# Adaptive IIR Phase Equalizers Based on Stochastic Search Algorithms

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

*Abstract* - **Two well known optimization algorithms, the Genetic Algorithm (GA) and the Simulated Annealing Algorithm (SAA), are investigated for IIR adaptive phase equalizers. For non-convex error surfaces, gradient-based algorithms often fail to find the global optimum. This work compares the ability of the GA and the SAA to achieve the global minimum solution for multi-order all-pass adaptive filters to be used for the phase equalization of minimum phase SAW filters.**

## 1. INTRODUCTION AND BACKGROUND

This work investigates the effectiveness of two well known stochastic optimization algorithms, the Genetic Algorithm (GA) and the Simulated Annealing Algorithm (SAA), for locating the global optimal solution for multi-order all-pass IIR adaptive filters. Both of these algorithms are known to be effective in multimodal optimization problems, which is the primary reason for the interest in these search mechanisms. The structure of the all-pass IIR adaptive phase equalizer proposed for this application is given by

$$H(z)_p = \prod_{n=1}^{N} \frac{a_n{}^* + z^{-1}}{1 - a_n z^{-1}}, \qquad (1)$$

where * denotes the complex conjugate. Note that an all-pass filter can be represented completely by its poles.

Wilborn [1] demonstrated that a one-pole, one-zero all-pass adaptive equalizer using the LMS algorithm is capable of equalizing the phase response of a minimum phase SAW filter for potential use in IS-95 CDMA wireless communication systems. Furthermore, it is expected that additional improvements can be made in the SNR by using higher order IIR phase equalizers.

The SAW filter is an important component in an IS-95 CDMA receiver. Although the ideal choice relative to performance criteria is a linear phase SAW filter, it has been shown previously that the use of a minimum phase SAW filter significantly reduces the cost of this critical component. A minimum phase SAW filter can achieve the same magnitude response as a linear phase SAW, but its nonlinear phase response must be compensated with a phase equalization stage. The purpose of this work is to determine whether higher order equalizers can be used to more effectively reduce the phase distortion of the minimum phase SAW filter. Both the GA and the SAA adaptive strategies were implemented to determine how well they perform in the application of phase equalization. The standard specified by the IS-95-A, for the minimum RMS phase error, was 3 .

To check for non-linearity, the total phase of the minimum phase SAW filter in series cascade with the equalizer was generated. The best-fit line for the phase of this combination was iteratively determined. The RMS phase error was calculated using the deviation between the phase plot of the minimum phase filter combined with the adaptive equalizer, and the best-fit linear phase characteristic [1. 4].

## 2. THE GENETIC ALGORITHM

Genetic Algorithms (GAs) are robust search and optimization techniques that are used widely in practical applications due to their capacity to locate the global optimum in a multimodal region [2]. Since the details of the GA for the IIR phase equalization problem were previously presented in [3], they will not be repeated here due to the lack of space. The following discussion focuses on demonstrating the behavior characteristics of the GA as a function of population size, mutation rate, and order of the equalizer (number of IIR sections).

### 2.1 Effect of Population Size

The genetic algorithm was investigated for various filter orders with different population sizes in order to analyze the effect of this parameter [4]. As a baseline, a typical learning curve for the GA is shown in Figure 1, where the complete search pattern is shown, along with the optimal selections. It was seen throughout the experiments that for lower order filters a larger population size led to convergence in a relatively large number of iterations, as compared to when a lower population size was used. In many of these situations a small population size provided enough search capacity to reach a neighborhood of the global optimum. However, a larger population size decreased the minimum mean square error further by approaching closer to the global optimum, but resulted in longer times to reach convergence.

Throughout the experiments, a higher population size gave better minimum errors at the cost of lower convergence rates. Thus the choice of population size represents a trade off among convergence rate, performance as measured by the minimum MSE, and computational complexity.

0-7803-8104-1/03/$17.00 ©2003 IEEE

Figure 2 and Table 1 show the effects of population size with a mutation rate of 30%. For a 2nd order filter a population size of 84 was too high and hence it took 280 iterations to converge giving an optimum error of –5.27 dB. This is lower than the error obtained with a population size of 60. But when the population size was reduced to 60, convergence was achieved in 136 iterations. Again, a population size of 30 appeared too small to permit the a 2nd order filter to achieve fast convergence.

### 2.2 Effect of Mutation Rate

Mutation rate is another important parameter in the genetic algorithm as it introduces diversity into the population. A low mutation rate may not introduce new individuals fast enough thus slowing convergence. This can be seen in Figure 3 with a 1.0 % mutation rate for a 2nd order filter. Alternatively, a high mutation rate may make the search too random and hence may lower the convergence rate. This is seen in Figure 3, where a mutation rate of 50% resulted in 407 iterations to reach convergence. It was observed that for higher order filters, lowering the mutation rate resulted in faster convergence, with a correspondingly higher error.

When the mutation rate is too large in a high order filter, the search becomes too random and may end prematurely. Figure 4 shows the effects of high and low mutation rates for a 7th order filter. For maintaining a search space that will facilitate convergence, the population size and mutation rate should be chosen in accordance with the order of the filter. In general a large population size combined with a high mutation rate would be too divergent, whereas lowering either one of the parameters and keeping the other high results in convergence with an optimum error and an acceptable convergence rate. Table 3 and Figure 5 show the learning for different order filters with a population size of 80 and a mutation rate of 30%. Table 4 shows the RMS phase error after equalization for various order equalizers

## 3. SIMULATED ANNEALING

Simulated Annealing (SA), as derived from statistical mechanics, exploits an analogy between the way in which a metal cools and freezes into a minimum energy crystalline structure (the annealing process) and the search for a minimum entropy condition in a more general optimization problem. The algorithm is based upon that of Metropolis et al. [5], which was originally proposed as a means of finding the equilibrium configuration of a collection of atoms at a given temperature.

### 3.1 The Physical Process

The physical process of annealing is one in which a solid in a heat bath is heated by increasing the temperature of the heat bath, until is reaches a value at which all particles of the solid randomly arrange themselves in the liquid phase. Then the temperature of the heat bath is slowly lowered, giving the particles a chance to arrange themselves in low-energy ground state. The minimum energy state will be reached provided the cooling is sufficiently slow. As the temperature is reduced, the atomic energies decrease. Starting off at a given value of the temperature, the temperature of the system is subsequently gradually reduced, after reaching thermal equilibrium at each temperature. In thermal equilibrium the probability of occurrence of a state with energy E is given by the Boltzmann distribution represented as

$$P(E) \quad e^{[-E/(kT)]} \qquad (2)$$

where E is the system energy, k is Boltzmann's constant, T is the temperature and P(E) is the probability that the system is in a state with energy E. If the temperature of the system is cooled non-gradually or in a fast manner, widespread irregularities and defects are seen in the crystal structure. This is known as rapid quenching. The system does not reach the minimum energy state and ends in a polycrystalline state, which has higher energy.

At high temperatures, P(E) converges to 1 for all energy states according to equation (2). It can also be seen that there exists a small probability that the system might have high energy even at low temperatures. Therefore, the statistical distribution of energies allows the system to escape from a local energy minimum.

### 3.2 The Simulated Annealing Algorithm

Simulated annealing employs a random search mechanism which not only accepts changes that decrease the objective function, but also accepts certain changes that increase it. In an analogy between an optimization problem and the annealing process, the states of the solid represent feasible solutions of the optimization problem. The energies of the states correspond to the values of the objective function at those solutions, and the minimum energy state corresponds to the optimal solution of the problem. Rapid quenching can be viewed as the process of terminating the search in a local minimum.

The algorithm consists of a sequence of iterations, with each iteration randomly changing the current solution to create a new one. Once a new solution is created the corresponding change in the cost function is computed to decide if the newly created solution is to be accepted. If accepted the update is made in the current solution. If the change in the cost function is negative, the newly produced solution is directly taken as the current solution. Otherwise it the decision on whether to accept the new solution is determined by the Metropolis Criterion.

### 3.3 The Metropolis Criterion

Metropolis' criterion is based on Boltzmann's probability. If the difference between the cost function values of the current and the newly produced solutions is equal to or larger than zero, a random number in [0,1] is generated from a uniform distribution and if

$$e^{(- \ E/T)} \qquad (3)$$

is satisfied, then the newly produced solution is accepted. If not, the solution is unchanged. In equation (3), E is the difference between the cost function values of the two solutions, and T is the temperature. Metropolis's criterion allows uphill moves, allowing the search to climb out of local minima.

### 3.4 The SA Algorithm with a Directed Approach

In this work the SAA was implemented using a directed approach, characterized as follows:

1. Starting with an initial temperature of 1, a random initial coefficient is chosen.
2. Vectors in predetermined directions were generated around this initial solution.
3. A new parameter value was determined by taking a step in the direction of the vectors generated. The magnitude of the vector was randomly determined by:

$$x_n = x_c + rv \ , \qquad (4)$$

   where $x_n$ is the new location, $x_c$ is the current location, r is a random number generated, which is between (-1, 1), and v is the vector generated in a specified direction. This new location was checked to see whether its energy was lower than the current solution. If it was lower, the solution was accepted; if higher, the Metropolis criterion was checked to decide whether or not to accept the new solution.
4. Step 3 was carried out until a predetermined N was reached.
5. Temperature was reduced using the cooling schedule.
6. If temperature limit was not reached the process returned to step 3, else exited.

### 3.5 Experimental Results

Throughout the experiments it was seen that as the number of iterations per each temperature were increased, the error was reduced and convergence took longer. This occurs as the search is prolonged, and hence, there is a better chance of finding the optimum. Figure 6 shows the working of the algorithm of the SAA; i.e. the actual search mechanism vs. only the optimal points [4].

It was seen throughout the experiments that lower order filters ($2^{nd}$ or $3^{rd}$ order) converged in a smaller number of iterations and achieved in good minimum error values. If the number of iterations was increased per temperature the error kept decreasing, but the convergence rate became relatively high. This is effect shown in Table 5 and Figure 7.

For higher order filters (like the $5^{th}$ order filter and higher) a low number of iterations per temperature did not give the process enough time to find the global minimum and the search ended prematurely. As the number of iterations increased the error decreased steadily until convergence was achieved. This is seen in Table 6 and Figure 8 for a $5^{th}$ order filter. At approximately 350 iterations per temperature (N) the error was as low as -5.66 dB, taking 4761 iterations to reach that level. Increasing N to 850 gave a 0.3 dB improvement but required 13398 total iterations.

Typically a system should be cooled gradually in order to assure convergence. If it cools too fast, it emulates rapid quenching and falls into local minima. To observe the effect of rapid quenching, the reduction factor was changed from 0.9 to 0.7. Then to observe the effect of gradual cooling, the factor was changed again to 0.99. This was done for a $2^{nd}$ order filter, with 60 iterations per temperature. With c = 0.7, the search terminates in 47 iterations, illustrating the mechanism of rapid quenching. For c = 0.7, the cooling was too fast, in comparison to the case of c = 0.9, as seen in Table 7 and Figure 9. With c = 0.9, the process has sufficient time to reach a low enough error in a moderate number of iterations. It reached an optimum error in 219 iterations, which is much faster than the case of c = 0.99. The factor of 0.9 was chosen since it gave results that were optimal for this application, especially for the higher order filters.

When the temperature was reduced using a factor of c = 0.99, the system cooled gradually, thus allowing sufficient time to find the optimum after 5566 iterations. The performance of three different order filters while being cooled by c = 0.99 are shown in Table 8 and Figure 10.

The behavior of different order filters for a start temperature of 1 and c = 0.9 was investigated. The experimental results are shown in Table 8. It is seen from these results that although the $3^{rd}$ order filter gives the best error of -5.3 dB, it requires many more iterations to converge than the $2^{nd}$ order filter.

After convergence the equalization ability was checked for different order filters. Table 9 gives the RMS phase error for different filter orders, along with the number of iterations required for convergence.

## 4. REFERENCES


[1] T. Wilborn, "Adaptive All-pass Phase Equalizer for Digital Receivers: A Case Study," MS Thesis, Department of Electrical and Computer Engineering, University of Illinois, Urbana Champaign, 1999.

[2] S. C Ng, S. H. Leung, C. Y. Chung, A. Luk, W.H. Lau, "The Genetic Search Approach: A New Learning Algorithm for Adaptive IIR Filtering"," *IEEE Sig. Proc. Mag.*, Volume 13, Issue 6, November 1996 Page(s): 38 – 46.

[3] V. Hegde, S. Pai, T.B. Wilborn, and W. K. Jenkins, "Genetic algorithms for adaptive phase equalization of minimum phase SAW filters, *Proceedings of the 2000 Annual Asilomar Conference on Signals, Systems, and Computers,* Pacific Grove, CA November 2000.

[4] S. Pai, "Global optimization Techniques for IIR Phase Equalizers," MS Thesis, Department of Electrical Engineering, Penn State University, University Park, PA, August 2001.

[5] P. J. M. van Laarhoven, *Theoretical and Computational Aspects of Simulated Annealing*. Netherlands, 1988.

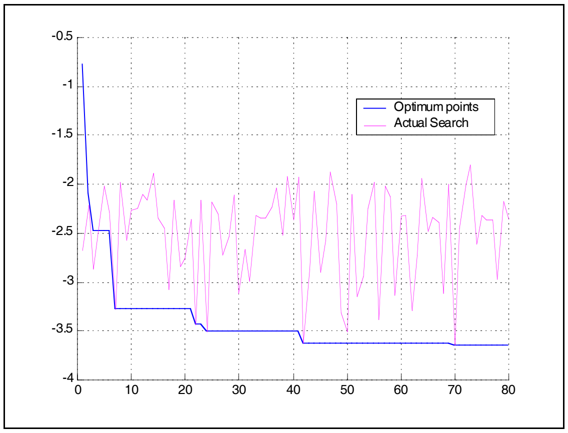


Figure 1. Actual search vs. optimal points for the GA.

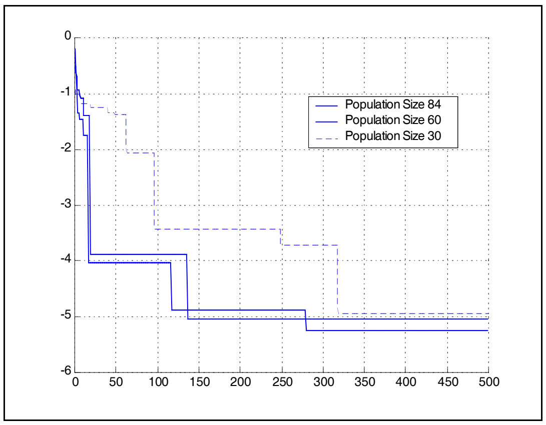


Figure 2. Population size vs. convergence rate.

Table 1. Effect of population size.

| ORDER | POP. | MUT. (%) | ERROR (dB) | ITER. |
|---|---|---|---|---|
| 2 | 84 | 30 | - 5.27 | 407 |
| 2 | 60 | 30 | - 5.10 | 136 |
| 2 | 30 | 30 | -4.90 | 318 |

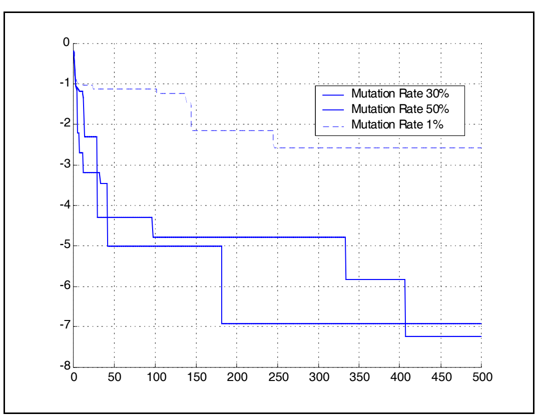


Figure 3. Mutation rate vs. convergence rate.

Table 2. Effect of mutation rate on learning characteristics.

| ORDER | POP. | MUT. (%) | ERROR (dB) | ITER. |
|---|---|---|---|---|
| 2 | 80 | 50 | - 7.24 | 407 |
| 2 | 80 | 30 | - 6.94 | 182 |
| 2 | 80 | 1 | -2.60 | 245 |

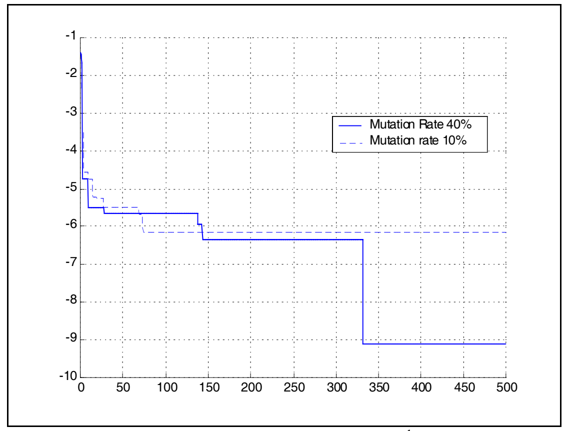


Figure 4. Effect of mutation rate for 7th order filter.

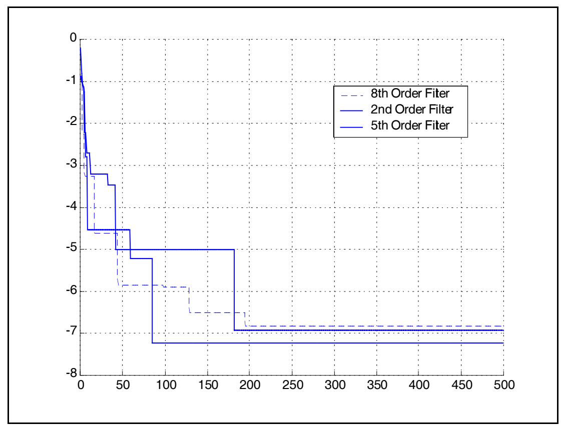


Figure 5. Performance of different order filters.

Table 3. Convergence of different order filters.

| ORDER | POP. | MUT. (%) | ERROR (dB) | ITER. |
|---|---|---|---|---|
| 2 | 80 | 30 | - 6.94 | 182 |
| 5 | 80 | 30 | - 7.25 | 85 |
| 8 | 80 | 30 | -6.84 | 194 |

Table 4. Rms phase error for different filter orders.

| FILTER ORDER | RMS PHASE ERROR (degrees) |
|---|---|
| 2 | 1.1084 |
| 4 | 0..9698 |
| 6 | 1.1909 |

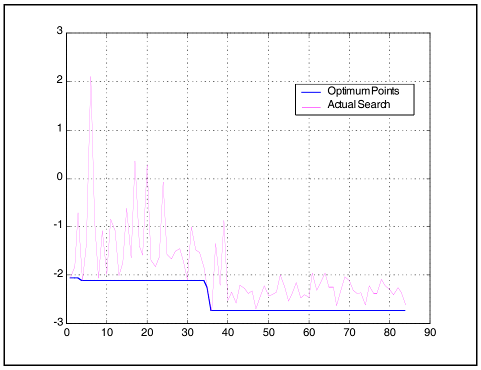


Figure 6. Actual search vs. optimal points for the SAA.

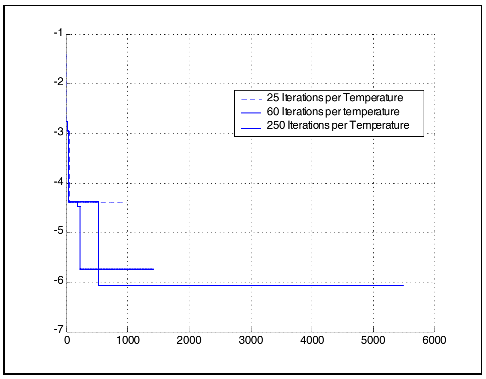


Figure 7. Convergence rate vs. iterations per temperature.

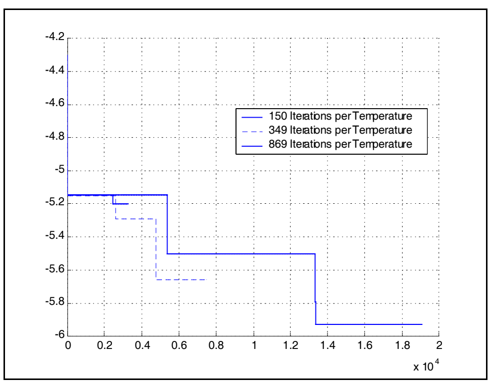


Figure 8. Convergence rate vs. iterations per temperature.

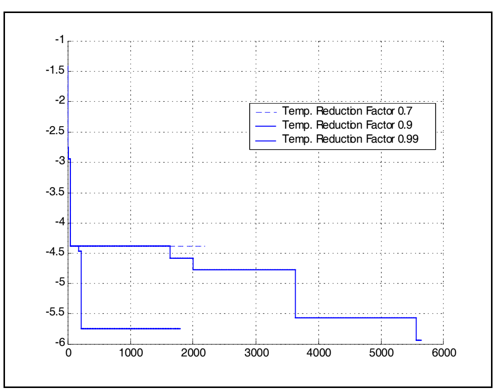


Figure 9. Effect of temperature reduction factors.

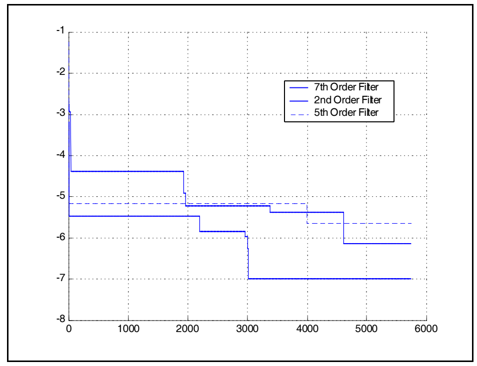


Figure 10. Performance of different filter orders with a different temperature reduction factor.

Table 5. Convergence for 2nd order filter.

| FILTER ORDER | REDUCTION FACTOR | OPTIMUM ERROR(dB) | CONVERGENCE RATE |
|---|---|---|---|
| 2 | 0.9 | - 4.4 | 47 |
| 2 | 0.9 | -5.75 | 219 |
| 2 | 0.9 | -6.1 | 522 |

Table 6. Convergence for 5th order filter.

| FILTER ORDER | REDUCTION FACTOR | OPTIMUM ERROR (dB) | CONVERGENCE RATE |
|---|---|---|---|
| 5 | 0.9 | -5.2 | 2473 |
| 5 | 0.9 | -5.66 | 4761 |
| 5 | 0.9 | -5.94 | 13398 |

Table 7. Different temperature reduction factors.

| FILTER ORDER | REDUCTION FACTOR | OPTIMUM ERROR (dB) | CONVERGENCE RATE |
|---|---|---|---|
| 2 | 0.7 | - 4.4 | 47 |
| 2 | 0.9 | -5.75 | 219 |
| 2 | 0.99 | -5.94 | 5566 |

Table 8. Convergence of different order filters.

| FILTER ORDER | REDUCTION FACTOR | OPTIMUM ERROR (dB) | CONVERGENCE RATE |
|---|---|---|---|
| 2 | 0.9 | -4.4 | 47 |
| 3 | 0.9 | -5.3 | 456 |
| 4 | 0.9 | -4.9 | 531 |

Table 9. RMS phase error for different order filters.

| FILTER ORDER | RMS PHASE ERROR (deg.) | NUMBER OF ITERATIONS |
|---|---|---|
| 2 | 1.2313 | 23 |
| 4 | 1.1074 | 905 |
| 6 | 2.8298 | 13613 |