

RESEARCH

Open Access

Adaptive reconfigurable V-BLAST type equalizer for cognitive MIMO-OFDM radios

Mehmet Tahir Ozden

Abstract

An adaptive channel shortening equalizer design for multiple input multiple output-orthogonal frequency division multiplexing (MIMO-OFDM) radio receivers is considered in this presentation. The proposed receiver has desirable features for cognitive and software defined radio implementations. It consists of two sections: MIMO decision feedback equalizer (MIMO-DFE) and adaptive multiple Viterbi detection. In MIMO-DFE section, a complete modified Gram-Schmidt orthogonalization of multichannel input data is accomplished using sequential processing multichannel Givens lattice stages, so that a Vertical Bell Laboratories Layered Space Time (V-BLAST) type MIMO-DFE is realized at the front-end section of the channel shortening equalizer. Matrix operations, a major bottleneck for receiver operations, are accordingly avoided, and only scalar operations are used. A highly modular and regular radio receiver architecture that has a suitable structure for digital signal processing (DSP) chip and field programable gate array (FPGA) implementations, which are important for software defined radio realizations, is achieved. The MIMO-DFE section of the proposed receiver can also be reconfigured for spectrum sensing and positioning functions, which are important tasks for cognitive radio applications. In connection with adaptive multiple Viterbi detection section, a systolic array implementation for each channel is performed so that a receiver architecture with high computational concurrency is attained. The total computational complexity is given in terms of equalizer and desired response filter lengths, alphabet size, and number of antennas. The performance of the proposed receiver is presented for two-channel case by means of mean squared error (MSE) and probability of error evaluations, which are conducted for time-invariant and time-variant channel conditions, orthogonal and nonorthogonal transmissions, and two different modulation schemes.

Keywords: V-BLAST; MIMO-DFE; MIMO-OFDM; Cognitive radio; Software defined radio

1 Introduction

The fundamental problem in the design of future wireless communication systems is to reliably and efficiently transmit and receive information signals over imperfect channels using substantially high data rates. One successful approach adopted in several wireless standards such as digital audio broadcasting (DAB), digital video broadcasting (DVB-T), local area networking (LAN), and metropolitan area networking (MAN) is orthogonal frequency division multiplexing (OFDM) in which the entire bandwidth is divided into several narrow subbands so that the frequency response over each individual subband is relatively flat, and each subband channel occupies only a small fraction of the original bandwidth. Nevertheless, OFDM-based wireless communication systems can

support peak rates of 54 Mbps at most, which is not enough to cover services the providers offer nowadays. Throughputs far beyond 54 Mbps can be provided when multiple input multiple output (MIMO) system approach is applied especially in a rich scattering environment [1,2]. Hence, the combination of OFDM and MIMO technologies [3,4] constitutes the basis for next-generation wireless communication systems such as IEEE 802.11n for wireless local area networks (WLAN), IEEE 802.16e for MAN, and evaluation of higher generational cellular systems.

The performance achieved through MIMO technology also entails a considerable increase in signal processing complexity in receiver, and there exists a major challenge in designing low-complexity receivers for multichannel wireless systems. A significant development for MIMO communications is the proposition of V-BLAST architecture [5], and two important areas of research on V-BLAST

Correspondence: mtozden@pirireis.edu.tr
Piri Reis University, Tuzla, 34940 Istanbul, Turkey

receivers are reduction of complexity, i.e., avoidance of matrix inversions, and extension to broadband implementation. Consequently, the recent research such as [6-8] focused on reducing the complexity of V-BLAST receiver architectures for frequency selective channels by developing efficient matrix inversion operations.

An important problem in realizing OFDM system designs, however, is the appendage of a cyclic prefix (CP) with a length at least equal to the channel length to each block of N IFFT coefficients, and this application may not be adequate in case the length of CP, ξ , is large relative to the data length, N , so that the channel throughput is reduced by a factor $N/(N + \xi)$. In addition, information about the channel length may not even be available in some cases. Accordingly, it is desired to design systems that guarantee a certain amount of throughput, i.e., $N/(N + \xi) \cong 1$, in all possible channel conditions. An elegant solution to this problem is to implement a time domain equalizer to shorten the channel memory and hence reduce the CP overhead [9].

A recent development in the design of next-generation wireless communication systems is the cognitive radio, built on a software radio, which functions as an intelligent system that is aware of its environment and uses the methodology of understanding-by-building to learn from the environment and adapt to statistical variations in the input stimuli in order to establish reliable communication by efficient utilization of the radio spectrum [10]. The concept of software radio on the other hand relies on the development of DSP technology, that is flexible, reconfigurable, and reprogrammable by software to adapt to an environment where there are multiple services, standards, and frequency bands [11]. Correspondingly, the infrastructure in a software radio system is generally required to use reconfigurable VLSI hardware components such as DSP chip sets [12], FPGAs [13], embedded processors [14], and even general purpose processors [15].

A typical cognitive radio cycle includes spectrum sensing, analysis, reasoning, and adaptation to new operating parameter steps [16]. The cognitive radio can detect the availability of a portion of frequency band through spectrum sensing and analysis steps [17]. During the reasoning step, it determines the optimum operating parameters, so that no harmful interference to other users of the spectrum is generated due to its transmission. In the adaptation step, the radio switches to transmission and reception mode using its reconfigurability and reprogrammability property and tunes its operating parameters according to its best response strategy. Another emerging requirement for cognitive radios is location and environment awareness that involves modeling the capabilities of human beings and bats for realization of advanced and autonomous location and environment awareness features [18]. Adaptive positioning, determining the

coordinates of a cognitive radio in space, is a step towards realization of accurate location awareness in cognitive radios. The author has recently proposed a spectrum estimation method in [19] and a range estimation method in [20] that are suitable for spectrum sensing and positioning functions of cognitive radios, respectively. In this paper, we focus on the reception mode of operation of cognitive MIMO-OFDM radios and propose a new minimum MSE channel shortening equalizer design, which consists of adaptive front-end MIMO-DFE and multiple Viterbi detection sections.

The optimum solution for the channel shortening equalization problem can be found using one of these constraints: (1) unit tap constraint and (2) unit energy constraint; and the performances under these two criteria were compared for single input single output (SISO) and MIMO channel shortening equalization in [21,22], respectively. Since the findings in these papers show that the unit energy constrained channel shortener equalization resulted in better performance, we have used unit energy constraint for the MIMO channel shortening optimization problem under consideration in this paper. Accordingly, the contributions of the paper can be stated as follows: (1) the proposed equalizer has a front-end MIMO-DFE as opposed to the MIMO feed forward equalizer (MIMO-FFE) in [22], (2) a modified version of sequential processing multichannel lattice stages (SPMLSs) [23] is utilized in the design of front-end MIMO-DFE and a complete modified Gram-Schmidt orthogonalization of multichannel input data, which avoids matrix inversions, enables scalar only operations and contributes to the flexibility, reconfigurability, and reprogrammability of the receiver, is attained, (3) the proposed equalizer can be viewed as a V-BLAST receiver for frequency selective channels, (4) spectrum sensing or range estimation can be accomplished at no cost by simply reconfiguring the front-end MIMO-DFE as multichannel spectral analysis or positioning filter as shown in [19,20], respectively, and (5) a detailed computational complexity and performance analysis is presented. The first contribution is important from the perspective of interference removal and, by means of that, error performance, whereas the second one is considered the key since matrix inversion is a major bottleneck in the design of embedded receiver architectures that increases computational complexity [24]. The third one is relevant since the receiver operations of a V-BLAST system can be considered as performing Gram-Schmidt orthogonalization [25], whereby inter-symbol interference (ISI) as well as inter-channel interference (ICI) effects are suppressed. The fourth contribution is crucial from the point of view of cognitive radio operation cycle, so that the filter structure of MIMO-DFE can be reused for spectrum sensing and range estimation functions of cognitive radio, and finally, a

comparative computational complexity and performance analysis of the proposed equalizer with respect to the other benchmark equalizers such as MIMO-DFE, MIMO-FFE, and multichannel Viterbi equalizer (VE), has been provided, which to the best of the author's knowledge, does not exist in the literature.

Various MIMO-DFEs for MIMO ISI channels have been proposed in the literature [26–28] after the introduction of the finite length MIMO-DFE in [29], and it has been delineated by Ginis and Cioffi in [30] that DFE is the basic principle behind the BLAST detection algorithm. Very recently, a QR decomposition-based MIMO-DFE has been presented by Wang et al. in [31]. In QR decomposition approaches, the Q matrix is implicitly formed and then used to compute the R matrix; whereas in the Gram-Schmidt approach, the inverse of the R is implicitly formed and then used to compute the Q matrix. As a consequence of this fact, Regalia and Bellanger showed in [32] that there exists a duality between QR and lattice methods and the possibility of combining elements of both approaches to obtain new hybrid algorithms. With respect to developing these hybrid algorithms, Ling showed in [33] that an orthogonal Givens rotation-based algorithm algebraically coincides with the recursive modified Gram-Schmidt-based lattice algorithm in [34]. In accordance with this perspective, we modify the SPMLS using Givens rotation-based lattice algorithms of [33] on the structure of the SPMLS, so that a sequential processing multichannel Givens lattice stage (SPMGLS) is obtained. SPMLSs are known for their modular, order-recursive, and regular structure as we previously used them in the decision feedback equalization of nonlinear communication channels in [35]. Additionally, good numerical properties are incurred by the use of Givens rotation-based lattice algorithms.

Subsequent to Givens lattice realization of the front-end MIMO-DFE, we perform a systolic array implementation of multiple adaptive Viterbi detectors [36], thereby a highly concurrent receiver structure is obtained. A two-channel (2×2) problem is considered in this presentation due to the ease of explanation and space limitations in developing the method. However, it is considered straightforward to apply the method to any number of channels, i.e., to massive MIMO implementations for next-generation wireless systems [37], at the expense of increased complexity. Even though the complete orthogonality and thereby the suppression of ISI and ICI is accomplished in the minimum mean square error sense for any number of transmit and receive antennas, the performance achieved in terms of MSE as well as in the probability of error will depend on how much the channel is ill-conditioned.

The organization of this paper is as follows. In Section 2, the adaptive multichannel channel shortening equalization optimization problem is introduced. In Section 3, we describe the adaptive multiple Viterbi detection section of the proposed equalizer. The computational complexity computations are treated in Section 4. In Section 5, we present the experimental results, and finally, Section 6 is concerned with the discussion of results and conclusions.

(\bullet)^{*} represents the complex conjugate of (\bullet). (\bullet)^T, and (\bullet)^H stands for the transpose and the Hermitian transpose of (\bullet), respectively. The variables m , i , and n are global while all other variables are local. The variable m represents the stage number while i and n are the time indexes related to data and coefficients, respectively, till we equate them in Section 3 to have a single time index.

2 Optimization problem statement

We consider the discrete-time baseband equivalent 2×2 channel shortening equalization problem depicted in Figure 1, where the number of transmitters (M_T) and receivers (M_R) is assumed equal, so that the number of antennas (M) is $M = M_T = M_R = 2$. Note that the k th baseband channel in Figure 1 models the effects of serial-to-parallel (S/P) and parallel-to-serial (P/S) conversions, the addition and removal of CP, the IDFT and DFT operations, and the physical baseband channel itself as delineated in Figure 2. Accordingly, the input signal to the k th receiver can be expressed as the sum of transmitted signals corrupted by ISI, ICI, and the noise:

$$y_k(n) = \sum_{\ell=1}^M \sum_{j=0}^{\mu_{\ell,k}} x_{\ell}(j) h_{\ell,k}(n-j) + u_k(n), \quad k = 1, 2, \quad (1)$$

in which $h_{\ell,k}(n)$ is the impulse response of the channel between the ℓ th transmitter and the k th receiver, whereas $x_{\ell}(n)$ is the transmitted sequence, and $\mu_{\ell,k}$ is the memory of channel $h_{\ell,k}(n)$. Accordingly, $h_{\ell,k}(n)$ for $\ell \neq k$ constitute ICI, and the elements of $h_{\ell,k}(n)$ for $\ell = k$ and $\mu_{\ell,k} \neq 0$ amount to ISI. $u_k(n)$ denotes the k th channel noise.

In the adaptive two-channel channel shortening equalizer design problem, the objective is to find an exponentially windowed, least squares (LS) solution for the coefficients of the k th adaptive DFE and the corresponding k th adaptive desired impulse response (ADIR) filter that minimizes the k th cost function:

$$J_k(n) = \sum_{i=0}^n \beta^{n-i} |e_n^k(i)|^2 \quad (2)$$

at each time instant n , where $k = 1, 2$ and β is the exponential weighting factor. Herein, the error signal, $e_n^k(i)$, is given by:

$$e_n^k(i) = d_n^k(i) - \hat{d}_n^k(i). \quad (3)$$

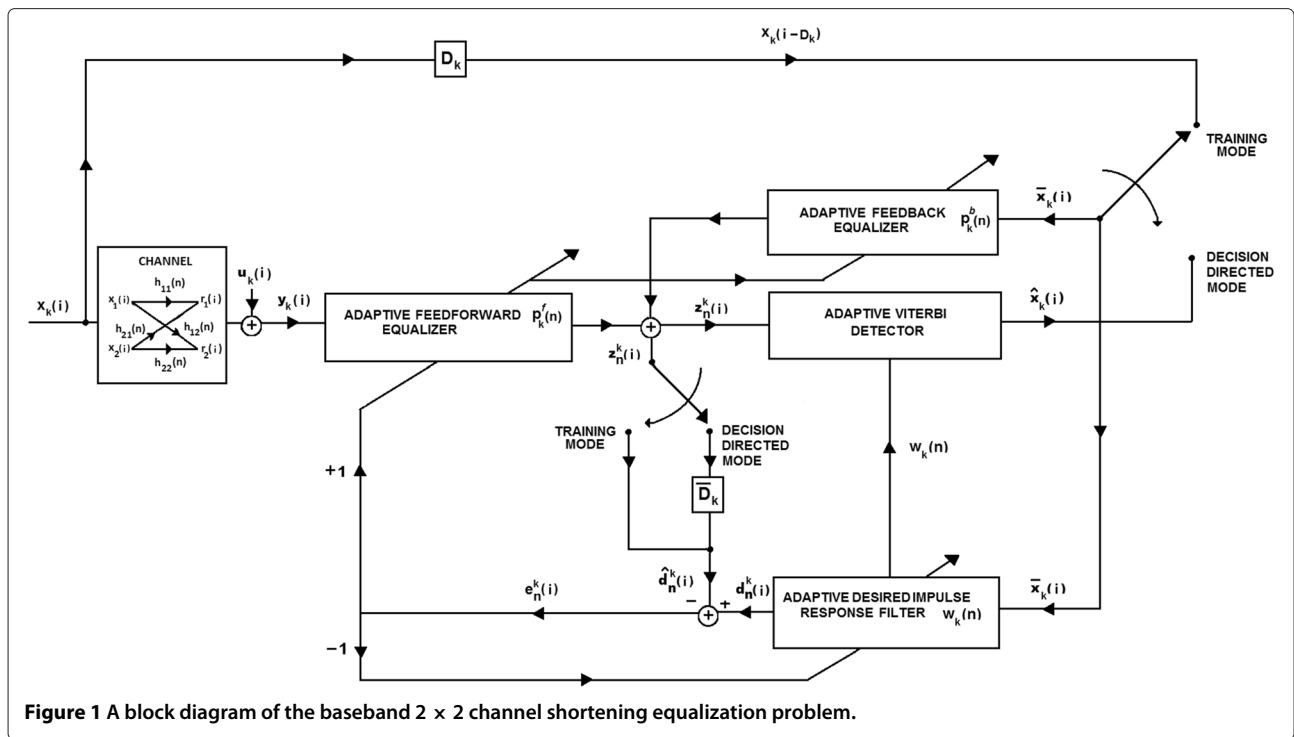


Figure 1 A block diagram of the baseband 2 x 2 channel shortening equalization problem.

The k th desired signal, $d_n^k(i)$, at the output of the k th ADIR is expressed as:

$$d_n^k(i) = \sum_{n=0}^{N_k^d-1} w_k(n) \bar{x}_k(i-n), \quad (4)$$

whereas the estimate of k th desired signal, $\hat{d}_n^k(i)$, in training mode of operation, is equal to the output signal of the k th DFE:

$$\hat{d}_n^k(i) = z_n^k(i) = \sum_{n=0}^{N_k^f-1} p_k^f(n) y_k(i-n) + \sum_{n=0}^{N_k^b-1} p_k^b(n) \bar{x}_k(i-n), \quad (5)$$

and it is identical to the delayed output signal of the k th DFE:

$$\hat{d}_n^k(i) = z_n^k(i - \bar{D}_k) = \sum_{n=0}^{N_k^f-1} p_k^f(n) y_k(i - \bar{D}_k - n) + \sum_{n=0}^{N_k^b-1} p_k^b(n) \bar{x}_k(i - \bar{D}_k - n) \quad (6)$$

in decision directed mode. Note that N_k^d in (4) and N_k^f and N_k^b in Equations (5) and (6) are the length of the k th ADIR filter, and the lengths of feed forward and feedback sections of the k th DFE, respectively, whereas $(N_k^d - 1)$, $(N_k^f - 1)$, and $(N_k^b - 1)$ are the corresponding filter and equalizer memories. Accordingly, \bar{D}_k is the delay experienced by the output signal of k th DFE, $z_n^k(i)$, during the Viterbi processing. The input signal $(\bar{x}_k(i))$ to the feedback section and to the k th ADIR filter in Equations (4),(5), and

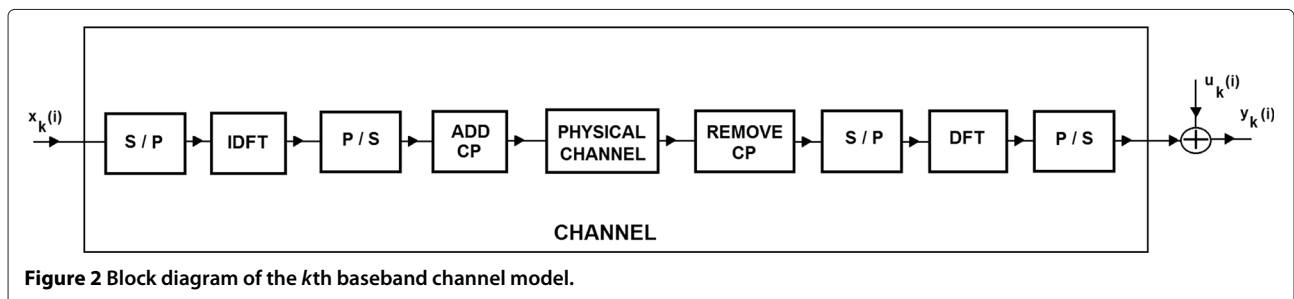


Figure 2 Block diagram of the k th baseband channel model.

(6) equals to the delayed k th channel input signal in training mode ($\bar{x}_k(i) = x_k(i - D_k)$), while it is equal to the k th detected signal in decision directed mode of operation of the receiver ($\bar{x}_k(i) = \hat{x}_k(i)$). Herein, D_k represents the delay experienced by the k th input signal, $x_k(i)$, through the corresponding channel during the training mode of operation.

Subsequently, we define the input vector to the k th DFE, $\mathbf{y}_k(i)$, at time instant i , as:

$$\mathbf{y}_k(i) = \left[y_k(i), y_k(i-1), \dots, y_k(i - N_k^f + 1), \bar{x}_k(i), \bar{x}_k(i-1), \dots, \bar{x}_k(i - N_k^b + 1) \right]^T, \quad (7)$$

and the corresponding coefficient vector, $\mathbf{p}_k(n)$, at time instant n , as:

$$\mathbf{p}_k(n) = \left[p_k^f(n), p_k^f(n-1), \dots, p_k^f(n - N_k^f + 1), p_k^b(n), p_k^b(n-1), \dots, p_k^b(n - N_k^b + 1) \right]^T. \quad (8)$$

The input vector to the k th ADIR filter at time instant i and its coefficient vector at the time instant n are also defined as:

$$\bar{\mathbf{x}}_k(i) = \left[\bar{x}_k(i), \bar{x}_k(i-1), \dots, \bar{x}_k(i - N_k^d + 1) \right]^T \quad (9)$$

and

$$\mathbf{w}_k(n) = \left[w_k(n), w_k(n-1), \dots, w_k(n - N_k^d + 1) \right]^T. \quad (10)$$

Note that we assume, without loss of generality, that $N_k^b \leq N_k^f$ for the k th DFE.

The main concern of the exponentially weighted LS problem under consideration is thus to find, at each time n , the k th optimal coefficient vectors, $\mathbf{p}_k(n)$ and $\mathbf{w}_k(n)$, that would minimize the cost function:

$$J_k(n) = \sum_{i=0}^n \beta^{n-i} \left| \mathbf{w}_k^H(n) \bar{\mathbf{x}}_k(i) - \mathbf{p}_k^H(n) \mathbf{y}_k(i) \right|^2, \quad (11)$$

which can be expressed in matrix form as follows:

$$J_k(n) = \mathbf{w}_k^H(n) \mathbf{R}_{\bar{x}_k \bar{x}_k}(n) \mathbf{w}_k(n) - \mathbf{w}_k^H(n) \mathbf{R}_{\bar{x}_k \mathbf{y}_k}(n) \mathbf{p}_k(n) - \mathbf{p}_k^H(n) \mathbf{R}_{\mathbf{y}_k \bar{x}_k}(n) \mathbf{w}_k(n) + \mathbf{p}_k^H(n) \mathbf{R}_{\mathbf{y}_k \mathbf{y}_k}(n) \mathbf{p}_k(n). \quad (12)$$

Herein, $\mathbf{R}_{\mathbf{y}_k \mathbf{y}_k}(n)$ is the $(N_k^f + N_k^b) \times (N_k^f + N_k^b)$ correlation matrix of $\mathbf{y}_k(i)$, which is given by:

$$\mathbf{R}_{\mathbf{y}_k \mathbf{y}_k}(n) = \sum_{i=0}^n \beta^{n-i} \mathbf{y}_k(i) \mathbf{y}_k^H(i), \quad (13)$$

$\mathbf{R}_{\bar{x}_k \mathbf{y}_k}(n)$ is the $N_k^d \times (N_k^f + N_k^b)$ cross-correlation matrix of $\bar{\mathbf{x}}_k(n)$ and $\mathbf{y}_k(n)$, which can be expressed as:

$$\mathbf{R}_{\bar{x}_k \mathbf{y}_k}(n) = \sum_{i=0}^n \beta^{n-i} \bar{\mathbf{x}}_k(i) \mathbf{y}_k^H(i), \quad (14)$$

and $\mathbf{R}_{\bar{x}_k \bar{x}_k}(n)$ is the $N_k^d \times N_k^d$ autocorrelation matrix of the k th ADIR filter input data vector $\bar{\mathbf{x}}_k(i)$, and is found by:

$$\mathbf{R}_{\bar{x}_k \bar{x}_k}(n) = \sum_{i=0}^n \beta^{n-i} \bar{\mathbf{x}}_k(i) \bar{\mathbf{x}}_k^H(i). \quad (15)$$

Note that $\mathbf{R}_{\bar{x}_k \mathbf{y}_k}(n) \triangleq \mathbf{R}_{\mathbf{y}_k \bar{x}_k}^H(n)$. Subsequently, the k th optimal coefficient vector for the equalizer is determined by differentiating $J_k(n)$ with respect to $\mathbf{p}_k(n)$, setting the derivative to zero, and solving for $\mathbf{p}_k(n)$:

$$\mathbf{p}_k^{\text{opt}}(n) = \mathbf{R}_{\mathbf{y}_k \mathbf{y}_k}^{-1}(n) \mathbf{R}_{\mathbf{y}_k \bar{x}_k}(n) \mathbf{w}_k(n). \quad (16)$$

In order to find a solution for the optimal coefficient vector of the k th ADIR filter, we substitute Equation (16) back into the cost function in Equation (12) and attain the following quadratic form in $\mathbf{w}_k(n)$:

$$J_k(n) = \mathbf{w}_k^H(n) \left[\mathbf{R}_{\bar{x}_k \bar{x}_k}(n) - \mathbf{R}_{\bar{x}_k \mathbf{y}_k}(n) \mathbf{R}_{\mathbf{y}_k \mathbf{y}_k}^{-1}(n) \mathbf{R}_{\mathbf{y}_k \bar{x}_k}(n) \right] \mathbf{w}_k(n). \quad (17)$$

Then, the expression enclosed by square brackets in Equation (17), a symmetric $N_k^d \times N_k^d$ matrix, is defined as:

$$\tilde{\mathbf{R}}_{\bar{x}_k \bar{x}_k}(n) = \mathbf{R}_{\bar{x}_k \bar{x}_k}(n) - \mathbf{R}_{\bar{x}_k \mathbf{y}_k}(n) \mathbf{R}_{\mathbf{y}_k \mathbf{y}_k}^{-1}(n) \mathbf{R}_{\mathbf{y}_k \bar{x}_k}(n), \quad (18)$$

so that the cost function in Equation (17) can be restated as:

$$J_k(n) = \mathbf{w}_k^H(n) \tilde{\mathbf{R}}_{\bar{x}_k \bar{x}_k}(n) \mathbf{w}_k(n). \quad (19)$$

In minimizing the expression in Equation (19), a unit energy constraint, $\mathbf{w}_k^H(n) \mathbf{w}_k(n) = 1$, is applied to the k th ADIR filter coefficients to avoid the trivial null equalizer solution [21], and the following Lagrangian expression is formed:

$$L(\mathbf{w}_k(n), \lambda) = \mathbf{w}_k^H(n) \tilde{\mathbf{R}}_{\bar{x}_k \bar{x}_k}(n) \mathbf{w}_k(n) + \lambda (\mathbf{w}_k^H(n) \mathbf{w}_k(n) - 1). \quad (20)$$

After taking the derivative of the expression in (20) and equating to zero, we get:

$$\tilde{\mathbf{R}}_{\bar{x}_k \bar{x}_k}(n) \mathbf{w}_k^{\text{opt}}(n) = \lambda \mathbf{w}_k^{\text{opt}}(n), \quad (21)$$

which shows that the optimal k th ADIR coefficient vector $\mathbf{w}_k^{\text{opt}}(n)$ and λ are the unit magnitude eigenvector and eigenvalue of the matrix $\tilde{\mathbf{R}}_{\bar{x}_k \bar{x}_k}(n)$, respectively. If the expression on the righthand side of Equation (21) is substituted for $\tilde{\mathbf{R}}_{\bar{x}_k \bar{x}_k}(n) \mathbf{w}_k(n)$ in Equation (19) and also

$\mathbf{w}_k^H(n)$ in Equation (19) is replaced with $\mathbf{w}_k^{Hopt}(n)$, then the minimum cost can be stated as follows:

$$J_k(n) = \lambda \mathbf{w}_k^{Hopt}(n) \mathbf{w}_k^{opt}(n) = \lambda_{\min} \quad (22)$$

which demonstrates that the cost function is minimized by choosing $\mathbf{w}_k^{opt}(n)$ to be equal to the eigenvector of the matrix $\tilde{\mathbf{R}}_{\tilde{x}_k \tilde{y}_k}$ and that the corresponding eigenvalue λ is the minimum eigenvalue of the matrix $\tilde{\mathbf{R}}_{\tilde{x}_k \tilde{y}_k}$ and is represented with λ_{\min} . Consequently, the optimal coefficient vectors for the k th equalizer and the k th ADIR filter, $\mathbf{p}_k^{opt}(n)$ and $\mathbf{w}_k^{opt}(n)$, are given by Equations (16) and (21), respectively.

2.1 V-BLAST type MIMO-DFE

We would like to use a V-BLAST type design approach for the front-end filter of the proposed equalizer, and we also require the design of a single, multichannel, and compact equalizer structure, so that two separate equalizers and direct evaluations as in (16) are avoided, and the same filter can be reconfigured as spectral analysis or positioning filter. These objectives can be accomplished by considering the equivalence of V-BLAST and modified Gram-Schmidt orthogonalization operations, and therefore by completely orthogonalizing the two-channel input data of DFE using SPMGLSs, which provide scalar only operations, good numerical properties as well as modularity, regularity, order recursiveness, and reconfigurability to the solution of equalization problem under consideration. Hereupon, we present the modifications we propose to make in SPMLSs so as to obtain SPMGLSs and then the design of front-end multichannel DFE using SPMGLSs.

2.2 SPMGLS

A SPMGLS has a block structure as shown in Figure 3, and the input signal vectors to a SPMGLS are defined as follows: the input forward prediction error vector:

$$\mathbf{f}_{\ell-1}(i) = [f_{\ell-1}^0(i), f_{\ell-1}^1(i), \dots, f_{\ell-1}^{p-1}(i), f_{\ell-1}^p(i)]^T, \quad (23)$$

the backward prediction error vector:

$$\mathbf{b}_{\ell-1}(i) = [b_{\ell-1}^0(i), b_{\ell-1}^1(i), \dots, b_{\ell-1}^{p-1}(i), b_{\ell-1}^p(i)]^T, \quad (24)$$

and the estimation error vector:

$$\mathbf{e}_{\ell-1}(i) = [e_{\ell-1}^0(i), e_{\ell-1}^1(i), \dots, e_{\ell-1}^{p-1}(i), e_{\ell-1}^p(i)]^T. \quad (25)$$

The elements of input forward and backward prediction error vectors in Equations (23) and (24) are orthogonalized by using self-orthogonalization processors (SOPs),

which are triangular-shaped processors in Figure 3. The outputs of SOPs are the orthogonalized forward prediction error vector:

$$\hat{\mathbf{f}}_{\ell-1}(i) = [\hat{f}_{\ell-1}^0(i), \hat{f}_{\ell-1}^1(i), \dots, \hat{f}_{\ell-1}^{p-1}(i), \hat{f}_{\ell-1}^p(i)]^T \quad (26)$$

and the orthogonalized backward prediction error vector:

$$\hat{\mathbf{b}}_{\ell-1}(i) = [\hat{b}_{\ell-1}^0(i), \hat{b}_{\ell-1}^1(i), \dots, \hat{b}_{\ell-1}^{p-1}(i), \hat{b}_{\ell-1}^p(i)]^T. \quad (27)$$

The elements of $\hat{\mathbf{f}}_{\ell-1}(i)$ are fed into a forward prediction reference-orthogonalization processor (ROP) in order to predict the elements of $\mathbf{b}_{\ell-1}(i-1)$ and to produce the stage output back prediction error vector $\mathbf{b}_{\ell}(i)$. The elements of $\hat{\mathbf{b}}_{\ell-1}(i)$ are fed into a ROP to perform p -channel joint process estimation and to produce the stage output estimation error vector $\mathbf{e}_{\ell}(i)$. Subsequently, the elements of $\hat{\mathbf{b}}_{\ell-1}(i)$ are delayed and are also fed into another ROP to obtain the stage output forward prediction error vector $\mathbf{f}_{\ell}(i)$.

There are two types of processing cells, single and double circular processors, in a SPMGLS as in the original SPMLS in [23]. Nevertheless, we change the processing equations implemented in these processing cells with the equations of the square root version of the Givens algorithm in [33]. The interconnections and signals propagating through these processing cells are shown in Figure 4. The processing cells symbolized with a double circle, which are also called boundary (angle computer) cells, perform the following equations:

$$d(i) = \sqrt{\beta d^2(i-1) + |x_{ref}(i)|^2} \quad (28)$$

$$c(i) = \frac{\sqrt{\beta} d(i-1)}{d(i)} \quad (29)$$

$$s(i) = x_{ref}^*(i)/d(i). \quad (30)$$

From Equations (28), (29), and (30), it can be shown that the parameters $c(i)$ and $s(i)$ satisfy the equation:

$$\begin{bmatrix} c(i) & s(i) \\ -s^*(i) & c(i) \end{bmatrix} \begin{bmatrix} \sqrt{\beta} d(i-1) \\ x_{ref}(i) \end{bmatrix} = \begin{bmatrix} d(i) \\ 0 \end{bmatrix}. \quad (31)$$

The connection between the input, $\gamma_{in}(i)$, and output, $\gamma_{out}(i)$, likelihood variables is defined as:

$$\gamma_{out}(i) = c(i) \gamma_{in}(i). \quad (32)$$

On the other hand, the processing cells symbolized with a single circle, which are called internal (rotator) cells, perform the following equations:

$$x_{out}(i) = c(i) x_{in}(i) - s^*(i) \sqrt{\beta} \kappa(i-1) \quad (33)$$

and

$$\kappa(i) = c(i) \sqrt{\beta} \kappa(i-1) + s(i) x_{in}(i). \quad (34)$$

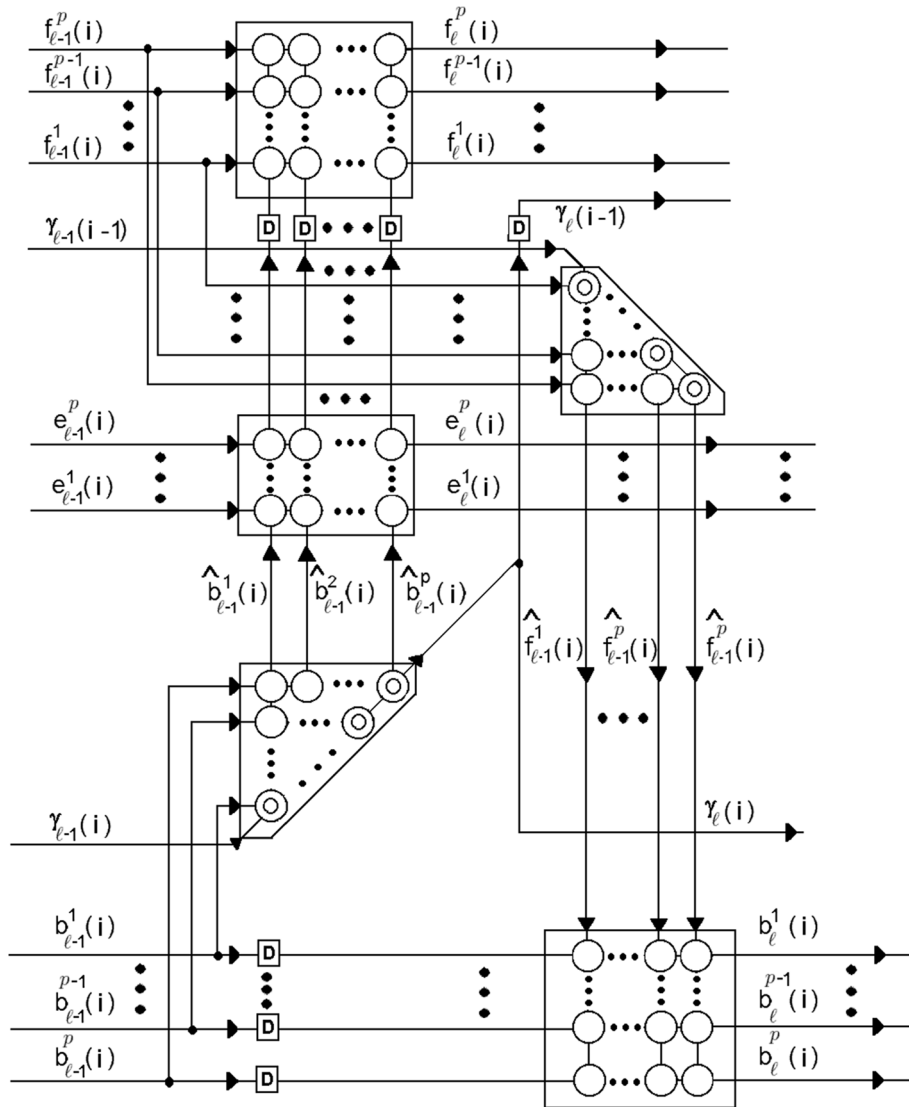


Figure 3 A block diagram of SPMGLS.

Accordingly, Equations (31), (33), and (34) can be combined into:

$$\begin{bmatrix} c(i) & s(i) \\ -s^*(i) & c(i) \end{bmatrix} \begin{bmatrix} \sqrt{\beta}d(i-1) & \sqrt{\beta}\kappa(i-1) \\ x_{\text{ref}}(i) & x_{\text{in}}(i) \end{bmatrix} = \begin{bmatrix} d(i) & \kappa(i) \\ 0 & x_{\text{out}}(i) \end{bmatrix}, \quad (35)$$

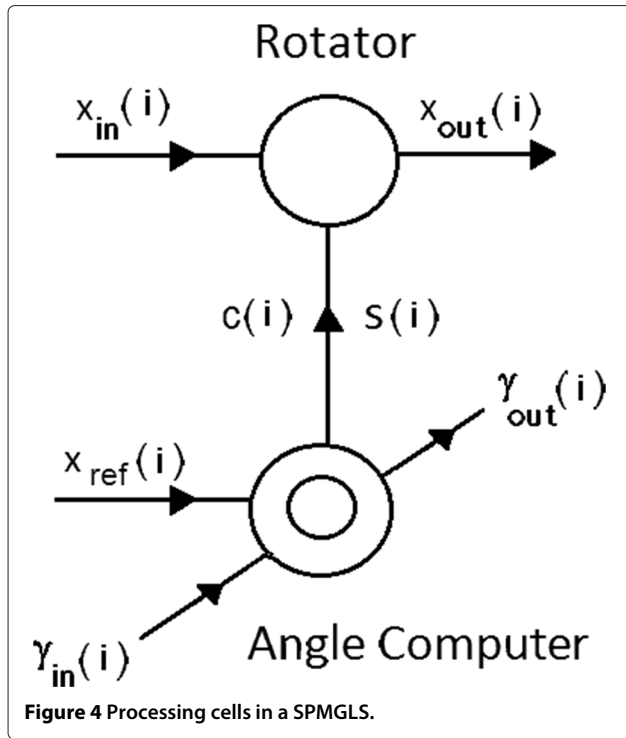
where

$$\mathbf{Q}(i) = \begin{bmatrix} c(i) & s(i) \\ -s^*(i) & c(i) \end{bmatrix} \quad (36)$$

with $|\mathbf{Q}(i)| = 1$, so that it performs Givens plane rotation in a complex plane, and thereby the stability of Givens algorithm is guaranteed.

2.3 Sequential givens lattice orthogonalization

The V-BLAST processing is made possible by utilizing SPMGLSs in the design of MIMO-DFE, so that the number of channels at different sections of the proposed multichannel lattice DFE is different due to the sequential processing nature of SPMGLSs. Therefore, we carry out the exponentially weighted LS optimization problem by taking into consideration each of these sections separately and assume that the proposed equalizer is comprised of three cascaded equalizers, which are two-channel, three-channel, and four-channel lattice sections; and we use a different index for each section while using m to indicate a stage in the whole equalizer. Henceforth, we focus on the case $N_1^f = N_2^f$ as it brings out the essential ideas



without unduly complicating the development and loss of generality.

In order to sequentially solve the exponentially weighted LS optimization problem under consideration, we first organize the elements of input signal vectors $\mathbf{y}_1(i) = [y_1(i), \dots, y_1(i - \ell + 1)]^T$ and $\mathbf{y}_2(i) = [y_2(i), \dots, y_2(i - \ell + 1)]^T$, according to the natural ordering of SPMGLSs as:

$$\bar{\mathbf{y}}_\ell(i) = \begin{bmatrix} y_1(i) \\ y_2(i) \\ y_1(i - 1) \\ y_2(i - 1) \\ \dots \\ y_1(i - \ell + 1) \\ y_2(i - \ell + 1) \end{bmatrix} \quad (37)$$

and input to two-channel stages for which the stage number m has a range of values given by $0 < m \leq (N_1^f - N_1^b)$.

Accordingly, we redefine Equations (13) and (14) using this new data vector as follows:

$$\Re_{\bar{\mathbf{y}}_\ell \bar{\mathbf{y}}_\ell}(n) = \sum_{i=0}^n \beta^{n-i} \bar{\mathbf{y}}_\ell(i) \bar{\mathbf{y}}_\ell^H(i), \quad (38)$$

and

$$\Re_{\bar{\mathbf{x}}_k \bar{\mathbf{y}}_\ell}(n) = \sum_{i=0}^n \beta^{n-i} \bar{\mathbf{x}}_k(i) \bar{\mathbf{y}}_\ell^H(i), \quad (39)$$

where $k = 1, 2$. The orthogonalization of input data using SPMGLSs corresponds to the transformation of (38) and (39) into:

$$\Xi_\ell(n) = \sum_{i=0}^n \beta^{n-i} \Omega_\ell(n) \bar{\mathbf{y}}_\ell(i) \bar{\mathbf{y}}_\ell^H(i) \Omega_\ell^H(n) \quad (40)$$

and

$$\Gamma_{\ell,k}(n) = \sum_{i=0}^n \beta^{n-i} \Omega_\ell(n) \bar{\mathbf{y}}_\ell(i) \bar{\mathbf{x}}_k^H(i) \quad (41)$$

respectively. Here, $\Omega_\ell(n)$ is the $2\ell \times 2\ell$ lower triangular transformation matrix and is realized stage-by-stage using 2×2 lower triangular transformation matrices:

$$\mathbf{L}_\ell(n) = \begin{bmatrix} 1 & 0 \\ \hat{\kappa}_\ell(n-1) & 1 \end{bmatrix} \quad (42)$$

whose diagonal elements are all equal to unity at time instant n , and $\hat{\kappa}_\ell(n)$ is the reflection coefficient computed at the single circular cell in the triangular shaped self-orthogonalization processor of the ℓ th two-channel SPMGLS. Then, the lattice joint process estimation coefficients are computed by means of:

$$\Theta_{\ell,k}(n) = \Xi_\ell^{-1}(n) \Gamma_{\ell,k}(n), \quad (43)$$

where $\Theta_{\ell,k}(n)$ represents the k th row of the $2 \times 2\ell$ lattice joint process estimation reflection coefficient matrix $\Theta_\ell(n)$ which is also sequentially computed stage-by-stage using 2×2 joint process estimation coefficient matrices:

$$\Delta_\ell(n) = \begin{bmatrix} \bar{\kappa}_{\ell,1,1}(n) & \bar{\kappa}_{\ell,1,2}(n) \\ \bar{\kappa}_{\ell,2,1}(n) & \bar{\kappa}_{\ell,2,2}(n) \end{bmatrix}, \quad (44)$$

in which $\bar{\kappa}_{\ell,k,j}(n)$ is the j th reflection coefficient related to the estimation of the k th desired signal, and it is computed at the (k, j) th single circular cell of the square shaped reference-orthogonalization processor related to joint process estimation at the ℓ th two-channel SPMGLS. Note that the matrix inversion operation in Equation (16) is transformed into a simple scalar inversion operation in (43) due to the diagonal nature of $\Xi_\ell(n)$.

After the processing of input signals by two-channel lattice stages, the first estimation error signal, $\bar{\mathbf{x}}_1(i) = e_1(i)$, which corresponds to the detected and fed back signal of the first channel, is incorporated at the $(N_1^f - N_1^b + 1)$ th stage as the third channel. Accordingly, we expand the optimization problem by organizing the elements of the input data vectors $\mathbf{y}_1(i) = [y_1(i), \dots, y_1(i - \alpha + 1)]^T$, $\mathbf{y}_2(i) =$

$[y_2(i), \dots, y_2(i - \alpha + 1)]^T$, and $\mathbf{x}_1(i) = [\bar{x}_1(i - 1), \dots, \bar{x}_1(i - \alpha + 1)]^T$ as follows:

$$\bar{\mathbf{y}}_\alpha(i) = \begin{bmatrix} y_1(i) \\ y_2(i) \\ \bar{x}_1(i - 1) \\ \text{---} \\ y_1(i - \alpha + 1) \\ y_2(i - \alpha + 1) \\ \bar{x}_1(i - \alpha + 1) \end{bmatrix}, \quad (45)$$

and input to the three-channel lattice section, where the stage number (m) takes values in the range given by $(N_1^f - N_1^b) < m \leq (N_2^f - N_2^b)$. Subsequently, we solve the optimization problem in (43) once again with the new input vector, in which case $\Omega_\alpha(n)$ and $\Theta_\alpha(n)$ are the $3\alpha \times 3\alpha$ lower triangular transformation and the $3 \times 3\alpha$ lattice joint process estimation coefficient matrices, respectively. $\Omega_\alpha(n)$ is computed sequentially by means of 3×3 lower triangular transformation matrices, $\mathbf{L}_\alpha(n)$ and $\Theta_\alpha(n)$, and is similarly realized stage-by-stage making use of 3×3 joint process estimation coefficient matrices, $\Delta_\alpha(n)$, at time instant n .

Finally, the optimization problem is expanded one more time with the inclusion of the second estimation error signal, $\bar{x}_2(i) = e_2(i)$, which is related to the detected and fed back signal of the second channel, and this time, the elements of input data vectors $\mathbf{y}_1(i) = [y_1(i), \dots, y_1(i - \vartheta + 1)]^T$, $\mathbf{y}_2(i) = [y_2(i), \dots, y_2(i - \vartheta + 1)]^T$, $\bar{\mathbf{x}}_1(i) = [\bar{x}_1(i - 1), \dots, \bar{x}_1(i - \vartheta + 1)]^T$, and $\bar{\mathbf{x}}_2(i) = [\bar{x}_2(i - 1), \dots, \bar{x}_2(i - \vartheta + 1)]^T$ are organized as:

$$\bar{\mathbf{y}}_\vartheta(i) = \begin{bmatrix} y_1(i) \\ y_2(i) \\ \bar{x}_1(i - 1) \\ \bar{x}_2(i - 1) \\ \text{---} \\ y_1(i - \vartheta + 1) \\ y_2(i - \vartheta + 1) \\ \bar{x}_1(i - \vartheta + 1) \\ \bar{x}_2(i - \vartheta + 1) \end{bmatrix}, \quad (46)$$

where the stage number (m) is in the range given by $(N_2^f - N_2^b) < m \leq N_2^f$ due to four-channel processing. Similar to two-channel and three-channel cases, we solve the optimization problem in (43) using the new data vector in Equation (46), in which case $\Omega_\vartheta(n)$ and $\Theta_\vartheta(n)$ are $4\vartheta \times 4\vartheta$ lower triangular transformation, and $4 \times 4\vartheta$ joint process estimation error coefficient matrices at the time instant n , respectively. Similar to previous cases, these matrices are computed stage-by-stage by the use of 4×4 lower triangular transformation matrices, $\mathbf{L}_\vartheta(n)$, and 4×4 joint process estimation error coefficient matrices, $\Delta_\vartheta(n)$, at time instant n , respectively.

2.4 Computation of error order updates

Due to the sequential nature of the proposed lattice structure, we carry out the multichannel error order update task by taking into consideration two-channel, three-channel, and four-channel sections separately, and therefore we assume that the filter is comprised of three cascaded filters as described in the previous subsection. The prediction and joint state estimation errors for the end of the observation interval $n = i$ at the output of the ℓ th order two-channel equalizer section, where $0 < m \leq (N_1^f - N_1^b)$, can be stated in terms of lattice parameters and the $(\ell - 1)$ th order prediction errors as follows:

$$\begin{bmatrix} f_\ell^1(n) \\ f_\ell^2(n) \end{bmatrix} = \begin{bmatrix} f_{\ell-1}^1(n) \\ f_{\ell-1}^2(n) \end{bmatrix} + \begin{bmatrix} \bar{\kappa}_{\ell,1,1}^{f*}(n-1) & \bar{\kappa}_{\ell,1,2}^{f*}(n-1) \\ \bar{\kappa}_{\ell,2,1}^{f*}(n-1) & \bar{\kappa}_{\ell,2,2}^{f*}(n-1) \end{bmatrix} \times \begin{bmatrix} 1 & 0 \\ \hat{\kappa}_\ell^{f*}(n-2) & 1 \end{bmatrix} \begin{bmatrix} b_{\ell-1}^1(n-1) \\ b_{\ell-1}^2(n-1) \end{bmatrix} \quad (47)$$

and

$$\begin{bmatrix} b_\ell^1(n) \\ b_\ell^2(n) \end{bmatrix} = \begin{bmatrix} b_{\ell-1}^1(n-1) \\ b_{\ell-1}^2(n-1) \end{bmatrix} + \begin{bmatrix} \bar{\kappa}_{\ell,1,1}^{b*}(n-1) & \bar{\kappa}_{\ell,1,2}^{b*}(n-1) \\ \bar{\kappa}_{\ell,2,1}^{b*}(n-1) & \bar{\kappa}_{\ell,2,2}^{b*}(n-1) \end{bmatrix} \times \begin{bmatrix} 1 & 0 \\ \hat{\kappa}_\ell^{b*}(n-1) & 1 \end{bmatrix} \begin{bmatrix} f_{\ell-1}^1(n) \\ f_{\ell-1}^2(n) \end{bmatrix} \quad (48)$$

where the lower triangular and square coefficient matrices are generated in triangular-shaped self-orthogonalization and square-shaped reference-orthogonalization processors in a two-channel SPMGLS as previously defined in Equations (42) and (44). The joint process estimation error updates are accordingly given as:

$$\begin{bmatrix} e_\ell^1(n) \\ e_\ell^2(n) \end{bmatrix} = \begin{bmatrix} e_{\ell-1}^1(n) \\ e_{\ell-1}^2(n) \end{bmatrix} + \begin{bmatrix} \bar{\kappa}_{\ell,1,1}^{e*}(n-1) & \bar{\kappa}_{\ell,1,2}^{e*}(n-1) \\ \bar{\kappa}_{\ell,2,1}^{e*}(n-1) & \bar{\kappa}_{\ell,2,2}^{e*}(n-1) \end{bmatrix} \times \begin{bmatrix} 1 & 0 \\ \hat{\kappa}_\ell^{e*}(n-1) & 1 \end{bmatrix} \begin{bmatrix} b_{\ell-1}^1(n) \\ b_{\ell-1}^2(n) \end{bmatrix}. \quad (49)$$

We then multiply the lower triangular and square coefficient matrices in Equations (47), (48), and (49), and make the following definitions:

$$\begin{aligned} \Gamma_\ell^f(n) &= \begin{bmatrix} \Gamma_{\ell,1,1}^f(n) & \Gamma_{\ell,1,2}^f(n) \\ \Gamma_{\ell,2,1}^f(n) & \Gamma_{\ell,2,2}^f(n) \end{bmatrix} \\ &= \begin{bmatrix} \bar{\kappa}_{\ell,1,1}^f(n) + \bar{\kappa}_{\ell,1,2}^f(n)\hat{\kappa}_\ell^f(n-1) & \bar{\kappa}_{\ell,1,2}^f(n) \\ \bar{\kappa}_{\ell,2,1}^f(n) + \bar{\kappa}_{\ell,2,2}^f(n)\hat{\kappa}_\ell^f(n-1) & \bar{\kappa}_{\ell,2,2}^f(n) \end{bmatrix}, \end{aligned} \quad (50)$$

$$\begin{aligned} \mathbf{\Gamma}_\ell^b(n) &= \begin{bmatrix} \Gamma_{\ell,1,1}^b(n) & \Gamma_{\ell,1,2}^b(n) \\ \Gamma_{\ell,2,1}^b(n) & \Gamma_{\ell,2,2}^b(n) \end{bmatrix} \\ &= \begin{bmatrix} \bar{\kappa}_{\ell,1,1}^b(n) + \bar{\kappa}_{\ell,1,2}^b(n)\hat{\kappa}_\ell^b(n) & \bar{\kappa}_{\ell,1,2}^b(n) \\ \bar{\kappa}_{\ell,2,1}^b(n) + \bar{\kappa}_{\ell,2,2}^b(n)\hat{\kappa}_\ell^b(n) & \bar{\kappa}_{\ell,2,2}^b(n) \end{bmatrix}, \end{aligned} \quad (51)$$

and

$$\begin{aligned} \mathbf{\Gamma}_\ell^e(n) &= \begin{bmatrix} \Gamma_{\ell,1,1}^e(n) & \Gamma_{\ell,1,2}^e(n) \\ \Gamma_{\ell,2,1}^e(n) & \Gamma_{\ell,2,2}^e(n) \end{bmatrix} \\ &= \begin{bmatrix} \bar{\kappa}_{\ell,1,1}^e(n) + \bar{\kappa}_{\ell,1,2}^e(n)\hat{\kappa}_\ell^e(n) & \bar{\kappa}_{\ell,1,2}^e(n) \\ \bar{\kappa}_{\ell,2,1}^e(n) + \bar{\kappa}_{\ell,2,2}^e(n)\hat{\kappa}_\ell^e(n) & \bar{\kappa}_{\ell,2,2}^e(n) \end{bmatrix}, \end{aligned} \quad (52)$$

in order to obtain compact versions of the Equations (47), (48), and (49) as follows:

$$\begin{bmatrix} f_\ell^1(n) \\ f_\ell^2(n) \end{bmatrix} = \begin{bmatrix} f_{\ell-1}^1(n) \\ f_{\ell-1}^2(n) \end{bmatrix} + \mathbf{\Gamma}_\ell^{f*}(n-1) \begin{bmatrix} b_{\ell-1}^1(n-1) \\ b_{\ell-1}^2(n-1) \end{bmatrix}, \quad (53)$$

$$\begin{bmatrix} b_\ell^1(n) \\ b_\ell^2(n) \end{bmatrix} = \begin{bmatrix} b_{\ell-1}^1(n-1) \\ b_{\ell-1}^2(n-1) \end{bmatrix} + \mathbf{\Gamma}_\ell^{b*}(n-1) \begin{bmatrix} f_{\ell-1}^1(n) \\ f_{\ell-1}^2(n) \end{bmatrix}, \quad (54)$$

and

$$\begin{bmatrix} e_\ell^1(n) \\ e_\ell^2(n) \end{bmatrix} = \begin{bmatrix} e_{\ell-1}^1(n) \\ e_{\ell-1}^2(n) \end{bmatrix} + \mathbf{\Gamma}_\ell^{e*}(n-1) \begin{bmatrix} b_{\ell-1}^1(n) \\ b_{\ell-1}^2(n) \end{bmatrix}. \quad (55)$$

The development of prediction and joint process estimation error order updates from $(\alpha - 1)$ th order to α th for the three-channel section, where the stage number (m) takes values in the range given by $(N_1^f - N_1^b) < m \leq (N_2^f - N_2^b)$, is carried out in a similar fashion to the two-channel section, and they can be expressed in compact form with the following equations:

$$\begin{bmatrix} f_\alpha^1(n) \\ f_\alpha^2(n) \\ f_\alpha^3(n) \end{bmatrix} = \begin{bmatrix} f_{\alpha-1}^1(n) \\ f_{\alpha-1}^2(n) \\ f_{\alpha-1}^3(n) \end{bmatrix} + \mathbf{\Gamma}_\alpha^{f*}(n-1) \begin{bmatrix} b_{\alpha-1}^1(n-1) \\ b_{\alpha-1}^2(n-1) \\ b_{\alpha-1}^3(n-1) \end{bmatrix}, \quad (56)$$

$$\begin{bmatrix} b_\alpha^1(n) \\ b_\alpha^2(n) \\ b_\alpha^3(n) \end{bmatrix} = \begin{bmatrix} b_{\alpha-1}^1(n-1) \\ b_{\alpha-1}^2(n-1) \\ b_{\alpha-1}^3(n-1) \end{bmatrix} + \mathbf{\Gamma}_\alpha^{b*}(n-1) \begin{bmatrix} f_{\alpha-1}^1(n) \\ f_{\alpha-1}^2(n) \\ f_{\alpha-1}^3(n) \end{bmatrix}, \quad (57)$$

and

$$\begin{bmatrix} e_\alpha^1(n) \\ e_\alpha^2(n) \\ e_\alpha^3(n) \end{bmatrix} = \begin{bmatrix} e_{\alpha-1}^1(n) \\ e_{\alpha-1}^2(n) \\ e_{\alpha-1}^3(n) \end{bmatrix} + \mathbf{\Gamma}_\alpha^{e*}(n-1) \begin{bmatrix} b_{\alpha-1}^1(n) \\ b_{\alpha-1}^2(n) \\ b_{\alpha-1}^3(n) \end{bmatrix}, \quad (58)$$

where

$$\begin{aligned} \mathbf{\Gamma}_\alpha^f(n) &= \begin{bmatrix} \bar{\kappa}_{\alpha,1,1}^f(n) & \bar{\kappa}_{\alpha,1,2}^f(n) & \bar{\kappa}_{\alpha,1,3}^f(n) \\ \bar{\kappa}_{\alpha,2,1}^f(n) & \bar{\kappa}_{\alpha,2,2}^f(n) & \bar{\kappa}_{\alpha,2,3}^f(n) \\ \bar{\kappa}_{\alpha,3,1}^f(n) & \bar{\kappa}_{\alpha,3,2}^f(n) & \bar{\kappa}_{\alpha,3,3}^f(n) \end{bmatrix} \\ &\times \begin{bmatrix} 1 & 0 & 0 \\ \hat{\kappa}_{\alpha,2,1}^f(n-1) & 1 & 0 \\ \hat{\kappa}_{\alpha,3,1}^f(n-1) & \hat{\kappa}_{\alpha,3,2}^f(n-1) & 1 \end{bmatrix}, \end{aligned}$$

$$\mathbf{\Gamma}_\alpha^b(n) = \begin{bmatrix} \bar{\kappa}_{\alpha,1,1}^b(n) & \bar{\kappa}_{\alpha,1,2}^b(n) & \bar{\kappa}_{\alpha,1,3}^b(n) \\ \bar{\kappa}_{\alpha,2,1}^b(n) & \bar{\kappa}_{\alpha,2,2}^b(n) & \bar{\kappa}_{\alpha,2,3}^b(n) \\ \bar{\kappa}_{\alpha,3,1}^b(n) & \bar{\kappa}_{\alpha,3,2}^b(n) & \bar{\kappa}_{\alpha,3,3}^b(n) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ \hat{\kappa}_{\alpha,2,1}^b(n) & 1 & 0 \\ \hat{\kappa}_{\alpha,3,1}^b(n) & \hat{\kappa}_{\alpha,3,2}^b(n) & 1 \end{bmatrix},$$

and

$$\mathbf{\Gamma}_\alpha^e(n) = \begin{bmatrix} \bar{\kappa}_{\alpha,1,1}^e(n) & \bar{\kappa}_{\alpha,1,2}^e(n) & \bar{\kappa}_{\alpha,1,3}^e(n) \\ \bar{\kappa}_{\alpha,2,1}^e(n) & \bar{\kappa}_{\alpha,2,2}^e(n) & \bar{\kappa}_{\alpha,2,3}^e(n) \\ \bar{\kappa}_{\alpha,3,1}^e(n) & \bar{\kappa}_{\alpha,3,2}^e(n) & \bar{\kappa}_{\alpha,3,3}^e(n) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ \hat{\kappa}_{\alpha,2,1}^e(n) & 1 & 0 \\ \hat{\kappa}_{\alpha,3,1}^e(n) & \hat{\kappa}_{\alpha,3,2}^e(n) & 1 \end{bmatrix}.$$

The prediction and joint process estimation error order update equations from $(\vartheta - 1)$ th order to ϑ th order for the four-channel section, where the stage number (m) is in the range given by $(N_2^f - N_2^b) < m \leq N_2^f$, can be derived by following a similar procedure to two- and three-channel sections, and 4×1 error order update matrices can be subsequently obtained with 4×4 lower triangular and square coefficient matrices.

2.5 Matrix visualization

In order to visualize the cascading and functioning of two-channel, three-channel, and four-channel sections as a single equalizer, we provide a matrix representation of sequential Givens lattice orthogonalization by considering $(N_1^f, N_1^b) = (13, 7)$ and $(N_2^f, N_2^b) = (13, 2)$ DFEs for the first and second channels, and also organizing the elements of input data vectors $\mathbf{y}_1(n) = [y_1(n), y_1(n-1), \dots, y_1(n-11), y_1(n-12)]^T$, $\mathbf{y}_2(n) = [y_2(n), y_2(n-1), \dots, y_2(n-11), y_2(n-12)]^T$, $\mathbf{x}_1(n) = [\bar{x}_1(n-1), \dots, \bar{x}_1(n-6)]^T$, and $\mathbf{x}_2(n) = [\bar{x}_2(n-1)]^T$ as columns of a matrix:

$$\begin{bmatrix} y_1(n) & y_1(n-1) & \dots & y_1(n-6) & y_1(n-7) & \dots & y_1(n-11) & y_1(n-12) \\ y_2(n) & y_2(n-1) & \dots & y_2(n-6) & y_2(n-7) & \dots & y_2(n-11) & y_2(n-12) \\ & & & \bar{x}_1(n-1) & \dots & \bar{x}_1(n-5) & \bar{x}_1(n-6) & \\ & & & & & & & \bar{x}_2(n-1) \end{bmatrix} \quad (59)$$

by taking into consideration different numbers of parameters in the feed forward and feedback channels and shifting properties of input data. This matrix helps us to visualize the orthogonalization process, and thus to draw a diagram of the four channel DFE structure under consideration as in Figure 5. Note that the elements of the first and second rows are related to the input signals of the

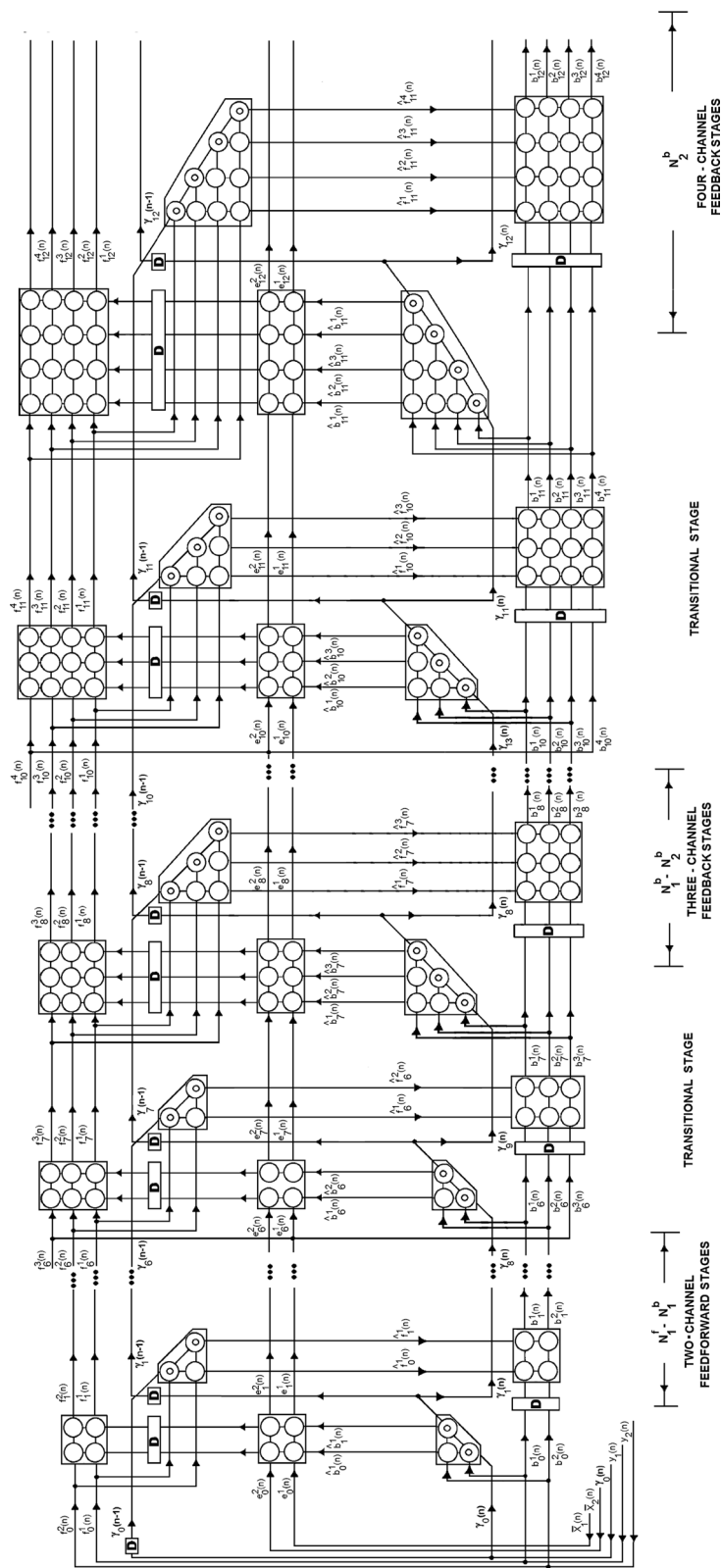


Figure 5 Diagram of four-channel DFE using SPMLGs.

first and the second channels of the DFE under consideration, while the third and fourth rows are associated with the detected and fed back signals. Lattice orthogonalization begins with the elements of the first two rows using two-channel sequential lattice processing stages until the first fed back channel is incorporated as the new channel at a transitional stage, which is the $(N_1^f - N_1^b + 1)$ th stage. Then, the orthogonalization continues with three-channel lattice stages until the fourth channel, which is related to the detected and fed back signal of the second channel, is taken into the process at another transitional stage, which is the $(N_2^f - N_2^b + 1)$ th stage, and so the orthogonalization of input data finalizes with four-channel stages when the mean squared estimation error performance requirements are met, and thereby the k th desired signal, $d^k(n)$, is sequentially estimated using self-orthogonalized backward prediction error signals as follows:

$$\begin{aligned} \hat{d}_n^k(i) = & \sum_{m=1}^{(N_1^f - N_1^b)} \sum_{j=1}^2 \bar{\kappa}_{m,k,j}^*(n-1) \hat{b}_{m-1}^j(i) \\ & + \sum_{m=(N_1^f - N_1^b + 1)}^{(N_2^f - N_2^b)} \sum_{j=1}^3 \bar{\kappa}_{m,k,j}^*(n-1) \hat{b}_{m-1}^j(i) \\ & + \sum_{m=(N_2^f - N_2^b + 1)}^{N_2^f} \sum_{j=1}^4 \bar{\kappa}_{m,k,j}^*(n-1) \hat{b}_{m-1}^j(i). \end{aligned} \quad (60)$$

Here, the first and second summations represent the estimation accomplished by the two-channel and three-channel sections, respectively, and the third summation is connected with the four-channel estimation section. In each section, $\bar{\kappa}_{m,k,j}(n)$ represents the j th estimation reflection coefficient at the m th stage related to the k th channel as defined in the previous subsection, and $\hat{b}_{m-1}^j(i)$ represents the j th element of the self-orthogonalized backward prediction error signal vector, $\hat{\mathbf{b}}_{m-1}(i)$, at the input of the m th stage. The self-orthogonalized backward prediction error vector, $\hat{\mathbf{b}}_{m-1}(i)$, is produced by the lower triangular transformation of the input backward prediction error vector, $\mathbf{b}_{m-1}(i)$, using $\mathbf{L}_m(n)$, and this operation is accomplished at the triangular shaped self-orthogonalization processor (related to forward prediction) of the m th SPMGLS. Note that the sizes of vectors, $\hat{\mathbf{b}}_{m-1}(i)$ and $\mathbf{b}_{m-1}(i)$, and matrix, $\mathbf{L}_m(n)$, at different sections of the proposed lattice equalizer are as follows: 2×1 and 2×2 in two-channel section, 3×1 and 3×3 in three-channel section, and 4×1 and 4×4 in four-channel section, respectively.

3 Adaptive multiple systolic Viterbi detection

In order to achieve an all systolic equalizer architecture, we propose to use the systolic array processor approach in [36] for the design and implementation of Viterbi detection, so that a high degree of computational concurrency is obtained by operating simultaneously and in synchronization with the rest of equalizer circuitry. Accordingly, the most computationally intensive operation in the Viterbi detection of sent data is related to the comparator metric, and a systolic computation of this metric is accomplished by multiply and accumulate operations:

$$\text{STMO}_k^j = \text{OSMI}_k^j + z_n^k(i) + \sum_{n=0}^{3N_k^d - 3} c_k(n) x_k(i-n), \quad j=1, 2, \dots, \nu, \quad (61)$$

for ν branches leading from states at time instant $i - 1$ to each state at time instant i as illustrated in Figure 6. Herein, ν stands for the alphabet size, OSMI_k^j and STMO_k^j represent old survivor state metric input (OSMI) and state metric output (STMO) for the j th branch leading to the state constituted by the vector $[x_k(i-1)x_k(i-2)\dots x_k(i-N_k^d+1)]$ related to the k th channel of equalizer, and $z_n^k(i)$ is the k th equalizer output as given in (5). Note that the elements of coefficient vector $\mathbf{c}_k(i)$:

$$\mathbf{c}_k(n) = [c_k(n), c_k(n-1), \dots, c_k(n-2N_k^d+2)]^T \quad (62)$$

are computed as:

$$c_k(n) = \sum_{i=0}^{N_k^d - 1} w_k(i) w_k(n-i) \quad (63)$$

where \mathbf{w}_k represents the coefficient vector for the k th ADIR filter as defined in (10). Note that a processing element (PE) of the systolic array in Figure 6 is symbolized with a circled PE, and the memory of computation cycle is designated as $L = 3N_k^d - 3$ for the ease of illustration.

4 Computational complexity

The computational complexity can be calculated by considering two main sections of the proposed channel shortening equalizer. The first section implements the MIMO-DFE while the second one is related to the Viterbi processing. The number of operations required for the MIMO-DFE can be calculated by thinking about the number of operations per stage and the number of stages. The number of operations for a single SPMGLS with two, three, and four channels have been computed by making use of complexity calculations in [23] and [33] as 84,171, and 288, respectively. There are $(N_1^f - N_1^b)$ stages in two-channel section, and thereby the number of required

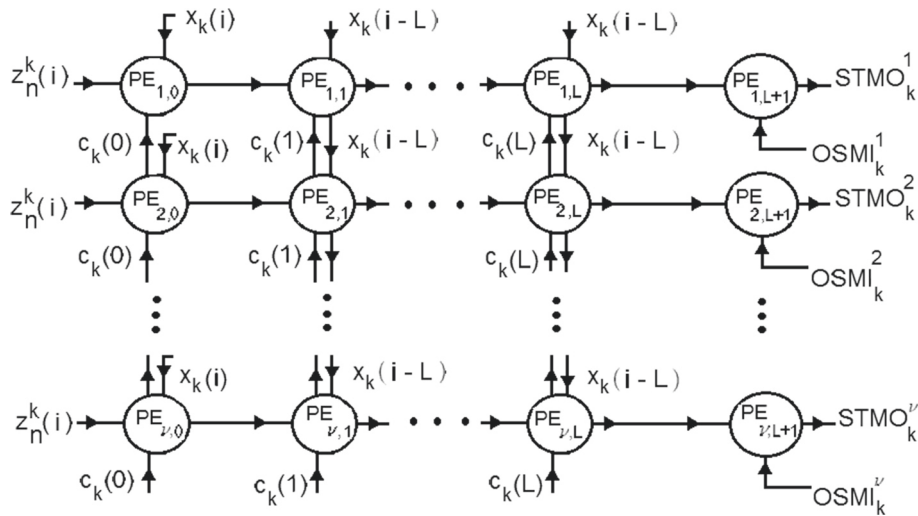


Figure 6 ν branches from states at time instant $i - 1$ to each state at time instant i .

operations for this section sums up to $84(N_1^f - N_1^b)$. Similarly, the number operations for three and four channel sections are calculated as $171(N_1^b - N_2^b)$ and $288N_2^b$, respectively. Therefore, the total number of operations for the MIMO-DFE in Figure 5 becomes $84N_1^f + 87N_1^b + 117N_2^b$.

The complexity calculation for systolic Viterbi section can be accomplished as follows. The total number of processing elements in the systolic array that implements Viterbi processing per channel is given as $\nu \times N_k^d$ in [36]. Each element in this array performs one addition and one

multiplication, which are counted together as one operation. Then, the total number of operations for the systolic implementation of M Viterbi detectors are calculated as $M \times \nu \times N_k^d$. Accordingly, the total computational complexity for the proposed equalizer taking into account both MIMO-DFE and multiple Viterbi detector sections becomes $84N_1^f + 87N_1^b + 117N_2^b + M\nu N_k^d$.

We compare the computational complexity of the proposed channel shortening equalizer using a front-end feed forward equalizer (CSFFE) and a front-end decision feedback equalizer (CSDFE) with those of VE in Figures 7,

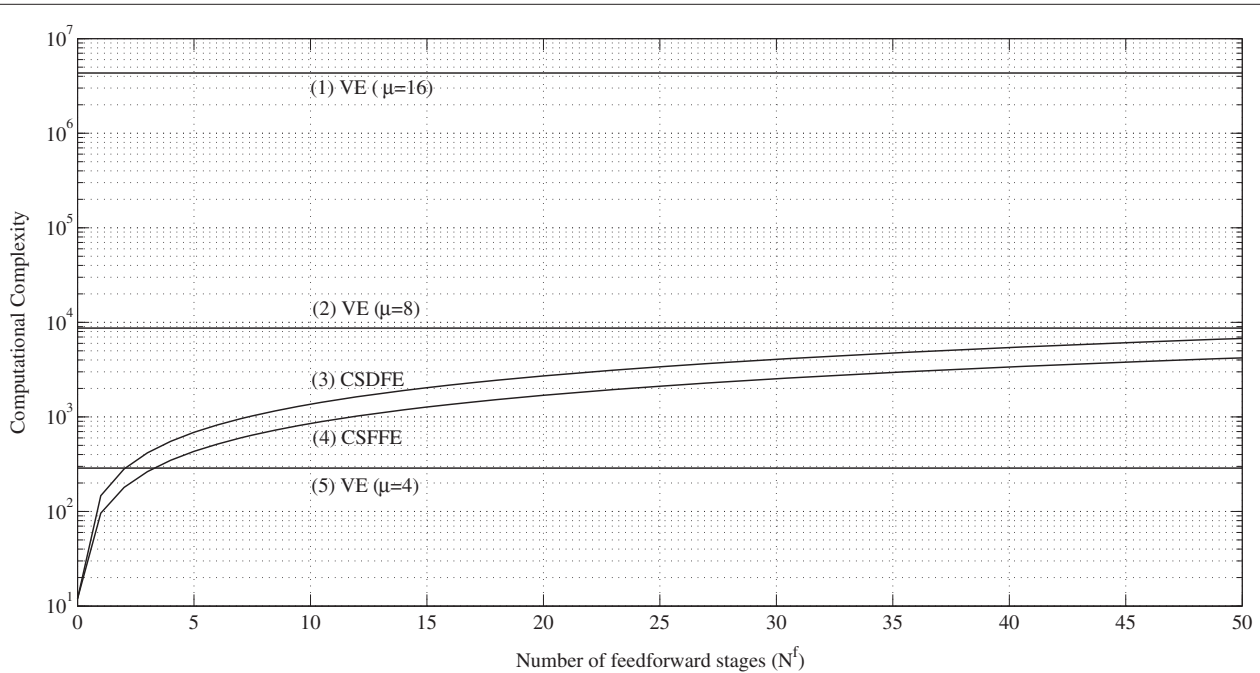
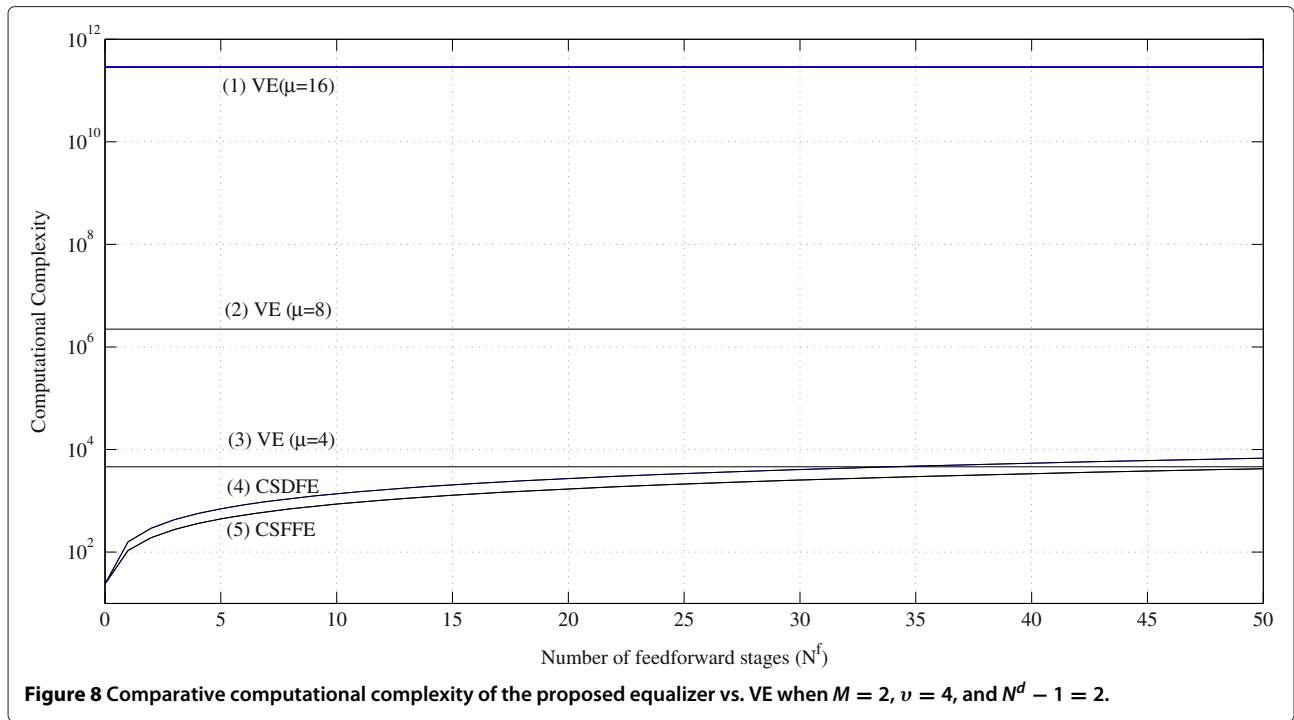
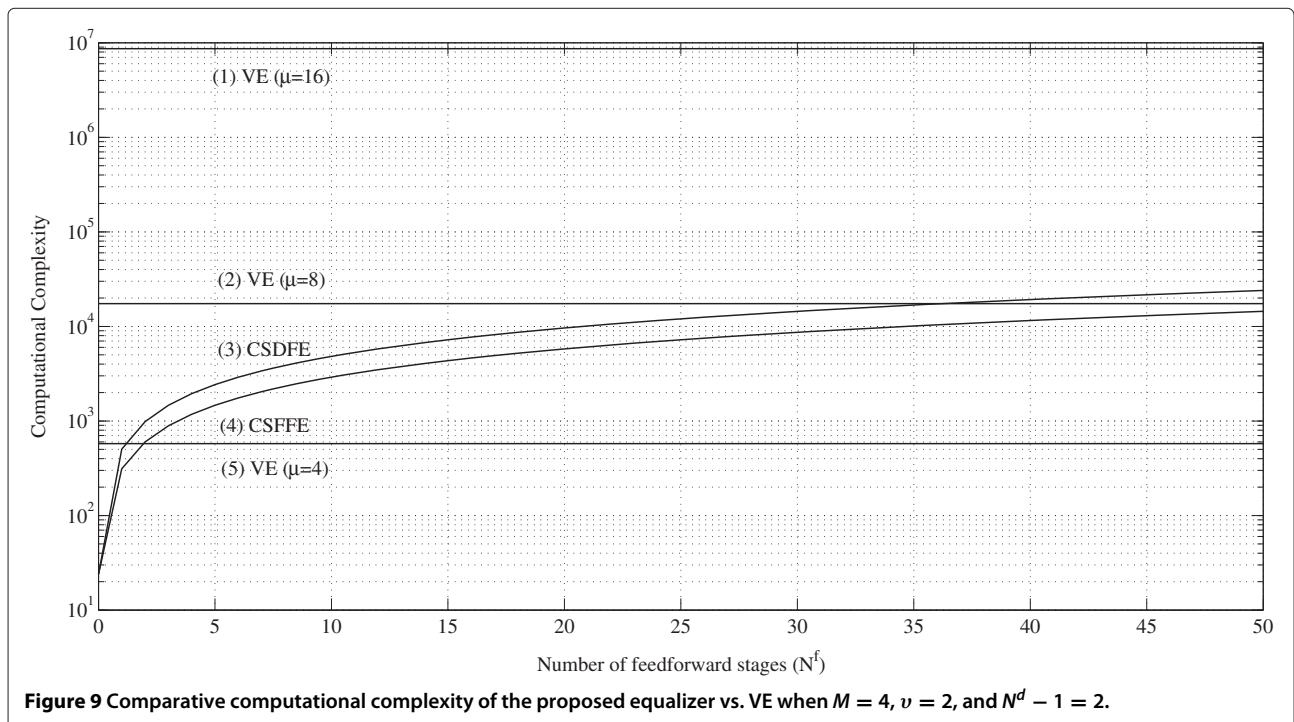


Figure 7 Comparative computational complexity of the proposed equalizer vs. VE when $M = 2$, $\nu = 2$, and $N^d - 1 = 2$.



8, and 9. When generating the complexity curves, we assumed that $N^f = N_1^f = N_2^f$ and $N^d = N_k^d = 3$ for both CSFFE and CSDFE and additionally $N_b = N_1^b = N_2^b = 0.25 \times N^f$ for CSDFE and $N_b = 1$ for CSFFE. The computational complexity for VE is calculated by taking into account the most computationally demanding operation, i.e., comparative metric calculation, and is

given as $M \times (4\mu + 3) \times v^{\mu-1}$ [38], where each channel from transmitter to receiver is assumed to have a memory of μ . In Figure 7, we plotted the complexity curves when $M = 2$, $v = 2$, and $\mu = 4, 8, 16$. It shows that the proposed method is not computationally advantageous when the channel memory ($\mu = 4$) is two times the ADIR filter memory ($N^d - 1 = 2$). However, the complexity of



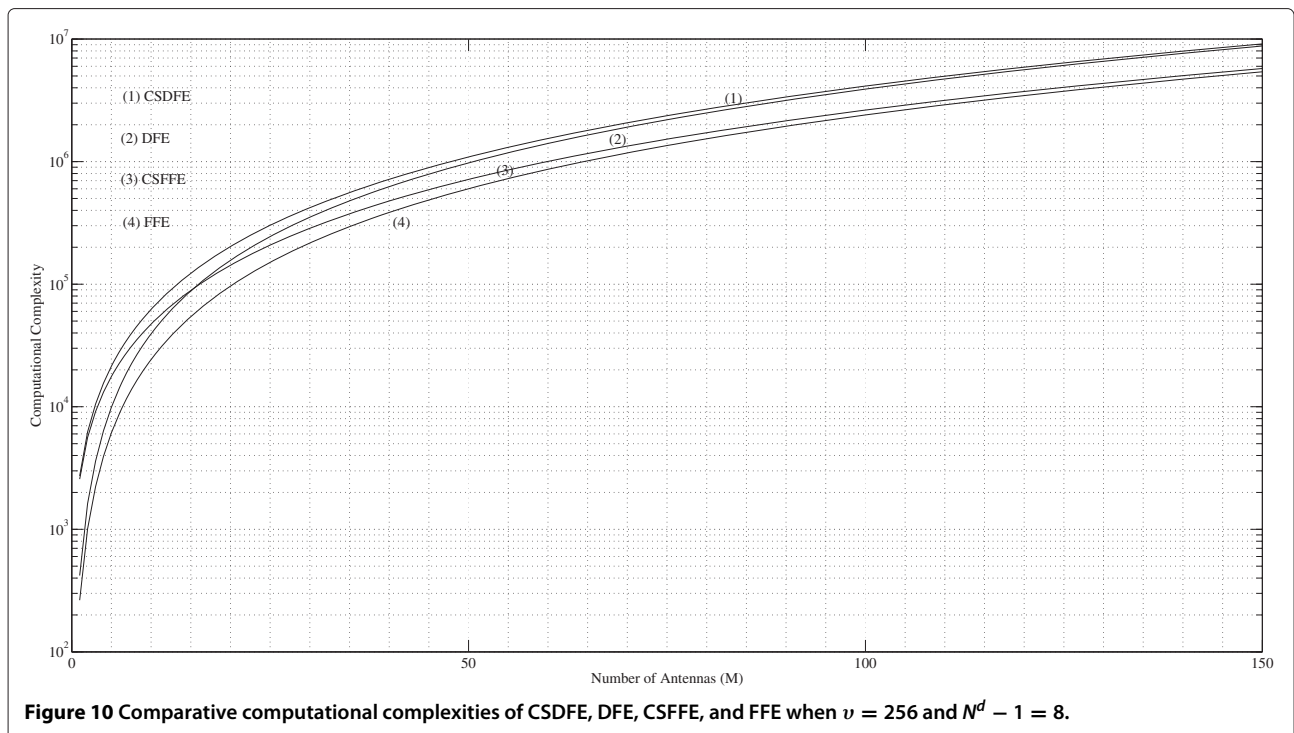
the proposed method becomes advantageous regardless of feed forward filter length (N^f) used when the channel memory is four times the ADIR filter memory and far more beneficial when the channel memory is eight times the ADIR filter memory. Figure 8 displays that the computational advantage of the proposed method comparing to VE becomes more attractive when the alphabet size is increased from $\nu = 2$ to $\nu = 4$ even for the channel memory values of $\mu = 4, 8$; and Figure 9 demonstrates that the computational advantage becomes less pronounced when the number of antennas increases from $M = 2$ to 4.

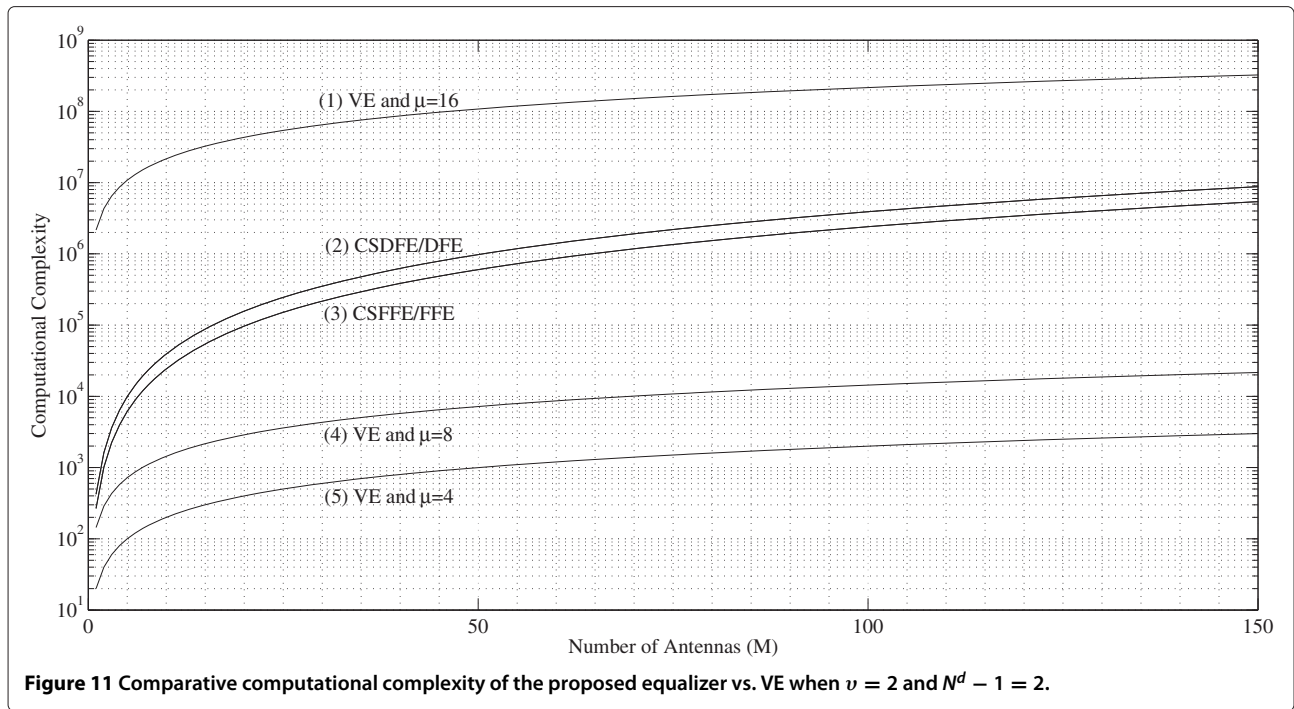
The computational complexity vs. the number of antennas analysis has also been carried out due to the recent interest in massive MIMO for next-generation wireless systems [37]. We would like to point out that the computational complexities of CSDFE and CSFFE are larger than DFE and FFE, respectively, by an amount of $M \times \nu \times N^d$, assuming $N^d = N_k^d, \forall k$. In this analysis, we assumed that $N^f = N_k^f = 12, \forall k, N^b = N_k^b = 0.25 \times N^f$, and $N^b = N_k^b = 1, \forall k$. In Figure 10, the computational complexity vs. number antennas curves for CSDFE, CSFFE, DFE, and FFE when $\nu = 256$ and $N_d - 1 = 8$, and M is increasing from 0 and to 150, are presented. For smaller values of ν and $N_d - 1$, the difference between the computational complexities of CSDFE and DFE or CSFFE and FFE, respectively, is minor, and the curves of CSDFE and DFE or CSFFE and FFE can not be discriminated, which implies that the performance increase by the implementation

of CSDFE instead of DFE or CSFFE instead of FFE, respectively, is achieved at the expense of almost negligible computational complexity cost, as will be clearer in the next section. In Figure 11, we compare the computational complexity of the proposed method with those of VE when the channel memory values of $\mu = 4, 8$, and 16 are used, the alphabet size is $\nu = 2$, the ADIR memory is assumed as $N_d - 1 = 2$, and M is increasing from 0 to 150. Finally, we repeat the same comparison in Figure 12 for $\nu = 4$ in order to demonstrate the effect of using larger alphabet size. Figure 11 shows that the proposed method is computationally advantageous comparing to VE when channel memory is larger than eight ($\mu > 8$), and in Figure 12, it can be seen that the computational complexity advantage of the proposed method improves when the alphabet size is increased from $\nu = 2$ to $\nu = 4$, and the proposed method becomes less complex comparing to VE even when $\mu = 8$.

5 Experimental results

The performance of the proposed receiver was investigated by means of MSE and probability of error simulations. In these evaluations, we considered linear time-invariant channels with spectral nulls as well as time-varying channels. The following channel impulse response matrices were defined in order to be used in simulations that demonstrate performance of the proposed equalizer:





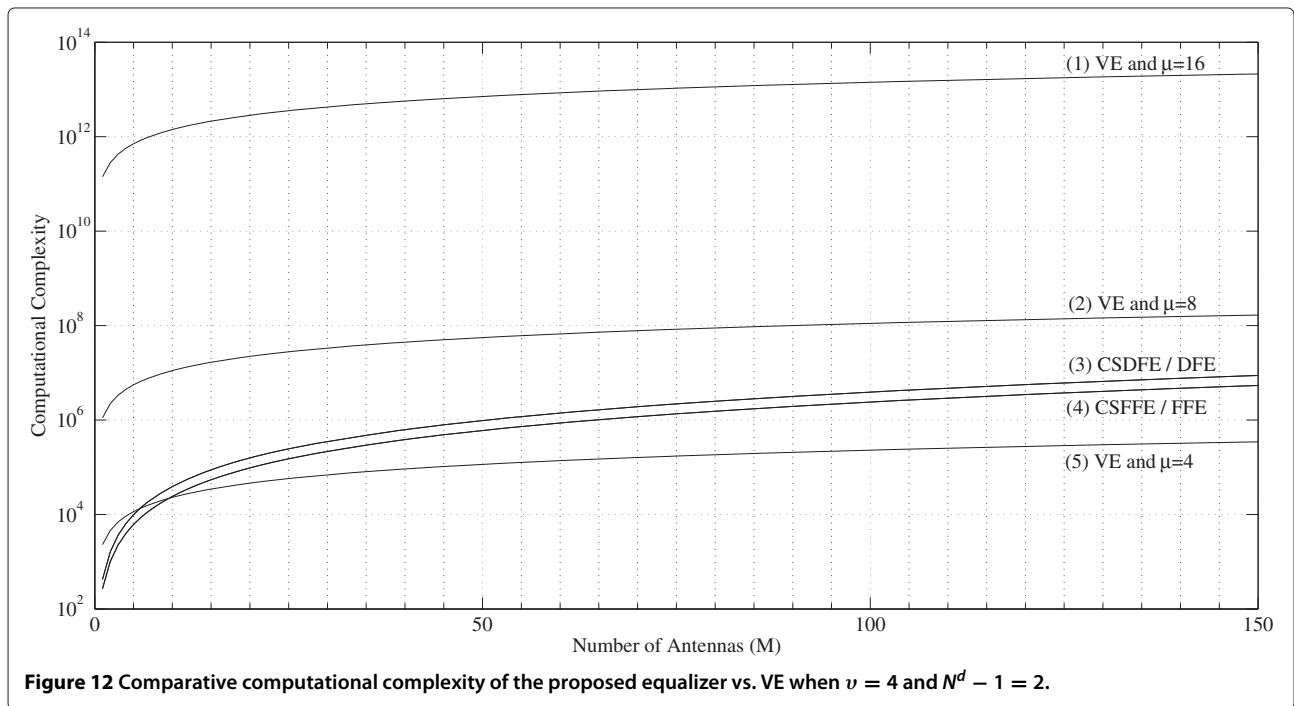
$$\mathbf{h}_1(n) = \begin{bmatrix} h_a(n) & \rho \cdot h_a(n-1) \\ \rho \cdot h_a(n-1) & h_a(n) \end{bmatrix}$$

$$\mathbf{h}_2(n) = \begin{bmatrix} \delta(n) & \rho \cdot \delta(n-1) \\ \rho \cdot \delta(n-1) & \delta(n) \end{bmatrix}$$

$$\mathbf{h}_3(n) = \begin{bmatrix} h_b(n) & \rho \cdot h_b(n-1) \\ \rho \cdot h_b(n-1) & h_b(n) \end{bmatrix} \quad (64)$$

where $\delta(n)$ represents the dirac delta function, and the channel impulse response $h_a(n)$ is defined as $h_a(n) = \sum_{j=0}^4 a_j \delta(n-j)$, which has spectral nulls and a large eigenvalue spread ($\chi = 1317.65$). Herein, the channel coefficients have been taken from [39], and are given by:

$$\mathbf{a}_j = [0.227045 \ 0.460091 \ 0.688136 \ 0.460091 \ 0.227045] \quad (65)$$



with $\sum_{j=0}^4 a_j^2 = 1$. Note that eigenvalue spread (χ) was determined by using the method described in [40], assuming a feed forward equalizer with a memory of $N^f - 1 = 18$, and a noise variance of 0.001. The time-varying channel impulse response is defined as $h_b(n) = \sum_{j=0}^4 b_j(n)\delta(n - j)$, where $b_j(n)$ represents the j th time-variant attenuation factor, generated independently using the improved version of Jakes' Rayleigh fading model in [41] by assuming a data rate of 100 kbytes/s and doppler shifts of $f_D = 50$ Hz and $f_D = 10$ Hz; $b_j(n)$ is also normalized such that $\sum_{j=0}^4 b_j^2(n) = 1$ for all n . By choosing the same channel for both direct and indirect channels, we demonstrate the performance under such a severe distortion situation that both ISI and ICI are significant, and therefore, the use of DFE is justified. On the other hand, we could have generated a longer channel with the same spectral characteristics (or eigenvalue spread), in which case we would not be able to benchmark the performance with those of VE, as the simulation of VE becomes computationally cumbersome for longer channels. Note that ρ represents the gain factor for the effect of ICI and takes values between 0 and 1. In this presentation, we consider two values of the gain factor, $\rho = 0$ and $\rho = 1$, which correspond to completely orthogonal and nonorthogonal transmissions, respectively [42].

In the simulations for the performance evaluation of the proposed method (CSDFE/CSFFE) with respect to FFE, DFE, and VE, and also in the simulations for the performance comparison of the proposed method when different ADIR filter memories are used, the input signal $x(n)$ applied to the channel was made of uniformly distributed bipolar $(+/-)1/\sqrt{2}$ random numbers because of relative simplicity it provided in simulations. Note that the uniformly distributed bipolar random numbers represent BPSK modulation supported by the IEEE 802.11n WLAN standard [43]. In order to account for the modulations that are both in the IEEE 802.11n WLAN and IEEE 802.16e MAN standards and to demonstrate the effect of higher modulation on probability of error performance, we also performed CSFFE/CSDFE simulations by using the input signal made of uniformly distributed random numbers taking values from $(1/2+i/2, -1/2+i/2, -1/2-i/2, 1/2-i/2)$, which represents QPSK modulation ($i = \sqrt{-1}$), and compared against the performance results of the proposed method for BPSK modulation in both time-invariant and time-variant channel cases. Moreover, we assume the complete knowledge of time-invariant channel and the knowledge of time-variant channel memory in VE and adaptive Viterbi equalizer (AVE) performance evaluations, respectively. On the other hand, the proposed method does not need any information about channel; however, if information about channel is already available and an evaluation on the badness of channel can be carried out, the result of this evaluation can be used

to determine the ADIR filter memories of the proposed equalizer so as to improve the performance.

The channel noise signal was additive white Gaussian noise (AWGN) with zero mean and is uncorrelated with the input signal. The received signal-to-noise ratio (SNR) per each channel of the receiver is defined as:

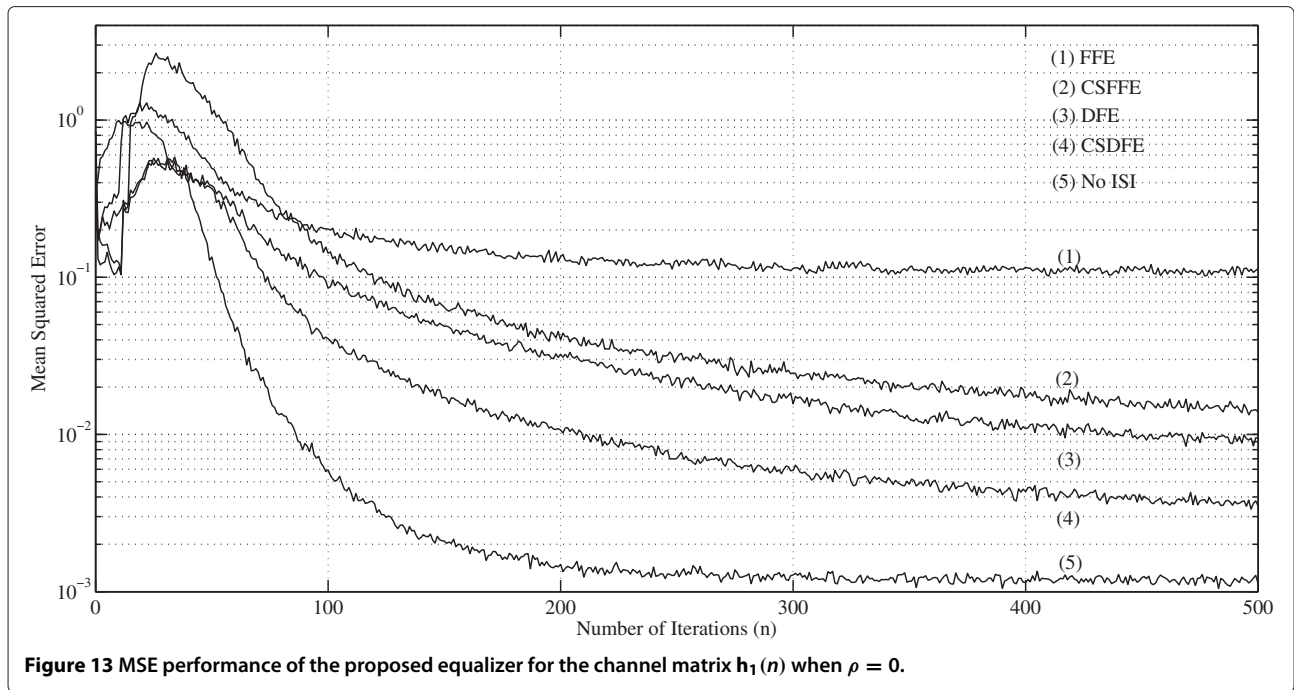
$$\text{SNR}_k \triangleq \frac{1}{2\sigma_{n_k}^2} \quad (66)$$

where $\sigma_{n_k}^2$ is the variance of AWGN for the k th channel. Accordingly, SNRs for all channels of the receiver are equal, and the system SNR is defined as $\text{SNR} = \text{SNR}_k$ for $k = 1, 2$. The exponential weighting factors were 0.99 and 1.0 for the front-end equalizer and ADIR filter, respectively, when the channel was time-invariant. In time-varying channel case, they were assumed as 0.975 and 1 in order to better track the signal. The probability of error evaluations were conducted using 4×10^5 samples so that 2×10^5 samples per channel were used, and the simulations were carried out in training mode of receiver operation. The delays (D_k) for the desired signals in this mode of operation were assumed equal ($D = D_k$) for $k = 1, 2$, and D was chosen so as to minimize MSE; that is, $D = (N^f - 1 + \aleph - 1)/2$ when only front-end FFE is utilized, $D = (N^f - 1 + N^b - 1 + \aleph - 1)/2$ when only front-end DFE is implemented, $D = (N^f - 1 + \aleph - 1)/2 - N^d + 1$ for CSFFE, and $D = (N^f - 1 + N^b - 1 + \aleph - 1)/2 - N^d + 1$ for CSDFE. Note that the memories of feed forward and feedback sections of DFEs, ADIR filters, and channel impulse responses for $k = 1, 2$ were assumed equal, i.e., $N^f - 1 = N_k^f - 1$, $N^b - 1 = N_k^b - 1$, $N^d - 1 = N_k^d - 1$, and $\aleph - 1 = \aleph_k - 1$. The noise variance per channel during MSE simulations was 0.001.

5.1 Time-invariant channel

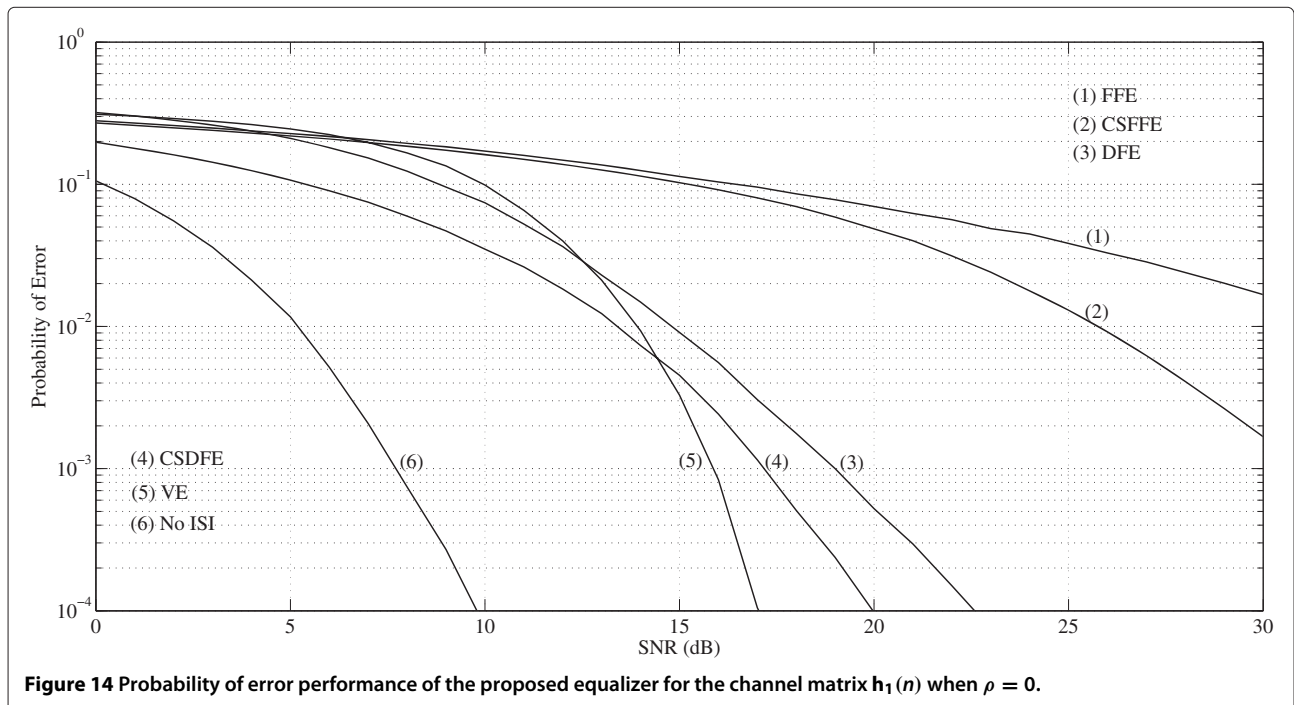
The objective of simulations with the channel matrix $\mathbf{h}_1(n)$ is to display the performance of the proposed CSDFE with respect to CSFFE, FFE, DFE, and VE. In the simulations, we have taken into account the modularity and regularity properties of SPMGLSs and started simulations with FFE; after observing the performance, we altered the equalizer to CSFFE, then we added new SPMGLSs to alter the equalizer to DFE, subsequently to CSDFE. The memory of FFE ($N^f - 1$) was 18, while the memory of feedback channels ($N^b - 1$) for DFE was 4, and the memory of ADIR ($N^d - 1$) filter was 2.

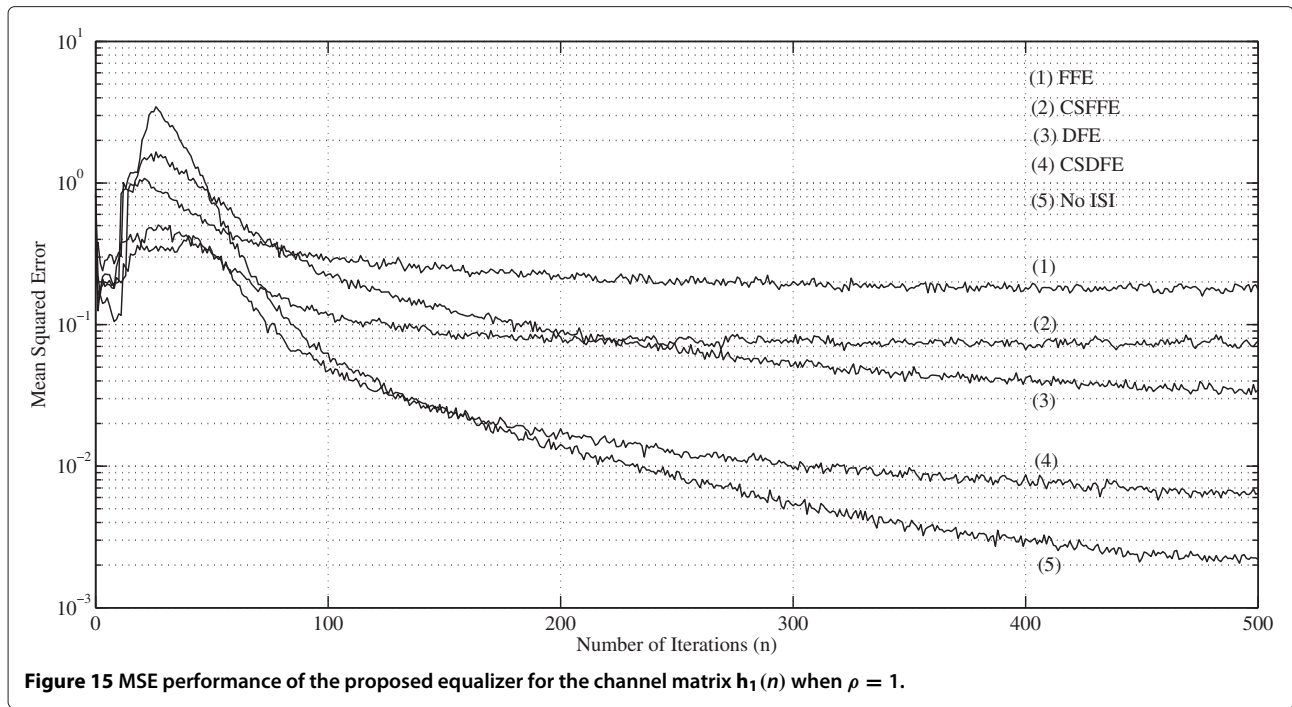
In Figure 13, we present the MSE performance of the proposed equalizer when orthogonal transmission is used, i.e., $\rho = 0$, and also compare its performance to those of CSFFE, FFE, and DFE together with the performance for the channel matrix $\mathbf{h}_2(n)$. In Figure 14, we provide the corresponding probability of error performance for orthogonal transmission and compare the performance



of CSDFE with those of CSFFE, FFE, DFE, VE, and with the performance for the channel matrix $\mathbf{h}_2(n)$. It can be seen in these figures that the performance of CSDFE is better than those of CSFFE, FFE, and DFE, respectively. It also has the closest probability of error performance to that of VE. Note that neither MSE nor probability of error performances for the channel matrix $\mathbf{h}_2(n)$ with $\rho = 0$ change with the use of different equalizers,

CSDFE, CSFFE, DFE, or FFE, since this channel matrix with $\rho = 0$ does not include ISI and ICI components. Figures 15 and 16, on the other hand, display MSE and probability of error comparisons when $\rho = 1$, and also provide comparison with respect to the performance of DFE when the channel matrix $\mathbf{h}_2(n)$ is used, since the channel matrix $\mathbf{h}_2(n)$ with $\rho = 1$ does not have ISI components.





The effect of ICI on the performance of CSDFE and CSFFE can be seen by comparing MSE values between Figures 13 and 15 and by collating probability of error values between Figures 14 and 16. It can be deduced from these comparisons that the performance improvement that can be achieved by the combination of CSDFE

and orthogonal transmission is far more beneficial than using CSFFE with orthogonal transmission. Furthermore, Figures 17 and 18 show the performance improvement that can be achieved by using an ADIR filter memory ($N^d - 1$) of 3 instead of 2 for both CSFFE and CSDFE in $\rho = 0$ and $\rho = 1$ cases, respectively.

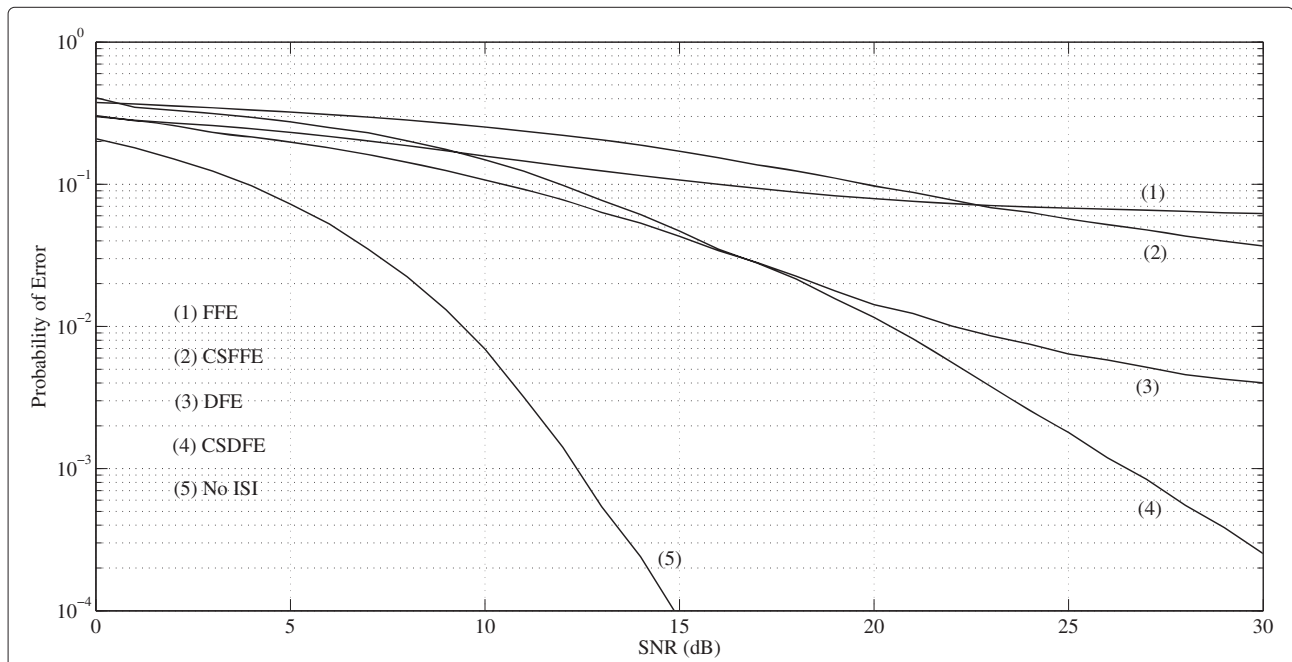
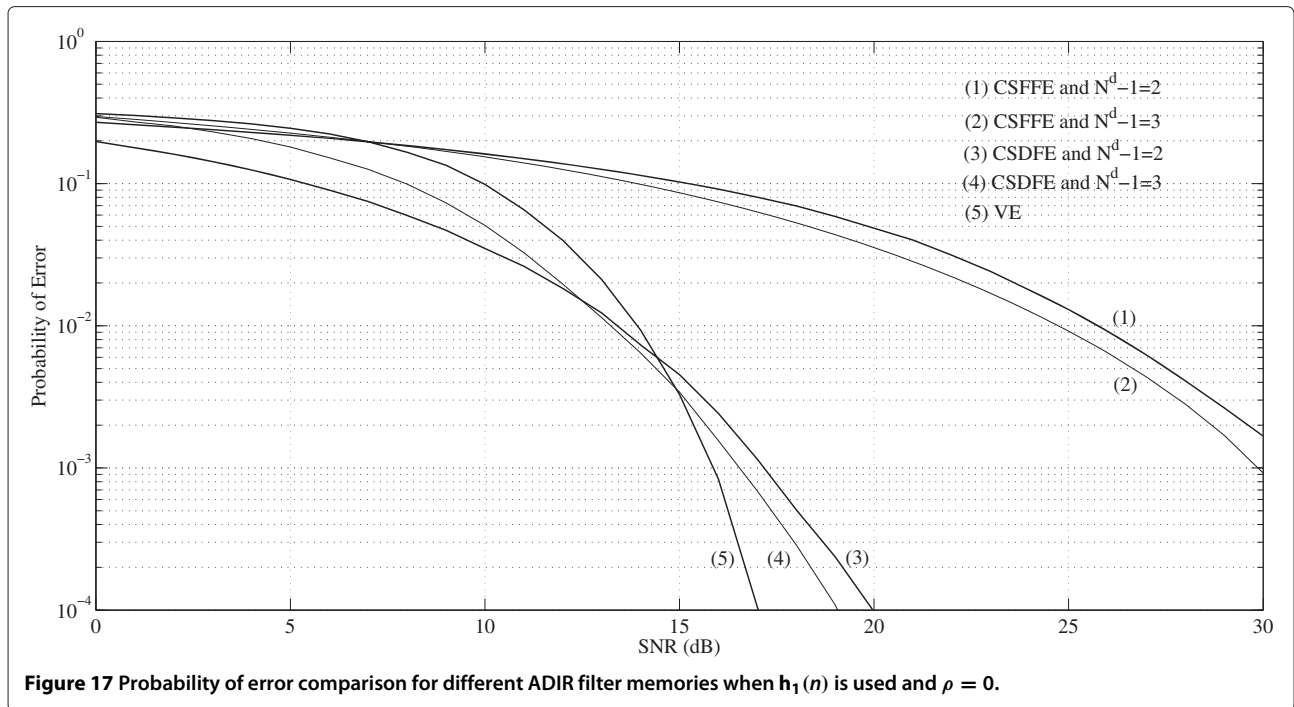


Figure 16 Probability of error performance of the proposed equalizer for the channel matrix $h_1(n)$ when $\rho = 1$.



In order to investigate the effect of channel memory for a given ADIR filter memory on the performance of the proposed method, we generated channels with longer impulse responses than that of $h_a(n)$, nevertheless, we made sure that these channels have exactly the same spectral characteristics or eigenvalue spread with $h_a(n)$. We then produced the corresponding channel matrices using the same channel impulse responses

for direct and indirect paths as in $\mathbf{h}_1(n)$ and repeated the aforementioned experiments and found out that the performance was not different from the ones displayed in Figures 13, 14, 15, 16, 17, and 18. Consequently, it can be said that, the eigenvalue spread of channel, not the memory as the channel shortener name implies, determines the performance of the proposed equalizer.

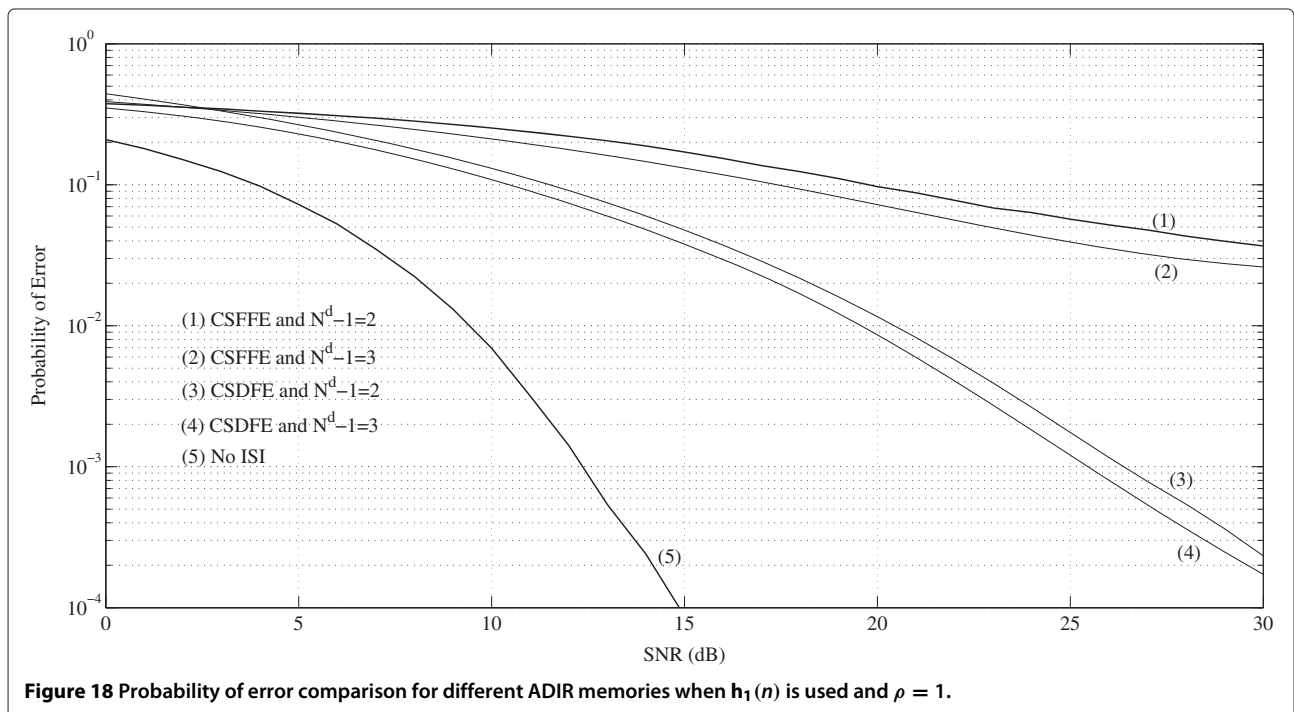
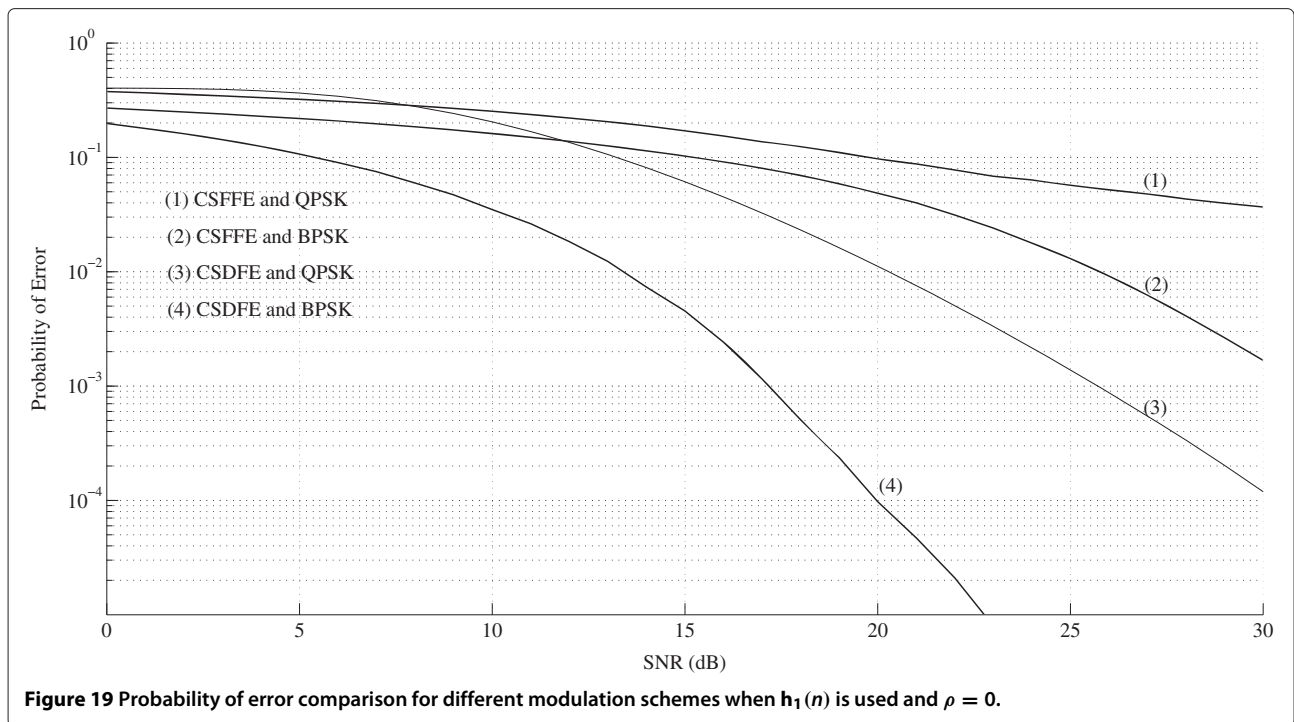


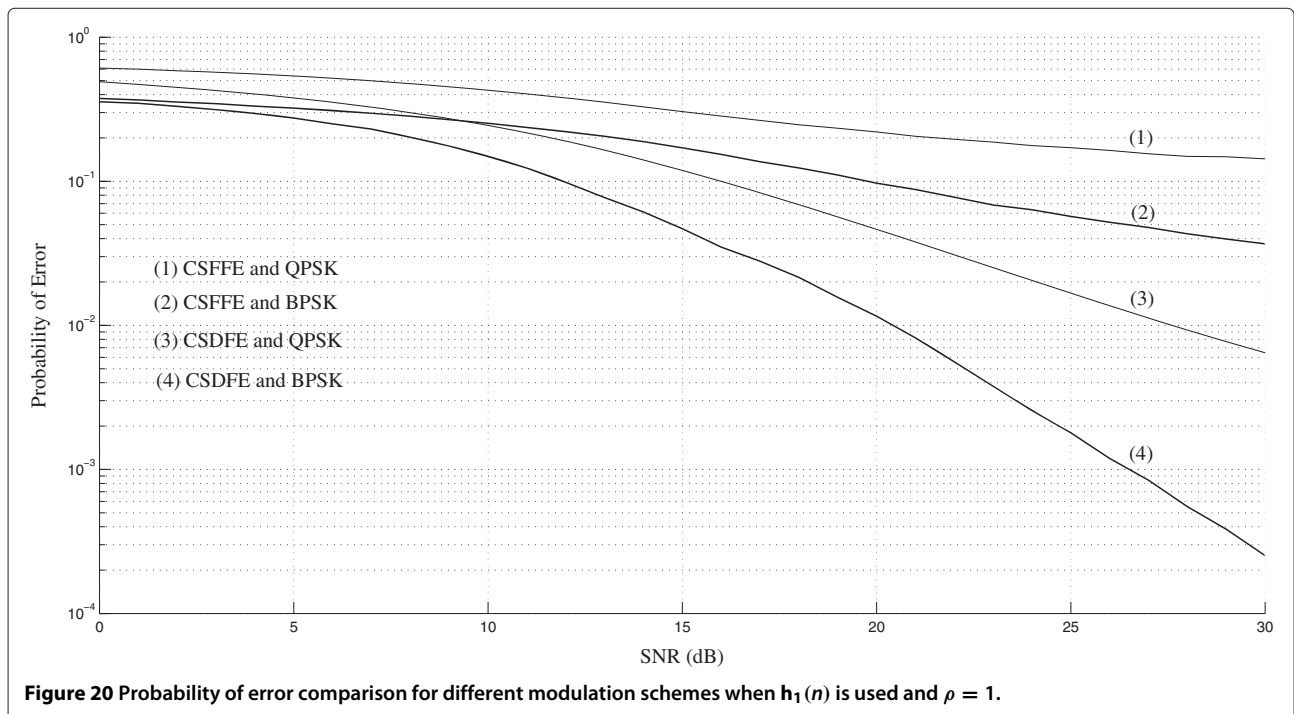
Figure 18 Probability of error comparison for different ADIR memories when $\mathbf{h}_1(n)$ is used and $\rho = 1$.

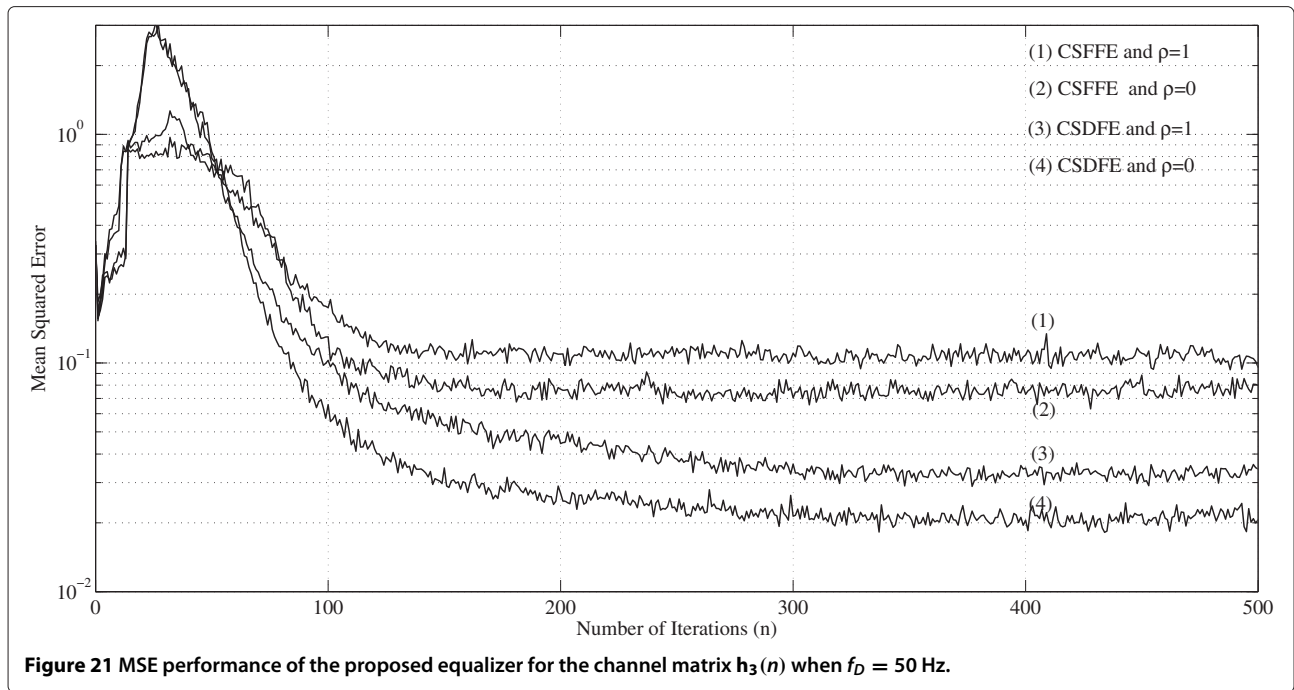


Subsequently, we examine the effect of using a higher modulation scheme on the probability of error performance of the proposed method, and in Figures 19 and 20, we present the performance degradation caused by switching modulation from BPSK to QPSK in orthogonal and nonorthogonal transmission cases, respectively.

5.2 Time-variant channel

Our objective in the simulations using the channel matrix $h_3(n)$ is to present the performance of the proposed equalizer under two different time-varying channel conditions. Figure 21 presents the MSE performance results when $f_D = 50$ Hz and provides comparisons between CSFFE

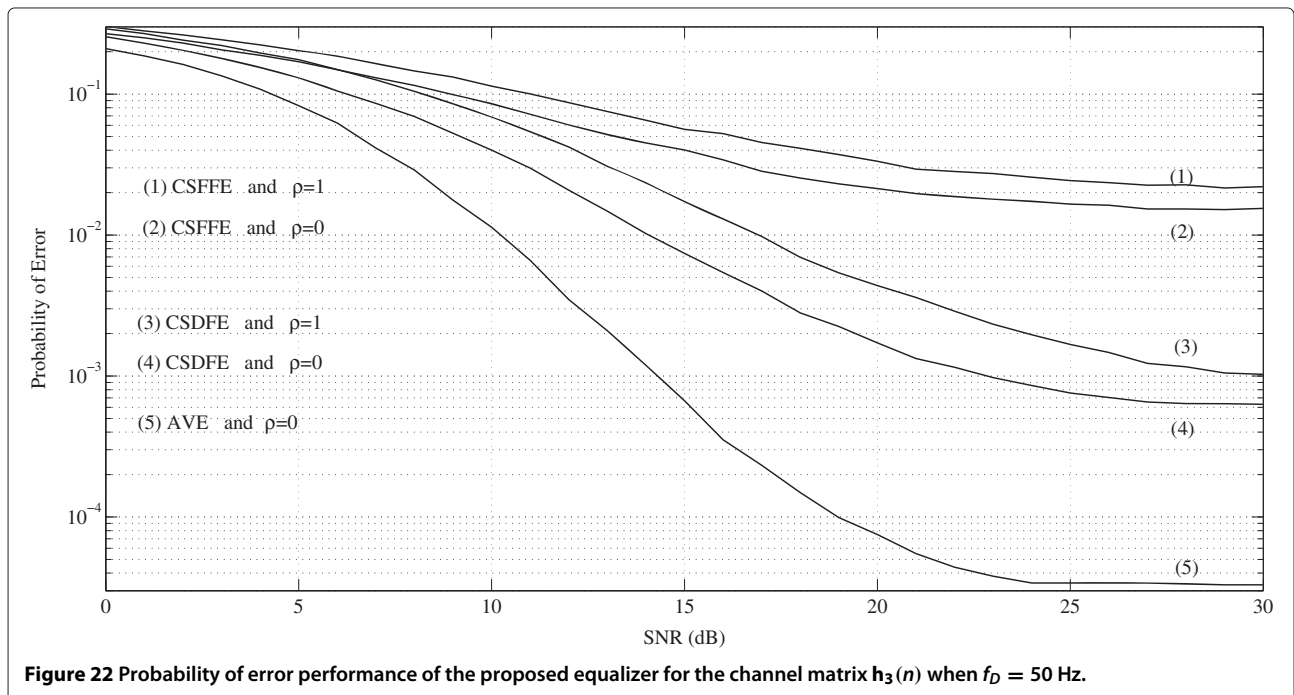


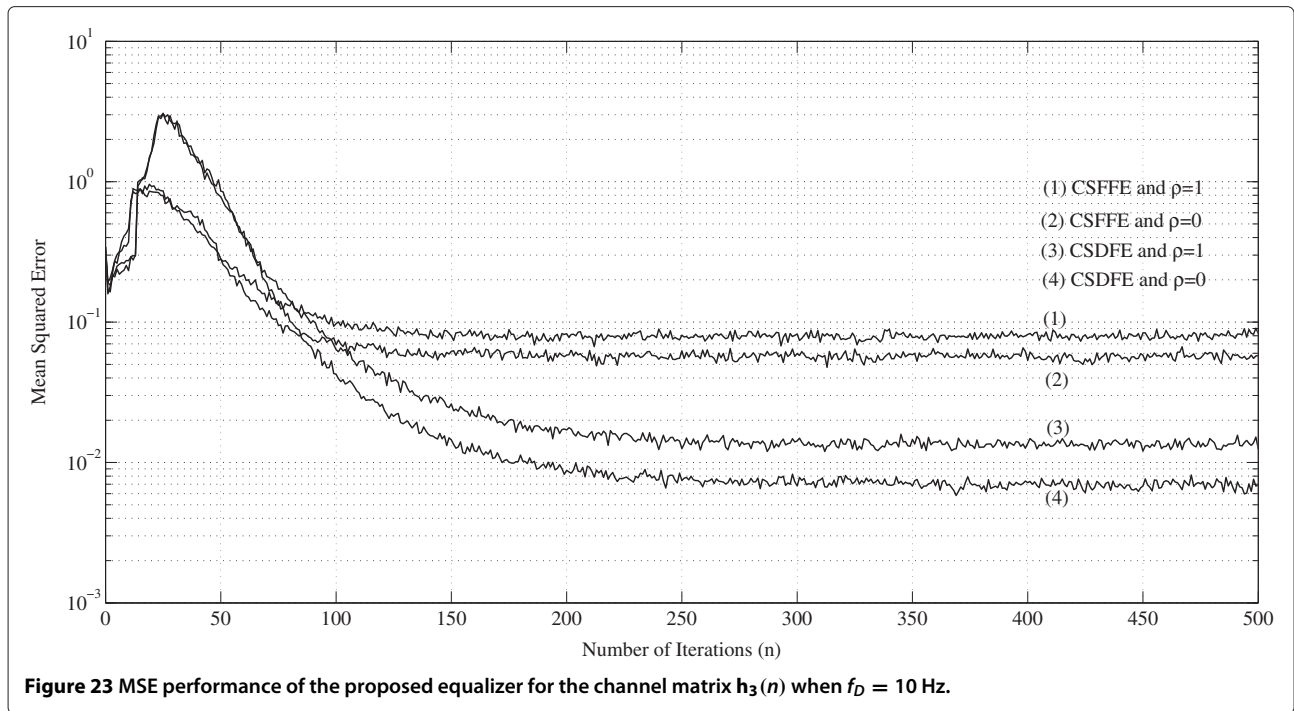


and CSDFE when $\rho = 1$ and $\rho = 0$. In Figure 22, we show the corresponding probability of error performances. Note that the probability of error values of CSFFE for $\rho = 1$ and $\rho = 0$ saturate approximately at SNR= 25 dB to 2×10^{-2} and 1.5×10^{-2} , respectively, the probability of error values of CSDFE, on the other hand, saturate approximately at SNR= 27 dB to 10^{-3} and 6.3×10^{-4} ,

which are closer to the probability of error values of AVE when $\rho = 0$, that converges to approximately 23.4×10^{-5} at SNR= 23 dB.

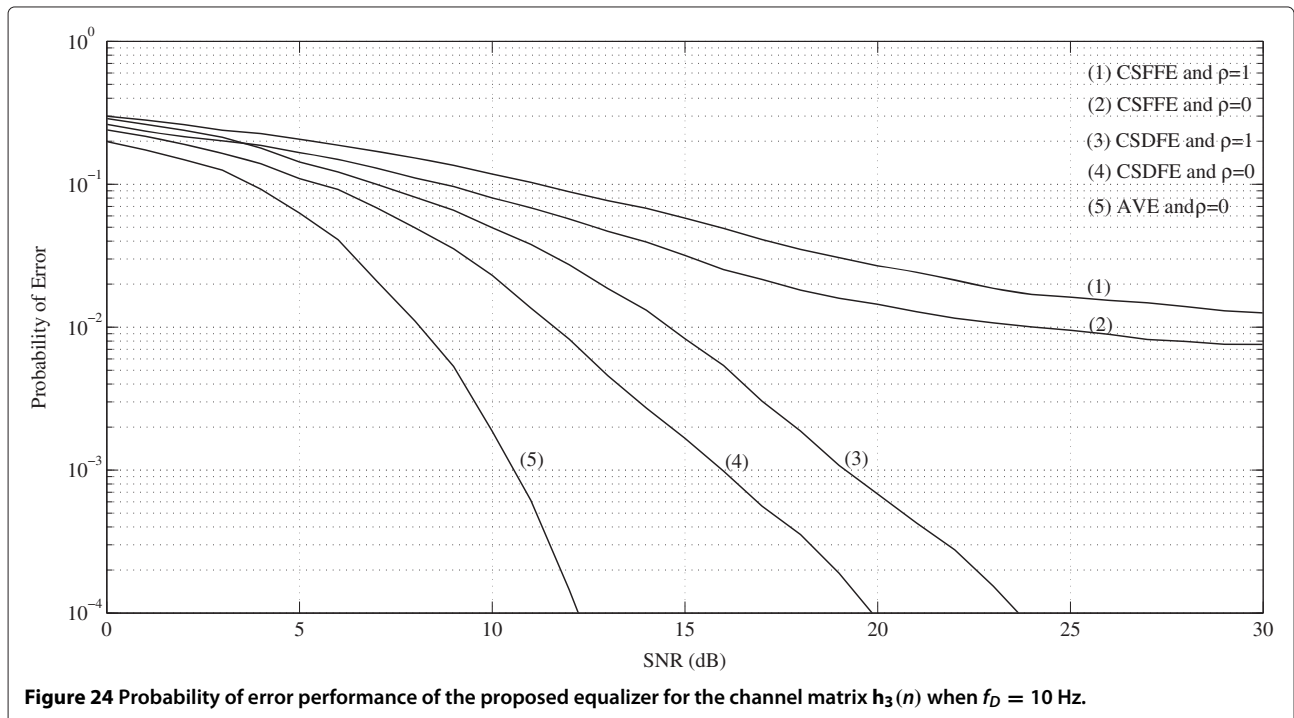
The second experiment under time-variant channel conditions was carried out using a lower doppler frequency so that the effect of doppler frequency on the equalizer performance can be demonstrated. When we

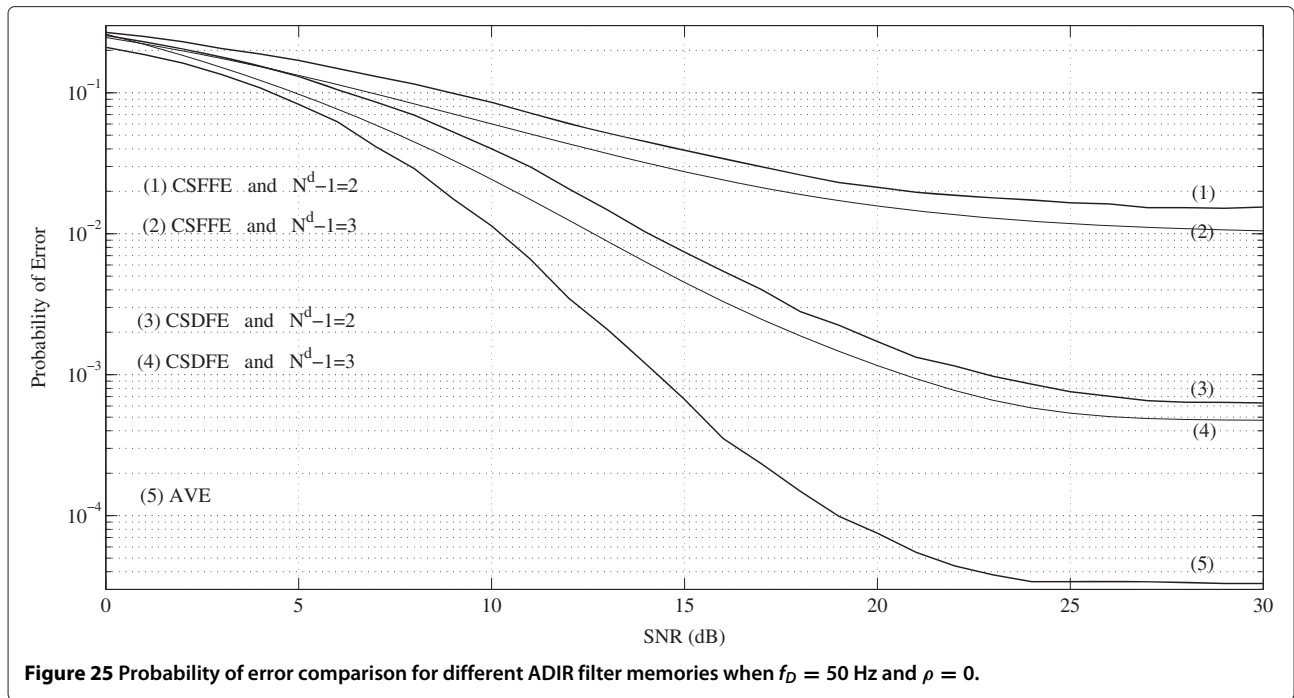




compare the MSE performance results in Figure 23 with those of Figure 21, which is related to the MSE performances for $f_D = 50$ Hz, we see that the MSE performances of CSFFE for $\rho = 1$ and $\rho = 0$ when $f_D = 50$ Hz are 1.33 and 1.22 times, respectively, higher than those of CSFFE when $f_D = 10$ Hz. The same comparison for CSDFE yields that the MSE performances when $f_D = 50$ Hz are 2 and

2.85 times higher for $\rho = 0$ and $\rho = 1$, respectively, than the MSE performances when $f_D = 10$ Hz. Similar evaluations can be done for the probability of error performances, and it can be seen in Figure 24 that the probability of error curves of CSFFE for $\rho = 0$ and $\rho = 1$ cases saturate to lower values than those of CSFFE in Figure 22. On the other hand, whereas probability of error curves of

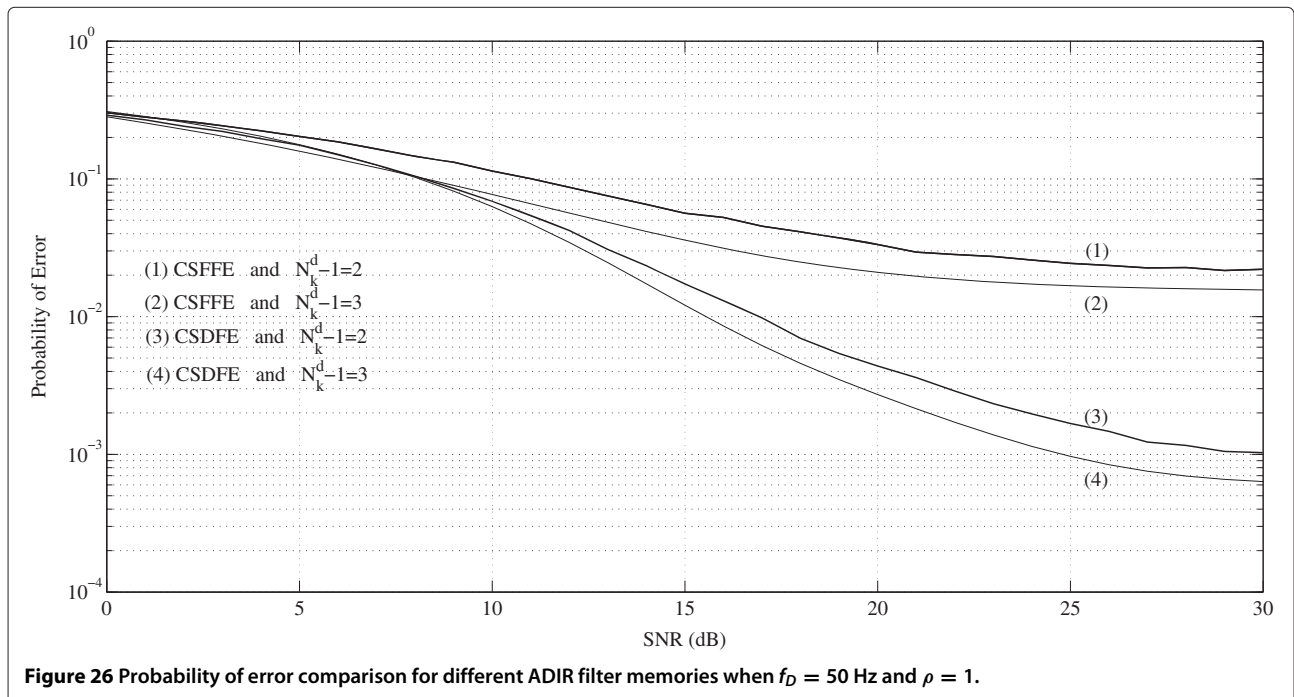


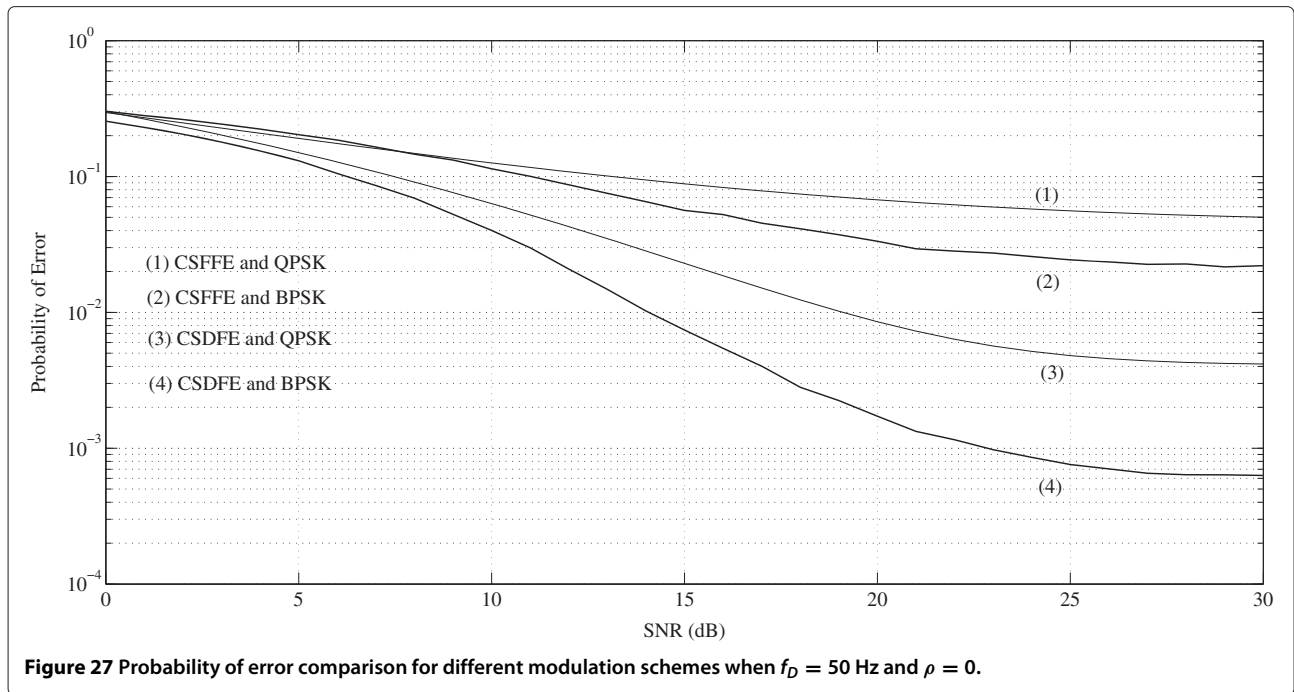


CSDFE for $\rho = 0$ and $\rho = 1$ in Figure 22 reach to the probability of error value of 10^{-2} at SNRs of 14 and 17 dB, respectively, they converge to the same probability of error value at approximately 3.5 dB lower SNRs, i.e., at 10.5 and 13.5 dB in Figure 24.

We have performed two more probability of error simulations for the time-variant channel case using $f_D = 50$ Hz,

the first of which was related to the performance improvement that can be gained by using an ADIR filter memory ($N^d - 1$) of 3 instead of 2. Accordingly, Figures 25 and 26 depict the achieved performance improvement for both CSFFE and CSDFE in $\rho = 0$ and $\rho = 1$ cases. The second simulation was for the effect of using QPSK modulation instead of BPSK, and Figures 27 and 28 therefore render



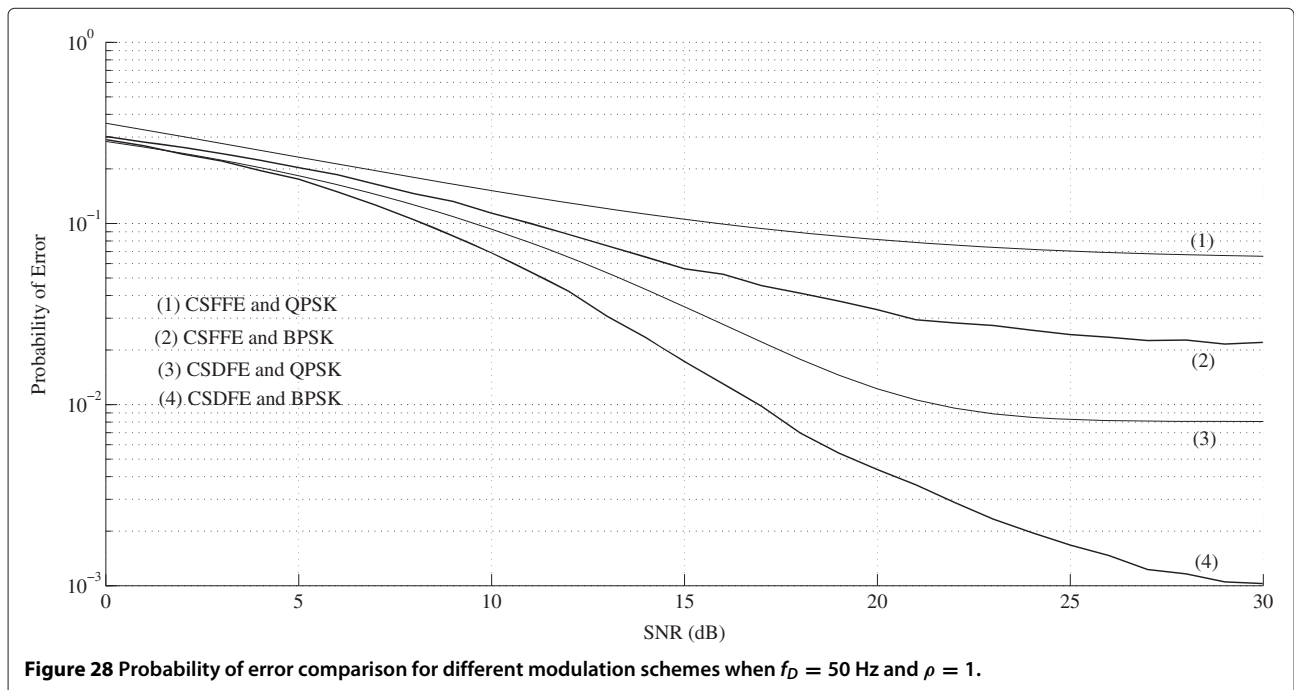


the probability of error performance comparison when using QPSK modulation instead of BPSK in $\rho = 0$ and $\rho = 1$ cases, respectively.

It can be seen in these figures that the performance improvement attainable through orthogonal transmission under time-variant channel conditions is not as significant as time-invariant channel conditions.

6 Conclusions

A V-BLAST type channel shortening equalizer design for cognitive MIMO-OFDM radios has been presented. The V-BLAST property for frequency-selective channels is realized by completely orthogonalizing the input data using SPMGLSs, so that a systolic MIMO-DFE is accomplished at the front-end of the proposed channel



shortening equalizer. Accordingly, ISI and ICI effects are suppressed due to completely orthogonalizing nature of the receiver structure. A systolic array implementation of multiple adaptive Viterbi detectors is also utilized in order to realize a channel shortening equalizer with high degree of computational concurrency. The matrix inversions, which are significant bottlenecks in receiver design, are avoided, scalar only operations are enabled. A highly modular, regular, order-recursive, and simple receiver architecture, which is suitable for the DSP chip- and FPGA-based signal processing implementations of MIMO-OFDM wireless communication systems, is obtained.

Spectrum sensing and positioning functions, important tasks for cognitive radios, can be accomplished at no cost by simply reconfiguring the front-end MIMO-DFE filter as spectral analysis and positioning filters, respectively. These properties make the proposed equalizer a good candidate for software defined cognitive radio receiver realizations of MIMO-OFDM systems.

The computational complexity of the proposed equalizer has been provided by separately taking into account the MIMO-DFE and multiple adaptive Viterbi detector sections. Then, the total complexity was compared to the complexity of VE for different channel and ADIR filter memories and different alphabet and antenna sizes.

The performance has been supplied in terms of MSE and probability of error analysis for orthogonal and nonorthogonal transmissions under time-invariant and time-variant channel conditions using two different modulation schemes, and it has been demonstrated that desirable performance results can be attained particularly under time-invariant channel conditions when orthogonal transmission and BPSK modulation is used together with CSDFE implementation. It has been also shown that the performance of a CSDFE is between those of Viterbi and DFEs under time-invariant as well as time-variant channel conditions.

It has been revealed that the channel shortener equalizer is indeed a reduced complexity Viterbi equalizer with its ADIR filter memory functioning as a trade-off parameter between performance and complexity for a given channel matrix. Another important property of the proposed equalizer is that it does not need channel information.

Competing interests

The author declares that he has no competing interests.

Acknowledgements

The author is grateful to the Editor Prof. Costas Berberitis and anonymous reviewers for their useful comments.

Received: 18 July 2014 Accepted: 18 January 2015

Published online: 29 January 2015

References

- GJ Foschini, MJ Gans, On limits of wireless communications in a fading environment when using multiple antennas. *Wireless Pers. Commun.* **6**(3), 311–335 (1998). doi:10.1023/A:1008889222784
- GG Raleigh, JM Cioffi, Spatio-temporal coding for wireless communication. *IEEE Trans. Commun.* **46**(3), 357–366 (1998). doi:10.1109/26.662641
- JR GL Stüber, SW Barry, YG Mclaughlin, MA Li, TG Ingram, Pratt, Broadband MIMO-OFDM wireless communications. *Proc. IEEE.* **92**(2), 271–294 (2004). doi:10.1109/JPROC.2003.821912
- H Bölcskei, Advances in smart antennas - MIMO-OFDM wireless systems : basics, perspectives, and challenges. *IEEE Wireless Commun.* **13**(4), 31–37 (2006). doi:10.1109/MWC.2006.1678163
- PW Wolniansky, GJ Foschini, GD Golden, RA Valenzuela, in *Proceedings of URSI Int. Symp. Signals, Systems and Electronics. V-BLAST: an architecture for realizing very high data rates over the rich-scattering wireless channel* (IEEE, Pisa, 1998), pp. 295–300. doi:10.1109/ISSSE.1998.738086
- Q Liu, L Yang, in *Proceedings of the 2004 IEEE Joint Conf. 10th Asia-Pacific Conf. Commun., and 5th Int. Symp. Multi-Dimensional Mobile Commun.* A simplified method for V-BLAST detection in MIMO OFDM communication, vol. 1 (IEEE, 2004), pp. 30–33. doi:10.1109/APCC.2004.1391645
- W Bei, Z Qi, in *Proceedings of the IEEE 4th Int. Conf. Wireless Commun. Netw. and Mobile Comput. (WiCOM'08)*. A low complex V-BLAST detection algorithm for MIMO-OFDM system (IEEE, Dalian, 2008), pp. 1–4. doi:10.1109/WiCom.2008.386
- L Ma, K Dickson, J McAllister, J McCanny, QR decomposition-based matrix inversion for high performance embedded MIMO receivers. *IEEE Trans. Signal Process.* **59**(4), 1858–1867 (2011). doi:10.1109/TSP.2011.2105485
- RK Martin, CR Johnson, Adaptive equalization: transitioning from single-carrier to multicarrier systems. *IEEE Signal Process. Mag.* **22**(6), 108–122 (2005). doi:10.1109/MSP.2005.1550193
- S Haykin, Cognitive radio: brain-empowered wireless communications. *IEEE J. Sel. Areas Commun.* **23**(2), 201–220 (2005). doi:10.1109/JSAC.2004.839380
- M Dillinger, K Madani, N Alonistioti, *Software Defined Radio: Architectures, Systems and Functions*. (John Wiley and Sons, England, 2003)
- O Anjum, T Ahonen, F Garzia, J Nurmi, C Brunelli, H Berg, State of the art baseband DSP platforms for software defined radio: a survey. *EURASIP. J. Wirel. Commun. Netw.* **2011**, 5 (2011). doi:10.1186/1687-1499-2011-5
- K He, L Crockett, R Stewart, Dynamic reconfiguration technologies based on FPGA in software defined radio system. *J. Signal Process. Syst.* **69**(1), 75–85 (2012). doi:10.1007/s11265-011-0646-2
- J Im, M Cho, Y Jung, Y Jung, J Kim, A low-power and low-complexity baseband processor for MIMO-OFDM WLAN systems. *J. Signal Process. Syst.* **68**(1), 19–30 (2012). doi:10.1007/s11265-010-0570-x
- AP Vinod, E M-K Lai, A Ormondi, Special issue on signal processing for software defined radio handsets. *J. Signal Process. Syst.* **62**(2), 113–115 (2011). doi:10.1007/s11265-009-0428-2
- B Wang, RJ Ray Liu, Advances in cognitive radio networks: a survey. *IEEE J. Sel. Topics Signal Process.* **5**(1), 5–23 (2011). doi:10.1109/JSTSP.2010.2093210
- E Axell, G Leus, EG Larsson, HV Poor, Spectrum sensing for cognitive radio: state-of-the-art and recent advances. *IEEE Signal Process. Mag.* **29**(3), 101–116 (2012). doi:10.1109/MSP.2012.2183771
- H Celebi, H Arslan, Enabling location and environment awareness in cognitive radios. *Computer Commun.* **31**, 1114–1125 (2008). doi:10.1016/j.comcom.2008.01.006
- MT Ozden, Adaptive multichannel sequential lattice prediction filtering method for ARMA spectrum estimation in subbands. *EURASIP J. Adv. Signal Process.* **2013**, 9 (2013). doi:10.1186/1687-6180-2013-9
- MT Ozden, in *Proceedings of the 2014 IEEE/ION Position, Location, and Navigation Symposium (PLANS 2014)*. Adaptive multichannel sequential lattice prediction filtering method for range estimation in cognitive radios, (2014), pp. 426–433. doi:10.1109/PLANS.2014.6851400
- N Al-Dhahir, JM Cioffi, Efficiently computed reduced-parameter input-aided MMSE equalizers for ML detection: a unified approach. *IEEE Trans. Inf. Theory.* **42**(3), 903–915 (1996). doi:10.1109/18.490553
- N Al-Dhahir, FIR channel-shortening equalizers for MIMO ISI channels. *IEEE Trans. Commun.* **49**(2), 213–218 (2001). doi:10.1109/26.905867
- F Ling, JG Proakis, A generalized multichannel least squares lattice algorithm based on sequential processing stages. *IEEE Trans. Acoust. Speech Signal Process.* **32**(2), 381–389 (1984). doi:10.1109/TASSP.1984.1164325

24. J Ma, GY Li, BH Juang, Signal processing in cognitive radio. *Proc. IEEE*. **97**(5), 805–823 (2009). doi:10.1109/JPROC.2009.2015707
25. GJ Foschini, Layered space-time architecture for wireless communications in a fading environment when using multielement antennas. *Bell. Labs. Tech. J.* **1**(2), 41–59 (1996). doi:10.1002/bltj.2015
26. A Lozano, C Papadias, Layered space-time receivers for frequency-selective wireless channels. *IEEE Trans. Commun.* **50**(1), 65–73 (2002). doi:10.1109/26.975751
27. Y Jiang, J Li, WW Hager, Uniform channel decomposition for MIMO communications. *IEEE Trans. Signal Process.* **53**(11), 4283–4294 (2005). doi:10.1109/TSP.2005.857052
28. R Merched, NR Yousef, Fast techniques for computing finite-length MIMO MMSE decision feedback equalizers. *IEEE Trans. Signal Process.* **54**(2), 701–711 (2006). doi:10.1109/TSP.2005.861900
29. N Al-Dhahir, AH Sayed, The finite length multi-input multi-output MMSE-DFE. *IEEE Trans. Signal Process.* **48**(10), 2921–2936 (2000). doi:10.1109/78.869048
30. G Ginis, JM Cioffi, On the relation between V-BLAST and the GDFE. *IEEE Commun. Lett.* **5**(9), 364–366 (2001). doi:10.1109/4234.951378
31. J Wang, Y Jiang, GE Sobelman, Iterative computation of FIR MIMO MMSE-DFE with flexible complexity-performance tradeoff. *IEEE Trans. Signal Process.* **61**(9), 2394–2404 (2013). doi:10.1109/TSP.2013.2245325
32. PA Regalia, MG Bellanger, On the duality between fast QR methods and lattice methods in least squares adaptive filtering. *IEEE Trans. Signal Process.* **39**(4), 879–891 (1991). doi:10.1109/78.80910
33. F Ling, Givens rotation based least squares lattice and related algorithms. *IEEE Trans. Signal Process.* **39**(7), 1541–1551 (1991). doi:10.1109/78.134393
34. F Ling, D Manolakis, JG Proakis, A recursive modified Gram-Schmidt algorithm for least-squares estimation. *IEEE Trans. Acoust. Speech Signal Process.* **34**(4), 829–835 (1986). doi:10.1109/TASSP.1986.1164877
35. MT Ozden, AH Kayran, Adaptive multichannel decision feedback equalization for Volterra type nonlinear communication channels. *AEU - Int. J. Electron. Commun.* **62**(6), 430–437 (2008). doi:10.1016/j.aeue.2007.06.005
36. JD Provence, SC Gupta, Systolic arrays for Viterbi processing in communication systems with a time-dispersive channel. *IEEE Trans. Commun.* **36**(10), 1148–1156 (1988). doi:10.1109/26.7532
37. E Larsson, O Edfors, F Tufvesson, ML Marzetta, Massive MIMO for next generation wireless systems. *IEEE Commun. Mag.* **52**(2), 186–195 (2014). doi:10.1109/MCOM.2014.6736761
38. HC Myburgh, JC Olivier, Low complexity MLSE equalization in highly dispersive Rayleigh fading channels. *EURASIP. J. Adv. Signal Process.* **2010**(1), 874874 (2010). doi:10.1155/2010/874874
39. JG Proakis, *Digital Communications* (McGraw-Hill, New York, 2000)
40. S Haykin, *Adaptive Filter Theory*, 4th edn. (Prentice-Hall, New Jersey, 2001)
41. Y Rosa Zheng, C Xiao, Simulation models with correct statistical properties for Rayleigh fading channels. *IEEE Trans. Commun.* **51**(6), 920–928 (2003). doi:10.1109/TCOMM.2003.813259
42. W Zhang, X-G Xia, KB Letaief, Space-time/frequency coding for MIMO-OFDM in next generation broadband wireless systems. *IEEE Wireless Commun.* **14**(3), 32–43 (2007). doi:10.1109/MWC.2007.386610
43. AF Molisch, *Wireless Communications*, 2nd edn. (John Wiley and Sons, UK, 2011)

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com
