Blind linear/nonlinear equalization for heavy noise-corrupted channels

Soo-Whan Han, Sung-Dae Park, *Member, KIMICS*

Abstract— In this paper, blind equalization using a modified Fuzzy *C*-Means algorithm with Gaussian Weights (MFCM_GW) is attempted to the heavy noise-corrupted channels. The proposed algorithm can deal with both of linear and nonlinear channels, because it searches for the optimal channel output states of a channel instead of estimating the channel parameters in a direct manner. In contrast to the common Euclidean distance in Fuzzy *C*-Means (FCM), the use of the Bayesian likelihood fitness function and the Gaussian weighted partition matrix is exploited in its search procedure. The selected channel states by MFCM_GW are always close to the optimal set of a channel even the additive white Gaussian noise (AWGN) is heavily corrupted in it. Simulation studies demonstrate that the performance of the proposed method is relatively superior to existing genetic algorithm (GA) and conventional FCM based methods in terms of accuracy and speed.

Index Terms — Fuzzy *C*-Means, Gaussian Weighted Partition Matrix, Bayesian Equalizer, Blind Equalization, Linear/Nonlinear Channel.

I. INTRODUCTION

It is well known that most of channels in digital communication systems suffer from inter-symbol interference (ISI) due to non-ideal channel characteristics. The problem becomes more severe in the presence of AWGN. Furthermore, the nonlinear character of ISI that often arises in high speed communication channels degrades the performance of the overall digital communication system [1]. Thus a channel equalizer is necessary for in these circumstances to equalize the channel so as to recover the signal from the noise-corrupted received signal with minimum error probability. There exist two kinds of equalizers in digital communication systems: data aided (trained) equalizers and blind equalizers. For trained equalizers, a reference signal is required, increasing the bandwidth. As a result, blind equalizers are preferred in high-speed communication systems to reduce ISI without increasing overhead costs [2][3].

To date, most available blind channel equalization methods focus on linear channel equalization because of its inherent simplicity [4]-[6]. However, this paucity does not mean blind nonlinear equalization methods have less significance. Considering that nonlinear distortion exists in many communication systems such as high power amplifiers as well as high-density magnetic and optical storage channels, studying blind nonlinear system equalization methods has significant practical importance. Thus in this study, the blind equalization method for both of linear and nonlinear channels should be investigated at the same time.

Most of early works for blind nonlinear channel equalizations focus on the channel estimation. The blind estimation of Volterra kernels, which characterize nonlinear channels, was presented in [7] while a maximum likelihood (ML) method implemented via expectation-maximization (EM) was introduced in [8]. In spite of the advantages of these approaches, these methods are not free from limitations. The Volterra approach suffers from enormous computational complexity and slow convergences. Furthermore the ML approach requires some prior knowledge of the nonlinear channel structure to estimate the channel parameters. The approach with a nonlinear structure such as multilayer perceptrons, being trained to minimize some cost function, have been investigated in [9]. However, in this method, the structure and complexity of the nonlinear equalizer must be specified in advance. The support vector (SV) equalizer proposed by Santamaria et al. [10] can be a possible solution for both of linear

Manuscript received June 8, 2009 ; revised July 31, 2009.

Soowhan Han is with the Department of Multimedia Eng., Dong-Eui University, Busan, 614-714, Korea (Tel:+82-51-890-1690, Fax:+82-51-890-2706, Email: swhan@deu.ac.kr)

Sungdae Park is with the Department of Digital Culture and Contents Eng., Dong-Eui University, Busan, 614-714, Korea

This study was supported by Dong-Eui University research fund, 2008AA199.

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. The deterministic approach based on second order statistic (SOS) in [11] also has limited practical application because of its computationally expensive cost (requiring two matrix eigen-decompositions). Another SOS based method provided by Raz et al. [12] successfully offers a linealizing Volterra filter equalizer and the linearized channel that result from the cascade connection of the blind nonlinear channel with Volterra filter equalizer. However, this algorithm uses the oversampling technique that translates a single input single output (SISO) system to a single input multi output (SIMO) system. Furthermore the resulting oversampled channel matrix must be invertible over the transmitted symbols. For this method, the sampling rate for the received signal has to be higher than the baud rate, otherwise a multi-sensor array must be utilized. In addition, the signal to noise ratio (SNR) should be kept relatively high. A unique approach to blind channel equalization was offered by Lin et al. [13]. In their study they used the simplex GA method to estimate the optimal channel output states instead of estimating the channel parameters in a direct manner. 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 and it works well under a simple SISO communication environment. Additionally, this kind of approach can be applied to a linear channel either, because it estimates directly the channel output states not the parameters of the channel. Recently this approach has been implemented with a hybrid genetic algorithm (that is genetic algorithm, GA merged with simulated annealing (SA); GASA) [14] and a modified Fuzzy *C*-Means (MFCM) algorithm [15] instead of the simplex GA. The resulting better performance in terms of speed and accuracy has been reported.

However, for real-time use, the estimation accuracy and convergence speed in search of the optimal channel output states needs further improvement. In other words, those search algorithms should be robust to the heavy noise communication environments. This leads us to propose a new modified Fuzzy *C*-Means algorithm with Gaussian weights (MFCM_GW) to overcome the heavy noise effect in digital communication channels. In the proposed algorithm, the Gaussian weighted partition matrix is developed and applied to the previous version of MFCM [15] for the reduction of noise effect. The presented MFCM_GW can estimate the optimal output states with the relatively high accuracy and fast convergence speed even the received symbols are corrupted by a heavy AWGN. Its performance is compared with those of a simplex GA, a GASA and a MFCM. In the experiments, both of linear and nonlinear channels with the heavy noise (SNR=0,1,2,..,5db) are evaluated. The optimal output states of each of channels are estimated making use of each of four search algorithms. Using the estimated channel output states, the desired channel states are derived and then utilized to compute the decision probability of Bayesian equalizer for the reconstruction of transmitted symbols.

II. OPTIMAL BAYESIAN SOLUTION FOR CHANNEL EQUALIZATION

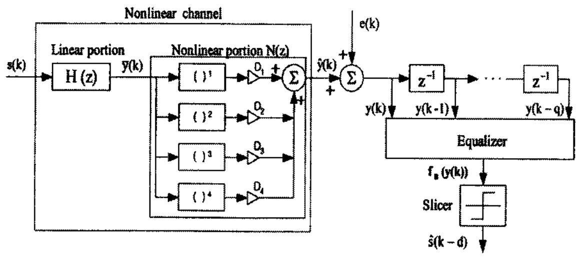


Fig. 1 An overall structure of channel equalization system.

A channel equalization system is shown in Fig. 1. A digital sequence *s(k)* is transmitted through the channel, which is composed of a linear portion described by *H(z)* and a nonlinear component *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 stands for the coefficient of the i^{th} nonlinear term. The transmitted symbol sequence *s(k)* is assumed to constitute an equiprobable and independent binary sequence taking values from a two-valued set$\{\pm 1\}$. We assume that the channel output is corrupted by an AWGN, *e(k)*. Given this, 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

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

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

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

The noise-free observation vector $\hat{y}(k)$ 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 (6) and (7), depending on the value of *s(k-d)*, where *d* is the desired time delay.

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

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

In case of a linear channel (D_1=1, D_2=0, D_3=0 and D_4=0), $\hat{y}(k)$ in (3), (5), (6) and (7) is replaced with $\bar{y}(k)$ in (1). 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. The determination of the value of *s(k-d)* becomes a decision problem. Bayes decision theory [16] provides the optimal solution to the general decision problem, and thus can be applied here to derive the optimal solution for the equalizer. The solution forming for the optimal Bayesian equalizer with the equiprobable transmitted symbols is given as follows [17][18]

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

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

where y_i^{+1} and y_i^{-1} are the desired channel states belonging to sets $Y_{q,d}^{+1}$ and $Y_{q,d}^{-1}$, respectively, and their number of elements in these sets are denoted by n_s^{+1} and n_s^{-1}. Furthermore, σ_e^2 is the noise variance.

The optimal equalizer solution in (8) depends on the desired channel states. In other words, the solution of blind channel equalization crucially depends on how to find the desired channel states, y_i^{+1} and y_i^{-1}, only from the observation vector ***y(k)***. The desired channel states can be constructed by using the relationship with the channel output states. It will be explained in the next section. Thus the proposed MFCM_GW algorithm is applied in search of the optimal output states of both of linear and nonlinear channels with the high accuracy and fast convergence speed. The optimal Bayesian decision probability in (8) is used to construct the fitness function of proposed algorithm, and it is also utilized as an equalizer, along with (9), for the reconstruction of the transmitted symbols.

III. CONSTRUCTION OF DESIRED CHANNEL STATES FROM CHANNEL OUTPUT STATES

The desired channel states, y_i^{+1} and y_i^{-1}, must be known for the Bayesian equalizer (8) - (9) in order to reconstruct the transmitted symbols. If the channel order is taken as *p*=2 with $H(z)=0.3482+0.8704z^{-1}+0.3482z^{-2}$, the equalizer order *q* is equal to 1, the time delay *d* is also set to 1, and the nonlinear portion is described by $D_1=1, D_2=0.2, D_3=0.0, D_4=0.0$ (see Fig. 1), then the sixteen different channel states ($2^{p+q+1}=16$) may be observed at the receiver in the noise-free case. Here 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, a_5, a_6, a_7, a_8}, where for this particular channel we have $a_1=2.0578, a_2=1.0219, a_3=-0.1679, a_4=-0.7189, a_5=1.0219,$ $a_6=0.1801, a_7=-0.7189$ and $a_8=-1.0758$. The length of dataset, $\tilde{n}$, is determined by the channel order, *p*, such as $2^{p+1}=8$. 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_2,a_3), (a_2,a_4), (a_5,a_1), (a_5,a_2), (a_6,a_3), (a_6,a_4), and (a_3,a_5), (a_3,a_6), (a_4,a_7), (a_4,a_8), (a_7,a_5), (a_7,a_6), (a_8,a_7), (a_8,a_8), respectively. This relation is always valid for the channel that has a one-to-one mapping between the channel inputs and outputs [13]. Thus the desired channel states can be derived from the channel output states if we assume that *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. In this particular channel for our simulations, the coefficients of channel are symmetric, which means the 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 have always 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.

It is known that the Bayesian likelihood (*BL*), given by (10), is always maximized with respect to the desired channel states derived from the optimal channel output states [18].

$$BL = \prod_{k=0}^{L-1} \max(f_B^{+1}(k), f_B^{-1}(k)) \tag{10}$$

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. Being more specific, the fitness function is taken as the logarithm of the *BL*, that is

$$FF = \sum_{k=0}^{L-1} \log(\max(f_B^{+1}(k), f_B^{-1}(k))) \tag{11}$$

The optimal channel output states, which maximize the fitness function *FF*, cannot be obtained with the use of the conventional gradient-based methods given the fact that the channel structure is not known in advance [13]. For carrying out search of these optimal output states in the heavy noise-corrupted channels, the MFCM_GW is utilized, and its performance is compared with those of three other search algorithms introduced in [13]-[15].

Table 1 The construction of desired channel states by using channel output states.

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, $\{a_1, a_2, \cdots, a_8\}$	$\hat{s}(k-1)$
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

IV. MODIGIED FUZZY C_MEANS WITH GAUSSIAN WEIGHTS (MFCM_GW)

The main idea of the proposed MFCM_GW for the problem of blind channel equalization is based on two important facts. First, as mentioned in Section 3, the desired channel states of a channel can be constructed by the channel output states and second, the Bayesian likelihood (*BL*) in (10) is always maximized with the optimal desired channel states. Therefore, 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 eight elements of channel output states ($2^{p+1}=8$) are required to construct the optimal desired channel states. If the candidates, $\{c_1,c_2,c_3,\ldots,c_8\}$, for the elements of optimal channel output states $\{a_1,a_2,a_3,\ldots,a_8\}$, are extracted from the centers of a conventional FCM algorithm (or randomly initialized at first), 20160(8!/2) different possible data sets of desired channel states can be constructed by completing matching between $\{c_1,c_2,c_3,\ldots,c_8\}$ and $\{a_1,a_2,a_3,\ldots,a_8\}$. To facilitate fast matching, the arrangements of $\{c_1,c_2,c_3,\ldots,c_8\}$ are saved to a certain mapping set C such as $C(1)$=1,2,3,4,5,6,7,8, $C(2)$=2,1,3,4,5,6,7,8, ..., $C(20160)$=7,6,5,4,3,2,1,8 before the search process starts. For example, $C(2)$=2,1,3,4,5,6,7,8 means that the set of desired channel states is constructed with c_2 for a_1, c_1 for a_2, c_3 for a_3, c_4 for a_4, and c_5 for a_5,..., *etc.*, in Table 1. The desired channel states for this set are described as $y_{i\_C(2)}$ ($y^{+1}_{i\_C(2)}$ and $y^{-1}_{i\_C(2)}$), and its fitness function in (11) is presented by $FF(2)$. The number of the possible arrangements C is rapidly increased by the channel order p in MFCM_GW. However, it is not necessary to construct all possible set of desired channel states during the entire procedure, because the set of desired channel states constructed by the arrangement $C(1)$ is always selected after the first couple of search procedure. It is explained more details in the end of this section. At the next stage, a data set of desired channel states, which has a maximum Bayesian fitness value, is selected as shown below

$$\begin{aligned}[index\_j,\ \max\_FF] \\ = \mathbf{max}(FF(1), FF(2), \cdots, FF(20160))\end{aligned} \tag{12}$$

This data set ($y_{i\_C(index\_j)}$), the set of desired channel states configured by the selected $C(index\_j)$, is utilized as a center set in the conventional FCM algorithm. Subsequently the partition matrix $\boldsymbol{U}$ is updated and a new center set is sequentially derived with the use of this updated matrix $\boldsymbol{U}$. However, the clustering process of conventional FCM algorithm is easily affected by a AWGN, because its partition matrix $\boldsymbol{U}$ and center set y_i are updated based on Euclidean distance measure [19]. As mentioned in section 2, the received symbol $\boldsymbol{y(k)}$ is a random process having conditional Gaussian density functions centered at each of the desired channel states because of the additive white Gaussian noise. Thus to avoid this noise effect, the MFCM_GW utilizes the Gaussian density function to update the membership matrix $\boldsymbol{U}$ and a new center set y_i by equations (13) and (14), respectively.

$$U_{ik}^{(m+1)} = \frac{exp(-\left\| y(k) - y_{i\_C(index\_j)}^{(m)} \right\|^2 / 2\sigma_e^2)}{\sum_{l=1}^{n_s} exp(-\left\| y(k) - y_{l\_C(index\_j)}^{(m)} \right\|^2 / 2\sigma_e^2)} \tag{13}$$

$$y_i^{(m+1)} = \sum_{k=0}^{L-1} U_{ik}^{(m+1)}\, y(k) \tag{14}$$

where $y_i^{(m+1)}$ is the estimated center set at the $(m+1)^{th}$ iteration and n_s is the total number of center vectors (n_s=16 for the channel in Table 1). The new eight candidates for the elements of optimal output states are extracted from this new center set, $y_i^{(m+1)}$, based on the relation presented in Table 1. The sixteen centers in the new center set are treated as the desired channel states constructed by the elements of channel output states, $\{a_1,a_2,a_3,\ldots,a_8\}$, shown in Table 1, and thus each value of the new $\{c_1,c_2,c_3,\ldots,c_8\}$is replaced with each one of the $\{a_1,a_2,a_3,\ldots,a_8\}$ in the new center set as in (15), respectively.

$$c_r^{(m+1)} = a_r \ \text{ in } \ y_i^{(m+1)} \ \text{ where } r=1,2,3,\ldots,8 \tag{15}$$

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_GW algorithm is described in Fig. 2.

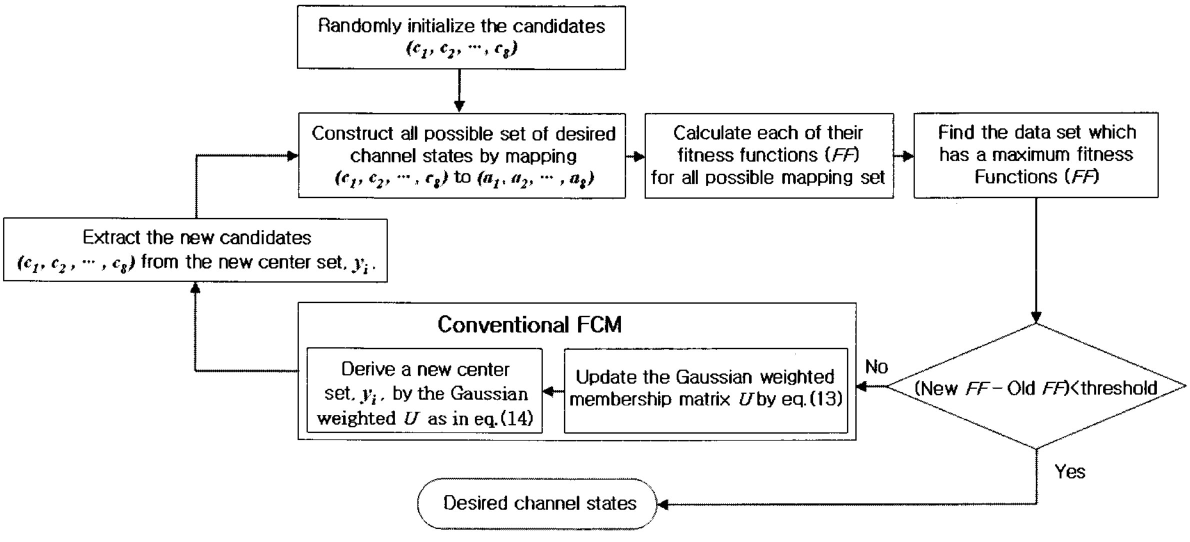


Fig. 2 The flowchart of MFCM_GW.

In the search process carried out by the MFCM_GW, a data set for the desired channel states which exhibits a maximum fitness value is always selected, and the candidates $\{c_1,c_2,c_3,\ldots,c_8\}$ for the elements of channel output states are extracted from the data set by using the pre-established relation as shown in Table 1. This means that the set of desired channel states produced by MFCM_GW is always close to the optimal set and it has the same structure as illustrated 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, as shown in Fig. 2, the MFCM_GW checks all of the possible arrangements, $C(1), C(2), \ldots, C(20160)$, to find the data set which has a maximum FF. However, for the fast searching of the MFCM_GW, it is not necessary to keep this work during the entire procedure. The derived new center set at the end of each epoch is treated as a data set for the desired channel states presented in Table 1 and each value of the new $\{c_1,c_2,c_3,\ldots,c_8\}$ for next loop is replaced with each of the $\{a_1,a_2,a_3,\ldots,a_8\}$ in the new center set, respectively, as shown in (15) (the new c_1 is replaced with a_1, the c_2 with a_2, the c_3 with a_3, the c_4 with a_4, and the c_5 with a_5,..., *etc.*). It means the new candidates, $\{c_1,c_2,c_3,\ldots,c_8\}$, are always updated by using the arrangement $C(1)$. Therefore, after the first couple of epochs, the set of desired channel states constructed by the arrangement $C(1)$ has always the maximum FF and the selected index j is quickly going to "1". This effect will be clearly shown in our experiments.

V. EXPERIMENTAL STUDIES

In this section, to demonstrate the effectiveness of the proposed method, we consider blind equalization realized with the use of the simplex GA[13], GASA[14], MFCM[15] and MFCM_GW, simultaneously. Two channels, one (channel 1) for a nonlinear model in [20] and the other (channel 2) for a linear model in [13], are discussed. Channel 1 is shown in Table 1 while channel 2 is described as follows.

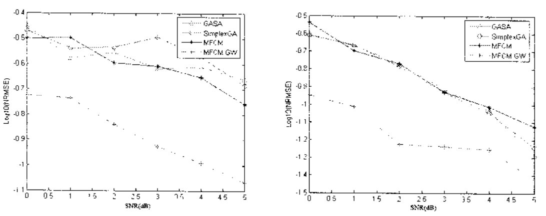
Channel 2(a linear model):
$H(z) = 0.5 + 1.0z^{-1}, D_1 = 1, D_2 = 0, D_3 = 0, D_4 = 0$, and d=1

In channel 1 for a nonlinear model, 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 ($2^{p+q+1} = 16$) composed of the eight channel output states ($2^{p+1} = 8$, $a_1,a_2,a_3,\ldots,a_8$) as shown in Table 1 will be observed at the receiver in the noise-free case. In channel 2 for a linear model, the nonlinear terms of channel, D_2, D_3, and D_4, are equal to zeros, and the eight desired channel states with the four channel output states ($2^{p+1} = 4$, a_1,a_2,a_3,a_4) exist because the channel order p is 1. The desired channel states, (a_1,a_1), (a_1,a_2), (a_3,a_1), (a_3,a_2), belong to $Y_{1,1}^{+1}$, and (a_2,a_3), (a_2,a_4), (a_4,a_3), (a_4,a_4) belong to $Y_{1,1}^{-1}$, where a_1,a_2,a_3,a_4 are 1.5, -0.5, 0.5 and -1.5, respectively.

In the experiments, 10 independent simulations for each of two channels with six different heavy-noise levels (SNR=0,1,2,3,4 and 5db) are performed with 1,000 randomly generated transmitted symbols. Afterwards the obtained results are averaged. The four search algorithms, simplex GA, GASA, MFCM and MFCM_GW, have been implemented in a batch mode to facilitate comparative analysis. With this regard, we determine the normalized root mean squared errors (NRMSE)

$$\text{NRMSE}=\frac{1}{\|\boldsymbol{a}\|}\sqrt{\frac{1}{N}\sum_{i=1}^{N}\|\boldsymbol{a}-\hat{\boldsymbol{a}}_i\|^2} \quad (16)$$

where $\boldsymbol{a}$ is the dataset of optimal channel output states, $\hat{\boldsymbol{a}}_i$ is the dataset of estimated channel output states in the i^{th} simulation, and N is the total number of independent simulations (N=10). As shown in Fig. 3, the proposed MFCM_GW comes with the lowest NRMSE for both channels, and the performance differences are more severe under the high noise levels such as SNR=0 and 1db, especially for channel 2. It is caused by the fact that the MFCM_GW uses the Gaussian weights shown in equations (13) and (14) to reduce the noise interference as mentioned in section 4. A sample of 1,000 received symbols under 0db SNR for channel 2 and its desired channel states constructed from the estimated channel output states by each of four search algorithms are shown in Fig. 4.



(a) channel 1 (b) channel 2

Fig. 3 NRMSE for channel 1(nonlinear) and channel 2(linear).

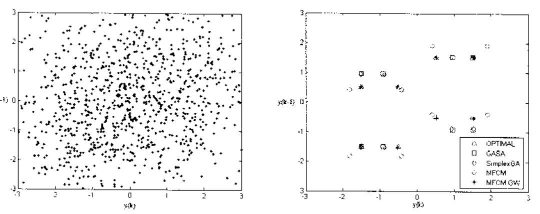
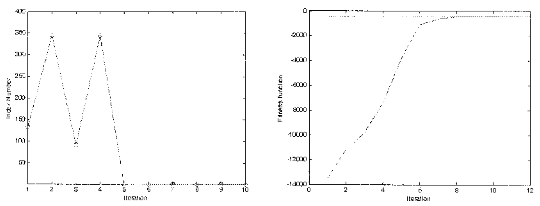


Fig. 4 A sample of received symbols under 0db SNR for channel 2 and its desired channel states produced by each of four search algorithms.

In addition, we compared the search time of the algorithms. As mentioned at the end of Section 4, for the fast convergence of the MFCM_GW, it is not necessary to check all of the possible arrangements (20160 arrangements for channel 1 and 12 for channel 2), C, during the entire procedure because the new candidates for next loop are always updated by using the arrangement $C(1)$. The selection index j for the maximum FF is not changed after the first couple loops, and it is quickly going to "1". A sample of variations of index j and the fitness function FF during the entire searching loops is shown in Fig. 5. Thus the selection stage for the index j in MFCM_GW is skipped if it has not changed during the last 5 epochs in our experiments. The relative search times of each of four algorithms for both channels, averaged by 10 independent simulations and normalized by the time of simplex GA under 0 db SNR (56.8953 real time sec for channel 1 and 30.9782 for channel 2 with Matlab 7.1 version of source code on Pentium 4), are included in Table 2; notably, the MFCM_GW and MFCM offer much faster search speed for both channels and this could be attributed to their simple structures. The basic architecture of MFCM_GW is shared with the one of MFCM introduced in [15]. However, all over the noise levels, the search speed of proposed MFCM_GW is much faster than those of anyone else including MFCM as shown in the performance of NRMSE.



(a) index j vs. epochs (b) its fitness function FF vs. epochs

Fig. 5 A sample of variation of the selection index j and the fitness function FF during the searching procedure of MFCM_GW for channel 1.

Table 2 The relative search times for each of four algorithms.

Channel	SNR	Simplex GA	GASA	MFCM	MFCM _GW
Channel 1 (nonlinear model)	0db	1.0000	0.9973	0.0525	0.0457
	1db	1.0362	1.0336	0.0562	0.0500
	2db	1.0370	1.0448	0.0523	0.0384
	3db	1.0390	1.0415	0.0579	0.0343
	4db	1.0422	1.0388	0.0902	0.0412
	5db	1.0416	1.0492	0.0583	0.0232
Channel	0db	1.0000	1.0134	0.0158	0.0099

2 (linear model)	1db	1.0286	1.0163	0.0152	0.0095
	2db	1.0582	1.0141	0.0135	0.0107
	3db	1.0584	1.0181	0.0112	0.0094
	4db	1.0137	1.0175	0.0105	0.0087
	5db	1.0267	1.0289	0.0089	0.0083

Finally, we investigated the bit error rates (BER) when using the Bayesian equalizer; refer to Table 3. It becomes apparent that the BER with the estimated channel output states realized by the MFCM_GW is almost the same as the one with the optimal output states for both channels. The decision boundaries of Bayesian equalizer with the optimal desired channel states and the channel states estimated by MFCM_GW for channel 1 are plotted in Fig. 6.

Table 3 Averaged BER(%) (no. of errors/no. of transmitted symbols)

Channel	SNR	with optimal states	Simplex GA	GASA	MFCM	MFCM_GW
Channel 1 (nonlinear model)	0db	21.03	21.19	21.65	21.51	21.49
	1db	19.93	20.22	20.23	20.22	20.43
	2db	17.81	18.99	18.57	18.55	18.18
	3db	16.39	17.61	16.47	16.29	16.33
	4db	13.58	14.20	14.11	13.82	13.69
	5db	11.93	12.24	12.49	11.98	11.91
Channel2 (linear model)	0db	18.16	18.71	18.69	20.60	20.72
	1db	16.98	17.62	17.64	17.93	17.14
	2db	14.83	15.23	15.27	14.92	14.89
	3db	12.74	12.94	12.95	12.83	12.91
	4db	10.21	10.28	10.28	10.25	10.14
	5db	8.96	9.14	9.03	9.08	9.05

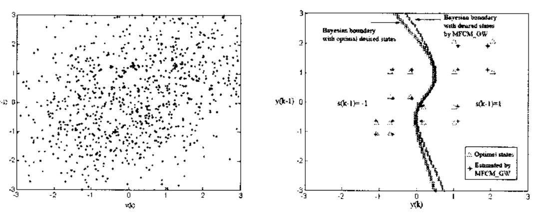


Fig. 6 A sample of decision boundaries with the optimal desired channel states and the channel states by MFCM_GW for channel 1 (SNR=0db).

VI. CONCLUSIONS

A modified Fuzzy *C*-Means clustering algorithm with Gaussian weights is presented and showed its application to blind equalization for both of linear and nonlinear channels with the heavy Gaussian noise (SNR=0,1,2,..,5db). In this method, the highly demanding modeling task of an unknown channel becomes unnecessary as the construction of the desired channel states is accomplished directly on a basis of the estimated channel output states. It has been shown that the proposed MFCM_GW with the Bayesian likelihood treated as the fitness function offers better performance in comparison to the solutions provided by the simplex GA, GASA, and the previous version of MFCM. In particular, it successively estimates the channel output states even when the received symbols are significantly corrupted by a heavy noise. It is caused by the fact that the proposed MFCM_GW uses the Gaussian weights instead of the common Euclidean distance measure. Therefore, the Bayesian equalizer based on MFCM_GW can be a valuable solution for the heavy noise communication environments. Our future research includes the use of MFCM_GW under more complex optimization environments, such as those encountered when dealing with channels of high dimensionality and equalizers of higher order. Additionally, for the real-time use, the MFCM_GW should not require the prior knowledge of channel order to estimate the optimal channel states.

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*, Vol.5, pp. 765-777, 1984.

[2] G. B. Giannakis and E. Serpedin, "Linear multichannel blind equalizers of nonlinear FIR volterra channels," *IEEE Transactions on Signal Processing*, vol. 45, pp. 67-81, 1997.

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

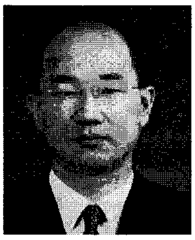
[4] E. Serpedin and G.B. Giannakis, “Blind channel identification and equalization with modulation-induced cyclostationarity”, *IEEE Trans. Signal Processing*, Vol. 46, pp. 1930-1944, 1998.

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

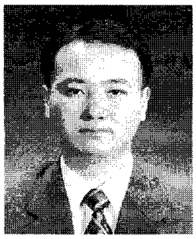
[6] Yun Ye and Saman S. Abeysekera, “Efficient blind estimation and equalization of non-minimum phase communication channels via the use of a zero forcing equalizer”, *Signal Processing*, Vol. 86, pp.1019-1034, 2006.

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

[8] 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.
[9] 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*, pp. 443-451, MA, U.S.A., 2001.
[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, pp. 1773-1782, 2004.
[11] R. Lopez-Valcarce and S. Dasgupta, "Blind equalization of nonlinear channels from second-order statistics," *IEEE Transactions on Signal Processing*, vol. 49, pp. 3084-3097, 2001.
[12] G. Raz and B. Van Veen, “Blind equalization and identification of nonlinear and IIR systems-a least squares approach”, *IEEE Trans. Signal Processing*, Vol. 48, pp. 192-200, 2000.
[13] 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.
[14] S. Han, W. Pedrycz and C. Han, “Nonlinear Channel Blind Equalization Using Hybrid Genetic Algorithm with Simulated Annealing”, *Mathematical and Computer Modeling*, Vol. 41, pp. 697-709, 2005.
[15] S. Han, “A Modified FCM for Nonlinear Blind Channel Equalization using RBF Networks,” *International Journal of KIMICS*, Vol. 5, pp. 35-41, 2007.
16] R.O. Duda and P.E. Hart, *Pattern Classification and Scene Analysis*, Wiley, New York, 1973.
[17] S. Chen, B. Mulgrew and M. Grant, “A clustering technique for digital communication channel equalization using radial basis function network”, *IEEE Trans. Neural Networks*, Vol. 4, pp. 570-579, 1993.
[18] H. Lin and K. Yamashita, “Blind equalization using parallel Bayesian decision feedback equalizer”, *Mathematics and Computers in Simulation*, Vol. 56, pp. 247-257, 2001.
[19] J.C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.
[20] S.K. patra and B. Mulgrew, “Fuzzy techniques for adaptive nonlinear equalization,” *Signal Processing*, Vol.80, pp.985-1000, 2000.



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.



Sungdae Park
Received B.S. degree in Multimedia Eng, Dong-eui University, Korea, in 2002, and M.S. degree in Computer Eng., Dong-eui University, Korea. in 2005, and Ph.D. degree in Computer & Software Eng., Dong-eui University, Korea, in 2008. He is currently a fulltime lecture in the Dept. of Digital and Culture Eng., Dong-eui University. His research interests are digital design & digital nonlinear editing, neural networks, pattern recognition and image processing.