A Modified Particle Swarm Optimization Algorithm for Adaptive Filtering

D. J. Krusienski and W. K. Jenkins
Department of Electrical Engineering
The Pennsylvania State University
University Park, PA

Abstract—**Recently Particle Swarm Optimization (PSO) has been studied for use in adaptive filtering problems where the mean squared error (MSE) surface is ill-conditioned. Although the swarm generally converges to a limit point, when the population of the swarm is small the entire swarm often stagnates before reaching the global minimum on the MSE surface. This paper examines enhancements designed to improve the performance of the conventional PSO algorithm. It is shown that an enhanced PSO algorithm, called the Modified PSO (MPSO) algorithm, is quite effective in achieving global convergence for IIR and nonlinear adaptive filters.**

1. Introduction

The conventional PSO algorithm [1] begins by initializing a random swarm of M particles, each having R unknown parameters to be optimized. At each epoch, the fitness of each particle is evaluated according to the selected fitness function (MSE). The algorithm stores and progressively replaces the most fit parameters of each particle (*pbest*$_i$, i=1,2,...,M) as well as a single most fit particle (*gbest*) as better fit parameters are encountered. The parameters are updated at each epoch (n) according to:

$$\overline{vel_i}(n) = w * \overline{vel_i}(n-1) + acc_1 * diag[e_1, e_2, ..., e_R]_{i1} * (gbest - p_i(n-1))$$
$$+ acc_2 * diag[e_1, e_2, ..., e_R]_{i2} * (pbest_i - p_i(n-1)) \quad (1)$$
$$p_i(n) = p_i(n-1) + \overline{vel_i}(n) \quad (2)$$

where $\overline{vel_i}(n)$ is the velocity vector of the i^{th} particle, e_r is a vector of random values within in the interval (0,1), acc_1 and acc_2 are the acceleration coefficients toward *gbest* and $pbest_i$ respectively, and w is the inertia weight.

One disadvantage of the conventional PSO algorithm is that while the swarm generally converges to a limit point when the parameters are properly set, the entire swarm often stagnates at a limit point that does not reach the global minimum. Figure 1 shows the conventional PSO used to identify a second order IIR system with a matched order IIR adaptive filter driven with white noise and an output noise floor of –40 dB. Since this example satisfies Stearn's conjecture [2] the MSE surface is guaranteed to be unimodal. With a population of 25 the swarm stagnates at a –30 dB. solution. Increasing the population to 50 allows the solution to come much closer to the –40 dB. noise floor, although swarm stagnation is still visible [3]

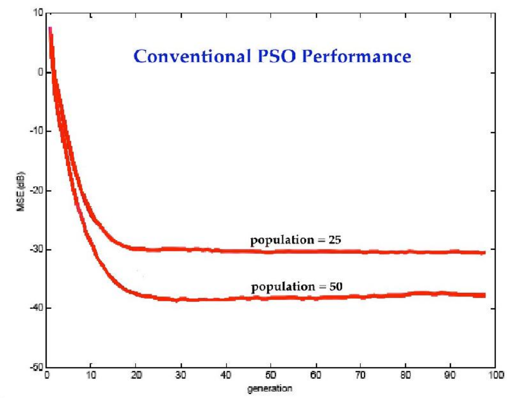

Figure 1. PSO stagnation on a unimodal surface.

2. PSO Enhancements and Variations

Various enhancements and variations discussed in this section are primarily a consequence of the observed limitations of conventional PSO. It is proposed that certain of these enhancements should be embedded in any practical implementation of the PSO algorithm, whereas the other suggested modifications can be applied on a problem dependent basis to increase the speed of convergence.

2.1 *Limitations of the Conventional PSO Algorithm*

In order to improve the efficiency and reliability of the conventional PSO, its weaknesses must be recognized. The following are a few of the concerns with conventional PSO and suggested improvements:

Concern 1: When a particle is found to be the new *gbest* of the swarm, all of the other particles begin to move toward it. If the new *gbest* particle is an outlying particle with respect to the swarm, the rest of the swarm tends to move toward the new *gbest* from the same general direction. This may potentially leave some critical region around the new minimum excluded from the search. This situation typically occurs when the global optimum occurs outside or at the perimeter of the convex region enclosing the current swarm.

Solutions: Increasing the acceleration coefficients to greater than unity can help to alleviate this problem to a limited extent, but may lead to other undesirable convergence issues. Alternately, when a new *gbest* is encountered, randomly selected particles can be rerandomized about the new *gbest*. This acts to ensure that the region around *gbest* is searched from all directions, while still keeping a portion of the swarm searching somewhat globally. This also allows the swarm to continue explore in directions outside the original search space if necessary. Randomly selected particles are chosen for the re-randomization rather than the least fit particles because re-randomizing the least fit particles can degrade the diversity of the search.

Concern 2: Particles closer to *gbest* tend to quickly converge on it and become stagnant (no longer update) while the other more distant particles continue to search. A stagnant particle is essentially useless because its fitness continues to be evaluated but it no longer contributes to the search.

Solutions: Stagnancy of particles can be eliminated by slightly varying the random parameters of each particle at every epoch, similar to mutation used in the genetic algorithm. This has little effect on particles distant from *gbest* because this random influence should be relatively small compared to the random update of equation. However, this eliminates any stagnant particles by forcing a finer search about *gbest*.

2.2 *Convergence Speed Enhancements*

When the problem in consideration is presumed to have a relatively docile error surface with few local minima, the convergence speed of PSO can be greatly enhanced with little concern for premature convergence to a local minimum. Other than optimizing the inertia weights and acceleration coefficients, increasing the acceleration toward *gbest*, or using a large population, the following are a few suggestions to enhance the convergence speed:

1) An enhancement that improves the speed and efficiency of the search is to independently adjust the inertia weight of each particle according to whether the new fitness is better than the previous fitness. The premise is that the inertia weight should be adaptable, either maintained or increased when a better fit position is encountered to keep the particle in a likely decent. For the opposite reason, if the particle does not attain a better fitness, its inertial influence should be less. This modification, however, does not prevent the hill-climbing capabilities of PSO, it merely increases the influence of potentially fruitful inertial directions, while decreasing the influence of potentially unfavorable inertial directions. A suggested relation of the adaptive inertia weight to the change in fitness is given by:

$$w_i(n) = \frac{1}{\left(1 + e^{\frac{-\Delta J_i(n)}{S}}\right)} \qquad (3)$$

where $w_i(n)$ is the inertia weight of the i^{th} particle, $\Delta J_i(n)$ is the change in particle fitness between the current and last generation, and S is a constant used to adjust the transition slope based on the expected fitness range. This relation limits the inertia on the interval (0,1), with the midpoint of 0.5 corresponding to zero change in fitness. Consequently, increases in fitness will lead to inertia weights larger than the recommended fixed experimental value of 0.5, and decreases in fitness will lead to inertia weights smaller than 0.5.

2) The convergence speed of the PSO algorithm can be enhanced by using a more peaky distribution, such as a Gaussian, to replace the uniform distribution for selecting the random direction vector components. This decreases the randomness of the search and enables the steps to be more directed toward *gbest*.

3) The search space can be constricted very rapidly, thus improving the convergence rate, by replacing some or all of the $pbest_i$'s with $pbest_i$'s that have superior fitness. This will act to quickly concentrate the particles in areas of interest within the search space, constricting the space and giving faster convergence.

2.3 *Search Capacity Enhancements*

When the problem in consideration is dynamic or has a greatly irregular error surface with many local minima, PSO can be enhanced to provide maximal search capabilities. The initial stages of the PSO algorithm tend to give a broad search where key points can be missed because the individual search regions typically decrease in size rather quickly as new bests are found. If *gbest* is at a local minimum and continues to be after sufficient epochs, which would likely be caused by an inadequate swarm size or large initial acceleration, the swarm can undesirably converge on the local minimum. The tradeoff that typically results from implementing these types of search capacity enhancements is that the convergence speed can be compromised.

Other than optimizing the inertia weights and acceleration coefficients or using a large population, the following are additional modifications that may enhance the search capacity:

1) The likelihood of premature convergence on a local minimum be decreased by re-randomizing a random portion of the particles over the entire parameter space and allowing them to converge. The re-randomization can be performed either continuously, or by monitoring the variance of the swarm. This will in effect continually generate unique search paths, which can increase the probability of finding the global optimum.

2) Another alternative to better facilitate the search includes swarming particles toward centers of mass defined by groups of particles or previous bests rather

than a single point and using multiple sub-swarms [1] that swarm toward separate centers. These two modifications can add more diversity and better distribute the swarm, decreasing the likelihood of converging on a local minimum.

3) To a lesser extent, the efficiency of the algorithm may be improved because particles closer to *gbest* may have already searched nearly the same region as more distant particles, which can be unnecessarily redundant. The re-randomization procedures described previously can be planned such that the space is searched more efficiently and redundancy is kept to a minimum. One technique for accomplishing this is to re-randomize according to an appropriate distribution. By the nature of the algorithm, the region close to the *gbest* is more efficiently searched because the density of the particles has been greatest in that region by the time of convergence. Re-randomizing with a distribution that is sparse about *gbest* would be more effective in terms of reducing unnecessary redundancy.

4) If the dimensionality of the problem becomes an issue, another variation is to separate the filter parameters into multiple independent swarms that will each search a lower dimensional space. This can potentially lead to better convergence properties if the parameters designated to each swarm can be de-coupled.

2.4 *Swarm Convergence and Parameter Coordination*

In order to ensure convergence of the swarm, the variance of the mutation and selected re-randomization distributions must decrease according to a prescribed schedule. One possible variance decay curve is illustrated in figure (2) and defined by equation (3):

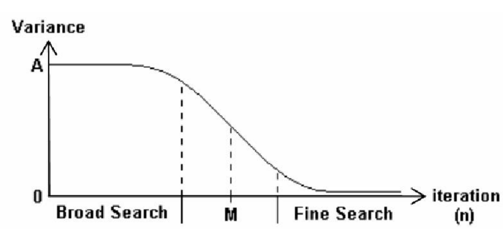

Figure 2. Transition schedule.

$$Variance(n) = \frac{-A}{\left(1 + e^{\frac{-n+M}{S}}\right)} + A \qquad (3)$$

M: transition midpoint
S: transition slope adjustment

This schedule specifies a wide search (high variance) initially, and then decays toward a finer search (reduced variance) at a suitable interval after which the space is presumed to be searched sufficiently. This schedule may also be applied to the acceleration coefficients to further tune the search. The re-randomization and acceleration schedules can be coordinated to optimize the convergence speed and search efficiency.

3. The Modified PSO Algorithm

Due to their observed effectiveness in overcoming the limitations of the conventional PSO algorithm, it is proposed that the enhancements of mutation, re-randomization about *gbest*, and adaptive inertia operations be routinely embedded in any variation of PSO, producing the Modified PSO (MPSO) algorithm. Because mutation is a search capacity operation, it tends to slow convergence speed. Therefore, the adaptive inertia operator can be implemented to offset this effect and improve the convergence speed [4].

This section provides an example to demonstrate how the enhancements provided by mutation, re-randomization about *gbest*, and adaptive inertia operations improve the convergence properties of the conventional PSO. For this example the standard PSO algorithm is used to adapt a simple multimodal IIR configuration, which has been used extensively to test global adaptive filtering algorithms [5]. This system consists of the following reduced order IIR adaptive filter:

$$H_{PLANT}(z^{-1}) = \frac{1}{(1-0.6z^{-1})^3} \qquad (4)$$

$$H_{AF}(z^{-1}) = \frac{p_i^1}{1 + p_i^2 z^{-1} + p_i^3 z^{-2}} \qquad (5)$$

$$H_{COLOR}(z^{-1}) = (1-0.6z^{-1})^2(1+0.6z^{-1})^2 \qquad (6)$$

The adaptive filters use a colored input generated by filtering unit variance white noise produced by the FIR filter of equation (6). This, in combination with the reduced order, creates a bimodal error surface as described in ref [6].

Performance improvements provided by each of these enhancements are illustrated in figure 3, for the test system described above. A flow chart for the Modified PSO (MPSO), is given in figure 3. The MPSO algorithm is capable of providing desirable performance and convergence properties in most any context. In addition to eliminating the concerns of conventional PSO, the algorithm is designed to balance the convergence speed and search quality tradeoffs of a stochastic search.

The capabilities of the MPSO are demonstrated in figures 4 and 5 and compared with the conventional SO and the Genetic Algorithm (GA) for the matched order IIR example (same system used in figure 1) and a reduced order LNL nonlinear cascade structure [6], respectively. Note that in the example of figure 4, using a modest population of 10, the conventional PSO stagnates at –20dB, whereas the MPSO is able to navigate all the way to the –80 dB. noise floor. The comparative performance between the conventional PSO and the MPSO for the LNL nonlinear cascade is similar, where once again it can be seen that the

MPSO enhancements prevent stagnation. In general it has been observed that for relatively high populations the MPSO and the GA exhibit similar convergence properties. However, for lower populations the MPSO often performs better than the GA for the types of IIR and nonlinear adaptive filters considered in this study.

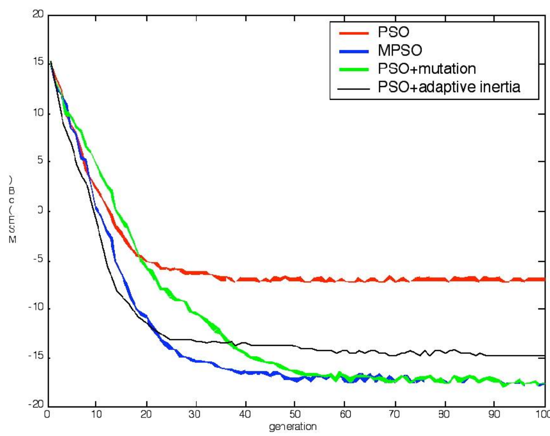

Figure 3. Effects of PSO enhancements (reduced order IIR)

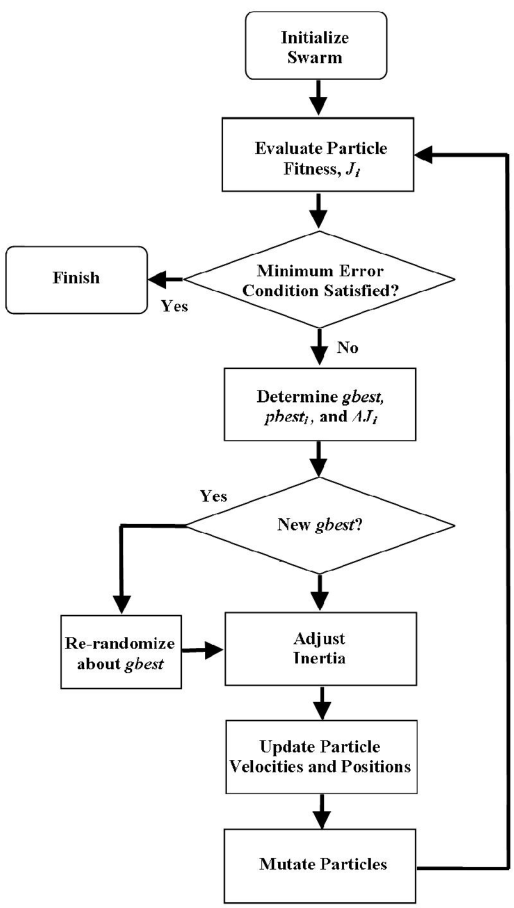

Figure 3. Flow chart for the MPSO algorithm

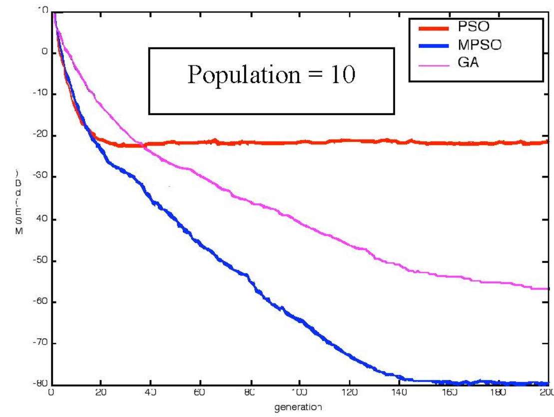

Figure 4. Performance on second order IIR surface.

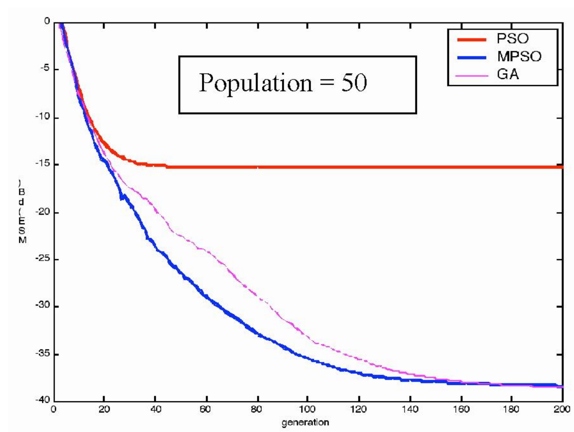

Figure 5. Performance on a nonlinear LNL example.

References

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

[2] S.D. Stearns, "Error Surfaces of Recursive Adaptive Filters," *IEEE Trans. on Circuits and Systems, vol. CAS-28, pp. 603-606, June 1981.*

[3] D. J. Krusienski and W. K. Jenkins, "The application of particle swarm optimization to adaptive IIR phase equalization," *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada, pp II-693 – II-696, May 17-21, 2004.

[4] D. J. Krusienski and W. K. Jenkins, "Design and Performance of Adaptive Systems Based on Structured Stochastic Optimization Strategies," *Circuits and Systems Magazine*, Vol. 5, No. 1, pp 8 –20, February 2005.

[5] W. K. Jenkins, S. Pal, and D. J. Krusienski, "Structured stochastic optimization strategies for problems with ill-conditioned error surfaces" *Proceedings of the International Symposium on Circuits and Systems,* Kobe, Japan, pp 2291 - 2294 – II-696, May 23 - 26, 2005.

[6] D. J. Krusienski, "Enhanced Structured Stochastic Global Optimization Algorithms for IIR and Nonlinear Adaptive Filtering", Ph.D. Thesis, The Pennsylvania State University, 2004.