2009-09-09

New Low-Complexity Equalizers for Offset QPSK

Qiang Lei
qiang.lei@gmail.com

Michael D. Rice
mdr@byu.edu

New Low-Complexity Equalizers for Offset QPSK

Qiang Lei and Michael Rice Brigham Young University, Provo, Utah

Abstract

Reduced-complexity linear and decision-feedback equalizers for use with offset QPSK are described and analyzed. The analysis and simulations show that reduced-complexity equalizers have 1/4 the complexity and achieve this at a loss that depends on the channel. As an example, the loss for the "Proakis B" channel is 2.7 dB for the linear equalizer and 0.8 dB for the decision feedback equalizer.

I. Introduction

Offset QPSK (OQPSK) is a modulation where the quadrature (Q) component is delayed by half a symbol period relative to the in-phase (I) component. The offset eliminates phase trajectories through the origin thus producing a waveform that is better suited [relative to the traditional non-offset counterpart] for use with a non-linear RF power amplifier operating at or near full saturation [1].

When operating over an ISI-channel, equalization is used to overcome the performance degradation caused by the ISI [2]. Traditional equalizers designed for non-offset modulations are suboptimal for offset QPSK because the mean squared error criterion[1] for non-offset modulations places unnecessary restrictions on the equalizer output. Instead of attempting to minimize the error on both dimensions of the OQPSK signal at the same time, the correct design approach is to minimize the error only for the dimension of interest and alternate between the two dimensions of interest bit-by-bit as explained by [3]. The residual error on the dimension that is not of interest does not contribute to the error rate.

Few papers have been published on equalization for OQPSK primarily because the design and analysis of equalization of OQPSK is a relatively straight-forward extension of the techniques for non-offset QPSK. Of mention are papers by Tu [3], Greenstein [4], Bello [5], [6], and Craig [7].

The question to be answered in this paper is illustrated in Figure 1. The block diagram in Figure 1 (a) represents the system used in the AWGN environment. The matched filter output is sampled at 2 samples/symbol and decisions are based on the real parts of the even-indexed samples and the imaginary parts of the odd-indexed samples. (This assumes, without loss of generality, that the even-indexed bits are carried on the I component and the odd-indexed bits are carried on the Q component.) The split into even- and odd-indexed samples is conceptualized by the commutator shown in the figure.

[1]The minimum mean squared error is one of three criteria that could be used to define the optimum equalizer. The other two are the minimum ISI (leading to the zero-forcing equalizer) and the minimum BER equalizer [2]. The minimum mean squared error criteria is the most mathematically tractable of the three and appears to be the most commonly used criterion. Due to space constraints, the discussion here is limited to the mean squared error criterion.

The basic idea with linear equalizers is to place a discrete-time filter in between the matched filter and the decision device. A system designer is faced with two choices for placement of the equalizer filter: Should the equalizer filter be placed *before the commutator* as illustrated in Figure 1 (b), or *after the commutator* as illustrated in Figure 1 (c) The system in Figure 1 (b) [and its decision-feedback counterpart illustrated in Figure 4 (a)] and the structure assumed in [3] – [7] and is the system suggested by maximum likelihood (ML) analysis denoted by ML equalizer system. The system in Figure 1 (c) has clear advantages with regards to computational complexity denoted by reduced complexity (RC) equalizer system. (and these advantages are discussed in Section VI).

This paper analyzes the performance and complexity tradeoffs of the two architecture options for linear equalizers and nonlinear (decision feedback) equalizers. We show that the low-complexity version reduces the computational burden by a factor of 4 and achieves this with a performance loss that is channel dependent. As an example, we show that the performance loss for the "Proakis B" channel is 2.7 dB at a bit error rate of 10^{-4} for the linear equalizers and 0.8 dB at the same bit error rate for the decision feedback equalizers.

II. The OQPSK system model

The OQPSK system model is illustrated in Figure 2. The bit information sequence c_k which, for convenience, are represented as a member of the set $\{-1, +1\}$. These bits stream form the input to an OQPSK modulator. The complex-baseband representation of the OQPSK signal is

$$s(t) = \sum_k \left[c_{2k} g\left(t - kT_s\right) + j c_{2k+1} g\left(t - \left(k + 0.5\right) T_s\right)\right] \tag{1}$$

T_s is the symbol period, and $g(t)$ is a unit-energy pulse shape.

The OQPSK signal experiences multipath propagation that is modeled as an LTI system with impulse response $h(t)$. The received signal is

$$\begin{aligned} r(t) &= s(t) * h(t) + w(t) \\ &= \sum_k \left[c_{2k} g\left(t - kT_s\right) * h(t) + j c_{2k+1} g\left(t - \left(k + 0.5\right) T_s\right) * h(t)\right] + w(t) \end{aligned} \tag{2}$$

where $*$ is the convolution operation and $w(t)$ is the additive noise that is modeled as a complex-valued Gaussian random process with zero mean and power spectral density N_0 W/Hz.

We make two important assumptions regarding the detector:

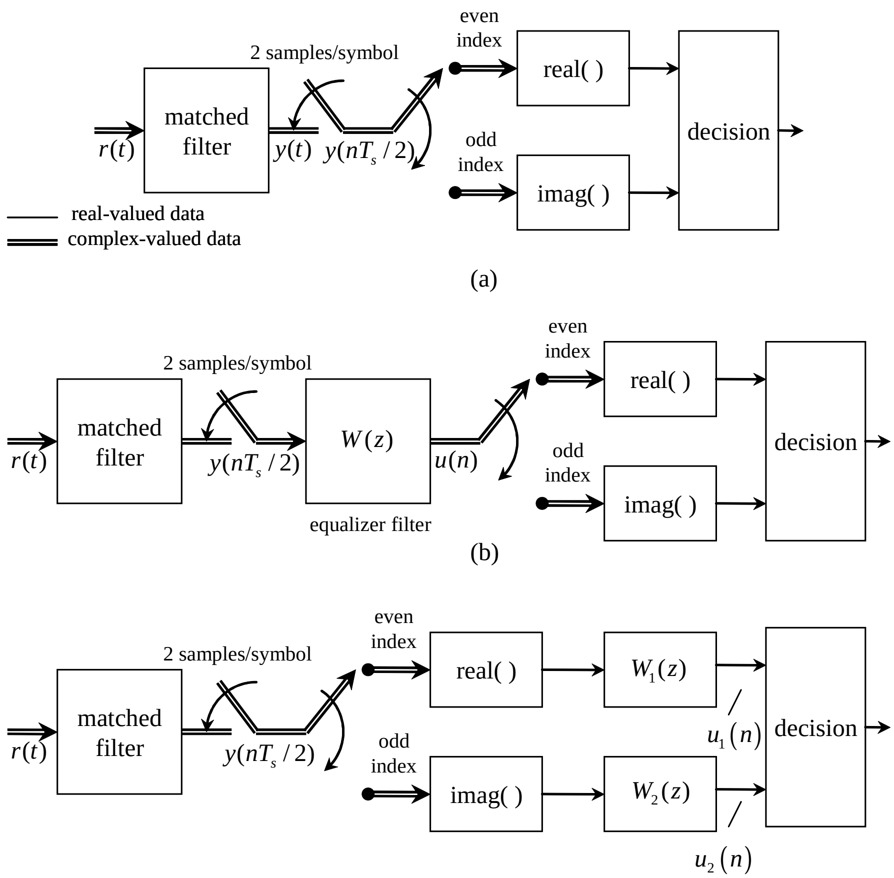

Fig. 1. OQPSK detectors: (a) detection in the AWGN environment; (b) The maximum likelihood linear MMSE equalizer; (c) The reduced-complexity linear MMSE equalizer.

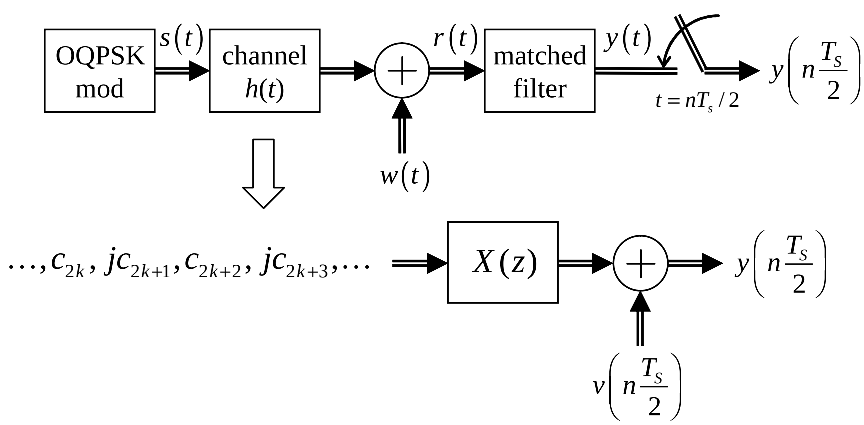

Fig. 2. The equivalent discre-time channel model.

1) The detector is equipped with a perfect carrier frequency offset synchronizer. The other two parameters, carrier phase and symbol timing, are incorporated into the channel estimate, which is discussed next.

2) The detector does not “know” the channel initially. Consequently, the matched filter is matched to the transmitted pulse shape $g(t)$ rather than the received pulse shape $g(t) * h(t)$. The presence of

pilot symbols accompanying the transmitted data is assumed. The pilot symbols are used to estimate the equivalent discrete-time channel presented by the sampled matched filter outputs. The equivalent discrete-time channel is used to compute the optimum equalizer filter coefficients as described below. For the purposes of analysis, we assume perfect channel estimates.

With these assumptions in place, the matched filter output $y(t)$, can be expressed as

$$\begin{aligned} y(t) &= \sum_k \left[c_{2k} g\left(t - kT_s\right) * h(t) * g(-t) + jc_{2k+1} g\left(t - (k+0.5)\, T_s\right) * h(t) * g(-t)\right] + w(t) * g(-t) \\ &= \sum_k \left[c_{2k} x\left(t - kT_s\right) + jc_{2k+1} x\left(t - (k+0.5)\, T_s\right)\right] + v(t) \end{aligned} \tag{3}$$

where $x(t) = g(t) * h(t) * g(-t)$ is the cascade of the transmitting filter $g(t)$, the channel $h(t)$, and the matched filter $g(-t)$ and $v(t) = w(t) * g(-t)$ is the matched filter output due to noise. The matched filter output $z(t)$ sampled at $t = nT_s/2$ to form the sequence

$$y(nT_s/2) = \sum_k \left[c_{2k} x\left((n-2k)\, T_s/2\right) + jc_{2k+1} x\left((n-2k-1)\, T_s/2\right)\right] + v(nT_s/2) \tag{4}$$

where the random variables $v(nTs/2)$ are a sequence of zero-mean complex-valued Gaussian random variables whose autocorrelation function is determined by the autocorrelation function of the pulse shape $g(t)$. For example, when $g(t)$ is unit-energy NRZ pulse shape, the auto correlation function of $v(nT_s/2)$ is

$$R_{vv}(n_0) = \begin{cases} N_0 & n_0 = 0 \\ N_0/2 & n_0 = \pm 1 \\ 0 & \text{otherwise} \end{cases} . \tag{5}$$

The sampled matched filter outputs $y(nTs/2)$ can be generated by an equivalent discrete time model as shown in Figure 2.

III. OQPSK Detection in the AWGN Environment

In the AWGN environment (i.e. $X(z) = 1$, without inter symbol interference (ISI)), the detection of OQPSK is based on the real parts of the even-indexed samples and the imaginary parts of the odd-indexed samples [1] as illustrated in Figure Figure 1 (a).

For this reason it is customary to express the sampled matched filter outputs as follows:

$$\cdots, y(mT_s), y((m+0.5)T_s), y((m+1)T_s), y((m+1.5)T_s), \cdots$$

where the integer multiples of T_s are the even sample indexes. When $g(t)$ is the unit-energy NRZ pulse shape, we have

$$y(mT_s) = j0.5c_{2m+1} + c_{2m} + j0.5c_{2m-1} + v(mT_s) \tag{6}$$

$$y((m+0.5)T_s) = 0.5c_{2m+2} + jc_{2m+1} + +0.5c_{2m} + v((m+0.5)T_s). \tag{7}$$

Writing $y(mT_s) = z_R(mT_s) + jz_I(mT_s)$ and $v(mT_s) = v_R(mT_s) + jv_I(mT_s)$, the sequence used for ML detection is[2]

$$\cdots, y_R(mT_s), y_I((m+0.5)T_s), y_R((m+1)T_s), \cdots$$

In this case, the noise variables used in detection are independent. This is shown as follows. The matched filter outputs are

$$y_R(mT_s) = c_{2m} + v_R(mT_s) \tag{8}$$

$$y_I((m+0.5)T_s) = c_{2m+1} + v_I((m+0.5)T_s) \tag{9}$$

$$y_R((m+1)T_s) = c_{2(m+1)} + v_R((m+1)T_s). \tag{10}$$

The random variables $v_R(mT_s)$ and $v_I((m+0.5)T_s)$ are independent because the underlying complex-valued Gaussian random process is assumed to be proper. The random variables $v_I(mT_s)$ and $v_I((m+1)T_s)$ are independent because the pulse shape is assumed to have no ISI — see (5).

IV. Linear MMSE Equalization for OQPSK

In the presence of ISI, because even and odd indexes are important, we will write the time index $n = 2k$ (for n even), and $n = 2k+1$ (for n odd). Using this index, the sampled matched filter output are

$$\begin{aligned} y(2kT_s/2) = &\cdots c_{2k+2}x(-2T_s/2) + jc_{2k+1}x(-T_s/2) + c_{2k}x(0) \\ &+ jc_{2k-1}x(T_s/2) + c_{2k-2}x(2T_s/2) + \cdots + v(2mT) \end{aligned} \tag{11}$$

and

$$\begin{aligned} y\left(\left(2k+1\right)T_s/2\right) = &\cdots jc_{2k+3}x\left(-2T_s/2\right) + c_{2k+2}x\left(-T_s/2\right) + jc_{2k+1}x\left(0\right) \\ &+ c_{2k}x\left(T_s/2\right) + jc_{2k-1}x\left(2T\right) + \cdots + v\left(\left(2k+1\right)T_s/2\right). \end{aligned} \tag{12}$$

Equation (11) and (12) express the equivalent discrete-time channel observed at the matched filter output. Suppose the equivalent discrete-time channel impulse response $x(nT_s/2)$ is FIR and be zero outside the interval $-L_1 \le n \le L_2$ where L_1 and L_2 are positive integers (which we assume are even for notational convenience below).

[2] The discarded matched filter outputs — $y_I(mT_s)$ and $y_R((m+0.5)T_s)$ — are used by the phase error detector in a carrier phase PLL and by the timing error detector in a symbol timing PLL [1].

A. Linear ML equalizer

First, consider the linear maximum likelihood (ML) equalizer as shown in Figure 1 (b).

$\mathbf{w}$ is $2K+1$-dimensional the equalizer complex coefficient vector as illustrate in Figure 3, $\mathbf{w} = [w(-K), w(-K+1), \cdots, w(0), \cdots, w(K-1), w(K)]^T$, At time n, we collect a length of $2K+1$ matched filter output vector $\mathbf{y}(n) = [y(n), y(n-1), \cdots, y(n-2K)]^T$ as the input to this equalizer. Without loss

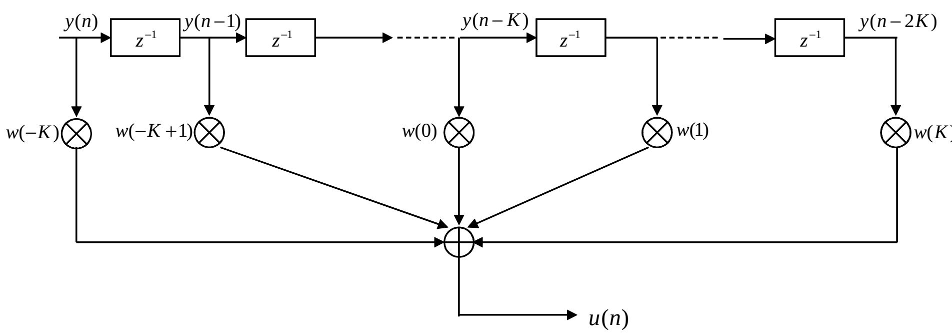

Fig. 3. The output of the linear ML equalizer.

of generality, we assume K is even, thus, the equalizer output error is defined as

$$e(n) = \begin{cases} c_{n-K} - \mathrm{Re}\{u(n)\} & n \text{ even} \\ c_{n-K} - \mathrm{Im}\{u(n)\} & n \text{ odd} \end{cases} \tag{13}$$

where $u(n) = \mathbf{w}^T\mathbf{y}(n)$. Our goal is to design $\mathbf{w}$ to minimize a modified MMSE criterion (i.e. we choose $\mathbf{w}$ to minimize $J = E\{e(n)e^*(n)\}$).

At even-indexed time n, the estimated bit after linear equalizer scheme I can be represented as

$$\hat{c}_{n-K} = \mathrm{Re}\{u(n)\} = \mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n). \tag{14}$$

where $\mathbf{w}_R$ denotes the real parts of the equalizer coefficient vector, $\mathbf{w}_I$ denotes the imaginary parts of the equalizer coefficient vector. And the real part of matched filter output vector $\mathbf{y}_R(n)$ can be represented in matrix form:

$$\mathbf{y}_R(n) = \mathrm{Re}\{\mathbf{y}(n)\} = \mathbf{X}_1\mathbf{c}(n) + \mathbf{v}_R(n) \tag{15}$$

where $\mathbf{c}(n) = [c_{n+L_1}, c_{n+L_1-1}, c_{n+L_1-2}, \cdots, c_{n-(2K+L_2)}]^T$,

and the $(2K+1)\times(L_1+L_2+2K+1)$ convolution matrix $\mathbf{X}_1$ is given by

$$\mathbf{X}_1 = \begin{bmatrix} x_R(-L_1) & -x_I(-L_1+1) & \cdots & x_R(L_2) & & & & \\ & -x_I(-L_1) & x_R(-L_1+1) & \cdots & -x_I(L_2) & & & \\ & & \ddots & & & & & \\ & & & x_R(-L_1) & -x_I(-L_1+1) & \cdots & x_R(L_2) \end{bmatrix} \tag{16}$$

$\mathbf{v}_R(n)$ is a zero mean real Gaussian vector and its autocorrelation matrix $\mathbf{P}$ is a $(2K+1)\times(2K+1)$ matrix which is defined as follows:

$$\mathbf{P} = N_0 \begin{bmatrix} 1 & 1/2 & & & \\ 1/2 & 1 & 1/2 & & \\ & 1/2 & \ddots & \ddots & \\ & & \ddots & & \end{bmatrix}. \tag{17}$$

And the imaginary part of the matched filter output vector $\mathbf{y}_I(n)$ can be represented in matrix form

$$\mathbf{y}_I(n) = \mathrm{Im}\{\mathbf{y}(n)\} = \mathbf{X}_2\mathbf{c}(n) + \mathbf{v}_I(n) \tag{18}$$

where the $(2K+1)\times(L_1+L_2+2K+1)$ convolution matrix $\mathbf{X}_2$ is given by

$$\mathbf{X}_2 = \begin{bmatrix} x_I(-L_1) & x_R(-L_1+1) & & x_I(L_2) & & & \\ & x_R(-L_1) & x_I(-L_1+1) & \cdots & x_R(L_2) & & \\ & & \ddots & & & & \\ & & x_I(-L_1) & x_R(-L_1+1) & \cdots & x_I(L_2) & \end{bmatrix} \tag{19}$$

$\mathbf{v}_I(n)$ is a zero mean real Gaussian vector and the autocorrelation matrix of $\mathbf{v}_I(n)$ is $\mathbf{P}$.

By using equation (14), we obtain the estimated error

$$e(n) = c_{n-K} - \hat{c}_{n-K} = c_{n-K} - (\mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)). \tag{20}$$

And the corresponding mean squared error, J, is all the same and it can be represented as

$$J = E\left\{e(n)e^*(n)\right\} = E\left\{(c_{n-K}-\hat{c}_{n-K})(c_{n-K}-\hat{c}_{n-K})^*\right\}. \tag{21}$$

Substitution of equation (15) and (18) into the expression for J given by equation (21), it yields the expression for mean squared error:

$$\begin{aligned} J =& 1 - \mathbf{w}_R^T\mathbf{X}_1\mathbf{\Delta} + \mathbf{w}_I^T\mathbf{X}_2\mathbf{\Delta} - \mathbf{\Delta}^H\mathbf{X}_1^H\mathbf{w}_R + \mathbf{\Delta}^H\mathbf{X}_2^H\mathbf{w}_I \\ &+ \mathbf{w}_R^T(\mathbf{X}_1\mathbf{X}_1^H + \mathbf{P})\mathbf{w}_R - \mathbf{w}_I^T\mathbf{X}_2\mathbf{X}_1^H\mathbf{w}_R - \mathbf{w}_R^T\mathbf{X}_1\mathbf{X}_2^H\mathbf{w}_I + \mathbf{w}_I^T(\mathbf{X}_2\mathbf{X}_2^H + \mathbf{P})\mathbf{w}_I \end{aligned} \tag{22}$$

where $*$ denotes the complex conjugate for scalars and H denotes the Hermitian (conjugate transpose) operation for vectors and matrices, and $\mathbf{\Delta}$ is the (L_1+L_2+2K+1)-dimensional column vector with 1 on the (L_1+K+1)-th row and zero everywhere else.

To minimize the mean square error, we need to solve the following equations

$$\frac{\partial J}{\partial \mathbf{w}_R^T} = 0 \ , \ \frac{\partial J}{\partial \mathbf{w}_I^T} = 0 \tag{23}$$

By solving equation (23), we obtain

$$(\mathbf{X}_1\mathbf{X}_1^H + \mathbf{P})\mathbf{w}_R - \mathbf{X}_1\mathbf{X}_2^H\mathbf{w}_I = \mathbf{X}_1\mathbf{\Delta} \tag{24}$$

$$-\mathbf{X}_2\mathbf{X}_1^H\mathbf{w}_R + (\mathbf{X}_2\mathbf{X}_2^H + \mathbf{P})\mathbf{w}_I = -\mathbf{X}_2\mathbf{\Delta} \tag{25}$$

Thus,

$$\begin{bmatrix} \mathbf{X}_1\mathbf{X}_1^H + \mathbf{P} & -\mathbf{X}_1\mathbf{X}_2^H \\ -\mathbf{X}_2\mathbf{X}_1^H & \mathbf{X}_2\mathbf{X}_2^H + \mathbf{P} \end{bmatrix} \begin{bmatrix} \mathbf{w}_R \\ \mathbf{w}_I \end{bmatrix} = \begin{bmatrix} \mathbf{X}_1\mathbf{\Delta} \\ -\mathbf{X}_2\mathbf{\Delta} \end{bmatrix} \tag{26}$$

Hence, the optimal filter coefficients vector can be represented as

$$\begin{bmatrix} \mathbf{w}_R \\ \mathbf{w}_I \end{bmatrix} = \begin{bmatrix} \mathbf{X}_1\mathbf{X}_1^H + \mathbf{P} & -\mathbf{X}_1\mathbf{X}_2^H \\ -\mathbf{X}_2\mathbf{X}_1^H & \mathbf{X}_2\mathbf{X}_2^H + \mathbf{P} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{X}_1\mathbf{\Delta} \\ -\mathbf{X}_2\mathbf{\Delta} \end{bmatrix} \tag{27}$$

A measure of the residual intersymbol interference and noise is obtained by evaluating the minimum value of mean squared error J, denoted by J_{min}. Apply the results of equation (24) and (25) to the expression for J given by (22), we obtain

$$\begin{aligned} J_{\text{min}} &= 1 - \begin{bmatrix} \mathbf{\Delta}^H\mathbf{X}_1^H & -\mathbf{\Delta}^H\mathbf{X}_2^H \end{bmatrix} \begin{bmatrix} \mathbf{w}_R \\ \mathbf{w}_I \end{bmatrix} \\ &= 1 - \begin{bmatrix} \mathbf{\Delta}^H\mathbf{X}_1^H & -\mathbf{\Delta}^H\mathbf{X}_2^H \end{bmatrix} \begin{bmatrix} \mathbf{X}_1\mathbf{X}_1^H + \mathbf{P} & -\mathbf{X}_1\mathbf{X}_2^H \\ -\mathbf{X}_2\mathbf{X}_1^H & \mathbf{X}_2\mathbf{X}_2^H + \mathbf{P} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{X}_1\mathbf{\Delta} \\ -\mathbf{X}_2\mathbf{\Delta} \end{bmatrix} \\ &= 1 - \begin{bmatrix} \mathbf{X}_1\mathbf{\Delta} \\ -\mathbf{X}_2\mathbf{\Delta} \end{bmatrix}^H \left(\begin{bmatrix} \mathbf{X}_1\mathbf{X}_1^H & -\mathbf{X}_1\mathbf{X}_2^H \\ -\mathbf{X}_2\mathbf{X}_1^H & \mathbf{X}_2\mathbf{X}_2^H \end{bmatrix} + \begin{bmatrix} \mathbf{P} & 0 \\ 0 & \mathbf{P} \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{X}_1\mathbf{\Delta} \\ -\mathbf{X}_2\mathbf{\Delta} \end{bmatrix}. \end{aligned} \tag{28}$$

Similarly, when time n is odd, we can repeat the derivation above which leads to the identical equations for optimal equalizer coefficients of equation (27). Hence, these filter coefficients are optimal for all the time.

B. Linear RC equalizer

Now consider the linear reduced complexity (RC) linear equalizer shown in Figure 1 (c). The two equalizer filters consists of real-valued coefficients operating on real-valued date. The filter $\mathbf{w}_1$ in the inphase channel (I-channel) consists of $2M+1$ filter coefficients indexed $w_1(-M), \cdots, w_1(0), \cdots, w_1(M)$ and the filter $\mathbf{w}_2$ in the quadrature channel (Q-channel) consists of $2M+1$ filter coefficients indexed in the same way.

At even-indexed n, in the I-channel, the input vector to the equalizer filter $\mathbf{w}_1$ at time n is $\mathbf{y}'_R(n) = [y_R(n), y_R(n-2), \cdots, y_R(n-4M)]^T$. The output of the MMSE equalizer scheme II in the I-channel can be represented as:

$$u_1(n) = \mathbf{w}_1^T\mathbf{y}'_R(n). \tag{29}$$

The equalizer output error is

$$e(n) = c_{n-2M} - u_1(n) = c_{n-2M} - \mathbf{w}_1^T\mathbf{y}'_R(n), \tag{30}$$

and the mean squared error is

$$\begin{aligned} J_1 &= E\{(c_{n-2M} - u_1(n))(c_{n-2M} - u_1(n))^*\} \\ &= E\{(c_{n-2M} - \mathbf{w}_1^T \mathbf{y}'_R(n))(c_{n-2M} - \mathbf{w}_1^T \mathbf{y}'_R(n))^*\}. \end{aligned} \tag{31}$$

And $\mathbf{y}'_R(n)$ can be represented in matrix form:

$$\mathbf{y}'_R(n) = \mathbf{X}_3 \mathbf{c}(n) + \mathbf{v}'_R(n). \tag{32}$$

$\mathbf{X}_3$ is formed from $\mathbf{X}_1$ by keeping every other row beginning with the first row where $\mathbf{X}_1$ is $(4M+1) \times (L_1 + L_2 + 4M + 1)$ version of equation (16). And $\mathbf{v}'_R(n)$ is a zero mean white Gaussian vector whose autocorrelation matrix is $N_0\mathbf{I}$ where $\mathbf{I}$ is $(2M+1) \times (2M+1)$ identity matrix.

By substituting equation (32) into the expression for J_1 given by (31) and simplify algebraic expressions, we obtain

$$J_1 = 1 - \mathbf{w}_1^T \mathbf{X}_3 \boldsymbol{\Delta} - \boldsymbol{\Delta}^T \mathbf{X}_3^T \mathbf{w}_1 + \mathbf{w}_1^T (\mathbf{X}_3 \mathbf{X}_3^T + N_0 \mathbf{I}) \mathbf{w}_1 \tag{33}$$

where $\boldsymbol{\Delta}$ is the $(L_1 + L_2 + 4M + 1)$-dimensional column vector with 1 on the $(L_1 + 2M + 1)$-th row and zeros everywhere else.

By solving $\frac{\partial J_1}{\partial \mathbf{w}_1{}^T} = 0$, we obtain

$$-\mathbf{X}_3 \boldsymbol{\Delta} + (\mathbf{X}_3 \mathbf{X}_3^T + N_0 \mathbf{I}) \mathbf{w}_1 = 0. \tag{34}$$

Hence, coefficient vector $\mathbf{w}_1$ can be calculated as follows

$$\mathbf{w}_1 = (\mathbf{X}_3 \mathbf{X}_3^T + N_0 \mathbf{I})^{-1} \mathbf{X}_3 \boldsymbol{\Delta}. \tag{35}$$

A measure of the residual intersymbol interference and noise is obtained by evaluating the minimum value of mean squared error J_1, denoted by $J_{1,\text{min}}$. By using equation (33) and (34), we have

$$J_{1,\text{min}} = 1 - \boldsymbol{\Delta}^T \mathbf{X}_3^T \mathbf{w}_1. \tag{36}$$

Then, substitution of the expression for optimal real coefficient vector $\mathbf{w}_1$ given by (35) in equation (36) yields the expression for minimum MSE in the form

$$J_{1,\text{min}} = 1 - \boldsymbol{\Delta}^T \mathbf{X}_3^T (\mathbf{X}_3 \mathbf{X}_3^T + N_0 \mathbf{I})^{-1} \mathbf{X}_3 \boldsymbol{\Delta}. \tag{37}$$

In the quadrature channel (Q-channel), the optimum MMSE equalizer coefficients $\mathbf{w}_2$ minimize the function $J_2 = E[e(n)e^*(n)]$, where $e(n) = c_{n-2M} - u_2(n)$ for odd indexed n, It can be shown that the optimum filter coefficients are

$$\mathbf{w}_2 = (\mathbf{X}_4 \mathbf{X}_4^T + \mathbf{N_0} \mathbf{I})^{-1} \mathbf{X}_4 \boldsymbol{\Delta} \tag{38}$$

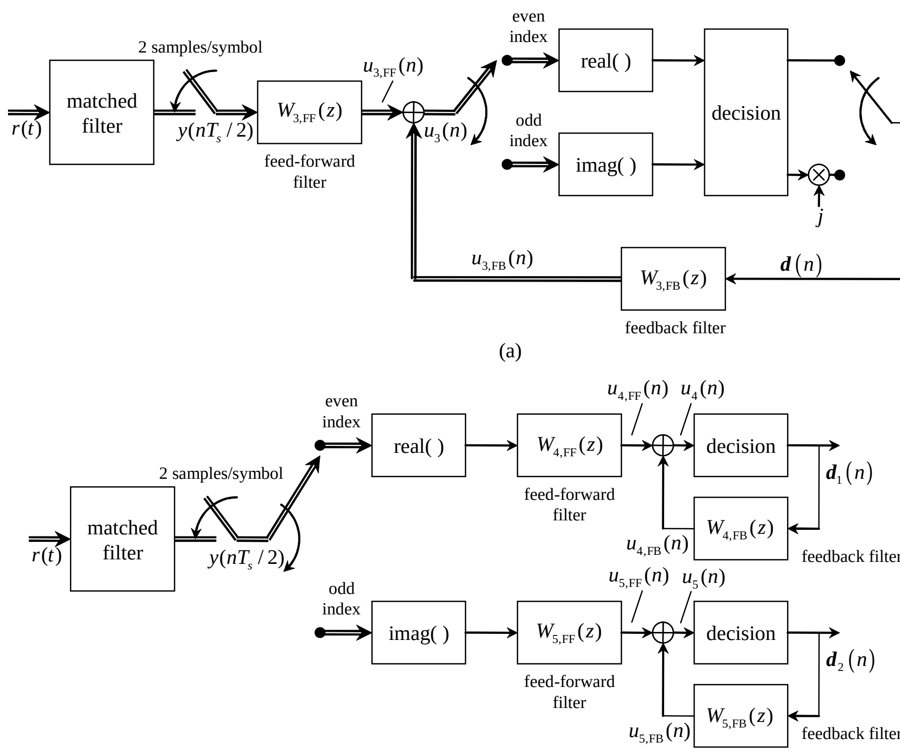

Fig. 4. Nonlinear equalizers for OQPSK: (a) The decision-feedback maximum likelihood equalizer; (b) The decision-feedback reduced-complexity equalizer.

where $\mathbf{w}_2$ is $(2M+1)\times 1$ column vector formed by the equalizer filter coefficients, $\mathbf{\Delta}$ and $\mathbf{I}$ were defined in conjunction with (33), $\mathbf{X}_4$ is formed from $\mathbf{X}_2$ by keeping taking other row from the first row where $\mathbf{X}_2$ is $(4M+1)\times(L_1+L_2+4M+1)$ version of (19), and the corresponding minimum mean squared error, $J_{2,\text{min}}$ is

$$J_{2,\text{min}} = 1 - \mathbf{\Delta}^T \mathbf{X}_4^T (\mathbf{X}_4 \mathbf{X}_4^T + N_0 \mathbf{I})^{-1} \mathbf{X}_4 \mathbf{\Delta}. \tag{39}$$

V. Non-Linear (Decision Feedback) Equalizations for OQPSK

A. DF-ML Equalizer

1) Notation: The decision feedback maximum likelihood (DF-ML) equalizer for offset QPSK is illustrated in Figure 4 (a). The matched filter output $y(t)$ is sampled at two samples/symbol. The n-th sample is filtered by FIR feed-forward filter with K_1+1 coefficients $\mathbf{w}_{3,FF}$,
where $\mathbf{w}_{3,FF} = [w_1(-K_1), w_1(-K_1+1), \cdots, w_1(0)]^T$ at time index n. The output of the feed forward filter, $u_{3,FF}(n)$ can be represented as

$$u_{3,FF}(n) = \mathbf{w}_{3,FF}^T \mathbf{y}(n) \tag{40}$$

where $\mathbf{y}(n)$ is matched filter output vector.

This output is combined with the length-K_2 feedback filter, $\mathbf{w}_{3,FB}$ to get the signal

$$u_3(n) = u_{3,FF}(n) + u_{3,FB}(n). \tag{41}$$

And the output of the feedback filter, $u_{3,FB}(n)$ is given by

$$u_{3,FB}(n) = \mathbf{w}_{3,FB}^T \mathbf{d}(n) \tag{42}$$

where $\mathbf{w}_{3,FB} = [w_2(1), w_2(2), \cdots, w_2(K_2)]^T$, $\mathbf{d}(n) = \left[\widehat{d}(n-1), \widehat{d}(n-2), \cdots, \widehat{d}(n-K_2)\right]^T$, $\widehat{d}(n)$ is the n-th bit decision given by

$$\widehat{d}(n) = \begin{cases} \text{sgn}\{\text{Re}[u_3(n)]\} & n \text{ even} \\ \text{jsgn}\{\text{Im}[u_3(n)]\} & n \text{ odd} \end{cases}. \tag{43}$$

Hence, the signal used by the decision device can be expressed as

$$u_3(n) = \mathbf{w}_{3,FF}^T \mathbf{y}(n) + \mathbf{w}_{3,FB}^T \mathbf{d}(n) \tag{44}$$

2) The closed form solution: At even indexed time n, we collect a length of K_1 received samples vector $\mathbf{y}(n)$ as the input to the feed forward filter, it can be represented as follows

$$\mathbf{y}(n) = \mathbf{X}\mathbf{s}(n) + \mathbf{v}(n) \tag{45}$$

where $\mathbf{y}(n) = [y(n), y(n-1), \cdots, y(n-K_1)]^T$, and $\mathbf{s}(n) = \left[s_{n+L_1}, s_{n+L_1-1}, \cdots, s_{n-(K_1+L_2)}\right]^T = \left[c_{n+L_1}, jc_{n+L_1-1}, \cdots, c_{n-(K_1+L_2)}\right]^T$. (Note that, $s_k = c_k$ when k even, and $s_k = jc_k$ when k odd. Without loss of generality, assume K_1, K_2 are even). The $(K_1+1) \times (K_1+L_1+L_2+1)$ convolutional matrix $\mathbf{X}$ is given by

$$\mathbf{X} = \begin{bmatrix} x(-L_1) & x(-L_1+1) & \cdots & x(L_2) & & & & \\ & x(-L_1) & x(-L_1+1) & \cdots & x(L_2) & & & \\ & & \ddots & & \ddots & & & \\ & & & x(-L_1) & x(-L_1+1) & \cdots & x(L_2) \end{bmatrix} \tag{46}$$

$\mathbf{v}(n)$ is a vector of zero mean Gaussian noise and its auto correlation matrix, $\mathbf{P}$ is $(K_1+1) \times (K_1+1)$ tri-diagonal matrix given by

$$\mathbf{P} = N_0 \begin{bmatrix} 1 & 1/2 & & \\ 1/2 & 1 & 1/2 & \\ & 1/2 & \ddots & \ddots \\ & & \ddots & \end{bmatrix}. \tag{47}$$

The equalizer makes symbol decisions $\widehat{d}(n)$ after a delay of $\tau \approx K_1$, since we need to the feedforward filter to fill up the received samples prior to decision. We assume the decisions are correct, so that

$$\begin{aligned} \mathbf{d}(n) &= [s(n-K_1-1), s(n-K_1-2), \cdots, s(n-K_1-K_2)]^T \\ &= [jc(n-K_1-1), c(n-K_1-2), \cdots, c(n-K_1-K_2)]^T. \end{aligned} \tag{48}$$

The total data used in the equalizer from the received sample and the decision feedback inputs can be expressed as

$$\widetilde{\mathbf{y}}(n) = \begin{bmatrix} \mathbf{y}(n) \\ \mathbf{d}(n) \end{bmatrix} = \begin{bmatrix} \mathbf{X} & \mathbf{0}_{(K_1+1)\times(K_2-L_2)} \\ \mathbf{0}_{K_2\times(K_1+L_1+1)} & \mathbf{T} \end{bmatrix} \widetilde{\mathbf{s}}(n) + \widetilde{\mathbf{v}}(n) \tag{49}$$

where $\mathbf{T}$ is $K_2 \times K_2$ identity matrix, $\mathbf{0}_{r\times c}$ is all zeros matrix with r rows and c columns.

$$\widetilde{\mathbf{s}}^T(n) = \left[\mathbf{s}(n), s_{n-(K_1+L_2-1)}, \cdots, s_{n-(K_1+K_2)}\right]. \tag{50}$$

$$\widetilde{\mathbf{v}}(n) = \left[v(n), v(n-1), \cdots, v(n-K_1), 0, \cdots, 0\right]^T. \tag{51}$$

To facilitate the examination of the real and imaginary components of the equalizer output, we need to also separate the received samples and filter taps into their orthogonal components. We use the subscripts R and I to denote the real and imaginary part of the variables, respectively (49) can be represented as

$$\begin{aligned} \widetilde{\mathbf{y}}_R(n) &= \widetilde{\mathbf{X}}_1\widetilde{\mathbf{c}}(n) + \widetilde{\mathbf{v}}_R(n) \\ \widetilde{\mathbf{y}}_I(n) &= \widetilde{\mathbf{X}}_2\widetilde{\mathbf{c}}(n) + \widetilde{\mathbf{v}}_I(n) \end{aligned} \tag{52}$$

where $\widetilde{\mathbf{c}}(n) = \left[c_{n+L_1}, c_{n+L_1-1}, \cdots, c_n, \cdots, c_{n-(K_1+K_2)}\right]^T$, $\widetilde{\mathbf{v}}_R(n)$, $\widetilde{\mathbf{v}}_I(n)$ are the real and imaginary components of the $\widetilde{\mathbf{v}}(n)$, their auto-correlation matrix are $\widetilde{\mathbf{P}}$, where $\widetilde{\mathbf{P}}$ is defined as

$$\widetilde{\mathbf{P}} = \begin{bmatrix} \mathbf{P} & \mathbf{0}_{(K_1+1)\times K_2} \\ \mathbf{0}_{K_2\times(K_1+1)} & \mathbf{0}_{K_2\times K_2} \end{bmatrix} \tag{53}$$

$\widetilde{\mathbf{y}}_R(n)$, $\widetilde{\mathbf{y}}_I(n)$ are defined as

$$\widetilde{\mathbf{y}}_R(n) = \begin{bmatrix} \mathbf{y}_R(n) \\ \mathbf{d}_R(n) \end{bmatrix} \tag{54}$$

$$\widetilde{\mathbf{y}}_I(n) = \begin{bmatrix} \mathbf{y}_I(n) \\ \mathbf{d}_I(n) \end{bmatrix} \tag{55}$$

$$\widetilde{\mathbf{X}}_1 = \begin{bmatrix} \mathbf{X}_1 & \mathbf{0}_{(K_1+1)\times(K_2-L_2)} \\ \mathbf{0}_{K_2\times(K_1+L_1+1)} & \mathbf{T}_1 \end{bmatrix} \tag{56}$$

$$\widetilde{\mathbf{X}}_2 = \begin{bmatrix} \mathbf{X}_2 & \mathbf{0}_{(K_1+1)\times(K_2-L_2)} \\ \mathbf{0}_{K_2\times(K_1+L_1+1)} & \mathbf{T}_2 \end{bmatrix} \tag{57}$$

where $\mathbf{X}_1$ is the $(K_1+1)\times(L_1+L_2+K_1+1)$ version of (16), $\mathbf{X}_2$ is the $(K_1+1)\times(L_1+L_2+K_1+1)$ version of (19).

$\mathbf{T}_1$, $\mathbf{T}_2$ are $K_2 \times K_2$ diagonal matrices

$$\mathbf{T}_1 = \begin{bmatrix} 0 & & & & \\ & 1 & & & \\ & & 0 & & \\ & & & 1 & \\ & & & & \ddots \end{bmatrix} \tag{58}$$

$$\mathbf{T}_2 = \begin{bmatrix} 1 & & & & \\ & 0 & & & \\ & & 1 & & \\ & & & 0 & \\ & & & & \ddots \end{bmatrix}. \tag{59}$$

The complex equalizer output can be separated into its real and imaginary components as follows:

$$\begin{aligned} u_3(n) &= \mathbf{w}_{3,FF}^T \mathbf{y}(n) + \mathbf{w}_{3,FB}^T \mathbf{d}(n) \\ &= \left(\mathbf{w}_{3,FF,R}^T + j\mathbf{w}_{3,FF,I}^T\right)(\mathbf{y}_R(n) + j\mathbf{y}_I(n)) + \left(\mathbf{w}_{3,FB,R}^T + j\mathbf{w}_{3,FB,I}^T\right)(\mathbf{d}_R(n) + j\mathbf{d}_I(n)) \\ &= \left(\mathbf{w}_{3,FF,R}^T \mathbf{y}_R(n) - \mathbf{w}_{3,FF,I}^T \mathbf{y}_I(n)\right) + j\left(\mathbf{w}_{3,FF,R}^T \mathbf{y}_I(n) + \mathbf{w}_{3,FF,I}^T \mathbf{y}_R(n)\right) \\ &\quad + \left(\mathbf{w}_{3,FB,R}^T \mathbf{d}_R(n) - \mathbf{w}_{3,FB,I}^T \mathbf{d}_I(n)\right) + j\left(\mathbf{w}_{3,FB,R}^T \mathbf{d}_I(n) + \mathbf{w}_{3,FB,I}^T \mathbf{d}_R(n)\right) \end{aligned} \tag{60}$$

Since only the real part of the equalizer output is needed at even-indexed time n. Choosing the real part of the equalizer output and rearranging the variables, the real component of (60) becomes

$$\text{Re}\{u_3(n)\} = \left(\mathbf{w}_{3,FF,R}^T \mathbf{y}_R(n) + \mathbf{w}_{3,FB,R}^T \mathbf{d}_R(n)\right) - \left(\mathbf{w}_{3,FF,I}^T \mathbf{y}_I(n) + \mathbf{w}_{3,FB,I}^T \mathbf{d}_I(n)\right) \tag{61}$$

Using the expressions defined in (54) and (55), $\text{Re}\{u_3(n)\}$ can then be rewritten as

$$\text{Re}\{u_3(n)\} = \mathbf{w}_{3,R}^T \widetilde{\mathbf{y}}_R(n) - \mathbf{w}_{3,I}^T \widetilde{\mathbf{y}}_I(n) \tag{62}$$

where

$$\mathbf{w}_{3,R} = \begin{bmatrix} \mathbf{w}_{3,FF,R} \\ \mathbf{w}_{3,FB,R} \end{bmatrix} \tag{63}$$

$$\mathbf{w}_{3,I} = \begin{bmatrix} \mathbf{w}_{3,FF,I} \\ \mathbf{w}_{3,FB,I} \end{bmatrix} \tag{64}$$

The equalizer makes a decision after a delay of $\tau = K_1$ (i. e. we make a decision at time n corresponding to the transmitted symbol of time $n - K_1$). The error between the output of the equalizer at time n and the transmitted symbol at time $n - K_1$ is

$$e(n) = \text{Re}\{s(n - K_1) - u_3(n)\} = c(n - K_1) - \text{Re}\{u_3(n)\} \tag{65}$$

Substitution of equation (62) into the expression for $e(n)$ given by (65), it yields

$$\begin{aligned} e(n) &= c_{n-K_1} - \left(\mathbf{w}_{3,R}^T \widetilde{\mathbf{y}}_R(n) - \mathbf{w}_{3,I}^T \widetilde{\mathbf{y}}_I(n)\right) \\ &= c_{n-K_1} - \left[\mathbf{w}_{3,R}^T\left(\widetilde{\mathbf{X}}_1 \widetilde{c}(n) + \widetilde{\mathbf{v}}_R(n)\right) - \mathbf{w}_{3,I}^T\left(\widetilde{\mathbf{X}}_2 \widetilde{c}(n) + \widetilde{\mathbf{v}}_I(n)\right)\right] \\ &= c_{n-K_1} - \mathbf{w}_{3,R}^T \widetilde{\mathbf{X}}_1 \widetilde{c}(n) - \mathbf{w}_{3,R}^T \widetilde{\mathbf{v}}_R(n) - \mathbf{w}_{3,I}^T \widetilde{\mathbf{X}}_2 \widetilde{c}(n) - \mathbf{w}_{3,I}^T \widetilde{\mathbf{v}}_I(n) \end{aligned} \tag{66}$$

Assuming that the received noise is zero mean and is uncorrelated with the input data, after algebraic simplifications, the expectation of the squared error J_3 becomes

$$\begin{aligned} J_3 &= E\{e(n)\,e^*(n)\} \\ &= 1 - \mathbf{w}_{3,R}^T\widetilde{\mathbf{X}}_1\zeta + \mathbf{w}_{3,I}^T\widetilde{\mathbf{X}}_2\zeta - \zeta^H\widetilde{\mathbf{X}}_1^H\mathbf{w}_{3,R} + \zeta^H\widetilde{\mathbf{X}}_2^H\mathbf{w}_{3,I} \\ &\quad + \mathbf{w}_{3,R}^T(\widetilde{\mathbf{X}}_1\widetilde{\mathbf{X}}_1^H + \widetilde{\mathbf{P}})\mathbf{w}_{3,R} - \mathbf{w}_{3,I}^T\widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_1^H\mathbf{w}_{3,R} - \mathbf{w}_{3,R}^T\widetilde{\mathbf{X}}_1\widetilde{\mathbf{X}}_2^H\mathbf{w}_{3,I} + \mathbf{w}_{3,I}^T(\widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_2^H + \widetilde{\mathbf{P}})\mathbf{w}_{3,I} \end{aligned} \tag{67}$$

where $*$ denotes the complex conjugate for scalars and H denotes the Hermitian (conjugate transpose) operation for vectors and matrices, and ζ is a $(L_1+1+K_1+K_2)$ dimensional column vector with 1 on the (K_1+L_1+1)-th row and zeros every where else.

To find a minimized solution for the expectation of the squared error, the gradient of J_3 is taken with respect to $\mathbf{w}_{3,R}^T$ and $\mathbf{w}_{3,I}^T$ and is set to zero as shown:

$$\frac{\partial J_3}{\partial \mathbf{w}_{3,R}^T} = 0\ ,\ \ \frac{\partial J_3}{\partial \mathbf{w}_{3,I}^T} = 0 \tag{68}$$

Solving these two equations leads to the following equations

$$(\widetilde{\mathbf{X}}_1\widetilde{\mathbf{X}}_1^H + \widetilde{\mathbf{P}})\mathbf{w}_{3,R} - \widetilde{\mathbf{X}}_1\widetilde{\mathbf{X}}_2^H\mathbf{w}_{3,I} = \widetilde{\mathbf{X}}_1\zeta \tag{69}$$

$$-\widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_1^H\mathbf{w}_{3,R} + (\widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_2^H + \widetilde{\mathbf{P}})\mathbf{w}_{3,I} = -\widetilde{\mathbf{X}}_2\zeta \tag{70}$$

Thus,

$$\begin{bmatrix} \widetilde{\mathbf{X}}_1\widetilde{\mathbf{X}}_1^H + \widetilde{\mathbf{P}} & -\widetilde{\mathbf{X}}_1\widetilde{\mathbf{X}}_2^H \\ -\widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_1^H & \widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_2^H + \widetilde{\mathbf{P}} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{3,R} \\ \mathbf{w}_{3,I} \end{bmatrix} = \begin{bmatrix} \widetilde{\mathbf{X}}_1\zeta \\ -\widetilde{\mathbf{X}}_2\zeta \end{bmatrix} \tag{71}$$

Hence, the optimal coefficients of the DF-MMSE equalizer can be written as

$$\begin{bmatrix} \mathbf{w}_{3,R} \\ \mathbf{w}_{3,I} \end{bmatrix} = \begin{bmatrix} \widetilde{\widetilde{\mathbf{X}}_1}\widetilde{\mathbf{X}}_1^H + \widetilde{\mathbf{P}} & -\widetilde{\mathbf{X}}_1\widetilde{\mathbf{X}}_2^H \\ -\widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_1^H & \widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_2^H + \widetilde{\mathbf{P}} \end{bmatrix}^{-1} \begin{bmatrix} \widetilde{\mathbf{X}}_1\zeta \\ -\widetilde{\mathbf{X}}_2\zeta \end{bmatrix} \tag{72}$$

A measure of the residual intersymbol interference and noise is obtained by evaluating the minimum value of mean squared error J_3, denoted by $J_{3,\min}$. Apply the results of equation (69) and (70) to the expression for J_3 given by (67), it yields

$$\begin{aligned} J_{3,\min} &= 1 - \begin{bmatrix} \zeta^H\widetilde{\mathbf{X}}_1^H & -\zeta^H\widetilde{\mathbf{X}}_2^H \end{bmatrix} \begin{bmatrix} \mathbf{w}_{3,R} \\ \mathbf{w}_{3,I} \end{bmatrix} \\ &= 1 - \begin{bmatrix} \zeta^H\widetilde{\mathbf{X}}_1^H & -\zeta^H\widetilde{\mathbf{X}}_2^H \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{X}}_1\widetilde{\mathbf{X}}_1^H + \widetilde{\mathbf{P}} & -\widetilde{\mathbf{X}}_1\widetilde{\mathbf{X}}_2^H \\ -\widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_1^H & \widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_2^H + \widetilde{\mathbf{P}} \end{bmatrix}^{-1} \begin{bmatrix} \widetilde{\mathbf{X}}_1\zeta \\ -\widetilde{\mathbf{X}}_2\zeta \end{bmatrix} \\ &= 1 - \begin{bmatrix} \widetilde{\mathbf{X}}_1\zeta \\ -\widetilde{\mathbf{X}}_2\zeta \end{bmatrix}^H \left(\begin{bmatrix} \widetilde{\mathbf{X}}_1\widetilde{\mathbf{X}}_1^H & -\widetilde{\mathbf{X}}_1\widetilde{\mathbf{X}}_2^H \\ -\widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_1^H & \widetilde{\mathbf{X}}_2\widetilde{\mathbf{X}}_2^H \end{bmatrix} + \begin{bmatrix} \widetilde{\mathbf{P}} & 0 \\ 0 & \widetilde{\mathbf{P}} \end{bmatrix} \right)^{-1} \begin{bmatrix} \widetilde{\mathbf{X}}_1\zeta \\ -\widetilde{\mathbf{X}}_2\zeta \end{bmatrix}. \end{aligned} \tag{73}$$

This derivation can be repeated for time n odd, we find that it leads to the identical equations for optimal equalizer coefficients of equation (72). Thus, these filter coefficients are optimal for all the time.

B. DF-RC equalizer

Now consider the decision feedback reduced-complexity (DF-RC) equalizer as illustrated in Figure 4 (b). In the inphase channel, the feed-forward filter $\mathbf{w}_{4,FF}$ are non-causal filter whose coefficients are indexed $w_{4,FF}(-M_1), \cdots, w_{4,FF}(0)$. The feedback filter $\mathbf{w}_{4,FB}$ has coefficients indexed $w_{4,FB}(1), \cdots, w_{4,FF}(M_2)$. In the Quadrature channel, the feed-forward filter $\mathbf{w}_{5,FF}$ consists of $M_1 + 1$ coefficients and the feedback filter $\mathbf{w}_{5,FB}$ have M_2 coefficients, both of which are indexed the same way as in the inphase channel.

First, derive the closed form solution for $\mathbf{w}_{4,FF}$, $\mathbf{w}_{4,FB}$ in the inphase channel. At even-indexed time n, the input vector to the feed-forward filter is $\mathbf{y}_{R,e}(n)$, $\mathbf{y}_{R,e}(n) = [y_R(n), y_R(n-2), \ldots, y_R(n-2M_1)]^T$, which can be represented as

$$\mathbf{y}_{R,e}(n) = \mathbf{X}_{1,e}\mathbf{c}(n) + \mathbf{v}_{R,e}(n) \tag{74}$$

where $\mathbf{c}(n)$ is $\left[c_{n+L_1}, c_{n+L_1-1}, \ldots, c_{n-(2M_1+L_2)}\right]^T$.

$\mathbf{X}_{1,e}$ is form from $\mathbf{X}_1$ by taking every other row beginning with top row, where $\mathbf{X}_1$ is $(2M_1+1) \times (2M_1+L_1+L_2+1)$ version of (16). $\mathbf{v}_{R,e}(n)$ is a zero mean white Gaussian vector and its auto-correlation matrix is $N_0\mathbf{I}_{(M_1+1)\times(M_1+1)}$.

The equalizer in the I-channel makes symbol decision after a delay of $\tau = 2M_1$, we assume the decisions are correct, so that

$$\mathbf{d}_1(n) = \left[c(n-2M_1-2), c(n-2M_1-4), \ldots, (n-2M_1-2M_2)\right]^T. \tag{75}$$

The total filter input data used in the I-channel may be written as

$$\widetilde{\mathbf{y}}_{R,e}(n) = \begin{bmatrix} \mathbf{y}_{R,e}(n) \\ \mathbf{d}_1(n) \end{bmatrix} = \mathbf{X}_4\widetilde{\mathbf{c}}(n) + \widetilde{\mathbf{v}}_{R,e}(n) \tag{76}$$

where

$$\mathbf{X}_4 = \begin{bmatrix} \mathbf{X}_{1,e} & \\ & \mathbf{I}_{M_2\times M_2} \end{bmatrix}, \tag{77}$$

$\widetilde{\mathbf{c}}(n) = \left[c_{n+L_1}, c_{n+L_1-1}, \cdots, c_n, \cdots, c_{n-(2M_1+2M_2)}\right]^T$. $\widetilde{\mathbf{v}}_{R,e}(n) = [\mathbf{v}_{R,e}(n); \mathbf{0}_{M_2\times 1}]$, its auto-correlation matrix is $\mathbf{Q}$, which is given by

$$\mathbf{Q} = \begin{bmatrix} N_0\mathbf{I}_{(M_1+1)\times(M_1+1)} & \\ & \mathbf{0}_{M_2\times M_2} \end{bmatrix}. \tag{78}$$

The signal used by the decision device is given by

$$\begin{aligned} u_4(n) &= \mathbf{w}_{4,FF}^T\mathbf{y}_{R,e}(n) + \mathbf{w}_{4,FB}^T\mathbf{d}_1(n) \\ &= \mathbf{w}_4\widetilde{\mathbf{y}}_{R,e}(n) \end{aligned} \tag{79}$$

where

$$\mathbf{w}_4 = \begin{bmatrix} \mathbf{w}_{4,FF} \\ \mathbf{w}_{4,FB} \end{bmatrix}. \tag{80}$$

The equalizer makes a decision after a delay of $\tau = 2M_1$. The error between the output of the equalizer at time n and the transmitted signal at time $n - 2M_1$ is

$$\begin{aligned} e(n) &= c_{n-2M_1} - u_4(n) \\ &= c_{n-2M_1} - \mathbf{w}_4 \widetilde{\mathbf{y}}_{R,e}(n). \end{aligned} \tag{81}$$

The mean squared error is

$$\begin{aligned} J_4 &= E\left\{e\left(n\right)e^*\left(n\right)\right\} \\ &= E\left\{\left(c_{n-2M_1} - \mathbf{w}_4 \widetilde{\mathbf{y}}_{R,e}\left(n\right)\right)\left(c_{n-2M_1} - \mathbf{w}_4 \widetilde{\mathbf{y}}_{R,e}\left(n\right)\right)^*\right\}. \end{aligned} \tag{82}$$

By substituting equation (76) into the equation for J_4 given by (82), we have

$$J_4 = E\left\{\left[c_{n-2M_1} - \mathbf{w}_4\left(\mathbf{X}_4 \widetilde{\mathbf{c}}\left(n\right) + \widetilde{\mathbf{v}}_{R,e}\left(n\right)\right)\right]\left[c_{n-2M_1} - \mathbf{w}_4\left(\mathbf{X}_4 \widetilde{\mathbf{c}}\left(n\right) + \widetilde{\mathbf{v}}_{R,e}\left(n\right)\right)\right]^*\right\}. \tag{83}$$

Assume $\widetilde{\mathbf{v}}_{R,e}(n)$ is zero mean and is uncorrelated with the input data, after algebraic simplifications, the mean square error J_4 becomes

$$J_4 = 1 - \mathbf{w}_4^T \mathbf{X}_4 \zeta - \zeta^T \mathbf{X}_4^T \mathbf{w}_4 + \mathbf{w}_4^T \left(\mathbf{X}_4 \mathbf{X}_4^T + \mathbf{Q}\right) \mathbf{w}_4 \tag{84}$$

where T denotes the transpose operation for vectors and matrices, and ζ is a $(2M_1 + L_1 + L_2 + 1) \times 1$ column vector with 1 on the $(2M_1 + L_1 + 1)$-th row, and zeros everywhere else.

By solving $\frac{\partial J_4}{\partial \mathbf{w}_4^T} = 0$, we obtain

$$-\mathbf{X}_4 \zeta + \left(\mathbf{X}_4 \mathbf{X}_4^T + \mathbf{Q}\right) \mathbf{w}_4 = 0. \tag{85}$$

Hence, coefficient vector $\mathbf{w}_4$ can be calculated as follows

$$\mathbf{w}_4 = \left(\mathbf{X}_4 \mathbf{X}_4^T + \mathbf{Q}\right)^{-1} \mathbf{X}_4 \zeta. \tag{86}$$

A measure of the residual intersymbol interference and noise is obtained by evaluating the minimum value of mean squared error J_4, denoted by $J_{4,\text{min}}$. By using equation (84) and (85), we have

$$J_{4,\text{min}} = 1 - \zeta^T \mathbf{X}_4^T \mathbf{w}_4. \tag{87}$$

Then, substitution of the expression for optimal real coefficient vector $\mathbf{w}_4$ given by (86) in equation (87) yields the expression for minimum MSE in the form

$$J_{4,\text{min}} = 1 - \zeta^T \mathbf{X}_4^T \left(\mathbf{X}_4 \mathbf{X}_4^T + \mathbf{Q}\right)^{-1} \mathbf{X}_4 \zeta. \tag{88}$$

Similarly, in the quadrature channel, the filter coefficients $\mathbf{w}_{5,FF}$, $\mathbf{w}_{5,FB}$ minimize $J_5 = E\{e(n)e^*(n)\}$ where $e(n) = c_{n-2M_1} - u_{5,FF}(n) - u_{5,FB}(n)$ for odd indexed n. It can be shown that

$$\begin{aligned} \mathbf{w}_5 &= \left[\begin{array}{c} \mathbf{w}_{5,FF} \\ \mathbf{w}_{5,FB} \end{array} \right] \\ &= (\mathbf{X}_5\mathbf{X}_5^T + \mathbf{Q})^{-1}\mathbf{X}_5\zeta \end{aligned} \tag{89}$$

where

$$\mathbf{X}_5 = \left[\begin{array}{cc} \mathbf{X}_{2,e} & \\ & \mathbf{I}_{M_2\times M_2} \end{array} \right], \tag{90}$$

ζ and $\mathbf{Q}$ were defined in conjunction with (84). $\mathbf{X}_{2,e}$ is form from $\mathbf{X}_2$ by taking every other row begining with top row, where $\mathbf{X}_2$ is $(2M_1+1)\times(2M_1+L_1+L_2+1)$ version of equation (19).

The minimum mean-squared error $J_{5,min}$ at the input of the decision device in the quadrature channel is

$$J_{5,\text{min}} = 1 - \zeta^T\mathbf{X}_5^T(\mathbf{X}_5\mathbf{X}_5^T + \mathbf{Q})^{-1}\mathbf{X}_5\zeta. \tag{91}$$

VI. Numerical Results

In this section,we evaluate the performance of both the linear and non-linear equalizers over the Proakis "B" discrete-time equivalent channel which are described in Chapter 9 of [2]. The Proakis "B" is defined in Figure 9.4-5 of [2] which is

$$X_1(z) = 0.407z + 0.815 + 0.407z^{-1}. \tag{92}$$

The subscript "1" is used to denote the fact that the channel model was derived from symbol-spaced samples of the matched filter output (i.e., the matched filter is sampled at 1 sample/symbol). This is sufficient for non-offset modulations. The equivalent discrete-time channel for offset QPSK is obtained by sampling the matched filter output at 2 samples/symbol. It is necessary to generate a version of $X_1(z)$ that operates at 2 samples/symbol. This requires the generation of data where there is no data. However with some reasonable assumptions and sound engineering judgement, a representative channel model can be produced. We assume that the receiver filter smooths the raw channel impulse response sufficiently to create a continuous composite impulse response. Our approach is summarized as follows.

1) Upsample the $x_1(kT_s)$ by 2 and apply a linear interpolation filter to produce $\tilde{x}_2(nT_s/2)$.
2) Truncate $\tilde{x}_2(nT_s/2)$ so that the time support of the truncated sequence equals that of $x_1(kT_s)$. This preserves the ISI span induced by the channel. Call the truncated sequence $x_2(nT_s/2)$.
3) Generate a unit-energy version of $x_2(nT_s/2)$ called $x(nT_s/2)$. That is, $x(nT_s/2) = Cx_2(nT_s/2)$ where $C = 1/\sqrt{\sum_n x_2^2(nT_s/2)}$.

4) Define $X(z) = \sum_n x(nT_s/2)z^{-n}$ and use $X(z)$ as the channel.

The upsampled version of the Proakis B channel (92) is

$$X(z) = 0.308z^2 + 0.463z^1 + 0.618 + 0.463z^{-1} + 0.308z^{-2}. \tag{93}$$

For each equalizer, we evaluate the signal-to-noise ratio (SNR) presented to the input of the decision device, simulate the bit-error-rate performance, and quantify the computational complexity using the number of real-valued multiplications and real-valued additions per symbol. For the bit error rate simulations, we used $K = 8$, $M = 4$ for the linear equalizers and $K_1 = K_2 = 8$ and $M_1 = M_2 = 4$ for the decision feedback equalizers.

A. Linear Equalizers

First, consider the linear equalizers of Figures 1 (b) and (c). The minimum achievable mean-squared error measures the residual ISI and noise at the input to the decision device. For linear ML equalizer, the minimum achievable mean-squared error J_{min} is defined in equation (28). The corresponding SNR presented to the input to the decision device is

$$\text{SNR} = \frac{1 - J_{\text{min}}}{J_{\text{min}}}. \tag{94}$$

For the reduced-complexity linear equalizer of Figure 1 (c), the minimum mean-squared errors $J_{1,\text{min}}$, $J_{2,\text{min}}$ at the inputs of the two decision devices can be computed using (37) and (39) respectively. The corresponding SNR presented to the inputs to the two decision devices is computed using (94) by $J_{1,\text{min}}$ or $J_{2,\text{min}}$ in place of J_{min}. The SNRs for $E_b/N_0 = 10$ dB and 20 dB over the channel (93) are listed in Table I. (Note that $J_{1,\text{min}} = J_{2,\text{min}}$.) The simulated bit error rate is shown in Figure 5. The linear ML equalizer is 2.7 dB better than the linear reduced-complexity equalizer at BER $= 10^{-4}$. This difference compares well with the SNR difference ($8.4 - 5.5 = 2.9$ dB) for $E_b/N_0 = 20$ dB listed in Table I.

The ML linear equalizer consists of a length-$(2K+1)$ filter with complex-valued coefficients operating on complex-valued data. Each output sample requires $8K + 4$ real-valued multiplications and $8K + 2$ real-valued additions. The reduced-complexity equalizer consists of two length-$(2M + 1)$ equalizers with real-valued coefficients operating on real-valued data at half the sample rate of ML equalizer. Each output sample pair requires $4M + 2$ real-valued multiplications $4M$ real-valued additions. The computational burden for the two linear equalizers is summarized in Table II. Because the reduced-complexity equalizer operates at one-half the sample rate, the lengths of the filters in the reduced complexity equalizer need to be one-half the length of the filter for the ML equalizer achieve the same temporal span. Using $2M = K$, we see that the computation burden of the reduced-complexity linear equalizer is approximately 1/4 that of the ML linear equalizer. These savings are achieved at a cost of a 2.7 dB loss for the "Proakis B" channel.

B. Non-Linear (Decision Feedback) Equalizers

Now consider the decision feedback equalizers of Figure 4. The minimum mean squared error of the ML DF equalizer, $J_{3,\text{min}}$ can be computed using equation (73).

The minimum mean squared errors of the reduced-complexity DF equalizer $J_{4,\text{min}}$,$J_{5,\text{min}}$ are defined in (88) and (91) respectively. From which the SNRs at the inputs to the two decision blocks can be computed as described above. The SNRs for $E_b/N_0 = 10$ dB and 20 dB over the channel (93) are listed in Table I. (Note that $J_{4,\text{min}} = J_{5,\text{min}}$.) The simulated bit error rate is shown in Figure 5. The ML DF equalizer is 0.8 dB better than the reduced-complexity DF equalizer at BER $= 10^{-4}$. This difference is overstated by the SNR difference ($12.8 - 10.4 = 2.4$ dB) for $E_b/N_0 = 20$ dB listed in Table I.

The ML DF equalizer consists of a length-$K_1 + 1$ feed-forward filter with complex-valued coefficients operating on complex-valued data. Each output sample requires $4K_1 + 4$ real-valued multiplications and $4K_1 + 2$ real-valued additions. The feedback filter consists of K_2 complex-valued coefficients operating on data drawn from either $\{-1, +1\}$ or $\{-j, +j\}$ in alternation. Consequently, the feedback filter requires no multiplications and $K_2 - 1$ complex-valued additions ($2K_2 - 2$ real-valued additions). The total computational burden is summarized in Table II.

The reduced-complexity DF consists of two length-$M_1 + 1$ feed-forward equalizers with real-valued coefficients operating on real-valued data and two length-M_2 feedback equalizers operating on data in the set $\{-1, +1\}$. Both filters operate at half the sample rate of their counterparts in the ML DF equalizer. Each feed-forward filter requires $M_1 + 1$ real-valued multiplications and M_1 real-valued additions. Each feedback filter requires $M_2 - 1$ real-valued additions. The computational burden for the reduced complexity DF equalizer is summarized in Table II. Because the reduced-complexity equalizer operates at one-half the sample rate, the filter lengths for the reduced-complexity DF equalizer need to be one-half the lengths of the corresponding filters in the ML DF equalizer. Using $2M_1 = K_1$ and $2M_2 = K_2$, we see that the computation burden of the reduced-complexity DF equalizer is approximately 1/4 that of the ML DF equalizer. These savings are achieved at a cost of a 0.8 dB loss for the "Proakis B" channel.

TABLE I

APPROXIMATE SNRS PRESENTED TO THE DECISION DEVICES.

Equalizer	E_b/N_0 10 dB	E_b/N_0 20 dB
Linear ML	3.8	8.4
Linear RC	1.9	5.5
DF ML	4.8	3.5
DF RC	12.8	10.4

Key: ML = maximum likelihood, RC = reduced complexity, DF = decision feedback.

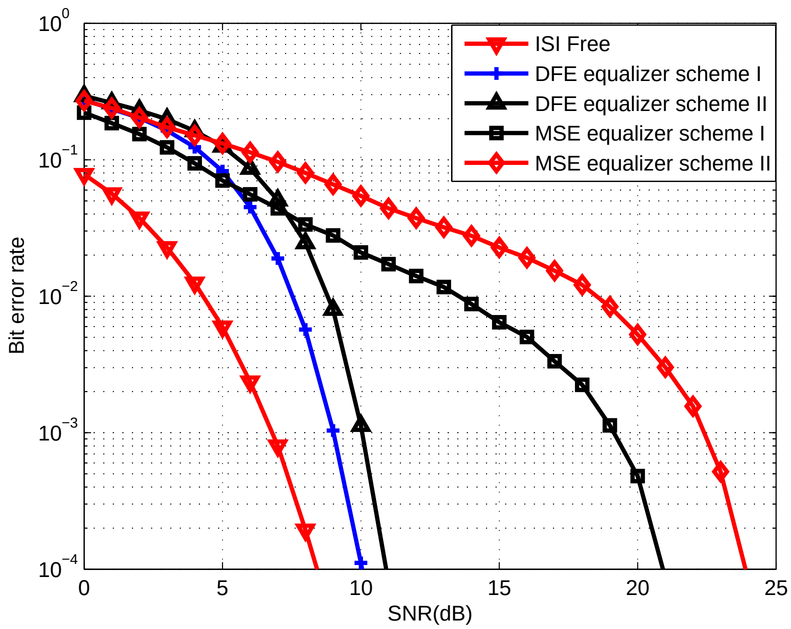

Fig. 5. Simulated bit error rate (BER) performance for OQPSK over the channel (93) using the four equalizers described in this paper.

TABLE II

COMPLEXITY COMPARISON FOR THE FOUR EQUALIZERS.

Equalizer	R-mults	R-adds
Linear ML	$8K+4$	$8K+2$
Linear RC	$4M+2$	$4M$
DF ML	$4K_1+4$	$4K_1+2K_2$
DF RC	$2M_1+2$	$2M_1+2M_2-2$

Key: R-mults = real-valued multiplications, R-adds = real-valued additions, ML = maximum likelihood, RC = reduced-complexity, DF = decision feedback.

VII. Conclusions

We have answered the question posed in the introduction and showed that the reduced-complexity equalizers offer an excellent performance/complexity trade-off to the system designer. The reduced-complexity equalizers have 1/4 the computational burden of the maximum likelihood equalizers. The bit error rate performance cost is channel dependent. For the "Proakis B" channel, this cost is 2.7 dB for the linear equalizer and 0.8 dB for the decision feedback equalizer.

VIII. Appendix

DERIVATION OF EQUATION (22)

$$\begin{aligned}
J &= E\left\{e(n)e(n)^*\right\} \\
&= E\left\{\left[c_{n-K} - \widehat{c}_{n-K}\right]\left[c_{n-K} - \widehat{c}_{n-K}\right]^*\right\} \\
&= E\left\{\left[c_{n-K} - \left(\mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)\right)\right]\left[c_{n-K} - \left(\mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)\right)\right]^*\right\} \\
&= \underbrace{E\{c_{n-K}c_{n-K}^*\}}_{\alpha} - \underbrace{E\{c_{n-K}^*\left(\mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)\right)\}}_{\beta} - \underbrace{E\{c_{n-K}\left(\mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)\right)^*\}}_{\gamma} \\
&\quad + \underbrace{E\{\left(\mathbf{w}_y^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)\right)\left(\mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)\right)^*\}}_{\delta}
\end{aligned} \tag{95}$$

$$\alpha = E\left\{c_{n-K}c_{n-K}^*\right\} = 1 \tag{96}$$

$$\begin{aligned}
\beta &= E\left\{c_{n-K}^*\left(\mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)\right)\right\} \\
&= E\left\{c_{n-K}^*\left[\mathbf{w}_R^T\left(\mathbf{X}_1\mathbf{c}(n) + \mathbf{v}_R(n)\right) - \mathbf{w}_I^T\left(\mathbf{X}_2\mathbf{c}(n) + \mathbf{v}_I(n)\right)\right]\right\} \\
&= E\left\{c_{n-K}^*\mathbf{w}_R^T\mathbf{X}_1\mathbf{c}(n)\right\} - E\left\{c_{n-K}^*\mathbf{w}_I^T\mathbf{X}_2\mathbf{c}(n)\right\} + E\left\{c_{n-K}^*\mathbf{w}_R^T\mathbf{v}_R(n)\right\} - E\left\{c_{n-K}^*\mathbf{w}_I^T\mathbf{v}_I(n)\right\} \\
&= \mathbf{w}_R^T\mathbf{X}_1\boldsymbol{\Delta} - \mathbf{w}_I^T\mathbf{X}_2\boldsymbol{\Delta}
\end{aligned} \tag{97}$$

where $\boldsymbol{\Delta}$ is column vector with 1 on the $(K + L_1 + 1)$-th row.

$$\boldsymbol{\Delta} = \left[0, 0, \cdots, 1, \cdots, 0, 0\right]^T. \tag{98}$$

$$\begin{aligned}
\gamma &= E\left\{c_{n-K}\left(\mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)\right)^*\right\} \\
&= E\left\{c_{n-K}\left[\mathbf{w}_R^T\left(\mathbf{X}_1\mathbf{c}(n) + \mathbf{v}_R(n)\right) - \mathbf{w}_I^T\left(\mathbf{X}_2\mathbf{c}(n) + \mathbf{v}_I(n)\right)\right]^*\right\} \\
&= E\left\{c_{n-K}\left[\left(\mathbf{c}(n)^H\mathbf{X}_1^H + \mathbf{v}_R(n)^H\right)\mathbf{w}_R^* - \left(\mathbf{c}(n)^H\mathbf{X}_2^H + \mathbf{v}_I(n)^H\right)\mathbf{w}_I^*\right]\right\} \\
&= E\left\{c_{n-K}\mathbf{c}(n)^H\mathbf{X}_1^H\mathbf{w}_R^*\right\} - E\left\{c_{n-K}\mathbf{c}(n)^H\mathbf{X}_2^H\mathbf{w}_I^*\right\} \\
&= \boldsymbol{\Delta}^H\mathbf{X}_1^H\mathbf{w}_R^* - \boldsymbol{\Delta}^H\mathbf{X}_2^H\mathbf{w}_I^*
\end{aligned} \tag{99}$$

$$\delta = E\left\{\left(\mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)\right)\left(\mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)\right)^*\right\}$$
$$= E\left\{\left(\mathbf{w}_R^T\mathbf{y}_R(n) - \mathbf{w}_I^T\mathbf{y}_I(n)\right)\left(\mathbf{y}_R^H(n)\mathbf{w}_R^* - \mathbf{y}_I^H(n)\mathbf{w}_I^*\right)\right\}$$
$$= \underbrace{E\left(\mathbf{w}_R^T\mathbf{y}_R(n)\mathbf{y}_R^H(n)\mathbf{w}_R^*\right)}_{a} - \underbrace{E\left(\mathbf{w}_I^T\mathbf{y}_I(n)\mathbf{y}_R^H(n)\mathbf{w}_R^*\right)}_{b}$$
$$- \underbrace{E\left(\mathbf{w}_R^T\mathbf{y}_R(n)\mathbf{y}_I^H(n)\mathbf{w}_I^*\right)}_{c} + \underbrace{E\left(\mathbf{w}_I^T\mathbf{y}_I(n)\mathbf{y}_I^H(n)\mathbf{w}_I^*\right)}_{d} \quad (100)$$

$$a = E\left(\mathbf{w}_R^T\mathbf{y}_R(n)\mathbf{y}_R^H(n)\mathbf{w}_R^*\right)$$
$$= E\left\{\mathbf{w}_R^T\left(\mathbf{X}_1\mathbf{c}(n) + \mathbf{v}_R(n)\right)\left(\mathbf{X}_1\mathbf{c}(n) + \mathbf{v}_R(n)\right)^H\mathbf{w}_R^*\right\}$$
$$= E\left\{\left(\mathbf{w}_R^T\mathbf{X}_1\mathbf{c}(n) + \mathbf{X}_R^T\mathbf{v}_R(n)\right)\left(\mathbf{c}(n)^H\mathbf{X}_1^H\mathbf{w}_R^* + \mathbf{v}_R(n)^H\mathbf{w}_R^*\right)\right\}$$
$$= E\left\{\mathbf{w}_R^T\mathbf{X}_1\mathbf{c}(n)\mathbf{c}(n)^H\mathbf{X}_1^H\mathbf{w}_R^* + \mathbf{w}_R^T\mathbf{v}_R(n)\mathbf{c}(n)^H\mathbf{X}_1^H\mathbf{w}_R^H + \mathbf{w}_R^T\mathbf{X}_1\mathbf{c}(n)\mathbf{v}_R(n)^H\mathbf{w}_R^* + \mathbf{w}_R^T\mathbf{v}_R(n)\mathbf{v}_R(n)^H\mathbf{w}_R^*\right\}$$
$$= \mathbf{w}_R^T\mathbf{X}_1\mathbf{X}_1^H\mathbf{w}_R^* + \mathbf{w}_R^T E\left\{\mathbf{v}_R(n)\mathbf{v}_R(n)^H\right\}\mathbf{w}_R^*$$
$$= \mathbf{w}_R^T\left(\mathbf{X}_1\mathbf{X}_1^H + E\left\{\mathbf{v}_R(n)\mathbf{v}_R(n)^H\right\}\right)\mathbf{w}_R^*$$
$$= \mathbf{w}_R^T\left(\mathbf{X}_1\mathbf{X}_1^H + \mathbf{P}\right)\mathbf{w}_R^* \quad (101)$$

$$b = E\left(\mathbf{w}_I^T\mathbf{y}_I(n)\mathbf{y}_R^H(n)\mathbf{w}_R^*\right)$$
$$= E\left\{\mathbf{w}_I^T\left(\mathbf{X}_2\mathbf{c}(n) + \mathbf{v}_I(n)\right)\left(\mathbf{X}_1\mathbf{c}(n) + \mathbf{v}_R(n)\right)^H\mathbf{w}_R^*\right\}$$
$$= E\left\{\left(\mathbf{w}_I^T\mathbf{X}_2\mathbf{c}(n) + \mathbf{w}_I^T\mathbf{v}_I(n)\right)\left(\mathbf{c}(n)^H\mathbf{X}_1^H\mathbf{w}_R^* + \mathbf{v}_R(n)^H\mathbf{w}_R^*\right)\right\}$$
$$= E\left\{\mathbf{w}_I^T\mathbf{X}_2\mathbf{c}(n)\mathbf{c}(n)^H\mathbf{X}_1^H\mathbf{w}_R^* + \mathbf{w}_I^T\mathbf{v}_I(n)\mathbf{c}(n)^H\mathbf{X}_1^H\mathbf{w}_R^* + \mathbf{w}_I^T\mathbf{X}_2\mathbf{c}(n)\mathbf{v}_R(n)^H\mathbf{w}_R^* + \mathbf{w}_I^T\mathbf{v}_I(n)\mathbf{v}_R(n)^H\mathbf{w}_R^*\right\}$$
$$= \mathbf{w}_I^T\mathbf{X}_2\mathbf{X}_1^H\mathbf{w}_R^* + \mathbf{w}_I^T E\left\{\mathbf{v}_I(n)\mathbf{v}_R(n)^H\right\}\mathbf{w}_R^*$$
$$= \mathbf{w}_I^T\mathbf{X}_2\mathbf{X}_1^H\mathbf{w}_R^* \quad (102)$$

$$c = E\left(\mathbf{w}_R^T\mathbf{y}_R(n)\mathbf{y}_I^H(n)\mathbf{w}_I^*\right)$$
$$= E\left\{\mathbf{w}_R^T\left(\mathbf{X}_1\mathbf{c}(n) + \mathbf{v}_R(n)\right)\left(\mathbf{X}_2\mathbf{c}(n) + \mathbf{v}_I(n)\right)^H\mathbf{w}_I^*\right\}$$
$$= E\left\{\mathbf{w}_R^T\mathbf{X}_1\mathbf{c}(n)\mathbf{c}(n)^H\mathbf{X}_2^H\mathbf{w}_I^* + \mathbf{w}_R^T\mathbf{v}_R(n)\mathbf{c}(n)^H\mathbf{X}_2^H\mathbf{w}_I^* + \mathbf{w}_R^T\mathbf{X}_1\mathbf{c}(n)\mathbf{v}_I(n)^H\mathbf{w}_I^* + \mathbf{w}_R^T\mathbf{v}_R(n)\mathbf{v}_I(n)^H\mathbf{w}_I^*\right\}$$
$$= \mathbf{w}_R^T\mathbf{X}_1\mathbf{X}_2^H\mathbf{w}_I^* \quad (103)$$

$$
\begin{aligned}
d &= E\left(\mathbf{w}_I^T \mathbf{y}_I(n)\, \mathbf{y}_I^H(n)\, \mathbf{w}_I^*\right) \\
&= E\left\{\mathbf{w}_I^T \left(\mathbf{X}_2\mathbf{c}(n) + \mathbf{v}_I(n)\right)\left(\mathbf{c}(n)^H\mathbf{X}_2^H + \mathbf{v}_I(n)^H\right)\mathbf{w}_I^*\right\} \\
&= E\left\{\left(\mathbf{w}_I^T\mathbf{X}_2\mathbf{c}(n) + \mathbf{w}_I^T\mathbf{v}_I(n)\right)\left(\mathbf{c}(n)^H\mathbf{X}_2^H\mathbf{w}_I^* + \mathbf{v}_I(n)^H\mathbf{w}_I^*\right)\right\} \\
&= E\left\{\mathbf{w}_I^T\mathbf{X}_2\mathbf{c}(n)\mathbf{c}(n)^H\mathbf{X}_2^H\mathbf{w}_I^* + \mathbf{w}_I^T\mathbf{v}_I(n)\mathbf{c}(n)^H\mathbf{X}_2^H\mathbf{w}_I^* + \mathbf{w}_I^T\mathbf{X}_2\mathbf{c}(n)\mathbf{v}_I(n)^H\mathbf{w}_I^* + \mathbf{w}_I^T\mathbf{v}_I(n)\mathbf{v}_I(n)^H\mathbf{w}_I^*\right\} \\
&= \mathbf{w}_I^T\mathbf{X}_2\mathbf{X}_2^H\mathbf{w}_I^* + \mathbf{w}_I^T E\left\{\mathbf{v}_I(n)\mathbf{v}_I(n)^H\right\}\mathbf{w}_I^* \\
&= \mathbf{w}_I^T\left[\mathbf{X}_2\mathbf{X}_2^H + E\left\{\mathbf{v}_I(n)\mathbf{v}_I(n)^H\right\}\right]\mathbf{w}_I^* \\
&= \mathbf{w}_I^T\left(\mathbf{X}_2\mathbf{X}_2^H + \mathbf{P}\right)\mathbf{w}_I^* \qquad (104)
\end{aligned}
$$

Hence,

$$
\begin{aligned}
J =& 1 - \mathbf{w}_R^T\mathbf{X}_1\boldsymbol{\Delta} + \mathbf{w}_I^T\mathbf{X}_2\boldsymbol{\Delta} - \boldsymbol{\Delta}^H\mathbf{X}_1^H\mathbf{w}_R^* + \boldsymbol{\Delta}^H\mathbf{X}_2^H\mathbf{w}_I^* \\
&+ \mathbf{w}_R^T\left(\mathbf{X}_1\mathbf{X}_1^H + \mathbf{P}\right)\mathbf{w}_R^* - \mathbf{w}_I^T\mathbf{X}_2\mathbf{X}_1^H\mathbf{w}_R^* - \mathbf{w}_R^T\mathbf{X}_1\mathbf{X}_2^H\mathbf{w}_I^* \\
&+ \mathbf{w}_I^T\left(\mathbf{X}_2\mathbf{X}_2^H + \mathbf{P}\right)\mathbf{w}_I^* \qquad (105)
\end{aligned}
$$

Since $\mathbf{w}_R = \mathbf{w}_R^*$, $\mathbf{w}_I = \mathbf{w}_I^*$, thus,

$$
\begin{aligned}
J =& 1 - \mathbf{w}_R^T\mathbf{X}_1\boldsymbol{\Delta} + \mathbf{w}_I^T\mathbf{X}_2\boldsymbol{\Delta} - \boldsymbol{\Delta}^H\mathbf{X}_1^H\mathbf{w}_R + \boldsymbol{\Delta}^H\mathbf{X}_2^H\mathbf{w}_I \\
&+ \mathbf{w}_R^T\left(\mathbf{X}_1\mathbf{X}_1^H + \mathbf{P}\right)\mathbf{w}_R - \mathbf{w}_I^T\mathbf{X}_2\mathbf{X}_1^H\mathbf{w}_R - \mathbf{w}_R^T\mathbf{X}_1\mathbf{X}_2^H\mathbf{w}_I \\
&+ \mathbf{w}_I^T\left(\mathbf{X}_2\mathbf{X}_2^H + \mathbf{P}\right)\mathbf{w}_I. \qquad (106)
\end{aligned}
$$

References

[1] M. Rice, *Digital Communications: A Discrete-Time Approach*. Prentice Hall, 2008.

[2] J. Proakis, *Digital Communications*. McGraw-Hill, 2001.

[3] J. Tu, “Optimum MMSE equalization for staggered modulation,” in *Proceedings of the IEEE Asilomar Conference on Signals, Systems, and Computers*, Asilomar, CA, 1–3 November 1993, pp. 1401–1406.

[4] L. J. Greenstein and V. K. Prabhu, “Analysis of multipath outage with applications to 90 Mbit/s PSK systems at 6 and 11 GHz,” *IEEE Transactions on Communications*, vol. COM-27, pp. 68–75, January 1979.

[5] P. A. Bello and K. Pahlavan, “Performance of adaptive equalization for staggered QPSK and QPR over frequency-selective LOS microwave channels,” in *IEEE International Conference on Communications*, 1982, pp. 3H.1.1–1.6.

[6] ——, “Adaptive equalization for SQPSK and SQPR over frequency selective microwave LOS channels,” *IEEE Transactions on Communications*, vol. COM-32, pp. 609–615, May 1984.

[7] J.W.Craig, “Equalization for unbalanced QPSK staggered QPSK,” in *IEEE Globecom*, Houston, TX, 1993.