A Modified FCM for Nonlinear Blind Channel Equalization using RBF Networks

Soo whan Han, *Member, KIMICS*

Abstract**—**In this paper, a modified Fuzzy C-Means (MFCM) algorithm is presented for nonlinear blind channel equalization. The proposed MFCM searches the optimal channel output states of a nonlinear channel, based on the Bayesian likelihood fitness function instead of a conventional Euclidean distance measure. In its searching procedure, all of the possible desired channel states are constructed with the elements of estimated channel output states. The desired state with the maximum Bayesian fitness is selected and placed at the center of a Radial Basis Function (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 that of a hybrid genetic algorithm (GA merged with simulated annealing (SA): GASA), and the relatively high accuracy and fast searching speed are achieved.

Index Terms — nonlinear blind equalization, modified Fuzzy C-Means, RBF equalizer.

I. 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.

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 presented 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. The approaches with nonlinear structures such as multilayer perceptrons and piecewise linear networks, being trained to minimize some cost function, have been investigated in [8] and [9], respectively. However, in those methods, the structure and complexity of the nonlinear equalizer must be specified in advance either. The support vector (SV) equalizer proposed by Santamaria et al. [10] can be a possible solution for both of linear and nonlinear blind channel equalization at the same time, but it still suffers from the high computational cost of its iterative reweighted quadratic programming procedure. A unique approach to nonlinear channel blind equalization was offered 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. The desired channel states were constructed from these estimated channel output states, and placed at the center of their RBF equalizer. With this method, the complex modeling of the nonlinear channel can be avoided. Recently this approach has been implemented with a hybrid genetic algorithm (GA merged with simulated annealing (SA): GASA) instead of the simplex GA. The resulting better performance in terms of speed and accuracy has been reported in [12]. However, for the real-time use, the estimation accuracy and convergence speed in search of the optimal channel output states needs further improvement.

Thus a new modified Fuzzy C-Means (MFCM) algorithm to find the optimal output states of a nonlinear channel is proposed in this study. The conventional Fuzzy C-Means (FCM) algorithm, based on Euclidean distance measure, was introduced in [13], and it has been widely used for pattern clustering problems. In the proposed MFCM, the construction stage for the possible data set of desired channel states by the elements of channel output states and the selection stage by the Bayesian likelihood fitness function are added to the

---

Manuscript received January 9, 2007.

Soowhan Han is with the Department of Multimedia Eng., Dongeui University, Busan, 614-714, Korea (Tel:+82-51-890-2700, Fax:+82-51-890-2706, Email: swhan@deu.ac.kr)

This study was supported by Dongeui University research fund 2006AA182

conventional FCM algorithm. These two additional stages make it possible to search the optimal output states of a nonlinear blind channel. The proposed MFCM shows the relatively high estimation accuracy with fast convergence speed, and its performance is compared with the one using the GASA. In the experiments, three different nonlinear channels are evaluated. The optimal output states of each of nonlinear channels are estimated by using both of MFCM and GASA. From these estimated output states, the desired channel states 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 [14].

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, MFCM with a Bayesian fitness function is introduced. The simulation results, including comparisons with GASA and the conclusions, are provided in sections 5 and 6, respectively.

II. NONLINEAR CHANNEL EQUALIZATION USING RBF NETWORKS

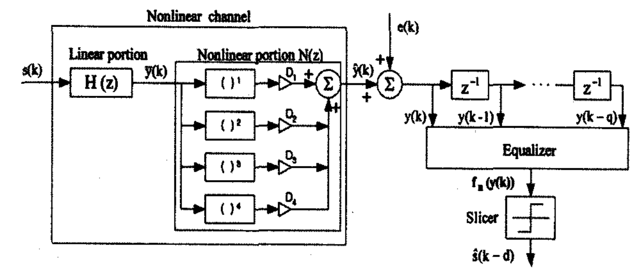


Fig.1 The structure of a nonlinear channel equalization system.

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,

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

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

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

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

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

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

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

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

The noise-free observation vector $\hat{\mathbf{y}}(k)$ 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.

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

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

The task of the equalizer is to recover the transmitted symbols *s(k-d)* based on the observation vector ***y(k)***. Because of the additive white Gaussian noise, the observation vector ***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 [15] can be applied to derive the optimal solution for the equalizer, and this optimal Bayesian equalizer solution is given by equations (8) and (9).

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

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

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, y_i^{+1} and 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) [16].

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

where n is the number of hidden units, c_i are the RBF centers, ρ_i is the width of the i^{th} units 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 a 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).

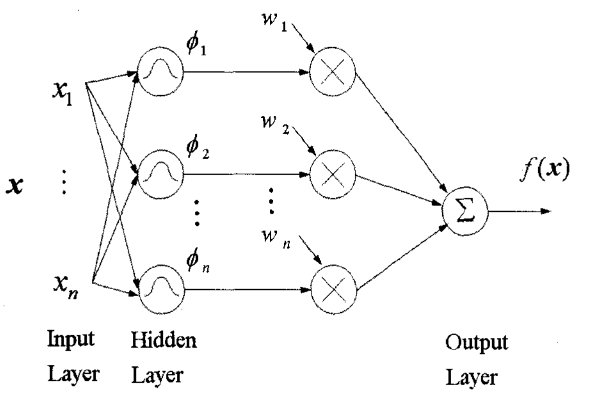


Fig. 2 The structure of a RBF network.

III. 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 is equal to 1, the time delay d is also set to 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 this table, 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 $a_1=1.89375, a_2=-0.48125, a_3=0.53125, a_4=-1.44375$ for this particular channel. The length of dataset, $\tilde{n}$, is determined by the channel order, p, such as $2^{p+1}=4$. In general, if q=1 and d=1, the desired 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.

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 [14][17].

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

where $f_B^{+1}(k)=\sum_{i=1}^{n_s^{+1}}\exp(-\|y(k)-y_i^{+1}\|^2/2\sigma_e^2)$, $f_B^{-1}(k)=\sum_{i=1}^{n_s^{-1}}\exp(-\|y(k)-y_i^{-1}\|^2/2\sigma_e^2)$ 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).

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

The optimal channel output states, which maximize the fitness function FF, cannot be obtained with the conventional gradient-based methods, because the mathematical formulation between the channel output states and FF cannot be accomplished without knowing the channel structure. These are shown in [11]. For carrying out search of these optimal channel output states, a new modified version of the FCM (MFCM) is developed, and its performance is compared with that of GASA introduced in [12].

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

IV. A MODIFIED FUZZY C-MEANS ALGORITHM (MFCM)

In comparison with the standard version of the FCM, the proposed modification of the clustering algorithm comes with two additional stages. One of them concerns the construction stage of possible data set of desired channel states with the derived elements of channel output states. The other is the selection stage for the optimal desired channel states among them based on the Bayesian likelihood fitness function. For the channel shown in Table 1, the four elements of channel output states are required to construct the optimal desired channel states. If the candidates, $\{c_1,c_2,c_3,c_4\}$, for the elements of optimal channel output states $\{a_1,a_2,a_3,a_4\}$, are extracted from the centers of a conventional FCM algorithm, twelve (4!/2) different possible data set of desired channel states can be constructed by completing matching between $\{c_1,c_2,c_3,c_4\}$ and $\{a_1,a_2,a_3,a_4\}$. For the fast matching, the arrangements of $\{c_1,c_2,c_3,c_4\}$ are saved to the set C such as $C(1)$=1,2,3,4, $C(2)$=1,2,4,3, ...,$C(12)$=3,2,1,4 before the search process starts. For example, $C(2)$=1,2,4,3 means the desired channel states is constructed with c_1 for a_1, c_2 for a_2, c_4 for a_3, and c_3 for a_4 in Table 1. At a next stage, a data set of desired channel states, which has a maximum Bayesian fitness value as described by (12), is selected. This data set is utilized as a center set used in the FCM algorithm. Subsequently the partition matrix U is updated and a new center set is sequentially derived with the use of this updated matrix U. The new four candidates for the elements of optimal output states are extracted from this new center set based on the relation presented in Table 1 (each value of the new $\{c_1,c_2,c_3,c_4\}$ is replaced with $\{a_1,a_2,a_3,a_4\}$ in the selected data set, respectively). These steps are repeated until the Bayesian likelihood fitness function has not been changed or the maximum number of iteration has been reached. The proposed MFCM algorithm has the following pseudo-code and its flowchart is shown in Fig. 3.

begin
save arrangements of candidates, $\{c_1,c_2,c_3,c_4\}$, *to C*
randomly initialize the candidates, $\{c_1,c_2,c_3,c_4\}$
while (new fitness function–old fitness function) <Th
for k=1 to C size
map the arrangement of candidates, C[k] , to $\{a_1,a_2,a_3,a_4\}$
construct a set of desired channel states based on the relation shown in Table 1
calculate its fitness function (FF[k]) by equation (12)
end
find a data set which has a maximum FF in k=1..C
update the membership matrix U by the data set utilized as a center set in the conventional FCM algorithm
derive a new center set by the U
extract the candidates, $\{c_1,c_2,c_3,c_4\}$, *from the new center set based on the relation shown in Table 1*
end
end

In the search process carried out by the MFCM, a data set for the desired channel states which exhibits a maximum fitness value is always selected, and the candidates $\{c_1,c_2,c_3,c_4\}$ for the elements of channel output states are extracted from the data set by using the pre-established relation in Table 1. This means that the set of desired channel states produced by MFCM is always close to the optimal set and it has the same structure as shown in Table 1. Thus the centers of the first half in its output present the desired channel states for $Y_{1,1}^{+1}$ and the rest present for $Y_{1,1}^{-1}$, or reversely. In addition, in the pseudo-code, the MFCM checks all of the possible arrangements, C, to find the data set which has a maximum FF in *while*-loop. However, for the fast searching of the MFCM, it is not necessary to keep this work during the entire procedure. The MFCM is forced to choose the data set with a maximum value of the *FF* and each value of the new candidates $\{c_1, c_2, c_3, c_4\}$ for next loop is always replaced with $\{a_1, a_2, a_3, a_4\}$ in the selected data set, respectively. Therefore, after the first couple of *while*-loops, the desired channel states constructed with the arrangement $C(1)$ has the maximum *FF* and the selected index *k* is quickly going to "1". This effect will be clearly shown in the experiments.

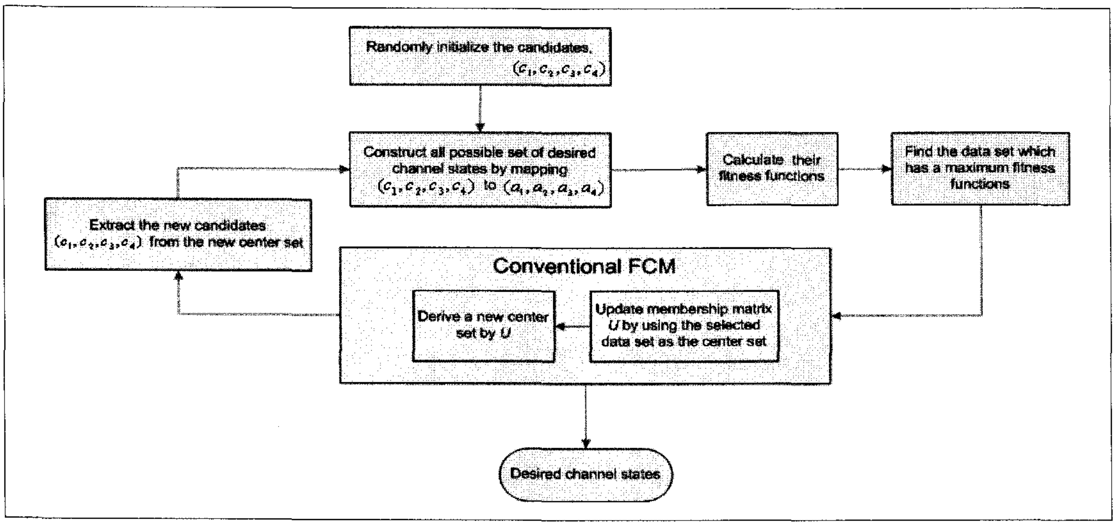


Fig. 3 MFCM flowchart

V. SIMULATION RESULTS AND PERFORMANCE ASSESSMENTS

The blind equalizations with GASA and MFCM are taken into account to show the effectiveness of the proposed algorithm. Three nonlinear channels in [11] are evaluated in the simulations. Channel 1 is shown in Table 1, and the other two channels are as follows.

Channel 2:
$H(z) = 0.5 + 1.0^{-1}$, $D_1 = 1, D_2 = 0.1, D_3 = -0.2, D_4 = 0.0$, and *d*=1
Channel 3:
$H(z) = 0.5 + 1.0^{-1}$, $D_1 = 1, D_2 = 0.0, D_3 = -0.9, D_4 = 0.0$, and *d*=1

The parameters of the optimization environments for each of the algorithms are included in Table 2, and these are fixed for all experiments. The choice of these specific parameter values is not critical to the performance of GASA and MFCM, and the equation (12) is utilized as the fitness function for both algorithms.

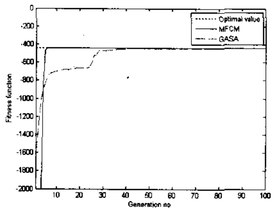
Table 2 Parameters of the optimization environments.

GASA	Population size	50
	Maximum number of generation	100
	Crossover rate	0.8
	Mutation rate	0.1
	Random initial temperature	[0, 1]
	Cooling rate	0.99
MFCM	Maximum number of iteration	100
	Minimum amount of improvement	10^{-5}
	Exponent for the matrix U	2
	Random initial output states	[-1 1]

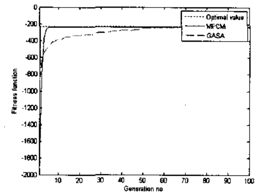
In the experiments, 10 independent simulations for each of three channels with five different noise levels (SNR=5,10,15,20 and 25db) are performed with 1000 randomly generated transmitted symbols, and the results are averaged. The proposed MFCM and GASA have been implemented in a batch way in order to obtain an accurate comparison. The averaged fitness functions with the desired channel states derived by MFCM and GASA under 25db noise level are shown in Fig. 4 for each of the three channels. The normalized root mean squared errors (NRMSE) for the estimation of channel output states, defined by equation (13), is also measured and they are shown in Fig. 5.

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

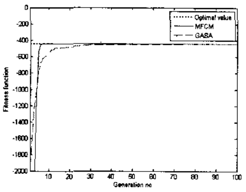
where $\boldsymbol{a}$ is the dataset of optimal channel output states, $\hat{\boldsymbol{a}}_i$ is the dataset of estimated channel output states, and m is the number of experiments performed (*m*=10). From Fig. 4 and 5, it is observed that the proposed MFCM presents the higher fitness function with fast speed and the lower NRMSE for all three channels. It means that the proposed MFCM is a very effective way to find the optimal output states for nonlinear channel blind equalization. A sample of 1000 received symbols under 5db SNR for channel 1 and its desired channel states constructed from the estimated channel output states by MFCM and GASA is shown in Fig. 6.



(a) channel 1

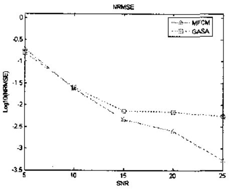


(b) channel 2

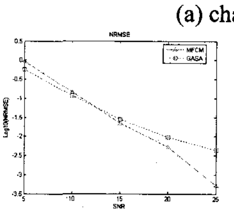


(c) channel 3

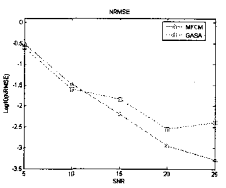
Fig. 4 Averaged fitness functions by MFCM and GASA.



(a) channel 1

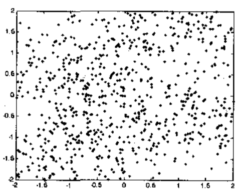


(b) channel 2



(c) channel 3

Fig. 5 NRMSE by MFCM and GASA.



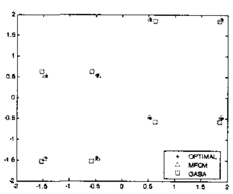
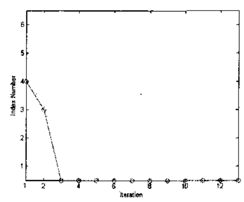
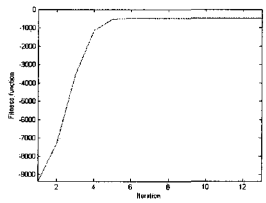


Fig. 6 A sample of received symbols for channel 1 and its desired channel states by MFCM and GASA.

Additionally, the searching speed for both of two algorithms is directly compared. As mentioned in the end of section 4, for the fast convergence speed of MFCM, it is not necessary to check all of the possible arrangements, C, in *while*-loop during the entire searching procedure because the new candidates for next loop are always updated by using the arrangement $C(1)$. The selection index k for the maximum FF is not changed after the first couple *while*-loops, and it is quickly going to "1". A sample of variations of index k and the fitness function during the searching procedure is shown in Fig. 7. Thus the *for*-loop in the pseudo-code of MFCM is skipped if the index k is not changed during the last 5 epochs. The searching time for MFCM and GASA is shown in Table 3, and the proposed MFCM offers much faster searching speed for all three channels because of its simple structure. Finally, the bit error rates (BER) are also checked by using a RBF equalizer. The desired channel states are constructed from the estimated channel output states and utilized as the centers in the RBF equalizer. The results are summarized in Table 4. It is shown that the BER, with the estimated channel output states by MFCM, is almost same as the one with the optimal output states for all three channels.



(a) index variation



(b) its fitness function

Fig. 7 A sample of variation of selection index k during the searching procedure in MFCM.

Table 3 The averaged searching time(in sec) for MFCM and GASA(Simulation : Pentium4 2.6Ghz, 512M Memory, coded by Matlab 6.5)

Channel	SNR	GASA	MFCM
Channel 1	5db	70.1922	0.3188
	10db	68.8266	0.2984
	15db	68.6516	0.2781
	20db	69.0344	0.3469
	25db	69.2734	0.3812
Channel 2	5db	76.2453	0.3672
	10db	76.8344	0.2766
	15db	75.475	0.2375
	20db	76.0375	0.2297
	25db	76.2812	0.225
Channel 3	5db	77.2781	0.3094
	10db	77.7781	0.2469
	15db	77.5562	0.2094
	20db	77.4016	0.2375
	25db	79.0734	0.2109

Table 4 Averaged BER(no. of errors/no. of transmitted symbols) for three nonlinear channels

Channel	SNR	with optimal states	GASA	MFCM
Channel 1	5db	0.0799	0.0822	0.0810
	10db	0.0128	0.0128	0.0127
	15db	0	0.0001	0.0001
	20db	0	0	0
	25db	0	0	0
Channel 2	5db	0.1573	0.1592	0.1595
	10db	0.0487	0.0494	0.0492
	15db	0.0040	0.0039	0.0038
	20db	0	0	0
	25db	0	0	0
Channel 3	5db	0.1078	0.1092	0.1089
	10db	0.0271	0.0274	0.0272
	15db	0.0002	0.0003	0.0002
	20db	0	0	0
	25db	0	0	0

VI. CONCLUSIONS

A new modified fuzzy c-means algorithm is introduced 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 MFCM with the Bayesian likelihood as the fitness function offers the better performance than GASA. It successively estimates the channel output states with relatively high speed and accuracy. Thus a RBF equalizer, based on MFCM, can be a possible solution for nonlinear blind channel equalization problems. For further research and real-time use, the searching speed of the proposed MFCM under more complex optimization environments, such as those with high dimensional channels and equalizer orders should be studied and evaluated.

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)*, 1984, pp. 765-777.

[2] Proakis, *J. G., 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. 5th European Symp. on Artificial Neural Networks*, Bruges, Belgium, 1997, pp. 267-272.

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

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

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

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

[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., 2001, pp. 443-451.

[9] X. Liu and T. Adaly, "Canonical piecewise linear network for nonlinear filtering and its application to blind equalization," *Signal Processing*, vol. 61, 1997, pp. 145-155.

[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, 2004, pp. 1773-1782.

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

[12] Soowhan Han, Imgeun Lee and Changwook Han, "A New Hybird Genetic Algorithm for Nonlinear Channel Blind Equalization," *International Journal of Fuzzy Logic and Intelligent Systems,* 2004, pp. 259-265.

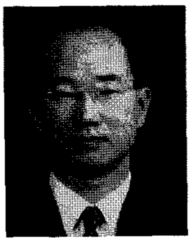
[13] J.C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*, Plenum Press, New York, 1981.

[14] 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, 1993, pp. 570-579.

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

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

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



Soowhan Han
Member of KIMICS. Received B.S. degree in electronics, Yonsei University, Korea, in 1986, and M.S. and Ph.D. degree in Electrical & Computer Eng., Florida Institute of Technology, U.S.A. in 1990 and 1993, respectively. From 1994 to 1996, he was an assistant professor of the Dept. of Computer Eng., Kwandong University, Korea. In 1997, he joined the Dept. of Multimedia Eng., Dongeui University, Korea, where he is currently a professor. His major interests of research include Digital Signal & Image Processing, Pattern Recognition and Neural Networks.