INPUT SEQUENCE ESTIMATION AND BLIND CHANNEL IDENTIFICATION IN HF COMMUNICATION

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCES
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By
M. Khamees Ben Hadj Miled
August 1999

INPUT SEQUENCE ESTIMATION AND BLIND CHANNEL IDENTIFICATION IN HF COMMUNICATION

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND

ELECTRONICS ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCES

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

M. Khames Ben Hadj Miled

August 1999

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Orhan Arıkan(Supervisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Enis Çetin

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Murat Alanyalı

Approved for the Institute of Engineering and Sciences:

Prof. Dr. Mehmet Baray
Director of Institute of Engineering and Sciences

ABSTRACT

INPUT SEQUENCE ESTIMATION AND BLIND CHANNEL IDENTIFICATION IN HF COMMUNICATION

M. Khames Ben Hadj Miled
M.S. in Electrical and Electronics Engineering
Supervisor: Assist. Prof. Dr. Orhan Arıkan
August 1999

Recent advances in blind channel equalization approaches and the availability of fast processors have made it possible to communicate reliably over long distances through HF communication links. Current research efforts focus on the improvement of the performance of the communication systems which degrades significantly during the "bad tropospheric conditions" when the channel characteristics show rapid variations.

In order to improve the performance of the HF communication links during these conditions, algorithms that can identify and track the channel characteristics are proposed in this thesis. Detailed simulation based comparisons with the existing algorithms show that the proposed approaches significantly improve the performance of the communication system and enable us to utilize HF communication in bad conditions even at 10 dB SNR.

Keywords: Blind channel identification, blind signal estimation, Kalman Filter, subspace tracker, fractional sampling.

ÖZET

GİRİŞİ BİLİNMEYEN İLETİŞİM KANALLARININ HATASIZ TANINMASI

M. Khames Ben Hadj Miled
Elektrik ve Elektronik Mühendisliği Bölümü Yüksek Lisans
Tez Yöneticisi: Yrd. Doç. Dr. Orhan Arıkan
Ağustos 1999

Gözü kapalı olarak kanal denkleştirme konusunda yakın zamanda geliştirilmiş olan yeni yöntemler hızlı sayısal işlemciler sayesinde kullanılarak HF linkleri üzerinden güvenilir şekilde haberleşmeyi olanaklı kılmışlardır. Dvam etmekte olan araştırmalr iletişim performansının önemli ölçüde düştüğü kanal karakteristiğinin hızlı değişim gösterdiği kötü atmosferik şartlarda artırılabilmesini amaçlamaktadir. Bu tezde HF linkleri üzerinden yapılan iletişimin performansını artırabilmek amacı ile HF kanalını tanıyabilecek ve zaman içerisindeki değişmini takip edebilecek yöntemler önerilmiştir. Var olan yöntemler ile yapilan benzetimlere dayalı detaylı kıyaslamalarda geliştirilen yöntemlerin önemli bir performans artırımı sağlayarak kötü atmosferik şartlarda 10 dB sinyal-gürültü oranında dahi iletişimi olanaklı kıldıkları gözlenmiştir.

Anahtar Kelimeler: Gözü kapalı kanal tanınması, gözü kapalı ters evrişim, sistem tanınması, kesirli örnekleme

ACKNOWLEDGMENTS

I would like to use this opportunity to express my deep gratitude to my supervisor Assist. Prof. Dr. Orhan Arıkan for his guidance, suggestions and invaluable encouragement throughout the development of this thesis.

I would like to thank Prof. Dr. Enis Çetin and Assist. Prof. Dr. Murat Alanyalı for reading and commenting on the thesis.

I express my special thanks to my family for their constant support, patience and sincere love.

Finally, many thanks to all of my close friends.

Contents

List of Figures

List of Tables

To the spirit of my father ...

Chapter 1

INTRODUCTION

Digital communication systems usually suffer from inter-symbol interference, ISI. This phenomenon is known to be caused by the channel memory, which spreads the transmitted symbols in time, or due to time-varying multi-paths. To combat the limitation in performance due to such factor, blind channel equalizers are usually built within receivers. The blind equalization techniques proposed in the literature, either perform direct equalization, which is the case of decision feedback equalizers, DFE, [3] [4], [5], or divide the problem into blind channel identification , BCI, and blind signal estimation, BSE, [6], [7] [8] [9] [10]. The channel identification is based either on statistical properties of the received sequence [6], [9], or on subspace methods, mainly in the case of multi-channel systems or fractionally spaced channels. In [10], a review, describing the main ideas behind statistical and deterministic approaches in BCI, is presented.

In this work, we address the problem of blind equalization for HF channels. These are mainly characterized by time varying paths and additive noise. The time variation in the channel response leads to a degradation in the performance of the equalizer as time progresses. As a result, a periodic transmission of a training sequence is required, which means a poor management of the channel bandwidth. Moreover, even with the use of such periodic sequences, the equalizer may fail and result connection break-down in poor conditions.

Recent research in the subject, tried to come up with robust equalizers [2], [11], and to avoid training sequences [4]. Different approaches like improvements of decision feed-back equalizers and adaptive algorithms were proposed. A commonly used technique is fractional sampling [10]. The latter provides channel diversity which helps for better tracking of the variation in the channel impulse response. Moreover, with the use of an appropriate low-pass filter, the fractionally spaced channels' output noise will have a considerably reduced variance.

In this thesis work, we propose distinct iterative approaches, assuming a fractionally spaced model of the channel, that aim a robust HF channel equalization. Starting with a training period, a reliable estimate of the channel can be obtained. Then at each iteration, we make use of the previous channel estimate to predict the input symbols and then use them to update the channel transfer function estimate. To ensure the convergence of the equalizer over a long period, we developed a K-delayed input sequence estimator that minimizes the cumulative mean square error over the last K output measurements. We also suggest different algorithms for the channel identification purpose.

The channel identifier consists of a cascade of two different blocks. The first is a Kalman filter that makes use of the estimated input vector to provide an estimate for the channel impulse response. The latter will be fed to a subspace tracker to update an estimate for a basis of the channel subspace. Once a reliable estimate is obtained, significant part of the noise can be eliminated through a projection. The described approach is based on the assumption that the slowly time varying channel belongs to a low-rank subspace. Based on different descriptions of the problem, distinct channel identification methods are developed.

Another relevant issue in HF communication is the modeling of the HF system. Any proposed model should reflect the physical characteristics of the transmission medium. A simple and commonly used model, proposed in [12], reflects the multi-path nature of the HF channel and the randomness in path delays and amplitude distortion. As a part of our contribution, we extend the described model a more realistic one that takes into account the limited bandwidth of the channel and the physical characteristics of the reflective ionospheric layers.

The proposed approaches are simulated for different noise realizations and time-variation conditions. The performance of the channel identification algorithms is tested in both, blind and open-eye, cases. In the case of known input sequence, all the proposed algorithms show rapid convergence and relatively small estimation error.

The organization of the thesis is as follows: First an overview of the HF communication is given and the new HF system model is described. In Chapter 3, the problem formulation is presented. Then, a description of the input

sequence estimator is provided. The different channel identification methods are explained and the algorithms are explicitly stated and compared in terms of performance and computational cost. In Chapter 4, simulation results of the best performing algorithms are shown and compared to the ones corresponding to already existing approaches. Finally, the thesis is concluded.

Chapter 2

AN OVERVIEW OF HF COMMUNICATION

2.1 A Typical HF Communication System

In HF communication, the transmission medium in between the transmitter and the receiver systems is the atmosphere. Transmitted signals may follow multiple paths as they are reflected by distinct ionospheric layers. A simple model for such channels was proposed in 1969 by Watterson [12], and experimentally verified in 1970, [13]. The model represents each signal path by a delayed impulse, the magnitude of which changes in time to reflect the random nature of the reflections in the ionosphere. In the Watterson model, surface ionospheric layers are assumed to be as shown in Figure 2.1, and each transmission path is represented by a single delayed impulse. As a result, the individual reflections in each multi-path may cause amplitude distortion and delay but

do not change the shape of a the transmitted signals. Therefore, the baseband

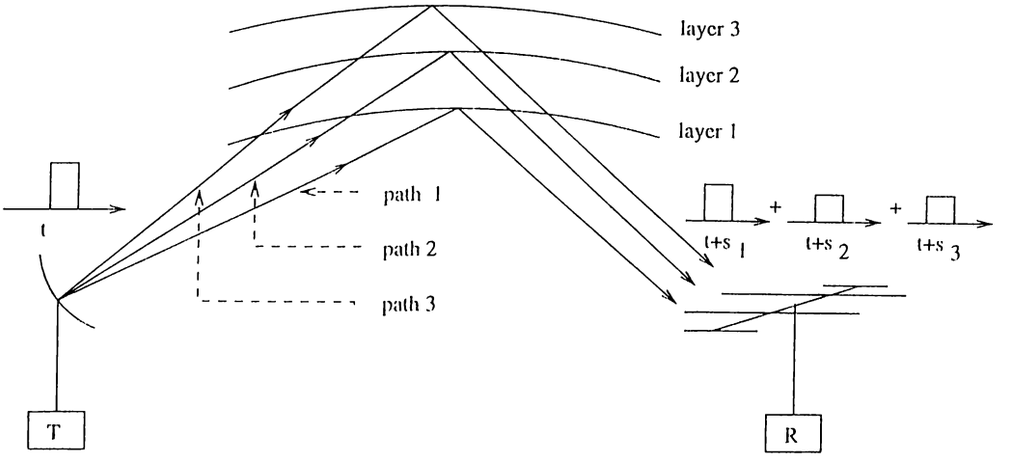

Figure 2.1: Surface ionospheric layers' model.

channel I/O equation for the Watterson model can be written as;

$$y(t) = \sum_i G_i(t)x(t - s_i) \tag{2.1}$$

where $G_i(t)$ models the time varying reflection and it is obtained by filtering white noise with a Gaussian function.

An improvement of the Watterson model can be obtained by incorporating diffuse reflective layers to the model (see Figure 2.2). This can be achieved approximately by modeling the channel transfer function as a sum of shifted Gaussian functions, each of which corresponding to a distinct transmission path. The intuition behind such a choice is the assumption that the reflection points are independent and identically distributed in the space occupied by the ionospheric layer. The improved baseband channel model becomes:

$$y(t) = \sum_i G_i(t)[x(t) * f_i(t - s_i)] \tag{2.2}$$

where $f_i(t) = \frac{1}{\sqrt{2\pi}\gamma_i} exp(\frac{-t^2}{2\gamma_i^2})$ and $(*)$ denoting the convolution operation. By choosing the shape: $G_i(t)$ and γ_i, and location: s_i, parameters of each channel

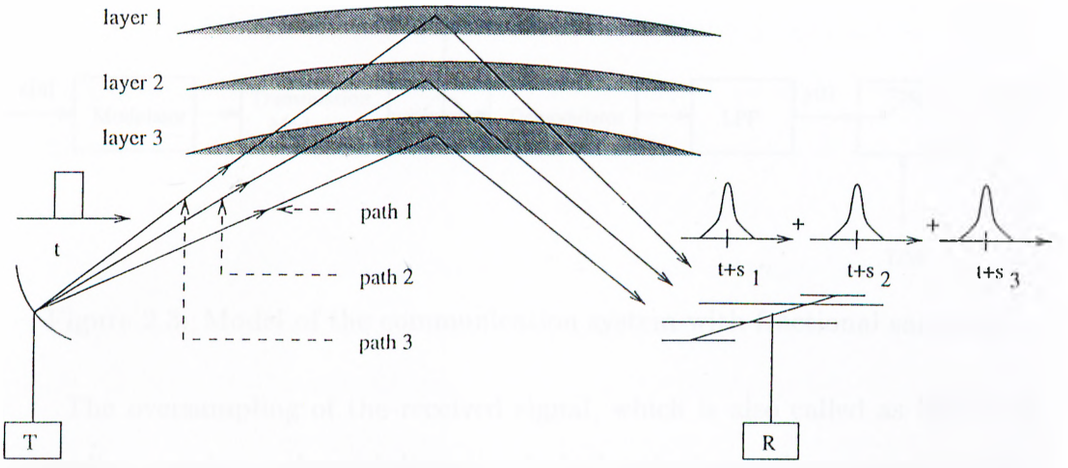

Figure 2.2: Diffused ionospheric layers' model.

transfer function as time-varying, various important physical characteristics of the HF channels such as the altitude and the thickness variations in the reflective layers can be modeled.

2.2 A Model for the Communication System.

A typical HF communication system with serial data-transmission can be modeled as shown in Figure 2.3. The input data symbols $x[n]$ are chosen from a finite alphabet. The transmission medium is the atmosphere, as previously described, and $y(t)$ and $w(t)$ are, respectively, the received and additive noise signals. Following the demodulator, the signal is low-pass filtered with a pass-band equal to that of the input signal bandwidth. Then the output of the low-pass filter is sampled with an over-sampling factor of M where T corresponds to the input symbol duration. The channel transfer function is assumed to be the cascade of the transmitter filter, the transmission medium, and the receiver filter.

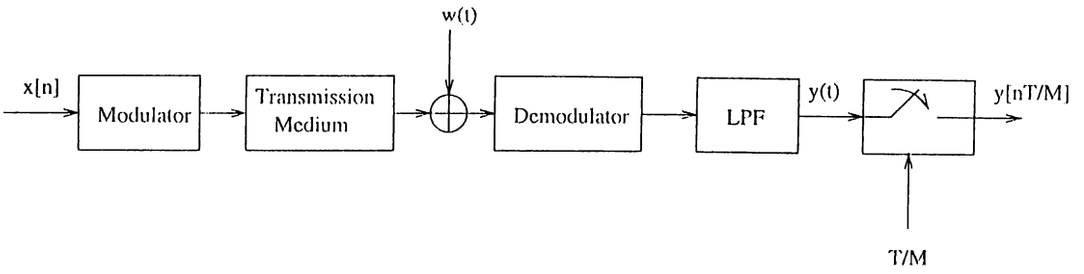

Figure 2.3: Model of the communication system with fractional sampling.

The oversampling of the received signal, which is also called as fractional sampling, results in channel diversity. In [14], it is shown that using fractional sampling, the communication system model becomes equivalent to a single-input multiple-output, SIMO, system as shown in Figure 2.4. This technique was extensively exploited in the literature, and the identifiability conditions for fractionally spaced channel are considered in [15] and [16].

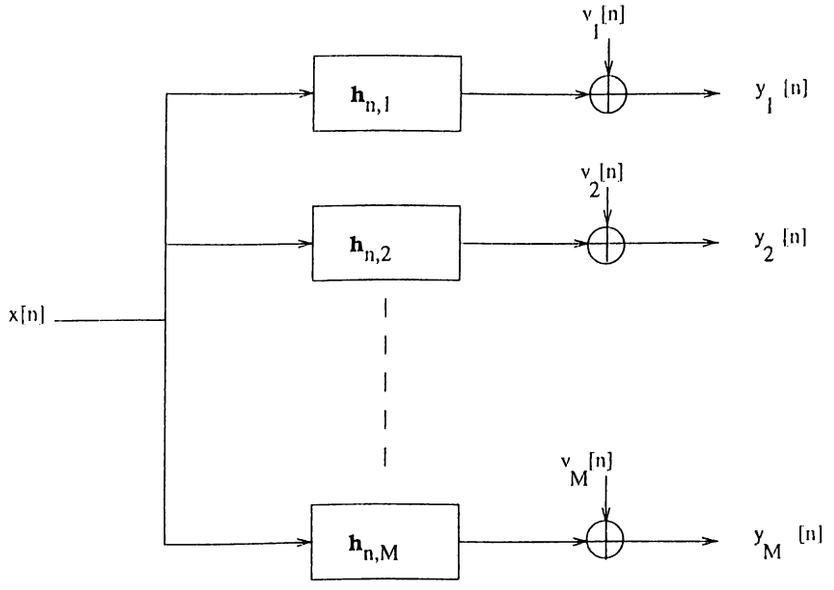

Figure 2.4: Multi-channel filter model of the baseband equivalent of the communication system.

In Figure 2.3, the noise signal $w(t)$ is low-pass filtered and then fractionally sampled to get the noise sequences $v_i[n]$, $i = 1, \ldots, M$, shown in Figure 2.4. In Appendix A, we show that the low-pass filter,LPF, together with the fractional sampling technique reduce the noise variance by a factor of $1/M$. Moreover the noise samples become correlated and the noise covariance matrix acquires a special form that depends on the LPF impulse response. Assuming a known noise variance, the noise covariance matrix can be precomputed. The required noise statistics can be estimated during silent intervals, or even during data transmission by computing spectral energy of the demodulator output out of the spectral support of the signal.

In this thesis we will address the following problem: given the output measurements $y_i[n]$, $i = 1, 2, \ldots, M$, and the second order statistics of the additive noise processes $v_i[n]$, obtain reliable estimates to the input sequence $\{x[n]\}_{n=0}^{\infty}$ and the channel transfer functions $\mathbf{h}_{i,n}$, (see Figure 2.4) .

In the following chapter, this problem will be treated in detail and the proposed algorithms will be presented.

Chapter 3

BLIND ESTIMATION OF INPUT SYMBOL SEQUENCE AND IDENTIFICATION OF THE CHANNEL RESPONSE

In HF communication, as in most of the communication systems, the ultimate purpose is to be able to estimate the transmitted symbol sequence as reliable as possible at the receiver. However, since the medium of transmission is the HF tropospheric channel, the receiver has to provide estimates to the input symbols in the absence of a precise channel transfer function. This problem is faced also in the mobile communication systems and has been commonly referred to as the problem of " blind estimation of the input symbol sequence". Another important problem in the HF communication is the identification and tracking of the time-varying HF channel response when the channel input sequence is

unknown. This latter problem is commonly referred to as "blind channel identification" and has found application in many other communication systems. As it can be expected the blind channel identification and input sequence estimation problems are very much related to each other, and in most applications a solution to one of them requires a solution to the other one. Therefore in the approaches proposed in this thesis, the problems of blind channel identification and input symbol estimation are iteratively solved by making use of the solution to one to get a solution to the other.

3.1 Input Sequence Estimation

In this section, we describe a method to estimate the input symbols when they are chosen from a binary alphabet. Assuming that the individual channel responses are of finite duration: $\mathbf{h}_{i,n} = [h_{i,n}[0], h_{i,n}[1], ..., h_{i,n}[L-1]]^T$,where L is the channel order, the individual channel outputs can be written as:

$$y_i[n] = \mathbf{h}_{i,n}^T \mathbf{x}_n + v_i[n] \tag{3.1}$$

where $\mathbf{x}_n = [x[n], x[n-1], ..., x[n-L+1]]^T$ is the channel input vector. Now, a precise statement of the problem can be given as:

$$\begin{aligned} &\textit{estimate} \quad x[n] \in \{\mp A\}, \quad \text{for } n \geq 0 \\ &\textit{by using} \quad y_i[n] = \mathbf{h}_{i,n}^T \mathbf{x}_n + v_i[n], \quad \text{for } 1 \leq i \leq M. \end{aligned}$$

As seen in the above formulation, the available measurements $y_i[n]$ depend on on the unknowns of the problem: $\mathbf{x}_n$ and $\mathbf{h}_{i,n}$ in the multiplicative form. Hence, the input sequence cannot be reliably estimated by using efficient estimators such as the Kalman filter.

In the following, we will provide alternative solution approaches where, after obtaining an initial estimate to the channel by using a short training period in which a known sequence is transmitted, the input sequence will be estimated by using the estimated channel and then the channel estimate will be updated by using the estimated input sequence. A block diagram for this iterative solution technique is shown in Figure 3.1.

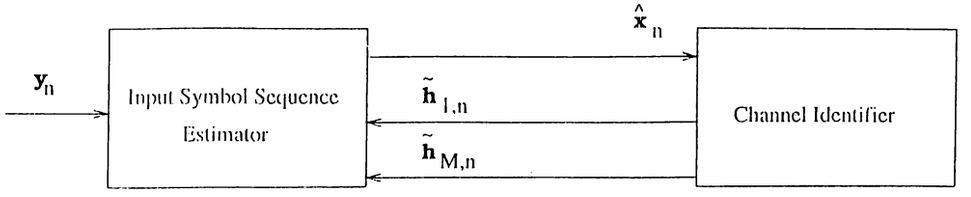

Figure 3.1: Block diagram of the iterative solution.

3.1.1 Estimation of Input Sequence in the Presence of a Channel Estimate

In this section, we will consider the estimation of the input sequence by assuming the availability of a reliable channel estimate $\tilde{\mathbf{h}}_n$. The statement of the input sequence estimation problem becomes:

Given $\tilde{\mathbf{h}}_{i,n}$ and the covariance of the zero-mean noise $v_i[n]$;

$$\begin{aligned} &\textit{estimate} \quad x[n] \in \{\mp A\}, \quad \text{for } n \geq 0 \\ &\textit{by using} \quad y_i[n] = \tilde{\mathbf{h}}_{i,n}^T \mathbf{x}_n + v_i[n], \quad \text{for } 1 \leq i \leq M. \end{aligned}$$

Many alternative estimation techniques are available for the above problem. Alternatives include Kalman filter and Viterbi algorithm [17]. In the following, we will propose an efficient but sub-optimal estimator for the input sequence estimation.

Given the past input symbols $x[n-1],\ldots,x[n-L+1]$, let $\mathbf{x}_n^1 = [A\ x[n-1]\ldots x[n-L+1]]^T$ and $\mathbf{x}_n^2 = [-A\ x[n-1]\ldots x[n-L+1]]^T$ be the two possible vector values of $\mathbf{x}_n$ when $x[n]$ is not known. Let $\tilde{\mathbf{h}}_{1,n-1},\ldots,\tilde{\mathbf{h}}_{M,n-1}$, be the last estimates of the sub-channels given by the channel identifier. The latters can be taken as reliable estimates of $\mathbf{h}_{i,n}$'s as far as the channel slow-variation assumption holds. We define the estimate of $\mathbf{x}_n$ as:

$$\hat{\mathbf{x}}_n = arg\min_{\mathbf{x}_n^q} \sum_{i=1}^{M} \left(y_i[n] - \tilde{\mathbf{h}}_{i,n-1}^T \mathbf{x}_n^q\right)^2 \qquad ,q = 1,2. \tag{3.2}$$

Note that the estimate of $x[n]$ can be extracted from $\mathbf{x}_n$.

In the case where the sub-channels are not sensitive to the present input symbol $x[n]$, i.e, $h_{i,n}[0] \simeq 0$, the output noise and the channels' identification error may drastically degrade the performance of the described estimator. Hence, a more realistic goal would be to estimate $x[n-K], K < L$, instead of $x[n]$, that is a K-delayed input decision.

Let $\mathbf{x}_n^q$, $q = 1,\ldots,2^{K+1}$, be the possible vector values of $\mathbf{x}_n$ when only the input symbols prior to $x[n-K]$ are known, i.e, $\mathbf{x}_n^1 = [A\ldots A\ x[n-K-1]\ldots x[n-L+1]]^T$. The estimate $\hat{\mathbf{x}}_n$ is defined as:

$$\hat{\mathbf{x}}_n = arg\min_{\mathbf{x}_n^q} \left| \left[\frac{1}{M}\sum_{i=1}^{M} \frac{1}{K+1} \sum_{k=0}^{K} \left(y_i[n-k] - \tilde{\mathbf{h}}_{i,n-1}^T \mathbf{x}_{n-k}^q\right)^2\right] - \sigma^2 \right| \ ,q = 1,2,....,2^{K+1}. \tag{3.3}$$

In the last cost function, $\tilde{\mathbf{h}}_{i,n-1}$, $i = 1,\ldots,M$, are used as estimates of $\tilde{\mathbf{h}}_{i,n},\ldots,\tilde{\mathbf{h}}_{i,n-K}$. Such an approximation is justified by the channel slow variation assumption and the reliable performance of the channel identifier.

By looking at the cost function, the algorithm seems to be computationally exhaustive. However, under the condition that $x[n] \in \{\pm A\}$, the terms

$\tilde{\mathbf{h}}_{i,n-k}^T \mathbf{x}_{n-k}^q$, which involve most of the multiplications in the cost function, can be seen as a summation of the terms $\pm A\tilde{h}_{i,n}[0], \pm A\tilde{h}_{i,n}[1], \ldots, \pm A\tilde{h}_{i,n}[L-1]$. If we let,

$$e_{i,n-k}^q = y_i[n-k] - \hat{\mathbf{h}}_{i,n-1}^T \mathbf{x}_{n-k}^q, \tag{3.4}$$

then a simple implementation of the hardware responsible for calculating $e_{i,n-k}^q$'s would be as shown in the figure below. Note that, the input symbols transmitted prior to x[n-k] are assumed to be known because the estimator should have decided on their values at previous iterations. With the described implementation, the computational cost of the input symbol sequence estimator can be reduced to $ML + (K+1)M$ instead of $ML^2 + (K+1)M$ multiplications.

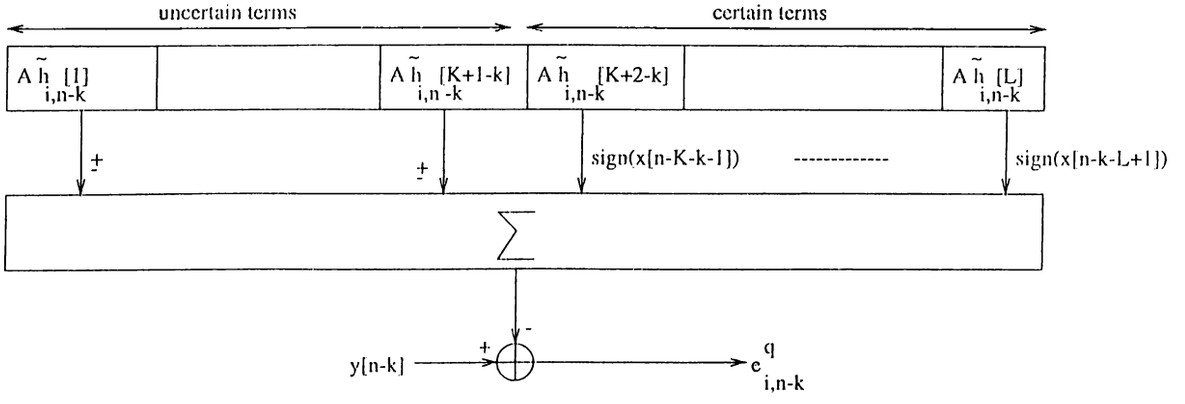

Figure 3.2: Efficient implementation of the signal estimator.

In Appendix B, the input sequence estimator is shown to converge to the correct solution. However, the output noise and channel estimation error may result in wrong decisions. In order to avoid such events, K should be chosen such that $h_{i,n}[K]$ is as large as possible for $i = 1, \ldots, M$. A safe choice would be $K \geq arg\max_j h_{i,n}[j]$ or even a close value would be fine.

3.2 Blind Channel Identification

The problem of blind channel identification has a variety of applications such as blind equalization of communication systems, seismic data processing, speech processing, and image restoration. In all these applications, the channel identification helps overcoming a distortion phenomenon and estimating an original signal. Furthermore, blind identification of tropospheric channels helps to probe into the physical characteristics of the troposphere [18], [19].

In this section, we focus on the case of noisy and slowly time-varying channels.Such channels are relevant, especially, in HF communications. The slow time-variation is assumed within one iteration. However, rapid variation is allowed in the long term sens.First attempts in blind equalization proposed periodic transmission of training sequences to update the channel estimates in equal time intervals [2]. Such approaches suffer an important loss in throughput and the effective channel rate may reduce to 50 % [10]. Current research on this issue aims at less or no use of training periods with negligible loss in the performance of the estimator. [4]. A commonly used method is the creation of redundancy and channel diversity through the fractionally spaced channel implementation described in Figure 2.4 [10]. However, even with fractional sampling, the problem still requires robust algorithms with fast converging rates in order to track the channel variation. As stated in [20], in the case of a ten-tap-weight FIR channel, to achieve convergence some of the previously proposed approaches may require more than 1000 training symbols.

Given the output measurements $y_i[n]$, $i = 1, \ldots, M$, and the covariances of the noise sequences $v_i[n]$;

$$\begin{aligned} &estimate \quad \mathbf{h}_{i,n} && \text{for } n \geq 0 \text{ and } 1 \leq i \leq M \\ &by\ using \quad y_i[n] &&= \mathbf{h}_{i,n}^T \mathbf{x}_n + v_i[n], \text{ for } 1 \leq i \leq M. \end{aligned}$$

Since both $\mathbf{x}_n$ and $\mathbf{h}_{i,n}$ are unknown, the channels' identification problem becomes a difficult task. Most of the already proposed approaches assume some prior knowledge of the input sequence.

Different methods and techniques were exploited in the BCI problem. They differ, mainly, according to the assumed prior knowledge about the input sequence and the considered channel model. An important and commonly used class of BCI approaches is known as the subspace methods. The idea consists of minimization of the following quadratic cost function [10],

$$\hat{\mathbf{h}} = arg \min_{\mathbf{h} \in S} \mathbf{h}^H \mathbf{Q} \mathbf{h}, \tag{3.5}$$

where $\mathbf{h}$ is the channel impulse response, which is in subspace S. In one avenue of approaches known as the cross relation [9], [10], the strong correlation between channel output pairs is exploited. Alternatively in the subspace approaches, the orthogonality property between the noise and the fractionally spaced channel subspaces is exploited in the estimation of $\mathbf{h}$ [20]. These methods, together with the least squares smoothing technique, assume no prior knowledge about the input signal. In the case where the second order statistics of the input sequence are known, the cyclostationarity property of the output signal is exploited, either in the time domain [6], [21], or in the frequency domain [8], to get an estimate of the channel transfer function. In all these approaches, the subspace algorithms are applied on the output sequence $y[n]$.

Apart from the subspace methods, the maximum likelihood estimator and the moment matching techniques were also proposed as competing alternatives in BCI problem.

In the presence of reliable estimates to the input, $\hat{\mathbf{x}}_n$, adaptive filters have been used with considerable success as well [1], [2] [22], [23] . In [1], a modified fast transversal filter is proposed and shown to achieve nice performance under the assumption of correct input decision. In [2], a set of efficient channel estimators were developed based on a state space representation that describes the channel as a linear combination of the basis vectors spanning a subspace. Although these algorithms do not assume the multi-channel model, their extension to the fractionally spaced channel is straight forward. However, they assume the periodic transmission of a training sequence. In the simulation part, we'll use these last two approaches as reference methods because they are known to perform well in the case of noisy time-varying channels.

In the next section we'll propose a set of new approaches to the channel identification problem in the presence of an estimate of the input sequence.

3.2.1 Channel Identification in the Presence of Input Sequence Estimate

In this section, we address the channel identification problem in the case of available reliable input sequence estimate. In other words, given $\hat{\mathbf{x}}_n$ and the received signals $y_i[n]$:

$$estimate \quad \mathbf{h}_{i,n} \qquad \text{for } n \geq 0 \text{ and } 1 \leq i \leq M$$

$$\textit{by using} \quad y_i[n] \;=\; \hat{\mathbf{x}}_n^T \mathbf{h}_{i,n} + v_i[n], \quad \text{for} \;\; 1 \leq i \leq M.$$

We will propose a spectrum of algorithms within this section. We present the proposed approaches and put them in perspective with some of the already existing approaches to clarify the intuition behind our contribution.

In the proposed approaches, the time variation in the channel response will be tracked by slowly rotating the basis vectors of the subspace S. Hence, we are after the tracking of the subspace basis which will help for a better tracking of the channel variation. First, by using the available estimates to the input sequence $x[n]$, an adaptive filter is used to get an estimate $\hat{\mathbf{h}}_n$ of the oversampled channel response vector $\mathbf{h}_n$. Second, a subspace tracker makes use of this estimate to update the subspace basis. Finally a more refined estimate $\tilde{\mathbf{h}}_n$ is obtained based on the updated subspace.

A formal description of the problem is a state-space representation with the state equation,

$$\mathbf{h}_n = \mathbf{h}_{n-1} + \mathbf{b}_n, \tag{3.6}$$

where $\mathbf{b}_n$ is a noise vector referring to the innovation in $\mathbf{h}_n$. The corresponding measurement I/O equation becomes

$$\mathbf{y}_n = \mathbf{C}_n \mathbf{h}_n + \mathbf{v}_n, \tag{3.7}$$

where

$$\mathbf{C}_n = \begin{bmatrix} \mathbf{c}_{n,0}^T \\ \mathbf{c}_{n,1}^T \\ \vdots \\ \mathbf{c}_{n,M-1}^T \end{bmatrix} \tag{3.8}$$

and

$$\mathbf{c}_{n,0} = \begin{bmatrix} x[n] & \mathbf{0}^T & x[n-1] & \cdots & x[n-L+1] & \mathbf{0}^T \end{bmatrix}^T \tag{3.9}$$

The zero vector separating each two consecutive input symbols in $\mathbf{c}_{n,0}$ has length $M-1$. The vectors $\mathbf{c}_{n,i}, i = 1, \ldots, M$ are obtained by shifting the vector $\mathbf{c}_{n,0}$ i times to the right in a circular manner. In this way, the input symbols in the i^{th} raw are multiplied by the tapweights corresponding to $\mathbf{h}_{i,n}$ only. The output vector $\mathbf{y}_n$ is defined as, $\mathbf{y}_n = [y_1[n]\ y_2[n]\ \ldots\ y_M[n]]^T$.

A robust estimator for $\mathbf{h}_n$ is the Kalman filter which is the optimal least mean square estimator [24]. However, the Kalman filter requires $O(p^3)$ number of operation for each new $\mathbf{h}_n$ estimate where $p = ML$. Therefore, for large p, direct use of Kalman filter may be prohibitive.

To reduce the number of operations needed in the Kalman filter, we propose a simpler state-space representation. The idea consists of tracking $\mathbf{h}_{l,n}$, where $l = \lceil \frac{M}{2} \rceil$, and approximating the other channel realizations with three-point linear interpolations of $\mathbf{h}_{l,n}$. This idea leads to the following representation of the all other sub-channel responses as a function of $\mathbf{h}_{l,n}$

$$\mathbf{h}_{i,n} \cong \mathbf{A}_i\ \mathbf{h}_{l,n} \qquad , \qquad i = 1, 2, \ldots, M. \tag{3.10}$$

where $\mathbf{A}_i$'s are the appropriate linear interpolation operators. Thus, we only need to obtain estimates for $\mathbf{h}_{l,n}$ which can be achieved by using Kalman filter in the following reduced state-space representation:

$$\mathbf{h}_{l,n} = \mathbf{h}_{l,n-1} + \mathbf{d}_n, \tag{3.11}$$

$$\mathbf{y}_n = \widetilde{\mathbf{C}}_n \mathbf{h}_{l,n} + \mathbf{v}_n + \eta_n. \tag{3.12}$$

The vector $\mathbf{d}_n$ is a noise vector similar to $\mathbf{b}_n$ with dimension $M \times 1$ instead of $ML \times 1$, and η_n is a white noise vector that is incorporated to the measurement

equation to partially compensate the approximation introduced by the interpolation relation in (3.10). The measurement errors, $\mathbf{v}_n$, and the interpolation errors, η_n are assumed to be independent. The modified measurement matrix $\widetilde{\mathbf{C}}_n$ has the following form:

$$\widetilde{\mathbf{C}}_\mathbf{n} = \begin{bmatrix} \mathbf{x}_n^T \mathbf{A}_1 \\ \\ \mathbf{x}_n^T \mathbf{A}_M \end{bmatrix}. \tag{3.13}$$

In Appendix C, a derivation of equation (3.13) is given, and a simpler way of computing $\widetilde{\mathbf{C}}_n$ that avoids the vector multiplications, is presented.

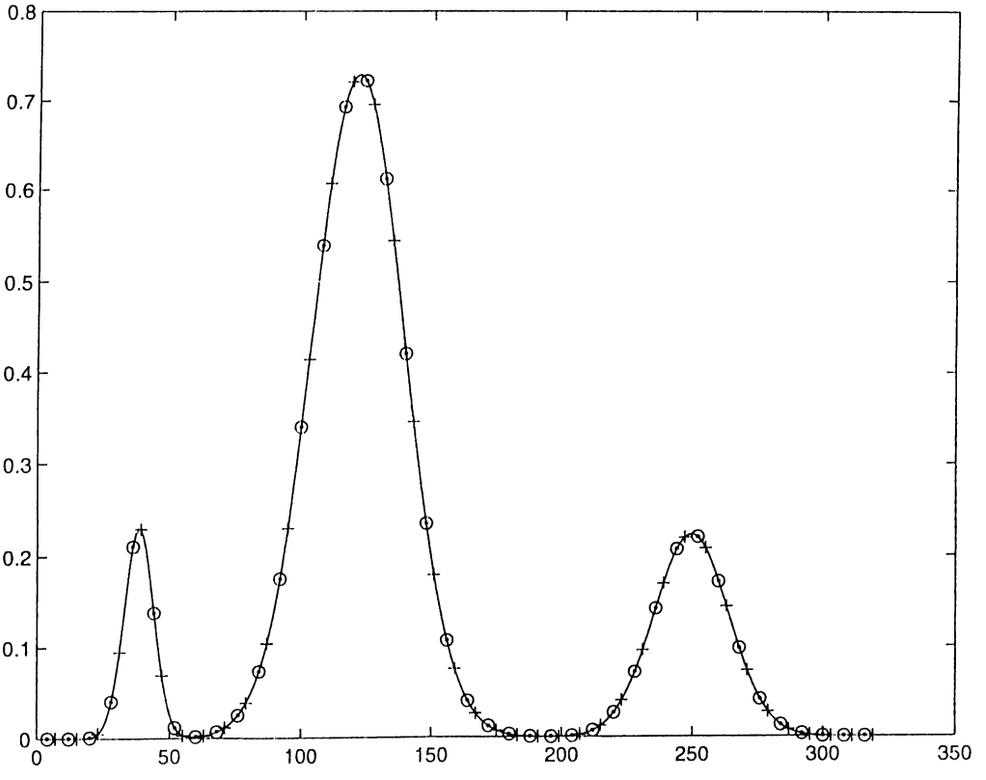

Figure 3.3: Available samples of $\mathbf{h}_{M/2,n}$ are shown with 'o'. The required samples of $\mathbf{h}_{i,n}$, which are shown with '+', can be interpolated by using a low order interpolator such as linear 3-point interpolator.

A further simplification can be obtained by assuming all $\mathbf{A}_i$ in (3.10) as identity operators. In this case, the model will have equation (3.11) as its state equation, whereas the output equation will be

$$\mathbf{y}_n = \widehat{\mathbf{C}}_n \mathbf{h}_{l,n} + \mathbf{v}_n + \eta_n, \tag{3.14}$$

where

$$\widehat{\mathbf{C}}_n = \begin{bmatrix} \mathbf{x}_n^T \\ \\ \mathbf{x}_n^T \end{bmatrix}. \tag{3.15}$$

As shown in Appendix D, in this case the computational complexity of the Kalman filter is $O(L^2)$ which is significantly less than $O((LM)^3)$ of the first state space representation. Further improvements in the processing speed can be achieved through the systolic array implementation of the Kalman filter as described in [25] and [26].

In all the previously described models, the innovation in the state vectors is modeled as an additive white noise. This implies that its covariance matrix

$$\mathbf{Q_d} = \sigma^2 \mathbf{I}_L, \tag{3.16}$$

where σ^2 is the noise variance. In [27], a better formulation of $\mathbf{Q_d}$ is proposed. The latter assumes a strong correlation between the innovation and the estimated state vector. That is,

$$\mathbf{Q_d} = \frac{\sigma^2}{\|\mathbf{h}_{l,n-1}\|^2} \begin{bmatrix} k_1 & 0 & 0 \\ 0 & k_2 & 0 \\ \\ 0 & 0 & k_L \end{bmatrix}. \tag{3.17}$$

where $k_j = \frac{1}{4}\left(|h_{l,n-1}[j-1]|^2 + 2|h_{l,n-1}[j]|^2 + |h_{l,n-1}[j+1]|^2\right)$. In the first formulation, where the state vector represents the oversampled channel impulse

response, $j = 1, 2, \ldots, ML$, the terms $h_{l,n-1}[j-1]$ and $h_{l,n-1}[j+1]$ should be replaced by $h_{l,n-1}[j-M]$ and $h_{l,n-1}[j+M]$, respectively. As it will be shown through the simulations, this method helps to reduce the high frequency noise in the estimates of the channel response.

The Kalman filter, although optimal in the MMSE sense, provides noisy estimates of the channel transfer function. To remove such a noise, we make use of subspace tracking methods. In other words, under the slow time-variation assumption, the matrix $\mathbf{H} = [\mathbf{h}_{n-N} \ \ldots \ \mathbf{h}_n]$ is of low rank. Its column space is the same as the one of $\mathbf{H}\mathbf{H}^T$ which is the channel covariance matrix over a time interval of length $N+1$. Hence by tracking the eigenspace of the channel covariance matrix, based on the estimates given by the Kalman filter, we can remove most of the noise components. In the literature, various algorithms that can accomplish such a task were proposed. Some of these are described in [28], [29], [30], and [31]. The main issues about such algorithms are numerical stability; that is robustness to numerical error buildup, and computational complexity. In our approach, we chose the algorithm LORAF 1 presented in [31], which is a law rank adaptive filter. For our purpose, we are interested in the *Subspace Tracking Section* only. This algorithm tracks the r dominant eigenvalues of the covariance matrix $\mathbf{\Phi} = E\{\hat{\mathbf{h}}_n \ \hat{\mathbf{h}}_n^T\}$, and their corresponding eigenvectors. With an appropriate choice of r, the effective dimension of the subspace, most of the noise components can be eliminated from the signal. That is, if

$$\mathbf{\Phi} = \mathbf{U}\mathbf{D}\mathbf{U}^T \tag{3.18}$$

where $\mathbf{U} = [\mathbf{u}_1 \ldots \mathbf{u}_p]$ is the matrix of eigenvectors and $\mathbf{D} = diag(\lambda_1 \ldots \lambda_p)$ is the diagonal matrix of the eigenvalues. Assuming that A better approximant

to the covariance matrix of the signal of interest only would be;

$$\mathbf{\Phi}_r = \mathbf{U}_r \mathbf{D}_r \mathbf{U}_r^T \tag{3.19}$$

where $\mathbf{U}_r = [\mathbf{u}_1 \dots \mathbf{u}_r]$ and $\mathbf{D}_r = diag(\lambda_1 \dots \lambda_r)$ assuming that λ_i are indexed in decreasing order. Since the channel is time-varying, its covariance matrix should be time-dependent. For each newly available h_n estimate, LORAF 1, performs the following rank-one update of $\mathbf{\Phi}_n$:

$$\mathbf{\Phi}_n = \alpha \mathbf{\Phi}_{n-1} + (1-\alpha)\hat{\mathbf{h}}_n \hat{\mathbf{h}}_n^T. \tag{3.20}$$

To perform a slowly time-varying update of $\mathbf{\Phi}_n$, the exponential forgetting factor α should be chosen close to but less than 1.

Once the eigenvalues and eigenvectors are obtained, the channel impulse response can be re-updated in two alternative ways. The first consists in projecting $\hat{\mathbf{h}}_n$ on to the subspace of eigenvectors.

$$\tilde{\mathbf{h}}_n = \mathbf{U}_n \mathbf{U}_n^T \hat{\mathbf{h}}_n$$

In the second method we estimate $\tilde{\mathbf{h}}_n \tilde{\mathbf{h}}_n^T$ as follows;

$$\mathbf{\Gamma}_n = \tilde{\mathbf{h}}_n \tilde{\mathbf{h}}_n^T = \frac{1}{1-\alpha}(\mathbf{\Phi}_n - \mathbf{\Phi}_{n-1}) \tag{3.21}$$

then extract $\tilde{\mathbf{h}}_n$ from the matrix $\mathbf{\Gamma}_n$ as stated in the respective tabular form of the algorithms.

3.3 Proposed Algorithms and their Simulated Performances

As previously described, the channel identification problem can be formulated in three different state-space representations. Moreover, two different forms of the innovation covariance matrix (3.16) and two distinct ways of re-updating the channel estimate based on the updated subspace are proposed (3.17). The combinations of all these alternatives result in twelve different algorithms.

The algorithms 1, 2, 3, and 4, shown in Table 1, assume the state-space representation described by the state equation (3.6) and the output equation (3.7). In algorithms 1 and 2, the Kalman filter estimate of the channel is projected on the updated subspace, whereas in algorithms 3 and 4, the final estimate is obtained from $\mathbf{\Gamma}_n$ as described in last section of the Table 1. The difference between algorithms 1 and 2, and similarly 3 and 4, is the form of the innovation covariance matrix $\mathbf{Q}_b$.

These four algorithms are compared through simulations over a one K-bit binary sequence. The fractional sampling factor is chosen as $M = 8$. We consider slow and rapid, in the long term sens, time-variations: STV and RTV respectively. Also, the signal to noise ratio is chosen as 23 dB and 10 dB and represented as HSNR and LSNR respectively. Furthermore, to investigate the effect of blind sequence estimation on the performance of the blind channel identification we perform the simulations with both estimated and exact input sequences. In each of these cases, the results are obtained over 10 independent realizations. The error measure is defined as $\epsilon[n] = 20\ log\ \frac{|\tilde{\mathbf{h}}_n - \mathbf{h}_n|}{|\mathbf{h}_n|_{av}}$ and ϵ_{av} is the mean of $\epsilon[n]$ in the steady state.

According to the simulation results shown in Table 3.1, all of the first four algorithms show good performance in the case of known input sequence. As expected, the increase in time-variation or the decrease in the SNR result in a larger channel estimation error. Based on the obtained results, the algorithms 2 and 4 outperform algorithms 1 and 3 respectively. This fact implies that defining the innovation covariance matrix as described in equation (3.17) improves the performance in the channel identification. Similarly, algorithms 2 and 4 outperform algorithms 1 and 3 respectively in the rapid time variation case and the opposite becomes true when the channel exhibits slow variation. This result can be explained by the fact that the projection onto the subspace acquires the algorithm a large memory of the past values and a reluctancy to perform rapid updates. When the input sequence is unknown we notice an increase in the estimation error. The latter is due to some bit uncertainties, in the input vector, that effect the performance of the channel identifier. However, even with less reliable channel estimates, the input sequence estimator is still capable of making the correct decision about the input symbols. Note that, as Tables 3.2 and 3.3 show, only few decision errors occured in the case where the channel is rapidly varying and having a low SNR.

The algorithms 5, 6, 7, and 8, shown in Table 2, assume the state space representation that described by the equations (3.11) and (3.12). The differences between these algorithms are emphasized in Table 2. By examining the Tables 3.4, 3.5 and 3.6, we can notice that algorithms 5 and 6 perform better than algorithms 7 and 8 in the blind case and also when the time variation is slow, whereas the last two algorithms exhibit better performance when the input sequence is known and the channel is rapidly varying.

Algorithms 9, 10, 11, and 12 are stated in Table 3. As previously mentioned, they approximate the sub-channels with a unique one. This assumption leads to a considerable reduction in the number of computations without degrading the performance of the algorithms. In fact, from the Tables 3.7, 3.8, and 3.9 we can see that algorithms 10 and 12 can be considered as the best, in terms of efficiency and computational cost, when the input sequence is known and the channel is rapidly varying, whereas algorithms 9 and 10 outperform all the others in the blind case or under the slow variation condition.

Algorithm 1 Table 1: Algorithms 1, 2, 3, and 4

Initialization:

$$\mathbf{P}_{0,0} = var(\mathbf{h}_0) \;\; ; \;\; \hat{\mathbf{h}}_{0,0} = E\{\mathbf{h}_0\}$$

$$\mathbf{U}_0 = \begin{bmatrix} \mathbf{I} \\ -- \\ \mathbf{0} \end{bmatrix} \;\; ; \;\; \Theta_0 = \mathbf{I} \;\; ; \;\; \mathbf{A}_0 = \mathbf{0} \;\; ; \;\; 0 \leq \alpha \leq 1 \;\; ; \;\; r$$

Input:

$\mathbf{x}_n \rightarrow$ form $\mathbf{C}_n$

Kalman Filter:

For Algorithm 1 & 3 do:

$\mathbf{Q}_b = \sigma^2 I$

For Algorithm 2 & 4 do:

form $\mathbf{Q}_b$, as described in eq. (3.17).

$$\begin{aligned} \mathbf{P}_{n,n-1} &= \mathbf{P}_{n-1,n-1} + \mathbf{Q}_b \\ \mathbf{G}_n &= \mathbf{P}_{n,n-1}\mathbf{C}_n^T(\mathbf{C}_n\mathbf{P}_{n,n-1}\mathbf{C}_n^T + \mathbf{R}_v)^{-1} \\ \mathbf{P}_{n,n} &= (\mathbf{I}_{ML} - \mathbf{G}_n\mathbf{C}_n)\mathbf{P}_{n,n-1} \\ \hat{\mathbf{h}}_{n/n} &= \hat{\mathbf{h}}_{n-1/n-1} + \mathbf{G}_n(\mathbf{y}_n - \mathbf{C}_n\hat{\mathbf{h}}_{n-1/n-1}) \end{aligned}$$

Input:

$\mathbf{z}_n = \hat{\mathbf{h}}_{n/n}$

Subspace Tracking Section:

$$\begin{aligned} \tau_n &= \mathbf{U}_{n-1}^T\,\mathbf{z}_n \\ \mathbf{A}_n &= \alpha\mathbf{A}_{n-1}\Theta_{n-1} + (1-\alpha)\mathbf{z}_n\tau_n^T \\ \mathbf{A}_n &= \mathbf{U}_n\,\mathbf{D}_n \qquad \mathbf{QR}\ \textit{factorization} \\ \Theta_n &= \mathbf{U}_{n-1}^T\,\mathbf{U}_n \end{aligned}$$

Updating the State Vector:

For Algorithm 1 & 2 do:

$$\tilde{\mathbf{h}}_n = \mathbf{U}_n\mathbf{U}_n^T\mathbf{z}_n$$

For Algorithm 3 & 4 do:

$$\begin{aligned} \Phi_n &= \mathbf{A}_n\mathbf{U}_n^T \\ \Gamma_n &= \tfrac{1}{1-\alpha}(\Phi_n - \Phi_{n-1}) \\ (M, j) &= \max(\mathbf{z}_n) \\ \tilde{\mathbf{h}}_n &= \tfrac{1}{M}\Gamma_n(:, j) \end{aligned}$$

Algorithm 2 Table 2: Algorithms 5, 6, 7, and 8

Initialization:

$$\mathbf{P}_{0,0} = var(\mathbf{h}_{l,0}) \;\; ; \;\; \hat{\mathbf{h}}_{0,0} = E\{\mathbf{h}_{l,0}\}$$

$$\mathbf{U}_0 = \begin{bmatrix} \mathbf{I} \\ -- \\ \mathbf{0} \end{bmatrix} \;\; ; \;\; \Theta_0 = \mathbf{I} \;\; ; \;\; \mathbf{A}_0 = \mathbf{0} \;\; ; \;\; 0 \leq \alpha \leq 1 \;\; ; \;\; r$$

Input:

$\mathbf{x}_n \rightarrow$ form $\widetilde{\mathbf{C}}_n$

Kalman Filter:

For Algorithm 5 & 7 do:

$\mathbf{Q}_{\mathrm{d}} = \sigma^2 I$

For Algorithm 6 & 8 do:

form $\mathbf{Q}_{\mathrm{d}}$ as described in eq. (3.17).

$$\begin{aligned}
\mathbf{P}_{n,n-1} &= \mathbf{P}_{n-1,n-1} + \mathbf{Q}_{\mathrm{d}} \\
\mathbf{G}_n &= \mathbf{P}_{n,n-1}\widetilde{\mathbf{C}}_n^T(\widetilde{\mathbf{C}}_n\mathbf{P}_{n,n-1}\widetilde{\mathbf{C}}_n^T + \mathbf{R}_v + \mathbf{R}_\eta)^{-1} \\
\mathbf{P}_{n,n} &= (\mathbf{I}_L - \mathbf{G}_n\widetilde{\mathbf{C}}_n)\mathbf{P}_{n,n-1} \\
\hat{\mathbf{h}}_{n/n} &= \hat{\mathbf{h}}_{n-1/n-1} + \mathbf{G}_n(\mathrm{y}_n - \widetilde{\mathbf{C}}_n\hat{\mathbf{h}}_{n-1/n-1})
\end{aligned}$$

Input:

$\mathbf{z}_n = \hat{\mathbf{h}}_{n/n}$

Subspace Tracking Section:

$$\begin{aligned}
\tau_n &= \mathbf{U}_{n-1}^T\,\mathbf{z}_n \\
\mathbf{A}_n &= \alpha\mathbf{A}_{n-1}\Theta_{n-1} + (1-\alpha)\mathbf{z}_n\tau_n^T \\
\mathbf{A}_n &= \mathbf{U}_n\,\mathbf{D}_n \qquad \mathbf{QR}\ \textit{factorization} \\
\Theta_n &= \mathbf{U}_{n-1}^T\,\mathbf{U}_n
\end{aligned}$$

Updating the State Vector:

For Algorithm 5 & 6 do:

$$\tilde{\mathbf{h}}_{l,n} = \mathbf{U}_n\mathbf{U}_n^T\mathbf{z}_n$$

For Algorithm 7 & 8 do:

$$\begin{aligned}
\Phi_n &= \mathbf{A}_n\mathbf{U}_n^T \\
\Gamma_n &= \tfrac{1}{1-\alpha}(\Phi_n - \Phi_{n-1}) \\
(M, j) &= \max(\mathbf{z}_n) \\
\tilde{\mathbf{h}}_{l,n} &= \tfrac{1}{M}\Gamma_n(:, j)
\end{aligned}$$

Algorithm 3 Table 3:Algorithms 9, 10, 11, and 12

Initialization:

$$\mathbf{P}_{0,0} = var(\mathbf{h}_{l,0}) \;\; ; \;\; \hat{\mathbf{h}}_{0,0} = E\{\mathbf{h}_{l,0}\}$$

$$\mathbf{U}_0 = \begin{bmatrix} \mathbf{I} \\ -- \\ \mathbf{0} \end{bmatrix} \;\; ; \;\; \Theta_0 = \mathbf{I} \;\; ; \;\; \mathbf{A}_0 = \mathbf{0} \;\; ; \;\; 0 \leq \alpha \leq 1 \;\; ; \;\; r$$

Input:

$\mathbf{x}_n \rightarrow$ form $\widehat{\mathbf{C}}_n$

Kalman Filter:

For Algorithm 9 & 11 do:

$\mathbf{Q}_d = \sigma^2 I$

For Algorithm 10 & 12 do:

form $\mathbf{Q}_d$ as described in eq. (3.17).

$$\begin{aligned}
\mathbf{P}_{n,n-1} &= \mathbf{P}_{n-1,n-1} + \mathbf{Q}_d \\
\mathbf{G}_n &= \mathbf{P}_{n,n-1}\widehat{\mathbf{C}}_n^T(\widehat{\mathbf{C}}_n\mathbf{P}_{n,n-1}\widehat{\mathbf{C}}_n^T + \mathbf{R}_v + \mathbf{R}_\eta)^{-1} \\
\mathbf{P}_{n,n} &= (\mathbf{I}_L - \mathbf{G}_n\widehat{\mathbf{C}}_n)\mathbf{P}_{n,n-1} \\
\hat{\mathbf{h}}_{n/n} &= \hat{\mathbf{h}}_{n-1/n-1} + \mathbf{G}_n(\mathbf{y}_n - \widehat{\mathbf{C}}_n\hat{\mathbf{h}}_{n-1/n-1})
\end{aligned}$$

Input:

$\mathbf{z}_n = \hat{\mathbf{h}}_{n/n}$

Subspace Tracking Section:

$$\begin{aligned}
\tau_n &= \mathbf{U}_{n-1}^T\,\mathbf{z}_n \\
\mathbf{A}_n &= \alpha\mathbf{A}_{n-1}\Theta_{n-1} + (1-\alpha)\mathbf{z}_n\tau_n^T \\
\mathbf{A}_n &= \mathbf{U}_n\,\mathbf{D}_n \qquad \mathbf{QR}\ \textit{factorization} \\
\Theta_n &= \mathbf{U}_{n-1}^T\,\mathbf{U}_n
\end{aligned}$$

Updating the State Vector:

For Algorithm 9 & 10 do:

$$\tilde{\mathbf{h}}_{l,n} = \mathbf{U}_n\mathbf{U}_n^T\mathbf{z}_n$$

For Algorithm 11 & 12 do:

$$\begin{aligned}
\Phi_n &= \mathbf{A}_n\mathbf{U}_n^T \\
\Gamma_n &= \tfrac{1}{1-\alpha}(\Phi_n - \Phi_{n-1}) \\
(M, j) &= \max(\mathbf{z}_n) \\
\tilde{\mathbf{h}}_{l,n} &= \tfrac{1}{M}\Gamma_n(:, j)
\end{aligned}$$

	STV/LSNR	STV/HSNR	RTV/LSNR	RTV/HSNR
Algorithm 1	-16.274	-23.120	-14.642	-20.754
Algorithm 2	-18.074	-25.618	-15.076	-22.002
Algorithm 3	-15.934	-22.226	-14.800	-21.106
Algorithm 4	-11.694	-24.534	-15.160	-22.514

Table 3.1: Average logarithmic error (in dB), ϵ_{av}, in the case of known input sequence for algorithms 1, 2, 3, and 4.

	STV/LSNR	STV/HSNR	RTV/LSNR	RTV/HSNR
Algorithm 1	-14.238	-18.132	-6.030	-14.258
Algorithm 2	-14.720	-18.543	-6.490	-18.586
Algorithm 3	-12.431	-16.521	-5.000	-14.160
Algorithm 4	-12.662	-16.012	-6.304	-12.566

Table 3.2: Average logarithmic error (in dB), ϵ_{av}, in the case of unknown input sequence for algorithms 1, 2, 3, and 4

	STV/LSNR	STV/HSNR	RTV/LSNR	RTV/HSNR
Algorithm 1	0	0	0.034	0
Algorithm 2	0	0	0.046	0
Algorithm 3	0	0	0.057	0
Algorithm 4	0	0	0.014	0

Table 3.3: Bit-error rate for algorithms 1, 2, 3, and 4.

	STV/LSNR	STV/HSNR	RTV/LSNR	RTV/HSNR
Algorithm 5	-18.458	-24.296	-16.928	-20.920
Algorithm 6	-20.132	-13.184	-17.396	-21.806
Algorithm 7	-17.774	-22.214	-17.728	-20.050
Algorithm 8	-21.154	-25.156	-17.788	-23.276

Table 3.4: Average logarithmic error (in dB), ϵ_{av}, in the case of known input sequence for algorithms 5, 6, 7, and 8.

	STV/LSNR	STV/HSNR	RTV/LSNR	RTV/HSNR
Algorithm 5	-15.357	-18.327	-11.166	-16.000
Algorithm 6	-15.447	-18.701	-12.574	-14.552
Algorithm 7	-12.725	-17.328	-12.024	-16.246
Algorithm 8	-13.013	-17.107	-10.742	-17.798

Table 3.5: Average logarithmic error (in dB), ϵ_{av}, in the case of un known input sequence for algorithms 5, 6, 7, and 8.

	STV/LSNR	STV/HSNR	RTV/LSNR	RTV/HSNR
Algorithm 5	0	0	0.0087	0
Algorithm 6	0	0	0.0011	0
Algorithm 7	0	0	0.0306	0
Algorithm 8	0	0	0.0077	0

Table 3.6: Bit-error rate for algorithms 5, 6, 7, and 8.

	STV/LSNR	STV/HSNR	RTV/LSNR	RTV/HSNR
Algorithm 9	-18.256	-23.756	-17.134	-21.054
Algorithm 10	-20.280	-24.498	-17.252	-21.594
Algorithm 11	-18.880	-23.390	-17.440	-22.294
Algorithm 12	-19.266	-24.318	-18.668	-22.930

Table 3.7: Average logarithmic error (in dB), ϵ_{av}, in the case of known input sequence for algorithms 9, 10, 11, and 12.

	STV/LSNR	STV/HSNR	RTV/LSNR	RTV/HSNR
Algorithm 9	-14.980	-18.737	-10.830	-17.902
Algorithm 10	-15.872	-18952	-11.334	-19.162
Algorithm 11	-11.378	-17.262	-9.117	-18.332
Algorithm 12	-10.128	-17.023	-9.282	-18.944

Table 3.8: Average logarithmic error (in dB), ϵ_{av}, in the case of un known input sequence for algorithms 9, 10, 11, and 12.

	STV/LSNR	STV/HSNR	RTV/LSNR	RTV/HSNR
Algorithm 9	0	0	0	0
Algorithm 10	0	0	0.0022	0
Algorithm 11	0	0	0.061	0
Algorithm 12	0	0	0.56	0

Table 3.9: Bit-error rate for algorithms 9, 10, 11, and 12.

Chapter 4

SIMULATION

In this chapter, we simulate two already existing approaches and compare them with the results found in the previous chapter. The reference algorithms were proposed in [1] and [2]. The former describes the channel as a linear combination of a subspace basis. At each iteration, an iterative updating of the channel and the subspace is made. The second algorithm is a fast transversal filter.

In the simulations, the channel transfer function is a summation of three shifted Guassian functions, each of which represents a distinct path. The channel order is 40 tapweights and the oversampling factor is 8. As in the previous chapter, the algorithms are simulated over ten different realizations.

When the input sequence is known, the algorithm described in [1] fails whenever the channel is rapidly varying or when the SNR is high. As shown in Table 4.1, the average error is high indicating convergence difficulties. The fast transversal filter described in [2], shows a smaller error. But when compared

	STV/HSNR	STV/LSNR	RTV/HSNR	RTV/LSNR
Algorithm in [1]	-17.4116	-4.841	-4.956	-4.841
Algorithm in [2]	-20.826	-7.610	-7.962	-7.352

Table 4.1: Average logarithmic error for algorithms in [1] and [2]

to our proposed approaches, it has a relatively inferior performance. figure 4.1 illustrates this fact.

In the blind case, both reference algorithms diverge and do not resist input decision errors, wheras our proposed approches are robust to such errors, (see Figure 4.2. These results clearly indicate the superior performance of our algorithms in blind and open-eye conditions.

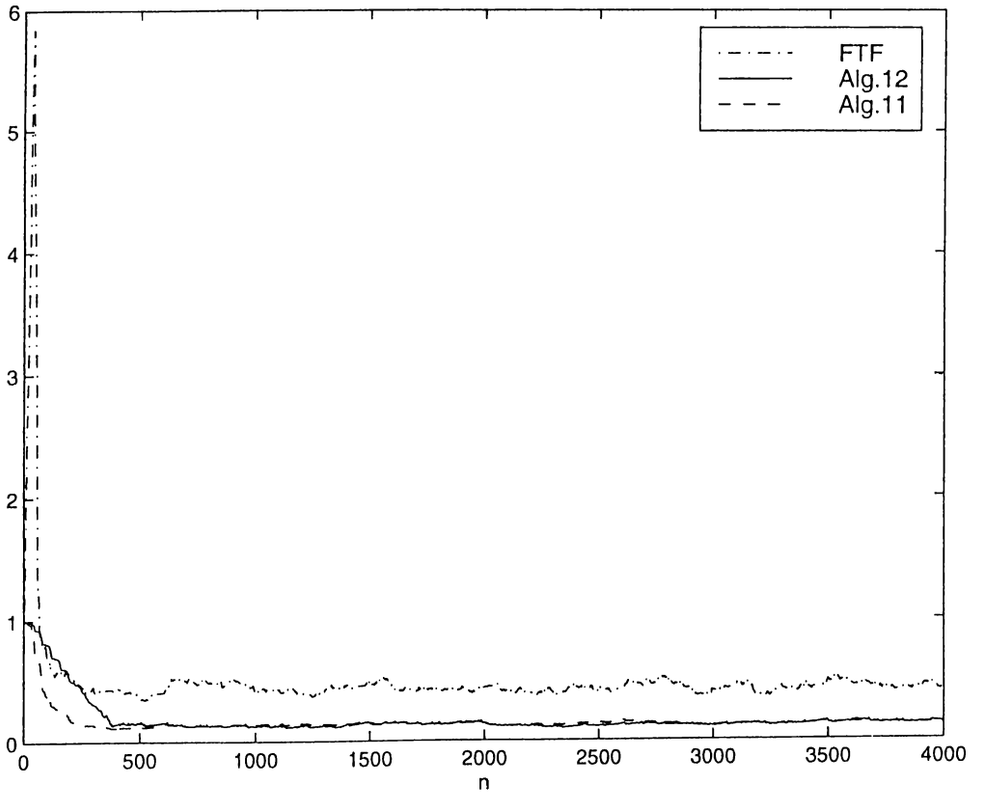

Figure 4.1: Normalized channel estimation error in the open-eye case with rapid time-variation and low SNR.

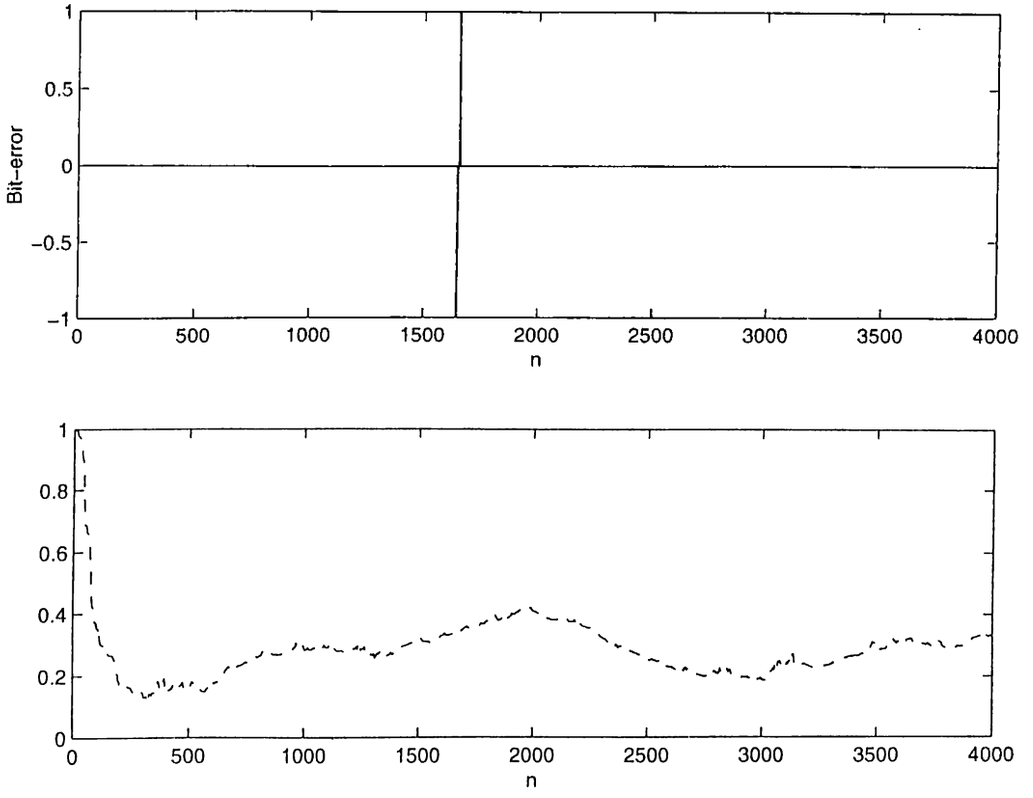

Figure 4.2: Bit-error and normalized channel estimation error for Algorithm 10 in the blind case with rapid time-variation and low SNR.

Chapter 5

CONCLUSIONS

The problems of input sequence estimation and blind channel identification in HF communication are investigated. In this thesis we developed a delayed input sequence estimator. It minimizes the cumulative mean square error over the last received signals. Moreover, different channel identification algorithms are proposed. They make use of a two-step estimation method. First a Kalman filter provides estimate to the channel transfer function. The latter is fed to a subspace tracker that eliminates the estimation noise. In this way we prevent abrupt changes in the channel estimate.

The channel identification algorithms were simulated in both cases of known and unknown input sequence. The estimation errors were found to be small indicating a reliable identification. In the blind case, the channel identifiers, operating together with the input sequence estimator, showed a robust behavior in recovering from an input decision error. When compared to other existing

approaches, the proposed algorithms were superior. In bad tropospheric conditions when the channel is rapidly varying, we are still able to estimate the input sequence reliably even with 10 dB signal-to-noise ratio.

Bibliography

[1] S. Hariharan and A. P. Clark, "HF channel estimation using a fast transversal filtering algorithm," *IEEE Trans. Acoust., Speech, and Signal Process.*, vol. 38, no. 8, pp. 1353–1362, 1990.

[2] A. P. Clark and A. Hariharn, "Efficient estimators for an HF radio link," *IEEE Trans. on Communications*, vol. 38, no. 8, pp. 1173–1180, 1990.

[3] A. P. Clark and S. F. Hau, "Robust adjustment of receiver for distorted digital signals," *Proc. IEEE*, vol. 131, pp. 526–536, 1984.

[4] J. Labat, O. Macchi, and C. Laot, "Adaptive decision feedback equalization: can you skip the training period," *IEEE Trans. on Communications*, vol. 46, no. 7, pp. 921–930, 1998.

[5] J. Thielecke, "A soft-decision state-space equalizer for fir channels," *IEEE Trans. on Communications*, vol. 45, no. 10, pp. 1208–1217, 1997.

[6] L. Tong, G. Xu, and T. Kailath, "Blind identification and equalization based on second-order statistics: a time domain approach," *IEEE Trans. Information Theory*, vol. 40, no. 2, pp. 340–349, 1994.

[7] J. K. Tugnait and U. Gummadavelli, "Blind equalization and channel estimation with partial response input signals," *IEEE Trans. on Communications*, vol. 45, no. 9, pp. 2307–2317, 1997.

[8] C. Becchetti, G. Scarno, and G. Jacovitti, "Fast blind identification for FIR communications channels," *IEEE Trans. Signal Process.*, vol. 45, pp. 2277–2292, 1997.

[9] G. Xu, H. Liu, L. Tong, and T. Kailath, "A least-squares approach to blind channel identification," *IEEE Trans. Signal Process.*, vol. 43, no. 12, pp. 2982–2993, 1995.

[10] L. Tong and S. Perreau, "Multichannel blind identification: from subspace to maximum likelihood methods," *Proc. IEEE*, vol. 86, no. 10, pp. 1951–1968, 1998.

[11] B. Farhang-Boroujeny, "Channel equalization via channel identification: Algorithms and simulation results for rapidly fading hf channels," *IEEE Trans. on Communications*, vol. 44, no. 11, pp. 1409–1412, 1996.

[12] C. C. Watterson, G. G. Ax, L. J. Demmer, and C. H. Johnson, "An ionospheric channel simulator." unpublished ESSA Tech. Memo ERLTM-ITS 198, pp. 1-44, 1969.

[13] C. C. Watterson, J. R. Juroshek, and W. D. Bensema, "Experimental confirmation of an HF channel model," vol. com-18, pp. 792–803, 1970.

[14] A. K. Ozdemir, "Exact blind channel estimator," Master's thesis, Bilkent University, 1998.

[15] J. K. Tugnait, "On blind identification of multipath channel using fractional sampling and second-order cyclostationary statistics," *IEEE Trans. Information Theory*, vol. 41, no. 1, pp. 308–311, 1995.

[16] Z. Ding, "On convergence analysis of fractionally spaced adaptive blind equalizers," *IEEE Trans. on Communications*, vol. COM–35, pp. 877–887, 1987.

[17] A. J. Viterbi, "Error bounds for convolutional codes and an assymptotically optimal decoding algorithm," *IEEE Trans. Information Theory*, vol. 13, pp. 260–269, 1967.

[18] R. Rifkin and B. D. Perry, "Fade measurements on the wide bandwidth HF channel," *Radio Science*, vol. 29, no. 5, pp. 1339–1353, 1994.

[19] R. Rifkin and P. A. Bello, "Representation of propagation mode fading for the midlatitude wide bandwidth HF channel," *Radio Science*, vol. 29, no. 4, pp. 717–722, 1994.

[20] E. Moulines, P. Duhamel, J. F. Cardoso, and S. Mayrargue, "Subspace methods for the blind identification of multichannel FIR filters," *IEEE Trans. Signal Process.*, vol. 43, no. 2, pp. 516–525, 1995.

[21] S. Pei and M. Shih, "Fractionally spaced blind equalization using polyperiodic linear filtering," *IEEE Trans. on Communications*, vol. 46, no. 1, pp. 16–19, 1998.

[22] A. P. Clark and R. Harun, "Assesement of Kalman-filter channel estimators for an HF radio link," *Proc. IEEE*, vol. 133, no. 6, pp. 513–521, 1986.

[23] A. P. Clark and S. Hariharan, "Adaptive channel estimator for an HF radio link," *IEEE Trans. on Communications*, vol. 37, no. 9, pp. 918–928, 1989.

[24] T. Kailath, *lectures on Wiener and Kalman Filtering.* Springer-Verlag, 1981.

[25] M. R. Azimi-Sadjadi, T. Lu, and E. M. Nebot, "Parralel and sequential block Kalman filtering and their implementation using syslotic arrays," *IEEE Trans. Signal Process.*, vol. 39, no. 1, pp. 137–147, 1991.

[26] S. Y. Kung and J.-N. Hwang, "Systolic array design for Kalman filtering," *IEEE Trans. Signal Process.*, vol. 39, no. 1, pp. 171–182, 1991.

[27] F. Arikan and O. Arikan, "Hf kanal dürtü tepkesinin kestirmici icin pratik bir yontem," *IEEE Trans. Signal Process.*, vol. 44, no. 12, pp. 2932–2947, 1996.

[28] G. Mathew, V. U. Reddy, and S. Dasgupta, "Adaptive estimation of eigen-subspaces," *IEEE Trans. Signal Process.*, vol. 43, no. 2, pp. 401–411, 1995.

[29] R. D. Degrot and R. A. Roberts, "Efficient, numerically stabilized rank-one eigenstructure updating," *IEEE Trans. Acoust., Speech, and Signal Process.*, vol. 38, no. 2, pp. 301–316, 1990.

[30] B. Champagne and Q. Liu, "Plane rotation-based evd updating schemes for efficient subspace tracking," *IEEE Trans. Signal Process.*, vol. 46, no. 7, pp. 1886–1900, 1998.

[31] P. Strobach, "Low-rank adaptive filters," *IEEE Trans. Signal Process.*, vol. 44, no. 12, pp. 2932–2947, 1996.

APPENDIX A

Noise Reduction Due to Fractional Sampling

In Figure 2.3, the received continuous signal $y(t)$ has two components

$$y(t) = h(t) \star x(t) + w(t) \star f_{LP}(t)$$

where $h(t)$ and $f_{LP}(t)$ are, respectively, the channel and the LPF transfer functions. The LPF pass-band is matched to the input symbol bandwidth of $[-f_0, f_0]$ with $f_0 = \frac{1}{2T}$. The resulting output noise sequence is $v[n] = v[n\frac{T}{M}]$ where

$$v(t) = w(t) \star f_{LP}(t). \tag{A.1}$$

The power spectrum of $v(t)$ is :

$$S_{vv}(j\Omega) = |H_{LP}(j\Omega)|^2 \quad S_{ww}(j\Omega) = \begin{cases} \sigma_w^2 & \text{if } -2\pi \leq \Omega \leq 2\pi \\ 0 & \text{elsewhere} \end{cases}$$

Since $V(t)$ is oversampled with a factor of M then,

$$S_{vv}(e^{j\omega}) = \begin{cases} \frac{1}{M}\sigma_w^2 & \text{if } -\frac{\pi}{M} \leq \omega \leq \frac{\pi}{M} \\ 0 & \text{elsewhere} \end{cases}$$

Hence the variance of the oversampled noise: $\sigma_v^2 = E\{|v[n]|^2\} = S_{vv}(e^{j0}) = \frac{\sigma_w^2}{M}$ which is M times smaller then the noise variance in the case of symbol rate sampling is utilized.

The low-pass filtering operation in Eq. (A.1) can be written in the following discrete-time operator form:

$$\mathbf{v}_n = \begin{bmatrix} \mathbf{f}_1^T \\ \vdots \\ \mathbf{f}_M^T \end{bmatrix} \tilde{\mathbf{w}}_n = F\ \tilde{\mathbf{w}}_n \tag{A.2}$$

where f_i are the appropriately delayed and sampled impulse response of the low-pass filter. Now, the autocorrelation matrix of the fractional noise samples can be written as :

$$\begin{aligned} \mathbf{R}_v &= E\{\mathbf{v}_n \mathbf{v}_n^T\} \\ &= E\{F\tilde{\mathbf{w}}_n \tilde{\mathbf{w}}_n^T F^T\} \\ &= FE\{\tilde{\mathbf{w}}_n \tilde{\mathbf{w}}_n^T\} F^T \end{aligned}$$

Since $\tilde{\mathbf{w}}_n$ is white

$$\mathbf{R}_v = \sigma_w^2 F F^T \tag{A.3}$$

Since the impulse response of the LPF is known, this final form of the autocorrelation matrix of $\mathbf{v}$ can be precomputed for computational savings.

APPENDIX B

Convergence of Input Sequence Estimator

Consider the cumulative error term

$$E_n = \frac{1}{M(K+1)} \sum_{i=1}^{M} \sum_{k=1}^{K} (y_i[n-k] - \tilde{\mathbf{h}}_{i,n-1}^T \mathbf{x}_{n-k}^q)^2 \tag{B.1}$$

Using Eq. (3.1), we can write

$$E_n = \frac{1}{M(K+1)} \sum_{i=1}^{M} \sum_{k=1}^{K} (v_i[n-k] + \mathbf{h}_{i,n-k}^T \mathbf{x}_{n-k} - \tilde{\mathbf{h}}_{i,n-1}^T \mathbf{x}_{n-k}^q)^2 \tag{B.2}$$

Since the noise is independent of the input sequence and the channel transfer function,

$$E_n \cong \frac{1}{M(K+1)} \sum_{i=1}^{M} \sum_{k=1}^{K} (v_i[n-k])^2 + (\mathbf{h}_{i,n-k}^T \mathbf{x}_{n-k} - \tilde{\mathbf{h}}_{i,n-1}^T \mathbf{x}_{n-k}^q)^2 \tag{B.3}$$

For $M(K+1)$ sufficiently large,

$$\frac{1}{M(K+1)} \sum_{i=1}^{M} \sum_{k=1}^{K} (v_i[n-k])^2 - \sigma_v^2 \simeq 0 \tag{B.4}$$

Let,

$$\mathbf{x}_{n-k}^{q} = \mathbf{x}_{n-k} + \varepsilon_{n-k}^{q}. \tag{B.5}$$

Note that the vectors ε_{n-k}^{q} may have non-zero values only for first $K+1$ entries. In case $\tilde{\mathbf{h}}_{i,n-1} \cong \mathbf{h}_{i,n-k}$, $k = 0, \ldots, K$ and $i = 1, \ldots, M$,

$$\mathbf{h}_{i,n-k}^{T}\mathbf{x}_{n-k} - \tilde{\mathbf{h}}_{i,n-1}^{T}\mathbf{x}_{n-k}^{q} \cong \mathbf{h}_{i,n-k}^{T}\, \varepsilon_{n-k}^{q}. \tag{B.6}$$

If K is chosen as described in section 3.1.1, then the term $\mathbf{h}_{i,n-k}^{T}\, \varepsilon_{n-k}^{q}$ will take large values for $\mathbf{x}_{n}^{q}$ having errors in the entries with indices larger than $K-1$. Hence such vectors won't be chosen by the input sequence estimator, which means no decision error will occur.

APPENDIX C

Proof of Equation 3.12

The output equation for fractionally spaced channel can be written as:

$$\mathbf{y}_n = \mathbf{H}_n \mathbf{x}_n + \mathbf{v}_n \tag{C.1}$$

where

$$\mathbf{H}_n = \begin{bmatrix} \mathbf{h}_{1,n}^T \\ \vdots \\ \mathbf{h}_{M,n}^T \end{bmatrix}. \tag{C.2}$$

If we approximate the different fractional channels by linear interpolations of $\mathbf{h}_{l,n}$ where $l = \frac{M}{2}$, we get

$$\mathbf{y}_n = \begin{bmatrix} (\mathbf{A}_1 \mathbf{h}_{l,n})^T \\ \\ (\mathbf{A}_M \mathbf{h}_{l,n})^T \end{bmatrix} \mathbf{x}_n + \mathbf{v}_n + \eta_n$$

For $i = 1, \ldots, l$

$$\mathbf{A}_i = \begin{bmatrix} \frac{l+i}{M} & & \\ \frac{l-i}{M} & \mathbf{0} & \\ & & \\ \mathbf{0} & \frac{l-i}{M} & \frac{l+i}{M} \end{bmatrix}$$

and for $i = l, \ldots, M$

$$\mathbf{A}_i = \begin{bmatrix} \frac{l-i}{M} & \frac{l+i}{M} & \mathbf{0} \\ & & \\ & \mathbf{0} & \frac{l+i}{M} \\ & & \frac{l-i}{M} \end{bmatrix}$$

Note that $\mathbf{A}_l = \mathbf{I}$. The corresponding output equation can be written as

$$\mathbf{y}_n = \begin{bmatrix} \mathbf{h}_{l,n}^T \mathbf{A}_1^T \mathbf{x}_n \\ \vdots \\ \mathbf{h}_{l,n}^T \mathbf{A}_M^T \mathbf{x}_n \end{bmatrix} + \mathbf{v}_n + \eta_n = \begin{bmatrix} (\mathbf{A}_1^T \mathbf{x}_n)^T \\ \\ (\mathbf{A}_M^T \mathbf{x}_n)^T \end{bmatrix} \mathbf{h}_{n,l} + \mathbf{v}_n + \eta_n. \qquad \text{(C.3)}$$

Now, by using Eq. (C.3), we obtain Eq. (3.12):

$$\mathbf{y}_n = \tilde{\mathbf{C}}_n \mathbf{h}_{n,l} + \mathbf{v}_n + \eta_n \qquad \text{(C.4)}$$

Using the special forms of A_i's we get

$$\mathbf{x}_n^T \mathbf{A}_i = \frac{l+i}{M} \mathbf{x}_n^T + \frac{l-i}{M} [0 \; x[n] \ldots x[n-L+2]]; \quad i \leq l$$

and

$$\mathbf{x}_n^T \mathbf{A}_i = \frac{l-i}{M} \mathbf{x}_n^T + \frac{l+i}{M} [x[n-1] \ldots x[n-L+1] \; 0]; \quad i \geq l$$

thus the rows of $\tilde{\mathbf{C}}_n$ are linear combinations of $\mathbf{x}_n^T$, $[0 \; x[n] \ldots x[n-L+2]]$ and $[x[n-1] \ldots x[n-L+1] \; 0]$.

APPENDIX D

Reduction in the Computational Cost of the Kalman Filter

Given the special form of the matrix $\widehat{\mathbf{C}}_n$ described in Eq. (3.15), we can write

$$\mathbf{P}_{n,n-1}\widehat{\mathbf{C}}_n^T = \begin{bmatrix} \mathbf{P}_{n,n-1}\mathbf{x}_n & \mathbf{P}_{n,n-1}\mathbf{x}_n \end{bmatrix} \tag{D.1}$$

and

$$\widehat{\mathbf{C}}_n\mathbf{P}_{n,n-1}\widehat{\mathbf{C}}_n^T = \mathbf{x}_n^T\mathbf{P}_{n,n-1}\mathbf{x}_n \begin{bmatrix} 1 & 1 \\ \vdots & \\ 1 & 1 \end{bmatrix}. \tag{D.2}$$

Since these matrix multiplications are involved in the computation of the Kalman gain matrix G_n, we reduce the number of multiplications to L^2 instead of L^3.

Let $\beta = \mathbf{x}_n^T\mathbf{P}_{n,n-1}\mathbf{x}_n$ and $\mathbf{a} = [\beta \ldots \beta]^T$. Then the matrix inversion used in computing G_n can be done as follows:

$$(\widehat{\mathbf{C}}_n\mathbf{P}_{n,n-1}\widehat{\mathbf{C}}_n^T + \mathbf{R})^{-1} = \mathbf{R}^{-1} - \frac{\mathbf{R}^{-1}\mathbf{a}\mathbf{a}^T\mathbf{R}^{-1}}{1 + \mathbf{a}^T\mathbf{R}^{-1}\mathbf{a}} \tag{D.3}$$

Since $\mathbf{R}^{-1}$ is supposed to be known, then the last equation involves only K^2 multiplications. Therefore, the computational cost of the Kalman filter becomes $O(L^2)$.