CONSTANT FALSE ALARM RATE (CFAR) DETECTION BASED ESTIMATORS WITH APPLICATIONS TO SPARSE WIRELESS CHANNELS

A Thesis Submitted to
the Graduate School of Engineering and Sciences of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in Electrical and Electronics Engineering

by
Ümit KARACA

October 2006
İZMİR

We approve the thesis of **Ümit KARACA**

Date of Signature

. . . . . **12 October 2006**

Assist. Prof. Serdar ÖZEN
Supervisor
Department of Electrical and Electronics Engineering
İzmir Institute of Technology

. . . . . **12 October 2006**

Assist. Prof. Mustafa Aziz ALTINKAYA
Department of Electrical and Electronics Engineering
İzmir Institute of Technology

. . . . . **12 October 2006**

Assoc. Prof. Olcay AKAY
Department of Electrical and Electronics Engineering
Dokuz Eylül University

. . . . . **12 October 2006**

Prof. F.Acar SAVACI
Head of Department
Department of Electrical and Electronics Engineering
İzmir Institute of Technology

. . . . . .

Assoc. Prof. Dr. Semahat ÖZDEMİR
Head of the Graduate School

ABSTRACT

CONSTANT FALSE ALARM RATE (CFAR) DETECTION BASED ESTIMATORS WITH APPLICATIONS TO SPARSE WIRELESS CHANNELS

We provide Constant False Alarm Rate (CFAR) based thresholding methods for training based channel impulse response (CIR) estimation algorithms for communication systems which utilize a periodically transmitted training sequence within a continuous stream of information symbols. After obtaining the CIR estimation by using known methods in the literature, there are estimation errors which causes performance loss at equalizers. The channel estimation error can be seen as noise on CIR estimations and CFAR based thresholding methods, which are used in radar systems to decide the presence of a target, can effectively overcome this problem. CFAR based methods are based on determining threshold values which are computed by distribution of channel noise. We provide exact and approximate distribution of channel noise appear at CIR estimate schemes. We applied Cell Averaging-CFAR (CA-CFAR) and Order Statistic-CFAR (OS-CFAR) methods on the CIR estimations. The performance of the CFAR estimators are then compared by their Least Square error in the channel estimates. The Signal to Interference plus Noise Ratio (SINR) performance of the decision feedback equalizers (DFE), of which the tap values are calculated based on the CFAR estimators, are also provided.

ÖZET

SABİT YANLIŞ ALARM ORANI SEZİMLEME TABANLI KANAL KESTİRİMİ VE YOĞUN OLMAYAN TEKİL KANALLARA UYGULAMALARI

Bu çalışmada, haberleşme sistemlerinde kullanılan Kanal Dürtü Yanıtı (CIR) kestirimlerinin eşiklemesinde kullanılmak üzere, Sabit Yanlış Alarm Oranı (CFAR) sezimleme tabanlı metotlar ele alınmıştır. Haberleşme literatüründe bilinen yöntemlerle elde edilen kanal dürtü yanıtları, kestirim hatası taşımakta, bu durum denkleştiricilerde performans kaybına neden olmaktadır. Bu kestirim hatası, kanal dürtü yanıtındaki gürültü olarak değerlendirilebilir. Radar sistemlerinde hedef tespit edilmesinde kullanılan Sabit Yanlış Alarm Oranı (CFAR) sezimleme tabanlı eşikleme metotları, bahsedilen bu gürültünün temizlenmesinde kullanılabilir. Sabit Yanlış Alarm Oranı (CFAR) sezimleme tabanlı metotlar, kanal gürültüsünün istatistiksel dağılımı yardımıyla hesaplanan eşik değerlerine dayanmaktadır.

Proje kapsamında, Hücre Ortalamalı (CA-CFAR) ve İstatistiksel Sıralamalı (OS-CFAR) Sabit Yanlış Alarm Oranı sezimleme tabanlı metotlar kullanılarak elde edilen eşik değerleri, çeşitli kanal kestirimlerine uygulanmıştır. Bahsedilen metotların performansları, eşikleme işleminden sonra elde edilen kanal kestirim sinyallerinin En Küçük Kareler Hataları (NLSE) karşılaştırılarak gösterilmiştir. Ayrıca, Sinyallerin Girişim ve Gürültüye Oranları (SINR), Karar Geridönüşümlü Denkleştiriciler (DFE) kullanılarak gösterilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AR	Auto-regressive
CFAR	Constant False Alarm Rate
CA-CFAR	Cell Averaging Constant False Alarm Rate
CAGO-CFAR	Cell Averaging Greatest-Of Constant False Alarm Rate
CIR	Channel Impulse Response
DFE	Decision Feedback Equalizer
DTV	Digital Television
ISI	Inter-Symbol Interference
LE	Linear Equalizer
LMS	Least Mean Square
LS	Least Squares
ML	Maximum Likelihood
MLSE	Maximum Likelihood Sequence Estimation (Estimator)
MMSE	Minimum Mean-squared Error
MP	Matching Pursuit
MSE	Mean-squared Error
OS-CFAR	Order Statistic Constant False Alarm Rate
RLS	Recursive Least Square
SINR	Signal-to-Interference-Plus-Noise Ratio
SNR	Signal-to-Noise Ratio
VSB	Vestigial Side-band

CHAPTER 1

INTRODUCTION

1.1. Motivation

In mobile wireless and digital television (DTV) channels multi-path phenomena is generally attributed to the randomly changing propagation characteristics as well as the reflection, diffraction and scattering of the transmitted waves from the buildings, large moving vehicles/airplanes, mountains, ionosphere, sea surface. Many types of impairments are observed on these channels such as Doppler spread, fading, multi-path spread (or delay spread), nonlinear distortion, frequency offset, phase jitter, thermal and/or impulsive noise, co- and adjacent channel interference. This research focuses mainly on the effects of multi-path spread and thermal noise. Large delay spread induces significant inter-symbol interference (ISI) where the received symbols are a function of several adjacent symbols. A large Doppler spread causes rapid variations in the channel impulse response characteristics, and necessitates a fast converging adaptive algorithm. When the channel exhibits a deep fade, it results in a very low received signal power. Among various remedies to these problems which have been proposed thus far, the diversity reception and the multi-carrier transmission combined with advanced adaptive signal processing algorithms to estimate and to track the channel variations are considered to be among the most prominent alternatives. However multi-carrier transmission is out of the scope of this research.

Multi-path channels are generally modelled in the form of *frequency selective* fading as well as *Rayleigh* or *Rician* fading. As the receivers employ diversity to combat the combined effects of multi-path and fading in the form of antenna arrays, as well as polarization diversity, more advanced signal processing is required at the receiver front-end to estimate and track the channel parameter variations. In order to be able to come up with better signal processing algorithms first the underlying wireless channel characteristics must be well understood and proper analytical models must be developed.

As the demand increases for higher data rates and more bandwidth, the effects

of all aforementioned channel impairments become more severe. Better channel models, and more advanced algorithms for channel estimation, tracking and equalization are still on the current agenda of communication system researchers and designers.

1.2. Organization and Contributions of the Thesis

This research has been completed under the following two major constraints:

- We considered that the transmission scheme is fixed and has been standardized as in ”(ATSC Standard A/53 1995)”; thus we were constrained to work with and implement algorithms which can only be implemented at the receivers. In other words we assumed no changes will take place to the existing North American DTV transmission standard. This eliminates the possibility of transmitter/receiver joint optimization, or any proposals of improvements to the transmission format.

- Due to having very high data rate, the digital TV system requires algorithms with relatively low complexity. Highly recursive algorithms or blind algorithms that require long averaging to get useful estimates is outside the focus of this research. By the time a blind algorithm converge to a reasonably reliable estimate of the channel, the channel may change significantly thus making the CIR estimate useless. For these reasons we mainly concentrated on training sequence based channel estimation algorithms.

Total of five chapters follow the Introduction chapter:

- Chapter 2 presents the channel and signal model where the signal model obeys the American DTV transmission standard ”(ATSC Standard A/53 1995)”. Notation and main important sets of equations are introduced which will be primarily used in the sequel of the thesis.

- Chapter 3 first overviews the method of (generalized) least squares, and applies it to the signal/channel model that has been introduced in Chapter 2. Then the correlation based initial channel estimation is also introduced. Correlation based channel estimation is considered, since it is readily available in digital receivers for *frame synchronization* purpose. Then the problems associated with both of these methods

will be presented. The most important contribution of Chapter 3 is the development of a detailed statistical model of the *baseline noise* which is a by-product of any type of correlation processing in the receiver. However, as will be shown, this statistical model is actually a function of the channel impulse response which we are trying to estimate. For this very reason we then provide an iterative algorithm to find the generalized least squares channel estimate where the covariance matrix of the baseline noise is incorporated into the channel estimation scheme. Each iteration also involves a thresholding, which is accomplished by two different methods: (i) constant false alarm rate (CA-CFAR) based thresholding, and (ii) order statistic (OS-CFAR) based thresholding.

- Chapter 4 overviews Constant False Alarm Rate (CFAR) based thresholding methods. Cell Averaging (CA) and Order Statistic (OS) based CFAR methods are introduced by the help of statistic distribution given in Chapter 2.

- In Chapter 5, we provide the derivation for the Decision Feedback Equalizers. The CIR estimates thresholded by CFAR methods provided in previous chapters are used for calculating the DFE filter tap values. We then provide the signal-to-interference-plus-noise (SINR) performance at the DFE output.

CHAPTER 2

SIGNAL AND CHANNEL MODEL

2.1. Introduction

For the communications systems utilizing periodically transmitted training sequence, *least-squares* (LS) based channel estimation algorithms or the *correlation* based channel estimation algorithms have been the most widely used two alternatives. Both methods use a stored copy of the known transmitted training sequence at the receiver. The properties and the length of the training sequence are generally different depending on the particular communication system's standard specifications. In addition, a training sequence could also be utilized in a communication system not only for channel estimation purpose but also for synchronization purpose, to indicate the beginning and/or end of a transmitted data frame or packet. In the sequel, although the examples following the derivations of the blended channel estimator will be drawn from the ATSC digital TV 8-VSB system "(Hillery 2001)", to the best of our knowledge it could be applied with minor modifications to any digital communication system with linear modulation which employs a training sequence.

Overview of the data transmission model in Section 2.2 provides the necessary background and introduces the notation which will be entirely used in the rest of the thesis.

2.2. Overview of the Data Transmission Model

We will briefly go over the data transmission model. The base-band transmitted signal waveform of data rate $1/T$ symbols/sec depicted in Figure 2.1 is represented by

$$s(t) \;=\; \sum_{k} I_k q(t - kT) \tag{2.1}$$

where $\{I_k \in \mathcal{A} \equiv \{\alpha_1, \ldots, \alpha_M\} \subset \mathbb{C}^1\}$ is the transmitted data sequence, which is a discrete M-ary sequence taking values on the generally complex M-ary alphabet $\mathcal{A}$,

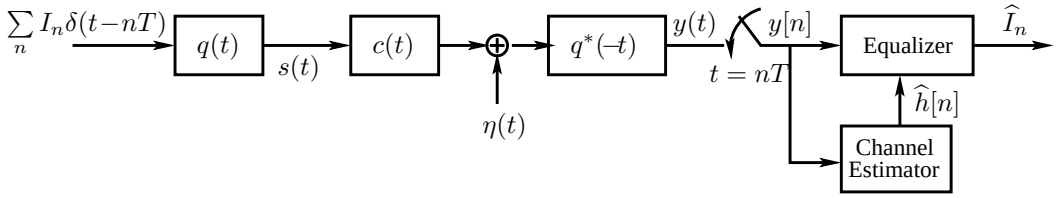

Figure 2.1: System block diagram. $h(t)$ is the composite channel including transmit end receive filters as well as the physical channel.

which also constitutes the two dimensional employed modulation constellation. $q(t)$ is the transmitter pulse shaping filter of finite support $[-T_q/2, T_q/2]$. The overall complex pulse shape will be denoted by $p(t)$ and is given by

$$p(t) \quad = \quad q(t) * q^*(-t) \tag{2.2}$$

where $q^*(-t)$ is the receiver matched filter impulse response. Although it is not required, for the sake of simplifying the notation, we assume that the span of the transmit and receive filters, T_q, is integer multiple of the symbol period, T; that is $T_q = N_q T = 2L_q T$, $L_q \in \mathbb{Z}^+$. We also note that for 8-VSB system "(Hillery 2001)" the transmitter pulse shape is the Hermitian symmetric root-raised cosine pulse, which implies $q(t) = q^*(-t)$. In the sequel $q[n] \equiv q(t)|_{t=nT}$ will be used to denote both the transmit and receive filters. The physical channel between the transmitter and the receiver is denoted by $c(t)$, and throughout this paper the concatenation of $p(t)$ and the channel will be denoted by $h(t,\tau)$, and is defined as

$$h(t,\tau) \quad = \quad q(t) * c(t,\tau) * q^*(-t) = p(t) * c(t,\tau). \tag{2.3}$$

The physical channel $c(t,\tau)$ is generally described by the impulse response

$$c(t,\tau) \quad = \quad \sum_{k=-K}^{L} c_k(\tau)\delta(t-\tau_k) \tag{2.4}$$

which describes a time-varying channel, and $\{c_k(\tau)\} \subset \mathbb{C}^1$, where $-K \leq k \leq L$, and $t, \tau \in \mathbb{R}$, $\{\tau_k\}$ denote the multipath delays, or the Time-Of-Arrivals (TOA). We will

assume that the time-variations of the channel is slow enough that $c(t,\tau) = c(t)$ can be assumed to be a fixed (static) inter-symbol interference (ISI) channel throughout the training period; that is we will assume that $c_k(\tau) = c_k$, which in turn implies

$$c(t) \;=\; \sum_{k=-K}^{L} c_k \delta(t - \tau_k) \tag{2.5}$$

for $0 \leq t \leq NT$, where N is the number of training symbols. In general $c_k = \tilde{c}_k e^{-j2\pi f_c \tau_k}$ with $\tilde{c}_k$ being the amplitude of the k'th multipath, and f_c is the carrier frequency. It is also inherently assumed that $\tau_k < 0$ for $-K \leq k \leq -1$, $\tau_0 = 0$, and $\tau_k > 0$ for $1 \leq k \leq L$. The particular choice of the summation indices K and L, the number of maximum anti-causal and causal multi-path delays respectively, will be clarified in the discussion of correlation based channel estimation and onwards. It is important to clarify that the multi-path delays τ_k are *not assumed to be at integer multiples of the sampling period* T. Indeed it is one of the main contributions of this paper that we show an accurate and robust way to recover the pulse shape back into the concatenated channel estimate when the multi-path delays are not exactly at the sampling instants. By combining Equations (2.3) and (2.5) (and by dropping the τ index) we get

$$h(t) \;=\; p(t) * c(t) = \sum_{k=-K}^{L} c_k p(t - \tau_k). \tag{2.6}$$

Since both $p(t)$ and $c(t)$ are complex valued functions, the overall channel impulse response $h(t)$ is also complex valued. We can write them in terms of their real and imaginary parts:

$$p(t) \;=\; p_I(t) + j p_Q(t), \tag{2.7}$$

$$c(t) \;=\; c_I(t) + j c_Q(t), \tag{2.8}$$

$$h(t) \;=\; h_I(t) + j h_Q(t). \tag{2.9}$$

Then

$$h(t) \;=\; p(t) * c(t) = (p_I(t) + j p_Q(t)) * (c_I(t) + j c_Q(t))$$

$$= \underbrace{(p_I(t) * c_I(t) - p_Q(t) * c_Q(t))}_{h_I(t)} + j \underbrace{(p_I(t) * c_Q(t) + p_Q(t) * c_I(t))}_{h_Q(t)}. \quad (2.10)$$

By using the notation introduced here the matched filter output $y(t)$ is given by

$$y(t) = \left(\sum_k I_k \delta(t - kT) \right) * h(t) + \nu(t) \quad (2.11)$$

where

$$\nu(t) = \eta(t) * q^*(-t)$$

denotes the complex (colored) noise process after the pulse matched filter, with $\eta(t)$ being a zero-mean white Gaussian noise process with spectral density σ_η^2 per real and imaginary part. Similar to Equations (2.7-2.9) $y(t)$ can be also written in terms of its real and imaginary parts: $y(t) = y_I(t) + j y_Q(t)$. Sampling the matched filter output at the symbol rate we obtain the discrete-time representation of the overall communication system as

$$y[n] \equiv y(t)|_{t=nT} = \sum_k I_k h[n-k] + \nu[n]. \quad (2.12)$$

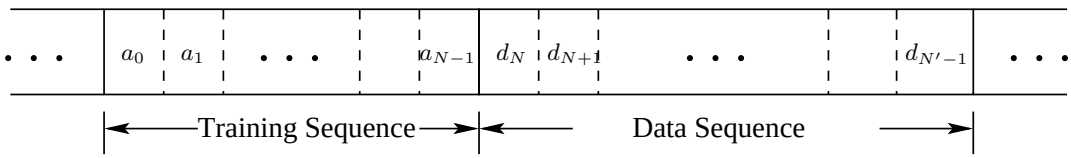

Figure 2.2: Data frame: N symbols of known training sequence followed by $N' - N$ information symbols.

Referring to Figure 2.2, the transmitted symbols are composed of *frame*s (or packets) of length N', where the first N symbols are the training symbols. Within a frame of length N', the symbols are denoted by

$$I_k = \begin{cases} a_k, & \text{for } 0 \le k \le N-1 \\ d_k, & \text{for } N \le n \le N'-1, \end{cases} \quad (2.13)$$

where the distinction of first N symbols is made to indicate that they are the known training symbols, and it is possible that the a_k's belong to a certain subset of the M-ary constellation alphabet $\mathcal{A}$; that is $\{a_k \in \tilde{\mathcal{A}} \subset \mathcal{A} \equiv \{\alpha_1, \ldots, \alpha_M\}\}$. In fact for the 8-VSB system the signal alphabet is $\mathcal{A} \equiv \{\pm 1, \pm 3, \pm 5, \pm 7\}$, while the training sequence can only take binary values within the set $\tilde{\mathcal{A}} \equiv \{-5, +5\}$.

In the sequel the sampled matched filter output signal $y[n]$ will be used extensively in vector form, and to help minimize introducing new variables, the notation of $\boldsymbol{y}_{[n_1:n_2]}$ with $n_2 \geq n_1$, will be adopted to indicate the column vector

$$\boldsymbol{y}_{[n_1:n_2]} = [y[n_1], y[n_1+1], \cdots, y[n_2]]^T.$$

Same notation will also be applied to the noise variables $\eta[n]$ and $\nu[n]$.

Without loss of generality symbol rate sampled, complex valued, composite CIR $h[n]$ can be written as a finite dimensional vector

$$\boldsymbol{h} = [h[-N_a], h[-N_a+1], \cdots, h[-1], h[0], h[1], \cdots, h[N_c-1], h[N_c]]^T \quad (2.14)$$

where N_a and N_c denote the number of anti-causal and the causal taps of the channel, respectively, and are given by

$$N_a = \text{round}\left\{\frac{\tau_{-K} - TN_q}{T}\right\}, \text{ and } N_c = \text{round}\left\{\frac{\tau_L + TN_q}{T}\right\},$$

and $N_a + N_c + 1$ is the total memory of the channel. Based on Equation (2.12) and assuming that $N \geq N_a + N_c + 1$, we can write the pulse matched filter output corresponding *only* to the known training symbols:

$$\begin{aligned}
y[N_c] &= h[-N_a]a_{N_c+N_a} + \cdots + h[0]a_{N_c} + \cdots + h[N_c]a_0 + \nu[N_c] \\
y[N_c+1] &= h[-N_a]a_{N_c+N_a+1} + \cdots + h[0]a_{N_c+1} + \cdots + h[N_c]a_1 + \nu[N_c+1] \\
&\vdots \qquad\qquad \vdots \\
y[N-1-N_a] &= h[-N_a]a_{N-1} + \cdots + h[0]a_{N-1-N_a} + \cdots + h[N_c]a_{N-1-N_a-N_c} + \nu[N-1-N_a]
\end{aligned}$$

which can be written compactly as

$$\boldsymbol{y}_{[N_c:N-N_a-1]} = \tilde{\boldsymbol{A}}\boldsymbol{h} + \boldsymbol{\nu}_{[N_c:N-N_a-1]} = \tilde{\boldsymbol{A}}\boldsymbol{h} + \tilde{\boldsymbol{Q}}\boldsymbol{\eta}_{[N_c-L_q:N-1-N_a+L_q]}, \quad (2.15)$$

where

$$\tilde{\boldsymbol{A}} = \mathcal{T}\left\{[a_{N_c+N_a}, \cdots, a_{N-1}]^T, [a_{N_c+N_a}, \cdots, a_0]\right\} \tag{2.16}$$

$$= \begin{bmatrix} a_{N_c+N_a} & a_{N_c+N_a-1} & \cdots & a_0 \\ a_{N_c+N_a+1} & a_{N_c+N_a} & \cdots & a_1 \\ \vdots & \vdots & \ddots & \vdots \\ a_{N-1} & a_{N-2} & \cdots & a_{N-1-N_a-N_c} \end{bmatrix}, \tag{2.17}$$

where $\tilde{\boldsymbol{A}}$ is $(N - N_a - N_c) \times (N_a + N_c + 1)$ Toeplitz convolution matrix with first column $[a_{N_c+N_a}, \cdots, a_{N-1}]^T$ and first row $[a_{N_c+N_a}, \cdots, a_0]$, and

$$\boldsymbol{\nu}_{[N_c:N-N_a-1]} = \tilde{\boldsymbol{Q}}\boldsymbol{\eta}_{[N_c-L_q:N-1-N_a+L_q]} \tag{2.18}$$

is the colored noise vector at the receiver matched filter output, with

$$\tilde{\boldsymbol{Q}} = \begin{bmatrix} \boldsymbol{q}^T & 0 & \cdots & 0 \\ 0 & \boldsymbol{q}^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{q}^T \end{bmatrix}_{(N-N_a-N_c)\times(N-N_a-N_c+N_q)} \tag{2.19}$$

and $\boldsymbol{q}$ is the vector containing time-reversed samples of the receiver matched filter sampled at the symbol rate and is

$$\boldsymbol{q} = [q[+L_q], \cdots, q[0], \cdots, q[-L_q]]^T. \tag{2.20}$$

Note that $\boldsymbol{q}$ has $N_q + 1 = 2L_q + 1$ samples.

Similarly the pulse matched filter output which includes *all* the contributions from the known training symbols (which includes the adjacent random data as well) can be written as

$$\boldsymbol{y}_{[-N_a:N+N_c-1]} = (\boldsymbol{A} + \boldsymbol{D})\,\boldsymbol{h} + \boldsymbol{\nu}_{[-N_a:N+N_c-1]} \tag{2.21}$$

$$= \boldsymbol{A}\boldsymbol{h} + \boldsymbol{D}\boldsymbol{h} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}, \tag{2.22}$$

where

$$\boldsymbol{A} = \mathcal{T}\left\{[a_0, \cdots, a_{N-1}, \underbrace{0, \cdots, 0}_{N_a+N_c}]^T, [a_0, \underbrace{0, \cdots, 0}_{N_a+N_c}]\right\} \tag{2.23}$$

is a Toeplitz matrix of dimension $(N+N_a+N_c)\times(N_a+N_c+1)$ with first column $[a_0, a_1, \cdots, a_{N-1}, 0, \cdots, 0]^T$, and first row $[a_0, 0, \cdots, 0]$, and

$$\boldsymbol{D} = \mathcal{T}\left\{[\underbrace{0, \cdots, 0}_{N}, d_N, \cdots, d_{N_c+N_a+N-1}]^T, [0, d_{-1}, \cdots, d_{-N_c-N_a}]\right\}, \tag{2.24}$$

is a Toeplitz matrix which includes the adjacent unknown symbols only, prior to and after the training sequence. The data sequence $[d_{-1}, \cdots, d_{-N_c-N_a}]$ is the unknown information symbols transmitted at the end of the frame prior to the current frame being transmitted. $\boldsymbol{Q}$ is of dimension $(N+N_a+N_c)\times(N+N_a+N_c+N_q)$ and has the same convolution matrix structure with $\tilde{\boldsymbol{Q}}$ as displayed in Equation (2.19), and is given by.

$$\boldsymbol{Q} = \begin{bmatrix} \boldsymbol{q}^T & 0 & \cdots & 0 \\ 0 & \boldsymbol{q}^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{q}^T \end{bmatrix}_{(N+N_a+N_c)\times(N+N_a+N_c+N_q)} \tag{2.25}$$

and $\boldsymbol{q} = [q[+L_q], \cdots, q[0], \cdots, q[-L_q]]^T$.

Equation (2.22) which includes *all* the contributions from the known training symbols (which includes the adjacent random data as well) can be written explicitly as

$$\boldsymbol{y}_{[-N_a:N+N_c-1]} = \begin{bmatrix} y[-N_a] \\ \vdots \\ y[N_c-1] \\ \left\{\begin{matrix} y[N_c] \\ \vdots \\ y[N-N_a-1] \end{matrix}\right\} \\ y[N-N_a] \\ \vdots \\ y[N+N_c-1] \end{bmatrix} = \begin{bmatrix} y[-N_a] \\ \vdots \\ y[N_c-1] \\ \cdots\cdots\uparrow\cdots\cdots \\ \boldsymbol{y}_{[N_c:N-N_a-1]} \\ \cdots\cdots\downarrow\cdots\cdots \\ y[N-N_a] \\ \vdots \\ y[N+N_c-1] \end{bmatrix} = \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}$$

$$+\Bigg(\underbrace{\begin{bmatrix} a_0 & 0 & 0 & \cdots & 0 \\ a_1 & a_0 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ a_{N_c+N_a-1} & a_{N_c+N_a-2} & \cdots & a_0 & 0 \\ \hdashline a_{N_c+N_a} & a_{N_c+N_a-1} & \cdots & & a_0 \\ a_{N_c+N_a+1} & a_{N_c+N_a} & \cdots & & a_1 \\ \vdots & \vdots & & & \vdots \\ a_{N-1} & a_{N-2} & \cdots & & a_{N-1-N_a-N_c} \\ \hdashline 0 & a_{N-1} & a_{N-2} & \cdots & a_{N-2-N_a-N_c} \\ 0 & 0 & a_{N-1} & \cdots & a_{N-2-N_a-N_c} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & a_{N-1} \end{bmatrix}}_{\boldsymbol{A}} + \underbrace{\begin{bmatrix} 0 & d_{-1} & d_{-2} & \cdots & d_{-N_a-N_c} \\ 0 & 0 & d_{-1} & \cdots & d_{-N_a-N_c+1} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & d_{-1} \\ \hdashline 0 & 0 & & \cdots & 0 \\ 0 & 0 & & \cdots & 0 \\ \vdots & \vdots & & & \vdots \\ 0 & 0 & & \cdots & 0 \\ \hdashline d_N & 0 & 0 & \cdots & 0 \\ d_{N+1} & d_N & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ d_{N_a+N_c+N-1} & & \cdots & d_N & 0 \end{bmatrix}}_{\boldsymbol{D}}\Bigg)\boldsymbol{h} \quad (2.26)$$

where the entries of the vector $\boldsymbol{y}_{[-N_a:N+N_c-1]}$ between the *dotted* lines denote the matched filter output corresponding *only* to the known training symbols which is provided in Equation (2.15). Note that the corresponding entries of the matrix $\boldsymbol{A}$ between the dotted lines are exactly the same as the entries of the matrix $\tilde{\boldsymbol{A}}$ which is provided in Equation (2.17), and the corresponding entries of the matrix $\boldsymbol{D}$ between the dotted lines are all zeros.

We now write the contributions of the unknown symbols $\boldsymbol{Dh}$ in Equation (2.22) in a different format which will prove to be more useful in the subsequent derivations. We first define $\boldsymbol{d} = \boldsymbol{S}\tilde{\boldsymbol{d}}$, or equivalently $\tilde{\boldsymbol{d}} = \boldsymbol{S}^T\boldsymbol{d}$, where

$$\tilde{\boldsymbol{d}} = [d_{-N_c-N_a}, \cdots, d_{-1}, \boldsymbol{0}_{1\times N}, d_N, \cdots, d_{N+N_c+N_a-1}]^T \quad (2.27)$$

$$\boldsymbol{d} = [d_{-N-N_a}, \cdots, d_{-1}, d_N, \cdots, d_{N+N_c+N_a-1}]^T \quad (2.28)$$

$$\boldsymbol{S} = \begin{bmatrix} \boldsymbol{I}_{N_a+N_c} & \boldsymbol{0}_{(N_a+N_c)\times N} & \boldsymbol{0}_{(N_a+N_c)\times(N_c+N_a)} \\ \boldsymbol{0}_{(N_a+N_c)\times(N_a+N_c)} & \boldsymbol{0}_{(N_a+N_c)\times N} & \boldsymbol{I}_{N_a+N_c} \end{bmatrix}_{(2(N_c+N_a))\times(N+2(N_a+N_c))} \quad (2.29)$$

where $\boldsymbol{S}$ is a *selection* matrix which retains the random data, eliminates the N zeros in the middle of the vector $\tilde{\boldsymbol{d}}$. We also introduce

$$\boldsymbol{\mathcal{H}} = \begin{bmatrix} \bar{\boldsymbol{h}}^T & 0 & \cdots & 0 \\ 0 & \bar{\boldsymbol{h}}^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \bar{\boldsymbol{h}}^T \end{bmatrix}_{(N+N_c+N_a)\times(N+2(N_a+N_c))} \tag{2.30}$$

$$\bar{\boldsymbol{h}} = [h[N_c], \cdots, h[1], h[0], h[-1], \cdots, h[-N_a]]^T = \boldsymbol{J}\boldsymbol{h} \tag{2.31}$$

$$\boldsymbol{J} = \begin{bmatrix} 0 & \cdots & 0 & 1 \\ 0 & \cdots & 1 & 0 \\ \vdots & \vdots & & \vdots \\ 1 & 0 & \cdots & 0 \end{bmatrix}_{(N_a+N_c+1)\times(N_a+N_c+1)} \tag{2.32}$$

$$\boldsymbol{H} = \boldsymbol{\mathcal{H}}\boldsymbol{S}^T \tag{2.33}$$

where $\bar{\boldsymbol{h}}$ is the time reversed version of $\boldsymbol{h}$ (re-ordering is accomplished by the permutation matrix $\boldsymbol{J}$), and $\boldsymbol{H}$ is of dimension $(N + N_a + N_c) \times (2(N_c+N_a))$ with a "hole" inside which is created by the selection matrix $\boldsymbol{S}$ as defined in Equation (2.29). Then it is trivial to show that

$$\boldsymbol{D}\boldsymbol{h} = \boldsymbol{\mathcal{H}}\tilde{\boldsymbol{d}} = \boldsymbol{\mathcal{H}}\boldsymbol{S}^T\boldsymbol{d} = \boldsymbol{H}\boldsymbol{d}. \tag{2.34}$$

Based on the Equations (2.27-2.34) we can rewrite Equation (2.22) as

$$\begin{aligned} \boldsymbol{y}_{[-N_a:N+N_c-1]} &= \boldsymbol{A}\boldsymbol{h} + \boldsymbol{D}\boldsymbol{h} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]} \\ &= \boldsymbol{A}\boldsymbol{h} + \boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}, \end{aligned} \tag{2.35}$$

CHAPTER 3

GENERALIZED LEAST SQUARES BASED CHANNEL ESTIMATION

3.1. Method of Least Squares

First we briefly overview the method of least squares in general terms that is very widely used in the statistical inference and estimation theory applications without considering any specific signal processing or communication system framework.

Consider the linear model

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\nu} \tag{3.1}$$

where $\boldsymbol{y}$ is the observation (or response) vector, $\boldsymbol{A}$ is the regression (or design) matrix, $\boldsymbol{x}$ is the vector of unknown parameters to be estimated, and $\boldsymbol{\nu}$ is the observation noise (or measurement error) vector, and are given by

$$\boldsymbol{y} = [y_1, \cdots, y_n]^T, \tag{3.2}$$

$$\boldsymbol{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,p} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,3} & a_{n,2} & \cdots & a_{n,p} \end{bmatrix}, \tag{3.3}$$

$$\boldsymbol{x} = [x_1, \cdots, x_p]^T, \tag{3.4}$$

$$\boldsymbol{\nu} = [\nu_1, \cdots, \nu_n]^T. \tag{3.5}$$

It is assumed that all the variables in Equation (3.1) are generally complex valued, that is $\boldsymbol{y}, \boldsymbol{\nu} \in \mathbb{C}^n$, $\boldsymbol{x} \in \mathbb{C}^p$ and $\boldsymbol{A} \in \mathbb{C}^{n \times p}$.

Then the *ordinary least squares* solution $\hat{\boldsymbol{x}}_{ols}$ can be obtained by minimizing the objective function

$$J_{OLS}(\boldsymbol{x}) \;=\; \boldsymbol{\nu}^H\boldsymbol{\nu} = \|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|^2 \tag{3.6}$$

is given by

$$\hat{\boldsymbol{x}}_{ols} \;=\; (\boldsymbol{A}^H\boldsymbol{A})^{-1}\boldsymbol{A}^H\boldsymbol{y}, \tag{3.7}$$

whenever the matrix $\boldsymbol{A}$ has rank p. The estimator of (3.7) is called the *best linear unbiased estimate* (BLUE) among all *linear* unbiased estimators if the noise covariance matrix is *known* to be "(Seber 1977, Casella and Berger 1990)"

$$\text{Cov}\{\boldsymbol{\nu}\} = \boldsymbol{K}_\nu \equiv \frac{1}{2}E\{\boldsymbol{\nu}\boldsymbol{\nu}^H\} = \sigma^2_{\boldsymbol{\nu}}\boldsymbol{I}. \tag{3.8}$$

The estimator of (3.7) is called the *minimum variance unbiased estimator* (MVUE) among *all* unbiased estimators (not only linear) if the noise is *known* to be Gaussian with zero mean and with covariance matrix $\boldsymbol{K}_\nu$ of (3.8), that is $\hat{\boldsymbol{x}}_{ols}$ is called MVUE if it is known that $\boldsymbol{\nu} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2_{\boldsymbol{\nu}}\boldsymbol{I})$.

However if it is known that the vector $\boldsymbol{\nu}$ is correlated, that is $\boldsymbol{K}_\nu \neq \sigma^2_{\boldsymbol{\nu}}\boldsymbol{I}$, then in order to achieve the BLUE property we must use a modified objective function. Since $\boldsymbol{K}_\nu$ is positive definite, there exists an $n \times n$ nonsingular matrix $\boldsymbol{V}$ such that $\boldsymbol{K}_\nu = \boldsymbol{V}\boldsymbol{V}^H$ "(Seber 1977)". Therefore setting $\boldsymbol{z} = \boldsymbol{V}^{-1}\boldsymbol{y}$, $\boldsymbol{B} = \boldsymbol{V}^{-1}\boldsymbol{X}$, and $\boldsymbol{\beta} = \boldsymbol{V}^{-1}\boldsymbol{\nu}$, we have the model

$$\boldsymbol{z} \;=\; \boldsymbol{B}\boldsymbol{x} + \boldsymbol{\beta} \tag{3.9}$$

where $\boldsymbol{B}$ is $n \times p$ of rank p, and $\text{Cov}\{\boldsymbol{\beta}\} = \boldsymbol{K}_{\boldsymbol{\beta}} = \boldsymbol{I}$. Then we define the (generalized) objective function for the model of (3.9) by

$$\begin{aligned} J_{GLS}(\boldsymbol{x}) &= \boldsymbol{\beta}^H\boldsymbol{\beta} = \|\boldsymbol{z}-\boldsymbol{B}\boldsymbol{x}\|^2 \\ &= \boldsymbol{\nu}^H\boldsymbol{V}^{-H}\boldsymbol{V}^{-1}\boldsymbol{\nu} = \boldsymbol{\nu}^H\boldsymbol{K}_\nu^{-1}\boldsymbol{\nu} \\ &= (\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x})^H\boldsymbol{K}_\nu^{-1}(\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}). \end{aligned} \tag{3.10}$$

The least squares estimate that minimizes Equation (3.10) is

$$\begin{aligned} \hat{\boldsymbol{x}}_{gls} &= (\boldsymbol{B}^H \boldsymbol{B})^{-1} \boldsymbol{B}^H \boldsymbol{z} \\ &= (\boldsymbol{A}^H \boldsymbol{K}_\nu^{-1} \boldsymbol{A})^{-1} \boldsymbol{A}^H \boldsymbol{K}_\nu^{-1} \boldsymbol{y}, \end{aligned} \tag{3.11}$$

The estimator of (3.11) is called the *best linear unbiased estimate* (BLUE) among all *linear* unbiased estimators if the noise covariance matrix is *known* to be $\text{Cov}\{\boldsymbol{\nu}\} = \boldsymbol{K}_\nu \equiv \frac{1}{2} E\{\boldsymbol{\nu}\boldsymbol{\nu}^H\} \neq \sigma_{\boldsymbol{\nu}}^2 \boldsymbol{I}$ "(Seber 1977, Casella and Berger 1990)". The estimator of (3.11) is called the *minimum variance unbiased estimator* (MVUE) among *all* unbiased estimators (not only linear) if the noise is *known* to be Gaussian with zero mean and with covariance matrix $\boldsymbol{K}_\nu$, that is $\hat{\boldsymbol{x}}_{ols}$ is called MVUE if it is known that $\boldsymbol{\nu} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K}_\nu)$.

3.2. Existing Channel Estimation Methods

Based on the model of Equation (2.12) we will briefly review the LS based channel estimation and correlation based channel estimation algorithms.

3.2.1. Least-Squares Channel Estimation

In order to fully estimate the channel of Equation (2.6) the LS based channel estimation algorithm assumes that the starting and the ending points of the channel taps are either known or can be bounded. This assumption plays a critical role in the overall quality and robustness of the LS estimation procedure, which will be investigated in the following section.

Recall that symbol rate sampled, complex valued, composite CIR $h[n]$ can be written as a finite dimensional vector

$$\boldsymbol{h} = [h[-N_a], h[-N_a+1], \cdots, h[-1], h[0], h[1], \cdots, h[N_c-1], h[N_c]]^T \tag{3.12}$$

where N_a and N_c denote the number of anti-causal and the causal taps of the channel, respectively, and $N_a + N_c + 1$ is the total memory of the channel. Recall also that the pulse matched filter output corresponding *only* to the known training symbols is given by

$$\boldsymbol{y}_{[N_c:N-N_a-1]} = \tilde{\boldsymbol{A}}\boldsymbol{h} + \boldsymbol{\nu}_{[N_c:N-N_a-1]} = \tilde{\boldsymbol{A}}\boldsymbol{h} + \tilde{\boldsymbol{Q}}\boldsymbol{\eta}_{[N_c-L_q:N-1-N_a+L_q]}, \quad (3.13)$$

where

$$\begin{aligned}\tilde{\boldsymbol{A}} &= \mathcal{T}\left\{[a_{N_c+N_a}, \cdots, a_{N-1}]^T, [a_{N_c+N_a}, \cdots, a_0]\right\} \\ &= \begin{bmatrix} a_{N_c+N_a} & a_{N_c+N_a-1} & \cdots & a_0 \\ a_{N_c+N_a+1} & a_{N_c+N_a} & \cdots & a_1 \\ \vdots & \vdots & \ddots & \vdots \\ a_{N-1} & a_{N-2} & \cdots & a_{N-1-N_a-N_c} \end{bmatrix}, \end{aligned} \quad (3.14)$$

where $\tilde{\boldsymbol{A}}$ is $(N - N_a - N_c) \times (N_a + N_c + 1)$ Toeplitz convolution matrix with first column $[a_{N_c+N_a}, \cdots, a_{N-1}]^T$ and first row $[a_{N_c+N_a}, \cdots, a_0]$, and

$$\boldsymbol{\nu}_{[N_c:N-N_a-1]} = \tilde{\boldsymbol{Q}}\boldsymbol{\eta}_{[N_c-L_q:N-1-N_a+L_q]}$$

is the colored noise at the receiver matched filter output, with

$$\tilde{\boldsymbol{Q}} = \begin{bmatrix} \boldsymbol{q}^T & 0 & \cdots & 0 \\ 0 & \boldsymbol{q}^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{q}^T \end{bmatrix}_{(N-N_a-N_c)\times(N-N_a-N_c+N_q)} \quad (3.15)$$

$$\boldsymbol{q} = [q[+L_q], \cdots, q[0], \cdots, q[-L_q]]^T. \quad (3.16)$$

The covariance matrix of the colored noise vector $\boldsymbol{\nu}_{[N_c:N-N_a-1]} = \boldsymbol{Q}\boldsymbol{\eta}_{[N_c-L_q:N-1-N_a+L_q]}$ is denoted by $\boldsymbol{K}_\nu$, and is given by

$$\text{Cov}\{\boldsymbol{\nu}\} = \boldsymbol{K}_\nu \equiv \frac{1}{2}E\{\boldsymbol{\nu}\boldsymbol{\nu}^H\} = \sigma_\eta^2 \tilde{\boldsymbol{Q}}\tilde{\boldsymbol{Q}}^H \quad (3.17)$$

where σ_η^2 is the variance of the noise sequence $\eta[n]$.

As long as the matrix $\tilde{\boldsymbol{A}}$ is a tall matrix and of full column rank, that is

(i) $N \geq 2(N_a + N_c) + 1$,

(ii) rank$\{\tilde{\boldsymbol{A}}\} = N_a + N_c + 1$

then the generalized least squares solution which minimizes the objective function

$$J_{GLS}(\boldsymbol{h}) = (\boldsymbol{y}_{[N_c:N-N_a-1]} - \tilde{\boldsymbol{A}}\boldsymbol{h})^H (\boldsymbol{K}_\nu)^{-1} (\boldsymbol{y}_{[N_c:N-N_a-1]} - \tilde{\boldsymbol{A}}\boldsymbol{h}) \tag{3.18}$$

exists and unique, and is given by

$$\begin{aligned} \widehat{\mathbf{h}}_{LS} &= (\tilde{\boldsymbol{A}}^H (\sigma_\eta^2 \tilde{\boldsymbol{Q}}\tilde{\boldsymbol{Q}}^H)^{-1} \tilde{\boldsymbol{A}})^{-1} \tilde{\boldsymbol{A}}^H (\sigma_\eta^2 \tilde{\boldsymbol{Q}}\tilde{\boldsymbol{Q}}^H)^{-1} \boldsymbol{y}_{[N_c:N-N_a-1]} \\ &= (\tilde{\boldsymbol{A}}^H (\tilde{\boldsymbol{Q}}\tilde{\boldsymbol{Q}}^H)^{-1} \tilde{\boldsymbol{A}})^{-1} \tilde{\boldsymbol{A}}^H (\tilde{\boldsymbol{Q}}\tilde{\boldsymbol{Q}}^H)^{-1} \boldsymbol{y}_{[N_c:N-N_a-1]}. \end{aligned} \tag{3.19}$$

3.2.2. Correlation Based Channel Estimation

We are assuming that in order to be able to use correlation based channel estimation schemes the training sequences must belong to certain classes of sequences, and thereby possess some certain "nice" correlation properties. One of these classes of sequences is *maximal length pseudo-noise* (PN) sequences. We will denote a PN-sequence of length n as PN_n. In general, the periodic autocorrelation of a binary valued ($\{+A, -A\}$) PN_n sequence is given by

$$r_{PN_n}[m] = \begin{cases} A^2 n, & \text{if } m = 0, \pm n, \pm 2n, \cdots \\ -A^2. & \text{otherwise.} \end{cases} \tag{3.20}$$

However if the PN sequence used is finite and the standard linear correlation is used, then the auto-correlation values corresponding to the non-zero lags will not constant and will not be as low as $-A^2$. As a simple illustration consider a sequence composed of six PN_{511} appended back to back, that is let

$$\mathbf{y} = [PN_{511}, PN_{511}, PN_{511}, PN_{511}, PN_{511}, PN_{511}]^T. \tag{3.21}$$

Then $r_{\mathbf{xy}}[m]$, with $\mathbf{x} = [PN_{511}]^T$, will be given as in Figure 3.1.

It is important to note that we will obtain a low correlation value of $-A^2$ for lags that are not multiples of $L = 511$, corresponding to the intermediate PN_{511} portions of the long sequence $\mathbf{y}$. However as illustrated in Figure 3.1, for outer most lags we will

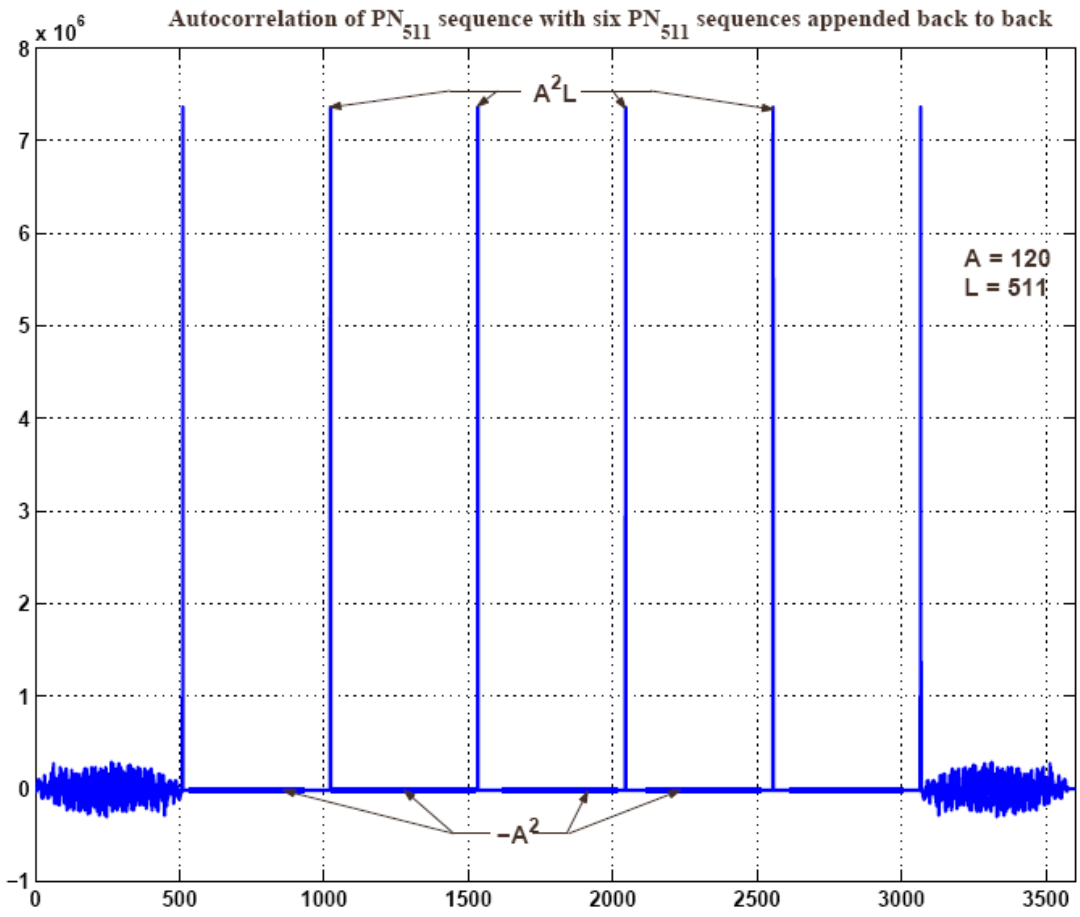

Figure 3.1: Correlation properties of finite PN sequences. Note the “noisy” correlation at both ends of the correlation values.

not achieve this constant and low correlation value; instead we will have a “noise” like correlation due to the finiteness of the sequences.

The training sequence used at the transmitter is, a part of the digital TV standard ”(ATSC Standard A/53 1995)”, which is actually

$$\tilde{\mathbf{s}} = [FS_4, PN_{511}, PN_{63}, \pm PN_{63}, PN_{63}]^T,$$

of length 704, where FS_4 stands for the frame synchronization symbols of length 4. We also have reserved frame bits and information bits right before and after the training sequence $\tilde{\mathbf{s}}$.

As a summary the correlations of the received signal with the stored sequence will be “noisy” because

- the PN sequences are finite in length, they won't achieve their low correlation value for non-zero lag;
- the span of the cross-correlation includes the known training sequence as well as the random data symbols and reserved data symbols.

In the sequel we will show how to "clean" the sidelobes of the finite correlation. However this still leaves the problem of having the cross-correlation span include random symbols which are located prior to and after the training sequence. In order to remedy this problem we will introduce "thresholding".

Cross correlating the stored training sequence with the received sequence, which is readily available in digital receivers for the primary purpose of *frame synchronization* "(Fimoff and Özen and Nereyanuru and Zoltowski and Hillery 2001)", yields a raw channel estimate

$$\hat{h}_u[n] = \frac{1}{r_a[0]} \sum_{k=0}^{N-1} a_k^* y[k+n],\ n = -N_a, \cdots, 0, \cdots, N_c \tag{3.22}$$

where $r_a[0] = \sum_{k=0}^{N-1} \|a_k\|^2$. Equivalently Equation (3.22) can be written as

$$\hat{\boldsymbol{h}}_u = \frac{1}{r_a[0]} \boldsymbol{A}^H \boldsymbol{y}_{[-N_a:N+N_c-1]}. \tag{3.23}$$

Recall that

$$\begin{aligned} \boldsymbol{y}_{[-N_a:N+N_c-1]} &= \boldsymbol{A}\boldsymbol{h} + \boldsymbol{D}\boldsymbol{h} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}, \\ &= \boldsymbol{A}\boldsymbol{h} + \boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}. \end{aligned} \tag{3.24}$$

Substituting Equation (3.24) into (3.23) we get

$$\begin{aligned} \hat{\boldsymbol{h}}_u &= \frac{1}{r_a[0]} \boldsymbol{A}^H \left(\boldsymbol{A}\boldsymbol{h} + \boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]} \right) \\ &= \frac{1}{r_a[0]} \boldsymbol{A}^H \boldsymbol{A}\boldsymbol{h} + \frac{1}{r_a[0]} \boldsymbol{A}^H \left(\boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]} \right). \end{aligned} \tag{3.25}$$

In order to get rid of the sidelobes of the aperiodic autocorrelation we can simply invert the normalized autocorrelation matrix $\boldsymbol{R}_{aa}$ of the training symbols, defined by

$$\boldsymbol{R}_{aa} = \frac{1}{r_a[0]}\boldsymbol{A}^H\boldsymbol{A}. \tag{3.26}$$

Then the *cleaned* channel estimate $\hat{\boldsymbol{h}}_c$ is obtained from

$$\begin{aligned}\hat{\boldsymbol{h}}_c &= \boldsymbol{R}_{aa}^{-1}\hat{\boldsymbol{h}}_u \\ &= (\boldsymbol{A}^H\boldsymbol{A})^{-1}\boldsymbol{A}^H\boldsymbol{y}_{[-N_a:N+N_c-1]}.\end{aligned} \tag{3.27}$$

Substituting Equation (3.23) into (3.27) we get

$$\hat{\boldsymbol{h}}_c = \boldsymbol{h} + \left(\boldsymbol{A}^H\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H\left(\boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\right). \tag{3.28}$$

As can be seen from Equation (3.28) the channel estimate $\hat{\boldsymbol{h}}_c$ has the contributions due to unknown symbols prior to and after the training sequence, which are elements of the vector $\boldsymbol{d}$, as well as the additive channel noise; only the sidelobes due to aperiodic auto-correlation is removed. The term $\left(\boldsymbol{A}^H\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H\left(\boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\right)$ is called *baseline noise* in the raw channel estimate "(Fimoff and Özen and Nereyanuru and Zoltowski and Hillery 2001, Hillery 2002)".

In order to further reduce the baseline noise we can incorporate the covariance matrix of the colored noise, $\boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}$, into the least squares equation of (3.27) yielding the *weighted* (or *generalized*) least squares solution

$$\begin{aligned}\hat{\boldsymbol{h}}_Q &= \left(\boldsymbol{A}^H(\sigma_\eta^2\boldsymbol{Q}\boldsymbol{Q}^H)^{-1}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H(\sigma_\eta^2\boldsymbol{Q}\boldsymbol{Q}^H)^{-1}\boldsymbol{y}_{[-N_a:N+N_c-1]} \\ &= \left(\boldsymbol{A}^H(\boldsymbol{Q}\boldsymbol{Q}^H)^{-1}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H(\boldsymbol{Q}\boldsymbol{Q}^H)^{-1}\boldsymbol{y}_{[-N_a:N+N_c-1]}\end{aligned} \tag{3.29}$$

where $\sigma_\eta^2\boldsymbol{Q}\boldsymbol{Q}^H$ is the covariance matrix of the colored noise vector $\boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}$, and σ_η^2 is the variance of the noise sequence $\eta[n]$. Note that the matrices, $\boldsymbol{A}, \boldsymbol{Q}$, that appear in Equation (3.29) are known, and the matrix $(\boldsymbol{A}^H(\boldsymbol{Q}\boldsymbol{Q}^H)^{-1}\boldsymbol{A})^{-1}\boldsymbol{A}^H(\boldsymbol{Q}\boldsymbol{Q}^H)^{-1}$ can be pre-computed and stored in the digital receiver to yield the initial channel estimate of $\hat{\boldsymbol{h}}_Q$, in place of the $\hat{\boldsymbol{h}}_c$.

3.3. Covariance Matrix Update Based Iterative Channel Estimation

The received signal model of Equation (2.35), which captures all the output corresponding to the N training symbols, is repeated here for convenience:

$$\boldsymbol{y}_{[-N_a:N+N_c-1]} \;=\; \boldsymbol{A}\boldsymbol{h} + \boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}.$$

We can denote the two terms on the right side of Equation (2.35) by

$$\boldsymbol{v} \;=\; \boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}. \tag{3.30}$$

Hence substituting Equation (3.30) into (2.35) we get

$$\boldsymbol{y}_{[-N_a:N+N_c-1]} \;=\; \boldsymbol{A}\boldsymbol{h} + \boldsymbol{v}. \tag{3.31}$$

As originally observed and documented first by C. Pladdy "(Pladdy and Özen and Nereyanuru and Zoltowski and Fimoff 2002)", we can *incorporate the covariance matrix of the vector* $\boldsymbol{v}$ into the generalized least squares solution. By noting the statistical independence of the random vectors $\boldsymbol{d}$ and $\boldsymbol{\eta}$, and also noting that both vectors are zero mean, the covariance matrix, $\boldsymbol{K}_{\boldsymbol{v}}$ of $\boldsymbol{v}$ is given by

$$\text{Cov}\{\boldsymbol{v}\} = \boldsymbol{K}_{\boldsymbol{v}} \equiv \frac{1}{2}E\{\boldsymbol{v}\boldsymbol{v}^H\} \;=\; \frac{\mathcal{E}_d}{2}\boldsymbol{H}\boldsymbol{H}^H + \sigma_\eta^2\boldsymbol{Q}\boldsymbol{Q}^H, \tag{3.32}$$

where $\mathcal{E}_d$ is the energy of the transmitted information symbols, and equals to 21 if the symbols $\{d_k\}$ are chosen from the set $\{\pm 1, \pm 3, \pm 5, \pm 7\}$. The generalized least squares objective function to be minimized is

$$J_{LS}(\boldsymbol{y}) \;=\; \left(\boldsymbol{y}_{[-N_a:N+N_c-1]} - \boldsymbol{A}\boldsymbol{h}\right)^H \boldsymbol{K}_{\boldsymbol{v}}^{-1} \left(\boldsymbol{y}_{[-N_a:N+N_c-1]} - \boldsymbol{A}\boldsymbol{h}\right). \tag{3.33}$$

Then the generalized least-squares solution to the model of Equation (3.31) which minimizes the objective function of (3.33) is given by

$$\hat{\boldsymbol{h}}_K \;=\; (\boldsymbol{A}^H\boldsymbol{K}_{\boldsymbol{v}}^{-1}\boldsymbol{A})^{-1}\boldsymbol{A}^H\boldsymbol{K}_{\boldsymbol{v}}^{-1}\boldsymbol{y}_{[-N_a:N+N_c-1]} \tag{3.34}$$

where $\boldsymbol{K_v}$ is given by Equation (3.32). The generalized least-squares channel estimate of Equation (3.34) is also called the *Best Linear Unbiased Estimate* (BLUE), due to fact that it achieves minimum variance among all linear unbiased estimates "(Seber 1977)". In our case the covariance matrix of the channel estimate is

$$\begin{aligned} \text{Cov}\{\hat{\boldsymbol{h}}_K\} &\equiv E\{(\hat{\boldsymbol{h}}_K - \boldsymbol{h})(\hat{\boldsymbol{h}}_K - \boldsymbol{h})^H\} \\ &= (\boldsymbol{A}^H \boldsymbol{K}_{\boldsymbol{v}}^{-1} \boldsymbol{A})^{-1}. \end{aligned} \tag{3.35}$$

When the covariance matrix of (3.35) is compared to any other estimators covariance matrix, the comparison is must be done in the sense $\boldsymbol{Z} - \boldsymbol{Y} > 0$ iff the matrix $[\boldsymbol{Z} - \boldsymbol{Y}]$ is positive definite.

The problem with Equation (3.34) is that the channel estimate $\hat{\boldsymbol{h}}_K$ is based on the covariance matrix $\boldsymbol{K_v}$, which is a function of the true channel impulse response vector $\boldsymbol{h}$ as well as the channel noise variance σ_η^2. In actual applications the BLUE channel estimate of Equation (3.34) can not be exactly obtained. Hence we need an iterative technique to calculate least squares estimate of (3.34) where every iteration produce an updated estimate of the covariance matrix as well as the noise variance.

3.3.1. Further Improvements to the Initial Channel Estimate

As discussed in Section 3 we can use either one of the initial channel estimates of Equations (3.27) or (3.29), the latter estimate is expected to produce slightly better channel estimates, where the performance measure is the normalized least-squares error which is defined by

$$\mathcal{E}_{NLS} = \frac{\|\boldsymbol{h} - \hat{\boldsymbol{h}}\|^2}{N_a + N_c + 1}. \tag{3.36}$$

We propose to further reduce the initial least squares estimation error by seeking an approximation in the form of assuming that the baseband representation of the physical channel $c(t)$ is a unit Dirac-delta function, that is assume that

$$c(t) = \delta(t) \tag{3.37}$$

which implies

$$h(t) \;=\; p(t) * c(t) = p(t). \tag{3.38}$$

Thus we can assume that our finite length channel impulse response vector can be (initially) approximated by

$$\tilde{\boldsymbol{h}} \;=\; [\underbrace{0, \cdots, 0}_{N_a - N_q}, p[-N_q], \cdots, p[-1], p[0], p[1], \cdots, p[N_q], \underbrace{0, \cdots, 0}_{N_c - N_q}]^T \tag{3.39}$$

with the assumptions of $N_a \geq N_q$ and $N_c \geq N_q$, that is the tail span of the composite pulse shape is well confined to within the assumed delay spread of $[-N_a T, N_c T]$. Then the approximation of (3.39) can be substituted into Equations (2.30-2.33) to yield an initial (approximate) channel convolution matrix $\tilde{\boldsymbol{H}}$ and is given by $\tilde{\boldsymbol{H}} = \tilde{\boldsymbol{\mathcal{H}}} \boldsymbol{S}^T$ where $\tilde{\boldsymbol{\mathcal{H}}}$ is formed as in Equation (2.30) with $\bar{\bar{\tilde{\boldsymbol{h}}}} = \boldsymbol{J}\tilde{\boldsymbol{h}}$. We can also assume a reasonable received Signal-to-Noise (SNR) ratio measured at the input to the matched filter which is given by

$$\mathrm{SNR} \;=\; \frac{\mathcal{E}_d \left\| (c(t) * q(t))|_{t=nT} \right\|^2}{\sigma_\eta^2} \tag{3.40}$$

$$\;=\; \frac{\mathcal{E}_d \left\| \boldsymbol{q} \right\|^2}{\sigma_\eta^2}. \tag{3.41}$$

For instance we can assume an approximate SNR of 20dB yielding an initial noise variance of

$$\tilde{\sigma}_\eta^2 \;=\; \frac{\mathcal{E}_d \|\boldsymbol{q}\|^2}{100}. \tag{3.42}$$

Then combining $\tilde{\boldsymbol{H}}$ and $\tilde{\sigma}_\eta^2$ we can pre-calculate the initial approximate covariance matrix where the covariance matrix of the approximate channel is given by

$$\tilde{\boldsymbol{K}}_{\boldsymbol{v}}(\tilde{\boldsymbol{H}}) \;=\; \frac{1}{2}\mathcal{E}_d \tilde{\boldsymbol{H}} \tilde{\boldsymbol{H}}^H + \tilde{\sigma}_\eta^2 \boldsymbol{Q}\boldsymbol{Q}^H, \tag{3.43}$$

which further leads to the initial channel estimate of

$$\hat{\boldsymbol{h}}_{\tilde{K}} \;=\; \left(\boldsymbol{A}^H [\tilde{\boldsymbol{K}}_{\boldsymbol{v}}(\tilde{\boldsymbol{H}})]^{-1} \boldsymbol{A} \right)^{-1} \boldsymbol{A}^H [\tilde{\boldsymbol{K}}_{\boldsymbol{v}}(\tilde{\boldsymbol{H}})]^{-1} \boldsymbol{y}_{[-N_a : N + N_c - 1]} \tag{3.44}$$

The equation (3.44) is called the approximate Best Linear Unbiased Estimator of the channel impulse response and this estimator will be referred to as ”aBLUE” in the sequel.

3.3.2. Statistical Analysis of Baseline Noise

The initial channel estimate of Equation (3.28) is repeated here for convenience

$$\hat{\boldsymbol{h}}_c \;=\; \boldsymbol{h} + \left(\boldsymbol{A}^H\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H\left(\boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\right),$$

where the channel estimate $\hat{\boldsymbol{h}}_c$ has the contributions due to unknown symbols prior to and after the training sequence, which are elements of the vector $\boldsymbol{d}$, as well as the additive channel noise; and the term $\left(\boldsymbol{A}^H\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H\left(\boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\right)$ is called *baseline noise* in the raw channel estimate ”(Fimoff and Özen and Nereyanuru and Zoltowski and Hillery 2001, Hillery 2002)”. Indeed we can summarize the baseline noise expression for the three different estimators of Equations (3.27), (3.29) and (3.44) by

$$\hat{\boldsymbol{h}} \;=\; \boldsymbol{h} + \boldsymbol{\xi} = \boldsymbol{h} + \boldsymbol{B}\left(\boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\right) \tag{3.45}$$

where the baseline noise vector $\boldsymbol{\xi}$ is defined by

$$\boldsymbol{\xi} \;=\; \boldsymbol{B}\left(\boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\right) \tag{3.46}$$

and the matrix $\boldsymbol{B}$ takes one of the three following different forms depending on the estimator used:

$$\boldsymbol{B} \;=\; \begin{cases} \left(\boldsymbol{A}^H\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H, & \text{for } \hat{\boldsymbol{h}}_c \text{ of Equation (3.27)} \\ \left(\boldsymbol{A}^H(\boldsymbol{Q}\boldsymbol{Q}^H)^{-1}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H(\boldsymbol{Q}\boldsymbol{Q}^H)^{-1}, & \text{for } \hat{\boldsymbol{h}}_Q \text{ of Equation (3.29)} \\ \left(\boldsymbol{A}^H[\tilde{\boldsymbol{K}}_{\boldsymbol{v}}(\tilde{\boldsymbol{H}})]^{-1}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H[\tilde{\boldsymbol{K}}_{\boldsymbol{v}}(\tilde{\boldsymbol{H}})]^{-1}, & \text{for } \hat{\boldsymbol{h}}_{\tilde{K}} \text{ of Equation (3.44)} \end{cases} \tag{3.47}$$

Although we can derive the exact probability distribution of the baseline noise term, we can alternatively make the assumption of *normality* (having Gaussian distribution) of the channel estimation error. This assumption can be asserted by invoking the

central limit theorem. We will state the central limit theorem without providing proof, since its proof can be found in several texts "(Casella and Berger 1990)".

Theorem 3.1 (*Central Limit Theorem*) *Let $X_1, X_2, \ldots$ be a sequence of iid random variables with $E\{X_i\} = \mu$ and $0 \leq Var\{X_i\} = \sigma^2 \leq \infty$. Define $\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$. Let $G_n(x)$ denote the distribution function of $\sqrt{n}(X_n - \mu)/\sigma$. Then for any x, $-\infty \leq x \leq \infty$,*

$$\lim_{n\to\infty} G_n(x) \;=\; \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy, \tag{3.48}$$

that is $\sqrt{n}(X_n - \mu)/\sigma$ has a limiting standard normal distribution (Gaussian with 0 mean and variance 1, $\mathcal{N}(0,1)$).

For kth tap of the channel estimate, $\hat{h}_k$, regardless of the estimator form, the channel estimation error term ξ_k consists of the sum of

- (scaled) linear combination of $2(N_c + N_a)$ random data;
- (scaled) linear combination of $N + N_a + N_c + 2L_q$ white Gaussian noise samples.

Thus we first invoke the Central Limit Theorem for the vector $\boldsymbol{Hd}$ and assert the *approximation*

$$\boldsymbol{Hd} \;\sim\; \mathcal{N}(\mathbf{0}, \frac{1}{2}\mathcal{E}_d \boldsymbol{H}\boldsymbol{H}^H). \tag{3.49}$$

The second term $\boldsymbol{Q\eta}_{[-N_a - L_q : N + N_c - 1 + L_q]}$ is already a Gaussian vector with zero mean and covariance matrix $\sigma_\eta^2 \boldsymbol{Q}\boldsymbol{Q}^H$, that is

$$\boldsymbol{Q\eta}_{[-N_a - L_q : N + N_c - 1 + L_q]} \;\sim\; \mathcal{N}(\mathbf{0}, \sigma_\eta^2 \boldsymbol{Q}\boldsymbol{Q}^H). \tag{3.50}$$

Then noting the independence of the random data symbols $\boldsymbol{d}$ and the noise vector $\boldsymbol{\eta}$ we conclude that the baseline noise vector $\boldsymbol{\xi}$ is has a limiting Gaussian distribution with zero mean and covariance matrix[1]

[1]The distribution in Equation (3.50) is exact; however the distributions of both (3.49) and (3.52) are approximations

$$\begin{aligned} \text{Cov}\{\boldsymbol{\xi}\} = \boldsymbol{K}_{\xi} &= \boldsymbol{B}(\frac{\mathcal{E}_d}{2}\boldsymbol{H}\boldsymbol{H}^H + \sigma_\eta^2 \boldsymbol{Q}\boldsymbol{Q}^H)\boldsymbol{B}^H \\ &= \boldsymbol{B}\boldsymbol{K}_{\boldsymbol{v}}\boldsymbol{B}^H \end{aligned} \tag{3.51}$$

that is

$$\begin{aligned} \boldsymbol{\xi} = \boldsymbol{B}(\boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a - L_q : N+N_c-1+L_q]}) &\sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{B}(\frac{\mathcal{E}_d}{2}\boldsymbol{H}\boldsymbol{H}^H + \sigma_\eta^2 \boldsymbol{Q}\boldsymbol{Q}^H)\boldsymbol{B}^H), \\ &\sim \mathcal{N}(\boldsymbol{0}, \underbrace{\boldsymbol{B}\boldsymbol{K}_{\boldsymbol{v}}\boldsymbol{B}^H}_{\boldsymbol{K}_\xi}) \end{aligned} \tag{3.52}$$

where $\boldsymbol{B}$ takes one of the appropriate forms as displayed in Equation (3.47), and $\boldsymbol{K}_{\boldsymbol{v}}$ is given in (3.32).

We conducted a simple experimentation to help visualize the normality of the baseline noise seen at, randomly selected, 418th tap value for $\hat{\boldsymbol{h}}_c$: Ten thousand realizations of the vectors $\boldsymbol{d}$ and $\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}$ are generated, then the histogram and the normality-plot of the real and imaginary parts of the 418th tap value of the term

$$\left(\boldsymbol{A}^H\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H\left(\boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\right)$$

are plotted in Figures 3.2 and 3.3 respectively. The purpose of a normality, or normal probability, plot is to graphically assess whether the data could come from a normal distribution. If the data are normal the plot will be linear. Other distribution types will introduce curvature in the plot.

It is also important to analyze the marginal distributions of the baseline noise as well as the distribution of the norm of the individual components of the baseline noise vector[2]. That is we are interested in finding the probability distributions of $|\xi_k|$ and $|\xi_k|^2$ where subscript k denotes the kth element of the baseline noise vector $\boldsymbol{\xi} = [\xi_1, \ldots, \xi_{N_a+N_c+1}]^T$. Based on (3.52) we can show that ξ_k has a Gaussian marginal distribution with zero mean and variance "(Casella and Berger 1990)"

$$\sigma_{\xi_k}^2 \equiv \frac{1}{2}E\{\xi_k\xi_k^*\} = \boldsymbol{1}_k^T\boldsymbol{B}\boldsymbol{K}_{\boldsymbol{v}}\boldsymbol{B}^H\boldsymbol{1}_k \tag{3.53}$$

[2]The probability distribution function of the norm, and square of the norm, of individual components of the baseline noise vector $\boldsymbol{\xi}$ will be useful later when we derive the constant false alarm rate (CFAR) based threshold.

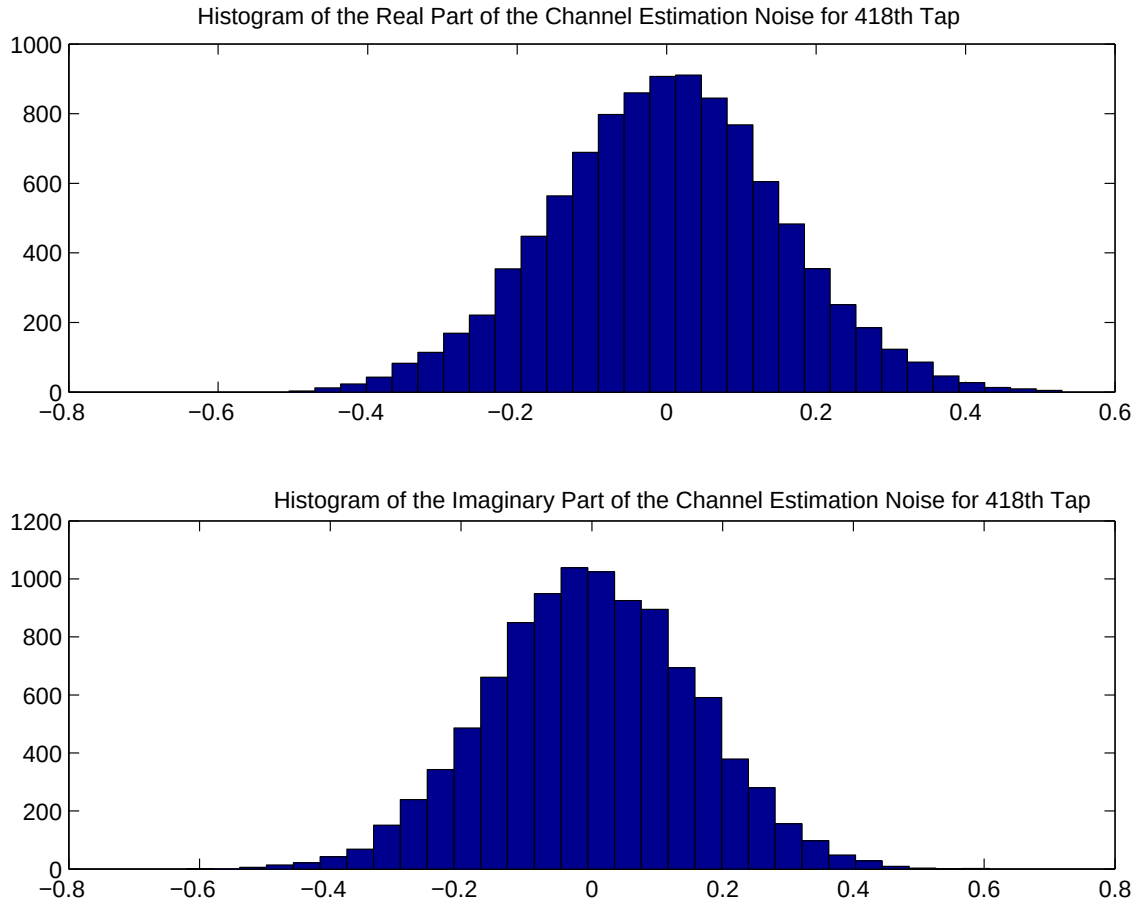

Figure 3.2: Ten thousand realizations of the vectors $\boldsymbol{d}$ and $\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}$ are generated; then the histogram of the real and imaginary parts of the 418th tap value of the term $\left(\boldsymbol{A}^H\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H\left(\boldsymbol{H}\boldsymbol{d}+\boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\right)$ are plotted.

that is

$$\xi_k = \mathbf{1}_k^T\boldsymbol{B}(\boldsymbol{H}\boldsymbol{d}+\boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}) \sim \mathcal{N}(0, \underbrace{\mathbf{1}_k^T\boldsymbol{B}\boldsymbol{K}_{\boldsymbol{v}}\boldsymbol{B}^H\mathbf{1}_k}_{\sigma^2_{\xi_k}}), \tag{3.54}$$

where $\boldsymbol{B}$ takes one of the appropriate forms as displayed in Equation (3.47), and $\mathbf{1}_k = [\underbrace{0,\ldots,0}_{k-1},1,0,\ldots,0]^T$ is the vector of zeros of appropriate dimension with a 1 at the kth position.

Now we state an important fact about the probability distributions of the norm and square-norm of the complex Gaussian random variables. The detailed discussion regarding their proofs and further properties can be found in ”(Stüber 1996)”.

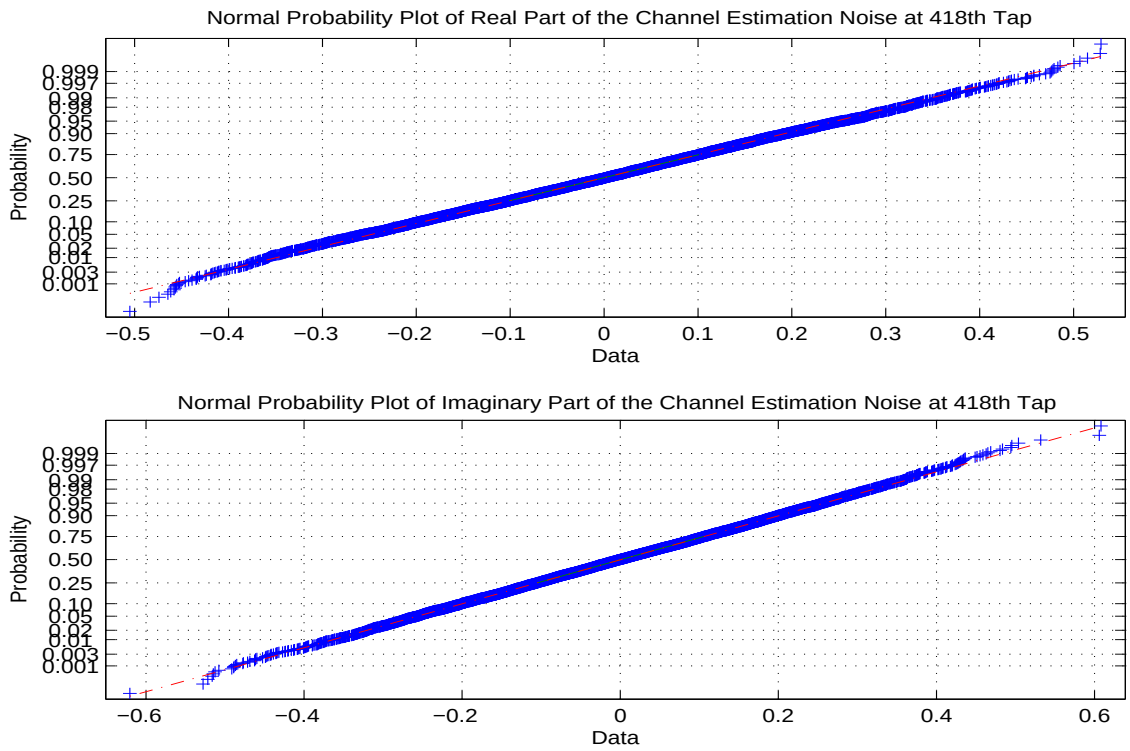

Figure 3.3: Ten thousand realizations of the vectors $\boldsymbol{d}$ and $\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}$ are generated; then the normality plot of the real and imaginary parts of the 418th tap value of the term $\left(\boldsymbol{A}^H\boldsymbol{A}\right)^{-1}\boldsymbol{A}^H\left(\boldsymbol{H}\boldsymbol{d}+\boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}\right)$ are plotted. The blue straight line corresponds to case where the samples are drawn from the exact normal distribution.

Lemma 3.1 *Let $\boldsymbol{\xi} = [\xi_1, \ldots, \xi_{N_a+N_c+1}]^T$ be a complex valued random vector, with statistically dependent real and imaginary parts ξ_r and ξ_q. Given that*

$$\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \underbrace{\boldsymbol{B}\boldsymbol{K}_{\boldsymbol{v}}\boldsymbol{B}^H}_{\boldsymbol{K}_\xi}) \tag{3.55}$$

then the random vector $\boldsymbol{S} = \sum_{i=1}^{k}\boldsymbol{\xi}_i\boldsymbol{\xi}_i^H$ is said to be Wishart distributed on k degrees of freedom "(Krzanowski 2000)", i.e.

$$\boldsymbol{S} \sim \mathcal{W}_k(\boldsymbol{K}_\xi, 1) \tag{3.56}$$

Lemma 3.2 *Let $\boldsymbol{\xi} = [\xi_1, \ldots, \xi_{N_a+N_c+1}]^T$ be a complex valued random vector, with statistically dependent real and imaginary parts ξ_r and ξ_q. Given that*

$$\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \underbrace{\boldsymbol{B}\boldsymbol{K}_{\boldsymbol{v}}\boldsymbol{B}^H}_{\boldsymbol{K}_\xi}) \tag{3.57}$$

then the random vector $\boldsymbol{Z} = \sum_{i=1}^{k} (\frac{\boldsymbol{\xi}}{\sigma_i})^2$ *is distributed according to the chi-square distribution on k degrees of freedom "(Hojbjerre and Sorensen and Eriksen and Andersen 1995)", i.e.,*

$$\boldsymbol{Z} \sim \chi_k^2 \tag{3.58}$$

The random vector $\boldsymbol{Z}$ given in Lemma 3.2 can be written as, (for $k = 1$)

$$\boldsymbol{Z} = (\frac{\boldsymbol{\xi}}{\sigma_\xi})^2 = \frac{1}{(\sigma_\xi)^2}\boldsymbol{\xi}\boldsymbol{\xi}^H. \tag{3.59}$$

Applying Lemma 3.1 to Equation (3.59) we conclude that, $\mathcal{W}_1(\boldsymbol{K}_\xi, 1)$ distribution is just $(\sigma_\xi)^2$ times the χ_k^2 distribution, so that the χ_k^2 distribution is a special case of the Wishart distribution "(Krzanowski 2000, Hojbjerre and Sorensen and Eriksen and Andersen 1995)".

If $u = \boldsymbol{z}^H\boldsymbol{\xi}$ is any linear transformation of $\boldsymbol{\xi}$, where $\boldsymbol{z}$ is nonzero vector, than u also has a normal distribution with zero mean and covariance matrix $\boldsymbol{z}^H\boldsymbol{K}_\xi\boldsymbol{z}$,

$$u \sim \mathcal{N}(0, \boldsymbol{z}^H\boldsymbol{K}_\xi\boldsymbol{z}). \tag{3.60}$$

It is also important to analyze the distribution of $\boldsymbol{z}^H\boldsymbol{S}\boldsymbol{z}$, where

$$\boldsymbol{S} = \boldsymbol{\xi}\boldsymbol{\xi}^H = \begin{bmatrix} \xi_1\xi_1^* & \xi_1\xi_2^* & \cdots & \xi_1\xi_{N_c+N_a+1}^* \\ \xi_2\xi_1^* & \xi_2\xi_2^* & \cdots & \xi_2\xi_{N_c+N_a+1}^* \\ \vdots & \vdots & \ddots & \vdots \\ \xi_{N_c+N_a+1}\xi_1^* & \xi_1\xi_2^* & \cdots & \xi_{N_c+N_a+1}\xi_{N_c+N_a+1}^* \end{bmatrix}, \tag{3.61}$$

$$\boldsymbol{z}^H\boldsymbol{S}\boldsymbol{z} = \boldsymbol{z}^H(\boldsymbol{\xi}\boldsymbol{\xi}^H)\boldsymbol{z} = (\boldsymbol{z}^H\boldsymbol{\xi})(\boldsymbol{\xi}^H\boldsymbol{z}) = uu^H = |u|^2 \tag{3.62}$$

Applying Lemma 3.1 to uu^H, we see that

$$\boldsymbol{z}^H \boldsymbol{S} \boldsymbol{z} \quad \sim \quad \mathcal{W}_1(\boldsymbol{z}^H \boldsymbol{K}_\xi \boldsymbol{z}, 1). \tag{3.63}$$

Since $\mathcal{W}_1(\boldsymbol{K}_\xi, 1)$ distribution is just $(\sigma_\xi)^2 = \boldsymbol{z}^H \boldsymbol{K}_\xi \boldsymbol{z}$ times the χ_1^2 distribution, $\boldsymbol{z}^H \boldsymbol{S} \boldsymbol{z}$ has also

$$\boldsymbol{z}^H \boldsymbol{S} \boldsymbol{z} \quad \sim \quad (\boldsymbol{z}^H \boldsymbol{K}_\xi \boldsymbol{z}) \chi_1^2. \tag{3.64}$$

Consider the case $\boldsymbol{z} = \mathbf{1}_k$, where $\mathbf{1}_k = [\underbrace{0, \ldots, 0}_{k-1}, 1, 0, \ldots, 0]^T$ is the vector of zeros of appropriate dimension with a 1 at the kth position. Then

$$\boldsymbol{z}^H \boldsymbol{S} \boldsymbol{z} = \mathbf{1}_k^H \boldsymbol{S} \mathbf{1}_k = s_{kk} = \xi_k \xi_k^* = |\xi_k|^2. \tag{3.65}$$

Also

$$\boldsymbol{z}^H \boldsymbol{K}_\xi \boldsymbol{z} = \mathbf{1}_k^T \boldsymbol{B} \boldsymbol{K}_{\boldsymbol{v}} \boldsymbol{B}^H \mathbf{1}_k = \sigma_{\xi_k}^2. \tag{3.66}$$

Finally, taking Equation (3.65) and Equation (3.66) in Equation (3.64), the distribution of $|\xi_k|^2$ is

$$|\xi_k|^2 \quad \sim \quad (\sigma_{\xi_k}^2) \chi_1^2, \tag{3.67}$$

and $|\xi_k|^2$ has a probability density function given by

$$p_{|\xi_k|^2}(r) = \frac{1}{\sigma_{\xi_k}^2 \sqrt{2\pi r}} exp(-\frac{r}{2\sigma_{\xi_k}^2}). \tag{3.68}$$

3.3.3. Approximations of Distribution

We can alternatively derive the probability density function of the channel estimation error under the assumption that the real and imaginary parts of the channel noise $\boldsymbol{\xi}$ are independent, in order to obtain a simpler thresholding rule in our further investigations.

Lemma 3.3 *Let $\xi = \xi_r + j\xi_q$ be a complex valued random variable, with statistically independent real and imaginary parts ξ_r and ξ_q. Given that ξ is Gaussian with 0 mean and variance $\sigma_\xi^2 = \sigma_{\xi_r}^2 = \sigma_{\xi_q}^2 = \frac{1}{2}E\{\xi\xi^*\}$, then the random variable $R = |\xi| = \sqrt{\xi_r^2 + \xi_q^2}$ is said to be Rayleigh distributed, and its density, denoted by $p_R(r)$, is given by*

$$p_R(r) = \begin{cases} \frac{r}{\sigma_\xi^2} e^{-\frac{r}{2\sigma_\xi^2}}, & r \geq 0 \\ 0, & r < 0. \end{cases} \tag{3.69}$$

Similarly the random variable defined by $Z = R^2 = |\xi|^2 = \xi_r^2 + \xi_q^2$ is exponentially distributed, and its density is given by

$$p_Z(z) = \begin{cases} \frac{1}{2\sigma_\xi^2} e^{-\frac{z}{2\sigma_\xi^2}}, & r \geq 0 \\ 0, & r < 0. \end{cases} \tag{3.70}$$

Applying Lemma 3.3 to (3.54) we immediately conclude that $|\xi_k|$ is a Rayleigh random variable with parameter $2\sigma_{\xi_k}^2$, and $|\xi_k|^2$ is an exponentially distributed random variable with parameter $2\sigma_{\xi_k}^2$, and their density functions are

$$p_{|\xi_k|}(r) = \begin{cases} \frac{r}{\sigma_{\xi_k}^2} e^{-\frac{r}{2\sigma_{\xi_k}^2}}, & r \geq 0 \\ 0, & r < 0. \end{cases} \tag{3.71}$$

$$p_{|\xi_k|^2}(r) = \begin{cases} \frac{1}{2\sigma_{\xi_k}^2} e^{-\frac{r}{2\sigma_{\xi_k}^2}}, & r \geq 0 \\ 0, & r < 0. \end{cases} \tag{3.72}$$

respectively, where $\sigma_{\xi_k}^2$ is defined by Equation (3.53). Note that

$$E\{|\xi_k|^2\} = \int_0^\infty r^2 p_{|\xi_k|}(r)dr = \int_0^\infty r p_{|\xi_k|^2}(r)dr = 2\sigma_{\xi_k}^2.$$

3.3.4. Iterative Algorithm to Calculate the Channel Estimate

The BLUE channel estimate is repeated here for convenience:

$$\hat{\boldsymbol{h}}_K = (\boldsymbol{A}^H \boldsymbol{K}_{\boldsymbol{v}}^{-1} \boldsymbol{A})^{-1} \boldsymbol{A}^H \boldsymbol{K}_{\boldsymbol{v}}^{-1} \boldsymbol{y}_{[-N_a:N+N_c-1]} \tag{3.73}$$

where $\boldsymbol{K}_{\boldsymbol{v}}$ is the covariance matrix given by

$$\boldsymbol{K_v} \equiv \frac{1}{2}E\{\boldsymbol{v}\boldsymbol{v}^H\} = \frac{\mathcal{E}_d}{2}\boldsymbol{H}\boldsymbol{H}^H + \sigma_\eta^2\boldsymbol{Q}\boldsymbol{Q}^H. \tag{3.74}$$

The problem with Equation (3.73) is that the channel estimate $\hat{\boldsymbol{h}}_K$ is based on the covariance matrix $\boldsymbol{K_v}$, which is a function of the true channel impulse response vector $\boldsymbol{h}$ that we are trying to estimate, as well as the channel noise variance σ_η^2. In actual applications the BLUE channel estimate of Equation (3.73) can not be exactly obtained. Hence we need an iterative technique to calculate least squares estimate of (3.73) where every iteration produce an updated estimate of the covariance matrix as well as the noise variance. Thus an only *approximate* BLUE estimate is possible.

Iterative algorithm to calculate the channel estimate, and noise variance is provided in Algorithm 1. In the very first step we can use any one of the three equations (3.27), (3.29), or (3.44) for an initial CIR estimate. However Equation (3.27) gives the highest normalized LS Error, and Equation (3.44) yields the lowest initial normalized LS error, hence Equation (3.44) is the most desirable of the three alternatives.

Once an initial channel estimate is obtained the channel noise variance can be obtained by

$$\widehat{\sigma^2}_\eta = \frac{1}{2\mathcal{E}_q(N - N_a - N_c)}|\hat{\boldsymbol{y}}_{[N_c:N-N_a]} - \boldsymbol{y}_{[N_c:N-N_a]}|^2 \tag{3.80}$$

which is based on the observation vector $\boldsymbol{y}_{[N_c:N-N_a]}$ of Equation (2.15), where $\mathcal{E}_q = |\boldsymbol{q}|^2$ and $\hat{\boldsymbol{y}}_{[N_c:N-N_a]} = \tilde{\boldsymbol{A}}\hat{\boldsymbol{h}}$. No subscript is used for $\hat{\boldsymbol{h}}$ to indicate that any channel estimate can be used at that stage to estimate the noise variance. Then an estimate of the covariance matrix of (3.74) can be obtained immediately since all the quantities involved in (3.74) are either known or has been estimated[3]. Then the covariance matrix estimate is used in Equation (3.73) to yield a better estimate of the channel. Then this process is iterated until a pre-specified number of iterations is reached, or a stopping criterion is achieved. A stopping criterion might be defined based on checking the norm of the

[3] $\mathcal{E}_d$ and $\boldsymbol{Q}$ are known; $\boldsymbol{H}$ and σ_η^2 are estimated. $\boldsymbol{H}$ is a function of the channel estimate $\hat{\boldsymbol{h}}$, and is constructed by following Equations (2.30-2.33), and the initial channel estimate $\hat{\boldsymbol{h}}$ to construct $\boldsymbol{H}$ can come from (3.27), (3.29), or (3.44).

Algorithm 1 Iterative Algorithm to obtain a CIR estimate via Generalized Least-Squares

[1] Get an initial CIR estimate using one of (3.27) or (3.29) or (3.44), and denote it by $\hat{\boldsymbol{h}}[0]$;

[2] Threshold the initial CIR estimate.

[3] Based on $\boldsymbol{y}_{[N_c:N-N_a]}$ of Equation (2.15), estimate the noise variance by

$$\widehat{\sigma^2}_{\eta}[0] \;=\; \frac{1}{2\mathcal{E}_q(N-N_a-N_c)}|\hat{\boldsymbol{y}}_{[N_c:N-N_a]}-\boldsymbol{y}_{[N_c:N-N_a]}|^2 \tag{3.75}$$

where $\mathcal{E}_q=|\boldsymbol{q}|^2$ and $\hat{\boldsymbol{y}}_{[N_c:N-N_a]}=\tilde{\boldsymbol{A}}\hat{\boldsymbol{h}}_{th}^{(0)}$;

[4]

for $k=1,\ldots,N_{iter}$ **do**

[4-a] Calculate the inverse of the (estimated) covariance matrix

$$\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[k] \;=\; \left[\frac{\mathcal{E}_d}{2}\boldsymbol{H}(\hat{\boldsymbol{h}}_{(th)}[k-1])\boldsymbol{H}^H(\hat{\boldsymbol{h}}_{(th)}[k-1])+\widehat{\sigma^2}_{\eta}[k-1]\boldsymbol{Q}\boldsymbol{Q}^H\right]^{-1}; \tag{3.76}$$

[4-b] Calculate

$$\hat{\boldsymbol{h}}_K[k]=(\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[k]\boldsymbol{A})^{-1}\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[k]\boldsymbol{y}_{[-N_a:N+N_c-1]}; \tag{3.77}$$

[4-c] Calculate (CFAR) $\varepsilon_{[n+N_a]}[k]$ for $-N_a\le n\le N_c$ according to Equation (4.20), and set

$$\hat{h}_{(th)}^{(k)}[n]=\begin{cases}0, & \text{if } \|\hat{h}_K^{(k)}[n]\|<\varepsilon_{[n+N_a]}[k]\\ \hat{h}_K^{(k)}[n], & \text{otherwise,}\end{cases} \tag{3.78}$$

for $-N_a\le n\le N_c$.

[4-d] Estimate the noise variance by

$$\widehat{\sigma^2}_{\eta}[k] \;=\; \frac{1}{2\mathcal{E}_q(N-N_a-N_c)}\|\hat{\boldsymbol{y}}_{[N_c:N-N_a]}^{(k)}-\boldsymbol{y}_{[N_c:N-N_a]}\|^2 \tag{3.79}$$

where $\hat{\boldsymbol{y}}_{[N_c:N-N_a]}^{(k)}=\tilde{\boldsymbol{A}}\hat{\boldsymbol{h}}_{th}[k]$.

end for

difference between the channel estimates obtained in the current iteration and the previous iteration.

There is one crucial detail that has not been discussed until this point. Right after obtaining a channel estimate, prior to using that channel estimate for noise variance, σ_η^2, calculation and prior to building the channel convolution matrix $\boldsymbol{H}$, the baseline noise has to be cleaned from the channel estimate. This cleaning can be achieved via thresholding. Previously we have used a fixed thresholding algorithm "(Özen and Zoltowski 2002)" in the form of

$$\text{set } \hat{h}_k^{(th)} = \begin{cases} 0, & \text{if } \|\hat{h}_k\|^2 < \varepsilon \\ \hat{h}_k, & \text{otherwise,} \end{cases} \tag{3.81}$$

for all k, to get rid of the baseline noise. We have observed that there can be significant performance loss if a fixed thresholding in the form of (3.81) is applied at every iteration. This performance loss is inevitable due to getting rid of significant amount of pulse tails embedded in the channel impulse response while getting rid of the baseline noise. To overcome this problem we propose two different thresholding schemes. The first one is called constant false alarm[4] rate (CFAR) based thresholding, and it is based on determining a thresholding bound based on the statistical distribution of the baseline noise which is already derived in Section(3.3.2). The second method is called the protection window based thresholding.

3.3.5. Other Approaches Background (Matching Pursuit)

There are some alternative channel estimation algorithms proposed as well as LS and a-BLUE methods. Matching Pursuit algorithm can be used for estimation of channels with large delay spread "(Cotter and Rao 2002)".

Recall the notation of multipath communication system again

$$\boldsymbol{y}_{[-N_a:N+N_c-1]} = \boldsymbol{A}\boldsymbol{h} + \boldsymbol{v}. \tag{3.82}$$

where

$$\begin{aligned} \boldsymbol{A} &= \mathcal{T}\left\{ [a_0, \cdots, a_{N-1}, \underbrace{0, \cdots, 0}_{N_a+N_c}]^T, [a_0, \underbrace{0, \cdots, 0}_{N_a+N_c}] \right\} \\ &= \begin{bmatrix} a_0 & 0 & \cdots & 0 \\ a_1 & a_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{N-1} & a_{N-2} & \cdots & a_0 \\ 0 & a_{N-1} & \cdots & a_1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{N-1} \end{bmatrix} \end{aligned} \tag{3.83}$$

[4] In statistical inference literature false alarm (rate) is referred to as the Type I error (probability).

In the MP algorithm ”(Cotter and Rao 2000)”, we first find the column in the matrix $\boldsymbol{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_N]$, which is best aligned with the signal vector $\boldsymbol{b_0} = \boldsymbol{y}_{[-N_a:N+N_c-1]}$ and this is denoted $\boldsymbol{a}_{k_1}$. Then the projection of $\boldsymbol{b_0}$ along $\boldsymbol{a}_{k_1}$ direction is removed from $\boldsymbol{b_0}$ and a new vector is obtained, called the residual, which is denoted by $\boldsymbol{b_1}$.

Now the column in $\boldsymbol{A}$, $\boldsymbol{a}_{k_2}$, which is best aligned with $\boldsymbol{b_1}$ is found and a new residual, $\boldsymbol{b_2}$, is formed.

In the pth iteration of the MP algorithm,the vector form $\boldsymbol{A}$ most closely aligned with the residual $\boldsymbol{b_{p-1}}$ is chosen according to

$$k_p = \arg\max_l \|\boldsymbol{P}_{a_l} \boldsymbol{b_{p-1}}\| \tag{3.84}$$

where $\boldsymbol{P}_{a_l}$ is the projection matrix onto the space spanned by $\boldsymbol{a_l}$

$$\boldsymbol{P}_{a_l} = \frac{\boldsymbol{a_l} \boldsymbol{a_l}^H}{\|a_l\|^2} \tag{3.85}$$

The new residual vector is then computed as

$$\boldsymbol{b_p} = \boldsymbol{b_{p-1}} - \boldsymbol{P}_{a_{k_p}} \boldsymbol{b_{p-1}} \tag{3.86}$$

and the tap value at position k_p is

$$\hat{\boldsymbol{h}}_{k_p} = \frac{\boldsymbol{a}_{k_p}^H \boldsymbol{b_{p-1}}}{\|a_l\|^2} \tag{3.87}$$

The iteration is repeated until a specified number of taps or the residual becomes sufficiently small i.e. $\|\boldsymbol{b_p}\| < \epsilon$.

CHAPTER 4

CONSTANT FALSE ALARM RATE (CFAR) BASED THRESHOLDING

4.1. Introduction

In radar systems, a constant false alarm rate (CFAR) detection is used to decide the presence of a target from a radar resolution cell ”(Levanon 1988)”. The purpose of CFAR design is maximization of detection probability while maintaining a desired false alarm rate. A CFAR detector should provide detection thresholds that are relatively immune to background noise and clutter variation with a CFAR. In parametric CFAR schemes, the most essential distinction of all CFAR algorithms is that the methods to form the average value representing the varying strength of clutter are different. The most widely used CFAR techniques are

- Cell averaging (CA) CFAR,
- Order Statistics (OS) CFAR.

Finn and Johnson ”(Finn and Johnson 1968)” first proposed the Cell-Averaging (CA) method. The adaptive method can play an effective part in many noise and clutter environments, and provide nearly the best ability of signal detection while preserving the enough constant false alarm rate. But in the existing interfering targets situation, the detection performance of CA decreases. Rohling ”(Rohling 1983)” presented the Order-Statistic (OS) CFAR detector; OS-CFAR possesses good ability to counter the multiple targets.

As an example, the CA-CFAR technique is outlined schematically in Figure 4.1. The returns from a given pulse are detected in a square-law detector, and a sample is taken from each channel tap. The channel tap under test is central tap. Its immediate neighbors are excluded from averaging process, because of fear of spillover from the channel tap under test. In the basic CA-CFAR the inputs from the M neighboring taps are averaged,

resulting an estimate of background noise(interference).the threshold is obtained by multiplying the estimated average by scaling factor α. This method works well when the background interference is statistically homogenous over range or Doppler, or both.

4.2. Using CFAR Techniques for Channel Estimation

The CFAR methods can be used to increase the estimation performance of channel impulse response.

Recall that the kth tap of the channel estimate vector can be expressed in the form

$$\hat{h}_k \;=\; h_k + \xi_k = h_k + \underbrace{\mathbf{1}_k^T \boldsymbol{B}\left(\boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a - L_q : N + N_c - 1 + L_q]}\right)}_{\xi_k}. \tag{4.1}$$

As has been presented the kth component of the channel estimation error vector ξ_k has a Gaussian distribution with zero mean and variance

$$\sigma_{\xi_k}^2 = \mathbf{1}_k^T \boldsymbol{B}(\frac{1}{2}\mathcal{E}_d \boldsymbol{H}\boldsymbol{H}^H + \sigma_\eta^2 \boldsymbol{Q}\boldsymbol{Q}^H)\boldsymbol{B}^H \mathbf{1}_k = \mathbf{1}_k^T \boldsymbol{B}\boldsymbol{K}_{\boldsymbol{v}}\boldsymbol{B}^H \mathbf{1}_k \tag{4.2}$$

where $\boldsymbol{B}$ takes one of the appropriate forms as displayed in Equation (3.47), and the random variable $|\xi_k|^2$ is exponentially distributed with parameter $2\sigma_{\xi_k}^2$.

The problem of deciding whether the kth tap estimate $\hat{h}_k$ is a zero tap or not can be formulated as a simple hypothesis testing problem. That is we can consider the following two hypotheses:

$$H_0 \;:\; \hat{h}_k = \xi_k, \tag{4.3}$$

$$H_1 \;:\; \hat{h}_k = h_k + \xi_k; \tag{4.4}$$

where under H_0 the hypothesis is that the kth channel tap is actually zero and we are observing only baseline noise, and under H_1 the hypothesis is that the channel tap is non-zero, and we are observing (non-zero) channel tap plus the baseline noise. From the earlier developments the probability distribution of the kth channel tap under null hypothesis H_0 is given by

$$H_0 \quad : \quad \hat{h}_k \sim \mathcal{N}(0, \sigma^2_{\xi_k}), \tag{4.5}$$

$$H_1 \quad : \quad \hat{h}_k \sim \mathcal{N}(h_k, \sigma^2_{\xi_k}). \tag{4.6}$$

After defining (4.5) and (4.6) we can come up with different decision rules on how to threshold the channel estimate $\hat{\boldsymbol{h}}$, however we choose to pursue the constant false alarm rate (CFAR) based thresholding. False alarm probability based decision rule is chosen so that the resulting threshold rule does not require any a priori knowledge of the distribution of the hypothesis H_1, it is solely based on H_0. False alarm rate is the probability of choosing H_1 when H_0 is true.

Our decision rule will be in the form of

$$\text{set } \hat{h}_k^{(th)} = \begin{cases} 0, & \text{if } |\hat{h}_k|^2 < \varepsilon_k \\ \hat{h}_k, & \text{otherwise.} \end{cases} \tag{4.7}$$

where $\hat{h}_k^{(th)}$ denotes the thresholded k'th tap of the channel impulse response.

Based on the rule of (4.7) the false alarm rate, denoted by p_{FA} is given by

$$\begin{aligned} p_{FA} &= \Pr\{|\hat{h}_k|^2 \geq \varepsilon_k | H_0 \text{ is true } \} \\ &= \int_{\varepsilon_k}^{\infty} \frac{1}{\sigma^2_{\xi_k}\sqrt{2\pi r}} exp(-\frac{r}{2\sigma^2_{\xi_k}}) dr \end{aligned} \tag{4.8}$$

By making change of variable $t = \frac{r}{2\sigma^2_{\xi_k}}$,

$$p_{FA} = \frac{1}{\sqrt{\pi\sigma^2_{\xi_k}}} \int_{\frac{\varepsilon_k}{2\sigma^2_{\xi_k}}}^{\infty} \frac{exp(-t)}{\sqrt{t}} dt \tag{4.9}$$

By making $x = \sqrt{t}$ change of variable we get

$$p_{FA} = \frac{2}{\sqrt{\pi\sigma^2_{\xi_k}}} \int_{\sqrt{\frac{\varepsilon_k}{2\sigma^2_{\xi_k}}}}^{\infty} exp(-x^2) dx$$

(4.10)

Recall the complementary error function, $erfc(x)$, "(Abramowitz and Stegun 1988)"

$$erfc(z) \;=\; \frac{2}{\sqrt{\pi}} \int_{z}^{\infty} exp(-t^2)dt, \tag{4.11}$$

then the p_{FA} can be written as

$$p_{FA} \;=\; \frac{1}{\sigma_{\xi_k}} erfc(\sqrt{\frac{\varepsilon_k}{2\sigma^2_{\xi_k}}}). \tag{4.12}$$

For the given level of false alarm probability p_{FA} the threshold level ε_k is given by

$$\varepsilon_k \;=\; 2\sigma^2_{\xi_k}[erfc^{-1}(\sigma_{\xi_k} p_{FA})]^2 \tag{4.13}$$

where $\sigma^2_{\xi_k}$ is given by (4.2).

Although we end up with an expression for the threshold of Equation (4.23), which should be applied to the channel estimate as in (4.7), we still have the problem of not knowing the true covariance matrix $\boldsymbol{B}(\frac{1}{2}\mathcal{E}_d \boldsymbol{H}\boldsymbol{H}^H + \sigma^2_\eta \boldsymbol{Q}\boldsymbol{Q}^H)\boldsymbol{B}^H$ and the kth diagonal element which we have denoted by $\sigma^2_{\xi_k}$. We can only have an estimate $\widehat{\sigma^2}_{\xi_k}$ available to be used in Equation (4.23). Thus it is natural to see some performance loss due to using the estimate $\widehat{\sigma^2}_{\xi_k}$ in place of the true variance as will be shown in the simulations. Indeed the thresholding step is going to be incorporated into the iterations of the channel estimation with covariance matrix updated at every iteration. Once the covariance matrix is updated at every iteration we would have a new, and presumably better, threshold ε_k since we will get a better estimate $\widehat{\sigma^2}_{\xi_k}$ at every iteration.

Note that the step **[4-b]** of the Algorithm is the main step to compute the channel estimate, and is repeated here for convenience (the square bracketed index $[n]$ denote the iteration step):

$$\hat{\boldsymbol{h}}_K[n] \;=\; \left(\boldsymbol{A}^H \widehat{\boldsymbol{K}}_{\boldsymbol{v}}^{-1}[n]\boldsymbol{A}\right)^{-1} \boldsymbol{A}^H \widehat{\boldsymbol{K}}_{\boldsymbol{v}}^{-1}[n]\boldsymbol{y}_{[-N_a:N+N_c-1]}, \tag{4.14}$$

where

$$\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[n] \;=\; \left[\frac{\mathcal{E}_d}{2}\boldsymbol{H}(\hat{\boldsymbol{h}}_{(th)}[n-1])\boldsymbol{H}^H(\hat{\boldsymbol{h}}_{(th)}[n-1] + \widehat{\sigma^2}_{\eta}[n-1]\boldsymbol{Q}\boldsymbol{Q}^H\right]^{-1} \quad (4.15)$$

is the inverse of the estimated covariance matrix of $\boldsymbol{v}$, and $\boldsymbol{H}(\hat{\boldsymbol{h}}_{(th)}[n-1])$ is the convolution matrix (with a "hole" inside) constructed as in Equations (2.30-2.33) from $\hat{\boldsymbol{h}}_{(th)}[n-1]$ which is the thresholded CIR vector estimated at the previous iteration. The baseline noise for the main channel estimation step of Equation (4.14) is

$$\boldsymbol{\xi} \;=\; \boldsymbol{B}(\boldsymbol{H}\boldsymbol{d} + \boldsymbol{Q}\boldsymbol{\eta}_{[-N_a-L_q:N+N_c-1+L_q]}), \quad (4.16)$$

where

$$\boldsymbol{B} \;=\; (\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[n]\boldsymbol{A})^{-1}\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[n]. \quad (4.17)$$

Thus the covariance matrix of the baseline noise at the n'th iteration step, denoted by $\boldsymbol{K}_{\xi}[n]$, is given by

$$\begin{aligned}\boldsymbol{K}_{\xi}[n] &= \boldsymbol{B}(\frac{\mathcal{E}_d}{2}\boldsymbol{H}\boldsymbol{H}^H + \sigma_{\eta}^2\boldsymbol{Q}\boldsymbol{Q}^H)\boldsymbol{B}^H \\ &= (\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[n]\boldsymbol{A})^{-1}\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[n]\boldsymbol{K}_{\boldsymbol{v}}\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[n]\boldsymbol{A}(\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[n]\boldsymbol{A})^{-1}. \quad (4.18)\end{aligned}$$

Since we can only use an estimate of the true covariance matrix $\boldsymbol{K}_{\boldsymbol{v}}$ (in the middle of Equation (4.18)), after the simplifications we get

$$\widehat{\boldsymbol{K}_{\xi}}[n] \;=\; (\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[n]\boldsymbol{A})^{-1}, \quad (4.19)$$

which is an estimate of the true covariance matrix of $\boldsymbol{\xi}$ of Equation (4.16) if we could have used the true covariance matrix $\boldsymbol{K}_{\boldsymbol{v}}$ in Equation (4.14) to begin with. Then the CFAR based threshold is given

$$\varepsilon_k \;=\; 2\widehat{\sigma_{\xi_k}}^2[erfc^{-1}(\widehat{\sigma_{\xi_k}}p_{FA})]^2 \quad (4.20)$$

where $\widehat{\sigma^2}_{\xi_k}$ is given by

$$\widehat{\sigma^2}_{\xi_k} \;=\; \mathbf{1}_k^T\widehat{\boldsymbol{K}_{\xi}}[n]\mathbf{1}_k = \mathbf{1}_k^T(\boldsymbol{A}^H\widehat{\boldsymbol{K}_{\boldsymbol{v}}}^{-1}[n]\boldsymbol{A})^{-1}\mathbf{1}_k. \quad (4.21)$$

4.2.1. Approximations and Further Simplifications

Recall that the kth tap of the channel estimate vector can be expressed in the form of Equation (4.1). In section (3.3.3), it has been shown that the norm and the norm-square of the channel estimation error ξ_k has been shown to obey Equations (3.71) and (3.72) under the assumption the real and imaginary parts of ξ_k are uncorrelated. Although this assumption is not true, this section shows how the threshold computation of (4.20) can be simplified.

Based on the rule of (4.7) the false alarm rate, denoted by p_{FA} is given by

$$\begin{aligned} p_{FA} &= \Pr\{|\hat{h}_k|^2 \geq \varepsilon_k | H_0 \text{ is true } \} \\ &= \int_{\varepsilon_k}^{\infty} \frac{1}{2\sigma^2_{\xi_k}} exp(-\frac{r}{2\sigma^2_{\xi_k}}) dr \\ p_{FA} &= exp(-\frac{\varepsilon_k}{2\sigma^2_{\xi_k}}). \end{aligned} \tag{4.22}$$

For the given level of false alarm probability p_{FA} the threshold level ε_k is given by

$$\varepsilon_k = -2\sigma^2_{\xi_k} ln(pfa) \tag{4.23}$$

where $\sigma^2_{\xi_k}$ is given by (4.2).

4.3. Cell Averaging (CA) CFAR Based Detection

In cell-averaging (CA) CFAR system the threshold adjustment for a specific component of channel estimation error (baseline noise) is based on the average detected input from its neighboring taps during the same pulse ”(Barboy and Lomes and Perkalski 1986)”. Thus we can use CA technique as a simple ”sliding-window” type estimator for the variance of the channel estimation error, $\widehat{\sigma^2}_{\xi_k}$.

$$y = \sum_{k=1}^{M} |\hat{\boldsymbol{h}}|^2 \tag{4.24}$$

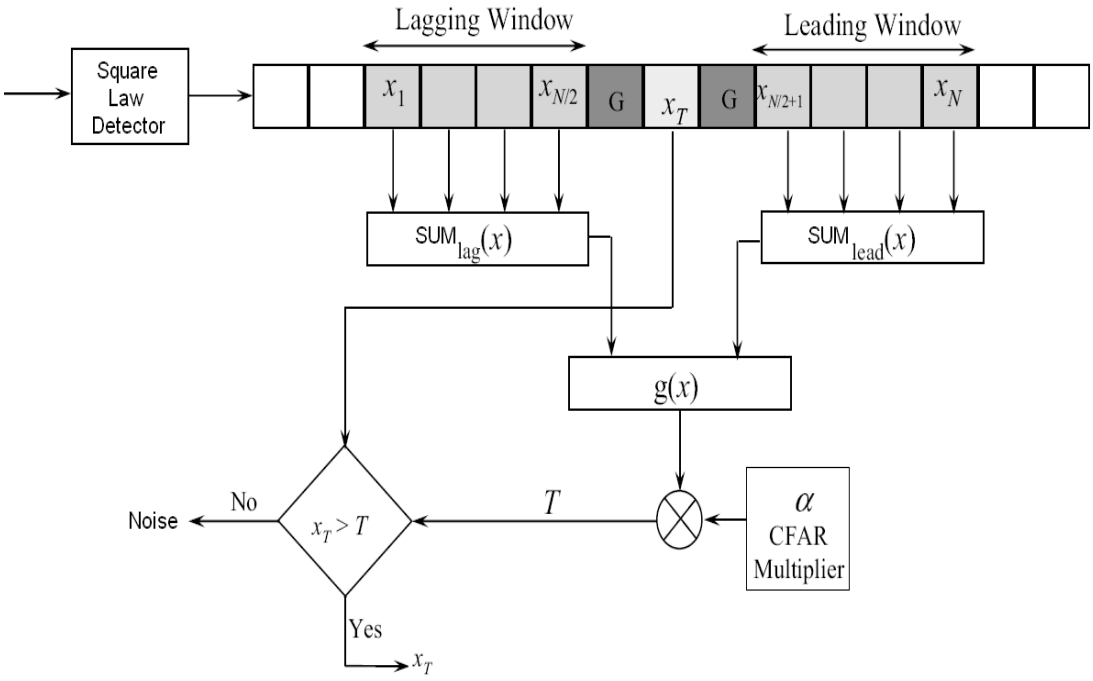

Figure 4.1: Cell-Averaging CFAR

$$Z_t = y\alpha/M \tag{4.25}$$

Note that Equation (4.24) gives us an estimate of the channel estimation error noise variance with the channel taps included in the summation. We can further improve this estimate by iterating cell averaging once (or possibly more) by first subtracting the thresholded channel estimate $\hat{h}_k^{(th)}$ from the initial channel estimate to get an estimate of the channel estimate error vector and then use the cell averaging procedure to this noise vector. First define

$$\widehat{\xi_k} = \hat{h}_k - \hat{h}_k^{(th)}. \tag{4.26}$$

We are proposing to use the CA technique to this improved estimated noise of Equation (4.26). Thus our CA algorithm is in the form of Algorithm (2).

The CA-CFAR technique is outlined schematically in Figure 4.1. The returns from a given pulse are detected in a square-law detector, and a sample is taken from

Algorithm 2 Cell Averaging CFAR Algorithm applied to any initial CIR estimate

[1] Get an initial CIR estimate using one of (3.27) or (3.29) or (3.44), and denote it by $\hat{\boldsymbol{h}}_{est}$; the length of $\hat{\boldsymbol{h}}_{est}$ is $N_a + N_c + 1$ where N_a and N_c denote the number of anti-causal and the causal taps of the channel.

[2]

for $k = N_a, \ldots, N_c$ **do**

[2-a] Estimate the noise variance of k^{th} tap by

$$\widehat{\sigma^2}_{\xi_k} = \frac{1}{2M} \sum_{s=k-M}^{k+M} |\hat{h}_{est}^{(s)}|^2 \tag{4.27}$$

where M denotes window size.

[2-b] Calculated the threshold by

$$\varepsilon_k = 2\widehat{\sigma^2}_{\xi_k} [erfc^{-1}(\widehat{\sigma}_{\xi_k} p_{FA})]^2 \tag{4.28}$$

[2-c] Threshold k_{th} tap of $\hat{\boldsymbol{h}}_{est}$ by using Equation (4.28) and set

$$\hat{h}_{th}^{(k)} = \begin{cases} 0, & \text{if } |\hat{h}_{est}^{(k)}|^2 < \varepsilon_k \\ \hat{h}_{est}^{(k)}, & \text{otherwise,} \end{cases} \tag{4.29}$$

end for

[3] Calculate

$$\widehat{\boldsymbol{\xi}}_k = \hat{\boldsymbol{h}}_{est} - \hat{\boldsymbol{h}}_{th}. \tag{4.30}$$

[4]

for $k = 1, \ldots, N_{iter}$ **do**

[4-a] Estimate the noise variance by

$$\widehat{\sigma^2}_{\xi_k} = \frac{1}{2M} \sum_{s=k-M}^{k+M} |\widehat{\xi}_k^{(s)}|^2 \tag{4.31}$$

[4-b] Calculated the threshold by

$$\varepsilon_k = 2\widehat{\sigma^2}_{\xi_k} [erfc^{-1}(\widehat{\sigma}_{\xi_k} p_{FA})]^2 \tag{4.32}$$

[4-c] Threshold k_{th} tap of $\hat{\boldsymbol{h}}_{est}$ by using Equation (4.32).

end for

each channel tap. The channel tap under test is central tap. Its immediate neighbors are excluded from averaging process, because of fear of spillover from the channel tap under test. In the basic CA-CFAR the inputs from the M neighboring taps are averaged, resulting an estimate of background noise(interference). The threshold is obtained by multiplying the estimated average by scaling factor α.

There are also modifications conventional CA-CFAR in the literature "(Hansen and Sawyers 1980)" and "(Smith and Varshney 2000)". One of these is Greater-Of CA-CFAR (CAGO-CFAR). Only difference in this approach is the way of estimating the noise variance in $[\mathbf{2}-\mathbf{a}]$ and $[\mathbf{4}-\mathbf{a}]$ in Algorithm (2).

$$\widehat{\sigma^2}_{\xi_k} = \frac{1}{2M}\max(\sum_{s=k-M}^{k}|\widehat{\xi}_k^{(s)}|^2, \sum_{s=k}^{k+M}|\widehat{\xi}_k^{(s)}|^2) \tag{4.33}$$

According to this modification, the $g(x)$ in Figure 4.1, for CAGO-CFAR method, will be

$$g(x) = \max(SUM_{lag}(x), SUM_{lead}(x)) \tag{4.34}$$

The Performance of CA-CFAR and CAGO-CFAR methods are compared in Figure 4.2.

4.4. Order Statistic (OS) CFAR Based Detection

In order to prove that order statistics is indeed a CFAR technique, we must first find the PDF of the threshold random variable (r.v.) and then average the probability of detection over all possible threshold values in a non-signal situation.

M is the total number of reference taps now ranked according to their input level

$$h_1 \leq h_2 \leq \cdots \leq h_K \leq \cdots \leq h_M \tag{4.35}$$

K is the rank of the sample which is selected to determine the threshold ε

$$\varepsilon = \alpha h_K \tag{4.36}$$

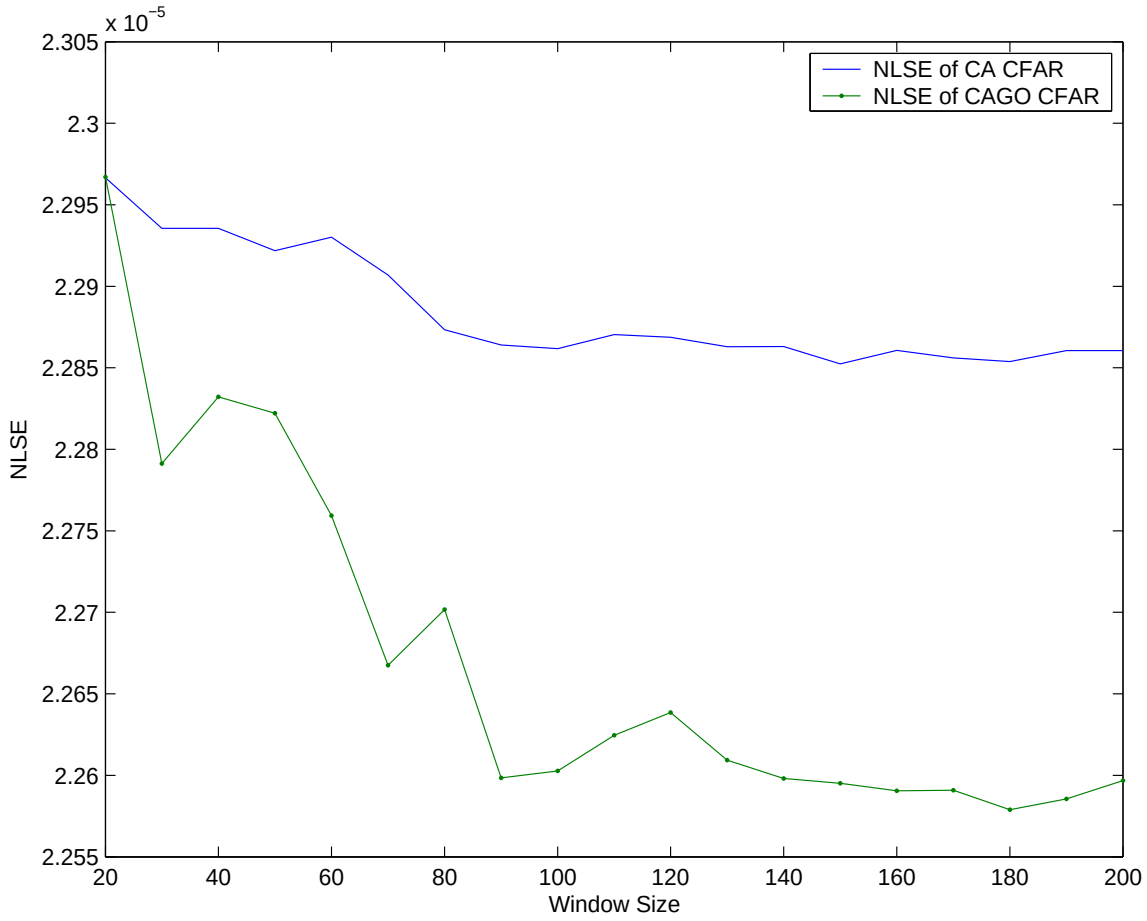

Figure 4.2: Normalized Least Square Error (NLSE) of CA-CFAR and CAGO-CFAR for different window size

The probability of a noise sample crossing the threshold ε_K is

$$P(h \geq \varepsilon_K \mid \varepsilon_K) = \int_{\varepsilon_K}^{\infty} p_K(r)dr \tag{4.37}$$

When h is a r.v. with PDF $p(h)$ and a distribution function $P(h)$, then the K_{th} ranked sample (out of a total of M samples) has a PDF as follows:

$$p_K(r) = K\binom{M}{K}[P(r)]^{K-1}[1-P(r)]^{M-K}p(r) \tag{4.38}$$

The random variable $|\xi_k|^2$ has a probability density function given by

$$p_{|\xi_k|^2}(r) = \frac{1}{\sigma_{\xi_k}^2\sqrt{2\pi r}} exp(-\frac{r}{2\sigma_{\xi_k}^2}) \tag{4.39}$$

and a probability distribution function given by

$$P_{|\xi_k|^2}(r) = \int_0^z p_{\|\xi_k\|^2}(r)dr = \frac{1}{\sigma_{\xi_k}} erf(\sqrt{\frac{r}{2\sigma_{\xi_k}^2}}) \tag{4.40}$$

Using Equation (4.39) and Equation (4.40) in Equation (4.38), we get the PDF of the ranked sample

$$p_K(r) = K\binom{M}{K}[\frac{1}{\sigma_{\xi_k}} erf(\sqrt{\frac{r}{2\sigma_{\xi_k}^2}})]^{K-1}[1 - \frac{1}{\sigma_{\xi_k}} erf(\sqrt{\frac{r}{2\sigma_{\xi_k}^2}})]^{M-K} \frac{e^{-\frac{r}{2\sigma_{\xi_k}^2}}}{\sqrt{2\pi r}\sigma_{\xi_k}^2} \tag{4.41}$$

Since it is difficult to find a closed form solution for p_{FA} by using Equation (4.41), we can alternatively use the approximate distribution of $|\xi_k|^2$ given in the Section (3.3.3).

In the Section (3.3.3), we have shown that the random variable $|\xi_k|^2$ is exponentially distributed with parameter $2\sigma_{\xi_k}^2$

$$p_{|\xi_k|^2}(r) \quad = \quad \begin{cases} \frac{1}{2\sigma_{\xi_k}^2} e^{-\frac{r}{2\sigma_{\xi_k}^2}}, & r \geq 0 \\ 0, & r < 0. \end{cases} \tag{4.42}$$

and

$$P_{|\xi_k|^2}(r) = \int_0^z p_{|\xi_k|^2}(r)dr = 1 - e^{-\frac{r}{2\sigma_{\xi_k}^2}}. \tag{4.43}$$

Using Equation (4.42) and Equation (4.43) in Equation (4.38), we get the PDF of the ranked sample

$$p_K(r) = K\binom{M}{K}\frac{r}{2\sigma_{\xi_k}^2}[1 - e^{-\frac{r}{2\sigma_{\xi_k}^2}}]^{K-1}[e^{-\frac{r}{2\sigma_{\xi_k}^2}}]^{M-K}\frac{1}{2\sigma_{\xi_k}^2}e^{-\frac{r}{2\sigma_{\xi_k}^2}}. \tag{4.44}$$

The probability of false alarm (p_{FA}) will be

$$p_{FA} \quad = \quad \int_0^\infty e^{-\alpha\frac{r}{2\sigma_{\xi_K}^2}} p_K(r)dr. \tag{4.45}$$

By putting Equation (4.44) as $p_K(r)$ and $u = e^{-\frac{r}{2\sigma_{\xi_k}^2}}$, p_{FA} take the form

$$p_{FA} = K\binom{M}{K}\int_0^1 u^{\alpha+M-K}(1-u)^{K-1}du \tag{4.46}$$

The integral found in Equation (4.46) can also be written by help of **Beta function** "(Abramowitz and Stegun 1988)".

The **Beta function**, also called the **Euler integral** of the first kind, is a special function defined by

$$\mathbf{B}(x,y) = \int_0^1 t^{x-1}(1-t)^{y-1}dt \tag{4.47}$$

for $\Re(x), \Re(y) > 0$.

Beta function has many other forms, including:

$$\mathbf{B}(x,y) = \frac{\mathbf{\Gamma}(x)\mathbf{\Gamma}(y)}{\mathbf{\Gamma}(x+y)} \tag{4.48}$$

where $\mathbf{\Gamma}$ is the **Gamma function**. By using Equation (4.47) and Equation (4.48), p_{FA} can be written as

$$\begin{aligned} p_{FA} &= K\binom{M}{K}\mathbf{B}((\alpha+M-K+1),K) \\ &= K\binom{M}{K}\frac{\mathbf{\Gamma}(\alpha+M-K)\mathbf{\Gamma}(K)}{\mathbf{\Gamma}(\alpha+M+1)} \end{aligned} \tag{4.49}$$

The **Gamma function** reduces to the factorial for a positive integer argument,

$$\mathbf{\Gamma}(x+1) = (x)! \tag{4.50}$$

where x is real integer.

Then for integer values of α, p_{FA} in Equation (4.49) becomes,

$$p_{FA} = K\binom{M}{K}\frac{(\alpha+M-K)!(K-1)!}{(\alpha+M)!}$$

$$= \frac{M(M-1)...(M-K+1)}{(M+\alpha)(M+\alpha-1)...(M+\alpha-K+1)}. \quad (4.51)$$

For $\alpha = 1$,

$$p_{FA} = \frac{(M-K+1)}{(M+1)} \cong \frac{(M-K)}{M}. \quad (4.52)$$

For $\alpha = 2$,

$$p_{FA} = \frac{(M-K+2)(M-K+1)}{(M+2)(M+1)} \cong \frac{(M-K)^2}{M^2}. \quad (4.53)$$

For $\alpha = 3$,

$$p_{FA} = \frac{(M-K+3)(M-K+2)(M-K+1)}{(M+3)(M+2)(M+1)} \cong \frac{(M-K)^3}{M^3}. \quad (4.54)$$

Finally, for any α,

$$p_{FA} \cong \frac{(M-K)^\alpha}{M^\alpha} = (1-\frac{K}{M})^\alpha, \quad (4.55)$$

therefore

$$(p_{FA})^{1/\alpha} = 1 - \frac{K}{M} \quad (4.56)$$

and

$$K = M[1-(p_{FA})^{1/\alpha}]. \quad (4.57)$$

We have compared the performance of approximated K value given in Equation (4.57) with the exact values obtained from Equation (4.49), and the comparison is tabulated in Table (4.1). It is clear that there is small difference at selected K value for given α and p_{FA}. Hence we will use Equation (4.57) to obtain K for any selected α and p_{FA}. We also provide relationship between α-K for different p_{FA} values in Figure (4.3).

	$p_{FA} = 1e-4$		$p_{FA} = 1e-5$		$p_{FA} = 1e-6$	
α	K_{ideal}	K_{app}	K_{ideal}	K_{app}	K_{ideal}	K_{app}
$\alpha = 1$	433	430	463	460	482	479
$\alpha = 2$	311	308	353	350	387	383
$\alpha = 3$	238	234	278	274	312	308
$\alpha = 4$	192	188	228	224	260	256
$\alpha = 5$	161	157	193	188	222	217
$\alpha = 6$	139	135	168	163	194	188
$\alpha = 7$	122	118	148	143	172	166
$\alpha = 8$	109	105	133	128	155	149
$\alpha = 9$	98	94	120	115	141	135
$\alpha = 10$	90	86	110	105	129	123
$\alpha = 15$	63	59	78	72	92	86
$\alpha = 20$	49	45	61	55	72	66
$\alpha = 25$	40	36	50	45	59	53
$\alpha = 30$	34	30	43	37	51	45
$\alpha = 35$	30	26	37	32	46	38
$\alpha = 40$	27	23	33	28	40	34
$\alpha = 45$	24	20	30	25	36	30
$\alpha = 50$	22	18	28	23	33	27
$\alpha = 100$	13	9	16	11	20	13
$\alpha = 200$	8	4	10	5	12	7
$\alpha = 300$	6	3	8	4	10	4
$\alpha = 400$	5	2	7	3	8	3
$\alpha = 500$	5	2	6	2	7	2

Table 4.1: K values computed from Equation (4.49)

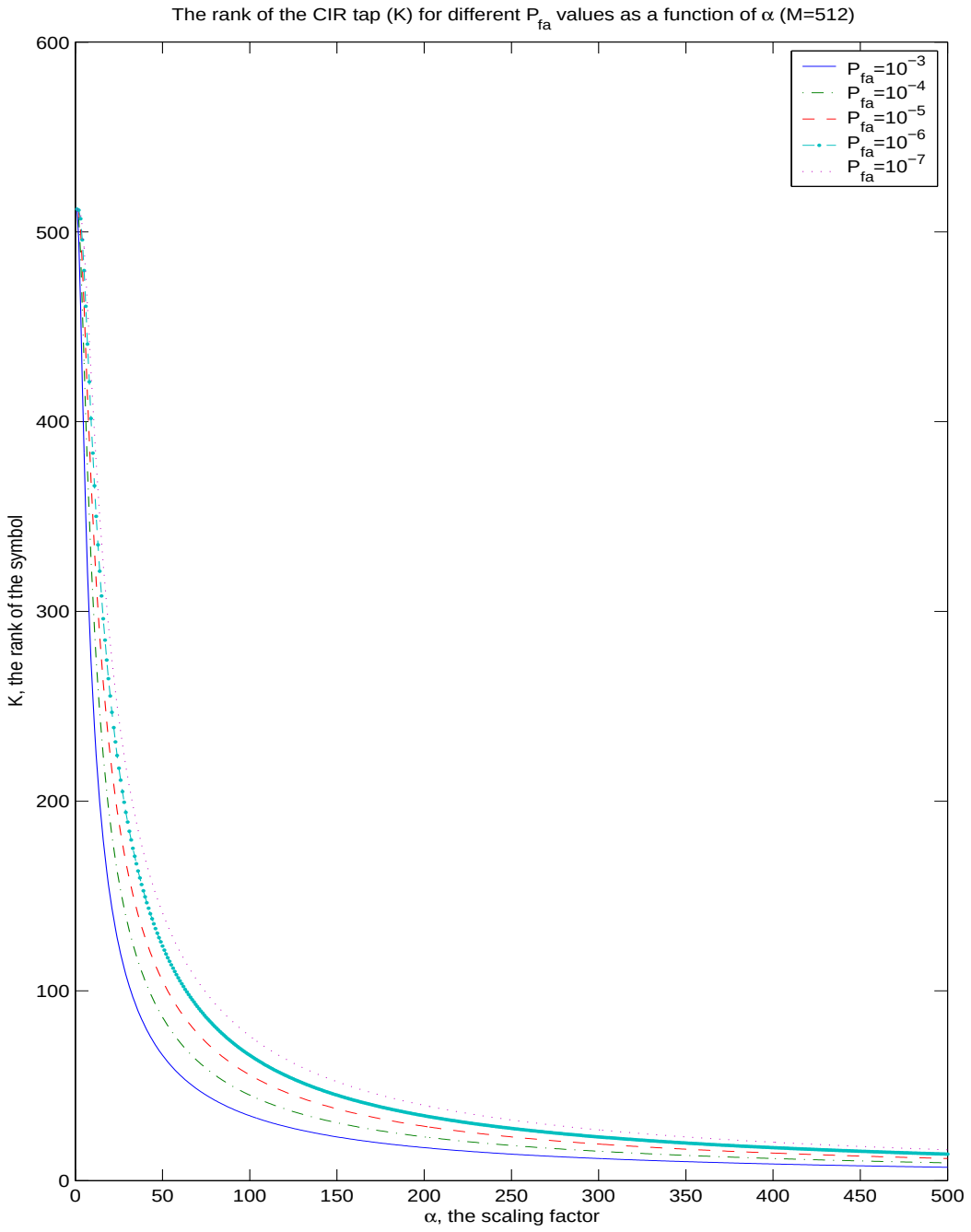

Figure 4.3: K versus alpha

4.5. Simulations

We provide CIR estimations of Least Squares, aBLUE, MP and aBLUE with CFAR (CA and OS based) thresholded. These estimations use 9 channels given at Appendix A. At each of Figures 4.4-4.12; Part(a) shows the real part of CIR for Channel at 25 dB SNR; Part (b) shows LS Estimate of real part of CIR for Channel based on Equation (3.28). Part (c) shows aBLUE Estimate real part of Channel based on Equation (3.44). Part (d) shows MP Estimate of real part of CIR for Channel 12 based on Equation (3.87). Part (e) shows aBLUE Estimate of real part of CIR for Channel thresholded by CA-CFAR method based on window of 60 tap. Part (f) shows aBLUE Estimate of real part of CIR for Channel thresholded by OS-CFAR method where K=255, α=20 and p_{FA}=1e-6.

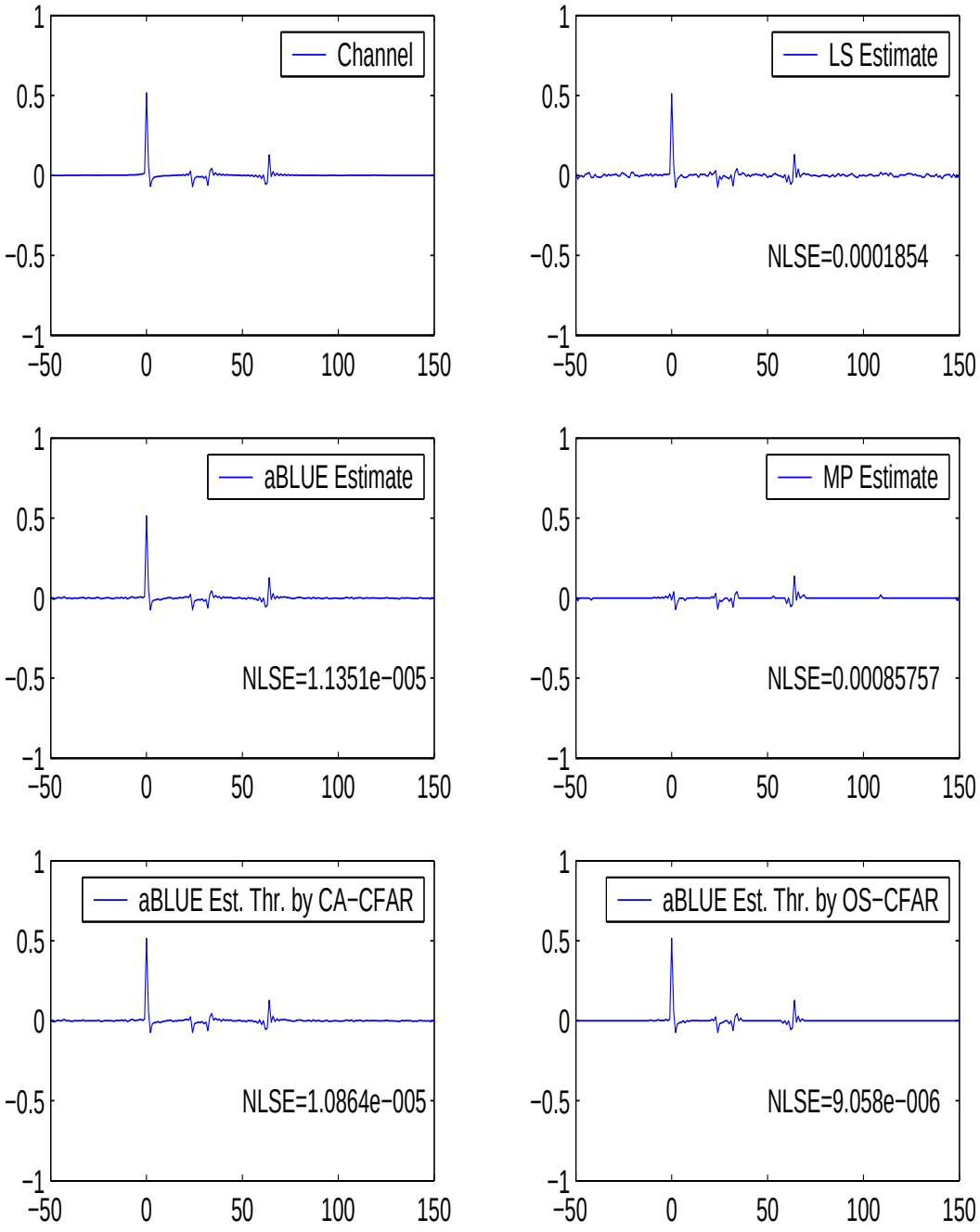

Figure 4.4: CIR estimations for Channel 1

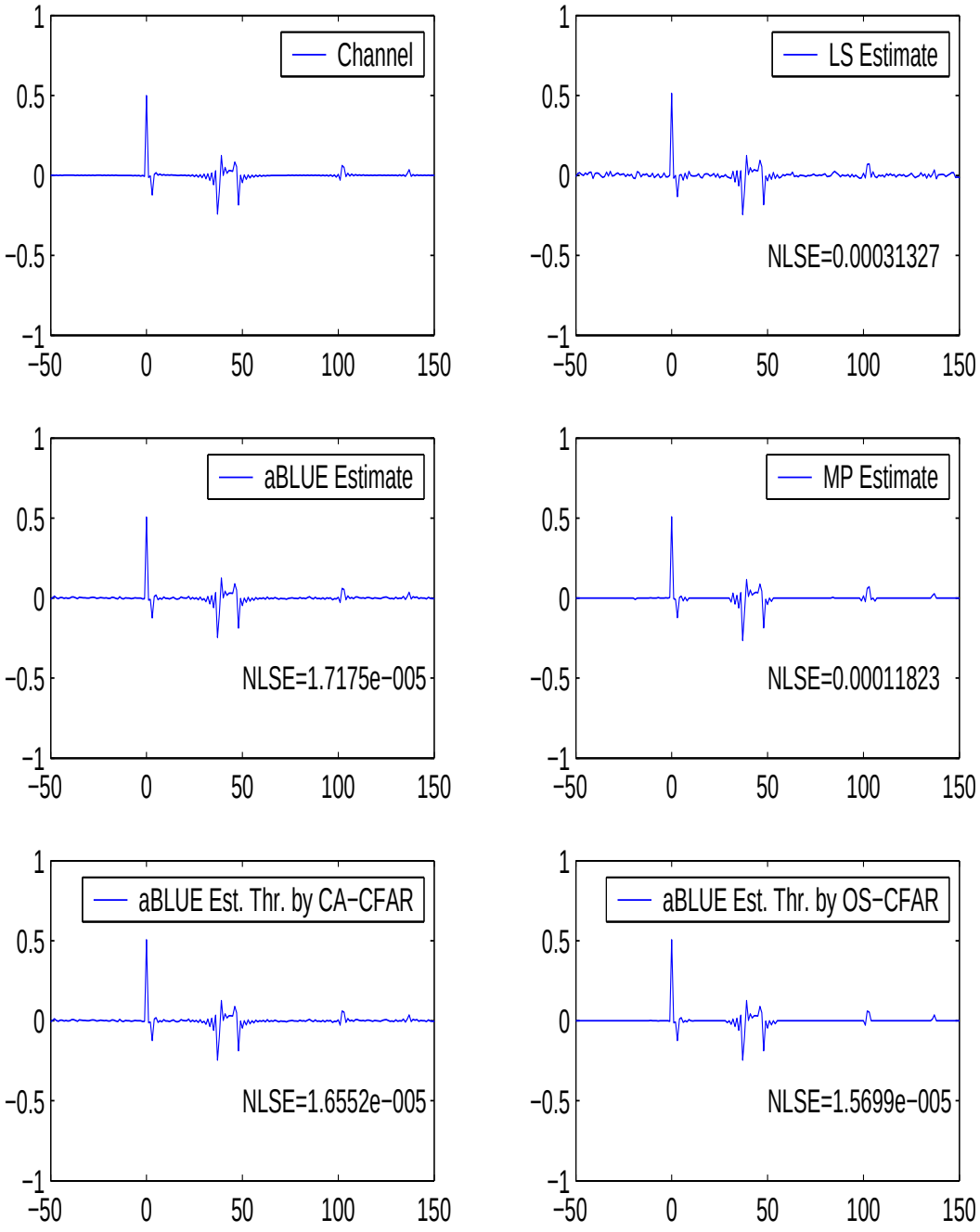

Figure 4.5: CIR estimations for Channel 2

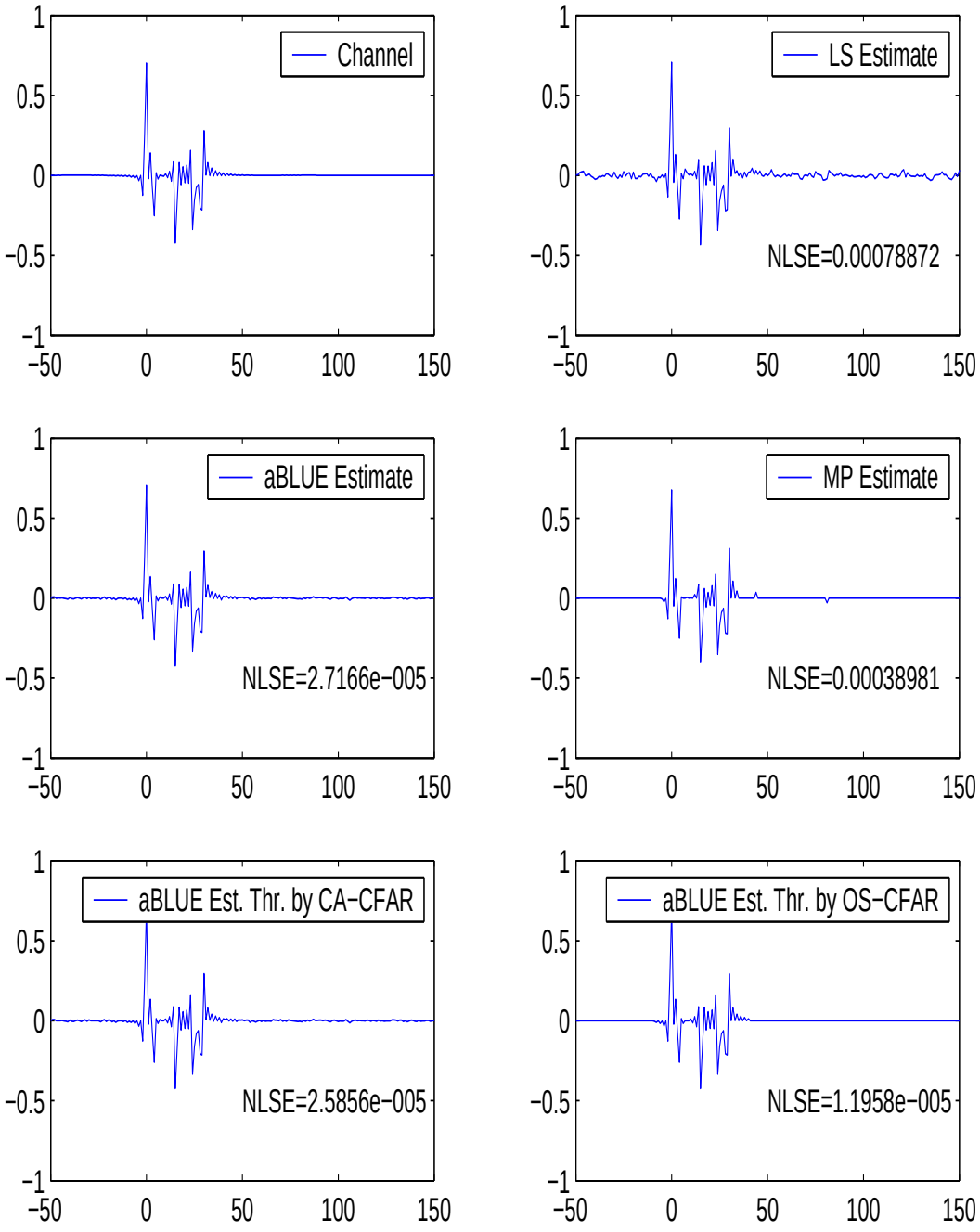

Figure 4.6: CIR estimations for Channel 3

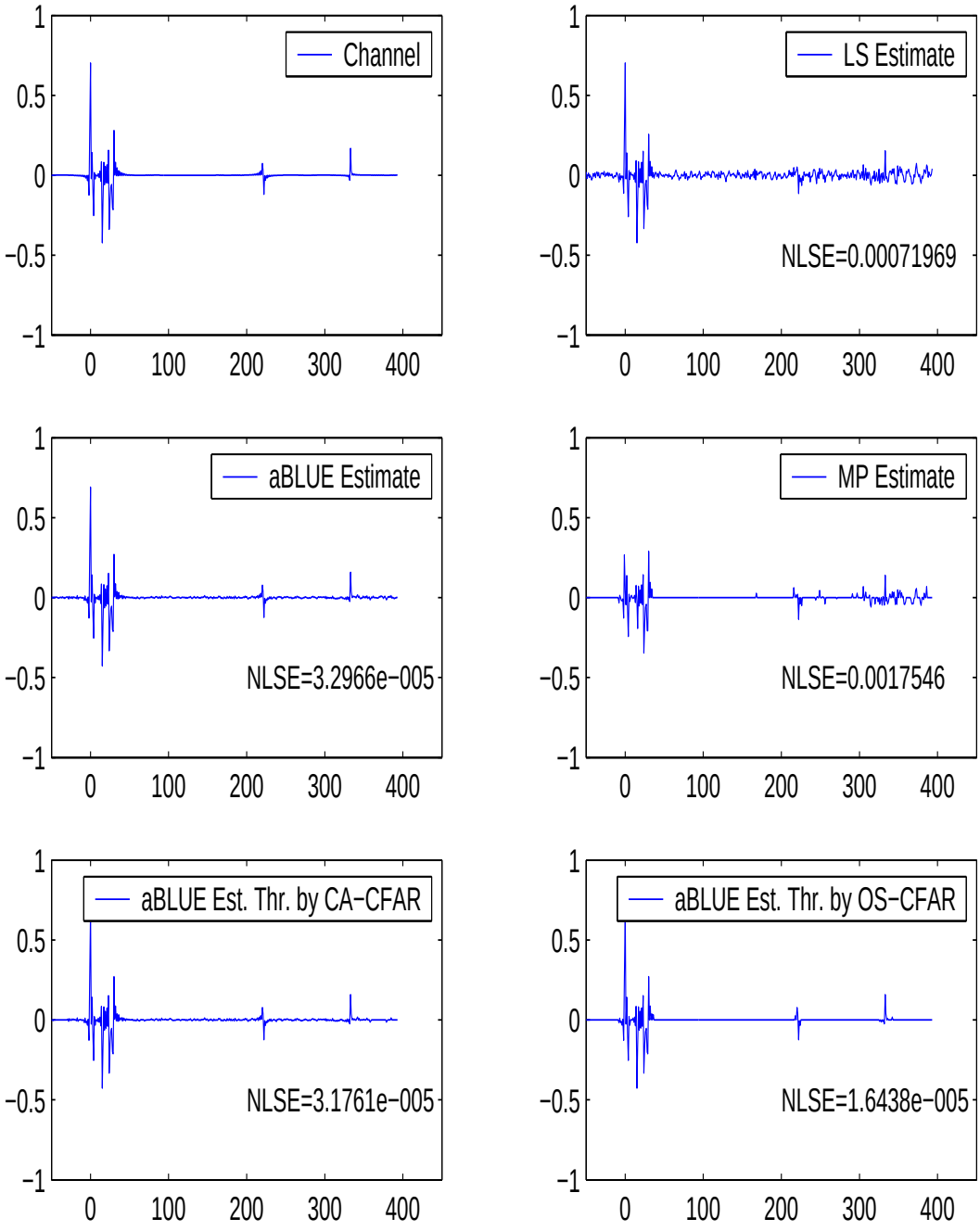

Figure 4.7: CIR estimations for Channel 3-plus

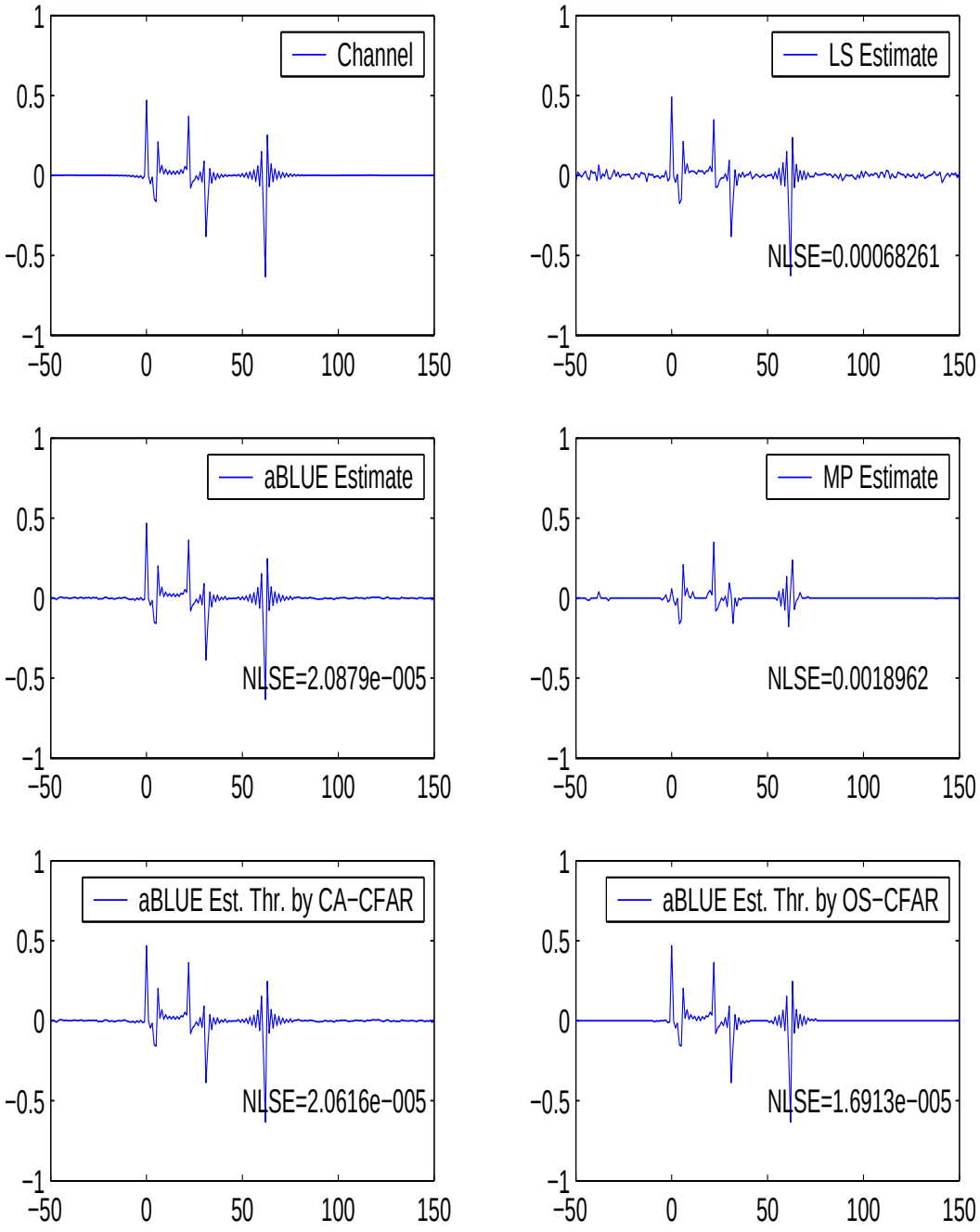

Figure 4.8: CIR estimations for Channel 4

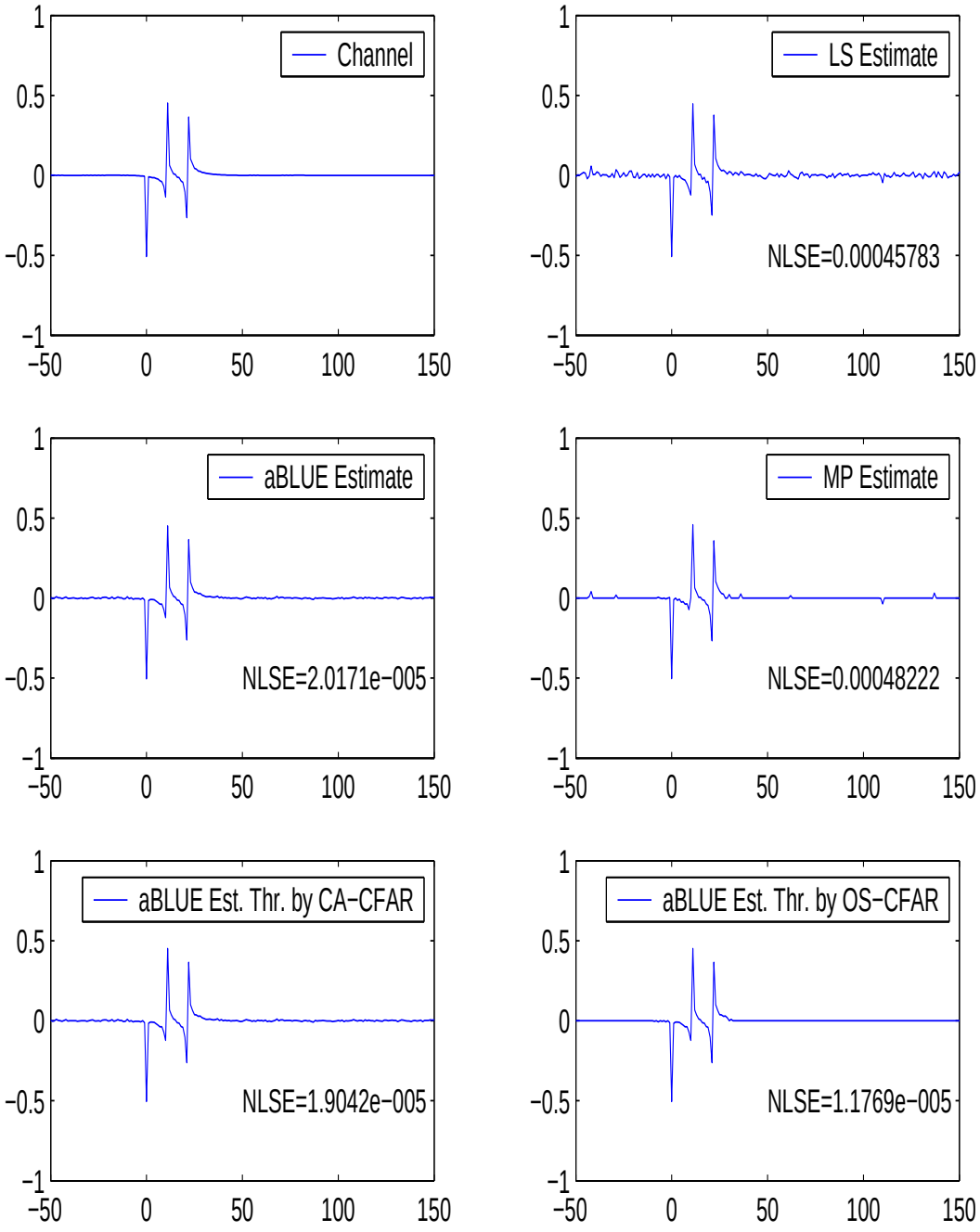

Figure 4.9: CIR estimations for Channel 5

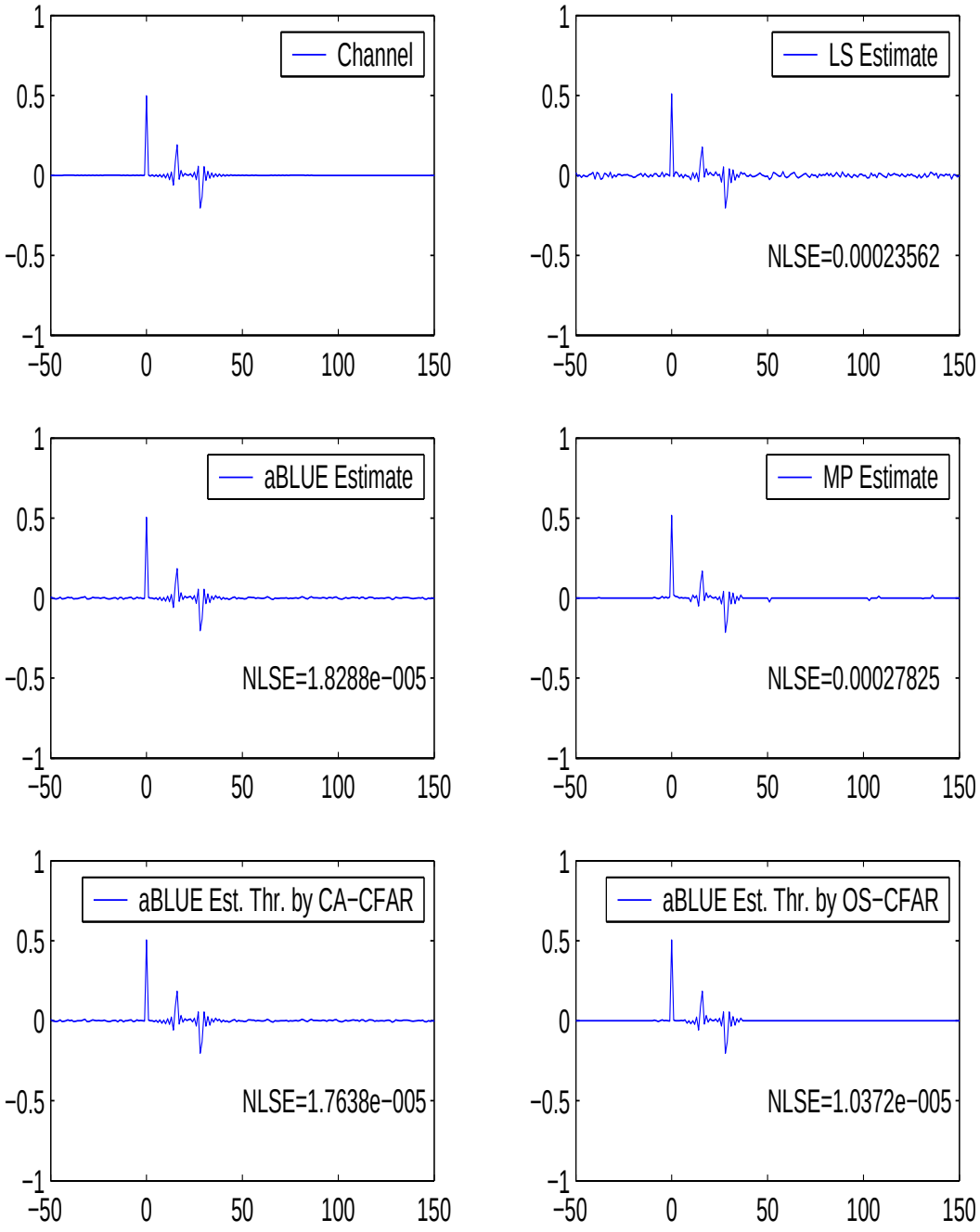

Figure 4.10: CIR estimations for Channel 6

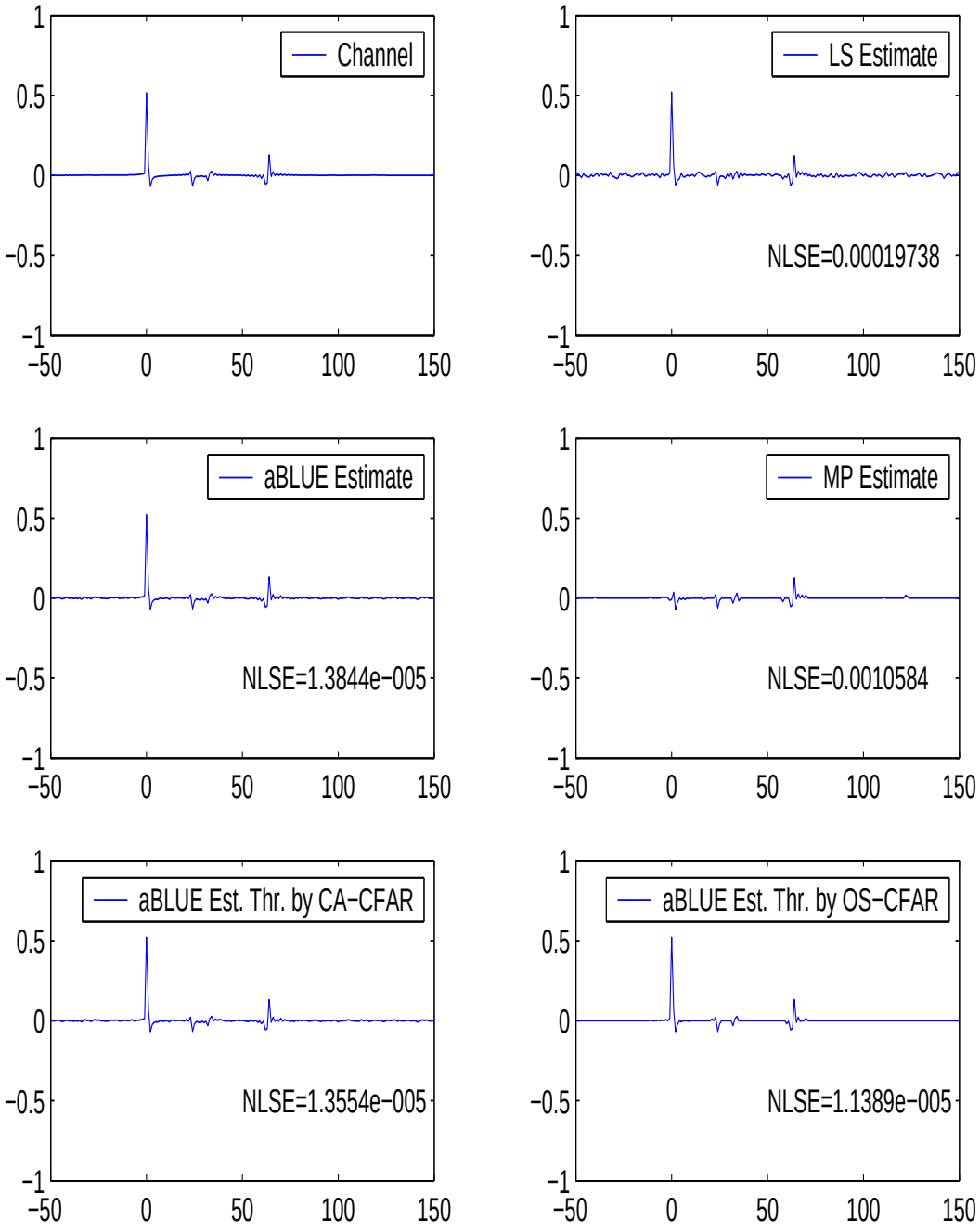

Figure 4.11: CIR estimations for Channel 7

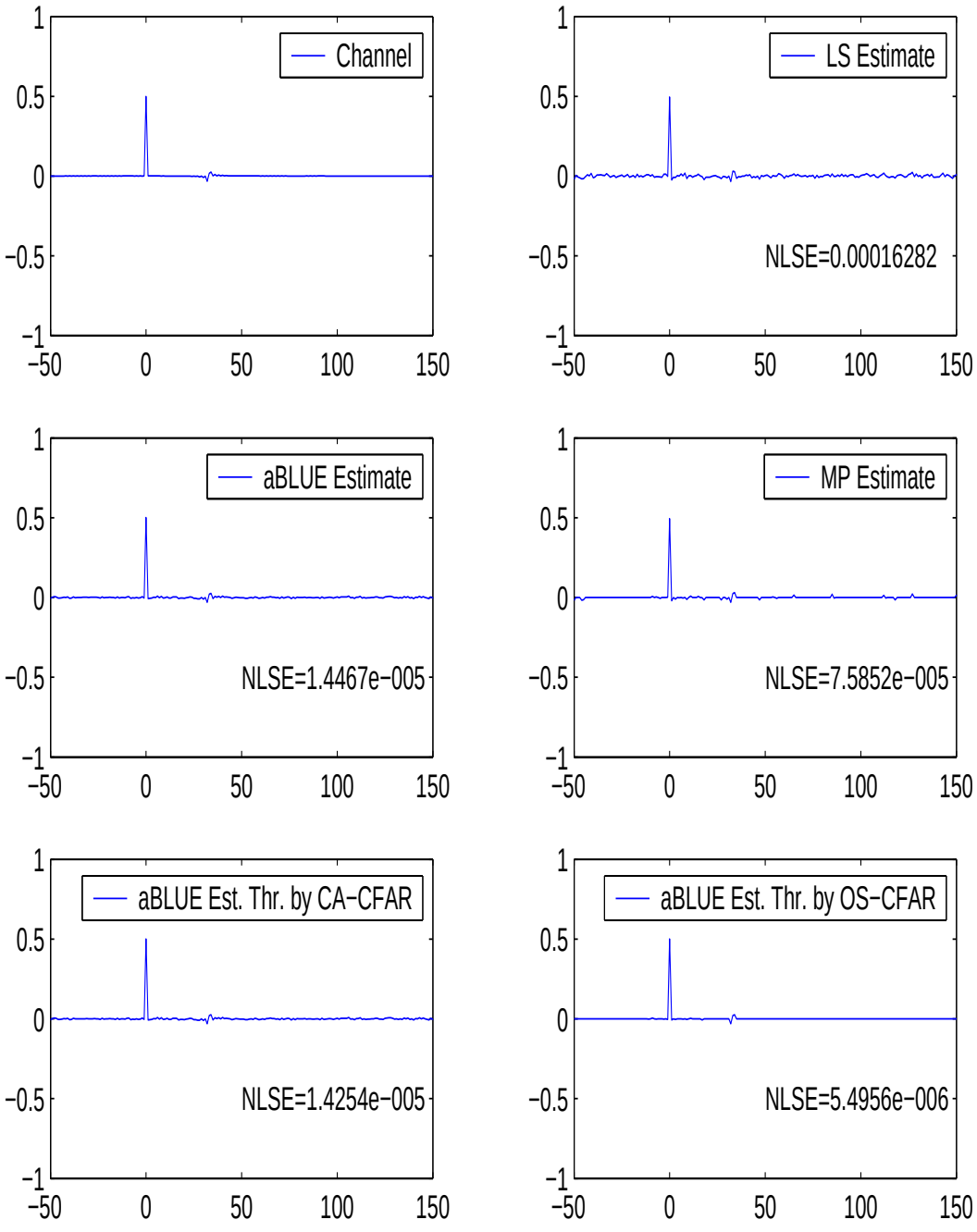

Figure 4.12: CIR estimations for Channel 8

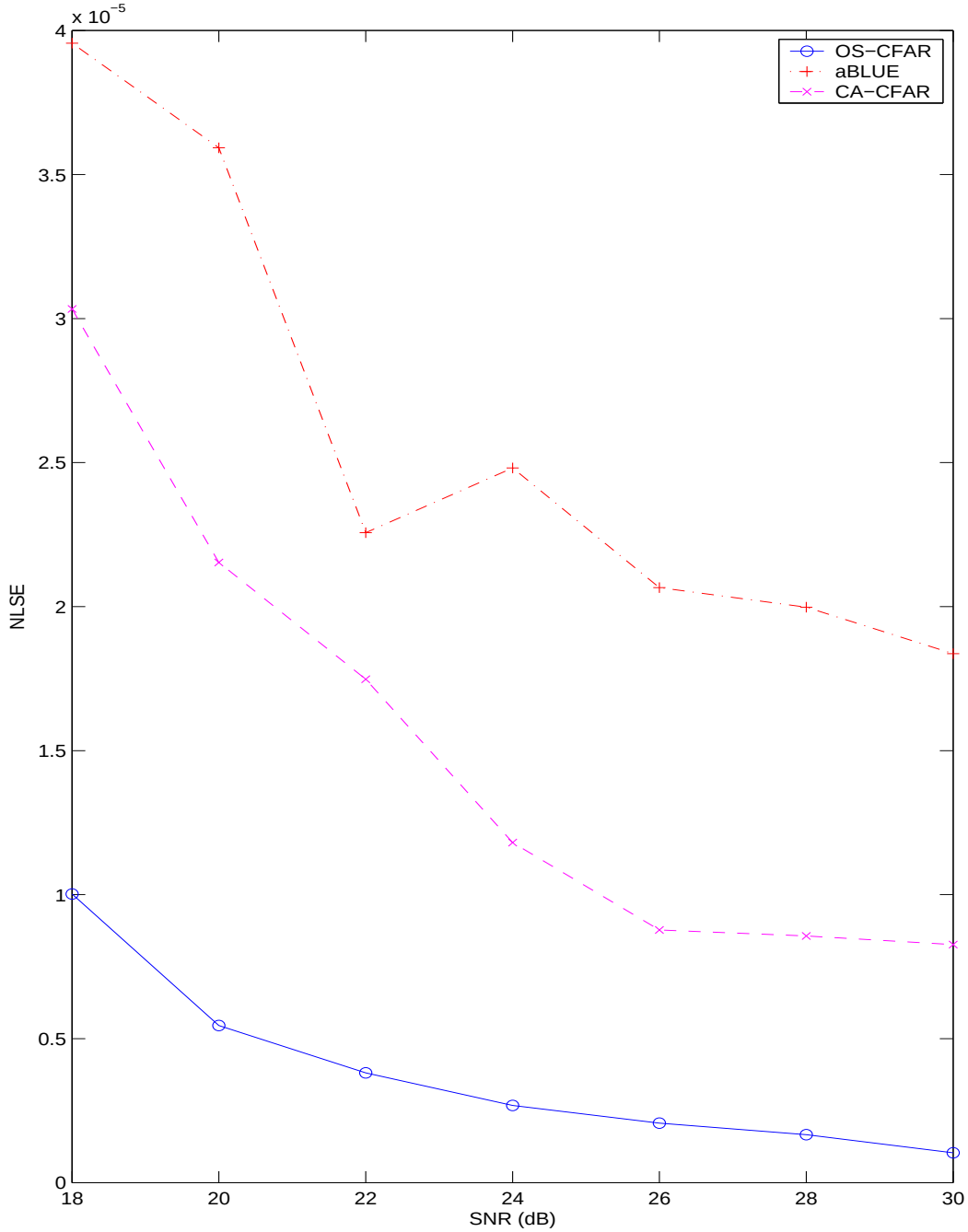

Figure 4.13: NLSE values of CIR estimates versus SNR(dB) for Channel 1

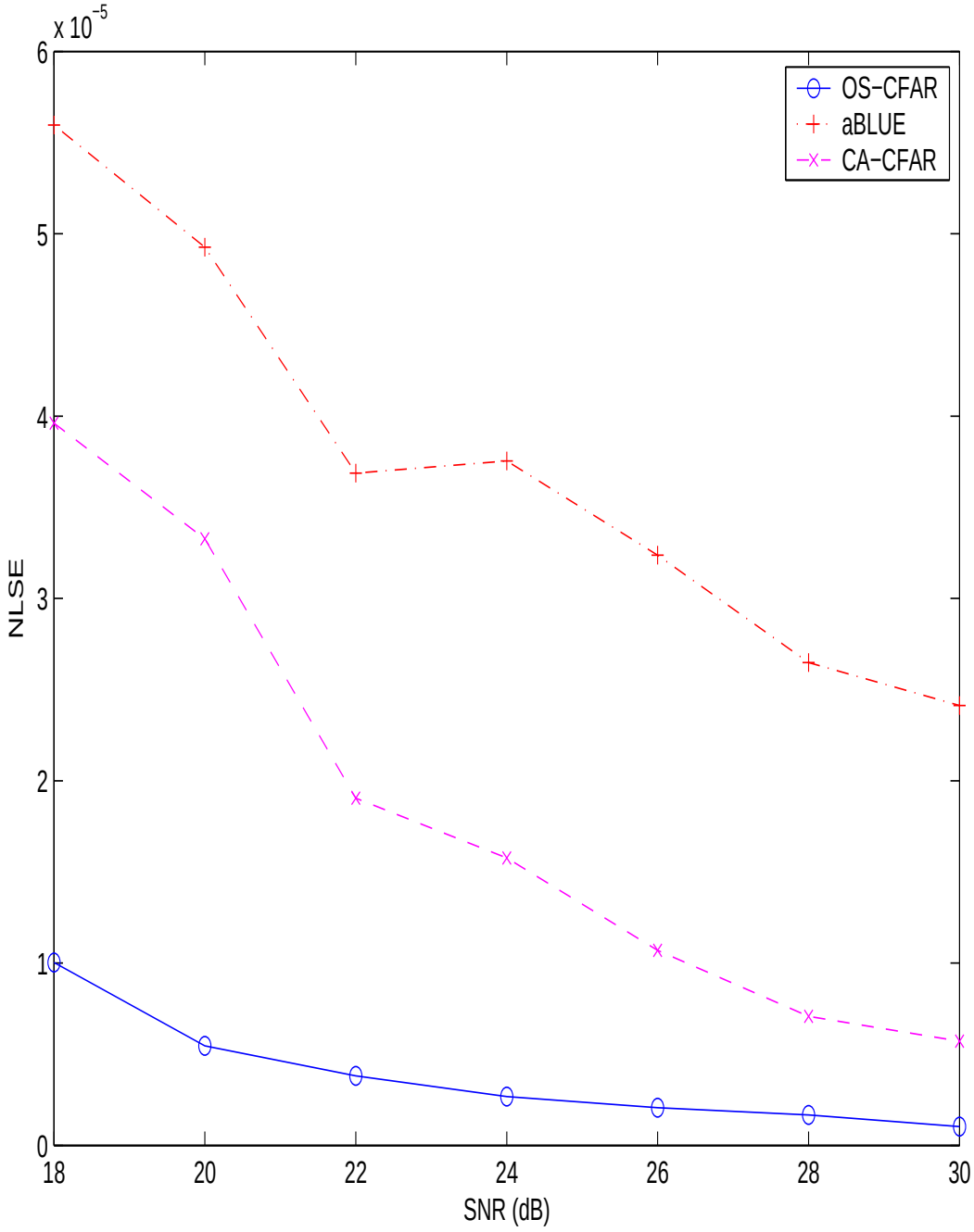

Figure 4.14: NLSE values of CIR estimates versus SNR(dB) for Channel 2

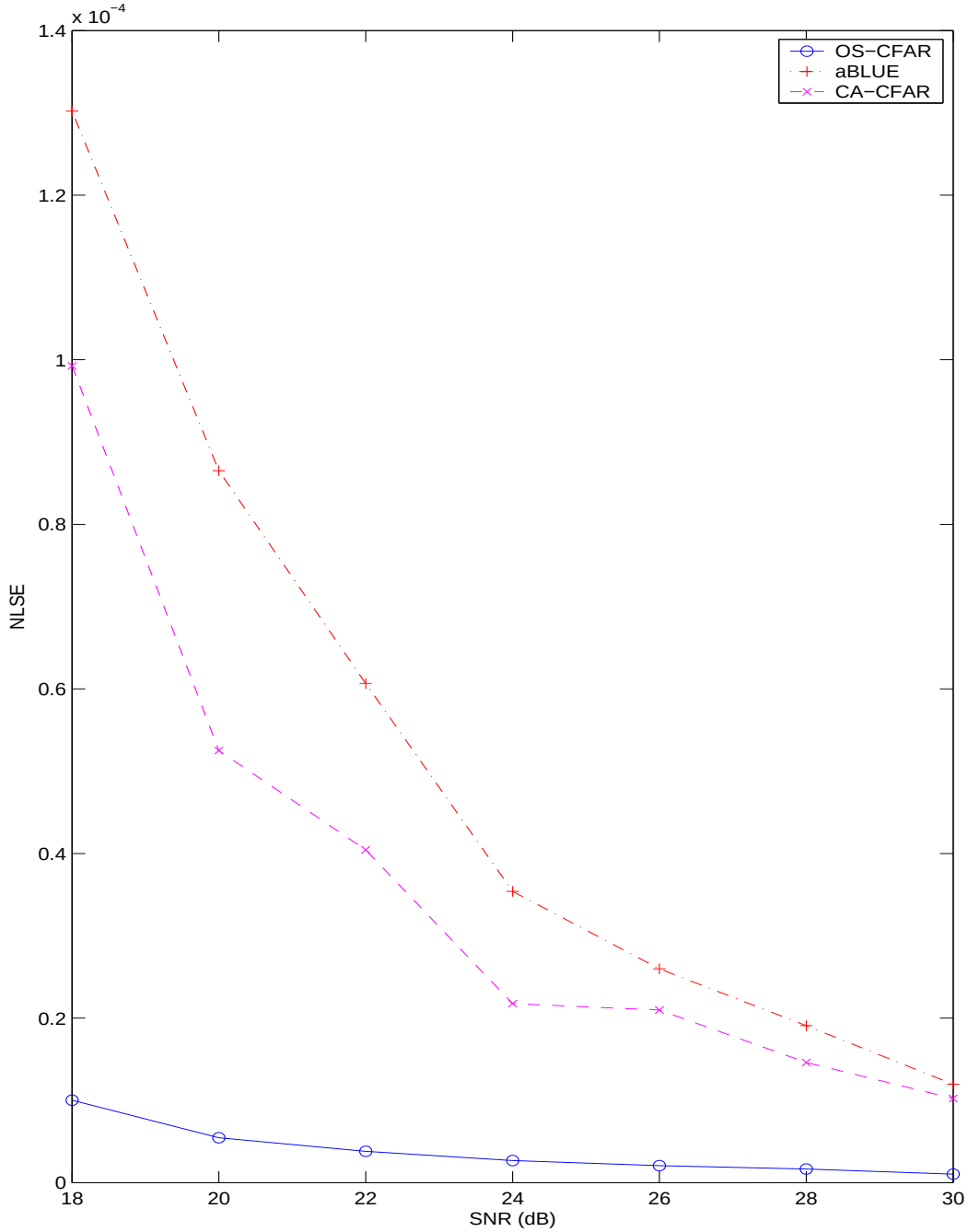

Figure 4.15: NLSE values of CIR estimates versus SNR(dB)for Channel 3

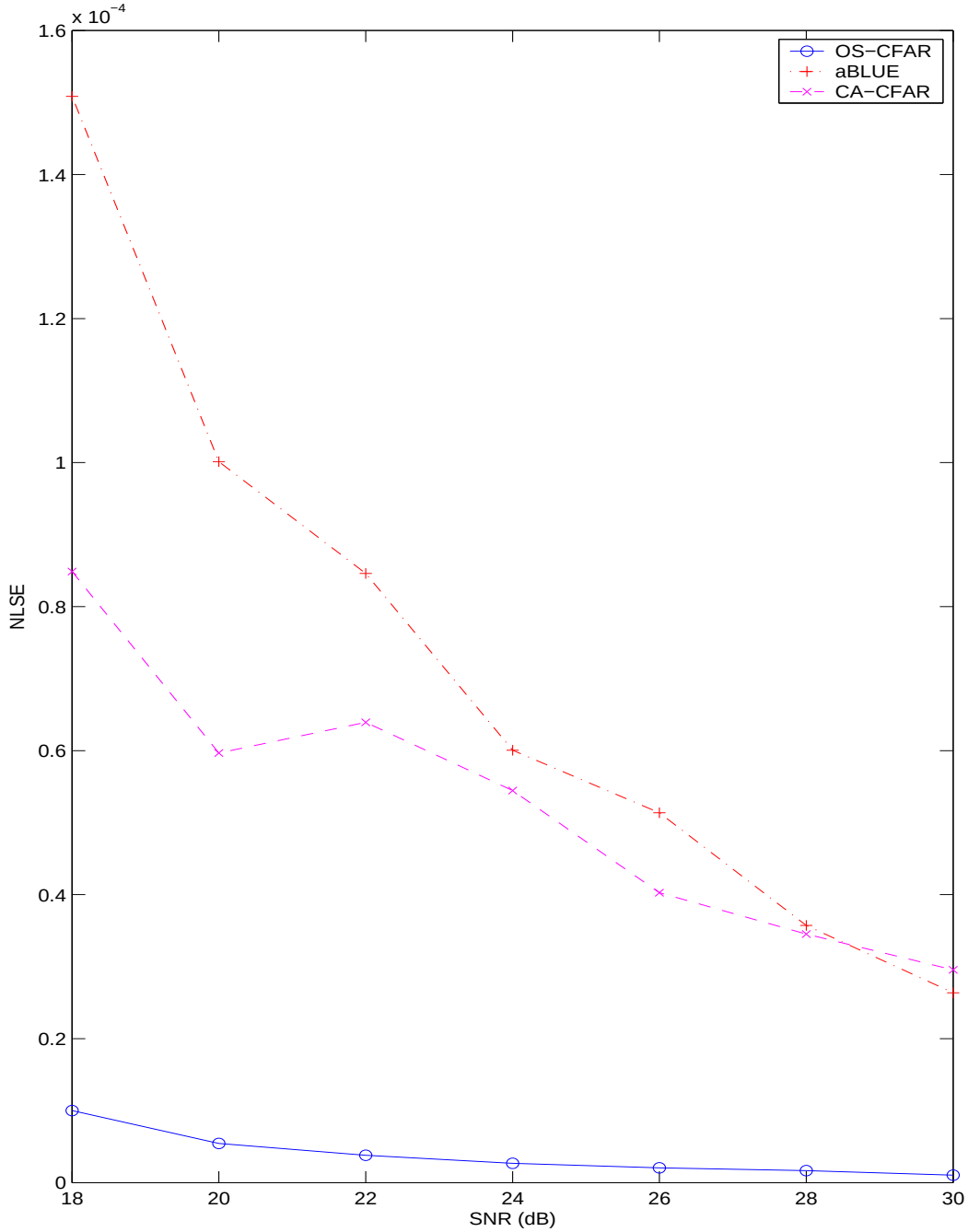

Figure 4.16: NLSE values of CIR estimates versus SNR(dB) for Channel 3-plus

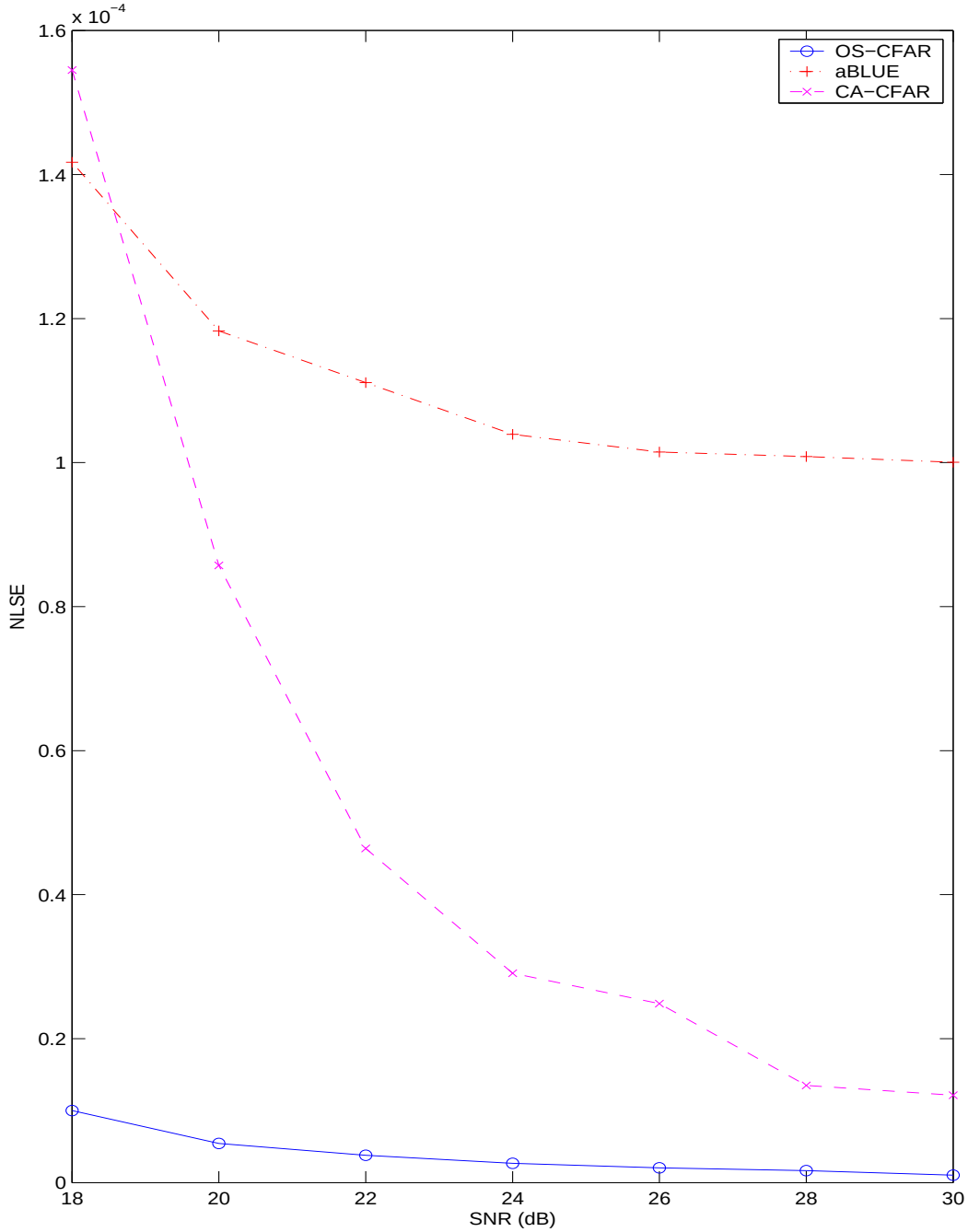

Figure 4.17: NLSE values of CIR estimates versus SNR(dB) for Channel 4

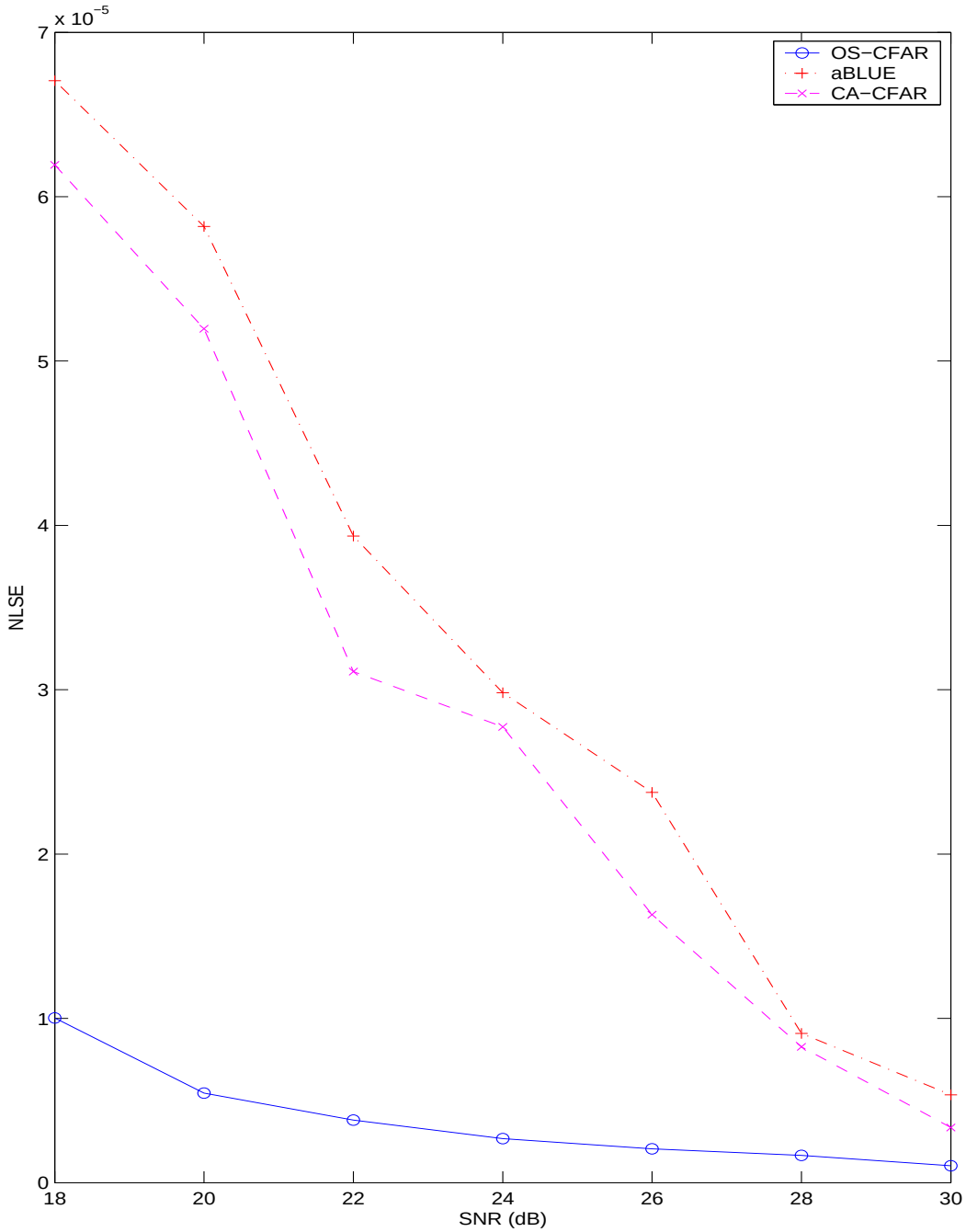

Figure 4.18: NLSE values of CIR estimates versus SNR(dB) for Channel 5

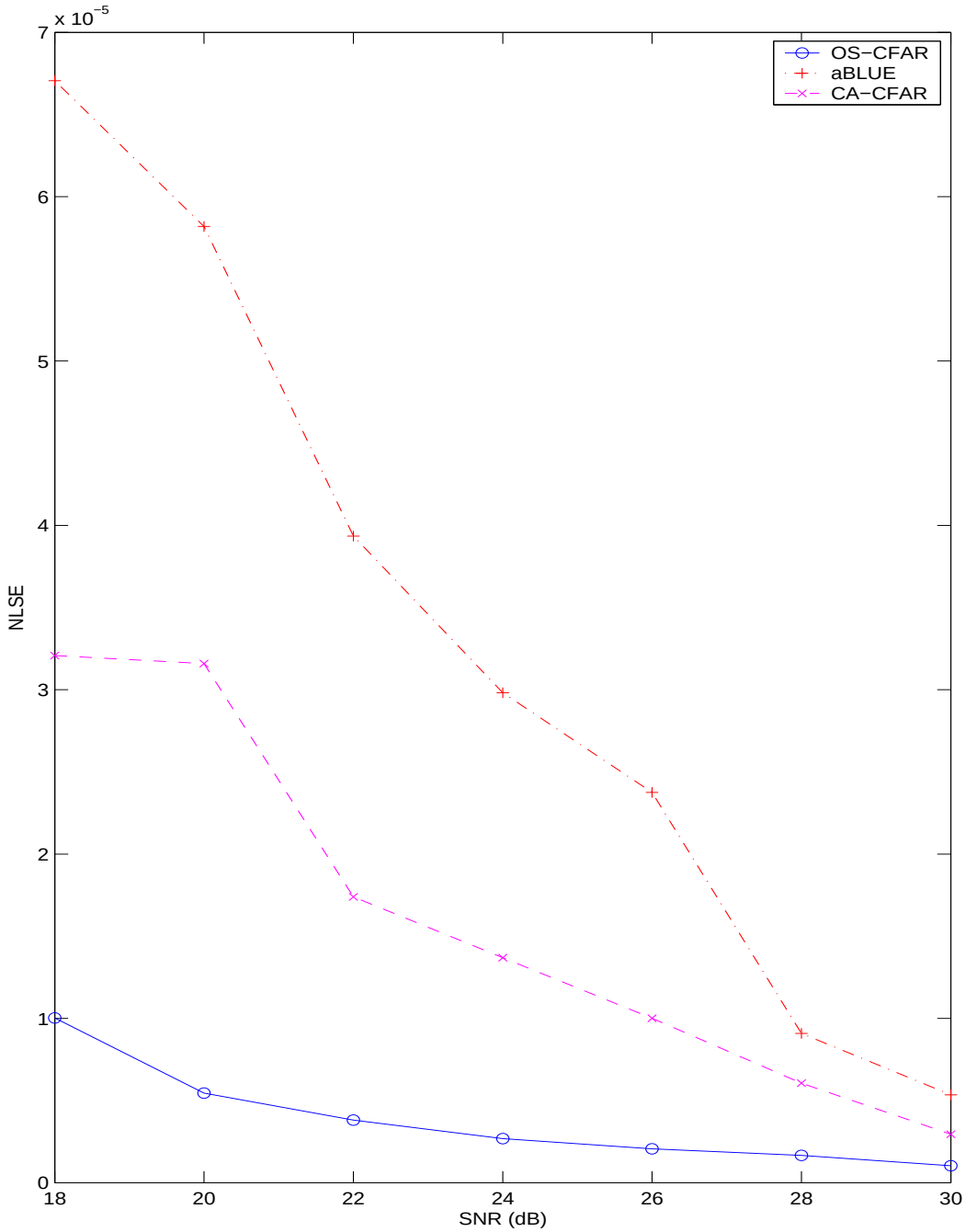

Figure 4.19: NLSE values of CIR estimates versus SNR(dB) for Channel 6

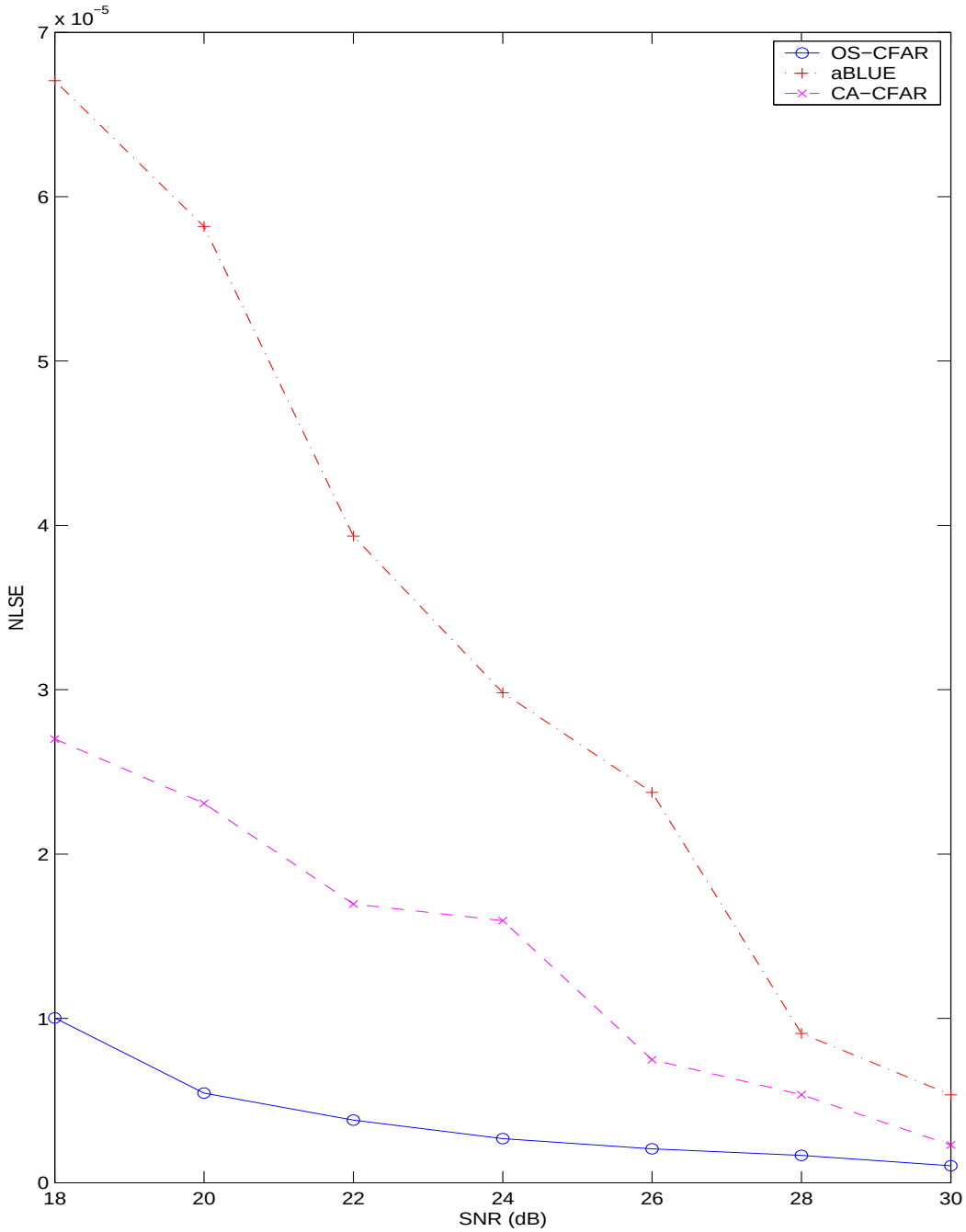

Figure 4.20: NLSE values of CIR estimates versus SNR(dB) for Channel 7

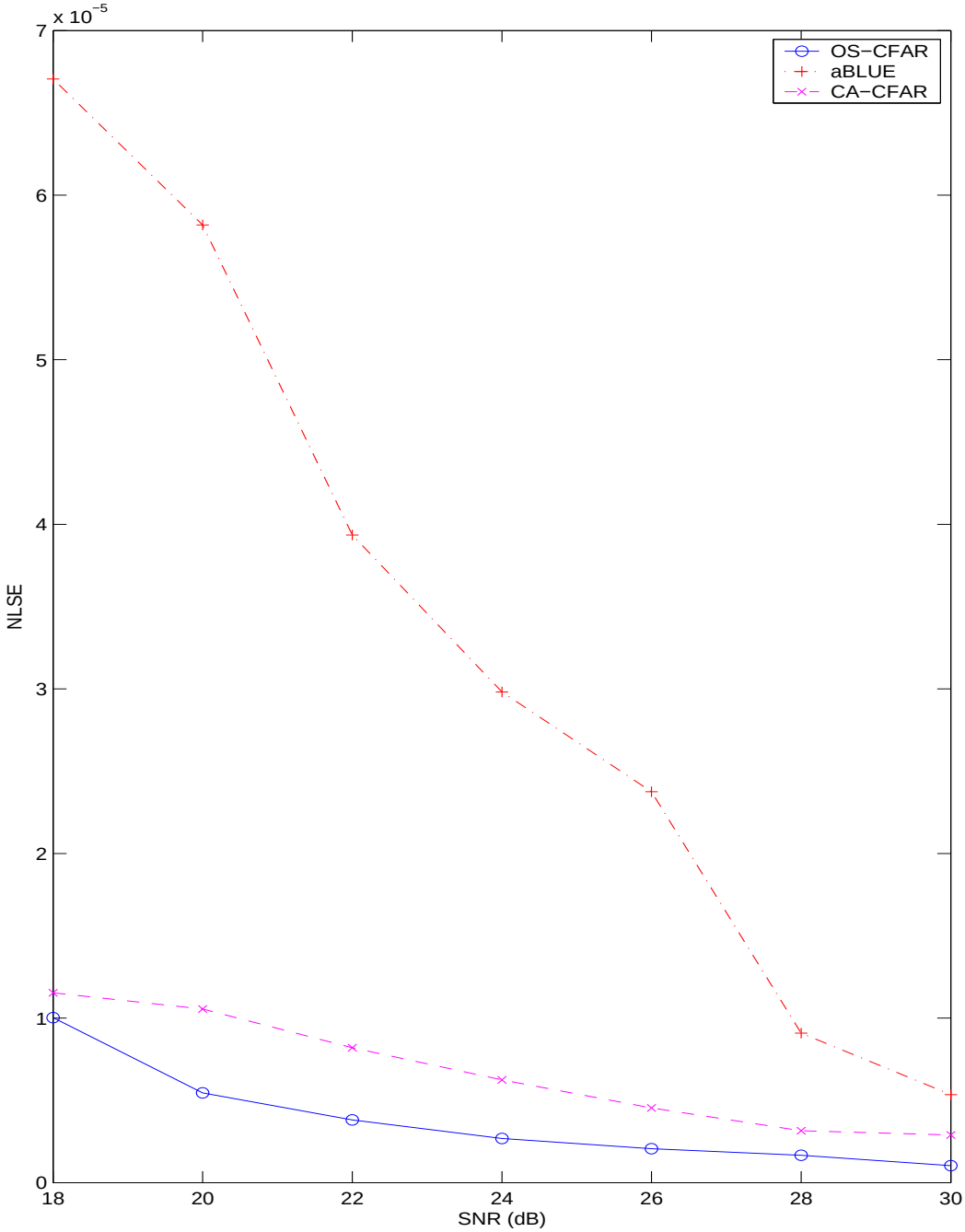

Figure 4.21: NLSE values of CIR estimates versus SNR(dB) for Channel 8

CHAPTER 5

CHANNEL ESTIMATE-BASED DECISION FEEDBACK EQUALIZERS

The derivations of the Decision Feedback Equalizer (DFE) equations are based on the recent work by Zoltowski et al "(Zoltowski and Hillery and Özen and Fimoff 2002)". The notation used in this chapter is slightly different from the rest of the thesis, and is better suited for the development of the DFE equations. Based on the minimum mean squared error (MMSE) criterion at the slicer input, the optimum feed-forward and feedback taps in a DFE are directly computable from an estimate of the channel[1].

The equalizer structures we consider in this chapter are illustrated in Figure 5.1. The equalizer is assumed to have $N_F + 1$ feedforward taps and N_B feedback taps. Assembling $N_F + 1$ consecutive samples of $y[k]$ into a vector $\boldsymbol{y}$ yields

$$\boldsymbol{y}[k] = \boldsymbol{H}\boldsymbol{s}[k] + \boldsymbol{Q}\boldsymbol{\eta}[k], \tag{5.1}$$

where $\boldsymbol{\eta}[k] = [\,\eta[k+L_q] \ldots \eta[k-L_q-N_F]\,]^T$, $\boldsymbol{s}[k] = [\,I[k+N_a] \ldots I[k-N_c-N_F]\,]^T$,

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{h}^T & 0 & \cdots & 0 \\ 0 & \boldsymbol{h}^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{h}^T \end{bmatrix}_{(N_F+1)\times(N_F+N_c+N_a+1)} \tag{5.2}$$

$$\boldsymbol{Q} = \begin{bmatrix} \boldsymbol{q}^T & 0 & \cdots & 0 \\ 0 & \boldsymbol{q}^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{q}^T \end{bmatrix}_{(N_F+1)\times(N_F+N_q)} \tag{5.3}$$

[1]This chapter has been presented in part at the Thirty-Sixth Asilomar Conference on Signals, Systems, ad Computers, Monterey, CA, November, 2002

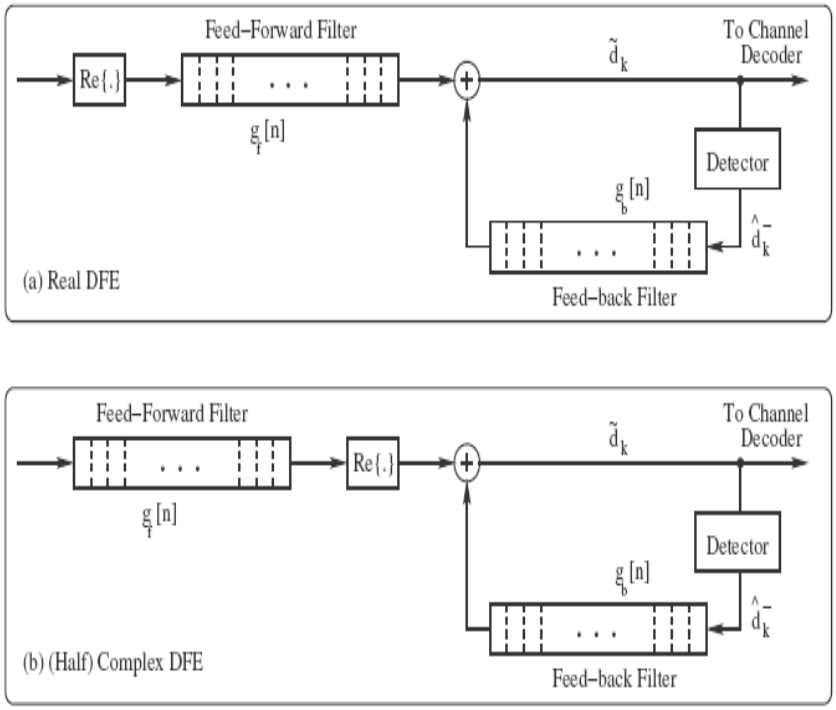

Figure 5.1: Part (a) shows the real equalizer where both the feed-forward and feed-back are real valued; Part (b) shows the (half)-complex equalizer where the feed-forward taps are complex valued

where $\boldsymbol{q}$ is the finite length representation of the receiver matched filter with length $N_q = 2L_q + 1$:

$$\boldsymbol{q} = [\, q[-L_q], \ldots, q[-1], q[0], q[1], \ldots, q[L_q] \,]^T . \tag{5.4}$$

The symbol estimate $\hat{I}[k-\delta]$ is given by

$$\hat{I}[k-\delta] = \mathrm{Re}\{\boldsymbol{g}_F{}^H \boldsymbol{y}[k]\} + \boldsymbol{g}_B{}^T \boldsymbol{I}_B[k-\delta-1], \tag{5.5}$$

where δ is the cursor location (defined below), and

$$\boldsymbol{g}_F = [\, g_F[0], \ldots, g_F[N_F] \,]^T ,$$

$$\boldsymbol{g}_B = [\, g_B[1], \ldots, g_B[N_B] \,]^T ,$$

$$\boldsymbol{s}_B[k] = [\, I[k], \ldots, I[k+1-N_B] \,]^T ,$$

and the superscript H denotes conjugate transpose. In this analysis we assume that all decisions are correct. If we adopt the convention that the subscripts R and I indicate the real and imaginary parts of quantities, then

$$\begin{aligned}\hat{I}[k-\delta] &= \boldsymbol{g}_{FR}{}^{T}\boldsymbol{y}_R[k] + \boldsymbol{g}_{FI}{}^{T}\boldsymbol{y}_I[k] + \boldsymbol{g}_B{}^{T}\boldsymbol{s}_B[k-\delta-1] \\ &= \boldsymbol{g}_{FC}{}^{T}\boldsymbol{y}_C[k] + \boldsymbol{g}_B{}^{T}\boldsymbol{s}_B[k-\delta-1], \end{aligned} \tag{5.6}$$

where $\boldsymbol{g}_{FC} = \left[\, \boldsymbol{g}_{FR}{}^{T}\ \boldsymbol{g}_{FI}{}^{T} \,\right]^{T}$ and $\boldsymbol{y}_C[k] = \left[\, \boldsymbol{y}_R{}^{T}[k]\ \boldsymbol{y}_I{}^{T}[k] \,\right]^{T}$. We may now write

$$\boldsymbol{y}_C[k] = \boldsymbol{H}_C\boldsymbol{s}[k] + \boldsymbol{Q}_C\boldsymbol{\eta}_C[k], \tag{5.7}$$

where $\boldsymbol{H}_C = \left[\, \boldsymbol{H}_R{}^{T}\ \boldsymbol{H}_I{}^{T} \,\right]^{T}$, $\boldsymbol{Q}_C = \begin{bmatrix} \boldsymbol{Q}_R & -\boldsymbol{Q}_I \\ \boldsymbol{Q}_I & \boldsymbol{Q}_R \end{bmatrix}$, and $\boldsymbol{\eta}_C[k] = \left[\, \boldsymbol{\eta}_R{}^{T}[k]\ \boldsymbol{\eta}_I{}^{T}[k] \,\right]^{T}$.

In the decision feedback equalizer, we use the cursor to define exactly which symbol is being estimated. To motivate the cursor definition, we use the real equalizer and consider a channel containing a single path with real gain at delay zero. In this case, $h_R[k]$ is a delta function at $k = 0$ and the sample $y_R[k]$ corresponds to the symbol $I[k]$ (since $y_R[k] = I[k] + \nu_R[k]$). Then the feedforward term in the symbol estimate is

$$\begin{aligned}\boldsymbol{g}_F{}^{T}\boldsymbol{y}_R[k] &= \sum_{n=0}^{N_F} g_F[n]y_R[k-n] \\ &= \sum_{n=0}^{N_F} g_F[n]\left(I[k-n]+\nu_R[k-n]\right), \end{aligned} \tag{5.8}$$

where we recall that $\boldsymbol{g}_F$ is real for this discussion. That is, the feedforward term in the equalizer output only depends on the symbols $I[k], I[k-1], \ldots, I[k-N_F]$. When there is multipath interference, the feedforward term will depend on symbols covering a wider time-span, but since we may encounter a channel where the multipath is negligible, we may only consider $I[k], I[k-1], \ldots, I[k-N_F]$ as candidates for the symbol to estimate. Therefore, we estimate the symbol $I[k-\delta]$, where $0 \le \delta \le N_F$, and call this symbol the *cursor*. Since there is a one-to-one correspondence between the candidate symbols and the taps in the feedforward section, we often identify the cursor by the corresponding

feedforward tap. This definition of the cursor is consistent with standard DFE theory and practice.

We now find the equalizer by minimizing the mean-square error (MSE), where

$$\begin{aligned} \text{MSE} &= \mathrm{E}\{(I[k-\delta] - \hat{I}[k-\delta])^2\} \\ &= \mathcal{E}_s - 2\boldsymbol{g}_{FC}{}^T \boldsymbol{r}_{sy_C} + 2\boldsymbol{g}_{FC}{}^T \boldsymbol{R}_{y_C s_B} \boldsymbol{g}_B \\ &\quad + \boldsymbol{g}_{FC}{}^T \boldsymbol{R}_{y_C y_C} \boldsymbol{g}_{FC} + \mathcal{E}_s \boldsymbol{g}_B{}^T \boldsymbol{g}_B \end{aligned} \tag{5.9}$$

with $\mathcal{E}_s = \mathrm{E}\{(I[k])^2\}$, $\boldsymbol{r}_{sy_C} = \mathrm{E}\{I[k-\delta]\boldsymbol{y}_C[k]\}$, $\boldsymbol{R}_{y_C s_B} = \mathrm{E}\{\boldsymbol{y}_C[k]\boldsymbol{I}_B{}^T[k-\delta-1]\}$, and $\boldsymbol{R}_{y_C y_C} = \mathrm{E}\{\boldsymbol{y}_C[k]\boldsymbol{y}_C^T[k]\}$. Here we have made use of the expressions $\boldsymbol{R}_{s_B s_B} = \mathrm{E}\{\boldsymbol{s}_B[k]\boldsymbol{s}_B{}^T[k]\} = \mathcal{E}_s \boldsymbol{I}_{N_B}$ and $\boldsymbol{r}_{ss_B} = \mathrm{E}\{I[k-\delta]\boldsymbol{s}_B[k]\} = \boldsymbol{0}$. Minimization of this expression yields

$$\boldsymbol{g}_{FC} = \left(\boldsymbol{R}_{y_C y_C} - \frac{1}{\mathcal{E}_s}\boldsymbol{R}_{y_C s_B}\boldsymbol{R}_{y_C s_B}{}^T\right)^{-1} \boldsymbol{r}_{sy_C} \tag{5.10}$$

$$\boldsymbol{g}_B = -\frac{1}{\mathcal{E}_s}\boldsymbol{R}_{y_C s_B}{}^T \boldsymbol{g}_{FC}, \tag{5.11}$$

where

$$\boldsymbol{r}_{sy_C} = \mathcal{E}_s \boldsymbol{H}_C \boldsymbol{1}_{\delta+N_a} \tag{5.12}$$

$$\boldsymbol{R}_{y_C y_C} = \mathcal{E}_s \boldsymbol{H}_C \boldsymbol{H}_C{}^T + N_0 \boldsymbol{Q}_C \boldsymbol{Q}_C{}^T \tag{5.13}$$

$$\boldsymbol{R}_{y_C s_B} = \mathcal{E}_s \boldsymbol{H}_C \boldsymbol{\Delta}_\delta. \tag{5.14}$$

The vector $\boldsymbol{1}_{\delta+N_a}$ contains all zeros except for a one in element $\delta + N_a + 1$. The matrix $\boldsymbol{\Delta}_\delta$ is defined by

$$\boldsymbol{\Delta}_\delta = \begin{bmatrix} \boldsymbol{0}_{(N_a+\delta+1)\times N_B} \\ \boldsymbol{I}_{N_B} \\ \boldsymbol{0}_{(N_F+N_c-\delta-N_B)\times N_B} \end{bmatrix} \tag{5.15}$$

when $0 \le \delta \le N_c + N_F - N_B$ and

$$\boldsymbol{\Delta}_\delta =$$

$$\begin{bmatrix} \mathbf{0}_{(N_a+\delta+1)\times(N_F+N_c-\delta)} & \mathbf{0}_{(N_a+\delta+1)\times(\delta+N_B-N_F-N_c)} \\ \boldsymbol{I}_{N_F+N_c-\delta} & \mathbf{0}_{(N_F+N_c-\delta)\times(\delta+N_B-N_F-N_c)} \end{bmatrix} \tag{5.16}$$

when $N_c + N_F - N_B < \delta \leq N_c + N_F - 1$. Here $\boldsymbol{I}_n$ is the identity matrix of size $n \times n$.

The derivation of the real equalizer is similar with the result that

$$\boldsymbol{g}_F = \left(\boldsymbol{R}_{y_R y_R} - \frac{1}{\mathcal{E}_s} \boldsymbol{R}_{y_R s_B} \boldsymbol{R}_{y_R s_B}{}^T \right)^{-1} \boldsymbol{r}_{s y_R} \tag{5.17}$$

$$\boldsymbol{g}_B = -\frac{1}{\mathcal{E}_s} \boldsymbol{R}_{y_R s_B}{}^T \boldsymbol{g}_{FR}, \tag{5.18}$$

where

$$\boldsymbol{r}_{s y_R} = \mathcal{E}_s \boldsymbol{H}_R \mathbf{1}_{\delta+N_a} \tag{5.19}$$

$$\begin{aligned} \boldsymbol{R}_{y_R y_R} &= \mathcal{E}_s \boldsymbol{H}_R \boldsymbol{H}_R{}^T + N_0 \left(\boldsymbol{Q}_R \boldsymbol{Q}_R{}^T + \boldsymbol{Q}_I \boldsymbol{Q}_I{}^T \right) \\ &= \mathcal{E}_s \boldsymbol{H}_R \boldsymbol{H}_R{}^T + N_0 \boldsymbol{I}_{N_F+1} \end{aligned} \tag{5.20}$$

$$\boldsymbol{R}_{y_R s_B} = \mathcal{E}_s \boldsymbol{H}_R \boldsymbol{\Delta}_\delta. \tag{5.21}$$

5.1. Simulations

We compare the accuracy of the channel estimates $\widehat{\boldsymbol{h}}_{BLS}$ and quantify their effectiveness by considering their use in the equalization of sparse channels. We considered a DFE with real valued taps, and with 256 feed-forward taps and 448 feedback taps. The cursor tap is placed at the 221st tap on the feed-forward filter.

We considered an 8-VSB ”(ATSC Standard A/53 1995)” receiver with a single antenna. 8-VSB system has a complex raised cosine pulse shape ”(ATSC Standard A/53 1995)”. The CIRs we considered are given in Appendix B, Tables B. The phase angles of individual paths for all the channels are taken to be

$$\arg\{c_k\} = \exp(-j2\pi f_c \tau_k), \quad k = 1, \cdots, 6 \tag{5.22}$$

where $f_c = \frac{50}{T_{sym}}$ and $T_{sym} = 92.9$nsec.

Based on each of channel estimates the optimal real valued DFE filter coefficients are obtained, and the equalization performance without *error propagation* is compared. The DFE output without error propagation (slicer input) signal-to-interference-plus-noise-ratio denoted by $\mathrm{SINR}_{\mathrm{DFE}}$, is the metric used for comparison of equalizers obtained from the channel estimates. The output of the equalizer is given by

$$\hat{I}[k] = \tilde{c}[0] * I[k] + \sum_{\forall n, n \neq 0} \tilde{c}[n] * I[k-n] + v[k] \tag{5.23}$$

where the first term in Equation (5.23) is the desired symbol with a multiplicative constant, and $\tilde{c}[k]$ is the effective channel at the output of the equalizer. $\tilde{c}[0] = 1$ only when the estimate is unbiased. The second term (the summation) in Equation (5.23) is the residual ISI and the last term ($v[k]$) is noise (colored). The residual channel impulse response after feed forward filter is shown in Figure (5.2). Figure (5.3) shows the residual channel impulse response after feed-back filter is applied to residual channel impulse response of Figure (5.2).

$\mathrm{SINR}_{\mathrm{DFE}}$ is then defined as the variance of the desired term divided by the variance of everything else. Since the symbols and noise are uncorrelated, this turns out to be

$$\mathrm{SINR}_{\mathrm{DFE}} \quad = \quad \frac{E_s \tilde{c}[0]^2}{E_s \sum\limits_{\forall n, n \neq 0} \tilde{c}[n]^2 + E\{v[k]^2\}}. \tag{5.24}$$

The $\mathrm{SINR}_{\mathrm{DFE}}$ values for each channel estimate is provided for 18 to 28dB Signal-to-Noise-Ratio (SNR) values measured at the input to the receive pulse matched filter, and it is calculated by

$$\mathrm{SNR} = E_s \left\| (c(t) * q(t))|_{t=nT} \right\|^2 / N_0, \tag{5.25}$$

where $E_s = 21$ is the symbol energy for 8-VSB system, and N_0 is the channel noise variance.

Figures 5.4-5.12 illustrate the $\mathrm{SINR}_{\mathrm{DFE}}$ versus SNR for Channels 1–8, and for ideal CIR based DFE, for the approximated BLUE estimate $\hat{h}_{aBLUE}$, the approximated BLUE estimate after CA-CFAR applied $\hat{h}_{CA-CFAR}$ and the approximated BLUE estimate after OS-CFAR applied $\hat{h}_{OS-CFAR}$

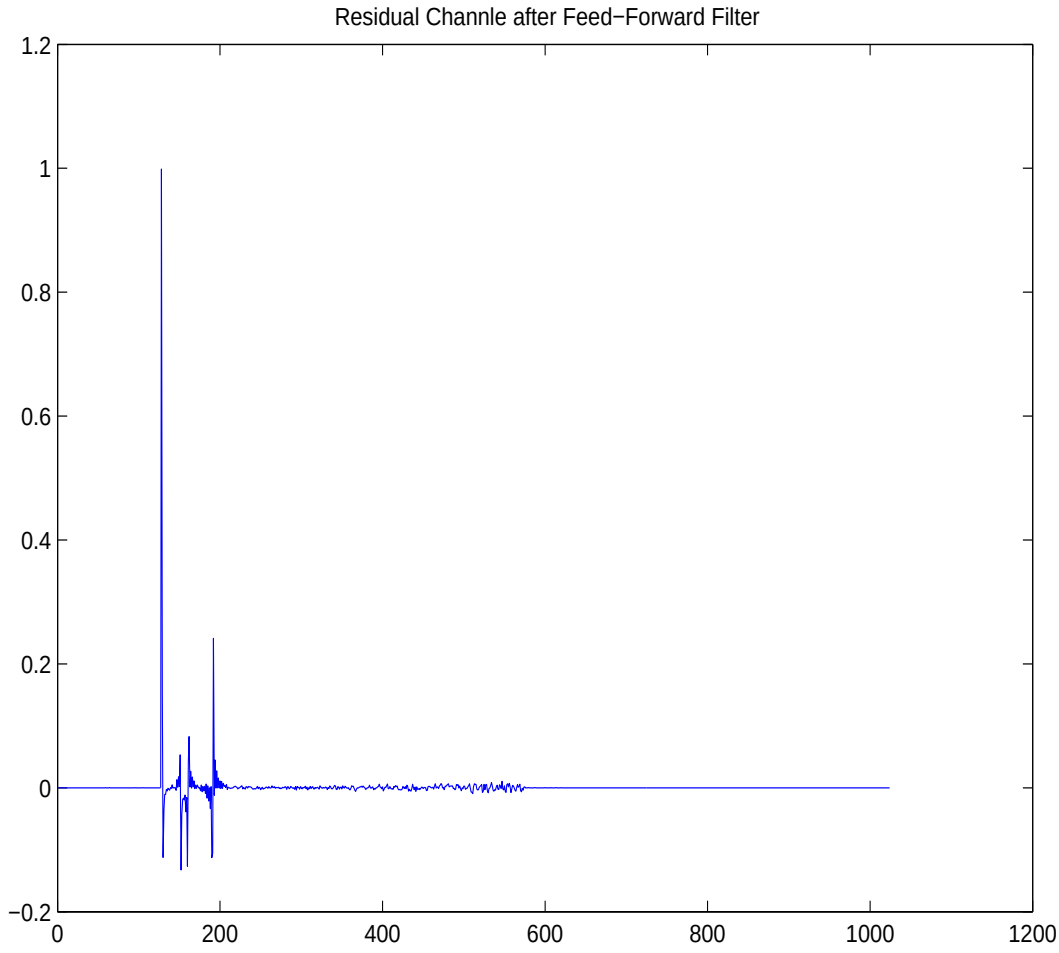

Figure 5.2: Residual Channel after Feed-Forward Filter

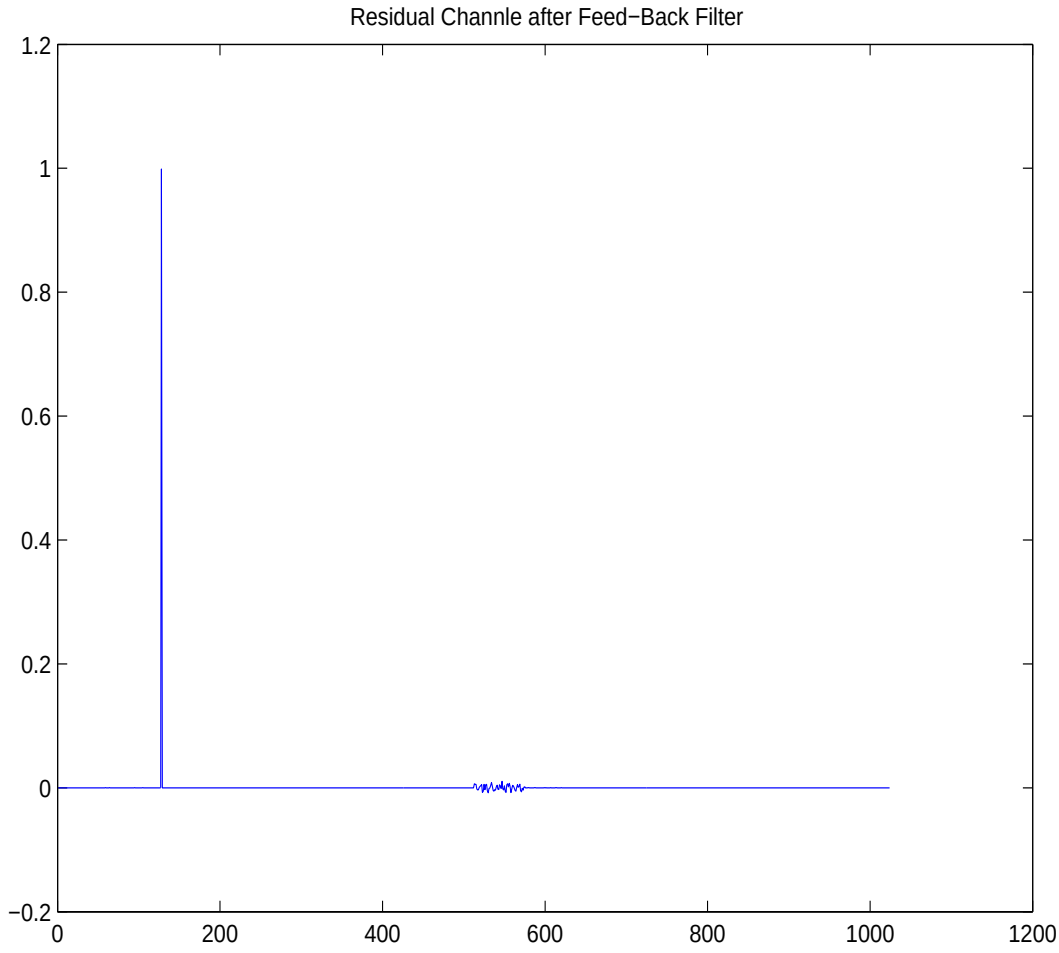

Figure 5.3: Residual Channel after Feed-Back Filter

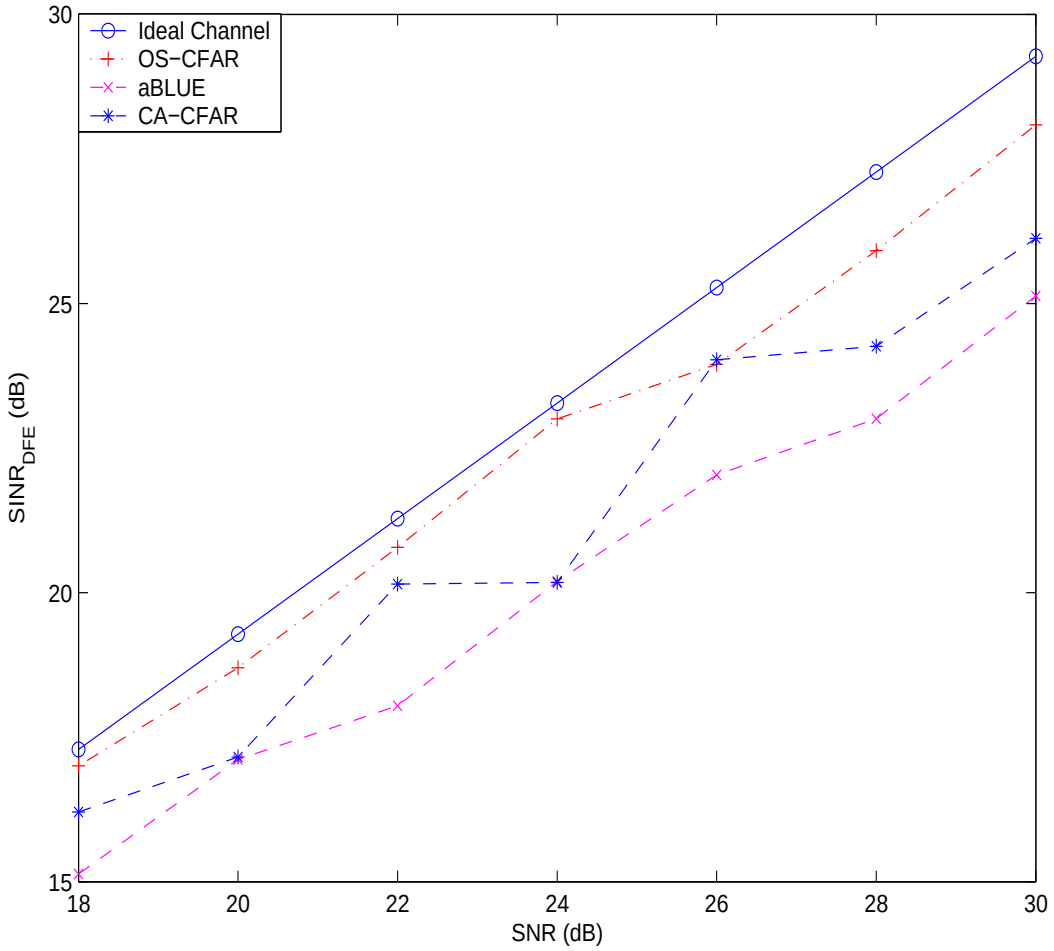

Figure 5.4: $\mathrm{SINR}_{\mathrm{DFE}}$ versus SNR for Channel 1 for various channel estimates.

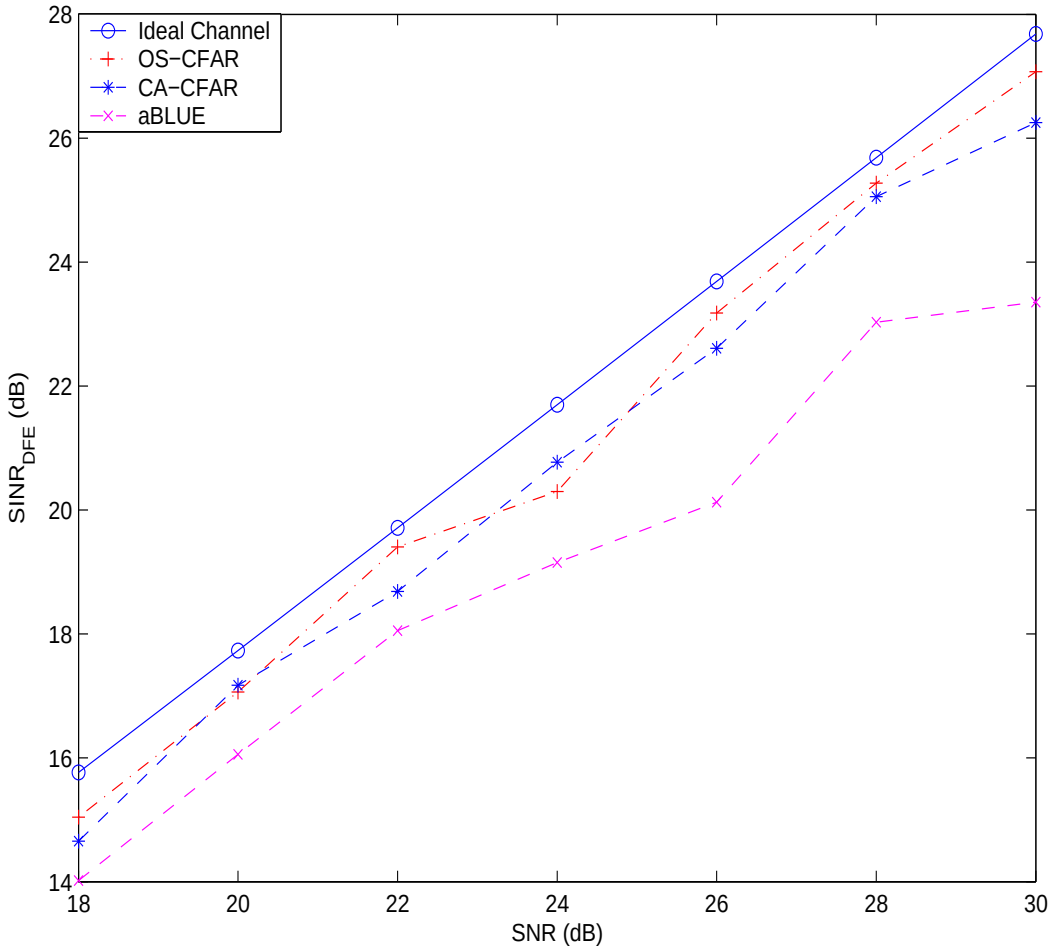

Figure 5.5: $\mathrm{SINR}_{\mathrm{DFE}}$ versus SNR for Channel 2 for various channel estimates.

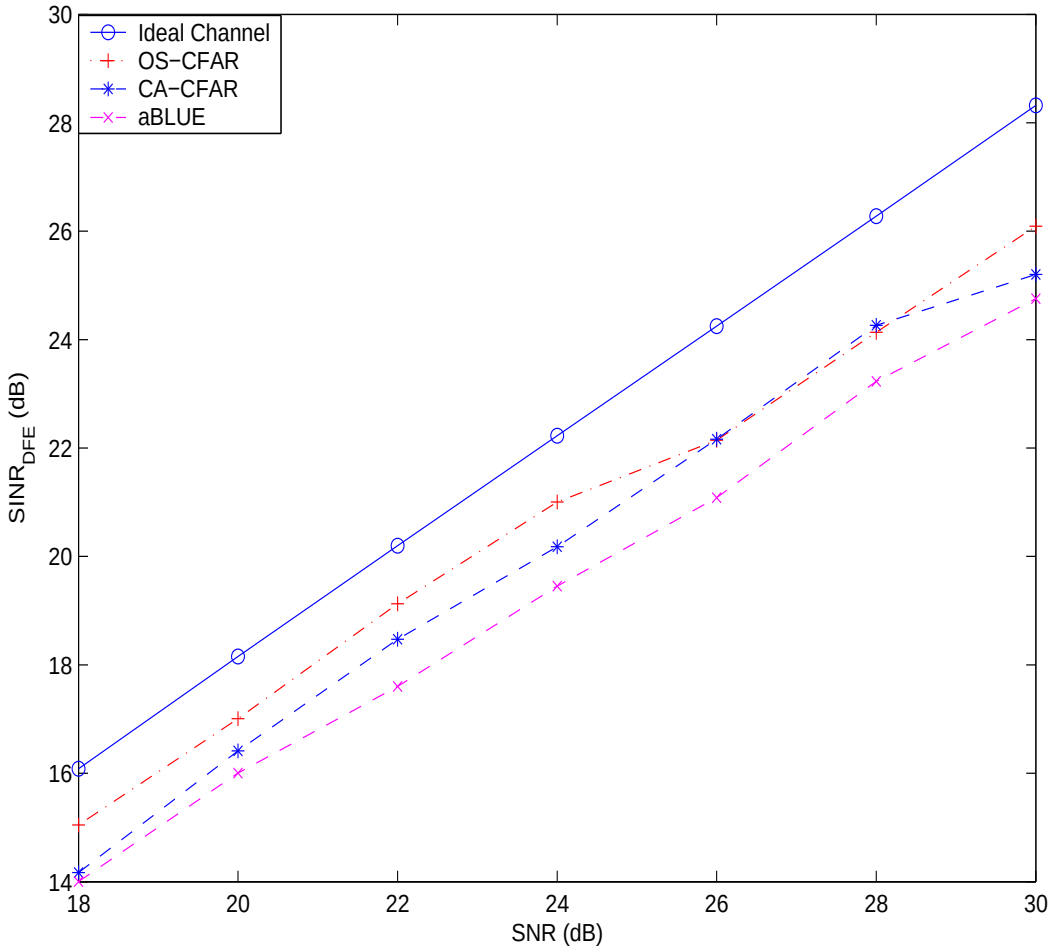

Figure 5.6: $\mathrm{SINR}_{\mathrm{DFE}}$ versus SNR for Channel 3 for various channel estimates.

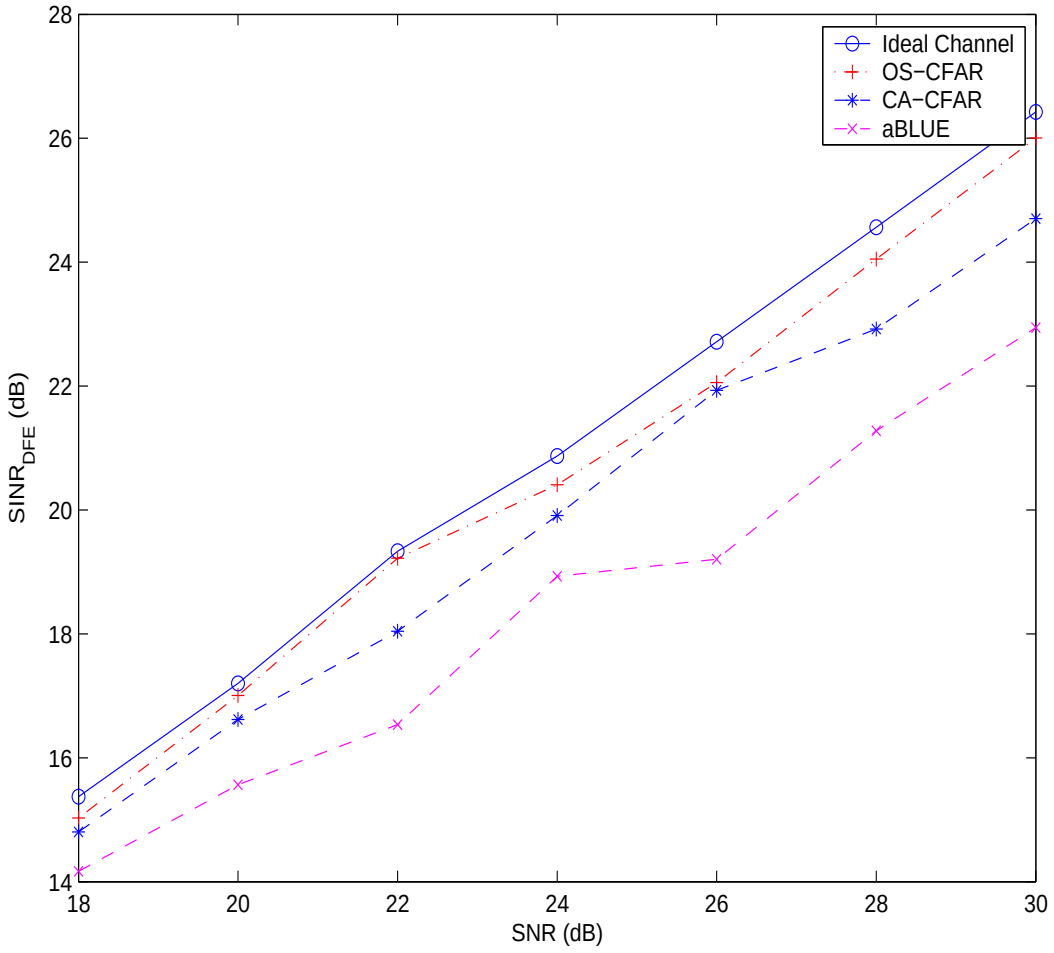

Figure 5.7: $\mathrm{SINR}_{\mathrm{DFE}}$ versus SNR for Channel 3-plus for various channel estimates.

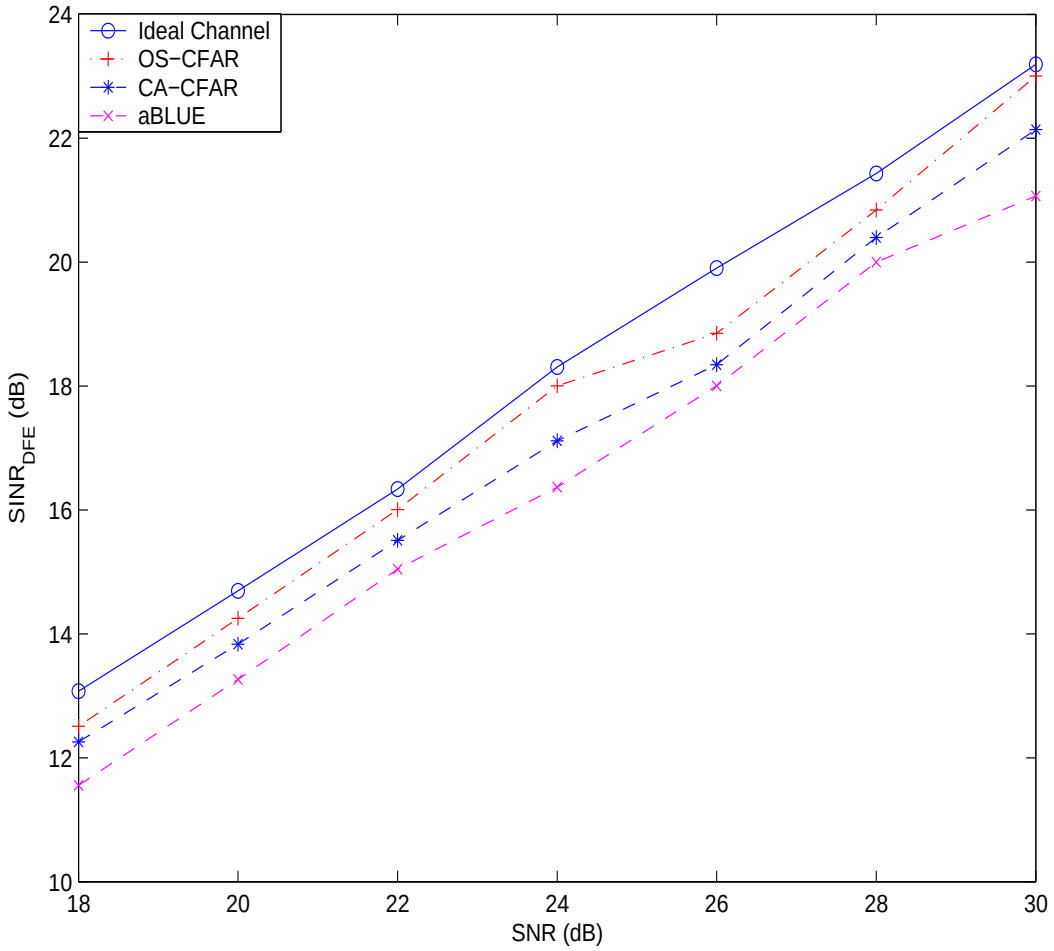

Figure 5.8: $\mathrm{SINR}_{\mathrm{DFE}}$ versus SNR for Channel 4 for various channel estimates.

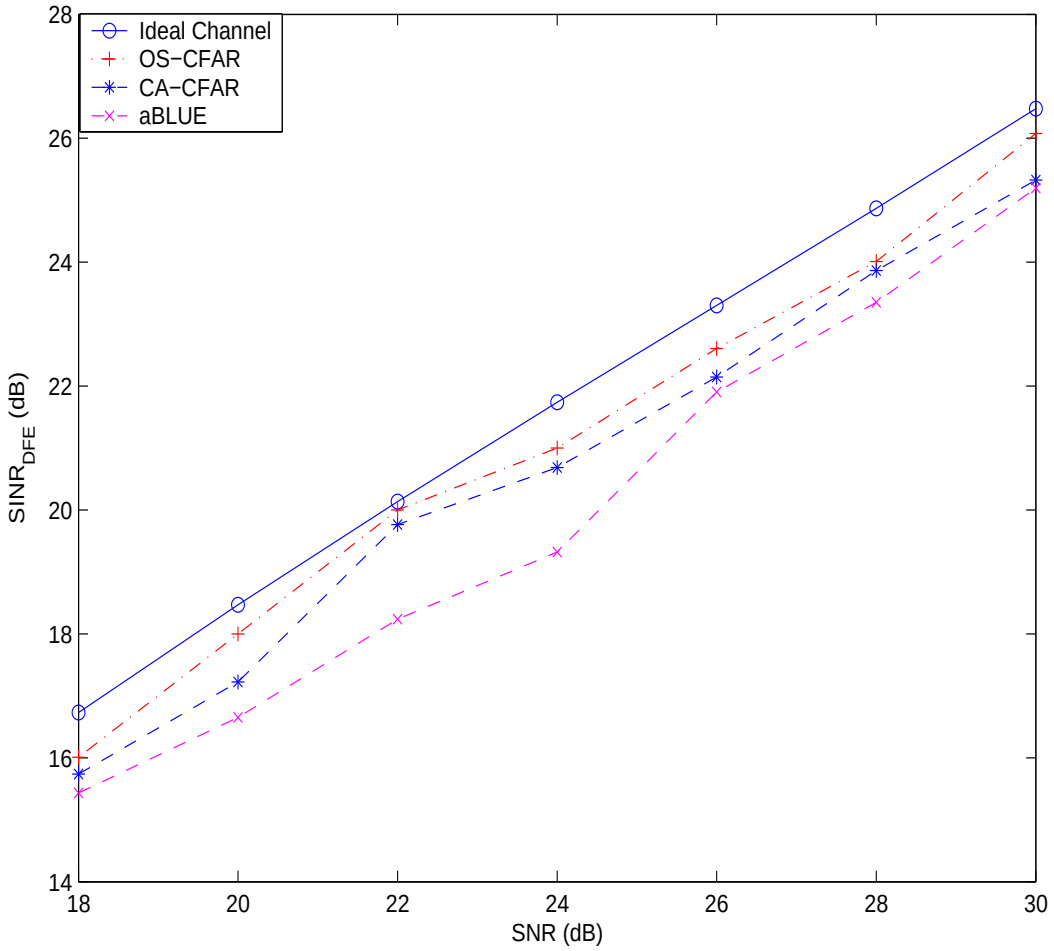

Figure 5.9: $\mathrm{SINR}_{\mathrm{DFE}}$ versus SNR for Channel 5 for various channel estimates.

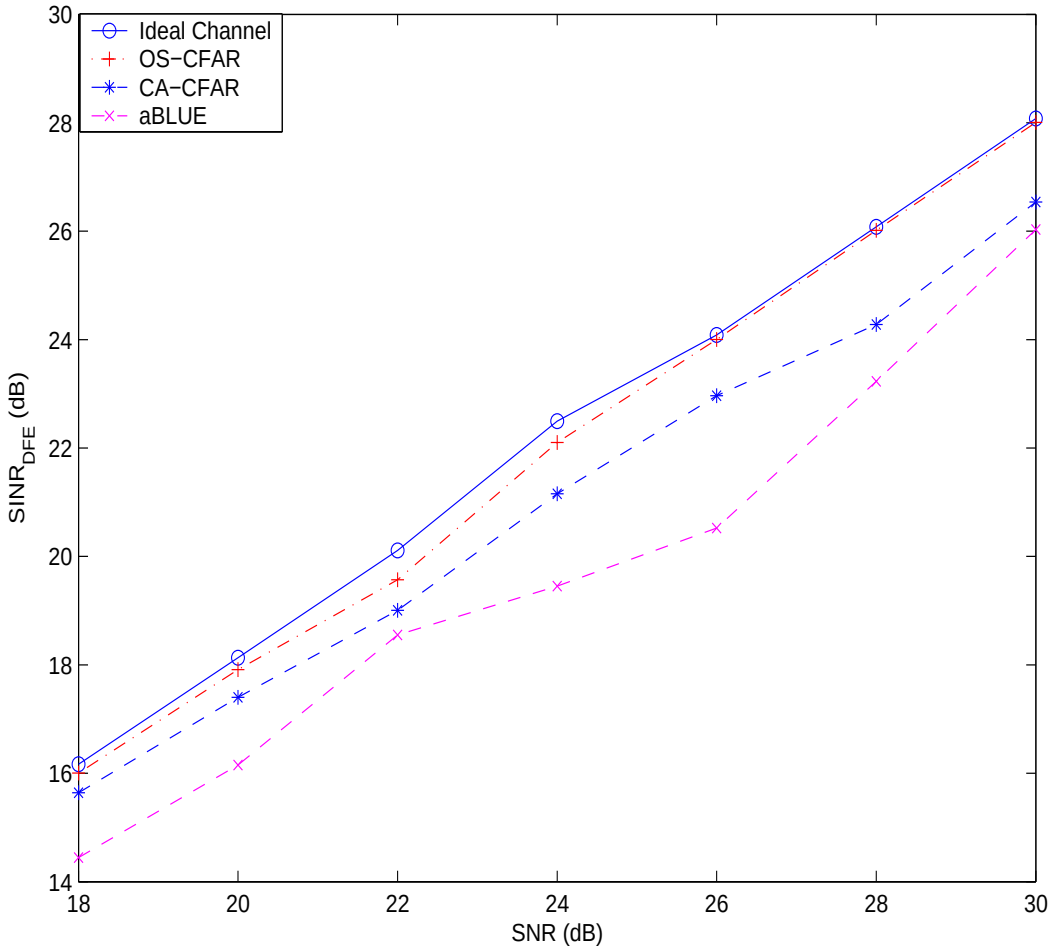

Figure 5.10: $\mathrm{SINR}_{\mathrm{DFE}}$ versus SNR for Channel 6 for various channel estimates.

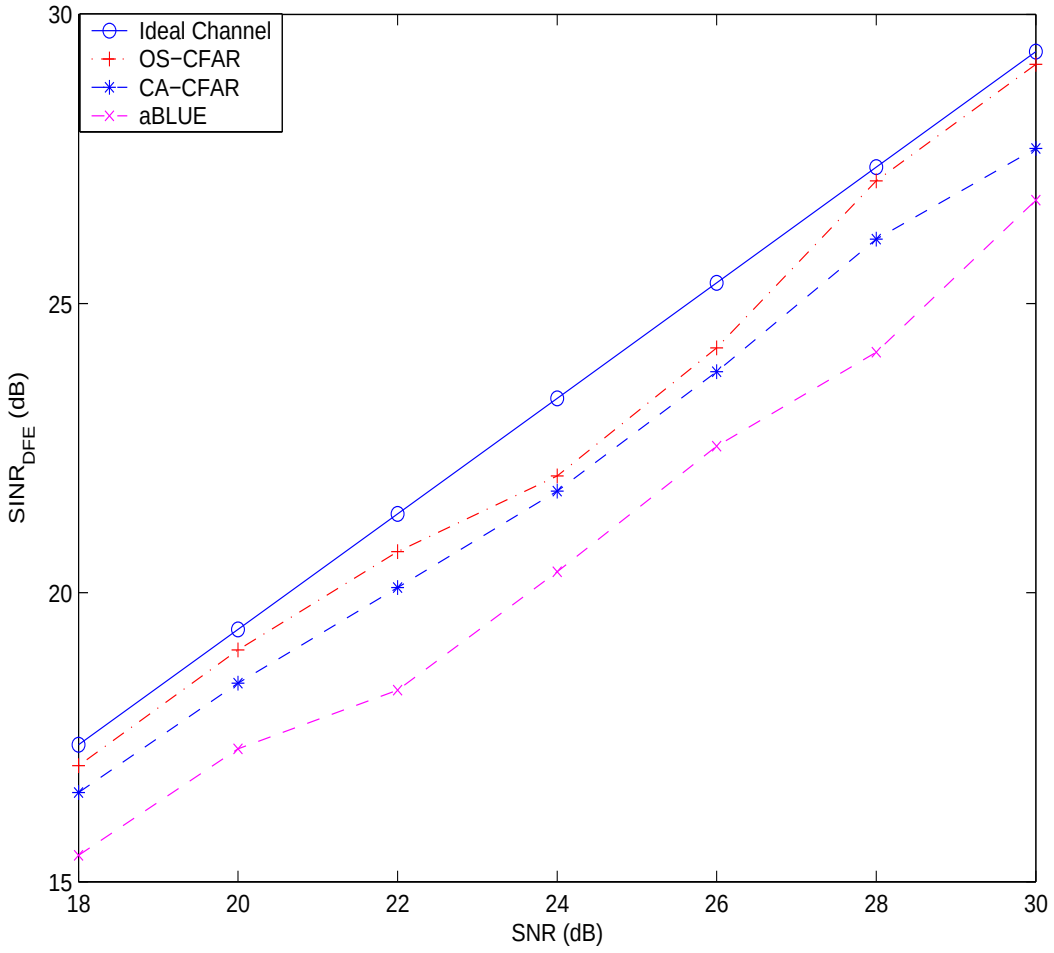

Figure 5.11: $\mathrm{SINR}_{\mathrm{DFE}}$ versus SNR for Channel 7 for various channel estimates.

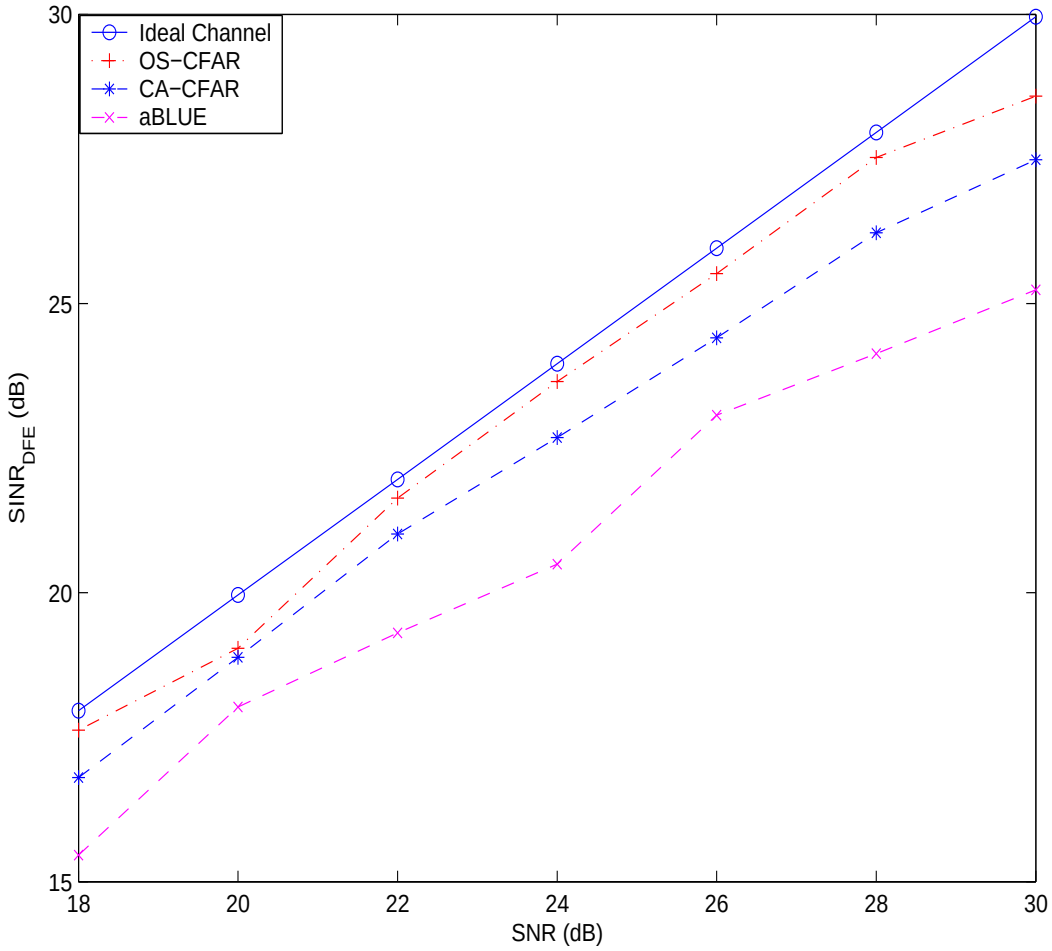

Figure 5.12: $\mathrm{SINR}_{\mathrm{DFE}}$ versus SNR for Channel 8 for various channel estimates.

REFERENCES

Abramowitz, M. and Stegun, I. A., 1988. *Handbook of Mathematical Functions*, (Dover Publications, New York), p. 398.

Advanced Television Systems Committee (ATSC) Standard A/53, 1995, ”Digital Television Standard”.

Barboy, B. and Lomes, A. and Perkalski, E., 1986. ”Cell-Averaging CFAR for Multiple Target Situations”, Proc. IEE (London), Vol. 133, pp. 176-186.

Casella, G. and Berger, R. L., 1990. *Statistical Inference*, (Duxbury Press), p.140.

Cotter, S. F. and Rao, B. D., 2002. ”Sparse Channel Estimation via Matching Pursuit with Application to Equaliztion”, *IEEE Trans. on Communications*, Vol. 50, pp. 374-377.

Cotter, S. F. and Rao, B. D., 2000. ”The Adaptive Matching Pursuit Algorithm for Estimation and Equalization of Sparse Time-Varying Channels”, Proceedings of the 34th Asilomar Conference on Signals, Systems and Computers, Monterey, California, (November), pp. 1772-1776.

Fimoff, M. and Özen, S. and Nereyanuru, S. and Zoltowski, M. D. and Hillery, W. 2001. ”Using The Result of 8-VSB Training Sequence Correlation As a Channel Estimate for DFE Tap Initialization”, Proceedings of 39th Annual Allerton Conference on Communication, Control and Computing, Illinois, (October).

Finn, H. M. and Johnson, R. S. 1968. ”Adaptive Detection Mode with Threshold Control as a Function of Spatially Sampled Clutter Estimates”, RCA Review, Vol. 29, pp. 414-464.

Hansen, G.V. and Sawyers, J.H., 1980. ”Detectability Loss Due To Greatest-Of Selection In A Cell-Averaging CFAR”, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 16, (October), pp. 115-118.

Hillery, W. J., 2001, ”8-VSB Digital TV Channel Model”, School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907-1285.

Hillery, W. J., 2002, ”Baseline Noise in Initial Channel Estimate for the Digital TV”, School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907-1285, (August).

Hojbjerre, M. and Sorensen, D. and Eriksen, P. S. and Andersen, H. H., 1995. *Linear and Graphical Models: For the Multivariate Complex Normal Distribution*, (Springer, New York), p. 40.

Krzanowski, W. J., 2000, *Principles of Multivariate Analysis: A User’s Perspective*, (Oxford University Press, New York), p. 205.

Levanon, N., 1988. *Radar Principles*, (John Wiley and Sons, New York), p. 247.

Özen, S. and Zoltowski, M. D. 2002. ”A novel channel estimation algorithm: Blending Correlation and Least-Squares Based Approaches”, Proceedings of International Conference on Acoustics, Speech and Signal Processing, ICASSP, Orlando, (May), Vol. 3, pp. 2281-2284.

Pladdy, C. and Özen, S. and Nereyanuru, S. and Zoltowski, M. D. and Fimoff, M. 2002. ”A Semi-blind Iterative Algorithm for Best Linear Unbiased Channel Estimation”, Zenith Electronics Corporation, Electronic Systems Research and Development, Lincolnshire, (October).

Rohling, H., 1983. ”Radar CFAR Thresholding in Clutter and Multiple Target Situations”, *IEEE Trans. on Aerospace and Electronic Systems*, AES-19, pp. 608-621.

Seber, G. A. F., 1977. *Linear Regression Analysis*, (John-Wiley and Sons).

Smith, M. E. and Varshney, P.K., 2000. ”Intelligent CFAR Processor Based on Data Variability”, *IEEE Transactions on Aerospace and Electronic Systems*, (March), Vol. 36, pp. 837-847.

Stüber, G. 1996. *Principles of Mobile Communication*, (Kluwer Academic Publishers, Massachusetts).

Zoltowski, M. D. and Hillery, W. and Özen, S. and Fimoff, M., 2002. ”Conjugate-Gradient-Based Decision Feedback Equalization With Structured Channel Estimation For Digital Television”, Proceedings of SPIE: *Digital Wireless Communications*, Vol. 4740.

APPENDIX A

8-VSB PULSE SHAPE

In this section we review the pulse shapes used by the 8-VSB digital TV systems ”(Hillery 2001)”. The 8-VSB DTV transmitter uses a transmit pulse shaping filter with complex root-raised cosine response. The transmit filter is denoted by $q(t)$. The receiver uses $q^*(-t)$ as the receive filter, which actually is the optimal matched filter for AWGN channel.

Here T denotes the symbol period, and $F_{sym} = 1/T$ is the symbol rate. Let

$$F_s = \frac{F_{sym}}{2} = \frac{1}{2T} \ . \tag{A.1}$$

The *roll-off factor*, which is also known as the *excess bandwidth*, for 8-VSB standard is $\beta = 0.115$.

1.1. Complex Root-Raised Cosine Pulse

The *complex root raised cosine pulse* $q(t)$ has the Fourier transform:

$$\begin{aligned} Q(F) &= \sqrt{P(F)} \\ &= Q_I(F) - jQ_Q(F) \\ &= \begin{cases} 1 & \frac{\beta}{2}F_s \le F \le (1-\frac{\beta}{2})F_s \\ \cos\left(\frac{\pi}{2\beta F_s}\left[F-(1-\frac{\beta}{2})F_s\right]\right) & (1-\frac{\beta}{2})F_s \le F \le (1+\frac{\beta}{2})F_s \\ \cos\left(\frac{\pi}{2\beta F_s}\left[F-\frac{\beta}{2}F_s\right]\right) & -\frac{\beta}{2}F_s \le F \le \frac{\beta}{2}F_s \end{cases} \end{aligned} \tag{A.2}$$

The in-phase and quadrature components are:

$$Q_I(F) = \begin{cases} \frac{1}{2} & \frac{\beta}{2}F_s \le |F| \le (1-\frac{\beta}{2})F_s \\ \frac{1}{\sqrt{2}}\cos\left(\frac{\pi}{2\beta F_s}F\right) & |F| \le \frac{\beta}{2}F_s \\ \frac{1}{2}\cos\left(\frac{\pi}{2\beta F_s}\left[|F|-(1-\frac{\beta}{2})F_s\right]\right) & (1-\frac{\beta}{2})F_s \le |F| \le (1+\frac{\beta}{2})F_s \end{cases} \tag{A.3}$$

$$Q_Q(F) = \begin{cases} \frac{1}{2}j\cos\left(\frac{\pi}{2\beta F_s}\left[F-(1-\frac{\beta}{2})F_s\right]\right) & (1-\frac{\beta}{2})F_s \le F \le (1+\frac{\beta}{2})F_s \\ \frac{1}{2}j & \frac{\beta}{2}F_s \le F \le (1-\frac{\beta}{2})F_s \\ \frac{1}{\sqrt{2}}j\sin\left(\frac{\pi}{2\beta F_s}F\right) & -\frac{\beta}{2}F_s \le F \le \frac{\beta}{2}F_s \\ -\frac{1}{2}j & -(1-\frac{\beta}{2})F_s \le F \le -\frac{\beta}{2}F_s \\ -\frac{1}{2}j\cos\left(\frac{\pi}{2\beta F_s}\left[F+(1-\frac{\beta}{2})F_s\right]\right) & -(1+\frac{\beta}{2})F_s \le F \le -(1-\frac{\beta}{2})F_s \end{cases} \quad \text{(A.4)}$$

The time domain signal is:

$$q(t) = e^{j\pi F_s t}p_{RRC}(t) \quad \text{(A.5)}$$

where

$$p_{RRC}(t) = \frac{F_s}{1-(4\beta F_s t)^2}\left[(1-\beta)\frac{\sin\left(\pi(1-\beta)F_s t\right)}{\pi(1-\beta)F_s t} + \frac{4\beta}{\pi}\cos\left(\pi(1+\beta)F_s t\right)\right] \quad \text{(A.6)}$$

Defining the real and imaginary parts of $q(t)$ by $q_I(t)$ and $q_Q(t)$, it is straightforward to show that

$$q_I(t) = q_I(-t), \quad \text{(A.7)}$$

$$q_Q(t) = -q_Q(-t), \quad \text{(A.8)}$$

which implies

$$q(t) = q^*(-t). \quad \text{(A.9)}$$

Equation (A.9) means that the pulse shape $q(t)$ is Hermitian symmetric. Due to this symmetry property, the receive filter is the same as the transmit filter.

1.2. Complex Raised Cosine Pulse

The convolution of the transmit and receive filters is denoted by $p(t)$ and is the composite pulse shape which is given by

$$p(t) = q(t) * q^*(-t) = q(t) * q(t). \quad \text{(A.10)}$$

The second part of the equality of Equation (A.10) follows from Equation(A.9).

The *complex raised cosine pulse* $p(t)$ has the Fourier transform:

$$\begin{aligned} P(F) &= P_I(F) - jP_Q(F) \\ &= \begin{cases} 1 & \frac{\beta}{2}F_s \le F \le (1-\frac{\beta}{2})F_s \\ 0.5\left[1+\cos\left(\frac{\pi}{\beta F_s}\left[F-(1-\frac{\beta}{2})F_s\right]\right)\right] & (1-\frac{\beta}{2})F_s \le F \le (1+\frac{\beta}{2})F_s \\ 0.5\left[1+\sin\left(\frac{\pi}{\beta F_s}F\right)\right] & -\frac{\beta}{2}F_s \le F \le \frac{\beta}{2}F_s \end{cases} \end{aligned} \tag{A.11}$$

The in-phase and quadrature components are:

$$P_I(F) = \begin{cases} 0.5 & 0 \le |F| \le (1-\frac{\beta}{2})F_s \\ 0.25\left[1+\cos\left(\frac{\pi}{\beta F_s}\left[|F|-(1-\frac{\beta}{2})F_s\right]\right)\right] & (1-\frac{\beta}{2})F_s \le |F| \le (1+\frac{\beta}{2})F_s \end{cases} \tag{A.12}$$

$$P_Q(F) = \begin{cases} 0.25j\left[1+\cos\left(\frac{\pi}{\beta F_s}\left[F-(1-\frac{\beta}{2})F_s\right]\right)\right] & (1-\frac{\beta}{2})F_s \le F \le (1+\frac{\beta}{2})F_s \\ 0.5j & \frac{\beta}{2}F_s \le F \le (1-\frac{\beta}{2})F_s \\ 0.5j\sin\left(\frac{\pi}{\beta F_s}F\right) & -\frac{\beta}{2}F_s \le F \le \frac{\beta}{2}F_s \\ -0.5j & -(1-\frac{\beta}{2})F_s \le F \le -\frac{\beta}{2}F_s \\ -0.25j\left[1+\cos\left(\frac{\pi}{\beta F_s}\left[F+(1-\frac{\beta}{2})F_s\right]\right)\right] & -(1+\frac{\beta}{2})F_s \le F \le -(1-\frac{\beta}{2})F_s \end{cases} \tag{A.13}$$

The time domain signal is:

$$p(t) = e^{j\pi F_s t}p_{RC}(t) \tag{A.14}$$

where

$$p_{RC}(t) = F_s\frac{\sin(\pi F_s t)}{\pi F_s t}\frac{\cos(\pi\beta F_s t)}{1-(2\beta F_s t)^2}, \tag{A.15}$$

is the raised cosine pulse.

APPENDIX B

TEST CHANNELS

We have provided 9 channel impulse responses to test channel estimation algorithms as well as the DFE performance. The following Table B lists the channel delays (TOAs) $\{\tau_k\}$ in symbol periods (T), and the relative gains $\{|c_k|\}$.

Taps	Channel 1		Channel 2		Channel 3	
k	$\{\tau_k\}$	$\{\|c_k\|\}$	$\{\tau_k\}$	$\{\|c_k\|\}$	$\{\tau_k\}$	$\{\|c_k\|\}$
–2						
–1					-0.957839182	0.7263
0	**0**	**1**	**0**	**1**	**0**	**1**
1	1.6143357	0.2045	3.2286714	0.2512	3.55153854	0.6457
2	23.89216836	0.1548	37.667833	0.631	15.25009125	0.9848
3	32.8248259	0.179	47.3538472	0.4467	24.03207745	0.7456
4	63.06671468	0.2078	102.241261	0.1778	29.16566498]	0.8616
5	63.82007134	0.1509	136.6804226	0.0794		
6						
7						
Taps	Channel 4		Channel 5		Channel 6	
k	$\{\tau_k\}$	$\{\|c_k\|\}$	$\{\tau_k\}$	$\{\|c_k\|\}$	$\{\tau_k\}$	$\{\|c_k\|\}$
–2						
–1						
0	**0**	**1**	**0**	$\mathbf{1 * exp(j * pi)}$	**0**	**1**
1	5.16587424	0.65575	10.762238	1	15.541	0.46388
2	22.27783266	0.75697	21.524476	1	28.39	0.54405
3	31.2104902	0.87482			246.66	0.22324
4	61.45237898	1.01565				
5	62.20573564	0.7379				
6						
7						
Taps	Channel 7		Channel 8		Channel 3-plus	
k	$\{\tau_k\}$	$\{\|c_k\|\}$	$\{\tau_k\}$	$\{\|c_k\|\}$	$\{\tau_k\}$	$\{\|c_k\|\}$
–2						
–1					-0.957839182	0.7263
0	**0**	**1**	**0**	**1**	**0**	**1**
1	1.6143357	0.2045	32.8248259	0.1	3.55153854	0.6457
2	23.89216836	0.15			15.25009125	0.9848
3	32.8248259	0.1			24.03207745	0.7456
4	63.06671468	0.2078			29.16566498	0.8616
5	63.82007134	0.1509			221.2345	0.315
6					332.981	0.349
7						

Table B.1: Simulated 9 Channel Impulse Responses. The channel delays (TOAs) $\{\tau_k\}$ in symbol periods (T), and the relative gains $\{|c_k|\}$