AN EFFICIENT LOW COMPLEXITY CLUSTERING-BASED MLSE EQUALIZER FOR FREQUENCY-SELECTIVE FADING CHANNELS

Yannis Kopsinis, Sergios Theodoridis, Eleftherios Kofidis

Dept. of Informatics and Telecommunications, University of Athens
Panepistimioupolis, Ilissia 15784, Athens, Greece
{kopsinis*, stheodor, kofidis}@di.uoa.gr

ABSTRACT

Recently, a sequence equalizer, the 1-D CBSE equalizer, which belongs to the family of Cluster-Based Sequence Equalizers and achieves the Maximum Likelihood solution to the equalization problem, was reported. The 1-D CBSE does not require the explicit estimation of the channel impulse response. Instead, it utilizes the estimates of the cluster centers formed by the received observations. In this paper, a novel cluster tracking scheme is presented which allows the application of the 1-D CBSE in time-varying transmission environments. Although the proposed equalizer exhibits similar performance with that of the classic MLSE-LMS equalizer, the overall required computational load is dramatically reduced. This is achieved because the new method provides the means for an efficient exploitation of the symmetries underlying the signaling scheme.

1. INTRODUCTION

Two of the major problems encountered in many contemporary mobile communication systems are those of InterSymbol Interference (ISI) and channel time variation. ISI, which is due to multipath, can be mitigated by the optimum sequence equalizer, known as maximum likelihood sequence equalizer (MLSE), which is effectively implemented via the Viterbi algorithm (VA) [1]. The MLSE equalizer utilizes the channel impulse response (CIR), which is estimated during the start up period, prior to the processing of each data block, based on an a priori known training sequence. However, under time varying conditions it is possible for the CIR to change significantly during the transmission of a data block, leading to serious performance degradation. For this reason, the CIR has to be continuously re-estimated utilizing the decisions obtained by the VA in order to track the channel variations.

Unfortunately, the VA provides reliable decisions after the reception of, approximately, $D = 5L$ data symbols, where L is the CIR length [1]. This decision delay is inherent in the VA and leads to poor performance since the channel taps used in the VA at the most recent time, k, are estimated D time instances earlier, i.e., $k-D$, and in the mean time the channel has changed. Many methods have been proposed in order to address this problem associated with the decision delay. These techniques have been evolved around the following main directions:

A) The channel tracking is achieved by utilizing tentative decisions obtained with a small fixed delay, d, usually as long as L. The length of this fixed delay, d, has to be selected carefully, since the adoption of a very small fixed delay, which leads to desirable short delayed channel estimates, implies erroneous tentative decisions [2]. B) The fixed delay, d, is chosen long enough, so as to avoid a large number of erroneous tentative decisions and the VA, at time k, is supplied with channel taps which are the result of an appropriately defined channel predictor [3]. C) A Per Survivor Processing (PSP) philosophy is adopted which results in no delayed channel estimates by using 4^{L-1} different channel estimators[1], where each one of them is associated with a specific state of the VA. [4].

The main drawback of all of the above MLSE equalizers is their high computational requirements, which may limit their practical use. In the current paper, a novel MLSE equalizer is presented that circumvents the problem of explicit CIR parametric modeling, leading to substantial computational savings in all the processing stages of the VA, including the start up and tracking phases. The proposed equalizer belongs to the family of Cluster-Based Sequence Equalizers (CBSE) [5], [6],[7],[8], and utilizes the clusters formed by the received observations. Initialization is achieved via an effective cluster center estimation technique [7], that exploits the structural symmetries underlying the generation mechanism of the clusters of the received samples. Furthermore, a novel cluster tracking scheme equivalent, but less complex, to the LMS algorithm is introduced.

2. DESCRIPTION OF THE COMMUNICATION SYSTEM

The equivalent baseband communication system model is illustrated in Fig. 1. x_k is the kth transmitted symbol, which takes values from the dataset $S = \{1+j, 1-j, -1+j, -1-j\}$, n_k is the white complex-valued additive noise and y_k denotes the kth received observation. The transmitted symbols have been assumed to be independent and identically distributed (i.i.d.). The received signal sampled at $t = kT$, with T being the transmission period of the symbols, is given by

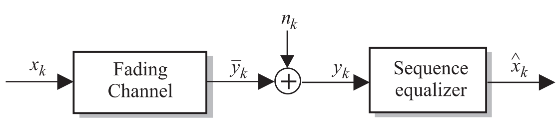

Figure 1: Communication system model.

$$y_k = \sum_{i=0}^{L-1} x_{k-i} h_{k,i}^* + n_k = \mathbf{h}_k^H \mathbf{x}_k + n_k \equiv \bar{y}_k + n_k \tag{1}$$

where $h_{k,i}^*$ are the complex - conjugated channel taps at time instance kT and $\bar{y}_k$ indicates the noiseless observation associated with the transmitted sequence of symbols $x_k, x_{k-1}, \cdots, x_{k-L+1}$. Thus, $\mathbf{h}_k = [h_{k,0},\ h_{k,1},\ \cdots, h_{k,L-1}]^T$ is the vector of the L complex taps of the CIR at time instance k and $\mathbf{x}_k = [x_k,\ x_{k-1},\ \cdots, x_{k-L+1}]^T$ is the vector of L successively transmitted symbols. The superscripts T, H, denote transposition and Hermitian transposition, respectively and the variance of the noise is σ^2 and the SNR is determined by $SNR = \left(\lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} E[|\bar{y}^i|^2]\right)/\sigma^2$

3. THE ONE-DIMENSIONAL CBSE

The MLSE equalizer is efficiently implemented via the Viterbi algorithm, which estimates the transmitted symbol sequence based on

*Currently with the School of Engineering and Electronics, University of Edinburgh, UK (e-mail: y.kopsinis@ed.ac.uk).

[1]The number 4 accounts for the four different symbols of the QPSK signaling scheme.

metric computations of the form $D_{k,\mathbf{x}} = |y_k - \mathbf{h}_k^H \mathbf{x}|^2$, with $\mathbf{x}$ ranging over the set of the 4^L possible L-tuples of input symbols. In fact, the CIR estimate is needed only for the computation of the 4^L convolutions, $\mathbf{h}_k^H \mathbf{x}$. However, eq. (1) suggests that what one really needs, instead, are the noiseless observations $\bar{y}_{k,\mathbf{x}} = \mathbf{h}_k^H \mathbf{x}$ and the computation of distance metrics of the simplified form

$$D_{k,\mathbf{x}} = |y_k - \bar{y}_{k,\mathbf{x}}|^2 \quad (2)$$

The possible values that $\bar{y}_{k,\mathbf{x}}$ can take are simply the points (centers), in the complex plane, around which the received samples y_k are clustered, due to the noise.

Equalizers that belong to the novel family of Clustering-Based Sequence Equalizers (CBSE), avoid the explicit estimation of the CIR, as an intermediate stage for the $\mathbf{h}_k^H \mathbf{x}$ computation for all possible L-tuples $\mathbf{x}$, by estimating $\bar{y}_{k,\mathbf{x}} = \mathbf{h}_k^H \mathbf{x}$ directly using any supervised clustering technique [9], e.g., a simple averaging.

Fig. 2 shows the received observations for a 2-tap channel in the presence of white Gaussian noise, with SNR=30 dB. The notation $\bar{y}^i_{[x_{L-1},\, x_{L-2}, \ldots,\, x_0]}$ denotes the ith cluster center, which is associated with a specific transmitted L-tuple $\mathbf{x} = [x_{L-1},\ x_{L-2}, \ldots,\ x_0]$. The sequence equalizer which utilizes the distance metric (2) will be referred to as the One-Dimensional CBSE (1-D CBSE). The 1-D CBSE is equivalent to the MLSE with respect to performance but it is less complex computationally, due to the fact that the computation of the convolutions is not required. The 1-D CBSE is a reduced

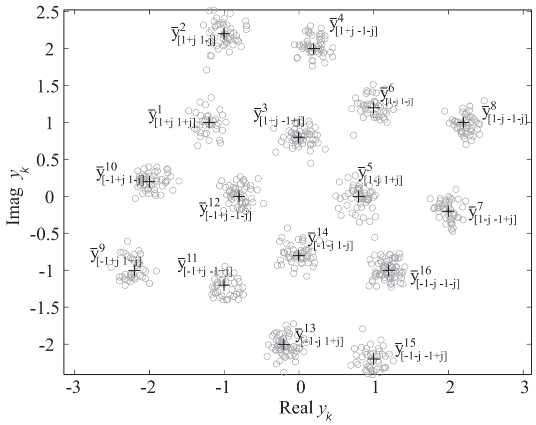

Figure 2: Plot in the complex plane of the clusters formed by the observations at the output of the two-tap channel $H(z) = (0.5 - j) + (-0.6 - 0.1j)z^{-1}$. Crosses denote cluster centers.

complexity extension of the previously proposed CBSE equalizers [5], [6].

When the channel varies with time, a center tracking scheme has to be included in order to allow the tracking of the cluster centers while they move away from their initial positions.

4. CLUSTER CENTER TRACKING TECHNIQUE

The major drawback of the CBSE equalizers, as well as of the symbol by symbol equalizers which also require cluster center estimation, e.g., [10], [11], is the need for a relatively long training sequence, so that all the clusters can be represented with sufficient observations in order to be able to estimate their centers accurately. Recently, a new method for the cluster center estimation has been presented [7], which does not require the direct information from all the clusters. In contrast, the new center estimation method exploiting the symmetries underlying the structure of the cluster centers in the complex plane, succeeds in estimating all the cluster centers, based on the direct estimates of only L properly selected centers, speeding up the training period dramatically. In the sequel, we briefly present the specific structure which is formed by the cluster centers. An extensive discussion of the center estimation method can be found in [7].

Let us assume a general L-taps channel with impulse response vector $\mathbf{h} = [h_0, h_1, \ldots, h_m, \ldots, h_{L-1}]^T$. The channel is considered to be approximately constant during the transmission of a short training sequence. We define as the *contribution*, c_x^m, of the mth tap, h_m, to the generation of the cluster centers the quantity $c_x^m = x h_m^*$, which takes 4 different values depending on $x \in \{1+j,\ 1-j,\ -1+j,\ -1-j\}$. In other words, this is the contribution of the h_m tap to the convolution sum in (1).

The $4^L = 16$ centers which correspond to the 2-taps example of Fig. 3, are denoted by (•). We can observe that the centers are positioned in the complex plane in a specific way. All the centers form 4 similar squares whose size and angle of rotation are determined by the contribution c_x^1 of the second tap. These squares are centered on the corners of a fifth central square, drawn in dashed line, which is associated with the contribution c_x^0 of the first tap.

Eq. (1) can be rewritten as $\bar{y}_{[x_k, x_{k-1}, \ldots, x_{k-L+1}]} = \sum_{m=0}^{L-1} c^m_{x_{k-m}}$ where $\bar{y}_{[x_k, x_{k-1}, \ldots, x_{k-L+1}]}$ is the cluster center associated with the transmitted $L-$tuple $[x_k, x_{k-1}, \ldots, x_{k-L+1}]$. We observe that the estimates of the L tap contributions c_x^m, $m = 0, \ldots, L-1$, is what one needs in order to compute the 4^L cluster centers[2]. In the tracking mode,

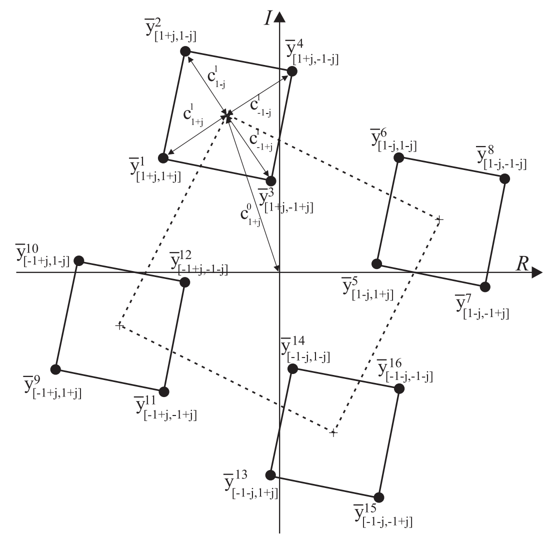

Figure 3: Cluster center constellation of the 2-taps channel $H(z) = (0.5 - j) + (-0.6 - 0.1j)z^{-1}$.

the MLSE equalizers exploit the delayed decisions, provided by the VA, $\hat{\mathbf{x}}_{k-d} = [\hat{x}_{k-d}, \hat{x}_{k-d-1}, \ldots, \hat{x}_{k-d-L+1}]^T$, to adapt the channel estimates. Similarly, the novel tracking scheme utilizes the same decisions so as to update the center estimates in the complex plane. Previously proposed center tracking methods, e.g., [10], update only one center per received sample. This results in very poor performance, because many centers remain constant for a long period of time even if, in the meantime, the transmission environment has significantly changed. In contrast to these methods, *the new technique succeeds in adapting all the centers simultaneously, each time a received symbol is detected*.

Even though the cluster centers are moving in the complex plane, their general structure of the squares is maintained. In order to track all the cluster centers, what is needed is to rotate and expand the associated squares toward the direction which is determined by

[2]It is easy to realize that only one value of a specific tap contribution, say c^m_{1+j}, need to be computed, and the rest can be obtained by simple $\frac{\pi}{2}$ rotations in the complex plane [7], e.g., $c^m_{1-j} = -jc^m_{1+j}$, $c^m_{-1-j} = -c^m_{1+j}$, e.t.c.

each detected symbol. Let us take a two taps example under noiseless transmission, for illustration purposes. Fig. 4 shows the center structure as it has been formed at time instance k. If the received sample y_{k-d+1}, which is denoted by $(\times)$, has been detected by the Viterbi algorithm as, e.g., $\mathbf{x} = [1+j,\ -1-j]$, then, after the adaptation, due to the absence of noise, the center $\bar{y}_{[1+j,\ -1-j]}$ should be moved to coincide with the point $(\times)$. In order to succeed in this, we can adopt a two-step procedure.

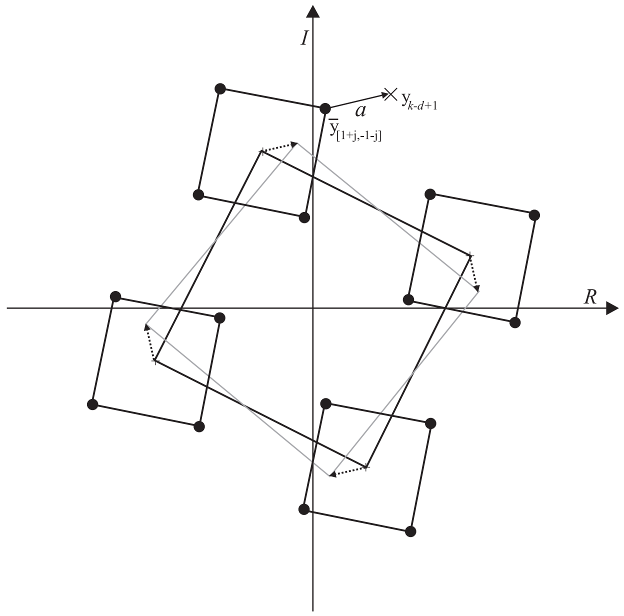

Figure 4: Adaptation procedure.

Firstly, the central square is expanded in the direction of the four small dotted arrows as it is shown in Fig. 4. In the sequel, the four external squares are "re-centered" on the corners of the adapted central square. According to this strategy, the center $\bar{y}_{[1+j,\ -1-j]}$ is moved towards point $(\times)$, covering half the distance[3] and the specific structure of the squares is also preserved. The result is shown in Fig. 5. Secondly, in a similar way, the four external squares are expanded, as it is shown by the dotted arrows in Fig. 5. The possitions of the centers after the adaptation procedure are shown in Fig. 5 by triangles $(\triangle)$.

Taking into account that both, the size and the angle of rotation of the squares, are determined by the corresponding tap contributions, the above procedure is efficiently realized as follows:

1. Compute the center $\hat{y}_{\mathbf{x}}$ based on the estimates $\hat{c}^i_{k-d}$, $0 \leq i \leq L-1$, of the tap contributions at time instance $k-d$, where $\mathbf{x} = [x_0, \ldots, x_i, \ldots, x_{L-1}]$ is provided by the tentative decisions of the Viterbi algorithm and corresponds to the received symbol y_{k-d+1}
$$\hat{y}_{\mathbf{x}} = \sum_{i=0}^{L-1} \hat{c}^i_{k-d,x_i}.$$
2. Compute the error between the received sample y_{k-d+1} and the corresponding center estimate
$$a = y_{k-d+1} - \hat{y}_{k-d,\mathbf{x}}.$$
3. Adapt the tap contribution estimates
$$\hat{c}^i_{k-d+1,x_i} = \hat{c}^i_{k-d,x_i} + \lambda a,\ i = 1,\ 2, \ldots,\ L-1.$$

[3] Basically, this assumes that the corresponding channel variation affects equally all the tap contributions.

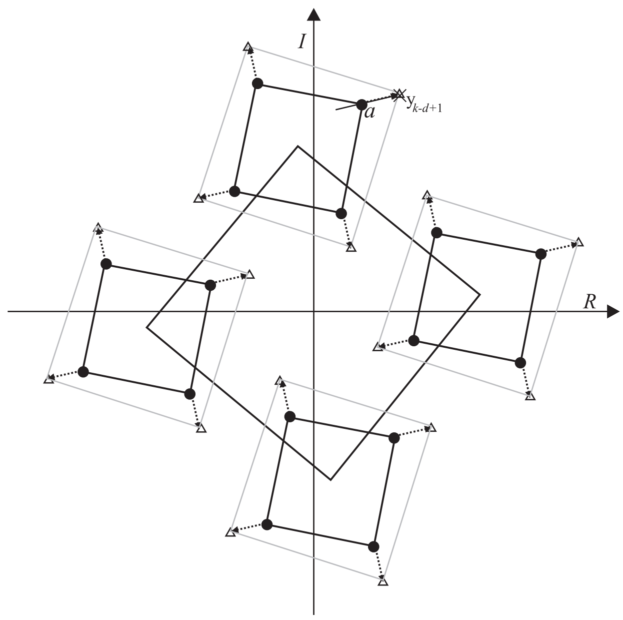

Figure 5: Adaptation procedure.

4. Compute the other values of the tap-contributions by $\frac{\pi}{2}$ rotations.

Actually, the proposed cluster tracking technique is equivalent to the LMS algorithm. Indeed, the last equation can be rewritten as $h^*_{k-d+1,i}x_i = h^*_{k-d,i}x_i + \lambda a$ or equivalently $h_{k-d+1,i} = h_{k-d,i} + \frac{\lambda}{x_i^*}a^*$. Taking into account that $\frac{1}{x_i^*} = \frac{1}{\|x_i\|^2}x_i$ and that $\|x_i\|^2 = 2$ for all the QPSK symbols, it turns out that $h_{k-d+1,i} = h_{k-d,i} + \frac{\lambda}{2}x_i a^*$, which defines the adaptation of the ith tap of the channel impulse response using the LMS algorithm with step parameter μ equal to $\mu = \lambda/2$.

5. PERFORMANCE EVALUATION

In the performance examples that follow, the time varying channel has been simulated based on the Smith's Rayleigh fading channel technique [12].

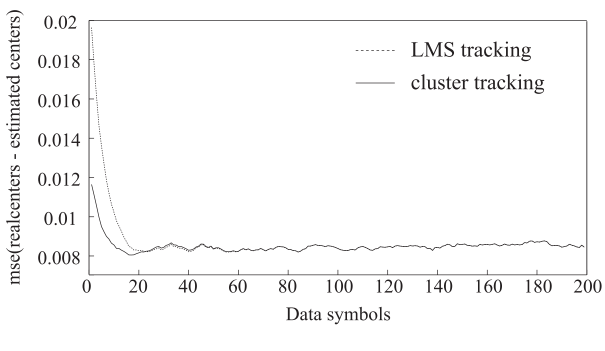

Figure 6: Tracking performance.

Fig. 6 shows the tracking performance of the new cluster tracking scheme together with that of the LMS algorithm. The curves are the result of the ensemble averaging of 1500 independent runs, where the adopted 3-taps fading channel corresponds to vehicle speed equal to 120 km/h. The symbol rate and carrier frequency is 200 Kbauds and 900 MHz, respectively and the RMS of the real and imaginary component of each path was set equal to 1. Both algorithms have been initialized utilizing 15 training symbols and the

Task \ Method		MUL / DIV		ADD / SUB		$(\cdot)^2$	
		MLSE-LMS	1-D CBSE	MLSE-LMS	1-D CBSE	MLSE-LMS	1-D CBSE
Initialization		$N_{tr}(8L+2)$	$2(2L+1)$	$N_{tr}8L$	$2(N_{tr}+L)+1$	—	—
Tracking	Tentative decisions	$N(8L+2)$	$2N$	$N8L$	$N4L$	—	—
Tracking	PSP	$N(8L+2)4^{L-1}$	$2N4^{L-1}$	$N8L4^{L-1}$	$N4L4^{L-1}$	—	—
VA	Tentative decisions or PSP	$N4L4^{L}$	0	$N(4L+1)4^{L}$	$N(2L+1)4^{L}$	$N2\cdot 4^{L}$	$N2\cdot 4^{L}$

Table 1: The computational complexities in terms of real operations for a data block consisted of N_{tr} training symbols and N data symbols.

adaptation procedure has been realized using the correct information symbols. The step parameters λ and μ were set equal to 0.08 and 0.04, respectively. The steady state performance of both algorithms, as it was expected (see Section 4), is the same. However, the cluster tracking technique exhibits better initialization behavior due to the enhanced convergence performance of the CE scheme [7]. Fig. 7 shows the performance curves of the 1-D CBSE equal-

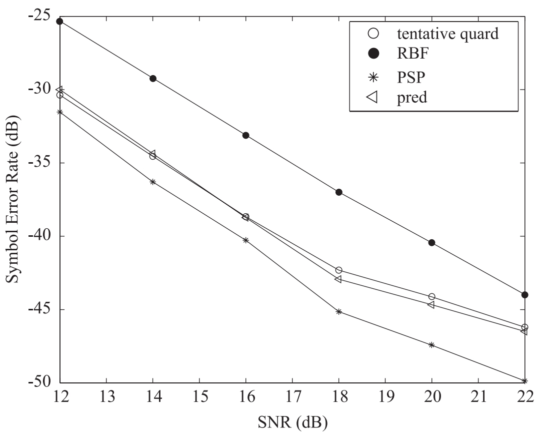

Figure 7: SER performance.

izer, when tracking is performed using 5 symbol delayed tentative decisions without predictor[4] (○), with predictor(◁) or via the PSP method (∗). Moreover, the performance curve of the finite memory decision feedback Bayesian equalizer (●) [11] is also shown for the purposes of the comparison. The performance of the MLSE equalizer is not shown, since it exhibits slightly worse performance than the 1-D CBSE due to the better start up provided by the latter algorithm. The fading channel parameters are the same with those mentioned in the tracking example of Fig. 6. Moreover, the transmission is realized in data blocks comprising 200 information and 15 training symbols which have been positioned in the beginning of each data block. Finally, the fading channel was left to evolve for 215 sec or equivalently for a time period of 43 million transmitted symbols.

With respect to complexity, as it is shown in table 1, the 1-D CBSE is dramatically less complex than the MLSE-LMS since it does not compute any convolutions in any one of the processing stages, included that of tracking and distance metric computation. For example, in the case of a 5-taps channel, $N = 200$ and $N_{tr} = 30$ symbols and channel tracking using tentative decisions, the MLSE-LMS algorithm needs more than 4.000.000 multiplications for the processing of a data block, in contrast to the CBSE which needs less than 500. The complexity of the finite memory Bayesian equalizer is not indicated since, taking into account the 4^L exponentials, which have to be computed per received sample, is much more complex than the 1-D CBSE [13].

[4]see Section 1

6. CONCLUSION

In this paper a novel technique for the design of MLSE equalizers for time-varying transmission environments was proposed. A novel cluster tracking scheme, which is equivalent to but less complex than the LMS tracking algorithm, has been incorporated in the structure of the 1-D CBSE equalizer. The simulation results show an enhanced performance of the new method compared to both the MLSE-LMS algorithm and the finite memory Bayesian-DFE equalizer. More importantly, this enhanced performance is achieved at a fraction of computational complexity.

REFERENCES

[1] J. G. Proakis, *Digital communications*, 3rd ed. McGraw-Hill International Editions, 1995.

[2] G. Castellini, F. Conti, E. D. Re, and L. Pierucci, “A continuously adaptive mlse receiver for mobile communications: Algorithm and performance,” *IEEE Trans. Commun.*, vol. 45, pp. 80–89, Jan. 1997.

[3] M. Chiu and C. Chao, “Analysis of lms-adaptive mlse equalization on multipath fading channels,” *IEEE Trans. Commun.*, vol. 44, pp. 1684–1692, Dec. 1996.

[4] R. Raheli, A. Polydoros, and C.-K. Tzou, “Per-survivor processing: A general approach to mlse in uncertain environments,” *IEEE Trans. Commun.*, vol. 43, pp. 354–364, Feb. 1995.

[5] S. Theodoridis, C. Cowan, C. Callender, and C. See, “Schemes for equalization of communications channels with nonlinear impairments,” *IEE Proc. Commun*, vol. 142, pp. 165–171, June 1995.

[6] K. Georgoulakis and S. Theodoridis, “Efficient clustering techniques for channel equalization in hostile environments,” *Signal Processing*, vol. 58, pp. 153–164, 1997.

[7] Y. Kopsinis and S. Theodoridis, “An efficient low-complexity technique for mlse equalizers for linear and nonlinear channels,” *IEEE Trans. Signal Processing*, vol. 51, pp. 3236–3248, Dec. 2003.

[8] ——, “A novel cluster based mlse equalizer for m-pam signaling schemes,” *Signal Processing*, vol. 83, pp. 1905–1918, 2003.

[9] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 2nd ed. Academic Press, 2003.

[10] 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, July 1993.

[11] S. Chen, S. McLaughlin, B. Mulgrew, and P. M. Grant, “Adaptive bayesian decision feedback equalizer for dispersive mobile radio channels,” *IEEE Trans. Commun.*, May 1995.

[12] T. S. Rappaport, *Wireless Communications*. Prentice Hall, 1996.

[13] Y. Kopsinis and S. Theodoridis, “Reduced-complexity clustering techniques for nonlinear channel equalization,” in *Proc. WCC2000(ICSP)*, Beijing, China, 2000.