JOURNAL OF KOREA MULTIMEDIA SOCIETY, VOL. 9, NO. 12, DECEMBER 2006(pp. 1689-1699)

Nonlinear Blind Equalizer Using Hybrid Genetic Algorithm and RBF Networks

Han, Soowhan†, Han, Changwook††

ABSTRACT

A nonlinear channel blind equalizer by using a hybrid genetic algorithm, which merges a genetic algorithm with simulated annealing, and a RBF network is presented. In this study, a hybrid genetic algorithm is used to estimate the output states of a nonlinear channel, based on the Bayesian likelihood fitness function, instead of the channel parameters. From these estimated output states, the desired channel states of the nonlinear channel are derived and placed at the center of a RBF equalizer to reconstruct transmitted symbols. In the simulations, binary signals are generated at random with Gaussian noise. The performance of the proposed method is compared with those of a conventional genetic algorithm (GA) and a simplex GA, and the relatively high accuracy and fast convergence of the method are achieved.

Keywords: Nonlinear blind equalizer, Hybrid genetic algorithm, RBF networks

1. INTRODUCTION

In digital communication systems, data symbols are transmitted at regular intervals. Time dispersion caused by non-ideal channel frequency response characteristics, or by multipath transmission, may create inter-symbol interference (ISI), and it has become a limiting factor in many communication environments. Furthermore, the nonlinear ISI that often arises in high speed communication channels degrades the performance of the overall communication system[1]. To overcome the effects of nonlinear ISI and to achieve high-speed reliable communication, nonlinear channel equalization is necessary.

※ Corresponding Author: Han, Soowhan, Address: (614-714) San 24, Gaya-dong, Jin-gu, Busan Korea, TEL: +82-51-890-2700, FAX: +82-51-890-2706, E-mail: swhan@deu.ac.kr
Receipt date : June 1, 2006, Approval date : Oct. 23, 2006
† Department of Multimedia Eng., Dongeui University
†† School of Electrical Eng. & Computer Science, Yeungnam University
(E-mail: cwhan@yumail.ac.kr)
※ This study was supported by Dongeui University research fund 2005AA183

The conventional approach to linear or nonlinear channel equalization requires an initial training period, with a known data sequence, to learn the channel characteristics. In contrast to standard equalization methods, the so-called blind (or self-recovering) equalization methods operate without a training sequence[2]. Because of its superiority, the blind equalization method has gained practical interest during the last few years. Most of the studies carried out so far are focused on linear channel equalization and this is required by the simplicity of the channel[3-5].

Only a few papers have dealt with nonlinear channel models. The blind estimation of Volterra kernels, which characterize nonlinear channels, was derived in[6], and a maximum likelihood (ML) method implemented via expectation-maximization (EM) was introduced in[7]. The Volterra approach suffers from its enormous complexity. Furthermore the ML approach requires some prior knowledge of the nonlinear channel structure to estimate the channel parameters. Recently, nonlinear structures such as multilayer perceptrons[8] and piecewise linear networks[9], trained to minimize some cost function, have been investigated for the blind nonlinear equalization. However, in those approaches, the structure and complexity of the nonlinear equalizer must be specified in advance. And the support vector (SV) equalizer proposed by Satamaria et al[10] can be a possible solution for both of linear and nonlinear blind channel equalization at the same time, but it is still suffering from the high computational cost of its iterative reweighted quadratic programming procedure.

A unique approach in nonlinear channel blind equalization was presented by Lin et al.[11], in which they used the simplex GA method to estimate the optimal channel output states instead of estimating the channel parameters directly. With this method, the complex modeling of the nonlinear channel can be avoided, and it has turned out that the nonlinear channel blind equalization problem can be transformed to the problem of determining the optimal channel output states. However, in this method, the performance of the equalizer is highly dependent on the searching algorithm for the optimal channel output states.

Therefore, for the better performance results, a hybrid genetic algorithm (GA merged with simulated annealing (SA): GASA) is investigated to search optimal output states of a nonlinear channel in this paper. GA[12] and SA[13], each of which represents a powerful optimization method, have complementary strengths and weaknesses. To get the synergy effect between GA and SA, many researchers have considered the combination of these two[14,15], and those algorithms have been successively used for the optimization problems [16,17]. Thus the GASA can be a better solution to find the optimal channel output states for nonlinear channel blind equalization. For our particular application, the proposed GASA has the Bayesian fitness function in the searching procedure. And by using random generated initial temperature in its selection procedure, it can reach the optimal global solution with a relatively high speed even when it is trapped in a local solution. Its performance is compared with those of a conventional GA and a simplex GA. In the experiments, the optimal output states of a nonlinear channel are estimated by using each of three different styles of GA algorithm. From these output states, the desired channel states of the nonlinear channel are derived and placed at the center of a RBF equalizer to reconstruct transmitted symbols. The RBF equalizer is an identical structure with the optimal Bayesian equalizer, and its important role is to place the optimal centers at the desired channel states[18].

The organization of this paper is as follows: Section 2 includes a brief introduction to the equalization of nonlinear channels using RBF networks; section 3 shows the relation between the desired channel states and the channel output states. In section 4, GASA with a Bayesian fitness function is introduced. The simulation results, including comparisons with the two other algorithms and the conclusions, are provided in sections 5 and 6, respectively.

2. NONLINEAR CHANNEL EQUALIZATION USING RBF NETWORKS

A nonlinear channel equalization system is shown in Fig. 1. A digital sequence $s(k)$ is transmitted through the nonlinear channel, which is composed of a linear portion $H(z)$ and a nonlinear portion $N(z)$, governed by the following expressions,

$$\bar{y}(k)=\sum_{i=0}^{p} h(i)s(k-i) \tag{1}$$

$$\hat{y}(k)=D_1\bar{y}(k)+D_2\bar{y}(k)^2+D_3\bar{y}(k)^3+D_4\bar{y}(k)^4 \tag{2}$$

where p is the channel order and D_i is the coefficient of the i^{th} nonlinear term. The transmitted symbol sequence $s(k)$ is assumed to be an equiprobable and independent binary sequence taking values from $\{\pm 1\}$, and the channel output is corrupted by an additive white Gaussian noise $e(k)$.

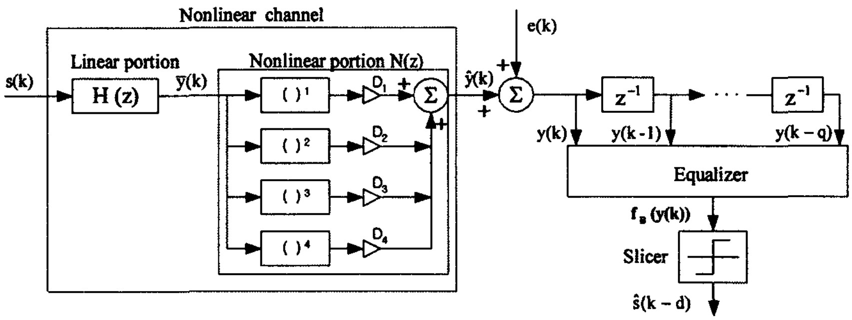


Fig. 1. The structure of a nonlinear channel equalization system.

Thus the channel observation $y(k)$ can be written as

$$y(k) = \hat{y}(k) + e(k) \tag{3}$$

If q denotes the equalizer order (number of tap delay elements in the equalizer), then there exist $M = 2^{p+q+1}$ different input sequences

$$\mathbf{s}(\boldsymbol{k}) = [s(k), s(k-1), \cdots, s(k-p-q)] \tag{4}$$

that may be received (where each component is either 1 or -1). For a specific channel order and equalizer order, the number of input patterns that influence the equalizer is M, and the input vector of equalizer without noise is

$$\hat{\mathbf{y}}(\boldsymbol{k}) = [\hat{y}(k), \hat{y}(k-1), \cdots, \hat{y}(k-q)] \tag{5}$$

The noise-free observation vector $\hat{\mathbf{y}}(\boldsymbol{k})$ is referred to as the desired channel states, and can be partitioned into two sets, $Y_{q,d}^{+1}$ and $Y_{q,d}^{-1}$, as shown in equations (6) and (7), depending on the value of $s(k-d)$, where d is the desired time delay.

$$Y_{q,d}^{+1} = \{ \hat{\mathbf{y}}(\boldsymbol{k}) \mid s(k-d) = +1 \} \tag{6}$$

$$Y_{q,d}^{-1} = \{ \hat{\mathbf{y}}(\boldsymbol{k}) \mid s(k-d) = -1 \} \tag{7}$$

The task of the equalizer is to recover the transmitted symbols $s(k-d)$ based on the observation vector $\boldsymbol{y(k)}$. Because of the additive white Gaussian noise, the observation vector $\boldsymbol{y(k)}$ is a random process having conditional Gaussian density functions centered at each of the desired channel states, and determining the value of $s(k-d)$ becomes a decision problem. Therefore, Bayes decision theory[19] can be applied to derive the optimal solution for the equalizer, and this optimal Bayesian equalizer solution is given by equations (8) and (9).

$$f_B(\mathbf{y}(\boldsymbol{k})) = \sum_{i=1}^{n_s^{+1}} \exp\left(-\left\|\mathbf{y}(\boldsymbol{k}) - \mathbf{y}_i^{+1}\right\|^2 / 2\sigma_e^2\right) - \sum_{i=1}^{n_s^{-1}} \exp\left(-\left\|\mathbf{y}(\boldsymbol{k}) - \mathbf{y}_i^{-1}\right\|^2 / 2\sigma_e^2\right) \tag{8}$$

$$\hat{s}(k-d) = \mathrm{sgn}(f_B(\mathbf{y}(\boldsymbol{k}))) = \begin{cases} +1, & f_B(\mathbf{y}(\boldsymbol{k})) \geq 0 \\ -1, & f_B(\mathbf{y}(\boldsymbol{k})) < 0 \end{cases} \tag{9}$$

where $\mathbf{y}_i^{+1}$ and $\mathbf{y}_i^{-1}$ are the desired channel states belonging to $Y_{q,d}^{+1}$ and $Y_{q,d}^{-1}$, respectively, and their numbers are denoted as n_s^{+1} and n_s^{-1}, and σ_e^2 is the noise variance. The desired channel states, $\mathbf{y}_i^{+1}$ and $\mathbf{y}_i^{-1}$, are derived by using their relationship with the channel output states, which will be explained in the next section. In this study, the optimal Bayesian decision probability shown in equation (8) is implemented by using a RBF network. The structure of a RBF network is shown in Fig. 2, and its output is given by equation (10)[11,20].

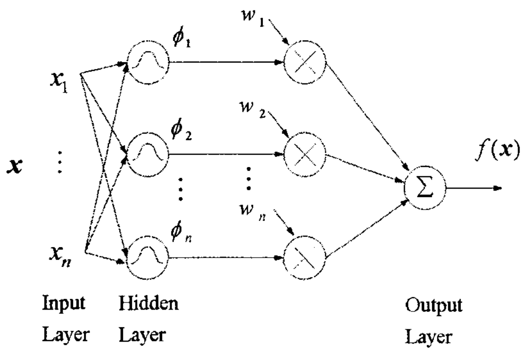


Fig. 2. The structure of a RBF network.

$$f(x)=\sum_{i=1}^{n}\omega_i\phi(\frac{\|x-c_i\|^2}{\rho_i}) \tag{10}$$

where n is the number of hidden units, c_i are the RBF centers, ρ_i is the width of the i^{th} unit and ω_i is its weight. The RBF network is an ideal processing means to implement the optimal Bayesian equalizer when the nonlinear function ϕ is chosen as the exponential function $\phi(x)=e^{-x}$ and all of the widths have the same value ρ, which is twice as large as the noise variance σ_e^2. For the case of equiprobable symbols, the RBF network can be simplified by setting half of the weights to 1 and the other half to -1. Thus the output of this RBF equalizer is same as the optimal Bayesian decision probability in equation (8), and its performance highly depends on whether the desired channel states, y_i^{+1} and y_i^{-1}, can be correctly positioned at its centers [11,18].

3. DESIRED CHANNEL STATES AND CHANNEL OUTPUT STATES

The desired channel states, y_i^{+1} and y_i^{-1}, are used as the centers of the hidden units in the RBF equalizer to reconstruct the transmitted symbols. If the channel order p=1 with $H(z)=0.5+1.0z^{-1}$, the equalizer order q=1, the time delay d=1, and the nonlinear portion $D_1=1, D_2=0.1, D_3=0.05, D_4=0.0$ in Fig. 1, then the eight different channel states ($2^{p+q+1}=8$) may be observed at the receiver in the noise- free case, and the output of the equalizer should be $\hat{s}(k-1)$, as shown in Table 1. From Table 1, it can be seen that the desired channel states $[\hat{y}(k),\hat{y}(k-1)]$ can be constructed from the elements of the dataset, called "channel output states", $\{a_1,a_2,a_3,a_4\}$, where $a_1=1.89375$, $a_2=-0.48125$, $a_3=0.53125$, and $a_4=-1.44375$ for this channel. The length of data-set, $\tilde{n}$, is determined by the channel order, p, such as $2^{p+1}=4$. In general, if q=1 and d=1, the desired

Table 1. The relation between desired channel states and channel output states

Nonlinear channel with $H(z)=0.5+1.0z^{-1}, D_1=1, D_2=0.1, D_3=0.05, D_4=0.0$, and d=1				
Transmitted symbols	Desired channel states			Output of equalizer
$s(k)\ s(k-1)\ s(k-2)$	$\hat{y}(k)$	$\hat{y}(k-1)$	By channel output states, $\{a_1,a_2,a_3,a_4\}$	$\hat{s}(k-1)$
1 1 1	1.89375	1.89375	(a_1,a_1)	1
1 1 -1	1.89375	-0.48125	(a_1,a_2)	1
-1 1 1	0.53125	1.89375	(a_3,a_1)	1
-1 1 -1	0.53125	-0.48125	(a_3,a_2)	1
1 -1 1	-0.48125	0.53125	(a_2,a_3)	-1
1 -1 -1	-0.48125	-1.44375	(a_2,a_4)	-1
-1 -1 1	-1.44375	0.53125	(a_4,a_3)	-1
-1 -1 -1	-1.44375	-1.44375	(a_4,a_4)	-1

channel states for $Y_{1,1}^{+1}$ and $Y_{1,1}^{-1}$ are (a_1,a_1), (a_1,a_2), (a_3,a_1), (a_3,a_2), and (a_2,a_3), (a_2,a_4), (a_4,a_3), (a_4,a_4), respectively. In the case of d=0, the channel states, (a_1,a_1), (a_1,a_2), (a_2,a_3), (a_2,a_4), belong to $Y_{1,1}^{+1}$, and (a_3,a_1), (a_3,a_2), (a_4,a_3), (a_4,a_4) belong to $Y_{1,1}^{-1}$. This relation is valid for the channel that has a one-to-one mapping between the channel inputs and outputs[11]. Thus the desired channel states can be derived from the channel output states if we assume p is known, and the main problem of blind equalization can be changed to focus on finding the optimal channel output states from the received patterns.

4. GASA WITH BAYESIAN FITNESS FUNCTION

It is known that the Bayesian likelihood (BL), defined in equation (11), is maximized with the channel states derived from the optimal channel output states[18,21].

$$BL = \prod_{k=0}^{L-1} \max(f_B^{+1}(k), f_B^{-1}(k)) \tag{11}$$

where $f_B^{+1}(k) = \sum_{i=1}^{n_s^{+1}} \exp\left(-\left\|y(k) - y_i^{+1}\right\|^2 / 2\sigma_e^2\right)$, $f_B^{-1}(k) = \sum_{i=1}^{n_s^{-1}} \exp\left(-\left\|y(k) - y_i^{-1}\right\|^2 / 2\sigma_e^2\right)$ and L is the length of received sequences. Therefore, the BL is utilized as the fitness function (FF) of the proposed algorithm to find the optimal channel output states after taking the logarithm, which is shown in equation (12).

$$FF = \sum_{k=0}^{L-1} \log(\max(f_B^{+1}(k), f_B^{-1}(k))) \tag{12}$$

The optimal channel output states, which maximize the fitness function FF, cannot be obtained with the conventional gradient methods, because the mathematical formulation between the channel output states and FF cannot be accomplished without knowing the channel structure[11]. Thus, genetic algorithm (GA) and simulated annealing (SA), each of which has shown successful performance in complex high dimensional optimal problems, are considered in order to find the optimal solution of equation (12).

A powerful optimization algorithm, GA, is a search algorithm based on an analogy with the process of natural selection and evolutionary genetics. It combines the survival of the fittest among string structures with a structured, yet randomized, information exchange to form a search algorithm with some of the innovative flair of a human search. It is guided largely by the machinations of three operators: selection, crossover, and mutation. In every generation, a new set of artificial creatures is created using bits and pieces of the old; an occasional new part is tried for good measure. More details of the conventional GA algorithm can be found in [12].

Another powerful optimization algorithm is SA and its basic idea comes from the physical annealing process done on metals and other substances. In metallurgical annealing, a metal body is heated to near its melting point and then slowly cooled back down to room temperature. This process will cause the global energy function of the metal to eventually reach an absolute minimum value. Thus SA allows a system to change its state to a higher energy state occasionally so that it has a chance to jump out of local minima and seek the global minimum. Its mathematical representation and detail optimization mechanism are given in [13].

Both of GA and SA have complementary strengths and weaknesses. While GA explores the search space by means of the population of search points, it suffers from poor convergence properties. SA, by contrast, has good convergence properties, but it cannot explore the search space by means of population. Many researchers have considered the combination of these two, and those algorithms have been successively used for the optimization problems[14-17].

Therefore, in our approach, the hybrid genetic algorithm, which combines the recombinative power of GA with the local selection of SA, is investigated to find the optimal solution of equation (12) for nonlinear channel blind equalization. The flowchart of the proposed GASA algorithm is described in Fig. 3. For our particular application, the Bayesian likelihood shown in equation (12) is utilized as the fitness function in the proposed GASA. And the typical selection of SA is reversed to have its fitness function maximized, which means uphill moves are always accepted, whereas downhill moves are accepted depending on the acceptance probability. For example, the function "*SA-selection (new, old, T)*" calculates the acceptance probability "*P=exp(-(old-new)/T)*". If "*new>old*", a "*new*" solution is selected, which means that an uphill move is always accepted. And also, if "*new≤ old*" and "*P>random number in [0, 1]*", a "*new*" solution will be selected, which means that a downhill move is occasionally accepted, depending on *P*. An "*old*" solution will be selected for all other cases. This selection of SA allows a downhill move (same as an uphill move in a typical SA which minimizes the fitness function) to explore the search space at higher temperatures, and to exploit the search space acceptance of the best solution's individual at lower temperatures. Moreover, the population stores a diversity of annealing schedules by using random generated initial temperature for each individual in the population, i.e., some individuals explore the search space with high initial temperature and at the same time some individuals exploit the search space with low initial temperature. Thus the proposed GASA can search the channel output states which maximize the Bayesian likelihood, and it can reach the optimal global solution with a relatively high speed even when it is trapped in a local solution. Its performance is compared with those of a conventional genetic algorithm (GA) and a simplex GA introduced in [11].

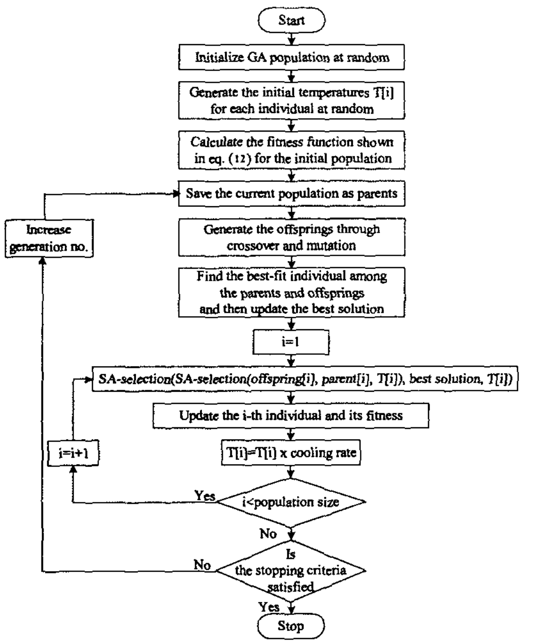


Fig. 3. Flowchart of the proposed GASA.

5. SIMULATION RESULTS AND PERFORMANCE ASSESSMENTS

The blind equalizations with GA, simplex GA, and GASA are taken into account to show the effectiveness of the proposed hybrid algorithm. Two nonlinear channels in [11] and [22] are evaluated in the simulations. Channel 1 is shown in Table 1, and channel 2 is as follows.

$$H(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2},$$

$$D_1 = 1, D_2 = 0.2, D_3 = 0.0, D_4 = 0.0, \text{ and } d=1$$

In channel 2, the channel order p, the equalizer order q, and the time delay d are 2, 1, 1, respectively. Thus the output of the equalizer should be $\hat{s}(k-1)$, and the sixteen desired channel states ($2^{p+q+1} = 16$) and the eight channel output states ($2^{p+1} = 8,\ a_1, a_2, a_3, \cdots, a_8$) may be observed at the receiver in the noise-free case. Those are shown in Table 2. The coefficients of channel 2 are symmetric, which means this channel has a linear phase characteristic. In this case, the number of observed channel output states becomes six instead of eight because a_2 and a_5, and a_4 and a_7 always have same values, 1.0219 and -0.7189 for this channel, respectively. However, in our simulations, each of all eight channel output states, $a_1, a_2, a_3, \cdots, a_8$, are searched and evaluated for more general cases.

The parameters of the optimization environments for each of the algorithms are included in Table 3, and these are fixed for all experiments. The choice of these specific parameter values is not critical in the performance of the proposed GASA. It is shown that the same quantities of population size, crossover rate, and mutation rate are used for the performance comparisons.

In the experiments, 10 independent simulations for each of two channels with five different noise levels (SNR=5, 10, 15, 20 and 25db) are performed with 1000 randomly generated transmitted symbols, and the results are averaged. The three algorithms, GA, simplex GA and proposed GASA, have been implemented in a batch way in order to obtain an accurate comparison among them. The computational efforts (the number of fitness function evaluations required for each generation) in GA and GASA are the same, while that in the simplex GA is greater by as much as Ω because of its concurrent version of the simplex operator. It means that, if the maximum number of generations is 100, and $\Omega = 4$ as in our simulations, 400(100×4) additional evaluations of the fitness function are re-

Table 2. The desired channel states and channel output states in channel 2

Nonlinear channel with $H(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$, $D_1 = 1, D_2 = 0.2, D_3 = 0.0, D_4 = 0.0$, and d=1							
Transmitted symbols				Desired channel states			Output of equalizer
$s(k)$	$s(k-1)$	$s(k-2)$	$s(k-3)$	$\hat{y}(k)$	$\hat{y}(k-1)$	By channel output states, $\{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8\}$	$\hat{s}(k-1)$
1	1	1	1	2.0578	2.0578	(a_1, a_1)	1
1	1	1	-1	2.0578	1.0219	(a_1, a_2)	1
1	1	-1	1	1.0219	-0.1679	(a_2, a_3)	1
1	1	-1	-1	1.0219	-0.7189	(a_2, a_4)	1
-1	1	1	1	1.0219	2.0578	(a_5, a_1)	1
-1	1	1	-1	1.0219	1.0219	(a_5, a_2)	1
-1	1	-1	1	0.1801	-0.1679	(a_6, a_3)	1
-1	1	-1	-1	0.1801	-0.7189	(a_6, a_4)	1
1	-1	1	1	-0.1679	1.0219	(a_3, a_5)	-1
1	-1	1	-1	-0.1679	0.1801	(a_3, a_6)	-1
1	-1	-1	1	-0.7189	-0.7189	(a_4, a_7)	-1
1	-1	-1	-1	-0.7189	-1.0758	(a_4, a_8)	-1
-1	-1	1	1	-0.7189	1.0219	(a_7, a_5)	-1
-1	-1	1	-1	-0.7189	0.1801	(a_7, a_6)	-1
-1	-1	-1	1	-1.0758	-0.7189	(a_8, a_7)	-1
-1	-1	-1	-1	-1.0758	-1.0758	(a_8, a_8)	-1

Table 3. Parameters of the optimization environments. 〈 () for channel 2〉

GA	Population size	50(100)
	Maximum number of generation	100
	Crossover rate	0.8
	Mutation rate	0.1
Simplex GA	Population size	50(100)
	Maximum number of generation	100
	Crossover rate	0.8
	Mutation rate	0.1
	Elitist number N	4
	Ω in the concurrent simplex method	4
GASA	Population size	50(100)
	Maximum number of generation	100
	Crossover rate	0.8
	Mutation rate	0.1
	Random initial temperature	[0, 1]
	Cooling rate	0.99

quired for the simplex GA. The averaged fitness functions in successive generations with 25db are shown in Fig. 4 for each of the two channels. It is observed that the proposed GASA converges with the highest speed because of its diversity of annealing schedules as mentioned in the previous section. Fig. 5 shows the averaged convergence speed (generation no.) for the fitness functions driven by the simplex GA and by the proposed GASA to reach within 10% difference with the optimal fitness function (conventional GA does not reach within 100 generations). We also measure the normalized root mean squared errors (NRMSE) for the estimation of channel output states, defined by equation (13), and they are shown in Fig. 6.

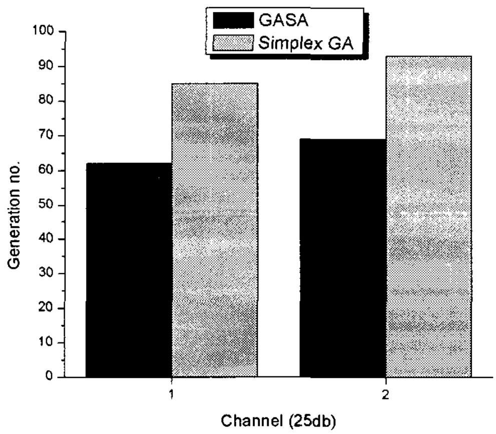


Fig. 5. Averaged generation no. to reach within 10% difference with optimal fitness function.

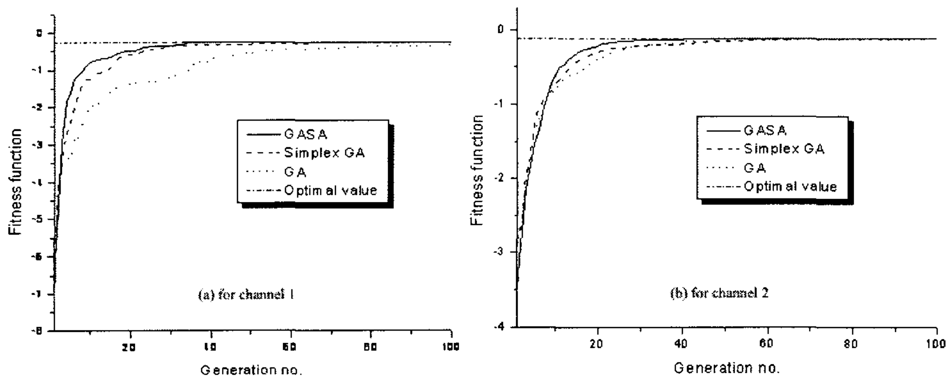


Fig. 4. Averaged fitness functions in successive 100 generations: (a) channel 1, (b) channel 2.

$$\text{NRMSE}=\frac{1}{\|a\|}\sqrt{\frac{1}{m}\sum_{i=1}^{m}\|a-\hat{a}_i\|^2} \tag{13}$$

where $\boldsymbol{a}$ is the dataset of optimal channel output states, $\hat{a}_i$ is the dataset of estimated channel output states, and m is the number of simulations performed (m=10). The GASA presents the lowest NRMSE over all of the SNR ranges, and it means that the proposed hybrid genetic algorithm is a very effective way to find optimal output states for nonlinear channel blind equalization. A sample of 1000 received symbols under 5db SNR for channel 2 and their desired channel states constructed from the estimated channel output states by GASA is shown in Fig. 7. Finally, the bit error rates (BER) are checked and summarized in Table 4. It is shown that the BER, with the estimated channel output states by GASA, is almost same as the one with the optimal output states for both of channel 1 and 2.

6. CONCLUSIONS

A hybrid genetic algorithm merged with SA (GASA) is investigated to find the optimal channel output states for nonlinear channel blind equalization. In this approach, the complex modeling of an unknown nonlinear channel becomes unnecessary by constructing the desired channel states directly from the estimated channel output states. It has been shown that the proposed GASA with the Bayesian likelihood as the fitness function offers better performance than conventional GA

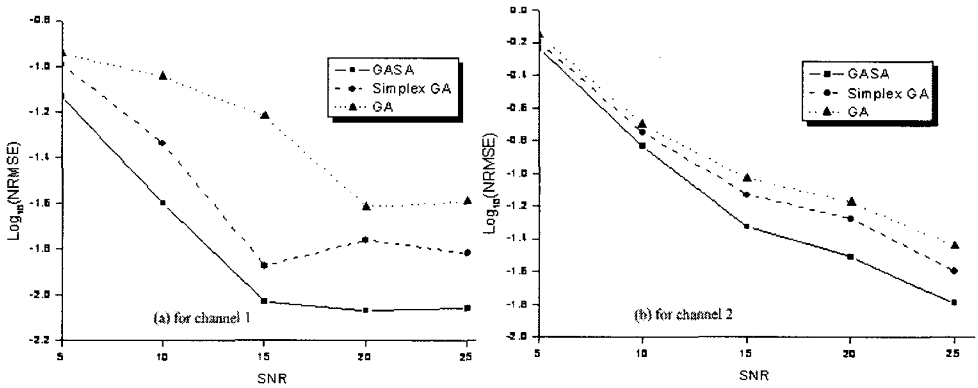


Fig. 6. NRMSE: (a) channel 1, (b) channel 2.

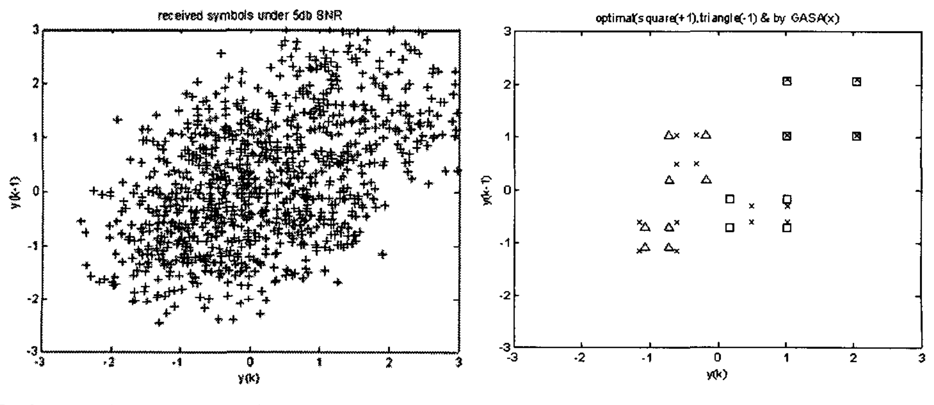


Fig. 7. A sample of received symbols for channel 2 and their desired channel states by GASA.

Table 4. Averaged BER(no. of errors/no. of transmitted symbols) for channels 1 and 2

SNR \ Estimation		with optimal states	GASA	Simplex GA	GA
Channel 1	5db	0.0797	0.0815	0.0824	0.0816
	10db	0.0121	0.0120	0.0128	0.0136
	15db	0	0	0	0.0003
	20db	0	0	0	0
	25db	0	0	0	0
Channel 2	5db	0.0970	0.1070	0.1162	0.1210
	10db	0.0420	0.0460	0.0492	0.0502
	15db	0.0100	0.0112	0.0114	0.0112
	20db	0.0008	0.0008	0.0008	0.0008
	25db	0	0	0	0

and simplex GA. It successively estimates the channel output states with relatively high speed and accuracy. Thus a RBF equalizer, based on GASA, can be a possible solution for nonlinear blind channel equalization problems. For further research and real-time use, the searching speed of the proposed GASA under more complex optimization environments, such as those with high dimensional channels and equalizer orders should be studied and evaluated.

7. REFERENCES

[1] E. Biglieri, A. Gersho, R. D. Gitlin, and T.L. Lim, "Adaptive cancellation of nonlinear intersymbol interference for voiceband data transmission," *IEEE J. Selected Areas Commun.* SAC-2(5), pp. 765-777, 1984.

[2] J. G. Proakis, *Digital Communications, Fourth Edition*, McGraw-Hill, New York, 2001.

[3] X. R. Cao and J. Zhu, "Blind equalization with a linear feedforward neural network," *Proc. 5^{th} European Symp. On Artificial Neural Networks, ESANN'97*, Bruges, Belgium, pp. 267-272, 1997.

[4] E. Serpedin and G. B. Giannakis, "Blind channel identification and equalization with modulation-induced cyclostationarity," *IEEE Trans. Siganl Processing*, Vol. 46, pp. 1930-1944, 1998.

[5] Y. Fang, W. S. Chow, and K. T. Ng, "Linear neural network based blind equalization," *Siganl Processing*, Vol. 76, pp. 37-42, 1999.

[6] T. Stathaki and A. Scohyers, "A constrained optimization approach to the blind estimation of Volterra kernels," *Proc. of the IEEE International Conf. on ASSP*. Vol. 3, pp. 2373-2376, Munich, Germany, 1997.

[7] G. K. Kaleh and R. Vallet, "Joint parameter estimation and symbol detection for linear or nonlinear unknown channels," *IEEE Trans. Commun.* Vol. 42, pp. 2406-2413, 1994.

[8] D. Erdogmus, D. Rende, J.C. Principe, and T.F. Wong, "Nonlinear channel equalization using multilayer perceptrons with information theoretic criterion," *Proc. Of IEEE workshop Neural Networks and Signal Processing*, MA, U.S.A., pp. 443-451, 2001.

[9] X. Liu and T. Adaly, "Canonical piecewise linear network for nonlinear filtering and its application to blind equalization," *Signal Process.*, Vol. 61, No.2, pp. 145-155, 1997.

[10] I. Santamaria, C. Pantaleon, L. Vielva, and J. Ibanez, "Blind Equalization of Constant Modulus Signals Using Support Vector

Machines," *IEEE Trans. Signal Processing*, Vol. 52, No. 6, pp. 1773-1782, 2004.

[11] H. Lin and K. Yamashita, "Hybrid simplex genetic algorithm for blind equalization using RBF networks," *Mathematics and Computers in Simulation*, Vol. 59, pp. 293-304, 2002.

[12] D. E. Goldberg, *Genetic algorithms in search, optimization and machine learning*, Addison-Wesley, Reading, MA, 1989.

[13] F. Romeo and A. Sangiovanni-Vincentelli, "A theoretical framework for simulated annealing," *Algorithmica*, Vol. 6, pp. 302-345, 1991.

[14] A. H. Mantawy, Y. L. Abdel-Magid, and S. Z. Selim, "Integrating genetic algorithms, tabu search, and simulated annealing for the unit commitment problem," *IEEE Trans. Power Systems*, Vol. 14, No. 3, pp. 829-836, Aug. 1999.

[15] B. Li and W. Jiang, "A novel stochastic optimization algorithm," *IEEE Trans. SMC-B*, Vol. 30, No. 1, pp. 193-198, Feb. 2000.

[16] K. P. Wong and Y. W. Wong, "Combined genetic algorithm/simulated annealing/fuzzy set approach to short-term generation scheduling with take-or-pay fuel contract," *IEEE Trans. Power Systems*, Vol. 11, No. 1, pp. 128-136, 1996.

[17] V. Savchenko and L. Schmitt, "Reconstructing occlusal surfaces of teeth using a genetic algorithm with simulated annealing type selection," *Proceedings of the sixth ACM symposium on Solid modeling and applications*, Michigan, U.S.A., pp. 39-46, 2001.

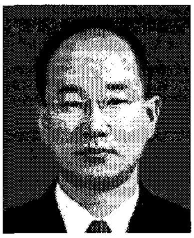
[18] S. Chen, B. Mulgrew, and P. M. Grant, "A Clustering Technique for Digital Communications Channel Equalization Using Radial Basis Function Networks," *IEEE Trans. Neural Networks*, Vol. 4, pp. 570-579, 1993.

[19] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, NewYork, Wiley, 1973.

[20] J. Lee, C. Beach, and N. Tepedelenlioglu, "Channel Equalization using Radial Basis Function Network," *ICASSP*, Vol. 3, pp. 1719-1722, Atlanta, Georgia, U.S.A., 1996.

[21] H. Lin and K. Yamashita, "Blind equalization using parallel Bayesian decision feedback equalizer," *Mathematics and Computers in Simulation*, Vol. 56, pp. 247-257, 2001.

[22] S. K. Patra and B. Mulgrew, "Fuzzy techniques for adaptive nonlinear equalization," *Signal Process*, Vol. 80, pp. 985-1000, 2000.



Han, Soowhan

1986. 2 Electronics, Yonsei University (B.S.)
1990. 3 Electrical & Computer Eng., Florida Institute of Technology (M.S.)
1993. 6 Electrical & Computer Eng., Florida Institute of Technology (Ph.D.)
1994. 3~1997. 2 Assistant prof., Dept. of Computer Eng., Kwandong University
1997. 3~present Associate prof., Dept. of Multimedia Eng., Dongeui University
2003. 3~2004. 2 Visiting prof., University of Alberta, Canada
Research interests: Digital Signal & Image Processing, Pattern Recognition, Fuzzy Logic & Neural Networks

Han, Changwook

1994 Dept. of Electronics, Youngnam Univ.(B.S.)
1996 Dept. of Electronics, Youngnam Univ.(M.S.)
2002 Dept. of Electronics, Youngnam Univ.(Ph.D.)
2002~2003 : University of Alberta, Post-Doc.
2004~present : Research Prof., in School of Electrical Eng. and Computer Science, Youngnam Univ.
Research interest : Fuzzy theory, Neural Networks, Genetic algorithm, Granular computing