performance even in the case of a nonlinear channel. It consumes lesser memory. Since it has faster convergence, it is efficient in tackling dynamic problems. Practically, the channels used in digital communication have characteristics that keep changing with time. Since PSO has faster convergence, it can keep track of changes in channel characteristics and adapt to those changes. This algorithm is simple and easy to implement which makes it an ideal choice for reducing ISI in digital communication.

II. Basic PSO

PSO (Kennedy and Eberhart, 1995 and Del Valle, Venayagamoorthy, Mohagheghi, Hernandez, Harley, 2008) starts with randomly chosen values from the problem space. They are called particles. In order to calculate MSE, a number of samples are considered for each particle. The whole set of particles is referred to as population. Each particle keeps track of its personal best fitness solution (*pbest*). A global version of PSO keeps track of the overall best value (*gbest*). Using pbest and gbest, the weights of the equalizer are changed according to the equations (1) and (2).

$$vel=iw\times vel+c1\times rand\times(pbest-w) + c2\times rand\times(gbest-w) \quad (1)$$

$$w = vel + w \quad (2)$$

where *iw* is inertia weight, c_1 and c_2 are acceleration constants towards pbest and gbest respectively, *rand* is a uniformly distributed random number in the range [0,1] and *w* refers to the weights of the equalizer. In order to prevent the velocity from going out of bounds, a fraction of velocity in a particular iteration is passed on to the next iteration. This is done by *iw* which has a value between [0,1]. The constant c_1 represents local search capabilities of the algorithm and pulls every particle towards its local best while c_2 represents social influence to converge to a globally optimal solution. The steps involved in the algorithm (Al-Awami, Zerguine, Cheded, Zidouli, Waleed-Saif (2010), Al-Awami, Saif, Zerguine, Zidouri, and Cheded (2007), Krusienski and Jenkins (2005) and Sandhya Yogi, Subhashini, Satapathy, Shiv Kumar (2010)) are given as flowchart in Figure.1.

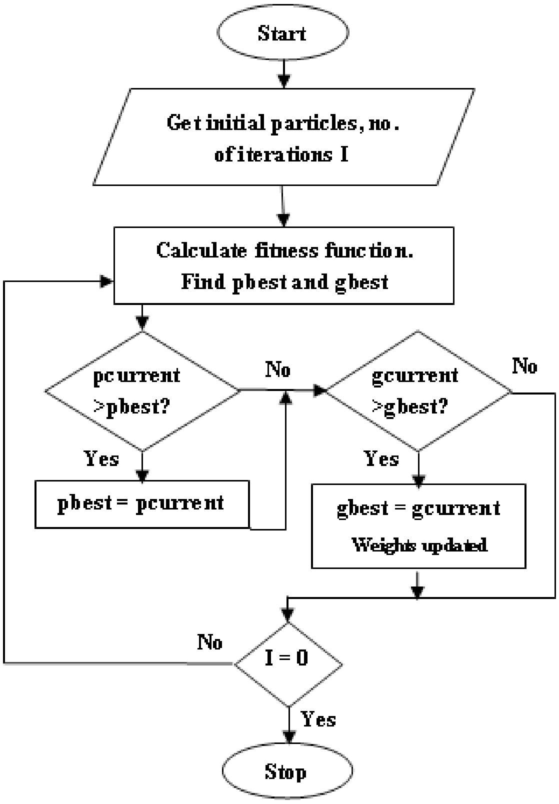

Figure. 1 Flow Chart for Basic PSO

III. Modifications to PSO

This section discusses about the various existing modifications suggested by several researchers to the basic PSO algorithm based on inertia weight.

A. Constant inertia weight

Here, a constant value between 0 and 1 (Shi and Eberhart, 1998) is multiplied with the velocity equation and the simulation results showed an increase in speed of convergence. Also, computational complexity and the time for execution reduced.

B. Inertia weights within a range

The values for inertia weight were chosen from a specific range (Shi and Eberhart, 1998) divided uniformly into '*I*' no of values, where *I* is the no of iterations. For every iteration, one value from this range is multiplied with velocity equation. The simulation results show a decrease in MSE value compared to the previous method. As the range increases the value of MSE with which the algorithm converges decreases even more.

C. Time decreasing inertia weight

This type of the inertia weight takes values adaptively based on the iteration count. Initially, large values are assigned to the inertia weight to enhance the algorithm's global search capability. As the iteration count increases, the algorithm would have arrived at an approximate globally fit value. So the value given to inertia weight is smaller at the end to enable an efficient local search around the global best(Kennedy and Eberhart, 1995).

$$iw = w_{\min} + (w_{\max} - w_{\min})\frac{(m-n)}{(m-1)} \qquad (3)$$

where *m* denotes the maximum number of iterations and *n* denotes current iteration. Here w_{min} and w_{max} can have values between 0 and 1.

D. Constriction factor

Constriction factor is a constant multiplied by inertia weight, but its value is fixed by the coefficients c_1 and c_2. It limits the dynamic value of *vel* according to the maximum value of weight w. It controls the path in which the particle travels towards the optimal solution by constricting the velocity values to a specific limit. The speed of convergence is the fastest in this case when compared to other inertia weight methods. For high speed applications this method will be fitting. Since it is a constant, time taken for computation is lesser. The expression for constriction factor (Shi and Eberhart, 1998) is given by,

$$K = \frac{k}{|2-\varphi-\sqrt{\varphi^2+4\varphi}|} \qquad (4)$$

where, *k*=2, *=c1+c2*

IV. Inertia weights based on fitness function

The technique suggested for channel equalization, which is a modification to basic PSO based on a fitness function of particles and success rate, has been explained in this section. These modifications have been analyzed for the application of adaptive channel equalization in this paper.

A. Inertia weight based on particle fitness

The values for inertia weight are updated for every iteration taking into consideration of the current global best value and average of local best values of all particles in that iteration (Arumugam and Rao, 2008). The parameters *gbest* and *pbest* refer to particle fitness function.

$$iw(it) = 1.1 - \frac{gbest_{it}}{pbest_{avg_{it}}} \qquad (5)$$

where $gbest_{it}$ is the *gbest* value of a particular iteration *it* and $pbest_{avgit}$ is the average of *pbest* values of all particles in a particular iteration.

B. Inertia weight based on success rate

The success rate is way of finding the present position of swarm in the problem space. Success in any iteration happens when the present fitness value of a particular particle is smaller than the previous. For particles that succeeded in minimizing its fitness value than the previous iteration, a value of 1 is assigned. For other particles, value of 0 is assigned. $P_s(t)$ represents the percentage of particles that have succeeded in moving towards an optimum solution. Inertia weight is assigned a value proportional to this percentage of success (Ahmad Nickabadi, Mohammad Mehdi Ebadzadeh, Reza Safabakhsh (2011)). Simulation results for this type showed an increase in error performance for both linear and nonlinear channels.

$$s(i,it) = \begin{Bmatrix} 1, \text{if pbest}_i^{it} > \text{pbest}_i^{it-1} \\ 0, \text{otherwise} \end{Bmatrix} \qquad (6)$$

$$P_s(it) = \frac{\sum_{i=1}^{n} s(i,it)}{n} \qquad (7)$$

$$iw(t) = (w_{\max} - w_{\min})P_s(t) + w_{\min} \qquad (8)$$

where w_{max} and w_{min} can have values in the range [0,1]. Simulation results show that the algorithm converges to a smaller value of MSE when the difference between w_{min} and w_{max} increases.

V. Simulation results

The various parameters involved in the proposed modifications are varied and simulation results are given in this section. The modification that has been proposed on the basis of success rate converges to a lower value of MSE compared to other modifications. The parameters are taken as population size *P*=40, window size *N*=200, number of iterations it_{max}=100, acceleration coefficients c_1=0. 55 c_2=0. 45, tap weights T=7, w_i =0.9 and w_f =0.3. The performance of PSO is analyzed for linear as well as non-linear channel. The linear channel consists of three paths and the impulse response of the channel is defined as follows:

(9) The factor *W* controls the amount of amplitude distortion and chosen as *W*=2.9. The channel is symmetric around n=2 and so the signal experiences a delay of 2 samples. The noise sequence $\{v_n\}$, has zero-mean and variance is 0.001. The input of the equalizer is given by the convolution sum of the random

input Bernoulli sequence with $\{a_n\} = \pm 1$ and the added random additive white Gaussian (AWGN) noise with SNR 30dB.

Another channel used in the analysis is a nonlinear channel used by Shi and Eberhart (1998) and Biglieri, Gersho, Gitlin, Lim(1984), where the output y(n) and input x(n) is related through the relation (10),

$$y(n) = a_1 x(n) + a_2 x^2(n) + a_3 x^3(n) \qquad (10)$$

The first, second and third-order coefficients, a_1, a_2, a_3 are chosen as 1, 0.1 and 0.05 respectively. The nonlinear channel, uses the linear channel h_n as the FIR filter.

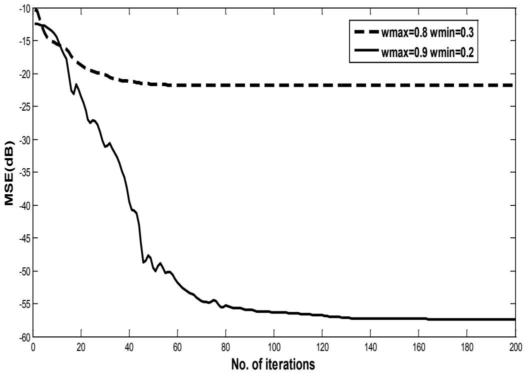

Figure.2 Inertia weight based onsuccess rate for different values of w_{max} and w_{min}

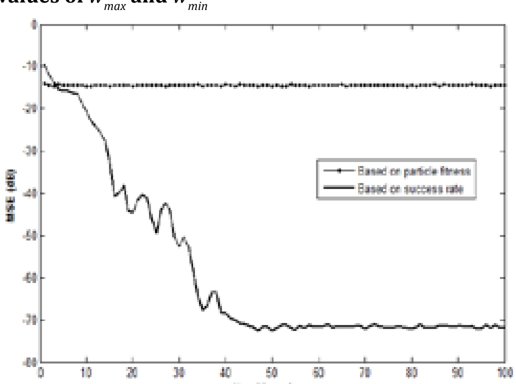

Figure.3 Performance of proposed modifications for a non-linear channel

From Figure.2 it is clear that the algorithm converges to smaller value of MSE when the difference between w_{max} and w_{min} increases. From Figure. 3 and Figure. 4, it is clear that PSO is an effective and efficient solution for both linear and nonlinear channels. In both the cases, inertia weight based on success rate converges to smaller value of MSE while inertia weight based on particle fitness takes a lesser number of iterations to converge.

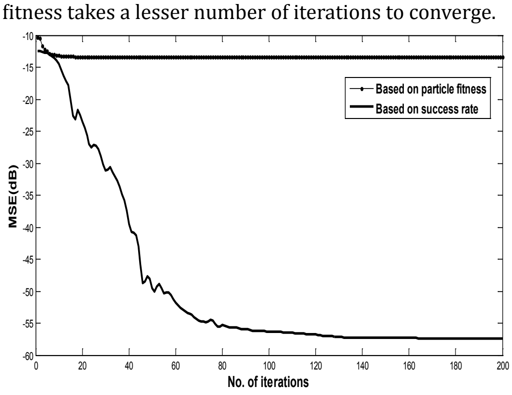

Figure.4 Performance of proposed modifications for a linear channel

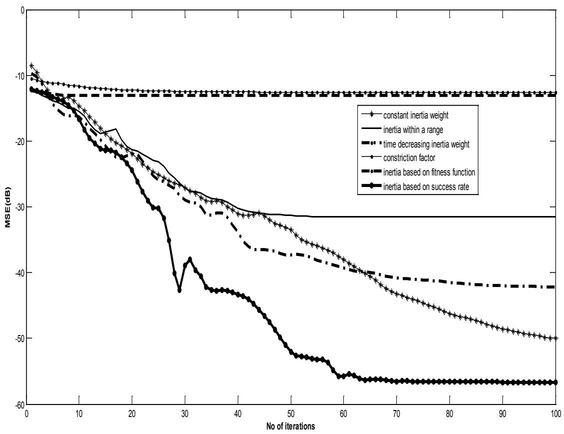

Figure.5 Performance of various types of PSO for a linear channel

Figure.5 and Figure.6 provide simulation results for the performance of various types of PSO for a linear and non-linear channel respectively. In Figure 5, for a linear channel, success rate provides better performance compared to other modifications in terms of MSE. In Figure.6, since distortions are higher in a non linear channel, the globally best value of MSE fluctuates initially and settles to a minimum value and converges.

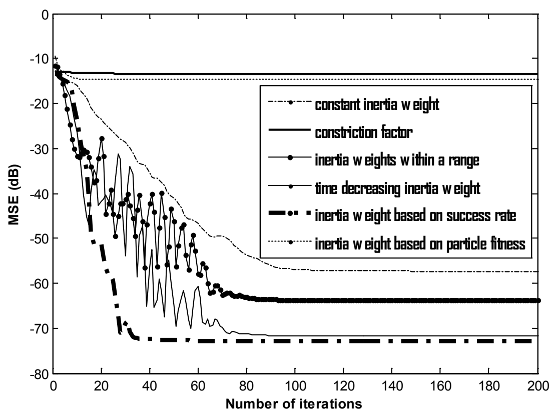

Figure. 6 Performance of various type of PSO for a non-linear channel

VI. Conclusions

This paper reviews the PSO algorithm and its modifications applied for channel equalization and then the adaptive inertia weight modification is applied in PSO for adaptive equalization. Extensive simulation results show that PSO is an ideal choice for channel equalization and provides an effective solution for time-varying channels by quickly adapting to the changes in the channel due to faster convergence towards a minimum value of MSE. It is clear that inertia weight based on success rate enables the algorithm to converge to a minimum value of MSE while the modification based on fitness of the particles enables the algorithm to converge faster with high MSE. Hence it is an ideal solution for time varying channels.

REFERENCE

[1] Ahmad Nickabadi, Mohammad Mehdi Ebadzadeh, Reza Safabakhsh. (2011). A novel particle swarm optimization algorithm with adaptive inertia weight, Applied Soft Computing 11, pp.3658–3670. | [2] Al-Awami A.T., Saif W., Zerguine A., Zidouri A., & Cheded L. (2007) "An Adaptive Equalizer Based On Particle Swarm Optimization Techniques," Proc. IEEE Signal Processing and Its Applications. | [3] Al-Awami A.T, Zerguine. A, Cheded. L., Zidouli A., WaleedSaif (2010). 'A new modified particle swarm optimization algorithm for adaptive equalization', Digital Signal Process, Elsevier Inc 2010. | [4] Arumugam M.S. & Rao M.V.C. (2008). On the improved performances of the particle swarm optimization algorithms with adaptive parameters, cross-over operators and root mean square (RMS) variants for computing optimal control of a class of hybrid systems, Applied Soft Computing, pp. 324–336. | [5] Biglieri E., Gersho A., Gitlin R.D., Lim T.L. (1984). "Adaptive cancellation of nonlinear intersymbol interference for voice-band data transmission", IEEE JSAC SAC- 2 (5), pp. 765–777. | [6] Del Valle. Y, Venayagamoorthy G.K. , Mohagheghi S., HernandezJ.-C., HarleyR.G.(April 2008), 'Particle swarm optimization: Basic concepts, variants and applications in power systems', IEEE Trans. Evol. Comput. 12 (2) pp.171–195. | [7] Haykin S. (1996). Adaptive Filter Theory, 3rd Edition, Prentice-Hall, Inc. | [8] Kennedy J. & Eberhart R. (1995) "Particle swarm optimization", Proc. IEEE Intl. Conf. Neural Networks, vol. 4 , pp. 1942-1948 | [9] Krusienski D.J. & Jenkins W.K. (2005). "Design and Performance of Adaptive Systems Based on Structured Stochastic Optimization Strategies", IEEE Circuits and system magazine, 2005. | [10] Qureshi.S (September 1985). "Adaptive Equalization," Proceedings of IEEE, vol. 73, pp.1349-1387. | [11] Shi Y. & Eberhart R. (May 1998) "A modified particle swarm optimizer", Proc. Intl. Conf. Evolutionary Computation, IEEE World Congress on Computational Intelligence, pp. 69–73. | [12] Shi Y.H. & Eberhart R.C. (1998). Parameter selection in particle swarm optimization, Annual Conference on Evolutionary Programming, San Diego. |