A channel subspace post-filtering approach to adaptive equalization

by

Rajesh Rao Nadakuditi
B.S. Electrical Engineering, Lafayette College, 1999

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfilment of the requirements for the degree of
Master of Science in Electrical Engineering and Computer Science
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
and the
WOODS HOLE OCEANOGRAPHIC INSTITUTION
February 2002.

The author hereby grants to MIT and WHOI permission to reproduce and distribute publicly paper and electronic copies of this thesis document in whole or in part

Author .....
DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE,
MASSACHUSETTS INSTITUTE OF TECHNOLOGY,
AND THE JOINT PROGRAM IN OCEANOGRAPHY AND OCEANOGRAPHIC ENGINEERING,
MASSACHUSETTS INSTITUTE OF TECHNOLOGY/
WOODS HOLE OCEANOGRAPHIC INSTITUTION
SEPTEMBER 25, 2001

Certified by .....
James C. Preisig
Associate Scientist, Woods Hole Oceanographic Institution
Thesis Supervisor

Accepted by .....
Michael S. Triantafyllou
Chair, Joint Committee for Applied Ocean Science and Engineering,
Massachusetts Institute of Technology/
Woods Hole Oceanographic Institution

Accepted by .....
Arthur C. Smith,
Chairman, Department Committee on Graduate Students

A channel subspace post-filtering approach to adaptive equalization

by

Rajesh Rao Nadakuditi

Submitted to the Electrical Engineering and Computer Science
on September 25, 2001, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

One of the major problems in wireless communications is compensating for the time-varying intersymbol interference (ISI) due to multipath. Underwater acoustic communications is one such type of wireless communications in which the channel is highly dynamic and the amount of ISI due to multipath is relatively large. In the underwater acoustic channel, associated with each of the deterministic propagation paths are macro-multipath fluctuations which depend on large scale environmental features and geometry, and micro-multipath fluctuations which are dependent on small scale environmental inhomogeneities. For arrivals which are unsaturated or partially saturated, the fluctuations in ISI are dominated by the macro-multipath fluctuations resulting in correlated fluctuations between different taps of the sampled channel impulse response. Traditional recursive least squares (RLS) algorithms used for adapting channel equalizers do not exploit this structure. A channel subspace post-filtering algorithm that treats the least squares channel estimate as a noisy time series and exploits the channel correlation structure to reduce the channel estimation error is presented. The improvement in performance of the algorithm with respect to traditional least squares algorithms is predicted theoretically, and demonstrated using both simulation and experimental data. An adaptive equalizer structure that explicitly uses this improved estimate of the channel impulse response is discussed. The improvement in performance of such an equalizer due to the use of the post-filtered estimate is also predicted theoretically, and demonstrated using both simulation and experimental data.

Acknowledgments

I would like to thank my advisor Dr. James Preisig for his encouragement, advise, and unwavering support. I thank all my friends for their encouragement and support. Finally, I would like to thank my family for their unconditional love. This research was supported by an ONR Graduate Traineeship Award Grant #N00014-00-10049. I am appreciative of the support that the WHOI Education office has provided.

Contents

List of Figures

Chapter 1

Introduction

In the past decade or so, there has been a tremendous increase in research on and development of underwater acoustic (UWA) communication systems. Most of this growing interest can be attributed to the diversification in the number of applications that require such a technology. While the primary catalyst for new technology, earlier on, used to be almost exclusively military in nature, commercial and scientific applications recently have broadened the scope and the need for such research. Much of the attention has been paid towards applications such as pollution monitoring in environmental systems and remote monitoring and control in the off-shore oil industry At the same time however, major technological strides in this field over the past few years have enabled teams of scientists to conjure up ambitious visions that could fundamentally advance our understanding in the several fields of oceanography. All of these visions, a recent one of which is the autonomous oceanographic sampling network (AOSN) [8], rely on underwater acoustic communication systems to make scientific data available to researchers in a manner that is unprecedented in both scope and scale. The increasing need for UWA communications capability at higher data rates and in more dynamic environments adds significant challenge to the research community.

Besides these important scientific and military applications involving underwater communication, the UWA channel presents particularly challenging problems for signal processing algorithm design. Many of the impairments encountered on wireless RF

channels are experienced at even more severe levels in the underwater acoustic channel. The underwater acoustic channel provides a useful context in which to explore and develop some of the most aggressive and signal-processing-intensive emerging techniques [25]. The channel subspace filtering approach to adaptive equalization, presented in this thesis, is one such aggressive signal processing technique that is particularly useful in the context of UWA communication but whose development and applicability is more universal in nature.

1.1 Prior work

Before the late 1970's, there were a few published designs of acoustic modems. The analog systems that were developed were essentially sophisticated loudspeakers that had no capability for mitigating the distortion introduced by the underwater acoustic channel. Paralleling the developments applied to severely fading radio frequency atmospheric channels, the next generation of systems employed frequency-shift-keyed (FSK) modulation of digitally encoded data [2, 18]. The use of digital techniques enabled the use of explicit error-correction algorithms to increase reliability of transmissions and permitted some level of compensation for the distortion introduced by the channel. On the heels of improved processor technology, variants of the FSK algorithm that exploited increased demodulation speeds were implemented. Despite their reliability, researchers started considering other modulation methods because such incoherent systems did not make effective use of the available limited bandwidth. Hence, they were ill-suited for high-data-rate applications such as image transmission or multiuser networks except at short ranges. This led to research in systems that employed coherent modulation schemes.

Work in the early 1990's resulted in the development of coherent systems that successfully operated in the horizontal ocean channel. The seminal work [32, 33] succeeded due to the use of a demodulation algorithm that coupled a decision feedback adaptive equalizer with a second-order phase-locked loop. The new generation of UWA communication systems, based on the principles of phase-coherent detection

techniques, was thus capable of achieving raw data throughputs that were an order of magnitude higher than those of systems that were based on non coherent detection methods. With the feasibility of bandwidth-efficient phase-coherent UWA communications established, current research is advancing in a number of directions, focusing on the development of more sophisticated processing algorithms which will enable robust and reliable data transmission in varying system configurations and channel conditions. Much of this research has centered around the development of effective receivers that are able to track and compensate for rapid changes in the underwater acoustic channel. In the subsequent sections, the issues and obstacles regarding receiver design are discussed in conjunction with an overview of relevant characteristics of the underwater acoustic channel that impact the selection and design of such receiver algorithms. This is then followed by a discussion on the use of generalized reduced complexity techniques, such as sparsing and subspace methods, in developing what shall be referred to as the channel subspace paradigm.

1.2 Receiver Design: Issues and Challenges

From the communications perspective, the UWA channel poses many challenges to the implementation of reliable, high-rate communications. Approaches to system design vary according to the techniques used for overcoming the effects of multipath and phase variations. Specifically, these techniques may be classified according to 1) the signal design i.e., the choice of modulation method, and 2) the demodulation algorithm. Signal design, particularly coherent modulation/detection method, is essential to achieving high-bandwidth efficiency. However, reliable data communications can be ensured only if the transmitter/receiver structure is designed in a manner that can overcome the distortions due to the channel. Since the demonstration of the feasibility of phase-coherent UWA communications, one of the most important areas of research has been the design of effective receiver structures. There has certainly been a lot of research in this area in the context of wireless microwave communications. Despite outward similarities, there are certain characteristics of the UWA channel

that makes the design of receivers for use in the UWA channel significantly more difficult. An overview of these issues is presented next and their impact on receiver design is discussed.

1.2.1 Characteristics of the UWA channel: Time-Varying Multipath

A fundamental limitation of the UWA channel is the amount of available bandwidth, which is severely limited by absorption loss that increases with both frequency and range [4, 7]. For example, a long-range system operating over several tens of kilometers is limited to few kHz of bandwidth; a medium-range system operating over several kilometers has a bandwidth on the order of ten kHz, while a short-range system, operating over several tens of meters may have available more than a hundred kHz. Within this limited bandwidth, the transmitted signal is subject to multipath propagation through the channel. The presence and extent of this multipath is a characteristic of the particular physical environment and is dependent on various parameters [4], many of which are time-varying in nature. Some of these parameters include the spatially varying sound speed, placement of the receiver and transmitter, and the bathymetric profile of the sea floor between them, among others. The link configuration, primarily designated as vertical or horizontal, affects the multipath as well. Generally, vertical channels exhibit much less multipath than horizontal channels because the spatial scale of the inhomogeneities is typically much greater in the horizontal than in the vertical and the primary source of boundary reflections are the bottom and the sea surface which are predominantly horizontal surfaces.

The multipath, as shown in figure 1-1 leads to intersymbol interference (ISI) whereby the received signal for any particular transmitted symbol spreads over multiple symbol intervals. While the multipath spread in radio channels is on the order of several symbol intervals, it can be on the order of several tens or hundreds of symbol intervals in the UWA channel. The ISI in the received signal depends on the duration of the transmitted pulse as well as the multipath. The mechanism of multipath

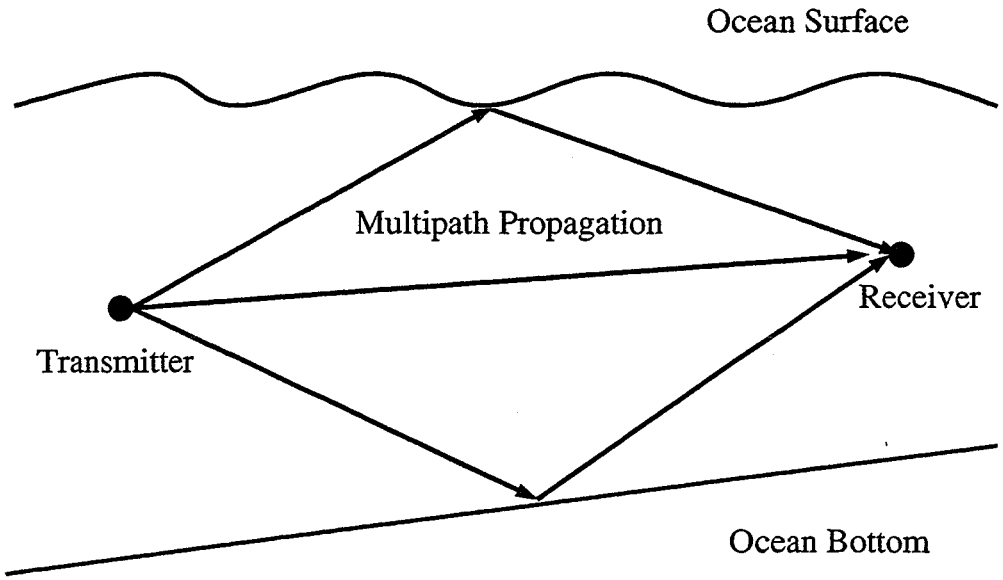

Figure 1-1: Multipath propagation in the UWA channel

formation depends on the channel geometry and also on the frequency of transmitted signals. Understanding of these mechanisms is based on the theory and models of sound propagation. Two principal mechanisms of multipath formation are reflection at the boundaries (bottom, surface and any objects in the water), and ray refraction. Associated with each of the deterministic propagation paths are macro-multipath fluctuations which depend on large scale environmental features and geometry, and micro-multipath fluctuations which are dependent on small scale environmental inhomogeneities [12, 13]. In the context of communication channel modeling, a sampled time-varying multipath channel impulse response is modeled as a tapped delay line and the tap gains are modeled as stochastic processes with certain distributions and power spectral densities. Multipath structures associated with mobile radio channels are often modeled as above, where the tap gains are derived from a Wide Sense Stationary Uncorrelated Scattering (WSSUS) process [3]. For arrivals which are unsaturated or partially saturated, the fluctuations in the complex tap amplitude can be correlated between different taps of the sampled channel impulse response. This is a very important distinction, and one that shall be utilized in our development of the channel subspace approach. Traditional recursive least squares (RLS) algorithms used for adapting channel equalizers do not exploit this structure; instead they assume

that the tap fluctuations are independent. The following section describes some of these algorithms and discusses the implications of not exploiting the low rank nature of the channel subspace.

1.2.2 Receiver Design: Current Techniques and Obstacles

From the discussion above it is obvious that to achieve higher data rates, the more sophisticated systems based on phase-coherent signaling methods must allow for the ISI in the received signal. Throughout this thesis, the term impulse response will refer to both time-invariant and various forms [3] of the time-varying impulse response. Attempts have been made in the past to design communication systems which use array processing to explicitly suppress multipath [14, 16]. These designs have relied on specific propagation models, on the basis of ray theory and the theory of normal modes [4], for predicting the multipath configuration. As can be imagined, the performance of such systems is highly dependent on the accuracy of the model and the available environmental data and is hence quite restrictive in scope and nature. Since it is not possible to eliminate ISI in the received signal, the focus of research shifts towards the design of receivers with algorithms that are able to track changes in the ISI induced by the channel, without compromising on the signaling rate, and are able to suitably compensate for these changes so that a robust, high-reliability communications link is maintained [18].

There has been long history of modifying algorithms, which were originally employed with great success in mobile radio communications, for use in the UWA channel. A broad survey of such adaptive methods is presented in [28] . Once the channel impulse response is modeled as a tapped delay line, as described earlier, the objective of such adaptive algorithms is to track the variations in the tap gains and to appropriately compensate for them. Often, in the context of mobile radio communications, optimal windows are designed that take into account apriori statistics of the Doppler power spectrum, using Jake's model [17] as in [27]. However, equivalent statistics are not available for the UWA channel, so such mathematically elegant optimal windows cannot be determined or applied. Similarly, many of the powerful

algorithms described in [28], such as the zero-forcing equalizer, that are applicable to slowly time-varying systems with well characterized propagation models and reliable apriori statistics are not as useful in the context of adaptive equalization in the UWA channel.

Algorithms such as the least-mean-square (LMS) [37], and the recursive least squares (RLS) and its variants [15] have been incorporated into receivers for UWA communications because of their effectiveness, even without making any statistical assumption, and relatively good tracking performance. Adaptive equalizers using a decision feedback type structure [22] have been shown to be successful when used in receivers for UWA communications particularly when estimates of the channel impulse response are used to adapt the taps of the equalizer [31, 34]. Despite the significant progress made in past decade or so in designing adaptive algorithms for UWA communication systems, there are still many underwater acoustic channels over which such algorithms are unable to establish reliable communications. This inability of conventional algorithms to track such channels is sometimes linked to the trade-off between the number of degrees of freedom incorporated into the channel model,traditionally viewed as the number of taps, and the rate of fluctuations which can be tracked.

1.2.3 Receiver Design: Complexity Reduction

Conventional algorithms, as mentioned earlier, implicitly model the tap gains of the channel impulse response as being derived from a WSSUS Gaussian process. Hence each tap gain is adapted independently because the WSSUS Gaussian assumption implies that the different tap gains are uncorrelated. The degrees of freedom in the adaptation algorithm is thus indirectly constrained to be equal to the number of taps used to model the channel impulse response. Under such a framework, rapid convergence is guaranteed if the channel coherence time, measured in samples, is about thrice the number of taps to be adapted [29]. For radio channels, which typically use about 10 to 15 taps to model their channel impulse, such algorithms assume a channel coherence time of about 30 to 50 symbols which is reasonable under realistic oper-

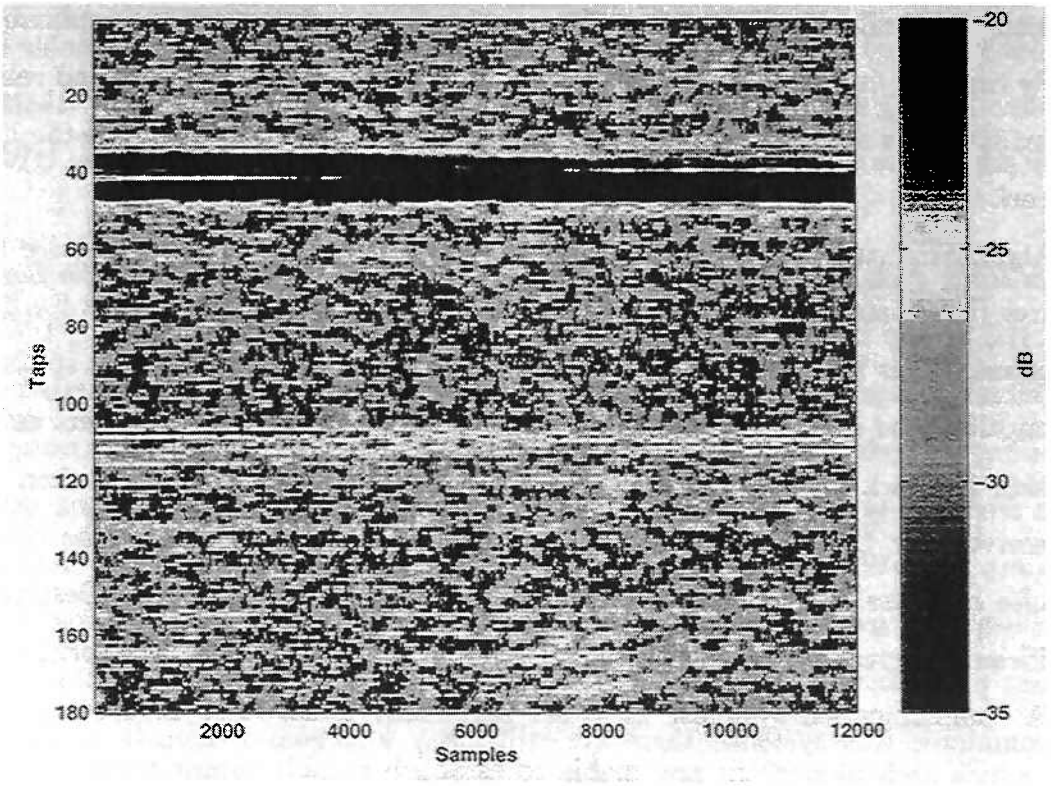

Figure 1-2: Time varying magnitude of channel impulse response

ating conditions given their high signaling rate. However, for UWA channels, about 100 to 150 taps are often required to model the channel impulse response adequately, resulting in an implicit assumed channel coherence time of 300 to 500 symbols. Since UWA communications systems have a much lower signaling rate, due to their limited bandwidth as discussed earlier, in a typical mid-range system transmitting at 5000 symbols per second, rapid convergence is assured only if the channel coherence time is greater than 60 to 100 milliseconds. This is an unreasonable assumption in many rapidly varying UWA channels under realistic operating conditions. In conventional algorithms, this trade-off between incorporating sufficient number of degrees of freedom, equal to the number of taps under the uncorrelated tap gains assumption, to adequately model the channel impulse response and the rate of convergence of these tap-gain weight vectors presents a significant obstacle in the design of high-performance adaptive algorithms. This trade-off can be offset, to varying degrees, by the use of reduced complexity techniques.

Reduced complexity methods exploit certain characteristics of the channel impulse response so that the number of parameters, i.e. the number of degrees of freedom, that need to be adapted independently is decreased. Such a reduced-complexity DFE, for use in mobile radio channels, that utilizes the sparseness of the mobile radio channel impulse response is described in [11]. Since channel impulse responses associated with a multipath structure are often sparse in nature, one in which many of the tap gains are zero or close to zero, only the nonzero tap weights are adapted independently. A threshold detector type test is used to determine whether a particular tap weight is classified as a "zero" weight or a "non-zero" weight. This technique is particularly effective when, under the WSSUS assumption [3], the different tap weights are uncorrelated. The performance improvement comes from the fact that since only the non-zero weights of the unknown system are adapted, the variance is not increased for each of the other uncorrelated zero weights of the sparse system [6]. Thus, the misadjustment error due to gradient noise can be greatly reduced depending on the number of tap gains that are classified as "zero" weight. Such techniques have been very successful in mobile radio channels. The system described in [19] uses a direct adaptation equalizer and uses such a sparsing technique to determine the equalizer taps. A sparse channel response as described in [6, 19] exists when many diagonal elements of the channel impulse correlation matrix are close to zero. A generalization of this concept is the 'low rank' channel where all diagonal elements of the channel correlation matrix have significant magnitude but many eigenvalues of the matrix are close to zero. Even though the technique of sparse equalization is quite powerful, it does not fully exploit the characteristics of the UWA channel when, unlike mobile radio channels, the tap gains in the channel impulse response are correlated. Hence, even if a particular tap gain is designated "zero" weight, performance improvement due to lowered variance of the "zero" weight is offset by the increased bias because of the correlation between the different weights.

A generalized sparsing technique that exploits this correlation between the tap gains of the channel impulse response, should be able to overcome this previously irreconcilable trade-off ,in several realistic UWA channels, between the bias and vari-

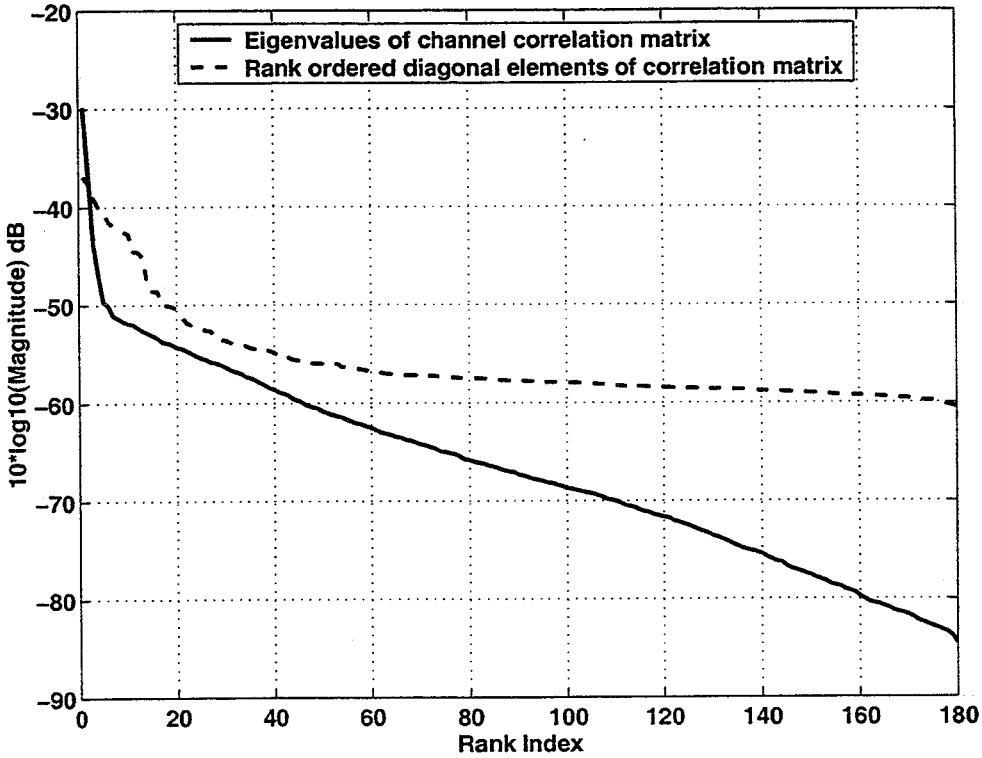

Figure 1-3: Sparse nature of the eigenvalues and ordered diagonal elements of the channel correlation matrix

ance of misadjustment error when sparsing methods are used. The structure of the channel impulse response correlation matrix would have to be exploited in such a generalized sparsing algorithm. Figure 1-2 shows a sample channel estimate time series showing its time varying nature.

Figure 1-3 compares ordered eigenvalues of the channel impulse response correlation matrix with the ordered diagonal elements of the correlation matrix. A sliding window of size 400 was used to compute the channel estimates and the channel correlation matrix was computed using 12000 channel estimates. The time constant for the process corresponded to about 1600 samples which implies that the channel correlation matrix was estimated over a time frame of about 7.5 time constants. As seen in figure 1-3 the eigenvalues of the channel correlation matrix are much sparser than the correlated taps of the channel impulse response.

Conventional sparsing algorithms, with the implicit assumption that the different tap gains are uncorrelated, use the traditional Euclidean multidimensional bases in

identifying "zero" and "non-zero" weights by their relative position on the tapped delay line. A generalized sparsing algorithm uses the eigenvectors of the channel impulse response correlation matrix as the bases and identifies the "zero" and "non-zero" weights by appropriately deciding whether the corresponding bases vector belongs to the nullspace of the channel or not. Essentially, such an algorithm would have to appropriately decide the rank of the channel subspace, equal to the number of "non-zero" weights. Since these generalized weights are now uncorrelated, it is conceivable that analogous techniques that exploited this "sparse" structure would be able to achieve an improvement in performance. It is trivial to observe that the situation with uncorrelated tap gains results in the Euclidean bases as being the equivalent bases and the tap gains as being the corresponding weights. This forms the basis of the channel subspace approach, which, as a generalization to prior channel sparsing methods, attempts to exploit the correlation between the different tap weights of the time varying channel impulse response so that only the reduced rank, sparsed in a generalized sense, subspace occupied by the channel is tracked. Naturally, this improvement in performance, analogous to previous channel sparsing methods, depends on the extent and the efficiency with which the channel can be "sparsed" in the generalized sense, or have its rank reduced. A cursory examination of the ill-conditioned eigenvalue spread of the channel impulse response correlation matrix shown in Figure 1-3 makes it conceivable to believe that if properly exploited, the channel subspace approach could indeed result in drastic rank reduction and a corresponding improvement in performance. Although this interpretation of tracking the reduced rank channel subspace in the context of UWA communication is unique, there is an extensive body of literature dealing with adaptive signal processing techniques using reduced rank subspace methods. An understanding of the assumptions and limitations of such methods is needed so that practical algorithms based on channel-subspace approach can be developed and analyzed.

1.3 Prior work on subspace methods

Subspace methods were originally developed as a way of reducing the mean squared error of estimators. Such methods inherently rely on shaping the rank of the subspace of the relevant signal/parameter that is being estimated/detected. As a result, the variance of the estimator is reduced at the expense of introducing model bias and hopefully, the net result is reduced mean-squared error (MMSE). Adaptive algorithms that rely on subspace techniques are very powerful tools, used extensively in aspects of digital communications. One of the first applications for such techniques was proposed by Tong, Xu and Kailath [36]. Their proposal relied on oversampling, compared to the symbol rate, of the received signal. A subspace decomposition of the signal correlation matrix then separated the signal subspace and the noise subspace resulting in rapid convergence and better performance of the equalization algorithms. Different techniques were subsequently developed to help adaptively identify the dimension of the signal subspace. Once effective discrimination of the signal and noise subspaces was achieved, the channel identification problem became analogous to a scenario where the channel was estimated within the subspace occupied by the signal, while rejecting the subspace occupied by the noise. The orthogonality of the two subspaces guaranteed that channel coefficients obtained by operating on these subspaces were uncorrelated and independent. A reduction in error was obtained once the true dimension of the signal subspace was estimated. Concurrently, extensive research resulted in generalized algorithms for subspace based blind channel identification [23, 1, 20, 9], equalization [35], data adaptive rank shaping [30], and linear prediction [1] that exploited the reduced rank nature of the signal subspace. These generalized methods are conceptually quite comparable to the types of algorithms that need to be developed to demonstrate the relevance and need for using a channel subspace based approach to solve equivalent problems in UWA communications. These algorithms generally exploit the reduced dimension of the signal subspace. This reduced dimension either arises in the context of a source separation problem where the dimension of the subspace is equal to the number of sources, or in the context

of highly oversampled data. Implicitly, these techniques make the assumption that the channel itself is full rank or equivalently that the channel correlation matrix does not have eigenvalues close to or equal to zero . In the case of UWA communications, the data signal subspace is generally full rank; however, as stated earlier, the channel subspace is not necessarily full rank. Despite these subtle differences between traditional signal subspace techniques, and the proposed channel subspace approach, there are striking similarities in that performance improvement in both cases is obtained by exploiting the reduced rank nature of the relevant parameter.

1.4 Thesis organization

This thesis is organized as follows. Chapter 2 introduces a theoretical framework for analyzing the tracking performance of least squares algorithms. Analytical expressions for the performance of the commonly used exponentially windowed and sliding windowed recursive least squares algorithms are derived. The channel subspace filtering (CSF) method is introduced and additional expressions are derived that explicitly indicate the improvement in performance to be expected by CSF of the least squares channel estimates. The special case of low-rank channels is examined and the performance of a directly constrained reduced rank least squares algorithm is analyzed using the methodology developed earlier in the chapter. The tracking performance due to CSF of the reduced rank least squares estimates is compared to and shown, analytically, to be equal to that due to CSF of conventional full rank least squares estimates. This equivalence is attributed to the ability of the CSF to exploit both the reduced dimensionality of such a low-rank channel, as well the correlation between the tracking error and the true channel impulse response.

These theoretical predictions and results from the previous chapter are corroborated using simulations in Chapter 3. Additionally, Chapter 3 examines the performance of the channel subspace post-filtering algorithm when a finite number of channel estimates, corresponding to finite number of received data samples, are used to estimate the eigenvectors of the channel subspace filter. It is shown that the

correlation between successive channel estimates affects the number of independent channel estimates available given a set of finite received data. The impact of a finite number of independent channel estimates on errors in eigenvectors estimation is discussed analytically and using simulated data.

Chapter 4 details an adaptive channel subspace filtering (CSF) algorithm for use in an experimental scenario. The performance of this adaptive CSF algorithm is also examined in the context of a finite number of received data samples. It is again shown that a finite number of channel estimates affects the estimation of the eigenvectors of the process and subsequently the estimation of the CSF coefficients thereby affecting performance. This impact, of finite data samples, is demonstrated using simulated data. Additionally, the adaptive CSF (AD-CSF) algorithm is used to process experimental data representing a range of channel conditions. The AD-CSF algorithm and a sub-optimal abrupt rank reduction (ARR) algorithm to be used in an experimental setup are described. The causal and non-causal variants of these algorithms are compared qualitatively in terms of their computational complexity requirements . These algorithms are then used to process actual experimental data and the improvement in performance due to CSF is presented and analyzed. The performance of the AD-CSF algorithm and ARR algorithm are compared to illustrate that, even in experimental data, the improvement in performance of the AD-CSF algorithm is due to both the exploiting of the reduced rank of the channel and the Wiener post-filtering. As expected, the non-causal variant of the AD-CSF algorithm outperforms the causal variant, however the deterioration in performance due to the use of a causal algorithm is not substantial. The causal variant of the algorithm still demonstrates considerable performance improvement over the traditional RLS algorithm and is certainly more applicable in a realistic communications perspective. The significant computational ease in implementing and comparing the results of the non-causal variant make it a suitable algorithm for rapidly demonstrating the applicability of the CSF technique on additional experimental data sets.

Chapter 5 describes the theoretical framework for a channel estimate based decision feedback equalizer (DFE). An analysis describing the impact of channel estima-

tion errors on the performance of such a DFE leads to a discussion of the theoretical improvements in performance expected due to a channel subspace post-filtered channel estimate based DFE. This improvement in is explicitly demonstrated on simulated data. The improvement in performance obtained when used to process the same experimental data used in Chapter 4 is also discussed.

Chapter 6 summarizes the contributions of this thesis and suggests possible directions for future research.

Chapter 2

Tracking time-varying systems

Developing the framework and methodology that has been alluded to so far needs work on two fronts. Firstly, the basic underlying theory must be uncovered. Equally importantly, the technique must be shown to be useful for real-world examples. In this chapter, least-squares metrics of tracking performance are introduced, and the improvement in performance expected due to CSF of RLS channel estimates is demonstrated analytically. For low-rank channels, it is shown that CSF of the unconstrained RLS estimates can yield similar tracking performance as a directly constrained reduced rank RLS algorithm to within the limits of the direct averaging assumption used to generate these analytical results. This forms the basis for the CSF paradigm for low-rank channels which suggests that the penalty incurred in conventional least-squares algorithms due to overestimation of the number of parameters to be tracked can be eliminated by using the CSF approach.

2.1 Markov model for system identification

A simplified time-varying first-order Gauss-Markov model for system identification is depicted in figure 2-1. The unknown dynamic system is modeled as a transversal filter whose tap-weight vector $\underline{h}(n)$, its impulse response, evolves according to a first

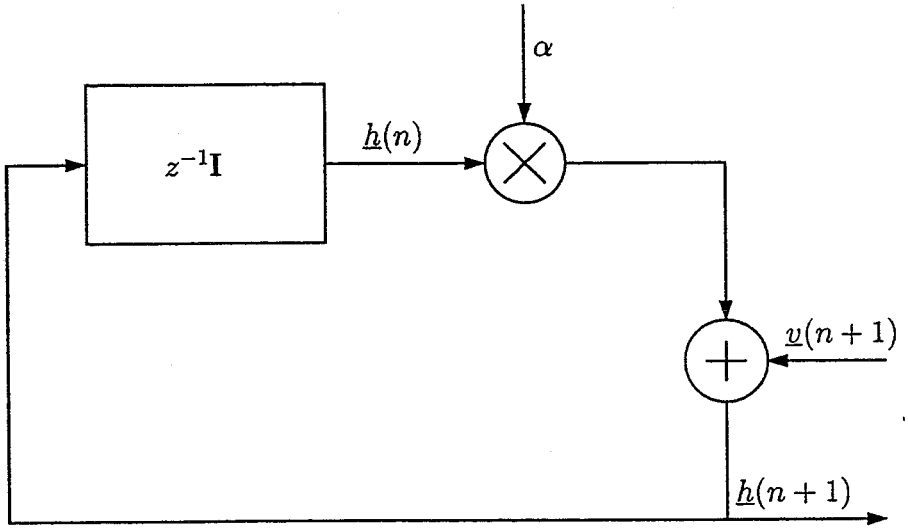

Figure 2-1: First-order Gauss-Markov model for time-varying system

order Markov process written in vector form as:

$$\underline{h}(n+1) = \alpha \underline{h}(n) + \underline{v}(n+1) \tag{2.1}$$

where the underscore denotes vectors, and all vectors are $N \times 1$ column vectors. The time-varying channel impulse response at time n is represented by the vector $\underline{h}(n)$ with the process noise $\underline{v}(n)$ having a correlation matrix $\mathbf{R}_{vv} = E[\underline{v}(n)\underline{v}^H(n)]$. The output of the system is given by:

$$y(n) = \underline{h}^H(n)\underline{x}(n) + w(n) \tag{2.2}$$

where the superscript H represents the conjugate transpose, the received data is $y(n)$, $\underline{x}(n)$ denotes the white transmitted data with correlation matrix $\mathbf{R}_{xx} = E[\underline{x}(n)\underline{x}^H(n)] = \mathbf{I}$, and $w(n)$ denotes white, Gaussian additive observation noise with a variance of σ^2. If $\mid \alpha \mid < 1$, (2.1) and (2.2) collectively describe a time-varying system with stationary statistics.

If $\hat{\underline{h}}(n \mid n-1)$ denotes an estimate of the true channel impulse response $\underline{h}(n)$ obtained using data up to time $n-1$, then the predicted data at time n, $\hat{y}(n \mid n-1)$

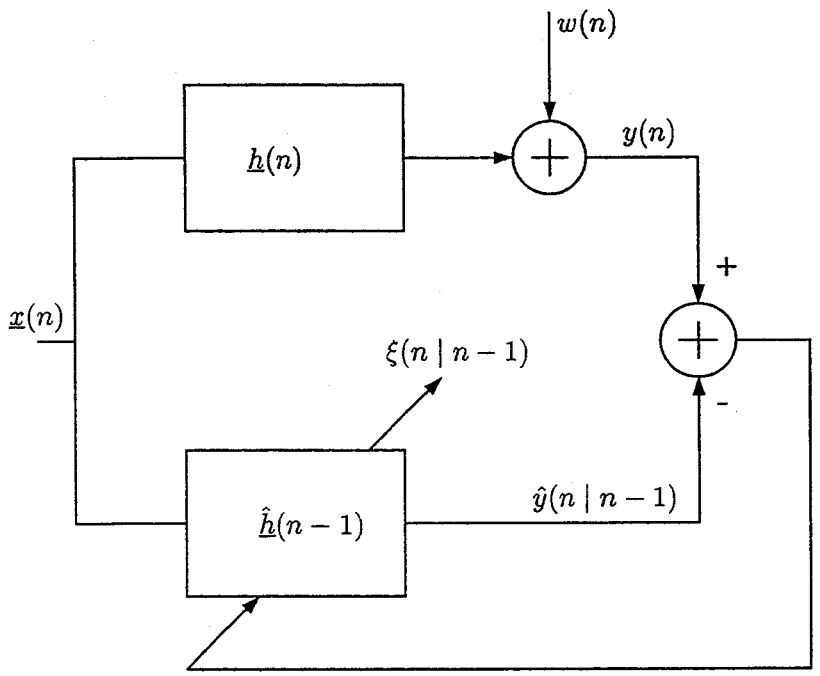

Figure 2-2: System identification using an adaptive filter

is given by:

$$\hat{y}(n \mid n-1) = \hat{\underline{h}}^{H}(n \mid n-1)\underline{x}(n) \tag{2.3}$$

The corresponding prediction error $\xi(n \mid n-1)$ is given by:

$$\begin{aligned} \xi(n \mid n-1) &= y(n) - \hat{y}(n \mid n-1) \\ &= y(n) - \hat{\underline{h}}^{H}(n \mid n-1)\underline{x}(n) \end{aligned} \tag{2.4}$$

The prediction error is used to the adapt the weights of the channel estimate $\hat{\underline{h}}(n \mid n-1)$. Throughout the remainder of this thesis, for notational simplicity, $\hat{\underline{h}}(n-1)$ is used to denote $\hat{\underline{h}}(n \mid n-1)$. The adaptation process is depicted in figure 2-2. The estimated channel impulse response $\hat{\underline{h}}(n)$ is assumed to be transversal in nature too. An assumption is also made that the number of taps in the unknown system $\underline{h}(n)$ is the same as the number of taps N in the adaptive filter $\hat{\underline{h}}(n)$ used to model and track the system. Although implicitly obvious, it is a very important subtlety that restricts the tracking performance of an adaptive filter, particularly when the number of taps that are tracked increases. The impact of this trade-off is discussed more formally in

later sections and the contribution of the channel subspace filtering approach shall become clearer then. It is also important to realize the assumptions made in using the above model. These assumptions, which shall be used in subsequent sections when analyzing the tracking performance of least-squares tracking algorithms, are listed below:

- The process noise vector is $\underline{v}(n)$ is independent of both the input (data) vector $\underline{x}(n)$ and the observation noise $w(n)$.
- The input data vector $\underline{x}(n)$ and observation noise $w(n)$ are independent of each other.

2.2 Criteria for tracking performance assessment

The channel estimation error vector, also referred to as the tracking error vector, may be defined as:

$$\underline{\epsilon}(n \mid n-1) = \underline{h}(n) - \hat{\underline{h}}(n-1) \tag{2.5}$$

As defined earlier, the prediction error is given by:

$$\begin{aligned} \xi(n \mid n-1) &= y(n) - \hat{y}(n \mid n-1) \\ &= y(n) - \hat{\underline{h}}^H(n-1)\underline{x}(n) \end{aligned} \tag{2.6}$$

The relationship between these two error metrics, assuming the linear channel model given in (2.2), can be written as:

$$\begin{aligned} \xi(n \mid n-1) &= \underline{h}^H(n)\underline{x}(n) + w(n) - \hat{\underline{h}}^H(n-1)\underline{x}(n) \\ &= \left(\underline{h}(n) - \hat{\underline{h}}(n-1)\right)^H \underline{x}(n) + w(n) \end{aligned} \tag{2.7}$$

Substituting the expression for the channel estimation error vector given in (2.5) results in:

$$\xi(n \mid n-1) = \underline{\epsilon}^H(n \mid n-1)\underline{x}(n) + w(n) \tag{2.8}$$

Based on the channel estimation error vector, $\underline{\epsilon}(n \mid n-1)$, a commonly used figure of merit known as the mean-square channel estimation error is defined such that:

$$\begin{aligned} \mathcal{D}(n) &= E[\| \underline{h}(n) - \underline{\hat{h}}(n-1) \|^2] \\ &= E[\| \underline{\epsilon}(n \mid n-1) \|^2] \end{aligned} \tag{2.9}$$

where the number of iterations n is assumed to be large enough for the adaptive filter's transient mode of operation to have finished. (2.9) may be alternately written as:

$$\mathcal{D}(n) = tr[\mathbf{R}_{\epsilon\epsilon}(n)] \tag{2.10}$$

where $\mathbf{R}_{\epsilon\epsilon}(n)$ is the correlation matrix of the error vector $\underline{\epsilon}(n \mid n-1)$:

$$\mathbf{R}_{\epsilon\epsilon}(n) = E[\underline{\epsilon}(n \mid n-1)\underline{\epsilon}^H(n \mid n-1)] \tag{2.11}$$

The mean-square prediction error can also be expressed in terms of the mean-square channel estimation error $\mathcal{D}(n)$ using (2.8) such that:

$$\begin{aligned} E[\| \xi(n \mid n-1) \|^2] &= tr\Big(E[\underline{\epsilon}(n)\underline{\epsilon}^H(n)] \underbrace{E[\underline{x}(n)\underline{x}^H(n)]}_{\mathbf{I}}\Big) + E[w^2(n)] \\ &= \mathcal{D}(n) + \sigma^2 \end{aligned} \tag{2.12}$$

where the transmitted data vector $\underline{x}(n)$, based on the assumptions stated earlier, is assumed to be uncorrelated with the channel estimation error vector $\underline{\epsilon}(n)$. The correlation matrix of the transmitted data vector is assumed to be white so that $\mathbf{R}_{xx} = E[\underline{x}(n)\underline{x}^H(n)] = \mathbf{I}$, and the additive observation noise $w(n)$ is assumed to white with variance $E[w^2(n)] = \sigma^2$ and uncorrelated with the channel impulse response $\underline{h}(n)$. Naturally, $\mathcal{D}(n)$ should be small for good tracking performance. Equivalently, given the relationship between the channel estimation error variance and the mean square prediction error in (2.12), if the statistics of the additive observation noise are stationary, then a reduction in channel estimation error leads to corresponding decrease in the prediction error. In a realistic scenario where the true channel impulse

response is unknown, it is difficult to ascertain the true channel estimation error. The mean-square prediction error can be used as a surrogate for the mean-square channel estimation error in gauging the performance of an adaptive algorithm. The notion of *improvement* in performance, given this framework, is context-independent because of the equivalence between the two error metrics, as expressed in (2.12).

The mean-square channel estimation error can be expressed as a sum of two components:

$$\mathcal{D}(n) = \mathcal{D}_1(n) + \mathcal{D}_2(n) \tag{2.13}$$

where the first term $\mathcal{D}_1(n) = tr(\mathbf{D}_1(n))$ is referred to the tracking error variance and is impacted by the time-varying process. The second term $\mathcal{D}_2(n) = tr(\mathbf{D}_2(n))$ is referred to as the observation noise induced error variance and depends on the additive observation noise. Since these terms are correlated there is no analytical way of computing them apriori. The terms in an analytical expression for $\mathcal{D}(n)$ are grouped into terms that depend on the dynamics of the process, denoted by $\mathcal{D}_1(n)$, and terms that depend on the observation noise variance, denoted by $\mathcal{D}_2(n)$. Separating these error terms, as above, provides insight into the tradeoff involved in designing an adaptive algorithm to track a time-varying system. For a given adaptive algorithm, best performance is achieved by selecting a rate of adaptation that balances the improvement due to any reduction in the tracking error variance $\mathcal{D}_1(n)$ with any resultant deterioration due an increase in the observation noise induced error variance $\mathcal{D}_2(n)$. The term $\mathcal{D}_1(n)$ is present solely because of the time-varying nature of the system, whereas the term $\mathcal{D}_2(n)$ is present because of the additive observation noise. If the algorithm were operating in a time-invariant environment, the tracking error variance would vanish so that the performance of the algorithm would be affected solely by the observation error induced variance. While evaluating any improvement in performance due to the use of the channel-subspace filtering (CSF) approach, the channel estimation error metric shall be used for convenience even though the notion of performance improvement is metric independent as discussed earlier. The following section formally introduces commonly used least-squares tracking algorithms, theoretically analyzes,

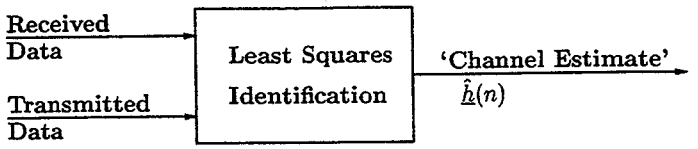

Figure 2-3: Least squares tracking algorithms

within a common framework, the tracking performance of these algorithms in terms of the channel estimation error and provides a basis for evaluating the improvement due to channel subspace post-filtering.

2.3 Least-squares tracking algorithms

Adaptive tracking algorithms generate a channel estimate $\hat{\underline{h}}(n)$ based on the knowledge of the transmitted data, the received data and an assumed linear model for the output of the time-varying system as in (2.2). Least squares tracking algorithms, as represented in figure 2-3, attempt to compute the channel estimate so as to minimize a cost function that is expressed as a weighted sum of the squared received signal prediction error. When ensemble averages are used to compute the channel estimate, it represents a Wiener filter solution to the minimum mean-squared error filtering problem. In a recursive implementation of the method of least squares, the estimate is initialized with assumed initial conditions and information contained in new data samples is used to update old estimates. The length of observable data to be used in making such an update is variable and depends on the specific nature of the algorithm. Accordingly, the problem of determining the channel estimate can be expressed as a least squares problem by introducing a cost function:

$$\begin{aligned} \mathcal{E}(n,\underline{h}) &= \sum_{k=1}^{n} \beta(n,k) \parallel y(k) - \underline{h}^H \underline{x}(k) \parallel^2 \\ &= \sum_{k=1}^{n} \beta(n,k) \parallel \xi_h(k) \parallel^2 \end{aligned} \tag{2.14}$$

where

$$\xi_h(k) = y(k) - \underline{h}^H \underline{x}(k) \tag{2.15}$$

is the prediction error between the received signal $y(n)$ and the predicted signal produced using a channel estimate $\underline{h}$. It is important to differentiate between the expressions $\xi_h(n)$ in (2.15) and $\xi(n \mid n-1)$ in (2.4). The specific contexts in which they are used should make their difference obvious. The weighting function $\beta(n,k)$ is intended to ensure that the distant past is "forgotten" in order to afford the possibility of following the variations in the observable data when the filter operates in a time-varying environment. In general, the weighting factor $\beta(n,k)$ has the property that:

$$0 < \beta(n,k) \leq 1, \qquad \text{for } k = 1, 2, \ldots, n \tag{2.16}$$

The least squares channel estimate $\hat{\underline{h}}(n)$ minimizes this cost function (2.14) for a given weighting factor $\beta(n,k)$ so that:

$$\begin{aligned} \hat{\underline{h}}(n) &= \arg\min_{\underline{h}} \mathcal{E}(n, \underline{h}) \\ &= \arg\min_{\underline{h}} \sum_{k=1}^{n} \beta(n,k) \parallel \xi_h(k) \parallel^2 \\ &= \arg\min_{\underline{h}} \sum_{k=1}^{n} \beta(n,k) \parallel y(k) - \underline{h}^H \underline{x}(k) \parallel^2 \end{aligned} \tag{2.17}$$

Two of the more popular least squares algorithms are the sliding window RLS (SW-RLS) and the exponentially windowed RLS (EW-RLS) algorithms. Window design, which impacts the cost function through the choice of the weighting function, is still an active area of research. Optimal window design is one of the many issues discussed in greater detail in [27]. The weighting function for the SW-RLS algorithm with a sliding window of size M is given by:

$$\beta(n,k) = \begin{cases} \frac{1}{M} & \text{if } k = n - M + 1, \ldots, n, \\ 0 & \text{if } k > n \text{ and } k < n - M + 1. \end{cases} \tag{2.18}$$

while the weighting function for the EW-RLS algorithm with a forgetting factor of λ is given by:

$$\beta(n,k) = (1-\lambda)\lambda^{n-k} \tag{2.19}$$

Solving (2.17) using the weighting functions specified in (2.18) and (2.19) results in the specific form for the SW-RLS and the EW-RLS algorithms respectively.

2.4 SW-RLS algorithm

Substituting the weighting function (2.18) in (2.17) results in the following least-squares error criterion:

$$\hat{\underline{h}}(n) = \arg\min_{\underline{h}} \sum_{k=n-M+1}^{n} \frac{1}{M} \parallel \underline{h}^H \underline{x}(k) - y(k) \parallel^2 \tag{2.20}$$

where M is the size of the sliding window. This equation can be solved deterministically with the channel estimate $\hat{\underline{h}}(n)$ given by:

$$\hat{\underline{h}}(n) = \mathbf{R}_{xx}^{-1}(n)\mathbf{R}_{xy}(n) \tag{2.21}$$

where the matrices $\mathbf{R}_{xx}(n)$ and $\mathbf{R}_{xy}(n)$ are computed as:

$$\mathbf{R}_{xx}(n) = \frac{1}{M} \sum_{k=n-M+1}^{n} \underline{x}(k)\underline{x}^H(k) \tag{2.22}$$

$$\mathbf{R}_{xy}(n) = \frac{1}{M} \sum_{k=n-M+1}^{n} \underline{x}(k)y^*(k) \tag{2.23}$$

Substituting the ensemble average for $\mathbf{R}_{xx}^{-1}(n)$ in (2.21) from (A.14) results in:

$$\begin{aligned} E[\mathbf{R}_{xx}^{-1}(n)] &= \frac{M}{M-N-1} \quad \mathbf{R}_{xx}^{-1} \\ &= \frac{M}{M-N-1} \quad \mathbf{I} \end{aligned} \tag{2.24}$$

where N is the number of taps in channel estimate vector $\hat{\underline{h}}(n)$ and the transmitted data correlation matrix $\mathbf{R}_{xx} = \mathbf{I}$ is assumed to be white. Substituting (2.24) and (2.23) in (2.21) it can be shown [24] that:

$$E[\hat{\underline{h}}(n)] \simeq \sum_{k=n-M+1}^{n} \frac{1}{M-N-1} \underline{h}(k) \tag{2.25}$$

Ling and Proakis used a frequency domain approach in [27] to establish the approximate equivalence between the SW-RLS and the EW-RLS algorithms in regimes where the fading rate of the channel is slow compared to the symbol rate. For the model given in (2.1), the parameter α characterizes the fading rate of the channel. A regime where the fading rate is slow is one where the value of the parameter α in (2.1) is very close to 1. In such a regime, the frequency domain methodology detailed in [27] leads to the following equivalence between an EW-RLS algorithm with tracking parameter λ, and an SW-RLS algorithm with a sliding window of size M when the channel estimate vector has N taps:

$$\lambda = 1 - \frac{2}{M-N-1} \tag{2.26}$$

The following section examines the tracking performance of the EW-RLS algorithm with the tacit assumption that its equivalence with the SW-RLS algorithm as expressed in (2.26) provides the basis for the use of a common framework in analyzing the tracking performance of both these RLS algorithms.

2.5 Tracking performance of EW-RLS algorithm

Substituting the weighting function (2.19) in (2.17) results in the following least-squares error criterion for the EW-RLS algorithm:

$$\hat{\underline{h}}(n) = \arg\min_{\underline{h}} \sum_{k=1}^{n} \lambda^{n-k} \parallel y(k) - \underline{h}^H \underline{x}(k) \parallel^2 \tag{2.27}$$

where λ is a positive constant close to, but less than 1. Although the solution to this equation, for any λ, is defined by a solution to a set of normal equations similar to the ones obtained for the SW-RLS algorithm, the solution may also be expressed in a recursive form. The generalized coefficient update equation for such a least squares algorithm is given by:

$$\underline{\hat{h}}(n) = \underline{\hat{h}}(n-1) + \underline{\mathbf{K}}(n)\xi^*(n \mid n-1) \tag{2.28}$$

where $\underline{\mathbf{K}}(n)$ is an adaptation gain vector and $\xi(n \mid n-1)$ is the prediction error defined in (2.4). The gain vector may be computed as:

$$\underline{\mathbf{K}}(n) = \Big(\sum_{i=1}^{n} \lambda^{n-i}\underline{x}(i)\underline{x}^H(i)\Big)^{-1}\underline{x}(n) \qquad \text{for } \lambda \in [0,1] \tag{2.29}$$

Subtracting (2.1) from (2.28) and using (2.5) results in:

$$\underline{\epsilon}(n+1 \mid n) = \alpha\underline{h}(n) + \underline{v}(n+1) - \Big(\underline{\hat{h}}(n-1) + \underline{\mathbf{K}}(n)\xi^*(n \mid n-1)\Big) \tag{2.30}$$

Substituting the expression for $\xi(n \mid n-1)$ in (2.4) and using (2.5) to rearrange some of the terms results in:

$$\begin{aligned}\underline{\epsilon}(n+1 \mid n) = (\alpha-1)\underline{h}(n) + \Big(\mathbf{I} - \underline{\mathbf{K}}(n)\underline{x}^H(n)\Big)\underline{\epsilon}(n \mid n-1) \\ - \underline{\mathbf{K}}(n)w^*(n) + \underline{v}(n+1)\end{aligned} \tag{2.31}$$

(2.1) and (2.31), together, form a coupled state space system that models the dynamics of the unknown system as well as that of the adaptive algorithm used to update the coefficients of the channel estimation error vector. Since their dynamics are coupled, their behavior in steady state may be analyzed by studying the dynamics of the equivalent extended state space system in steady state. Consider the augmented vector:

$$\underline{g}(n) = \begin{bmatrix} \underline{h}(n) \\ \underline{\epsilon}(n \mid n-1) \end{bmatrix} \tag{2.32}$$

Combining (2.31) and (2.1), the extended state space system governing the evolution of $\underline{g}(n)$ may be described as :

$$\begin{pmatrix} \underline{h}(n+1) \\ \underline{\epsilon}(n+1 \mid n) \end{pmatrix} = \begin{pmatrix} \alpha\mathbf{I} & \mathbf{0} \\ (\alpha-1)\mathbf{I} & \mathbf{I} - \underline{\mathbf{K}}(n)\underline{x}^H(n) \end{pmatrix} \begin{pmatrix} \underline{h}(n) \\ \underline{\epsilon}(n \mid n-1) \end{pmatrix} + \begin{pmatrix} \underline{v}(n+1) \\ \underline{v}(n+1) - \underline{\mathbf{K}}(n)w^*(n) \end{pmatrix} \tag{2.33}$$

Equivalently, the system may be rewritten as:

$$\underline{g}(n+1) = \mathbf{A}(n)\underline{g}(n) + \underline{B}(n) \tag{2.34}$$

where $\mathbf{A}(n)$ and $\underline{B}(n)$ are defined appropriately based on term by term comparison with (2.33). (2.34) is a stochastic difference equation in the augmented vector $\underline{g}(n)$ that has the system matrix equal to $\mathbf{A}(n)$. For α and λ close to 1, the expected value of the system matrix is approximately equal to the identity matrix for all n. This can be readily seen by noting that:

$$\mathbf{A}(n) = \begin{pmatrix} \alpha\mathbf{I} & \mathbf{0} \\ (\alpha-1)\mathbf{I} & \mathbf{I} - \underline{\mathbf{K}}(n)\underline{x}^H(n) \end{pmatrix} \tag{2.35}$$

where $E[\underline{\mathbf{K}}(n)\underline{x}^H(n)] = (1-\lambda)\mathbf{I}$ [10] so that $E[\mathbf{I} - \underline{\mathbf{K}}(n)\underline{x}^H(n)] = \lambda\mathbf{I}$. Thus:

$$E[\mathbf{A}(n)] = \begin{pmatrix} \alpha\mathbf{I} & \mathbf{0} \\ (\alpha-1)\mathbf{I} & \lambda\mathbf{I} \end{pmatrix} \approx \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \tag{2.36}$$

To study the convergence behavior of such a stochastic algorithm in an average sense, the direct averaging method [21] may be invoked. According to this method, the solution of the stochastic difference equation (2.34), operating under the assumption of α, λ close to 1, is close to the solution of another stochastic difference equation

whose system matrix is equal to the ensemble average:

$$E[\mathbf{A}(n)] = \begin{pmatrix} \alpha\mathbf{I} & \mathbf{0} \\ (\alpha-1)\mathbf{I} & \lambda\mathbf{I} \end{pmatrix} \tag{2.37}$$

More specifically, the stochastic difference equation in (2.34) may be replaced with another stochastic difference equation described by:

$$\underline{g}(n+1) = E[\mathbf{A}(n)]\underline{g}(n) + \underline{B}(n) \tag{2.38}$$

Generally, the notation in (2.38) should be different from that in the original difference equation in (2.34). However it has been chosen not to do so for the sake of convenience of presentation. The correlation matrix $\mathbf{R}_{gg}$ is given by:

$$\mathbf{R}_{gg} = E[\underline{g}(n+1)\underline{g}^H(n+1)] \tag{2.39}$$

where the expectation is taken with respect to the transmitted data vector $\underline{x}(n)$ and the time-varying channel impulse response $\underline{h}(n)$. Expressing $\underline{g}(n+1)$ as in (2.38), the steady state correlation matrix $\mathbf{R}_{gg}$ may be expressed as:

$$\begin{aligned} \mathbf{R}_{gg} &= E[\underline{g}(n+1)\underline{g}^H(n+1)] \\ &= E[\mathbf{A}(n)]E[\underline{g}(n)\underline{g}^H(n)]E[\mathbf{A}^H(n)] + E[\underline{B}(n)\underline{B}^H(n)] \end{aligned} \tag{2.40}$$

where it is assumed that the process noise vector $\underline{v}(n)$, the transmitted data vector $\underline{x}(n)$, Gaussian additive observation noise $w(n)$, and the adaptation gain vector $\underline{\mathbf{K}}(n)$ are uncorrelated with both the channel impulse response vector $\underline{h}(n)$ and the channel estimation error vector $\underline{\epsilon}(n \mid n-1)$. Since $\underline{g}(n)$ can be written as (2.32), the steady state correlation matrix $\mathbf{R}_{gg}$ can be expressed in terms of the corresponding sub matrices as:

$$\mathbf{R}_{gg} = \begin{pmatrix} \mathbf{R}_{hh} & \mathbf{R}_{h\epsilon} \\ \mathbf{R}_{\epsilon h} & \mathbf{R}_{\epsilon\epsilon} \end{pmatrix} \tag{2.41}$$

where:

$$\mathbf{R}_{hh} = E[\underline{h}(n)\underline{h}^H(n)] \tag{2.42}$$

$$\mathbf{R}_{h\epsilon} = E[\underline{h}(n)\underline{\epsilon}^H(n)] \tag{2.43}$$

$$\mathbf{R}_{\epsilon h} = E[\underline{\epsilon}(n)\underline{h}^H(n)] \tag{2.44}$$

$$\mathbf{R}_{\epsilon\epsilon} = E[\underline{\epsilon}(n)\underline{\epsilon}^H(n)] \tag{2.45}$$

are the corresponding correlation sub-matrices. For notational convenience $\underline{\epsilon}(n)$ shall be interchangeably used with $\underline{\epsilon}(n \mid n-1)$ to represent the channel estimation error vector given in (2.5). The correlation matrix $\mathbf{R}_{gg}$ of the augmented vector $\underline{g}(n)$ under steady state (when n is very large) can be written as:

$$\begin{aligned}\begin{pmatrix} \mathbf{R}_{hh} & \mathbf{R}_{h\epsilon} \\ \mathbf{R}_{\epsilon h} & \mathbf{R}_{\epsilon\epsilon} \end{pmatrix} &= \begin{pmatrix} \alpha\mathbf{I} & \mathbf{0} \\ (\alpha-1)\mathbf{I} & \lambda\mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{R}_{hh} & \mathbf{R}_{h\epsilon} \\ \mathbf{R}_{\epsilon h} & \mathbf{R}_{\epsilon\epsilon} \end{pmatrix} \begin{pmatrix} \alpha\mathbf{I} & \mathbf{0} \\ (\alpha-1)\mathbf{I} & \lambda\mathbf{I} \end{pmatrix}^H \\ &\quad + E[\underline{B}(n)\underline{B}^H(n)]\end{aligned} \tag{2.46}$$

It is assumed that the additive observation noise $w(n)$ is zero mean and uncorrelated with both the transmitted data vector $\underline{x}(n)$ and the process noise vector $\underline{v}(n+1)$. Furthermore, the independence between the Gaussian additive noise $w(n)$ and the transmitted data vector is exploited $\underline{x}(n)$ in deriving the expression:

$$\begin{aligned} E[\underline{\mathbf{K}}(n)w^*(n)\underline{\mathbf{K}}^H(n)w(n)] &= E[\underline{\mathbf{K}}(n)\underline{\mathbf{K}}^H(n)]E[w^2(n)] \\ &= \sigma^2(1-\lambda)^2\mathbf{R}_{xx}^{-1} \end{aligned} \tag{2.47}$$

so that $E[\underline{B}(n)\underline{B}^H(n)]$ in (2.46) can be written as:

$$\begin{aligned} &E[\underline{B}(n)\underline{B}^H(n)] \\ &= E\left[\begin{pmatrix} \underline{v}(n+1) \\ \underline{v}(n+1) - \underline{\mathbf{K}}(n)w^*(n) \end{pmatrix} \begin{pmatrix} \underline{v}^H(n+1) & \underline{v}^H(n+1) - \underline{\mathbf{K}}^H(n)w(n) \end{pmatrix} \right] \end{aligned} \tag{2.48}$$

so that:

$$E[\underline{B}(n)\underline{B}^H(n)] = \begin{pmatrix} \mathbf{R}_{vv} & \mathbf{R}_{vv} & \\ \mathbf{R}_{vv} & \mathbf{R}_{vv} & +\sigma^2(1-\lambda)^2\mathbf{R}_{xx}^{-1} \end{pmatrix} \tag{2.49}$$

By expanding out terms on the right hand side of (2.46) using (2.49) and relating them to the corresponding terms of the resultant sub matrices on either side of the expression, the following relationships are obtained:

$$\mathbf{R}_{hh} = \frac{1}{(1-\alpha^2)}\mathbf{R}_{vv} \tag{2.50}$$

$$\mathbf{R}_{h\epsilon} = \alpha(\alpha-1)\mathbf{R}_{hh} + \alpha\lambda\mathbf{R}_{h\epsilon} + \mathbf{R}_{vv} \tag{2.51}$$

$$\mathbf{R}_{\epsilon h} = \alpha(\alpha-1)\mathbf{R}_{hh} + \alpha\lambda\mathbf{R}_{h\epsilon} + \mathbf{R}_{vv} \tag{2.52}$$

$$\mathbf{R}_{\epsilon\epsilon} = (\alpha-1)^2\mathbf{R}_{hh} + (\alpha-1)\lambda\mathbf{R}_{\epsilon h} + (\alpha-1)\lambda\mathbf{R}_{h\epsilon} + \lambda^2\mathbf{R}_{\epsilon\epsilon} + \mathbf{R}_{vv} + \sigma^2(1-\lambda)^2\mathbf{R}_{xx}^{-1} \tag{2.53}$$

From (2.52) and (2.51) it is clear that:

$$\mathbf{R}_{\epsilon h} = \mathbf{R}_{h\epsilon} \tag{2.54}$$

Substituting (2.54) in the equations above and eliminating common terms the following relationships are obtained:

$$\mathbf{R}_{h\epsilon} = \frac{1}{(1+\alpha)(1-\alpha\lambda)}\mathbf{R}_{vv} \tag{2.55}$$

$$\mathbf{R}_{\epsilon\epsilon} = \underbrace{\frac{2}{(1+\alpha)(1+\lambda)(1-\alpha\lambda)}\mathbf{R}_{vv}}_{\mathbf{D}_1} + \underbrace{\frac{(1-\lambda)}{(1+\lambda)}\sigma^2\mathbf{R}_{xx}^{-1}}_{\mathbf{D}_2} \tag{2.56}$$

where the underbraced terms in (2.56) represent the previously mentioned tracking error variance $\mathcal{D}_1 = tr(\mathbf{D}_1)$ and the observation noise induced error variance $\mathcal{D}_2 = tr(\mathbf{D}_2)$ terms. Since $\underline{\epsilon}(n \mid n-1) = \underline{h}(n) - \underline{\hat{h}}(n-1)$, $\mathbf{R}_{\epsilon h}$ and $\mathbf{R}_{h\epsilon}$ may be expressed

as:

$$\mathbf{R}_{\epsilon h} = \mathbf{R}_{hh} - \mathbf{R}_{\hat{h}h} \tag{2.57}$$

$$\mathbf{R}_{h\epsilon} = \mathbf{R}_{hh} - \mathbf{R}_{h\hat{h}} \tag{2.58}$$

so that using (2.54), (2.57) and (2.58) it is clear that:

$$\mathbf{R}_{\hat{h}h} = \mathbf{R}_{h\hat{h}} \tag{2.59}$$

Based on (2.59) and (2.54) the following relationships are obtained:

$$\mathbf{R}_{h\hat{h}} = \mathbf{R}_{\hat{h}h} = \mathbf{R}_{hh} - \mathbf{R}_{h\epsilon} \tag{2.60}$$

$$\mathbf{R}_{\hat{h}\hat{h}} = \mathbf{R}_{hh} - \mathbf{R}_{h\epsilon} - \mathbf{R}_{\epsilon h} + \mathbf{R}_{\epsilon\epsilon} \tag{2.61}$$

After substituting (2.50), (2.55), and (2.56) into the above equations the following simplified expressions are obtained:

$$\mathbf{R}_{\hat{h}h} = \frac{\alpha(1-\lambda)}{(1-\alpha^2)(1-\alpha\lambda)}\mathbf{R}_{vv} \tag{2.62}$$

$$\mathbf{R}_{\hat{h}\hat{h}} = \frac{(1+\alpha\lambda)(1-\lambda)}{(1-\alpha^2)(1-\alpha\lambda)(1+\lambda)}\mathbf{R}_{vv} + \frac{(1-\lambda)}{(1+\lambda)}\sigma^2\mathbf{R}_{xx}^{-1} \tag{2.63}$$

The steady state mean square channel estimation error $tr(\mathbf{R}_{\epsilon\epsilon})$ obtained from (2.56) expresses the inherent tradeoff between tracking error variance $\mathcal{D}_1$ and the observation noise induced error variance $\mathcal{D}_2$ when a rate of adaptation λ is chosen. Furthermore, the channel estimate $\underline{\hat{h}}(n)$ and the channel impulse response vector $\underline{h}(n)$ are correlated and their cross correlation matrix $\mathbf{R}_{h\hat{h}} = \mathbf{R}_{\hat{h}h}$ is given by (2.62).

2.6 MMSE post-filtering

The EW-RLS channel estimate (2.27) may be treated as a noisy time series so that from (2.5):

$$\underline{\hat{h}}(n-1) = \underline{h}(n) - \underline{\epsilon}(n \mid n-1) \tag{2.64}$$

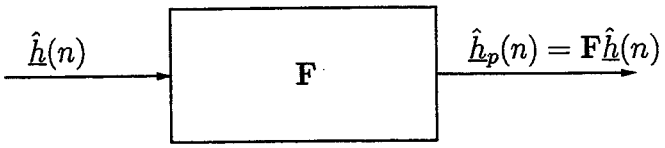

Figure 2-4: MMSE post-filtering

This noisy channel estimate can be post-filtered as shown in figure 2-4 so that :

$$\hat{\underline{h}}_p(n) = \mathbf{F}\hat{\underline{h}}(n) \tag{2.65}$$

where $\hat{\underline{h}}_p(n)$ is the post-filtered channel estimate and $\mathbf{F}$ is the post-filter. A MMSE post-filter $\mathbf{F}$ is given by:

$$\mathbf{F} = \mathbf{R}_{\hat{h}\hat{h}}^{-1}\mathbf{R}_{\hat{h}h} \tag{2.66}$$

In the absence of any other structure, computing the elements of the post-filter $\mathbf{F}$ as given by (2.66) would involve computation of $N \times N$ elements of the matrix $\mathbf{F}$. However, there is additional structure in the problem that can be exploited. The channel subspace post-filter presented in the following section exploits this structure.

2.7 Channel Subspace Filtering

The eigenvalue decomposition of $\mathbf{R}_{vv}$ is given by:

$$\mathbf{R}_{vv} = \mathbf{U}\Sigma_{vv}\mathbf{U}^H \tag{2.67}$$

where the columns of $\mathbf{U}$ are the eigenvectors of $\mathbf{R}_{vv}$ and Σ_{vv} is a diagonal matrix. Since, from (2.50), $\mathbf{R}_{hh} = \mathbf{R}_{vv}/(1-\alpha^2)$, the eigenvectors of the process correlation matrix $\mathbf{R}_{vv}$ are also the eigenvectors of the channel correlation matrix $\mathbf{R}_{hh}$. Also, under the white transmitted data assumption:

$$\mathbf{R}_{xx} = \mathbf{I} = \mathbf{U}\mathbf{U}^H \tag{2.68}$$

From (2.62)-(2.63), it is evident that the eigenvectors of the channel correlation matrix are also the eigenvectors of the correlation matrices $\mathbf{R}_{\hat{h}h}$ and $\mathbf{R}_{\hat{h}\hat{h}}$. Hence the post-filter $\mathbf{F}$ can be expressed in terms of these eigenvectors $\mathbf{U}$ as:

$$\mathbf{F} = \mathbf{U}\Sigma_{ff}\mathbf{U}^H \tag{2.69}$$

where $\Sigma_{ff} = \text{diag}(\sigma_{f1}^2, \sigma_{f2}^2, \ldots \sigma_{fN}^2)$. The post-filter $\mathbf{F}$ is referred to as a channel subspace filter (CSF) because its eigenvectors $\mathbf{U}$ are the same as those of the channel correlation matrix. The CSF coefficients which are the diagonal elements of Σ_{ff} are given by:

$$\sigma_{fi}{}^2 = \frac{\alpha\sigma_{hi}{}^2}{\frac{(1+\alpha\lambda)}{(1+\lambda)}\sigma_{hi}{}^2 + \frac{(1-\alpha\lambda)}{(1+\lambda)}\sigma^2} \leq 1 \tag{2.70}$$

where i=1, 2, 3, ..., N and $\sigma_{hi}^2 = \sigma_{vi}^2/(1-\alpha^2)$ is the energy corresponding to the i^{th} eigenvector. Thus the CSF coefficients depend on parameters α, λ, the observation noise energy σ^2, and the energy distribution in subspaces (subspace profile). This Wiener channel subspace filter weighs the subspaces with higher energy more favorably than subspaces with lower energy and eliminates the observation noise error associated with tracking any null subspaces.

The correlation matrix of the tracking error after post-filtering is diagonalized by the same eigenvectors $\mathbf{U}$ so that:

$$\mathbf{R}_{\epsilon f} = \mathbf{R}_{hh} - \mathbf{R}_{\hat{h}h}^H \mathbf{R}_{\hat{h}\hat{h}}^{-1} \mathbf{R}_{\hat{h}h} \tag{2.71}$$

or equivalently :

$$\mathbf{R}_{\epsilon f} = \mathbf{U}\Sigma_{\epsilon f}\mathbf{U}^H \tag{2.72}$$

whose diagonal elements $\Sigma_{\epsilon f} = \text{diag}(\sigma_{\epsilon f1}^2, \sigma_{\epsilon f2}^2, \ldots, \sigma_{\epsilon fN}^2)$ are given by:

$$\sigma_{\epsilon fi}^2 = \sigma_{hi}^2 \Big(1 - \frac{\frac{\alpha^2(1-\lambda)^2\sigma_{hi}^2}{(1-\alpha\lambda)^2}}{\frac{(1-\lambda)(1+\alpha\lambda)\sigma_{hi}^2}{(1-\alpha\lambda)(1+\lambda)} + \frac{1-\lambda}{1+\lambda}\sigma^2}\Big) \tag{2.73}$$

From (2.56), the equivalent diagonal matrix for the unfiltered RLS estimate is given

by:

$$\sigma_{\epsilon i}^2 = \frac{2(1-\alpha)}{(1+\lambda)(1-\alpha\lambda)}\sigma_{hi}^2 + \frac{1-\lambda}{1+\lambda}\sigma^2 \tag{2.74}$$

(2.74) is an analytical expression for the mean-square channel estimation error when an EW-RLS algorithm is used to track the time-varying system given in (2.1). Correspondingly (2.73) is analytical expression for the steady state mean-square channel estimation error of a CSF post-filtered EW-RLS algorithm. For an SW-RLS algorithm that produces a $N \times 1$ channel estimate $\underline{\hat{h}}(n)$ with a sliding window of size M, the channel estimation error correlation matrix before and after post-filtering is also approximated by the same diagonal matrices above, except that the tracking parameter $\lambda = 1 - \frac{2}{M-N-1}$ as given in (2.26). Since the post-filter $\mathbf{F}$ is an MMSE Wiener filter, a performance improvement would certainly be expected and even a cursory look at the forms of (2.74) and (2.73) validates this intuition.

Before the nature of this performance improvement is discussed in any greater detail, it is important to the extend the CSF framework by considering other interesting cases. Specifically, it is necessary to analyze the performance of a uncorrelated parameter form of least squares tracking algorithm, which is one where apriori information about the process eigenvectors is explicitly utilized in the design of an RLS algorithm. Subsequently, the performance of an uncorrelated parameter reduced rank RLS algorithm is considered for use in low-rank channels. The performance of the CSF approach for such an RLS algorithm is evaluated. These analyses are presented in the following sections and form the basis of the CSF paradigm for low-rank channels.

2.8 Uncorrelated parameter least-squares tracking

Consider the same first-order model for the time-varying system given in (2.1):

$$\underline{h}(n+1) = \alpha\underline{h}(n) + \underline{v}(n+1) \tag{2.75}$$

If the columns of $\mathbf{U}$ are the eigenvectors of the channel correlation matrix $\mathbf{R}_{hh}$, the projections onto the subspaces given by $\underline{p}_h(n) = \mathbf{U}^H \underline{h}(n)$ obey the same first order model such that:

$$\underline{p}_h(n+1) = \alpha \underline{p}_h(n) + \underline{p}_v(n+1) \tag{2.76}$$

where $\underline{p}_v(n) = \mathbf{U}^H \underline{v}(n)$ is the projection of the driving process $\underline{v}(n)$ onto the subspaces defined by the eigenvectors $\mathbf{U}$. Correspondingly, the true channel impulse response is related to the projected channel impulse response by the expression:

$$\underline{h}(n) = \mathbf{U}\underline{p}_h(n) \tag{2.77}$$

From (2.50) it is apparent that the eigenvectors of the channel correlation matrix are also the eigenvectors of the process correlation matrix $\mathbf{R}_{vv}$. In other words, these eigenvectors $\mathbf{U}$ diagonalize the the channel correlation matrix and hence the projected channel correlation matrix $\mathbf{R}_{p_h p_h} = \Sigma_{hh}$ is a diagonal matrix. The different "taps" of $\underline{p}_h(n)$ are now independent and hence uncorrelated. The $N \times N$ identity matrix represents the eigenvectors of this "new" process such that $\mathbf{U}_p = \mathbf{I}$. Given the knowledge of these eigenvectors, a uncorrelated parameter least squares algorithm can be formulated using (2.17) so that:

$$\underline{\hat{p}}_h(n) = \arg\min_{\underline{p}_h} \sum_{k=1}^{n} \beta(n,k) \parallel \xi_h(k) \parallel^2 \tag{2.78}$$

where apriori knowledge of the subspace structure is exploited by expressing $\xi_h(n)$ in (2.78) as :

$$\begin{aligned} \xi_h(n) &= y(n) - \underline{p}_h^H \mathbf{U}^H \underline{x}(n) \\ &= y(n) - \underline{p}_h^H \underline{x}_p(n) \end{aligned} \tag{2.79}$$

where $\underline{x}_p(n) = \mathbf{U}^H \underline{x}(n)$ is the projection of the data vector onto the eigenvectors represented by $\mathbf{U}$. This "new" data vector $\underline{x}_p(n)$ is also white in nature since:

$$
\begin{aligned}
E[\underline{x}_p(n)\underline{x}_p^H(n)] &= E[\mathbf{U}^H\underline{x}(n)\underline{x}^H(n)\mathbf{U}] \\
&= \mathbf{U}^H E[\underline{x}(n)\underline{x}^H(n)]\mathbf{U} \\
&= \mathbf{U}^H\mathbf{I}\mathbf{U} \\
&= \mathbf{U}^H\mathbf{U} = \mathbf{I}
\end{aligned}
\tag{2.80}
$$

Based on (2.79) , the least-squares problem of channel estimation in (2.78) can be rewritten as:

$$
\underline{\hat{p}}_h(n) = \arg\min_{\underline{p}_h} \sum_{k=1}^{n} \beta(n,k) \parallel y(k) - \underline{p}_h^H \underline{x}_p(k) \parallel^2 \tag{2.81}
$$

where the transmitted data vector $\underline{x}_p(n)$ is also white in nature as shown in (2.80). This modified formulation for the least-squares channel estimation problem is identical to the conventional formulation in (2.17). Thus the same methodology can used to evaluate the tracking performance of these modified uncorrelated parameter SW-RLS and the EW-RLS algorithms.

2.8.1 Uncorrelated parameter SW-RLS algorithm

Based on the uncorrelated parameter least-squares formulation in (2.81), the channel estimate obtained using an SW-RLS algorithm is :

$$\underline{\hat{h}}_p(n) = \mathbf{R}^{-1}_{x_p x_p}(n)\mathbf{R}_{x_p y}(n) \tag{2.82}$$

where the matrix $\mathbf{R}_{x_p x_p}(n)$ is given by:

$$\begin{aligned}\mathbf{R}_{x_p x_p}(n) &= \frac{1}{M}\sum_{i=n-M+1}^{n} \underline{x}_p(i)\underline{x}_p^H(i) \\ &= \frac{1}{M}\sum_{i=n-M+1}^{n} \mathbf{U}^H\underline{x}(i)\underline{x}_p^H(i)\mathbf{U} \\ &= \frac{1}{M}\mathbf{U}^H\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)\underline{x}^H(i)\Big)\mathbf{U}\end{aligned} \tag{2.83}$$

while the matrix $\mathbf{R}_{x_p y}(n)$ is given by:

$$\begin{aligned}\mathbf{R}_{x_p y}(n) &= \frac{1}{M}\sum_{i=n-M+1}^{n} \underline{x}_p(i)y^*(i) \\ &= \frac{1}{M}\sum_{i=n-M+1}^{n} \mathbf{U}^H\underline{x}(i)y^*(i) \\ &= \frac{1}{M}\mathbf{U}^H\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)y^*(i)\Big)\end{aligned} \tag{2.84}$$

The SW-RLS estimate can thus be expressed as in terms of $\mathbf{R}_{x_p x_p}(n)$ and $\mathbf{R}_{x_p y}(n)$ as:

$$\begin{aligned}\underline{\hat{h}}_p(n) &= \Big(\mathbf{U}^H\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)\underline{x}^H(i)\Big)\mathbf{U}\Big)^{-1}\mathbf{U}^H\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)y^*(i)\Big) \\ &= \mathbf{U}^H\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)\underline{x}^H(i)\Big)^{-1}\underbrace{\mathbf{U}\mathbf{U}^H}_{\mathbf{I}}\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)y^*(i)\Big) \\ &= \mathbf{U}^H\underbrace{\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)\underline{x}^H(i)\Big)^{-1}}_{\mathbf{R}_{xx}(n)}\underbrace{\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)y^*(i)\Big)}_{\mathbf{R}_{xy}(n)}\end{aligned} \tag{2.85}$$

The underbraced terms in (2.85) can be replaced by the terms $\mathbf{R}_{xx}(n)$ and $\mathbf{R}_{xy}(n)$ obtained from the conventional solution in (2.22) and (2.23) so that:

$$\underline{\hat{h}}_p(n) = \mathbf{U}^H \mathbf{R}_{xx}(n)^{-1} \mathbf{R}_{xy}(n) \tag{2.86}$$

which can be further simplified using (2.21) to:

$$\underline{\hat{h}}_p(n) = \mathbf{U}^H \underline{\hat{h}}_{conv}(n) \tag{2.87}$$

where $\underline{\hat{h}}_{conv}(n)$ is the conventional SW-RLS channel estimate from (2.21). The estimated channel impulse response is obtained from the projected channel impulse response using (2.77) and (2.87) as:

$$\begin{aligned} \underline{\hat{h}}(n) &= \mathbf{U}\underline{\hat{h}}_p(n) \\ &= \mathbf{U}\mathbf{U}^H \underline{\hat{h}}_{conv}(n) \\ &= \underline{\hat{h}}_{conv}(n) \end{aligned} \tag{2.88}$$

Thus a channel estimate produced using a uncorrelated parameter SW-RLS algorithm is equivalent, except for a unitary, invertible transformation $\mathbf{U}$, to a channel estimate produced using the conventional conventional SW-RLS algorithm. The tracking performance of such a uncorrelated parameter algorithm is also expected to be identical to that of the conventional SW-RLS algorithm as as expressed in (2.74) where λ is given by (2.26) as:

$$\lambda = 1 - \frac{2}{M - N - 1} \tag{2.89}$$

The following section considers an analogous uncorrelated parameter EW-RLS algorithm and examines its tracking performance to determine if knowledge of the eigenvectors can be used to improve its performance.

2.8.2 Uncorrelated parameter EW-RLS algorithm

Based on the uncorrelated parameter least squares formulation in (2.81), a channel estimate obtained using an EW-RLS algorithm can be written as:

$$\underline{\hat{p}}_h(n) = \underline{\hat{p}}_h(n-1) + \underline{\mathbf{K}}_p(n)\xi_p^*(n \mid n-1) \tag{2.90}$$

where $\xi_p(n \mid n-1)$, analogous to the prediction error in (2.4) is given by:

$$\xi_p(n \mid n-1) = y(n) - \underline{\hat{p}}_h^H(n-1)\underline{x}_p(n) \tag{2.91}$$

and:

$$\begin{aligned}
\underline{\mathbf{K}}_p(n) &= \left(\sum_{i=1}^{n}\lambda^{n-i}\underline{x}_p(i)\underline{x}_p^H(i)\right)^{-1}\underline{x}_p(n) \\
&= \left(\sum_{i=1}^{n}\lambda^{n-i}\mathbf{U}^H\underline{x}(i)\underline{x}^H(i)\mathbf{U}\right)^{-1}\mathbf{U}^H\underline{x}(n) \\
&= \left(\mathbf{U}^H\Big(\sum_{i=1}^{n}\lambda^{n-i}\underline{x}(i)\underline{x}^H(i)\Big)\mathbf{U}\right)^{-1}\mathbf{U}^H\underline{x}(n)
\end{aligned} \tag{2.92}$$

is the equivalent projected adaptation gain vector. The conventional adaptation gain vector $\underline{\mathbf{K}}(n)$ is given by (2.29). Based on (2.92), and using the relationship $(\mathbf{UAU}^H)^{-1} = \mathbf{UA}^{-1}\mathbf{U}^H$, the projected adaptation gain vector $\underline{\mathbf{K}}_p(n)$ can be expressed in terms of the conventional adaptation gain vector $\underline{\mathbf{K}}(n)$ as:

$$\begin{aligned}
\underline{\mathbf{K}}_p(n) &= \left(\mathbf{U}^H\Big(\sum_{i=1}^{n}\lambda^{n-i}\underline{x}(n)\underline{x}^H(n)\Big)\mathbf{U}\right)^{-1}\mathbf{U}^H\underline{x}(n) \\
&= \mathbf{U}^H\left(\sum_{i=1}^{n}\lambda^{n-i}\underline{x}(n)\underline{x}^H(n)\right)^{-1}\underbrace{\mathbf{UU}^H}_{\mathbf{I}}\underline{x}(n) \\
&= \mathbf{U}^H\left(\sum_{i=1}^{n}\lambda^{n-i}\underline{x}(n)\underline{x}^H(n)\right)^{-1}\underline{x}(n) \\
&= \mathbf{U}^H\underline{\mathbf{K}}(n)
\end{aligned} \tag{2.93}$$

The projection channel estimate vector $\underline{\hat{p}}_h$ given in (2.90) can thus be rewritten in terms of the conventional adaptation gain vector $\underline{\mathbf{K}}(n)$ using the relationship in (2.93) so that:

$$\underline{\hat{p}}_h(n) = \underline{\hat{p}}_h(n-1) + \mathbf{U}^H \underline{\mathbf{K}}(n) \xi_p^*(n \mid n-1) \tag{2.94}$$

which can be further simplified using (2.28) to:

$$\underline{\hat{h}}_p(n) = \mathbf{U}^H \underline{\hat{h}}_{conv}(n) \tag{2.95}$$

where $\underline{\hat{h}}_{conv}(n)$ is the conventional EW-RLS channel estimate from (2.27). The estimated channel impulse response is obtained from the projected channel impulse response using (2.77) and (2.95) as:

$$\begin{aligned} \underline{\hat{h}}(n) &= \mathbf{U} \underline{\hat{h}}_p(n) \\ &= \mathbf{U}\mathbf{U}^H \underline{\hat{h}}_{conv}(n) \\ &= \underline{\hat{h}}_{conv}(n) \end{aligned} \tag{2.96}$$

Thus a channel estimate produced using a uncorrelated parameter EW-RLS algorithm is equivalent, except for a unitary, invertible transformation $\mathbf{U}$, to a channel estimate produced using the conventional conventional EW-RLS algorithm. The tracking performance of such a uncorrelated parameter algorithm is also expected to be identical to that of the conventional EW-RLS algorithm and is given by (2.74).

2.8.3 Channel subspace post-filtering

For both the SW-RLS and the EW-RLS algorithms, knowledge of the channel subspace eigenvectors does not lead to any performance improvement over their respective conventional variants. In both these cases, however, a uncorrelated parameter MMSE channel subspace filter $\mathbf{F}_p$ can be constructed such:

$$\mathbf{F}_p = \mathbf{R}_{\hat{p}_h \hat{p}_h}^{-1} \mathbf{R}_{\hat{p}_h p_h} \tag{2.97}$$

It was shown in (2.88) and (2.96) that the uncorrelated parameter RLS channel estimate is was related to the conventional RLS channel estimate by a unitary transformation $\mathbf{U}$ so that:

$$\underline{\hat{p}}_h(n) = \mathbf{U}^H \underline{\hat{h}}(n) \tag{2.98}$$

The correlation matrices $\mathbf{R}_{\hat{p}_h\hat{p}_h}$ and $\mathbf{R}_{\hat{p}_h p_h}$ can thus be rewritten in terms of the eigendecomposition of the matrices $\mathbf{R}_{\hat{h}\hat{h}}$ and $\mathbf{R}_{\hat{h}h}$ as:

$$\begin{aligned} \mathbf{R}_{\hat{p}_h\hat{p}_h} &= E[\mathbf{U}^H \underline{\hat{h}}(n) \underline{\hat{h}}^H(n)\mathbf{U}] \\ &= \mathbf{U}^H \mathbf{R}_{\hat{h}\hat{h}} \mathbf{U} \\ &= \mathbf{U}^H \mathbf{U} \Sigma_{\hat{h}\hat{h}} \mathbf{U}^h \mathbf{U} \\ &= \Sigma_{\hat{h}\hat{h}} \end{aligned} \tag{2.99}$$

and

$$\begin{aligned} \mathbf{R}_{\hat{p}_h p_h} &= E[= \mathbf{U}^H \underline{\hat{h}}(n) \underline{h}^H(n)\mathbf{U}] \\ &= \mathbf{U}^H \mathbf{R}_{\hat{h}h} \mathbf{U} \\ &= \mathbf{U}^H \mathbf{U} \Sigma_{\hat{h}h} \mathbf{U}^h \mathbf{U} \\ &= \Sigma_{\hat{h}h} \end{aligned} \tag{2.100}$$

so that the diagonal CSF post-filter $\mathbf{F}_p$ is given by:

$$\begin{aligned} \mathbf{F}_p &= \Sigma_{\hat{h}\hat{h}}^{-1} \Sigma_{\hat{h}h} \\ &= \mathbf{I} \Sigma_{f_p f_p} \mathbf{I}^H \end{aligned} \tag{2.101}$$

and whose diagonal coefficients are still given by (2.70). The tracking performance of the post-filtered uncorrelated parameter channel estimates is also identical to the performance of the post-filtered conventional channel estimates and is given by (2.73). Thus, the uncorrelated parameter RLS channel estimates can be post-filtered using an an equivalent CSF $\mathbf{F}_p$ resulting in the same improvement in performance as the conventional case. In other words, *merely decorrelating the parameters that are being tracked does not result in any performance improvement for either the RLS or the*

CSF algorithms.

The following section considers another special case corresponding to low rank channels whose channel correlation matrix is not full rank. A uncorrelated parameter reduced rank RLS algorithm is formulated that exploits the knowledge of the reduced rank eigenvectors of such a channel. The tracking performance before and after CSF post-filtering is evaluated for both the SW-RLS and the EW-RLS algorithms.

2.9 Reduced rank uncorrelated parameter least-squares tracking

Consider the same first-order model for the time-varying system given in (2.1):

$$\underline{h}(n+1) = \alpha \underline{h}(n) + \underline{v}(n+1) \tag{2.102}$$

Let the columns of $\mathbf{U}$ be the eigenvectors of the channel correlation matrix $\mathbf{R}_{hh}$, and let the channel correlation matrix have rank $r < N$ where N is the number of taps in the tapped delay line representation of the time-varying channel impulse response $\underline{h}(n)$. The matrix $\mathbf{U}$ can be partitioned as:

$$\mathbf{U} = \left(\mathbf{U}_r \mid \mathbf{U}_n\right) \tag{2.103}$$

where $\mathbf{U}_r$ represents the channel subspace eigenvectors and $\mathbf{U}_n$ represents the channel null-space eigenvectors. If the eigenvectors represented by $\mathbf{U}$ and the rank r of the channel correlation matrix were known apriori, then the projections onto the subspaces given by $\underline{p}_h(n) = \mathbf{U}_r^H \underline{h}(n)$ obey the same first order model such that:

$$\underline{p}_h(n+1) = \alpha \underline{p}_h(n) + \underline{p}_v(n+1) \tag{2.104}$$

where $\underline{p}_v(n) = \mathbf{U}_r^H \underline{v}(n)$ is the projection of the driving process $\underline{v}(n)$ onto the reduced rank channel subspace defined by the eigenvectors $\mathbf{U}_r$. Correspondingly, the true channel impulse response is related to the projected channel impulse response by the

expression:

$$\underline{h}(n) = \mathbf{U}_r \underline{p}_h(n) \tag{2.105}$$

From (2.50), it is apparent that the eigenvectors of the channel correlation matrix are also the eigenvectors of the process correlation matrix $\mathbf{R}_{vv}$. The reduced rank channel correlation matrix $\mathbf{R}_{hh}$ can be expressed in terms of the eigenvectors $\mathbf{U}$ as:

$$\begin{aligned} \mathbf{R}_{hh} &= \begin{pmatrix} \mathbf{U}_r & \mathbf{U}_n \end{pmatrix} \begin{pmatrix} \Sigma_{h_r h_r} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{U}_r^H \\ \mathbf{U}_n^H \end{pmatrix} \\ &= \mathbf{U}_r \Sigma_{h_r h_r} \mathbf{U}_r^H \end{aligned} \tag{2.106}$$

where $\Sigma_{h_r h_r}$ is a full rank diagonal $r \times r$ matrix. The projected channel correlation matrix $\mathbf{R}_{p_h p_h} = \Sigma_{h_r h_r}$ is an $r \times r$ diagonal matrix as well. The different "taps" of $\underline{p}_h(n)$ are now independent and hence uncorrelated. The $r \times r$ identity matrix represents the eigenvectors of this "new" process such that $\mathbf{U}_p = \mathbf{I}_{r \times r}$. Given the knowledge of the reduced rank eigenvectors $\mathbf{U}_r$, an uncorrelated parameter least squares algorithm can be formulated using (2.17) so that:

$$\underline{\hat{p}}_h(n) = \arg\min_{\underline{p}_h} \sum_{k=1}^{n} \beta(n,k) \parallel \xi_h(k) \parallel^2 \tag{2.107}$$

where apriori knowledge of the subspace structure is exploited by expressing $\xi_h(n)$ in (2.107) as :

$$\begin{aligned} \xi_h(n) &= y(n) - \underline{p}_h^H \mathbf{U}_r^H \underline{x}(n) \\ &= y(n) - \underline{p}_h^H \underline{x}_p(n) \end{aligned} \tag{2.108}$$

where $\underline{x}_p(n) = \mathbf{U}_r^H \underline{x}(n)$ is the projection of the data vector onto the reduced rank eigenvectors represented by $\mathbf{U}_r$. This "new" data vector $\underline{x}_p(n)$ is also white in nature since:

$$
\begin{aligned}
E[\underline{x}_p(n)\underline{x}_p^H(n)] &= E[\mathbf{U}_r^H \underline{x}(n)\underline{x}^H(n)\mathbf{U}_r] \\
&= \mathbf{U}_r^H E[\underline{x}(n)\underline{x}^H(n)]\mathbf{U}_r \\
&= \mathbf{U}_r^H \mathbf{I} \mathbf{U}_r \\
&= \mathbf{U}_r^H \mathbf{U}_r = \mathbf{I}_{r\times r}
\end{aligned}
\tag{2.109}
$$

Using (2.108), the least-squares problem of channel estimation in (2.107) can be rewritten as:

$$
\underline{\hat{p}}_h(n) = \arg\min_{\underline{p}_h} \sum_{k=1}^{n} \beta(n,k) \parallel y(k) - \underline{p}_h^H \underline{x}_p(k) \parallel^2 \tag{2.110}
$$

where the transmitted data vector $\underline{x}_p(n)$ is also white in nature as shown in (2.109). This modified formulation for the least-squares channel estimation problem is similar to the original conventional formulation in (2.17). Thus the same methodology can used to evaluate the tracking performance of these modified uncorrelated parameter SW-RLS and the EW-RLS algorithms.

2.9.1 Reduced rank uncorrelated parameter SW-RLS algorithm

Based on the uncorrelated parameter least-squares formulation in (2.110), the channel estimate obtained using an SW-RLS algorithm is given by:

$$
\underline{\hat{h}}_p(n) = \mathbf{R}_{x_p x_p}^{-1}(n)\mathbf{R}_{x_p y}(n) \tag{2.111}
$$

where the matrix $\mathbf{R}_{x_p x_p}(n)$ is given by:

$$\begin{aligned}\mathbf{R}_{x_p x_p}(n) &= \frac{1}{M}\sum_{i=n-M+1}^{n} \underline{x}_p(i)\underline{x}_p^H(i) \\ &= \frac{1}{M}\sum_{i=n-M+1}^{n} \mathbf{U}_r^H \underline{x}(i)\underline{x}_p^H(i)\mathbf{U}_r \\ &= \frac{1}{M}\mathbf{U}_r^H\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)\underline{x}^H(i)\Big)\mathbf{U}_r\end{aligned} \tag{2.112}$$

while the matrix $\mathbf{R}_{x_p y}(n)$ is given by:

$$\begin{aligned}\mathbf{R}_{x_p y}(n) &= \frac{1}{M}\sum_{i=n-M+1}^{n} \underline{x}_p(i)y^*(i) \\ &= \frac{1}{M}\sum_{i=n-M+1}^{n} \mathbf{U}_r^H \underline{x}(i)y^*(i) \\ &= \frac{1}{M}\mathbf{U}_r^H\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)y^*(i)\Big)\end{aligned} \tag{2.113}$$

The SW-RLS estimate can thus be expressed in terms of $\mathbf{R}_{x_p x_p}(n)$ and $\mathbf{R}_{x_p y}(n)$ as:

$$\underline{\hat{h}}_p(n) = \Big(\mathbf{U}_r^H\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)\underline{x}^H(i)\Big)\mathbf{U}_r\Big)^{-1}\mathbf{U}_r^H\Big(\sum_{i=n-M+1}^{n} \underline{x}(i)y^*(i)\Big) \tag{2.114}$$

Unlike the uncorrelated parameter case in the previous section, it is not possible to express the SW-RLS estimate in (2.114) in terms of $\mathbf{R}_{xx}(n)$ and $\mathbf{R}_{xy}(n)$ obtained from the conventional solution in (2.22) and (2.23). The tracking performance of such an uncorrelated parameter algorithm is expected to be the same as that of a directly constrained reduced rank EW-RLS algorithm for which the tracking parameter λ is given using (2.26) as:

$$\lambda_{rr} = 1 - \frac{2}{M-r-1} \tag{2.115}$$

where the "new" tap weight vector $\underline{p}_h(n)$ is an $r \times 1$ column vector. The following section considers the uncorrelated parameter reduced rank EW-RLS algorithm and examines its tracking performance to determine if knowledge of the reduced rank

eigenvectors can be used to improve the performance of the RLS algorithms.

2.9.2 Reduced rank uncorrelated parameter EW-RLS algorithm

Based on the uncorrelated parameter least squares formulation in (2.110), a channel estimate obtained using an EW-RLS algorithm is given by:

$$\underline{\hat{p}}_h(n) = \underline{\hat{p}}_h(n-1) + \underline{\mathbf{K}}_p(n)\xi_p^*(n \mid n-1) \tag{2.116}$$

where $\xi_p(n \mid n-1)$, analogous to the prediction error in (2.4) is given by:

$$\xi_p(n \mid n-1) = y(n) - \underline{\hat{p}}_h^H(n-1)\underline{x}_p(n) \tag{2.117}$$

and:

$$\begin{aligned}
\underline{\mathbf{K}}_p(n) &= \left(\sum_{i=1}^{n} \lambda^{n-i}\underline{x}_p(i)\underline{x}_p^H(i)\right)^{-1}\underline{x}_p(n) \\
&= \left(\sum_{i=1}^{n} \lambda^{n-i}\mathbf{U}_r^H\underline{x}(i)\underline{x}^H(i)\mathbf{U}_r\right)^{-1}\mathbf{U}_r^H\underline{x}(n) \\
&= \left(\mathbf{U}_r^H\left(\sum_{i=1}^{n} \lambda^{n-i}\underline{x}(i)\underline{x}^H(i)\right)\mathbf{U}_r\right)^{-1}\mathbf{U}_r^H\underline{x}(n)
\end{aligned} \tag{2.118}$$

is the equivalent projected adaptation gain vector. The conventional adaptation gain vector $\underline{\mathbf{K}}(n)$ is given by (2.29). Based on (2.118), and since $(\mathbf{U}_r\mathbf{A}\mathbf{U}_r^H)^{-1} \neq \mathbf{U}_r\mathbf{A}^{-1}\mathbf{U}_r^H$ for $r < N$, the projected adaptation gain vector $\underline{\mathbf{K}}_p(n)$ cannot be expressed in terms of the conventional adaptation gain vector $\underline{\mathbf{K}}(n)$. It is therefore not possible, in this scenario, to find a linear relationship between the reduced rank channel estimate and the conventional channel estimate. Nonetheless, it is still possible to evaluate the tracking performance of the directly constrained reduced rank EW-RLS algorithm with the help of the same techniques used to evaluate the tracking performance of the conventional EW-RLS algorithm. The projected reduced rank channel estimation

error vector is given by:

$$\underline{\epsilon}_p(n \mid n-1) = \underline{p}_h(n) - \underline{\hat{p}}_h(n-1) \tag{2.119}$$

and the $r \times r$ error correlation matrix $\mathbf{R}_{\epsilon_p\epsilon_p}$ can be computed similarly as:

$$\mathbf{R}_{\epsilon_p\epsilon_p} = \underbrace{\frac{2}{(1+\alpha)(1+\lambda)(1-\alpha\lambda)}\mathbf{R}_{v_pv_p}}_{\mathbf{D}_1} + \underbrace{\frac{(1-\lambda)}{(1+\lambda)}\sigma^2\mathbf{R}^{-1}_{x_px_p}}_{\mathbf{D}_2} \tag{2.120}$$

The corresponding $N \times N$ error correlation matrix for the conventional EW-RLS algorithm in (2.56) has been rewritten here for comparison:

$$\mathbf{R}_{\epsilon\epsilon} = \underbrace{\frac{2}{(1+\alpha)(1+\lambda)(1-\alpha\lambda)}\mathbf{R}_{vv}}_{\mathbf{D}_1} + \underbrace{\frac{(1-\lambda)}{(1+\lambda)}\sigma^2\mathbf{R}^{-1}_{xx}}_{\mathbf{D}_2} \tag{2.121}$$

For these EW-RLS algorithms, the eigenvalues of the error correlation matrix are given by:

$$\Sigma_{\epsilon_p\epsilon_p} = \underbrace{\frac{2}{(1+\alpha)(1+\lambda)(1-\alpha\lambda)}\Sigma_{p_vp_v}}_{\mathbf{D}_1} + \underbrace{\frac{(1-\lambda)}{(1+\lambda)}\sigma^2\mathbf{I}^{-1}_{r\times r}}_{\mathbf{D}_2} \tag{2.122}$$

$$\Sigma_{\epsilon\epsilon} = \underbrace{\frac{2}{(1+\alpha)(1+\lambda)(1-\alpha\lambda)}\Sigma_{vv}}_{\mathbf{D}_1} + \underbrace{\frac{(1-\lambda)}{(1+\lambda)}\sigma^2\mathbf{I}^{-1}_{n\times n}}_{\mathbf{D}_2} \tag{2.123}$$

where the substitutions $\mathbf{R}_{x_px_p} = \mathbf{I}_{r\times r}$ and $\mathbf{R}_{xx} = \mathbf{U}\mathbf{I}_{n\times n}\mathbf{U}^H$ have been made in simplifying the above expressions. For the low-rank channel:

$$\Sigma_{vv} = \begin{pmatrix} \Sigma_{p_vp_v} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \tag{2.124}$$

(2.122) and (2.123) demonstrate the improvement in tracking performance due to reduced rank uncorrelated parameter least squares tracking. The diagonal elements of $\Sigma_{\epsilon_p\epsilon_p}$ and $\Sigma_{\epsilon\epsilon}$ represent the error associated with tracking a particular subspace.

For the low-rank channel, the conventional RLS algorithm incurs a penalty of $(N-r)\sigma^2(1-\lambda)/(1+\lambda)$ due to the implicit tracking of the null subspaces. The uncorrelated parameter reduced rank RLS algorithm avoids this penalty. At the same time, however, the error associated with tracking a particular non-null subspace remains the same. Based on the earlier discussion of the CSF algorithm as an MMSE post-filtering, this seems to suggest that the CSF of the conventional EW-RLS channel estimates could perhaps yield comparable performance compared to the uncorrelated parameter reduced rank EW-RLS algorithm. It is important to recognize that this result holds as long as the EW-RLS or the SW-RLS algorithm is operating in the range of λ or M respectively, where the direct averaging method is applicable for the original N dimensional problem. The following section examines the tracking performance of the CSF algorithm.

2.9.3 Channel subspace post-filtering

For both the SW-RLS and the EW-RLS algorithms, knowledge of the reduced rank channel subspace eigenvectors leads to an improvement in performance with respect to their conventional variants. This improvement is attributed to the fact that the penalty incurred in tracking the null subspaces is eliminated. Nonetheless, the directly constrained reduced rank channel estimate can still be treated as a noisy time series. In both these cases, an uncorrelated parameter MMSE channel subspace filter $\mathbf{F}_p$ can be constructed such that:

$$\mathbf{F}_p = \mathbf{R}_{\hat{p}_h\hat{p}_h}^{-1}\mathbf{R}_{\hat{p}_h p_h} \tag{2.125}$$

The correlation matrices $\mathbf{R}_{\hat{p}_h\hat{p}_h}$ and $\mathbf{R}_{\hat{p}_h p_h}$ can be rewritten in terms of the eigendecomposition of the matrices $\mathbf{R}_{\hat{h}\hat{h}}$ and $\mathbf{R}_{\hat{h}h}$ as:

$$\begin{aligned}
\mathbf{R}_{\hat{p}_h\hat{p}_h} &= E[\mathbf{U}_r^H\underline{\hat{h}}(n)\underline{\hat{h}}^H(n)\mathbf{U}_r] \\
&= \mathbf{U}_r^H\mathbf{R}_{\hat{h}\hat{h}}\mathbf{U}_r \\
&= \mathbf{U}_r^H\mathbf{U}\Sigma_{\hat{h}\hat{h}}\mathbf{U}^h\mathbf{U}_r \\
&= \Sigma_{\hat{h}_r\hat{h}_r}
\end{aligned} \tag{2.126}$$

and

$$\begin{aligned}\mathbf{R}_{\hat{p}_h p_h} &= E[\mathbf{U}_r^H \hat{\underline{h}}(n) \underline{h}^H(n) \mathbf{U}_r] \\ &= \mathbf{U}_r^H \mathbf{R}_{\hat{h}h} \mathbf{U}_r \\ &= \mathbf{U}_r^H \mathbf{U} \Sigma_{\hat{h}h} \mathbf{U}^h \mathbf{U}_r \\ &= \Sigma_{\hat{h}_r h_r}\end{aligned} \tag{2.127}$$

where $\Sigma_{\hat{h}_r \hat{h}_r}$ and $\Sigma_{\hat{h}_r h_r}$ are the $r \times r$ diagonal sub matrices of $\Sigma_{\hat{h}\hat{h}}$ and $\Sigma_{\hat{h}h}$ expressed respectively as:

$$\Sigma_{\hat{h}\hat{h}} = \begin{pmatrix} \Sigma_{\hat{h}_r \hat{h}_r} & \mathbf{0} \\ \mathbf{0} & \sigma^2 \frac{(1-\lambda)}{(1+\lambda)} \mathbf{I}_{N-r} \end{pmatrix} \tag{2.128}$$

$$\Sigma_{\hat{h}h} = \begin{pmatrix} \Sigma_{\hat{h}_r h_r} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \tag{2.129}$$

The diagonal reduced rank CSF post-filter $\mathbf{F}_p$ is given by:

$$\begin{aligned}\mathbf{F}_p &= \Sigma_{\hat{p}_h \hat{p}_h}^{-1} \Sigma_{\hat{p}_h p_h} \\ &= \Sigma_{\hat{h}_r \hat{h}_r}^{-1} \Sigma_{\hat{h}_r h_r} \\ &= \mathbf{I}_r \Sigma_{f_p f_p} \mathbf{I}_r^H\end{aligned} \tag{2.130}$$

whose diagonal coefficients are written in a form similar to (2.70) so that:

$$\begin{aligned}\sigma_{f_p i}^2 &= \frac{\alpha {\sigma_{h_p i}}^2}{\frac{(1+\alpha\lambda)}{(1+\lambda)} {\sigma_{h_p i}}^2 + \frac{(1-\alpha\lambda)}{(1+\lambda)} \sigma^2} \\ &= \frac{\alpha {\sigma_{hi}}^2}{\frac{(1+\alpha\lambda)}{(1+\lambda)} {\sigma_{hi}}^2 + \frac{(1-\alpha\lambda)}{(1+\lambda)} \sigma^2}\end{aligned} \tag{2.131}$$

for $i = 1, 2, \ldots, r$. The CSF post-filter $\mathbf{F}$ for the conventional EW-RLS algorithm had been computed earlier as:

$$\mathbf{F} = \mathbf{U} \Sigma_{ff} \mathbf{U}^H \tag{2.132}$$

where the coefficients of the diagonal matrix Σ_f as expressed in (2.70) are given by:

$$\sigma_{fi}^2 = \frac{\alpha {\sigma_{hi}}^2}{\frac{(1+\alpha\lambda)}{(1+\lambda)}{\sigma_{hi}}^2 + \frac{(1-\alpha\lambda)}{(1+\lambda)}\sigma^2} \tag{2.133}$$

for $i = 1, 2, \ldots, N$. If $r < N$ is the rank of the channel correlation matrix, $\sigma_{h_i}^2 = 0$ for $i > r$. The reduced rank CSF coefficients can be written in terms of the conventional CSF coefficients using (2.131) as:

$$\sigma_{fi}^2 = \begin{cases} \sigma_{f_p i}^2 & \text{for } i = 1, 2, \ldots r \\ 0 & \text{for } i = r+1, \ldots, N \end{cases} \tag{2.134}$$

Even though the conventional EW-RLS algorithm implicitly tracks the null-subspaces of the low-rank channel, CSF of the conventional channel estimates eliminates the penalty associated with tracking these subspaces. This is readily apparent when the expressions for the tracking performance of both algorithms after CSF are compared:

$$\sigma_{\epsilon f_p i}^2 = \sigma_{h_r i}^2 \left(1 - \frac{\frac{\alpha^2(1-\lambda)^2\sigma_{h_r i}^2}{(1-\alpha\lambda)^2}}{\frac{(1-\lambda)(1+\alpha\lambda)\sigma_{h_r i}^2}{(1-\alpha\lambda)(1+\lambda)} + \frac{1-\lambda}{1+\lambda}\sigma^2}\right) \qquad \text{for } i = 1, 2, \ldots, r \tag{2.135}$$

$$\sigma_{\epsilon f i}^2 = \sigma_{h_r i}^2 \left(1 - \frac{\frac{\alpha^2(1-\lambda)^2\sigma_{h_r i}^2}{(1-\alpha\lambda)^2}}{\frac{(1-\lambda)(1+\alpha\lambda)\sigma_{h_r i}^2}{(1-\alpha\lambda)(1+\lambda)} + \frac{1-\lambda}{1+\lambda}\sigma^2}\right) \qquad \text{for } i = 1, 2, \ldots, N \tag{2.136}$$

These expressions indicate that the performance of the directly constrained EW-RLS algorithm can be improved by CSF. Furthermore, it also indicates, that *CSF of the conventional EW-RLS algorithm is canonically equivalent, performance wise, to CSF of the uncorrelated parameter reduced rank EW-RLS algorithm*, to within the limits of the direct averaging method used to derive these expressions. This is a rather surprising result with rather significant implications. In a time-varying environment with stationary statistics, this result suggests that due to CSF, additional effort invested in apriori estimation of the reduced rank eigenvectors of the channel subspace and a reformulation of the RLS algorithms *does not result in any performance ben-*

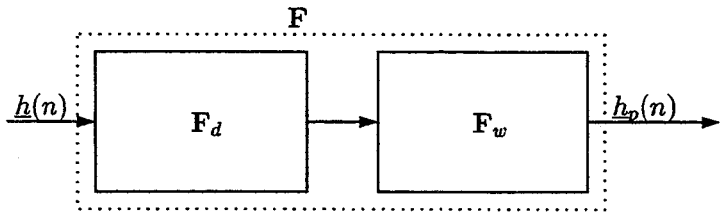

Figure 2-5: CSF as a cascade of two filters

efits to within the limits of the direct averaging method. Throughout this thesis, the impact of the assumptions used, when invoking the direct averaging method in the derivations presented earlier, on the validity of the expressions for the tracking performance of an EW-RLS algorithms is acknowledged and implicit in any such statement.

For the SW-RLS algorithm, however, there is a performance improvement due to the use of the uncorrelated parameter reduced rank algorithm. This improvement can be attributed to the difference in the number of taps used to represent the tap weight vector for these algorithms. The channel estimate vector for an conventional SW-RLS algorithm is a $N \times 1$ vector while the projected channel estimate vector for an uncorrelated parameter reduced rank SW-RLS algorithm is a $r \times 1$ vector. The performance the conventional and reduced rank SW-RLS algorithms is equivalent to that of an EW-RLS algorithm with a tracking parameter given accordingly by λ_{conv} and λ_{rr} respectively where based on (2.26) and (2.115):

$$\lambda_{conv} = 1 - \frac{2}{M - N - 1} \tag{2.137}$$

$$\lambda_{rr} = 1 - \frac{2}{M - r - 1} \tag{2.138}$$

Since $r < N$, it implies that $\lambda_{conv} < \lambda_{rr}$. This leads to an increase in the observation noise induced error variance $\mathcal{D}_2$ resulting in the reduced rank uncorrelated parameter SW-RLS having a lower channel estimation error than the conventional SW-RLS algorithm. CSF post-filtering still improves the tracking performance of estimates produced using either of these algorithms and the improvement is still given by (2.135) which was shown to be equal to (2.136).

2.10 CSF paradigm in low rank channel

For low rank channels, based on the above discussion, the channel subspace filter $\mathbf{F}$, as represented in figure 2-5, can, for analytical purposes, be treated as a cascade of two filters $\mathbf{F}_d$ and $\mathbf{F}_w$. The dimensionality reduction filter $\mathbf{F}_d$ is given by:

$$\mathbf{F}_d = \begin{pmatrix} \mathbf{U}_r & \mathbf{U}_n \end{pmatrix} \begin{pmatrix} \mathbf{I}_{r\times r} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{U}_r^H \\ \mathbf{U}_n^H \end{pmatrix} \tag{2.139}$$

and may be equivalently expressed as:

$$\mathbf{F}_d = \mathbf{U}_r \mathbf{U}_r^H \tag{2.140}$$

The dimensionality reduction filter $\mathbf{F}_d$ eliminates the penalty, reflected in equation (2.74), incurred by RLS algorithms due to the implicit tracking of null channel subspaces. If only the reduced dimensionality of the process were exploited, i.e. only $\mathbf{F}_d$ were applied, the resulting performance would be the same as that of uncorrelated parameter reduced rank RLS algorithm. The additional filter, which is the Wiener channel subspace filter $\mathbf{F}_w$, treats the RLS channel estimates as a noisy time series and exploits the correlation between the channel estimation error and the true channel impulse response so that:

$$\mathbf{F}_w = \mathbf{U}_r \Sigma_{f_p f_p} \mathbf{U}_r^H \tag{2.141}$$

where Σ_{f_p} is a diagonal matrix representing the channel subspace filtering coefficients of the non-null channel subspaces and is given by (2.70). Channel subspace filtering (CSF), as represented in figure 2-6, achieves an improvement in performance by exploiting *both* the low rank nature of the channel as well as the correlation between the estimate and the true channel impulse response.

If a low-rank channel with rank r has independent, uncorrelated taps, then only r taps would be needed in order to adapt to the variations in the observed data. For an conventional algorithm that uses $N > r$ taps to track such a system, this is referred to as over-modeling of the system parameters. If a conventional RLS algorithm were

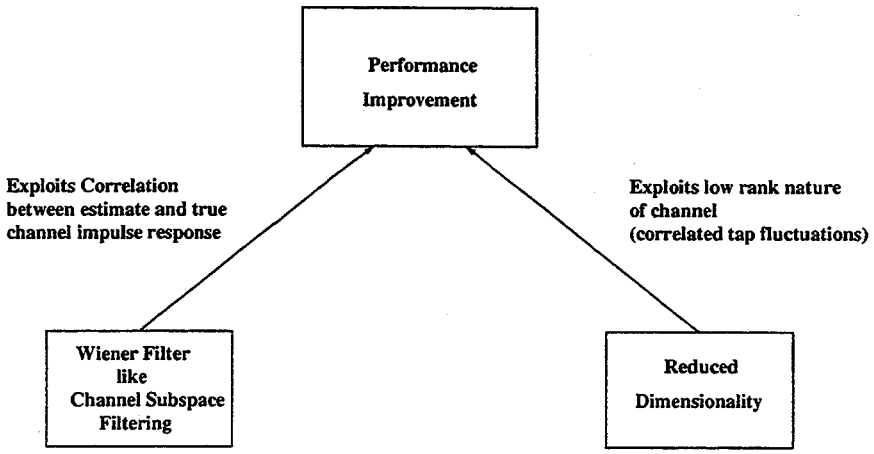

Figure 2-6: Performance improvement due to CSF

used to adapt the coefficients of the channel estimate, this over-modeling would lead to an increased channel estimation error. Based on the discussion earlier, the penalty incurred in tracking these null "taps" is the cause for the increase in the channel estimation error.

With the CSF approach, assuming that the system has stationary statistics, this penalty due to over-modeling errors can be eliminated *and* the correlation between the channel estimation error vector and the true channel impulse response vector can be exploited to considerably reduce the channel estimation error. From the form of (2.63) it seems feasible that the eigenvectors of the process may be estimated from the channel estimate time series. The CSF post-filter could then be constructed based on the knowledge of these eigenvectors thereby reducing, if not eliminating, the penalty incurred due to the implicit tracking of the null channel subspaces. The CSF approach seems to suggest that in a stationary environment, within the limits of the direct averaging method, that *a conventional overmodeled RLS algorithm may perform as well an RLS algorithm that explicitly relies on the precise identification of the reduced rank eigenvectors.*

The following chapter uses simulated data to corroborate the theoretical predictions made in this chapter. The impact of finite received data on the performance of

the CSF algorithm is analyzed using simulated data.

Chapter 3

Performance analysis of CSF algorithm

In the preceding chapter, the concept of the CSF algorithm was introduced in the context of improving the tracking performance of least-squares based adaptive algorithms. Concurrently, theoretical predictions of tracking performance of such algorithms were made using a simplified first-order Gauss-Markov model for the unknown time-varying system. Several simplifying assumptions were made in deriving these analytical expressions. This chapter compares the theoretical performance thus predicted with the performance achieved using simulated data. Subsequently, factors possibly contributing to any mismatch between the theoretical and observed performance are discussed and the relative magnitude of their impact is assessed.

3.1 Simulation methodology

The models for the time-varying system described in (2.1) and (2.2) were used to generate data used in the simulations. Throughout this thesis any reference to simulated data shall, unless specifically mentioned otherwise, refer to data generated in the manner described subsequently. The time-varying channel impulse response vector $\underline{h}(n)$ was modeled using a $N = 35$ tap transversal vector. The eigenvalues of the driving process, i.e. the diagonal elements of Σ_{vv}, determine the energy associated with each

of the channel subspace eigenvectors. The correlation matrix was modeled with a rank of 5. The eigenvalues of the correlation matrix were normalized so that the energy in the driving process was unity, i.e. $tr(\Sigma_{vv}) = 1$. A Gaussian $N \times 1$ vector time series, denoted by $\underline{z}(n)$, was generated using the randn function in MATLAB. This ensured that the samples of the vector as well as the individual elements of the vector were uncorrelated. The energy of the non-null subspaces, i.e. the non-zero eigenvalues of Σ_{vv}, were assigned to be $[0.6056, 0.1831, 0.1217, 0.0608, 0.0303]$. The eigenvectors of the process $\mathbf{U}$ were generated randomly and the vector $\underline{z}(n)$ was shaped using a filter constructed as $\mathbf{U}\Sigma_{vv}^{1/2}\mathbf{U}^H$. The resulting vector corresponds to the driving process vector $\underline{v}(n) = \mathbf{U}\Sigma_{vv}^{1/2}\mathbf{U}^H\underline{z}(n)$ in (2.1). The time-varying system was evolved as (2.1) with the parameter $\alpha = 0.9995$. This resulted in the time-varying channel impulse response $\underline{h}(n)$ having an energy given by (2.50) as $tr(\mathbf{R}_{hh} = tr(\mathbf{R}_{vv})/(1-\alpha^2)$ which for $\alpha = 0.9995$ corresponds to an an energy of about 30 dB.

The output of the system $y(n)$ was generated using (2.2) with additive Gaussian observation noise $w(n)$ having a variance of σ^2corresponding to a range of SNR's from 20 dB to -20 dB. The transmitted data vector $\underline{d}(n) = [d(n), d(n-1), \ldots, d(n-N+1)]^T$ was constructed using equiprobable BPSK symbols where:

$$d(n) = \begin{cases} 1 & \text{with probability } 1/2 \\ -1 & \text{with probability } 1/2 \end{cases} \tag{3.1}$$

so that transmitted data correlation matrix is white i.e. $\mathbf{R}_{xx} = \mathbf{I}$. The procedure described above was used to generate 50,000 samples of the received data $y(n)$ corresponding to an equal number of samples of the true channel impulse response vector $\underline{h}(n)$ and the transmitted data vector $\underline{x}(n)$.

Post-filtering, particularly the methodology for CSF, was developed in the previous chapter using a common theoretical framework for the EW-RLS and SW-RLS algorithms. An equivalence between these two variants of the traditional RLS algorithms was also derived based on an analytical technique formulated by Proakis [27]. As a result, it was claimed that the tracking performance of both the RLS and the

CSF algorithms could be expressed in terms of the tracking parameter λ as given in (2.74) and (2.73). The tracking parameter was parameterized as $\lambda_{eff} = f(\lambda, N, M)$ where λ was the exponential forgetting factor of the EW-RLS algorithm, N was the number of taps in the sampled channel impulse response model, and M was the size of the sliding window in the SW-RLS algorithm. The analytical expressions for the channel estimation error of the RLS and the CSF algorithms were then rewritten in terms of this common tracking parameter λ_{eff} so that:

$$\text{RLS algorithm} : \sigma^2_{\epsilon i} = \frac{2(1-\alpha)}{(1+\lambda_{eff})(1-\alpha\lambda_{eff})}\sigma^2_{hi} + \frac{1-\lambda_{eff}}{1+\lambda_{eff}}\sigma^2 \tag{3.2}$$

$$\text{CSF algorithm} : \sigma^2_{\epsilon fi} = \sigma^2_{hi}\left(1 - \frac{\frac{\alpha^2(1-\lambda_{eff})^2\sigma^2_{hi}}{(1-\alpha\lambda_{eff})^2}}{\frac{(1-\lambda_{eff})(1+\alpha\lambda_{eff})\sigma^2_{hi}}{(1-\alpha\lambda_{eff})(1+\lambda_{eff})} + \frac{1-\lambda_{eff}}{1+\lambda_{eff}}\sigma^2}\right) \tag{3.3}$$

The theoretical curves for these these performance metrics were plotted apriori. Based on these curves, a finite set of tracking parameter values were selected. These values corresponded to samples on the performance curves that were capable of capturing the performance characteristics of the RLS and the CSF algorithms. Once these tracking parameter values were determined, channel estimates $\underline{\hat{h}}(n)$ were generated using the EW-RLS and SW-RLS algorithms. For the conventional and the uncorrelated parameter reduced rank EW-RLS algorithms, the corresponding exponential forgetting factor $\lambda_{EW-RLS} = \lambda_{RREW-RLS}$ used to generate the channel estimates is related to the common tracking parameter by the relationship:

$$\lambda_{EW-RLS} = \lambda_{eff} \tag{3.4}$$

For the conventional SW-RLS algorithm, the sliding window size M_{SW-RLS} used to generate the channel estimates is related to the common tracking parameter by (2.26) here as:

$$M_{SW-RLS} = \frac{2}{1-\lambda_{eff}} + N - 1 \tag{3.5}$$

For the uncorrelated parameter SW-RLS algorithm, the sliding window size $M_{RRSW-RLS}$ used to generate the channel estimates is analogously related to the common tracking

parameter as:

$$M_{RRSW-RLS} = \frac{2}{(1-\lambda_{eff})} + r - 1 \tag{3.6}$$

where $r < N$ is the reduced rank of the channel subspace. Once the channel estimates were generated using these variants of the RLS algorithms, the channel subspace post-filter $\mathbf{F} = \mathbf{U}\Sigma_{ff}\mathbf{U}^H$ was constructed using apriori knowledge of the channel parameters. The diagonal matrix Σ_{ff} represent the channel subspace filtering coefficients in (2.70) and was expressed in terms of the common tracking parameter λ_{eff} as:

$$\sigma_{fi}{}^2 = \frac{\alpha\sigma_{hi}{}^2}{\frac{(1+\alpha\lambda_{eff})}{(1+\lambda_{eff})}\sigma_{hi}{}^2 + \frac{(1-\alpha\lambda_{eff})}{(1+\lambda_{eff})}\sigma^2} \tag{3.7}$$

The adaptive RLS algorithms were assumed to have reached steady state after 8000 iterations, i.e. $n > 8000$. The remaining 42,000 channel estimates were used to estimate the relevant performance metrics namely the channel estimation error and any other metrics that defined in subsequent sections.

The estimated channel estimation errors for the RLS and the CSF algorithms were compared to the analytical expressions for the same in (3.2) and (3.3). Additionally, these estimated performance metrics were also used to corroborate or debunk some of the qualitative predictions made in the previous chapter regarding the tracking performance of the different variants of the RLS algorithms in low rank channels.

3.2 Simulated performance of conventional RLS algorithm

Figure 3-1 compares the tracking performance of the EW-RLS and CSF algorithms with the theoretical predictions in (3.2) and (3.3). There is good agreement between the predicted and computed values of performance for values of λ_{eff} close to 1. At smaller values of λ_{eff} there is a mismatch of around 1 dB between the predicted and observed values.

Figure 3-2 compares the tracking performance of the SW-RLS and CSF algorithms

with the theoretical predictions in (3.2) and (2.73). Unlike the EW-RLS algorithm there is a persistent mismatch between the theoretical and observed values of the channel estimation error. The mismatch is greater at higher values of SNR implying an that the expression for the process noise variance term in (3.2) is incorrect. Comparing figures 3-1 and 3-2 allows a comparison of the tracking performance of the EW-RLS and the SW-RLS algorithms against each other. Since the data points used to generate these figures are parameterized by the common tracking parameter λ_{eff}, an agreement in the computed metrics for the EW-RLS and SW-RLS is expected. There is clearly no agreement indicating that the equivalence between the SW-RLS and the EW-RLS algorithms alluded to so far is imprecise. Henceforth, throughout this thesis, any comparison between subsequently developed analytical expressions and their computed values shall implicitly refer to the EW-RLS algorithm. Nonetheless, it is encouraging to observe that the performance of both the EW-RLS and SW-RLS algorithms can be improved by using the CSF approach.

The tracking performance of the SW-RLS algorithm on simulated data shall still be presented in subsequent sections for the sake of completeness. The theoretical metrics that it will be compared to merely serve to compare its tracking performance to that of the EW-RLS algorithm.

3.3 Simulated performance of reduced rank uncorrelated parameter RLS algorithm

Figure 3-3 compares the tracking performance of the reduced rank uncorrelated EW-RLS and the CSF algorithms with the theoretical predictions in (3.2) and (3.3). The "Th-CSF known basis" plot refers to tracking performance on simulated data of the theoretically computed CSF filter when the basis are known apriori. As can be seen, there is a good match between the theoretically predicted and observed values for the same. There is better agreement at higher SNR's than in the conventional EW-RLS algorithm case. Figure 3-5 compares the tracking performance of the reduced rank

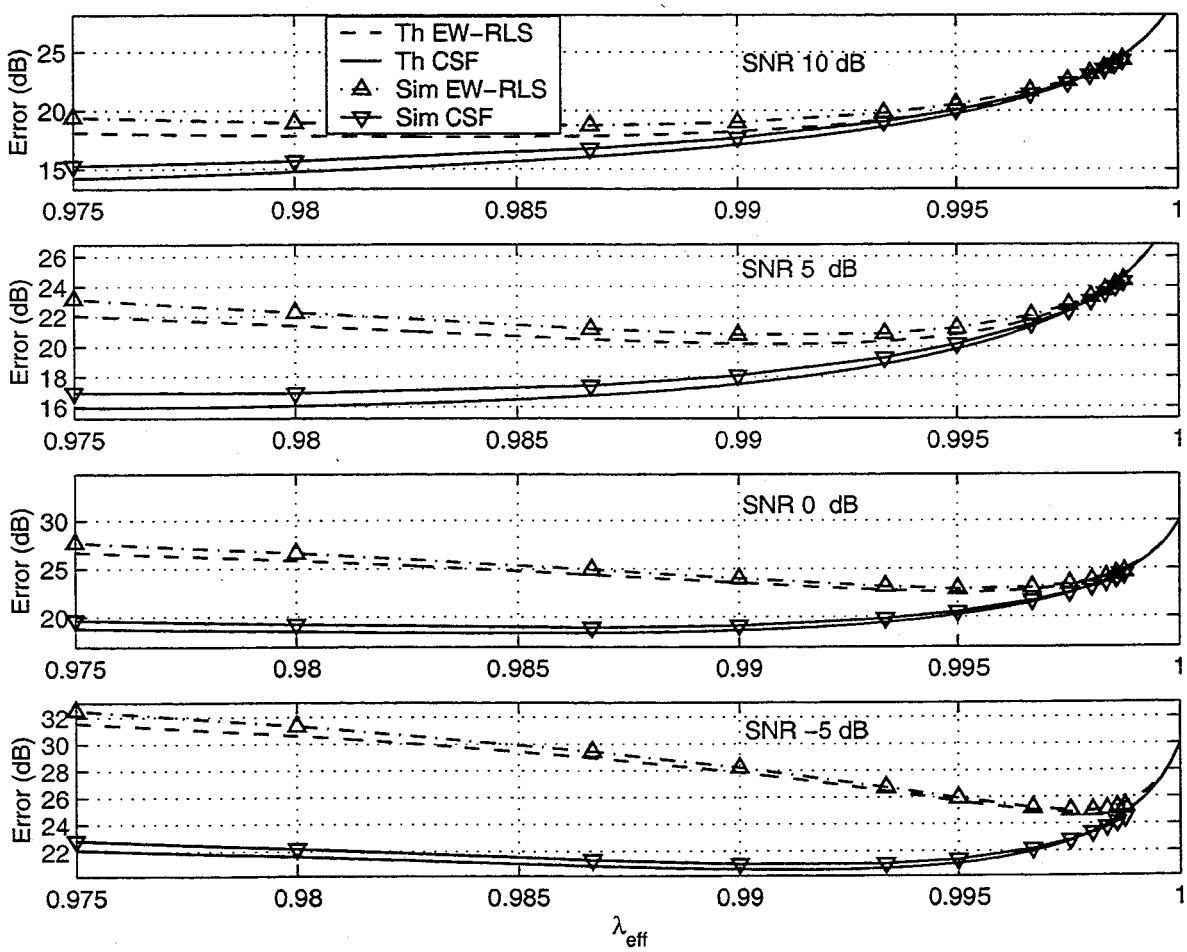

Figure 3-1: Tracking performance of EW-RLS and CSF algorithm on simulated data

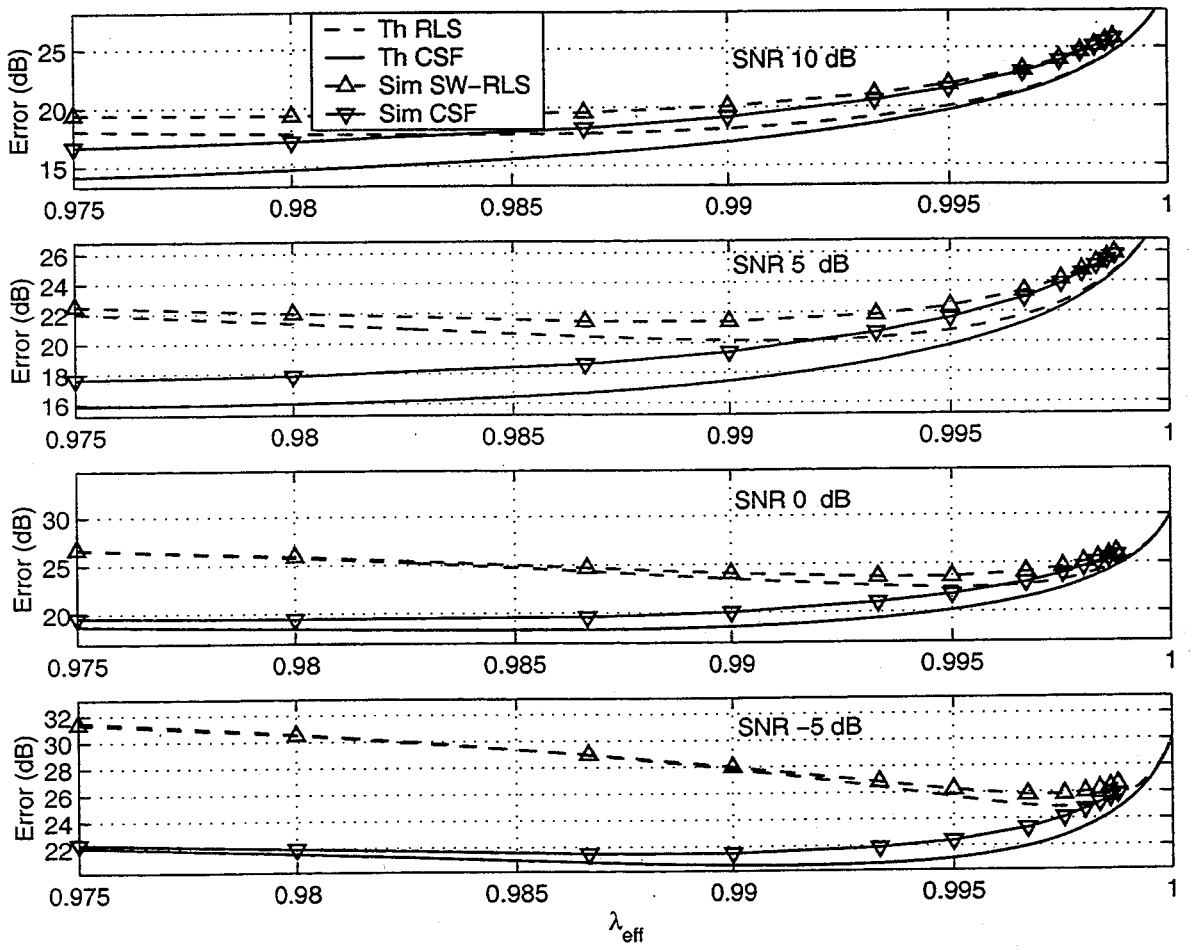

Figure 3-2: Tracking performance of SW-RLS and CSF algorithm on simulated data

uncorrelated parameter algorithm with that of the conventional EW-RLS algorithm. The "ARR conv EW-RLS" plot refers to the tracking performance on simulated data of an abrupt rank reduced conventional EW-RLS algorithm. As can be seen, there is a good agreement between the abrupt rank reduced conventional EW-RLS tracking performance and that of the reduced rank uncorrelated parameter EW-RLS algorithm. There is better agreement, as discussed in the previous chapter, in the tracking performance of the CSF variants of these algorithms.

Figure 3-4 compares the tracking performance of the reduced rank uncorrelated SW-RLS and CSF algorithms with the theoretical predictions in (3.2) and (3.3). The "Th-CSF known basis" plot refers to tracking performance on simulated data of the theoretically computed CSF filter when the basis are known apriori. As can be seen, there is still a mismatch between the theoretically predicted and observed values for the same. The mismatch at higher SNR's is comparable to that of the conventional SW-RLS algorithm case. Figure 3-6 compares the tracking performance of the reduced rank uncorrelated parameter algorithm with that of the conventional SW-RLS algorithm. The "ARR conv SW-RLS" plot refers to the tracking performance on simulated data of an abrupt rank reduced conventional SW-RLS algorithm. As can be seen, there is good agreement between the abrupt rank reduced conventional SW-RLS tracking performance and that of the reduced rank uncorrelated parameter SW-RLS algorithm. There is still, however, a good agreement in the tracking performance of the CSF variants of these algorithms.

For RLS algorithms, these simulations have confirmed that for low rank channels, the CSF approach can:

1. Improve the tracking performance of *both* the EW-RLS and SW-RLS algorithms

2. Eliminate the penalty incurred due to implicit tracking of the null channel subspaces so that even the conventional RLS algorithm can achieve similar tracking performance as a reduced rank RLS algorithm that explicitly exploits the channel subspace structure

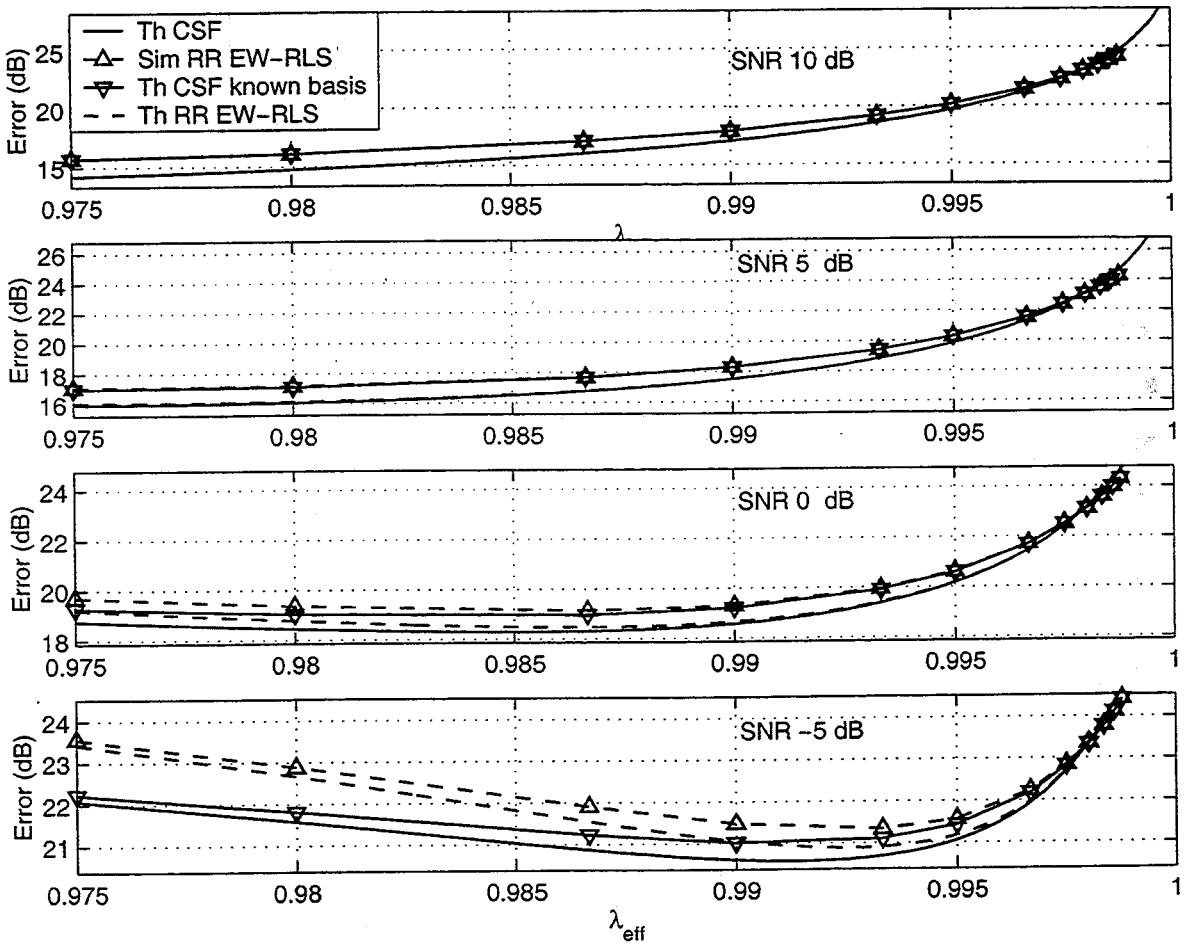

Figure 3-3: Tracking performance of reduced rank uncorrelated parameter EW-RLS and CSF algorithm

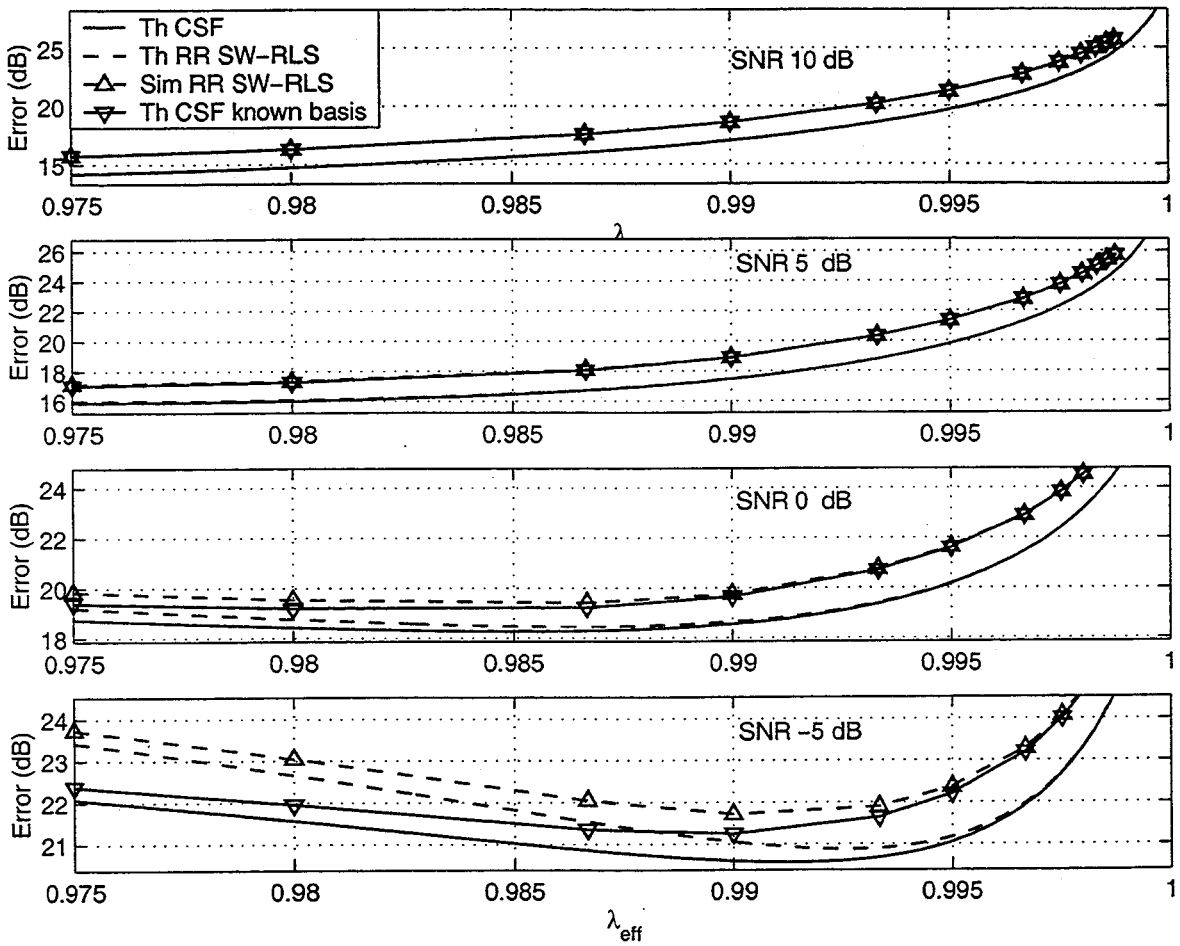

Figure 3-4: Tracking performance of reduced rank uncorrelated parameter SW-RLS and CSF algorithm

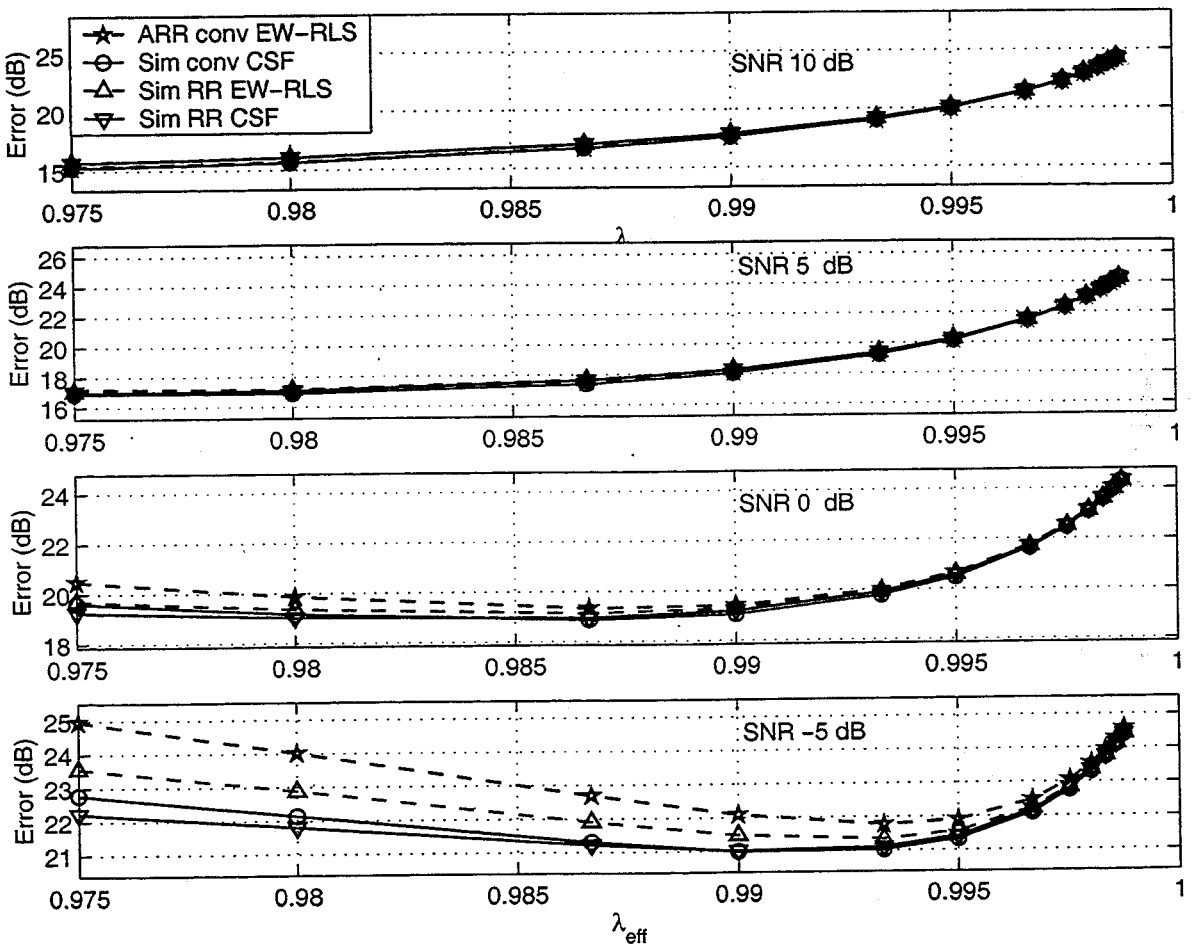

Figure 3-5: Comparison of the simulated tracking performance of the reduced rank uncorrelated parameter EW-RLS and CSF algorithms with their conventional variants

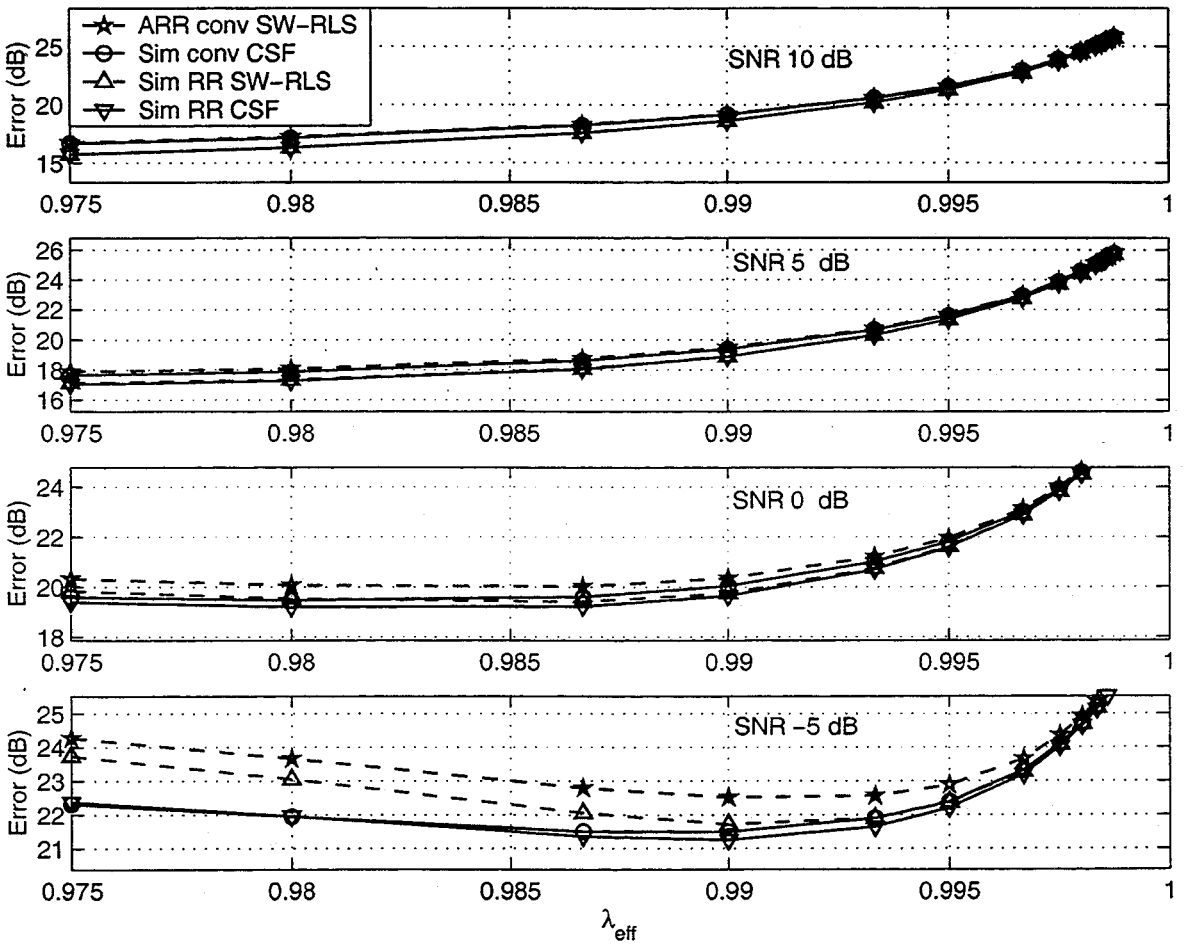

Figure 3-6: Comparison of the simulated tracking performance of the reduced rank uncorrelated parameter SW-RLS and CSF algorithms with their conventional variants

3.4 Factors affecting the theoretical performance of the CSF algorithm

While deriving analytical expressions for the performance of the tracking performance of RLS and the CSF algorithms, it was assumed that the system was in steady state. Such an assumption was made so that any transient effects associated with the initialization of the algorithm (particularly in the EW-RLS case) could be ignored. For RLS algorithms to be in steady state, the number of iterations n was assumed to be large enough. When the direct averaging method was used to solve the stochastic difference equation in (2.34), it was assumed that the parameters α and λ were close to 1. The statistical ensemble average for $\mathbf{A}(n)$ was used instead of its instantaneous value in deriving the analytical expressions for the tracking performance of the EW-RLS algorithm. As a result, even if the parameters α, λ_{eff} and the eigenvectors $\mathbf{U}$ are known apriori, a mismatch exists between the analytically derived results and the corresponding results using simulated data when the assumptions are not valid.

3.5 Effect of finite data on eigenvector estimation errors

The MMSE post-filter $\mathbf{F}$ has been referred to as a channel subspace filter because it exploits the channel subspace structure. It is expressed as $\mathbf{F} = \mathbf{U}\Sigma_{ff}\mathbf{U}^H$, where the eigenvectors of the channel correlation matrix $\mathbf{R}_{hh}$ are represented by $\mathbf{U}$ and Σ_{ff} represents the CSF coefficients. In a slightly more realistic scenario, the eigenvectors of the channel correlation matrix are not known apriori. If, however, the parameter α in (2.1) and the variance σ^2 of the additive noise $w(n)$ in (2.2) are known apriori then for an RLS algorithm parameterized by the common tracking parameter λ_{eff}, the CSF coefficients may be computed using (3.7). The channel subspace filter $\mathbf{F}$ may be then be computed as $F = \hat{\mathbf{U}}\Sigma_{ff}\hat{\mathbf{U}}^H$ using the estimated eigenvectors $\hat{\mathbf{U}}$.

(2.62) suggests that the eigenvectors $\hat{\mathbf{U}}$ may be estimated from an eigenvalue de-

composition of the channel estimate correlation matrix $\mathbf{R}_{\hat{h}\hat{h}}$. Given a channel estimate time series $\underline{\hat{h}}(n)$, the channel estimate correlation matrix may be estimated as::

$$\hat{\mathbf{R}}_{\hat{h}\hat{h}} = \frac{1}{M_u} \sum_n \underline{\hat{h}}(n) \underline{\hat{h}}^H(n) \tag{3.8}$$

where M_u is the number of independent channel estimates used to estimate $\hat{\mathbf{R}}_{\hat{h}\hat{h}}$. The eigenvectors could then be estimated by computing the eigenvalue decomposition of $\mathbf{R}_{\hat{h}\hat{h}}$ such that:

$$\hat{\mathbf{R}}_{\hat{h}\hat{h}} = \hat{\mathbf{U}} \hat{\Sigma}_{\hat{h}\hat{h}} \hat{\mathbf{U}}^H \tag{3.9}$$

If the estimated eigenvectors $\hat{\mathbf{U}}$ are used, instead of the true eigenvectors $\mathbf{U}$, to compute the channel subspace filter $\hat{\mathbf{F}} = \hat{\mathbf{U}} \Sigma_{ff} \hat{\mathbf{U}}^H$, then it is certainly foreseeable that errors in eigenvector estimation may impact the performance of the CSF algorithm.

Consider a process represented by (2.1), where the taps of the channel impulse response vector $\underline{h}(n)$ are indeed uncorrelated. This case is considered for the sake of simplicity of analysis and the inferred results provide insight into the general case where the taps are correlated. The correlation matrix $\mathbf{R}_{hh} = E[\underline{h}(n)\underline{h}^H(n)]$ would then be a diagonal matrix with the $N \times N$ identity matrix $\mathbf{I}$ as its eigenvectors. If (3.8) and (3.9) were used to estimate the eigenvectors of the process then the $< i, j >^{th}$ element, i.e. the element in the i^{th} row and the j^{th} column of $\hat{\mathbf{R}}_{\hat{h}\hat{h}}$, would be given by:

$$\left(\hat{\mathbf{R}}_{\hat{h}\hat{h}}\right)_{<i,j>} = \frac{1}{M_u} \sum_n \hat{h}_i(n) \hat{h}_j^*(n) \tag{3.10}$$

where $\hat{h}_i(n)$ and $\hat{h}_j(n)$ are the i^{th} and j^{th} elements of the $N \times 1$ channel estimate vector $\underline{\hat{h}}(n)$.

The channel impulse response vector $\underline{h}(n)$ is a zero mean Gaussian random vector. The distribution of the RLS channel estimate $\underline{\hat{h}}(n)$ is very difficult to compute analytically because of the complex data dependencies and the inherent non-linear relationships. There are limited published results dealing with the distribution of the RLS channel estimate vector $\underline{\hat{h}}(n)$. None of these results pertain to the channel model in (2.1). However, there are comparable results for the least mean square (LMS) algo-

rithm. In [5], Bucklew et. al. show that the fluctuations of the weight error vector $\underline{\epsilon}(n)$ about zero are asymptotically Gaussian in distribution. This asymptotic distribution was shown to have been the result of two notions: a form of the central limit theorem and an assumed ergodicity of the input and disturbances in the LMS algorithm. Since $\underline{\hat{h}}(n-1) = \underline{h}(n) - \underline{\epsilon}(n)$ and the sum, or difference, of two uncorrelated Gaussian random vectors is also a Gaussian random vector, the channel estimate vector $\underline{\hat{h}}(n)$ may also be assumed to asymptotically Gaussian in distribution. Though the LMS algorithm has not been formally discussed in this thesis, the form of the algorithm [37] is certainly reminiscent of the RLS algorithms. This motivates the use of the result for LMS channel estimate vector in determining the distribution of the RLS channel estimate vector. The zero mean RLS channel estimate vector $\underline{\hat{h}}(n)$ is thus assumed to be asymptotically Gaussian in distribution. The eigenvectors of $\mathbf{R}_{hh}$ are also the eigenvectors of $\mathbf{R}_{\hat{h}\hat{h}}$. Since the taps of the channel impulse response $\underline{h}(n)$ are uncorrelated, the taps of the channel estimate vector $\underline{\hat{h}}(n)$ are also uncorrelated and consequently independent, since the channel estimate vector is asymptotically Gaussian in distribution. The mean and the variance of the estimated channel estimate correlation matrix in (3.8) may thus be respectively expressed as:

$$\begin{aligned} E\left[\left(\hat{\mathbf{R}}_{\hat{h}\hat{h}}\right)_{<i,j>}\right] &= E[\frac{1}{M_u}\sum_n \hat{h}_i(n)\hat{h}_j^*(n)] \\ &= \frac{1}{M_u}\sum_n E[\hat{h}_i(n)\hat{h}_j^*(n)] \end{aligned} \tag{3.11}$$

and

$$\begin{aligned} var\left(\left(\hat{\mathbf{R}}_{\hat{h}\hat{h}}\right)_{<i,j>}\right) &= var(\frac{1}{M_u}\sum_n \hat{h}_i(n)\hat{h}_j^*(n)) \\ &= \frac{1}{M_u^2}\sum_n var(\hat{h}_i(n)\hat{h}_j^*(n)) \end{aligned} \tag{3.12}$$

For $i \neq j$, (3.11) and (3.12) may be written as:

$$\begin{aligned} E\left[\left(\hat{\mathbf{R}}_{\hat{h}\hat{h}}\right)_{<i,j>}\right] &= \frac{1}{M_u}\sum_n E[\hat{h}_i(n)]E[\hat{h}_j^*(n)] \\ &= \frac{1}{M_u}\sum_n 0 \\ &= 0 \end{aligned} \tag{3.13}$$

and

$$\begin{aligned} var\left(\left(\hat{\mathbf{R}}_{\hat{h}\hat{h}}\right)_{<i,j>}\right) &= \frac{1}{M_u^2}\sum_n var(\hat{h}_i(n))var(\hat{h}_j^*(n)) \\ &= \frac{1}{M_u^2}M_u var(\hat{h}_i(n))var(\hat{h}_j^*(n)) \\ &= \frac{1}{M_u}var(\hat{h}_i(n))var(\hat{h}_j^*(n)) \rightarrow 0 \text{ as } M_u \rightarrow \infty \end{aligned} \tag{3.14}$$

while for $i = j$, (3.11) and (3.12) may be rewritten as:

$$\begin{aligned} E\left[\left(\hat{\mathbf{R}}_{\hat{h}\hat{h}}\right)_{<i,i>}\right] &= \frac{1}{M_u}\sum_n E[\hat{h}_i(n)\hat{h}_i^*(n)] \\ &= \frac{1}{M_u}\sum_n var(\hat{h}_i(n)) \\ &= \sigma_{\hat{h}i}^2 \end{aligned} \tag{3.15}$$

and using the Gaussian moment factoring theorem [15] for fourth-order moments:

$$\begin{aligned} var\left(\left(\hat{\mathbf{R}}_{\hat{h}\hat{h}}\right)_{<i,i>}\right) &= \frac{1}{M_u^2}\sum_n var(\hat{h}_i(n)\hat{h}_i^*(n)) \\ &= \frac{1}{M_u^2}M_u var(\hat{h}_i^2(n))) \\ &= \frac{1}{M_u}var(\hat{h}_i(n)) \rightarrow 0 \text{ as } M_u \rightarrow \infty \end{aligned} \tag{3.16}$$

From (3.15), (3.13), (3.16) and (3.14), for a channel impulse response vector with independent tap fluctuations, each $< i, j >^{th}$ element of the estimated channel estimate correlation matrix $\hat{\mathbf{R}}_{\hat{h}\hat{h}}$ converges asymptotically in the mean square sense to the corresponding element of the true channel estimate correlation matrix $\mathbf{R}_{\hat{h}\hat{h}}$. By

extension, the entire estimated channel estimate correlation matrix also converges asymptotically, in the mean square sense, to the true channel estimate correlation matrix.

$$\hat{\mathbf{R}}_{\hat{h}\hat{h}} = \mathbf{R}_{\hat{h}\hat{h}} + \Delta_{\hat{h}\hat{h}} \tag{3.17}$$

where $\mathbf{R}_{\hat{h}\hat{h}}$ is the true diagonal correlation matrix and $\Delta_{\hat{h}\hat{h}}$ is the non-diagonal matrix representing the error in estimating the channel estimate correlation matrix. As the number of independent channel estimated M_u increases, the error in estimating the channel correlation matrix asymptotically decreases. The error in estimating the channel estimate correlation matrix results in eigenvector estimation errors. The errors in eigenvector estimation results in a deterioration in performance of the CSF algorithm. The deterioration in performance depends on the number of *independent* channel estimates available. Since the channel estimates are correlated, the number of *independent* channel estimates is generally less than the number of available channel estimates and depends on the degree of correlation between successive estimates.

Figures 3-7 and 3-8 show the marginal deterioration in the performance of the CSF algorithm when a finite number of channel estimates are used to estimate the eigenvectors. In this particular case the eigenvectors were estimated using 42,000 channel estimates. Even though several of these estimates are correlated, it still represents a sufficient number of independent channel estimates so that the errors in eigenvector estimation are relatively marginal. Despite this it can be seen that the error in eigenvector estimation affects the performance of the CSF algorithm. For the time varying system model in (2.1), the degree of correlation between successive samples of the true channel impulse response vector is given by the parameter α. The degree of correlation between successive RLS channel estimates is, however, not immediately apparent. The following section explicitly derives an analytical expression for the degree of correlation between successive EW-RLS channel estimates. This is used to derive an expression for the number of independent channel estimates available for a given number of samples of received data M_{data}.

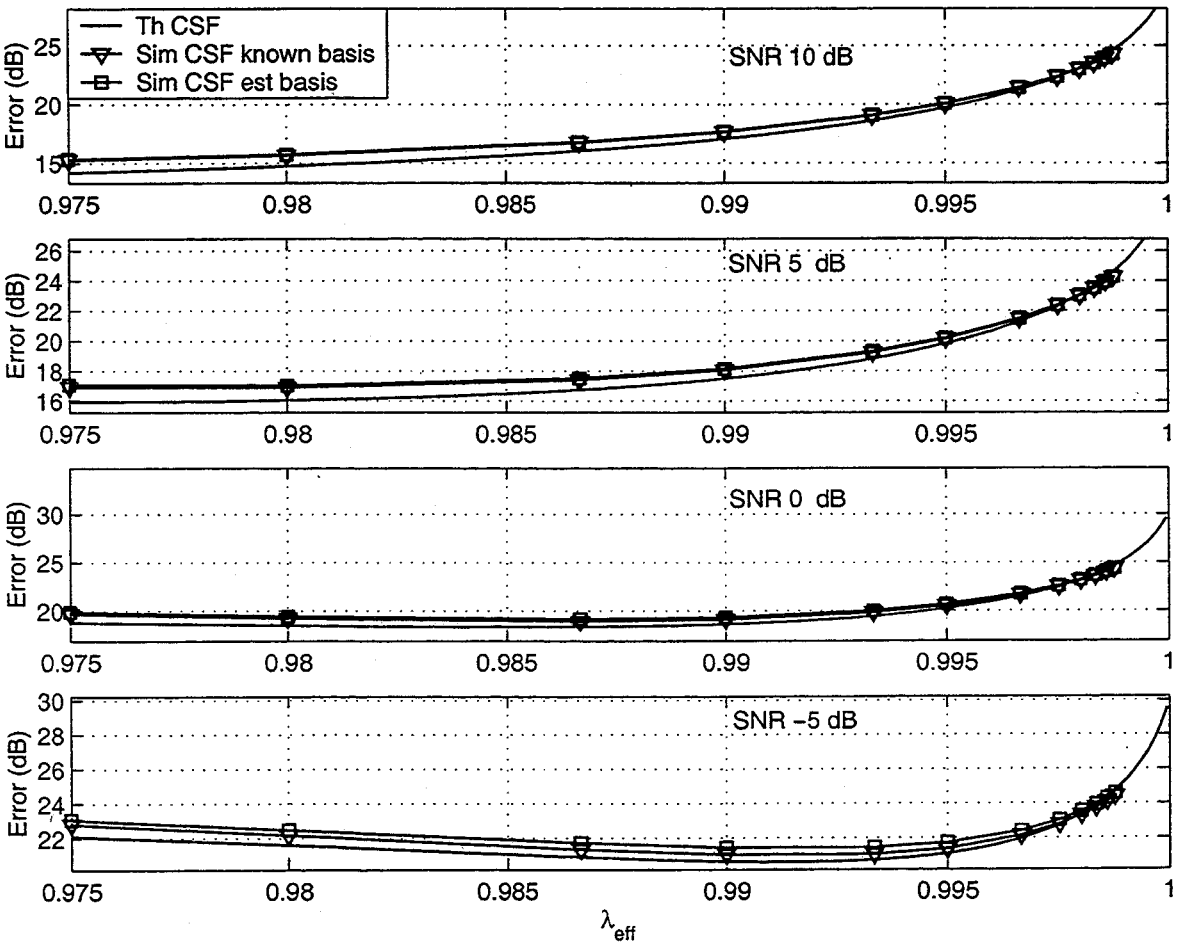

Figure 3-7: Tracking performance of EW-RLS and CSF using known and estimated eigenvectors

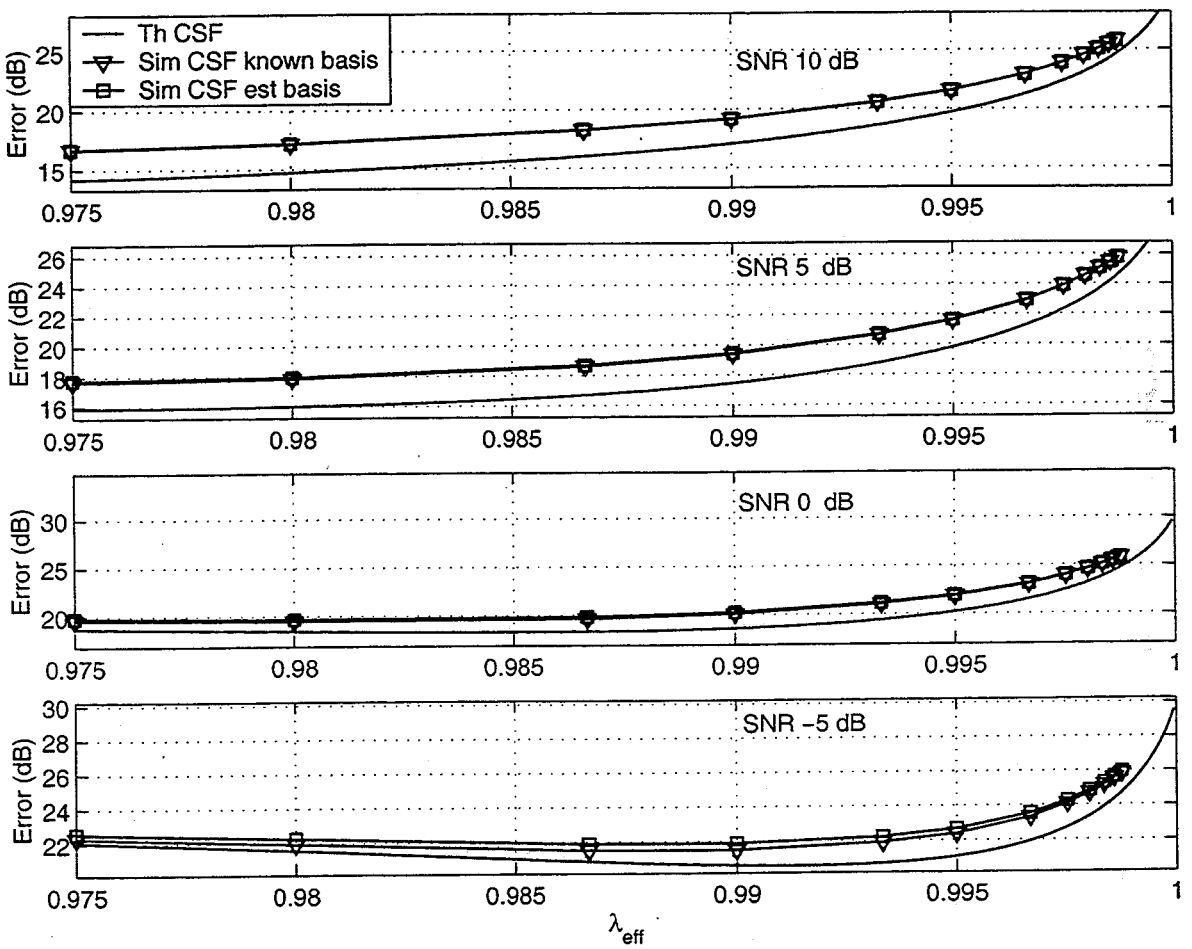

Figure 3-8: Tracking performance of SW-RLS and CSF using known and estimated eigenvectors

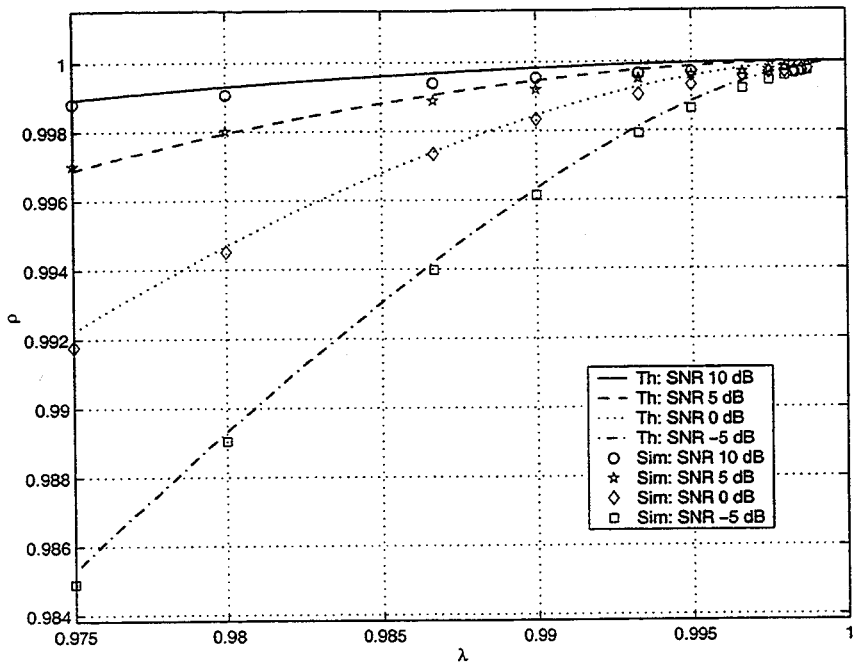

Figure 3-9: Comparision of the theoretical and simulated values for the degree of correlation between successive EW-RLS estimates

3.5.1 Impact of correlation of successive estimates on eigenvector estimation errors

From (2.28), the EW-RLS channel estimate is given by:

$$\underline{\hat{h}}(n) = \underline{\hat{h}}(n-1) + \underline{\mathbf{K}}(n)\xi^*(n \mid n-1) \tag{3.18}$$

The cross-correlation metric ρ is defined as:

$$\rho = \frac{< \underline{\hat{h}}(n), \underline{\hat{h}}(n-1) >}{\sqrt{< \underline{\hat{h}}(n), \underline{\hat{h}}(n) >< \underline{\hat{h}}(n-1), \underline{\hat{h}}(n-1) >}} \tag{3.19}$$

Using (2.28), an expression for the cross product $< \underline{\hat{h}}(n), \underline{\hat{h}}(n-1) >= E[\underline{\hat{h}}^H(n)\underline{\hat{h}}(n-1)]$ can be derived. The steps in this derivation are given below:

$$E[\underline{\hat{h}}^H(n)\underline{\hat{h}}(n-1)] = E[\underline{\hat{h}}^H(n-1)\underline{\hat{h}}(n-1)] + E[\underline{\hat{h}}^H(n-1)\underline{\mathbf{K}}(n)\xi^*(n \mid n-1)] \tag{3.20}$$

Substituting (2.4) for $\xi(n \mid n-1)$ and using the relationship $E[\underline{\mathbf{K}}(n)\underline{x}^H(n)] = (1-\lambda)\mathbf{I}$ [15], the terms in the equation above can be rearranged so that:

$$E[\underline{\hat{h}}^H(n)\underline{\hat{h}}(n-1)] = \mathrm{tr}(\mathbf{R}_{\hat{h}\hat{h}}) + (1-\lambda)\mathrm{tr}(\mathbf{R}_{\hat{h}\epsilon}) \tag{3.21}$$

Substituting this in the expression for the degree of correlation between successive estimates given in (3.19) results in:

$$\rho = 1 + (1-\lambda)\frac{\mathrm{tr}(\mathbf{R}_{\hat{h}\epsilon})}{\mathrm{tr}(\mathbf{R}_{\hat{h}\hat{h}})} \tag{3.22}$$

where it is assumed that the system is in steady state so that $E[\underline{\hat{h}}^H(n-1)\underline{\hat{h}}(n-1)] = E[\underline{\hat{h}}^H(n)\underline{\hat{h}}(n)] = tr(\mathbf{R}_{\hat{h}\hat{h}})$. Substituting (2.62) and (2.63) and using the relationship $\mathbf{R}_{\hat{h}\epsilon} = \mathbf{R}_{\hat{h}h} - \mathbf{R}_{\hat{h}\hat{h}}$, (3.22) can be written as:

$$\begin{aligned}\rho &= 1 - (1-\lambda)\frac{(1-\alpha)\mathrm{tr}(\mathbf{R}_{hh}) + (1-\alpha\lambda)\sigma^2\mathrm{tr}(\mathbf{R}_{\mathbf{xx}}^{-1})}{(1+\alpha\lambda)\mathrm{tr}(\mathbf{R}_{hh}) + (1-\alpha\lambda)\sigma^2\mathrm{tr}(\mathbf{R}_{\mathbf{xx}}^{-1})} \\ &= 1 - (1-\lambda)\frac{(1-\alpha)\mathrm{tr}(\mathbf{R}_{hh}) + (1-\alpha\lambda)\sigma^2\mathrm{tr}(\mathbf{I})}{(1+\alpha\lambda)\mathrm{tr}(\mathbf{R}_{hh}) + (1-\alpha\lambda)\sigma^2\mathrm{tr}(\mathbf{I})} \leq 1\end{aligned} \tag{3.23}$$

Given this dependence, the correlation window size, i.e. the number of successive estimates that are correlated, is defined as $W = 1/(1-\rho)$. The correlation window size W can be expressed using (3.23) as:

$$\begin{aligned}W &= \frac{1}{1-\lambda}\left(\frac{(1+\alpha\lambda)\mathrm{tr}(\mathbf{R}_{hh}) + (1-\alpha\lambda)\sigma^2\mathrm{tr}(\mathbf{I})}{(1-\alpha)\mathrm{tr}(\mathbf{R}_{hh}) + (1-\alpha\lambda)\sigma^2\mathrm{tr}(\mathbf{I})}\right) \\ &= \frac{1}{1-\lambda}\left(\frac{(1+\alpha\lambda)\mathrm{SNR} + N(1-\alpha\lambda)}{(1-\alpha)\mathrm{SNR} + N(1-\alpha\lambda)}\right)\end{aligned} \tag{3.24}$$

where the signal to noise ratio $SNR = \mathrm{tr}(\mathbf{R}_{hh})/\sigma^2$. (3.23) shows the relationship between the degree of correlation ρ and the tracking parameter λ of the EW-RLS algorithm. The functional dependence of the correlation window size W on the channel parameter α, the tracking parameter λ and the SNR of the channel will at times be denoted using the notation $W(\alpha, \lambda, SNR)$. Figure 3-9 shows the dependence of ρ on λ and SNR, with the parameter $\alpha = 0.9995$. The analytical expression for

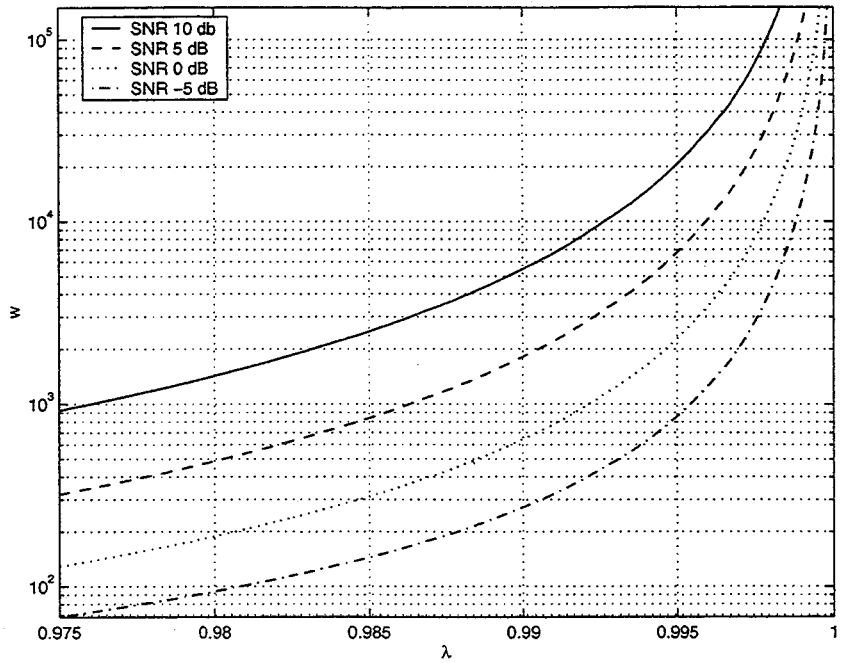

Figure 3-10: Number of successive correlated EW-RLS estimates

the degree of correlation (3.22) is corroborated using simulations. The agreement between the theoretically predicted degree of correlation and the computed value for the same is apparent in figure 3-9. As can be seen, at high SNR's, the channel estimates are more correlated while at low SNR's, the channel estimates are less correlated for same value of the tracking parameter λ. Figure 3-10 shows the number of successive correlated channel estimates W as a function of SNR and λ as given by (3.24), for the same parameter α. At high SNR's the number of correlated W is greater than at moderate to low SNR's for the same tracking parameter λ. Based on the discussion in the previous section this implies that an even greater number of actual channel estimates, roughly given by $M_{data} = M_u \times W$, would be needed for reliable eigenvector estimation at higher SNR's than at lower SNR's; however at high SNR the observation noise induced error in the channel estimate is smaller and a smaller λ would be used to track the channel. When only a finite number of samples M_{data} of the received data $y(n)$ are available, the number of independent channel estimates is approximately given by $M_u = M_{data}/W(\alpha, \lambda, SNR)$. The finite number of independent channel estimates M_u thus available results in errors in estimation of the

channel estimate correlation matrix and hence in the estimation of the eigenvectors $\hat{\mathbf{U}}$. These factors result in a loss of performance of the CSF algorithm when finite data is used to estimate the eigenvectors of the estimated channel impulse response correlation matrix. The following section explicitly analyzes the impact of eigenvector estimation on the performance of the CSF algorithm.

3.6 Impact of errors in eigenvector estimation on performance of CSF

The estimated channel subspace filter is computed as:

$$\hat{\mathbf{F}} = \hat{\mathbf{U}}\Sigma_{ff}\hat{\mathbf{U}}^H \tag{3.25}$$

The error in estimating the channel estimate correlation matrix $\hat{\mathbf{R}}_{\hat{h}\hat{h}}$ leads to errors in estimating the eigenvectors $\mathbf{U}$. Due to errors in eigenvector estimation, the estimated eigenvectors may be written in terms of the true eigenvectors as:

$$\begin{aligned} \hat{\mathbf{U}} &= \mathbf{U} + \left(\hat{\mathbf{U}} - \mathbf{U}\right) \\ &= \mathbf{U} + \Delta_u \end{aligned} \tag{3.26}$$

where Δ_u represents the error in eigenvector estimation due to finite number M_u of independent channel estimates. The estimated channel subspace filter can be rewritten as:

$$\begin{aligned} \hat{\mathbf{F}} &= (\mathbf{U} + \Delta_u)\Sigma_{ff}(\mathbf{U} + \Delta_u)^H \\ &= \underbrace{\mathbf{U}\Sigma_{ff}\mathbf{U}^H}_{\mathbf{F}} + \Delta_u\Sigma_{ff}\mathbf{U}^H + \mathbf{U}\Sigma_{ff}\Delta_u^H + \Delta_u\Sigma_{ff}\Delta_u^H \\ &= \mathbf{F} + \Delta_f \end{aligned} \tag{3.27}$$

where

$$\Delta_f = \Delta_u\Sigma_{ff}\mathbf{U}^H + \mathbf{U}\Sigma_{ff}\Delta_u^H + \Delta_u\Sigma_{ff}\Delta_u^H \tag{3.28}$$

represents the error in computing the channel subspace filter due to errors in eigenvector estimation. The estimated channel subspace filter in (3.27) is used to post-filter the least-squares channel estimate $\underline{\hat{h}}(n)$ so that:

$$\underline{\hat{h}}_{\hat{p}}(n) = \hat{\mathbf{F}}\underline{\hat{h}}(n) \tag{3.29}$$

(3.29) can be rewritten in terms of the error in computing the channel subspace filter by substituting the expression for $\hat{\mathbf{F}}$ in (3.27):

$$\begin{aligned} \underline{\hat{h}}_{\hat{p}}(n) &= (\mathbf{F} + \Delta_f)\underline{\hat{h}}(n) \\ &= \mathbf{F}\underline{\hat{h}}(n) + \Delta_f\underline{\hat{h}}(n) \\ &= \underline{\hat{h}}_p(n) + \underline{\delta h}_p(n) \end{aligned} \tag{3.30}$$

where $\underline{\hat{h}}_p(n) = \mathbf{F}\underline{\hat{h}}(n)$ is the post-filtered channel estimate assuming perfect estimation of the channel subspace filter and $\underline{\delta h}_p(n) = \Delta_f\underline{\hat{h}}(n)$ is the error in the post-filtered channel estimate due to imperfect eigenvector estimation. The resultant tracking error vector due to the use of imperfect eigenvector estimates in estimating the channel subspace filter $\mathbf{F}$ is given by:

$$\underline{\hat{\epsilon}}_f(n) = \underline{h}(n) - \underline{\hat{h}}_{\hat{p}}(n-1) \tag{3.31}$$

Substituting the expression for $\underline{\hat{h}}_p(n)$ from (3.30) in (3.31) results in:

$$\begin{aligned} \underline{\hat{\epsilon}}_f(n) &= \underline{h}(n) - \underline{\hat{h}}_p(n-1) - \underline{\delta h}_p(n-1) \\ &= \underline{\epsilon}_f(n) - \underline{\delta h}_p(n-1) \end{aligned} \tag{3.32}$$

The tracking error correlation matrix is then given by:

$$\mathbf{R}_{\hat{\epsilon}_f\hat{\epsilon}_f} = \mathbf{R}_{\epsilon_f\epsilon_f} + \mathbf{R}_{\delta h_p\delta h_p} - \mathbf{R}_{\epsilon_f\delta h_p} - \mathbf{R}_{\epsilon_f\delta h_p} \tag{3.33}$$

so the resultant mean square tracking error is:

$$\begin{aligned} E[\| \hat{\underline{\epsilon}}_f(n) \|^2] &= \mathbf{R}_{\hat{\epsilon}_f\hat{\epsilon}_f} \\ &= tr(\mathbf{R}_{\epsilon_f\epsilon_f}) + tr(\mathbf{R}_{\delta h_p \delta h_p}) - tr(\mathbf{R}_{\epsilon_f \delta h_p}) - tr(\mathbf{R}_{\epsilon_f \delta h_p}) \end{aligned} \tag{3.34}$$

where $\underline{\epsilon}_f(n)$ is the post-filtered channel estimation error vector if there were no errors in eigenvector estimation and $\mathbf{R}_{\epsilon_f\epsilon_f}$ is the corresponding tracking error correlation matrix. The additional correlation matrix terms in (3.33) represent the error incurred due to errors in eigenvector estimation. The correlation matrix $\mathbf{R}_{\epsilon_f \delta h_p}$ can be expressed in terms of the error in eigenvector estimation using (3.28) as:

$$\begin{aligned} \mathbf{R}_{\epsilon_f \delta h_p} &= E[\underline{\epsilon}_f(n)\underline{\delta h}_p^H(n-1)] \\ &= E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\Delta_f^H] \\ &= E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\mathbf{U}\Sigma_{ff}\Delta_u^H] + E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\Delta_u^H\Sigma_{ff}\mathbf{U}^H] \\ &+ E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\Delta_u\Sigma_{ff}\Delta_u^H] \end{aligned} \tag{3.35}$$

As can be seen from (3.35), the correlation matrix $\mathbf{R}_{\epsilon_f \delta h_p}$ has terms dependent on $\mathbf{U}$, Δ_u, and Σ_{ff}. Since the eigenvectors $\hat{\mathbf{U}}$ used to construct the channel subspace filter $\hat{\mathbf{F}}$ in (3.25) are estimated from the the channel estimate time series according to (3.8) and (3.9), the eigenvector estimation error term Δ_u is in general correlated with the channel estimate $\hat{\underline{h}}(n-1)$. By extension, it is also correlated with the ideal post-filtered tracking error vector $\underline{\epsilon}_f(n) = \underline{h}(n) - \hat{\underline{h}}(n-1)$. Hence, computing the expected value of the correlation matrix $\mathbf{R}_{\epsilon_f \delta h_p}$ is a considerably difficult task. In the previous section, an expression for the number of successive correlated estimates was derived as a function of the system parameters. Suppose it is assumed that the system is in steady state and that a relatively large number of independent channel estimates are available. Since (3.24) indicates that only a finite number of successive channel estimates are correlated, the errors in estimating the channel estimate correlation matrix in (3.8) and hence in estimating the eigenvectors (3.9) may be assumed to be independent of the instantaneous channel estimate $\hat{\underline{h}}(n-1)$ and by extension of the instantaneous post-filtered channel estimation error vector $\underline{\epsilon}_f(n)$. The error in

estimating the CSF filter Δ_f in (3.28) may thus also be assumed to be independent of the instantaneous channel estimate $\underline{\hat{h}}(n-1)$ and of the instantaneous channel estimation error vector $\underline{\epsilon}(n)$. The correlation matrix $\mathbf{R}_{\epsilon_f \delta h_p}$ may now expressed as:

$$\begin{aligned}
\mathbf{R}_{\epsilon_f \delta h_p} &= E[\underline{\epsilon}_f(n)\underline{\hat{h}}^H(n-1)\Delta_f^H] \\
&= E[E[\underline{\epsilon}_f(n)\underline{\hat{h}}^H(n-1)\Delta_f^H \mid \Delta_f]] \\
&= E[\underbrace{E[\underline{\epsilon}_f(n)\underline{\hat{h}}^H(n-1)]}_{\mathbf{0}}\Delta_f^H] \\
&= \mathbf{0}
\end{aligned} \tag{3.36}$$

where $E[\underline{\epsilon}_f(n)\underline{\hat{h}}^H(n-1)] = \mathbf{0}$ is a consequence of the fact that $\mathbf{F} = \mathbf{R}_{\hat{h}\hat{h}}^{-1}\mathbf{R}_{\hat{h}h}$ is an MMSE post-filter. The result in (3.36) has been verified using simulation when a large number of channel estimates are used to estimate the eigenvectors. From (3.36), $\mathbf{R}_{\delta h_p \epsilon_f} = \mathbf{0}$. (3.33) may be rewritten as:

$$\mathbf{R}_{\hat{\epsilon}_f \hat{\epsilon}_f} = \mathbf{R}_{\epsilon_f \epsilon_f} + \mathbf{R}_{\delta h_p \delta h_p} \tag{3.37}$$

and the resultant channel estimation error is given by:

$$\begin{aligned}
E[\| \underline{\hat{\epsilon}}_f(n) \|^2] &= tr(\mathbf{R}_{\hat{\epsilon}_f \hat{\epsilon}_f}) \\
&= tr(\mathbf{R}_{\epsilon_f \epsilon_f}) + tr(\mathbf{R}_{\delta h_p \delta h_p}) \\
&= E[\| \underline{\epsilon}_f(n) \|^2] + tr(\mathbf{R}_{\delta h_p \delta h_p})
\end{aligned} \tag{3.38}$$

where $\mathbf{R}_{\delta h_p \delta h_p}$ is given by:

$$\mathbf{R}_{\delta h_p \delta h_p} = E[\Delta_f \underline{\hat{h}}(n-1)\underline{\hat{h}}^H(n-1)\Delta_f^H] \tag{3.39}$$

Thus (3.38) has terms which depend on the eigenvector estimation error Δ_u. From the discussion in a previous section, it was concluded that the estimated channel estimate correlation matrix in (3.8) convergence asymptotically in the mean square sense to the true channel estimate correlation matrix. Since the eigenvectors $\hat{\mathbf{U}}$ are

obtained from the estimated channel estimate correlation matrix as given in (3.9), the estimated eigenvectors also converge asymptotically in the mean square sense to the true eigenvectors. Thus the error in eigenvector estimation $\Delta_u \rightarrow 0$ as $M_u \rightarrow \infty$ where $M_u = M_{data}/W(\alpha, \lambda, SNR)$ is the number of independent channel estimates available when the number of available channel estimates is given by M_{data} and the correlation window size is given by $W(\lambda, SNR)$. As expressed in (3.38), the performance of the CSF algorithm approaches the theoretical performance predicted in (2.73) as $M_u \rightarrow \infty$. Figure 3-11 shows the deterioration in performance of CSF as the number of EW-RLS channel estimates used to estimate the eigenvectors is varied. Periodically spaced samples from channel estimate time series are selected and used to estimate the eigenvectors. For example, for the graph labeled '50 estimates', every 840'th sample from the 42,000 available channel estimates is used to generate the eigenvectors. Figure 3-10 provides an estimate for the number of successive correlated channel estimates. If this correlation window size is less than the periodicity of the channel estimate and the periodicity of the channel estimates is increased than the number of vectors may be considered independent decreases and there will be a discernible deterioration in performance. If however, the correlation window size is much greater than the periodicity of the channel estimate samples and the periodicity of the channel estimate samples is increased, than the vectors might still be considered to be dependent and the deterioration in performance might not be as significant or even noticeable. This suggests that the CSF algorithm corresponding to smaller values of λ_{eff} and lower values of SNR would experience a greater deterioration in performance than at higher values of λ and SNR. As can be seen, as the number of channel estimates used to estimate the eigenvectors is decreased there is a significant deterioration in the performance of the CSF algorithm for smaller values of λ_{eff} and SNR. At higher values of λ_{eff} and SNR the deterioration is barely noticeable, as suggested. Figure 3-12 shows the increase in the error term $tr(\mathbf{R}_{\delta h_p \delta h_p})$ as fewer estimates are used to compute the eigenvectors. The cross correlation term $\mathbf{R}_{\delta h_p \epsilon_f}$ was computed using simulations to be negligible, as assumed earlier.

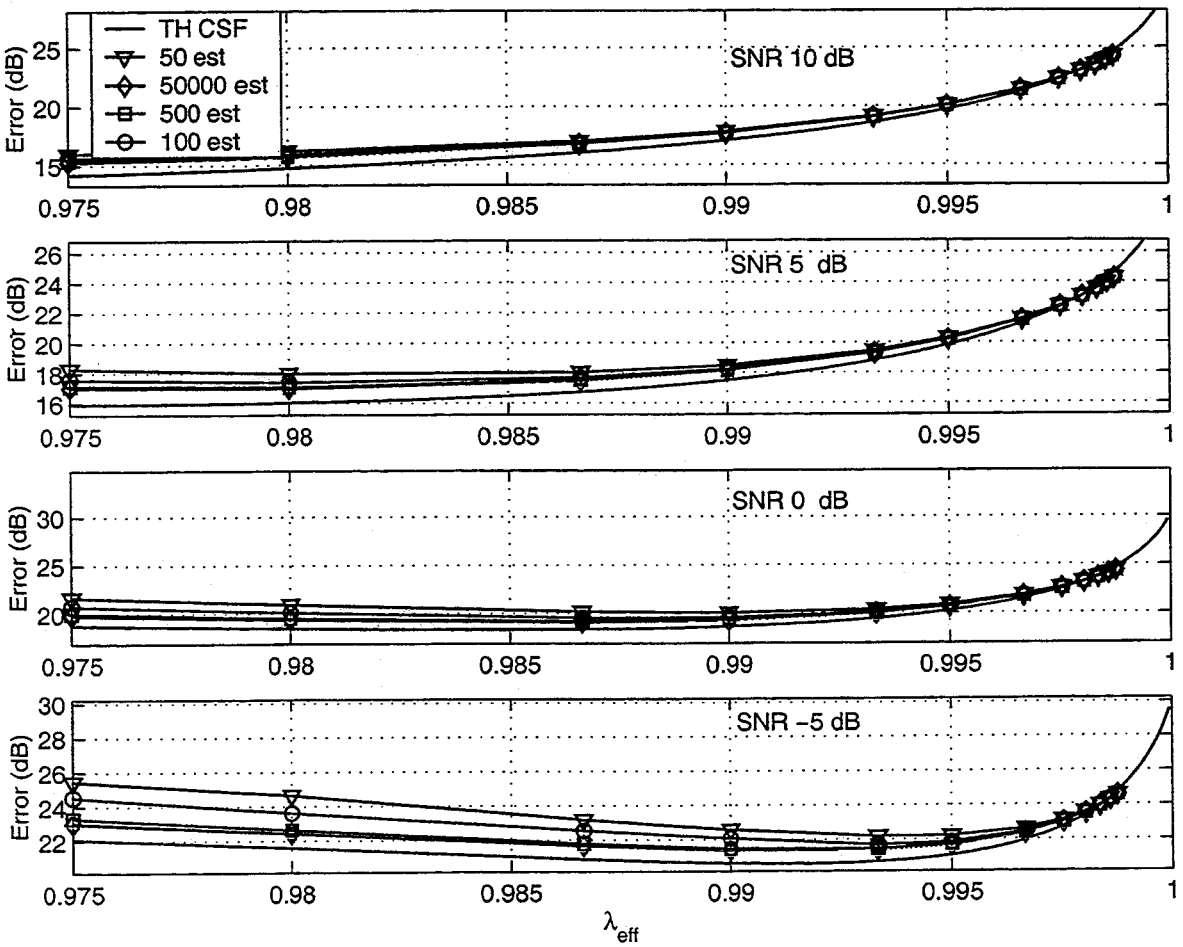

Figure 3-11: Effect of finite data used to estimate eigenvectors on performance of CSF

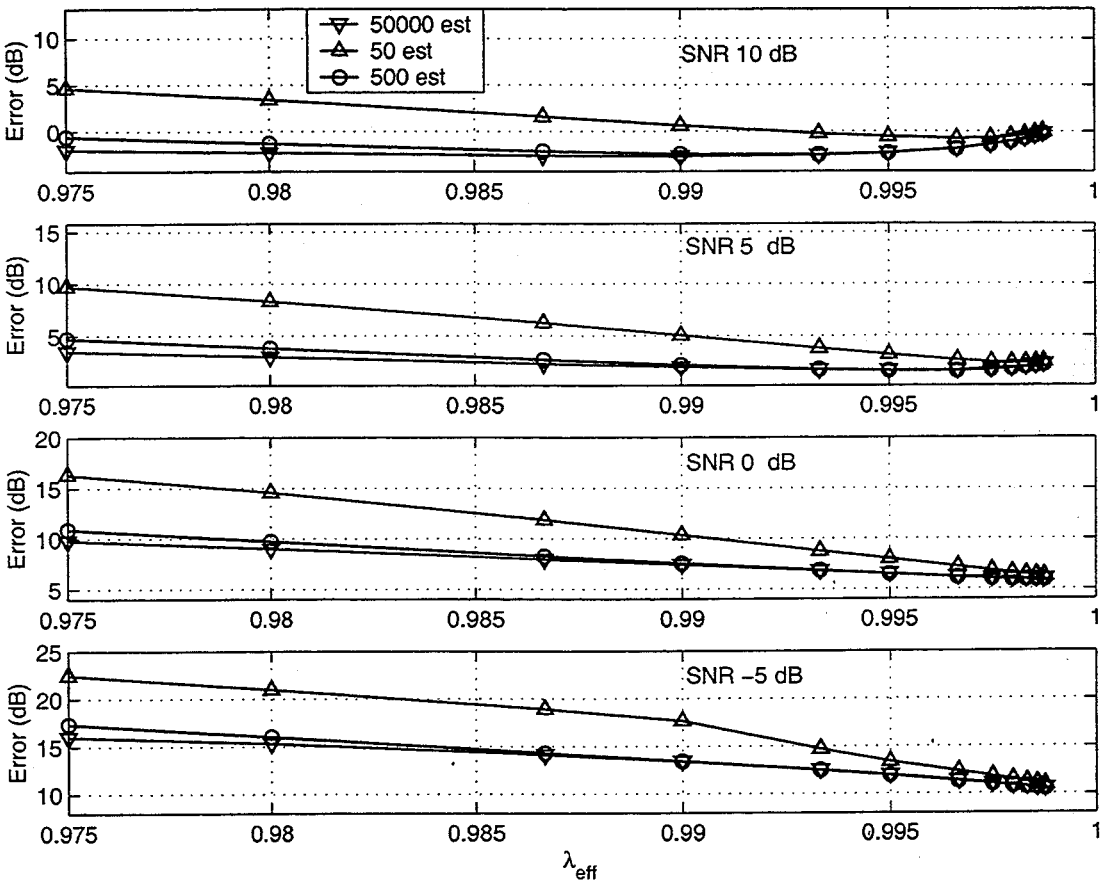

Figure 3-12: Effect of finite data used to estimate eigenvectors on $tr(\mathbf{R}_{\delta h_p \delta h_p})$

3.7 Summary

The results on simulated data have shown that the finite number of available data samples affects the performance of the CSF algorithm. Errors in eigenvector estimation due to correlation of successive RLS channel estimates leads to a deterioration in the performance of the algorithm compared. Nonetheless, it was shown that CSF post-filtering of RLS estimates in low rank channels results in improved tracking performance. The following chapter presents an adaptive CSF algorithm that could be used in an experimental scenario and uses some of the results presented in this chapter to assess the performance of the adaptive CSF algorithm.

Chapter 4

Adaptive CSF algorithm

In the previous chapter, the performance of the CSF algorithm was analyzed. While comparing the theoretical performance of the CSF algorithm with the simulation results, it was assumed that the channel subspace filtering coefficients could be calculated using (3.7) based on apriori knowledge of the channel parameters. Once the CSF coefficients were computed, the CSF post-filter $\mathbf{F} = \mathbf{U}\Sigma_{ff}\mathbf{U}^H$ was calculated using either apriori known or estimated eigenvectors. The impact of errors in eigenvector estimation on the performance of the CSF algorithm was discussed in detail and examined using simulated data. The discussion provided insight into the factors that affected the performance of the CSF algorithm.

In a realistic scenario, however, it is not possible to compute the CSF coefficients apriori since the channel parameters are generally unknown. Hence, even if the eigenvectors were estimated from the channel estimate time series, it would not be possible to compute the CSF post-filter $\mathbf{F}$ using (3.7). An adaptive CSF algorithm would eliminate the need for any explicit dependence on the apriori knowledge of the channel parameters and implicitly of the time-varying channel model. Naturally, such an adaptive algorithm would have to yield similar performance benefits as the previously discussed CSF algorithm. The following section details an analytical formulation that forms the basis for such an algorithm.

4.1 Analytical formulation

The MMSE post-filter $\mathbf{F} = \mathbf{R}_{\hat{h}\hat{h}}^{-1}\mathbf{R}_{\hat{h}h}$ was analytically derived using explicit knowledge of the expressions for the correlation matrices $\mathbf{R}_{\hat{h}\hat{h}}$ and $\mathbf{R}_{\hat{h}h}$. In a realistic scenario, neither the true channel impulse response $\underline{h}(n)$ nor its statistics are known apriori. The received data $y(n)$ represents the only source of information about the true channel impulse response and the n_0- step prediction error given by:

$$\xi_{n_0} = y(n+n_0) - \underline{\hat{h}}^H(n)\mathbf{F}\underline{x}(n+n_0) \tag{4.1}$$

is the surrogate for the channel estimation error. The problem of determining the CSF filter can be reformulated such that the deterministically computed channel subspace filter $\hat{\mathbf{F}}$ minimizes the n_0-step prediction error over a given block of received data. This can be expressed as:

$$\hat{\mathbf{F}} = \arg\min_{\mathbf{F}} \sum_{n} \| \, y(n+n_0) - \underline{\hat{h}}^H(n)\mathbf{F}\underline{x}(n+n_0) \, \|^2 \tag{4.2}$$

where $n_0 \geq 1$ ensures that the transmitted data vector $\underline{x}(n+n_0)$ is independent of the RLS channel estimate $\underline{\hat{h}}(n)$. In the absence of any imposed structure, computing the $N \times N$ elements of the matrix $\hat{\mathbf{F}}$ in the deterministic manner suggested above would be constrained by finite number samples of received data. This limitation gets more pronounces as the number of taps N increases. For example, in a system with $N = 35$ taps, there would at least have to be $35 \times 35 = 1225$ received data samples just to avoid an under determined system of equations! In a realistic experimental scenario, hundreds of taps are used to represent the channel impulse response vector. In such cases, an adaptive CSF algorithm that attempts to compute each of the $N \times N$ individual elements of $\mathbf{F}$ using the deterministic formulation in (4.2) would converge at an unreasonably slow rate. If, however, the adaptive CSF filter exploits the channel subspace structure so that $\hat{\mathbf{F}} = \hat{\mathbf{U}}\hat{\Sigma}_{ff}\hat{\mathbf{U}}^H$ where $\hat{\mathbf{U}}$ are the eigenvectors of the estimated channel estimate correlation matrix given in (3.8) then the post-filter

can be written in terms of its modal decomposition:

$$\hat{\mathbf{F}} = \sum_{i=1}^{r} \sigma_{fi} \hat{\underline{u}}_i \hat{\underline{u}}_i^H \tag{4.3}$$

where $\hat{\underline{u}}_i$ is the i^{th} eigenvector of the estimated channel estimate impulse response correlation matrix $\hat{\mathbf{R}}_{\hat{h}\hat{h}}$, σ_{fi} is the corresponding CSF coefficient and $r \leq N$ is the conjectured reduced rank of the channel. The eigenvector $\hat{\underline{u}}_i$ is equivalently the i^{th} column of the matrix $\hat{\mathbf{U}}$ where $\hat{\mathbf{R}}_{\hat{h}\hat{h}} = \hat{\mathbf{U}}\hat{\Sigma}_{\hat{h}\hat{h}}\hat{\mathbf{U}}^H$ and from (3.8):

$$\hat{\mathbf{R}}_{\hat{h}\hat{h}} = \frac{1}{M_u} \sum_{k} \hat{\underline{h}}(k) \hat{\underline{h}}^H(k) \tag{4.4}$$

The adaptive CSF algorithm is classified as causal or non-causal depending on whether causal or non-causal data is used to estimate the eigenvectors and the CSF coefficients in (4.2). Using the model decomposition of the CSF in (4.3), (4.2) can be re-expressed as:

$$diag(\hat{\Sigma}_{ff}) = \underset{diag(\Sigma_{ff})}{\arg\min} \sum_{n} \| \, y(n+n_0) - \sum_{i=1}^{r} \left(\sigma_{fi} \hat{\underline{h}}^H(n) \hat{\underline{u}}_i \hat{\underline{u}}_i^H \underline{x}(n+n_0) \right) \|^2 \tag{4.5}$$

where $\hat{\mathbf{F}} = \hat{\mathbf{U}}\hat{\Sigma}_{ff}\hat{\mathbf{U}}^H$. This equation can be solved deterministically in a causal or non-causal manner. Regardless of the causality of the solution, introducing the following variables allows (4.5) to be posed as traditional least-squares problem:

$$\underline{\sigma}_f = [\sigma_{f1}, \sigma_{f2} \ldots \sigma_{fr}]^H \tag{4.6}$$

$$\underline{g}(n+n_0) = \begin{bmatrix} \hat{\underline{h}}^H(n) \hat{\underline{u}}_1 \\ \ldots \\ \hat{\underline{h}}^H(n) \hat{\underline{u}}_r \end{bmatrix} .* \begin{bmatrix} \hat{\underline{u}}_1^H \underline{x}(n+n_0) \\ \ldots \\ \hat{\underline{u}}_r^H \underline{x}(n+n_0) \end{bmatrix} \tag{4.7}$$

$$\tag{4.8}$$

where r is the conjectured rank of the channel and $(.*)$ denotes the element by element multiplication of the vectors. Using these variables, the deterministic least squares

problem can be posed as:

$$\underline{\hat{\sigma}}_f = \arg\min_{\underline{\sigma}_f} \sum_n \| y(n+n_0) - \underline{\sigma}_f^H \underline{g}(n+n_0) \|^2 \tag{4.9}$$

whose conjectured rank r dependent solution is given by:

$$\underline{\hat{\sigma}}_f = \mathbf{R}_{gg}(n)^{-1}\mathbf{R}_{gy}(n) \tag{4.10}$$

where:

$$\mathbf{R}_{gg}(n) = \frac{1}{M_a}\sum_n \underline{g}(n+n_0)\underline{g}^H(n+n_0) \tag{4.11}$$

$$\mathbf{R}_{gy}(n) = \frac{1}{M_a}\sum_n \underline{g}(n+n_0)y^*(n+n_0) \tag{4.12}$$

and M_a is an appropriate number of estimates used to compute the above matrices. Hence the CSF filter (4.3) is given by:

$$\hat{\mathbf{F}} = \sum_{i=1}^{r} \hat{\sigma}_{fi}\hat{\underline{u}}_i\hat{\underline{u}}_i^H \tag{4.13}$$

The post-processed channel estimate $\underline{\hat{h}}_p(n)$ is given by:

$$\underline{\hat{h}}_p(n) = \hat{\mathbf{F}}\underline{h}(n) \tag{4.14}$$

and the resulting n_0-step prediction error is given by:

$$\xi(n+n_0 \mid n) = y(n+n_0) - \underline{\hat{h}}_p^H(n)\underline{x}(n+n_0) \tag{4.15}$$

The adaptive CSF solution given by (4.10) depends on the conjectured rank r. If the statistics of the system are stationary then the adaptive CSF solution corresponding to the true rank of the channel r^* results in the minimum n_0-step prediction error.The adaptive CSF coefficients computed in (4.10) do not, at first glance, seem to equal the CSF coefficients computed in (2.70). The following section explicitly reconciles

this apparent difference between the two forms of the algorithm.

4.2 Relationship with theoretical CSF algorithm

(4.6) - (4.10) outline the adaptive CSF algorithm. Even though the matrices $\mathbf{R}_{gg}(n)$ and $\mathbf{R}_{gy}(n)$ are correlated, they are computed separately in an algorithmic implementation. Hence their expected values will be analyzed individually. Furthermore, the expected value of the matrix $\mathbf{R}_{gg}(n)$ is analyzed instead of that of its inverse. The $r \times r$ matrix $\mathbf{R}_{gg}(n)$ is experimentally computed as:

$$\mathbf{R}_{gg}(n) = \frac{1}{M_a} \sum_n \underline{g}(n+n_0)\underline{g}^H(n+n_0) \tag{4.16}$$

while the $r \times 1$ vector $\mathbf{R}_{gy}(n)$ is computed as:

$$\mathbf{R}_{gy}(n) = \frac{1}{M_a} \sum_n \underline{g}(n+n_0)y^*(n+n_0) \tag{4.17}$$

where $\underline{g}(n)$ is given by (4.7) and $y(n)$ is the received data. The summation is carried out over a rectangular causal or anti-causal window of size M_a. Based on (4.16) and substituting the expression for $\underline{g}(n+n_0)$ in (4.7), the $< i,j >^{th}$ element of the $r \times r$ matrix $\mathbf{R}_{gg}(n)$ can be individually computed as:

$$\Big(\mathbf{R}_{gg}(n)\Big)_{<i,j>} = \frac{1}{M_a} \sum_n \Big(\hat{\underline{h}}^H(n)\hat{\underline{u}}_i\hat{\underline{u}}_i^H\underline{x}(n+n_0)\underline{x}^H(n+n_0)\hat{\underline{u}}_j\hat{\underline{u}}_j^H\hat{\underline{h}}(n)\Big) \tag{4.18}$$

while the $< i >^{th}$ element of the $r \times 1$ vector $\mathbf{R}_{gy}(n)$ is given by:

$$\Big(\mathbf{R}_{gy}(n)\Big)_{<i>} = \frac{1}{M_a} \sum_n \Big(\hat{\underline{h}}^H(n)\hat{\underline{u}}_i\hat{\underline{u}}_i^H\underline{x}(n+n_0)\Big(\underline{x}^H(n+n_0)\underline{h}(n+n_0) + w^*(n+n_0)\Big)\Big) \tag{4.19}$$

where $\hat{\underline{u}}_i$ and $\hat{\underline{u}}_j$ are the i^{th} and j^{th} eigenvectors of the estimated channel impulse correlation matrix $\hat{\mathbf{R}}_{\hat{h}\hat{h}}$ respectively and $y(n)$ is given by (2.2). Since the additive observation noise $w(n)$ is uncorrelated with the true channel impulse response $\underline{h}(n)$ and the transmitted data vector $\underline{x}(n+n_0)$is assumed to be white so that $E[\underline{x}(n)\underline{x}^H(n)] = \mathbf{I}$

and independent of the channel estimate $\underline{\hat{h}}(n)$, the expected value of the expressions in (4.18) and (4.19) can be written as :

$$\begin{aligned} E[\Big(\mathbf{R}_{gg}(n)\Big)_{<i,j>}] &= E[\frac{1}{M_a}\sum_n \Big(\underline{\hat{h}}^H(n)\underline{\hat{u}}_i\underline{\hat{u}}_i^H\underline{x}(n+n_0)\underline{x}^H(n+n_0)\underline{\hat{u}}_j\underline{\hat{u}}_j^H\underline{\hat{h}}(n)\Big)] \\ &= \frac{1}{M_a}\sum_n E[\Big(\underline{\hat{h}}^H(n)\underline{\hat{u}}_i\underline{\hat{u}}_i^H\underbrace{\underline{x}(n+n_0)\underline{x}^H(n+n_0)}_{\mathbf{I}}\underline{\hat{u}}_j\underline{\hat{u}}_j^H\underline{\hat{h}}(n)\Big)] \\ &= \frac{1}{M_a}\sum_n E[\Big(\underline{\hat{h}}^H(n)\underline{\hat{u}}_i\underline{\hat{u}}_i^H\underline{\hat{u}}_j\underline{\hat{u}}_j^H\underline{\hat{h}}(n)\Big)] \end{aligned} \tag{4.20}$$

$$\begin{aligned} E[\Big(\mathbf{R}_{gy}(n)\Big)_{<i>}] &= E[\frac{1}{M_a}\sum_n \Big(\underline{\hat{h}}^H(n)\underline{\hat{u}}_i\underline{\hat{u}}_i^H\underline{x}(n+n_0)\Big(\underline{x}^H(n+n_0)\underline{h}(n+n_0)+w(n+n_0)\Big)\Big)] \\ &= \frac{1}{M_a}\sum_n E[\Big(\underline{\hat{h}}^H(n)\underline{\hat{u}}_i\underline{\hat{u}}_i^H\underbrace{\underline{x}(n+n_0)\underline{x}^H(n+n_0)}_{\mathbf{I}}\underline{h}(n+n_0)\Big)] \\ &= \frac{1}{M_a}\sum_n E[\Big(\underline{\hat{h}}^H(n)\underline{\hat{u}}_i\underline{\hat{u}}_i^H\underline{h}(n+n_0)\Big)] \end{aligned} \tag{4.21}$$

Since the eigenvectors form the columns of $\hat{\mathbf{U}}$ and are mutually orthogonal such that:

$$\underline{\hat{u}}_i^H\underline{\hat{u}}_j = \delta_{ij} = \begin{cases} 1 & \text{if } i=j, \\ 0 & \text{if } i\neq j \end{cases} \tag{4.22}$$

the terms in (4.20) and (4.21) can be rearranged so that:

$$E[\Big(\mathbf{R}_{gg}(n)\Big)_{<i,j>}] = \frac{1}{M_a}\sum_n \Big(\underline{\hat{u}}_i^H\underbrace{E[\underline{\hat{h}}(n)\underline{\hat{h}}^H(n)]}_{\mathbf{R}_{\hat{h}\hat{h}}}\underline{\hat{u}}_i\Big)\delta_{ij} \tag{4.23}$$

$$\begin{aligned} E[\Big(\mathbf{R}_{gy}(n)\Big)_{<i>}] &= \frac{1}{M_a}\sum_n \Big(\underline{\hat{u}}_i^H E[\underline{\hat{h}}(n)\underline{h}^H(n+n_0)]\underline{\hat{u}}_i\Big) \\ &= \frac{1}{M_a}\sum_n \alpha^{-(n_0-1)}\Big(\underline{\hat{u}}_i^H\underbrace{E[\underline{\hat{h}}(n)\underline{h}^H(n+n)]}_{\mathbf{R}_{\hat{h}h}}\underline{\hat{u}}_i\Big) \end{aligned} \tag{4.24}$$

where from (2.1) $E[\underline{\hat{h}}(n)\underline{h}^H(n+n_0)] = \alpha^{-(n_0-1)}\underline{\hat{h}}(n)\underline{h}^H(n+n)]$. The underbraced term indicates substitution of the expected value of the argument in place of the

instantaneous value. Further rearranging of the terms in (4.23) and (4.24) will reveal the relationship between the form of the adaptive CSF algorithm and the theoretical MMSE post-filter. Exploiting the subspace structure as in the theoretical derivation and substituting $\hat{\mathbf{R}}_{\hat{h}\hat{h}} = \hat{\mathbf{U}}\Sigma_{\hat{h}\hat{h}}\hat{\mathbf{U}}^H$ and $\hat{\mathbf{R}}_{\hat{h}h} = \hat{\mathbf{U}}\Sigma_{\hat{h}h}\hat{\mathbf{U}}^H$ in (4.23) and (4.24) results in:

$$E[\Big(\mathbf{R}_{gg}(n)\Big)_{<i,j>}] = \frac{1}{M_a}\sum_n \Big(\underline{\hat{u}}_i^H\hat{\mathbf{U}}\Sigma_{\hat{h}\hat{h}}\hat{\mathbf{U}}^H\underline{\hat{u}}_i\Big)\delta_{ij}$$
$$= \begin{cases} \hat{\sigma}^2_{\hat{h}\hat{h}i} & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases} \tag{4.25}$$

$$E[\Big(\mathbf{R}_{gy}(n)\Big)_{<i>}] = \frac{\alpha^{-(n_0-1)}}{M_a}\sum_n \big(\underline{\hat{u}}_i^H\hat{\mathbf{U}}\Sigma_{\hat{h}h}\hat{\mathbf{U}}^H\underline{\hat{u}}_i\big) \tag{4.26}$$
$$= \alpha^{-(n_0-1)}\sigma^2_{\hat{h}hi}$$

so that the matrices $\mathbf{R}_{gg}(n)$ and $\mathbf{R}_{gy}(n)$ may be written as:

$$E[\mathbf{R}_{gg}(n)] = \Sigma_{\hat{h}\hat{h}} \tag{4.27}$$

$$E[\mathbf{R}_{gy}(n)] = \alpha^{-(n_0-1)}diag(\Sigma_{\hat{h}h}) \tag{4.28}$$

From (4.10), the computed adaptive CSF solution is obtained by separately computing $\mathbf{R}_{gg}(n)$ and $\mathbf{R}_{gy}(n)$ so that:

$$E[\mathbf{R}_{gg}(n)]^{-1}E]\mathbf{R}_{gy}(n)] = \alpha^{-(n_0-1)}\Sigma^{-1}_{\hat{h}\hat{h}}diag(\Sigma_{\hat{h}h}) \tag{4.29}$$

For $\alpha \approx 1$ and a small n_0, it would be expected that the estimated channel subspace post-filter $\hat{\mathbf{F}}$ is given by:

$$\hat{\mathbf{F}} \approx \mathbf{F}_{MMSE} \tag{4.30}$$

For a time-varying system with stationary statistics, the adaptive CSF post-filter is equivalent to the MMSE post-filter solution constructed using apriori knowledge of

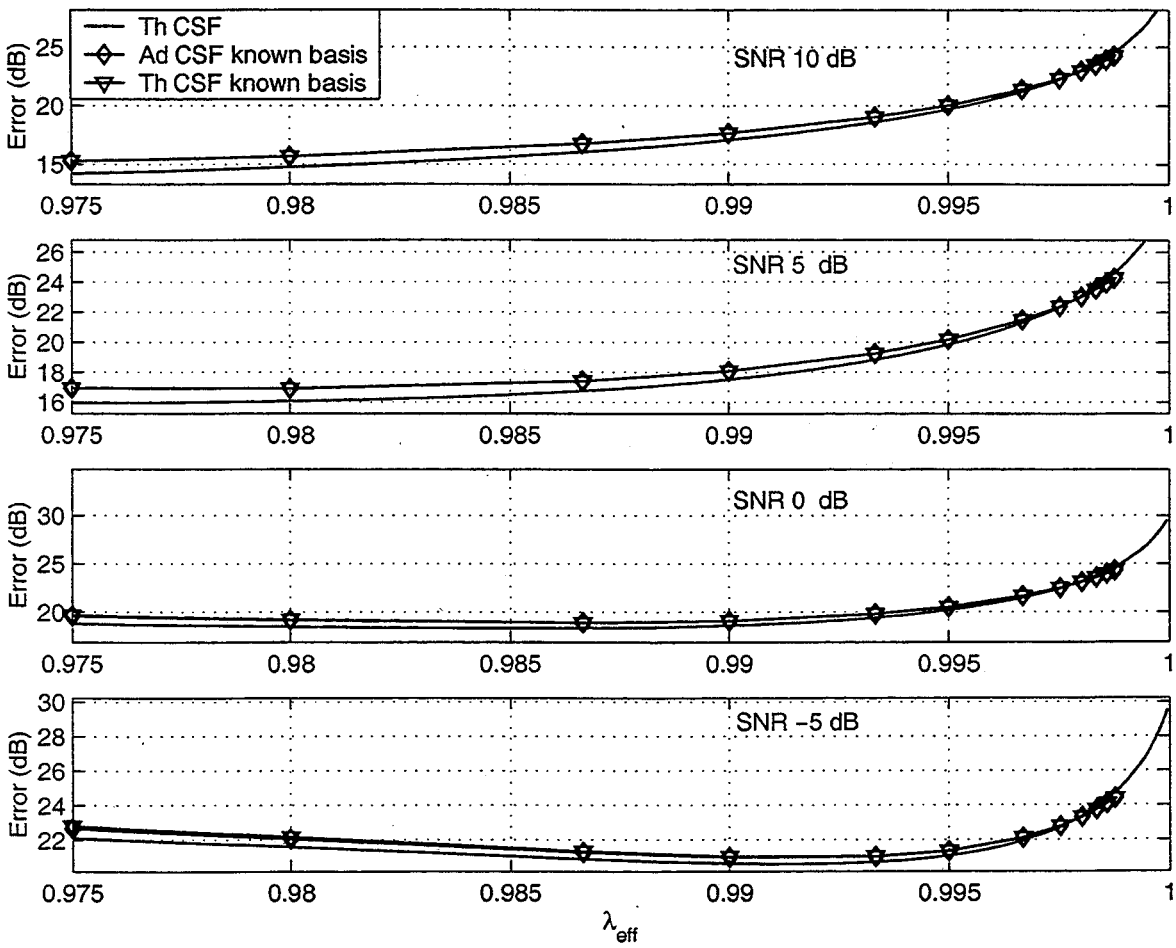

Figure 4-1: Theoretically computed CSF vs Adaptive CSF with known eigenvectors for EW-RLS

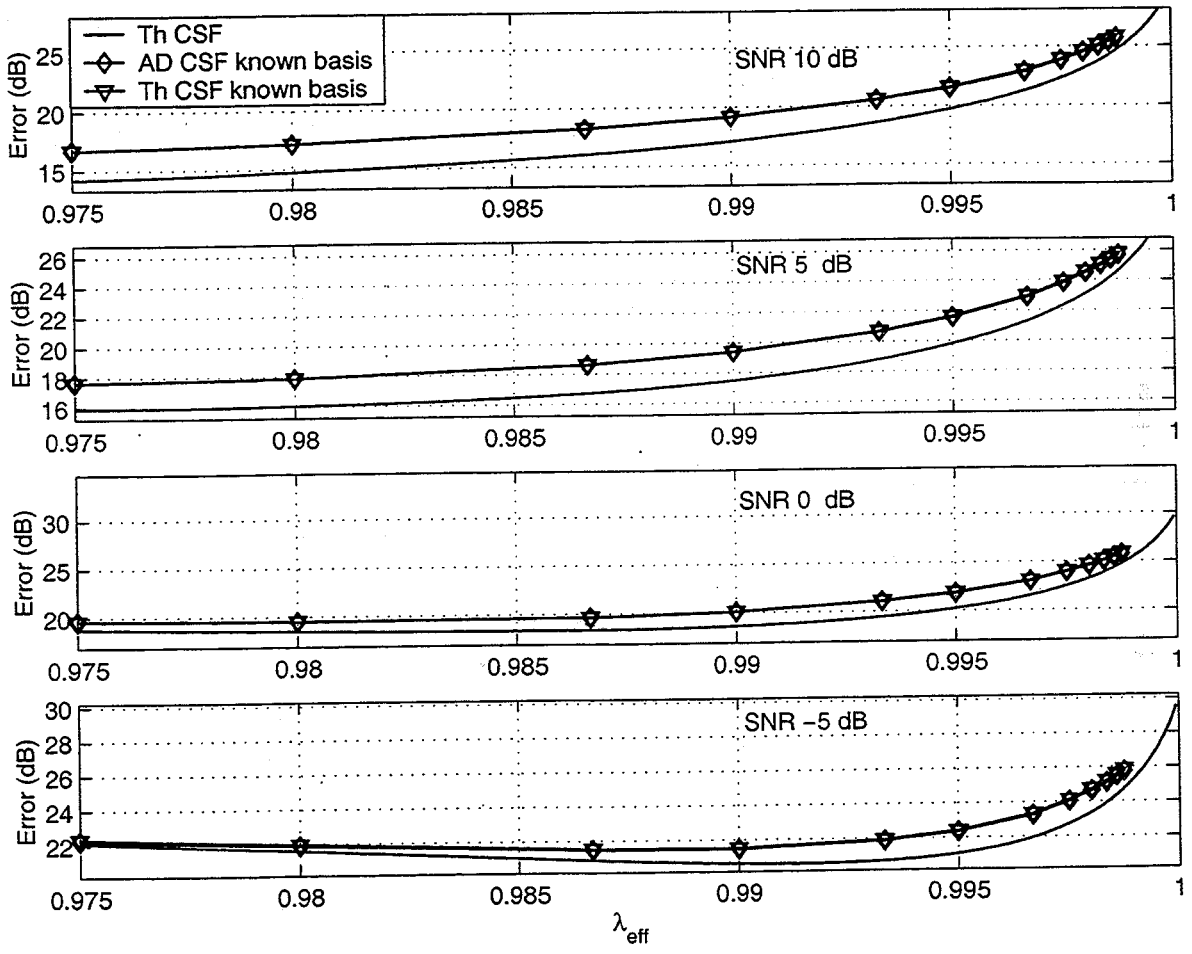

Figure 4-2: Theoretically computed CSF vs Adaptive CSF with known eigenvectors for SW-RLS

the channel model and relevant statistics. Figure 4-1 compares the tracking performance of the adaptive CSF (Ad CSF known basis) and the theoretically computed CSF (Th CSF known basis) algorithm for the EW-RLS channel estimates assuming that the eigenvectors are known apriori. Similarly, figure 4-2 compares the tracking performance of the adaptive CSF (Ad CSF known basis) and the theoretically computed CSF (Th CSF known basis) algorithm for the SW-RLS channel estimates assuming that the eigenvectors are known apriori. The analytically computed performance curve (Th CSF) is also plotted for comparison. As can be seen, the demonstrated theoretical equivalence between the adaptive and theoretical CSF algorithms is apparent in the simulated data for both the EW-RLS and the SW-RLS algorithms. The following section considers the impact of channel eigenvector estimation errors on the performance of the adaptive CSF algorithm.

4.3 Impact of eigenvector estimation errors on performance

In a realistic scenario, the eigenvectors used to compute the CSF post-filter $\mathbf{F}$ are not known apriori either. As discussed in the previous chapter, finite number of samples of received data results in errors in estimating the channel estimate correlation matrix and leads to eigenvector estimation errors. The estimated channel subspace filter is computed as:

$$\hat{\mathbf{F}} = \hat{\mathbf{U}}\hat{\Sigma}_{ff}\hat{\mathbf{U}}^H \tag{4.31}$$

Due to errors in eigenvector estimation, the estimated eigenvectors may be written in terms of the true eigenvectors as:

$$\begin{aligned} \hat{\mathbf{U}} &= \left(\hat{\mathbf{U}} - \mathbf{U}\right) + \mathbf{U} \\ &= \mathbf{U} + \Delta_u \end{aligned} \tag{4.32}$$

where Δ_u represents the error in eigenvector estimation due to finite number of independent channel estimates. Additionally, due to errors in the estimation of the channel subspace filtering coefficients, the error in the estimated CSF coefficients may be written as:

$$\begin{aligned}\hat{\Sigma}_f &= \left(\hat{\Sigma}_{ff} - \Sigma_{ff}\right) + \Sigma_{ff} \\ &= \Sigma_{ff} + \Delta_\Sigma\end{aligned} \tag{4.33}$$

The estimated channel subspace filter can be rewritten as:

$$\begin{aligned}\hat{\mathbf{F}} &= (\mathbf{U} + \Delta_u)(\Sigma_{ff} + \Delta_\Sigma)(\mathbf{U} + \Delta_u)^H \\ &= \underbrace{\mathbf{U}\Sigma_{ff}\mathbf{U}^H}_{\mathbf{F}} + \Delta_u\Sigma_{ff}\mathbf{U}^H + \mathbf{U}\Sigma_{ff}\Delta_u^H + \Delta_u\Sigma_{ff}\Delta_u^H \\ &+ \quad \mathbf{U}\Delta_{\Sigma_{ff}}\mathbf{U}^H + \Delta_u\Delta_{\Sigma_{ff}}\mathbf{U}^H + \mathbf{U}\Delta_{\Sigma_{ff}}\Delta_u^H + \Delta_u\Delta_{\Sigma_{ff}}\Delta_u^H \\ &= \mathbf{F} + \Delta_f\end{aligned} \tag{4.34}$$

where Δ_f represents the error in computing the channel subspace filter due to eigenvector estimation and the resultant incorrect estimation of the CSF coefficients and is given by:

$$\begin{aligned}\Delta_f &= \Delta_u\Sigma_{ff}\mathbf{U}^H + \mathbf{U}\Sigma_{ff}\Delta_u^H + \Delta_u\Sigma_{ff}\Delta_u^H \\ &+ \quad \mathbf{U}\Delta_{\Sigma_{ff}}\mathbf{U}^H + \Delta_u\Delta_{\Sigma_{ff}}\mathbf{U}^H + \mathbf{U}\Delta_{\Sigma_{ff}}\Delta_u^H + \Delta_u\Delta_{\Sigma_{ff}}\Delta_u^H\end{aligned} \tag{4.35}$$

The estimated channel subspace filter in (4.34) is used to post-filter the least-squares channel estimate $\underline{\hat{h}}(n)$ so that:

$$\underline{\hat{h}}_{\hat{p}}(n) = \hat{\mathbf{F}}\underline{\hat{h}}(n) \tag{4.36}$$

(4.36) can be rewritten in terms of the error in computing the channel subspace filter by substituting the expression for $\hat{\mathbf{F}}$ in (4.34):

$$\begin{aligned}\underline{\hat{h}}_{\hat{p}}(n) &= (\mathbf{F}+\Delta_f)\underline{\hat{h}}(n-1) \\ &= \mathbf{F}\underline{\hat{h}}(n-1)+\Delta_f\underline{\hat{h}}(n-1) \\ &= \underline{\hat{h}}_p(n)+\underline{\delta h}_p(n)\end{aligned} \tag{4.37}$$

where $\underline{\hat{h}}_p(n) = \mathbf{F}\underline{\hat{h}}(n-1)$ is the post-filtered channel estimate assuming perfect estimation of the channel subspace filter and $\underline{\delta h}_p(n)$ is the error in the post-filtered channel estimate due to imperfect eigenvector estimation. The resultant tracking error vector due to the use of imperfect eigenvector estimates and imperfect channel subspace filter coefficients in estimating the channel subspace filter $\hat{\mathbf{F}}$ is given by:

$$\underline{\hat{\epsilon}}_f(n) = \underline{h}(n) - \underline{\hat{h}}_{\hat{p}}(n) \tag{4.38}$$

Substituting the expression for $\underline{\hat{h}}_{\hat{p}}(n)$ from (4.37) in (4.38) results in:

$$\begin{aligned}\underline{\hat{\epsilon}}_f(n) &= \underline{h}(n) - \underline{h}_{\hat{p}}(n) - \underline{\delta h}_f(n) \\ &= \underline{\epsilon}_f(n) - \underline{\delta h}_p(n)\end{aligned} \tag{4.39}$$

The tracking error correlation matrix is then given by:

$$\mathbf{R}_{\hat{\epsilon}_f\hat{\epsilon}_f} = \mathbf{R}_{\epsilon_f\epsilon_f} + \mathbf{R}_{\delta h_p\delta h_p} - \mathbf{R}_{\epsilon_f\delta h_p} - \mathbf{R}_{\epsilon_f\delta h_p} \tag{4.40}$$

so the resultant mean square tracking error is:

$$\begin{aligned}E[\| \underline{\hat{\epsilon}}_f(n) \|^2] &= \mathbf{R}_{\hat{\epsilon}_f\hat{\epsilon}_f} \\ &= tr(\mathbf{R}_{\epsilon_f\epsilon_f}) + tr(\mathbf{R}_{\delta h_p\delta h_p}) - tr(\mathbf{R}_{\epsilon_f\delta h_p}) - tr(\mathbf{R}_{\epsilon_f\delta h_p})\end{aligned} \tag{4.41}$$

where $\underline{\epsilon}_f(n)$ is the tracking error vector if there were no errors in eigenvector estimation and $\mathbf{R}_{\epsilon_f\epsilon_f}$ is the corresponding tracking error correlation matrix. The additional correlation matrix terms in (4.40) represent the error incurred due to errors in eigen-

vector estimation. The correlation matrix $\mathbf{R}_{\epsilon_f \delta h_p}$ can be expressed in terms of the error in eigenvector and filter coefficient estimation using (4.35) as:

$$\begin{aligned}
\mathbf{R}_{\epsilon_f \delta h_p} &= E[\underline{\epsilon}_f(n)\underline{\delta h}_p^H(n-1)] \\
&= E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\Delta_f^H] \\
&= E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\mathbf{U}\Sigma_{ff}\Delta_u^H] + E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\Delta_u^H\Sigma_{ff}\mathbf{U}^H] \\
&+ E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\Delta_u\Sigma_{ff}\Delta_u^H] + E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\mathbf{U}\Delta_{\Sigma_{ff}}\mathbf{U}^H \\
&+ E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\Delta_u\Delta_{\Sigma_{ff}}\mathbf{U}^H + E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\mathbf{U}\Delta_{\Sigma_{ff}}\Delta_u^H \\
&+ E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\Delta_u\Delta_{\Sigma_{ff}}\Delta_u^H
\end{aligned} \tag{4.42}$$

As can be seen from (4.42), the correlation matrix $\mathbf{R}_{\epsilon_f \delta h_p}$ has terms dependent on $\mathbf{U}$, Δ_u, $\Delta_{\Sigma_{ff}}$ and Σ_{ff}. Since the eigenvectors $\hat{\mathbf{U}}$ used to construct the channel subspace filter $\hat{\mathbf{F}}$ in (4.31) are estimated from the channel estimate time series according to (3.8) and (3.9), the eigenvector estimation error term Δ_u is in general correlated with the channel estimate $\hat{\underline{h}}(n-1)$. By extension, it is also correlated with the ideal post-filtered tracking error vector $\underline{\epsilon}_f(n) = \underline{h}(n) - \hat{\mathbf{F}}\hat{\underline{h}}(n-1)$. Additionally, errors in eigenvector estimation result in errors in the estimation of the CSF coefficients. The errors in estimating the CSF coefficients $\Delta_{\Sigma_{ff}}$ are even harder to characterize because of the complex data dependencies of the AD-CSF algorithm. Hence, computing the expected value of the correlation matrix $\mathbf{R}_{\epsilon_f \delta h_p}$ analytically is a considerably difficult task. In the previous chapter, an expression for the number of successive correlated estimates was derived as a function of the system parameters. Suppose it is assumed that the system is in steady state and that a relatively large number of independent channel estimates are available. Since (3.24) indicates that only a finite number of successive channel estimates are correlated, the errors estimated channel correlation matrix in (3.8) and hence the estimated eigenvectors (3.9) may be assumed to be independent of the instantaneous channel estimate $\hat{\underline{h}}(n-1)$ and by extension the instantaneous channel estimation error vector $\underline{\epsilon}(n)$. The error in estimating the CSF filter Δ_f in (4.35) may thus also be assumed to be independent of the instantaneous channel estimate $\hat{\underline{h}}(n-1)$ and the instantaneous channel estimation error vector $\underline{\epsilon}(n)$.

The correlation matrix $\mathbf{R}_{\epsilon_f \delta h_p}$ may now expressed as:

$$\begin{aligned}
\mathbf{R}_{\epsilon_f \delta h_p} &= E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\Delta_f^H] \\
&= E[E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)\Delta_f^H \mid \Delta_f]] \\
&= E[\underbrace{E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)]}_{\mathbf{0}}\Delta_f^H] \\
&= \mathbf{0}
\end{aligned} \tag{4.43}$$

where $E[\underline{\epsilon}_f(n)\hat{\underline{h}}^H(n-1)] = \mathbf{0}$ is a consequence of the fact that $\mathbf{F} = \mathbf{R}_{\hat{h}\hat{h}}^{-1}\mathbf{R}_{\hat{h}h}$ is an MMSE post-filter. The result in (4.43) has been verified using simulation when a large number of channel estimates are used to estimate the eigenvectors. From (4.43), $\mathbf{R}_{\delta h_p \epsilon_f} = \mathbf{0}$. (4.40) may be rewritten as:

$$\mathbf{R}_{\hat{\epsilon}_f \hat{\epsilon}_f} = \mathbf{R}_{\epsilon_f \epsilon_f} + \mathbf{R}_{\delta h_p \delta h_p} \tag{4.44}$$

and the resultant channel estimation error is given by:

$$\begin{aligned}
E[\mid \hat{\underline{\epsilon}}_f(n) \mid^2] &= tr(\mathbf{R}_{\hat{\epsilon}_f \hat{\epsilon}_f}) \\
&= tr(\mathbf{R}_{\epsilon_f \epsilon_f}) + tr(\mathbf{R}_{\delta h_p \delta h_p}) \\
&= E[\mid \underline{\epsilon}_f(n) \mid^2] + tr(\mathbf{R}_{\delta h_p \delta h_p})
\end{aligned} \tag{4.45}$$

where $\mathbf{R}_{\delta h_p \delta h_p}$ is given by:

$$\mathbf{R}_{\delta h_p \delta h_p} = E[\Delta_f \hat{\underline{h}}(n-1)\hat{\underline{h}}^H(n-1)\Delta_f^H] \tag{4.46}$$

(4.45) has terms which depend on the eigenvector estimation error Δ_u. From the discussion in a previous section, it was concluded that the estimated channel estimate correlation matrix in (3.8) convergence asymptotically in the mean square sense to the true channel estimate correlation matrix. Since the eigenvectors $\hat{\mathbf{U}}$ are obtained from the estimated channel estimate correlation matrix as given in (3.9), the estimated eigenvectors also converge asymptotically in the mean square sense to the true

eigenvectors. Thus the error in eigenvector estimation $\Delta_u \rightarrow 0$ as $M_u \rightarrow \infty$ where $M_u = M_{data}/W(\alpha, \lambda, SNR)$ is the number of independent channel estimates available when the number of available channel estimates is given by M_{data} and the correlation window size is given by $W(\lambda, SNR)$. As expressed in (4.45), the performance of the AD-CSF algorithm approaches the theoretical performance predicted in (2.73) as $M_u \rightarrow \infty$.

Figure 4-3 compares the tracking performance of the adaptive CSF using known (Ad CSF known basis) and estimated eigenvectors (Ad CSF est basis) with the theoretically computed CSF algorithm using known eigenvectors (Th CSF known basis) for the EW-RLS channel estimates. Similarly, figure 4-4 compares the tracking performance of the adaptive CSF using known (Ad CSF known basis) and estimate eigenvectors (Ad CSF est basis) with the theoretically computed CSF algorithm using known eigenvectors (Th CSF known basis)for the SW-RLS channel estimates. For both these sets of figures, the analytical performance curve for the CSF algorithm (Th CSF) is also plotted for comparison. As can be seen, the demonstrated theoretical equivalence between the adaptive and theoretical CSF algorithms is apparent in the simulated data as well. The "Ad CSF est basis" refer to an adaptive CSF algorithm where the eigenvectors were estimated using 42,000 channel estimates. For such a situation, the errors in eigenvector estimation would be expected to be minimal and corresponding, the Ad CSF solution with estimated eigenvectors performs almost as well as the Ad CSF solution with known eigenvectors.

Analogous to the discussion in the previous chapter, errors in eigenvector estimation and hence in CSF coefficient estimation due to finite data and the correlation between successive channel estimates leads to a deterioration in the performance of the adaptive CSF algorithm. Figures 4-5 shows the deterioration in performance of the CSF algorithm when the number of data samples are used estimate the CSF coefficients for the EW-RLS channel estimates is decreased even when the eigenvectors are known perfectly. Figure 4-6 shows the resulting deterioration in performance in the adaptive CSF algorithm when the number of data samples used to estimate the CSF coefficients for the EW-RLS channel estimates is reduced and the eigenvectors

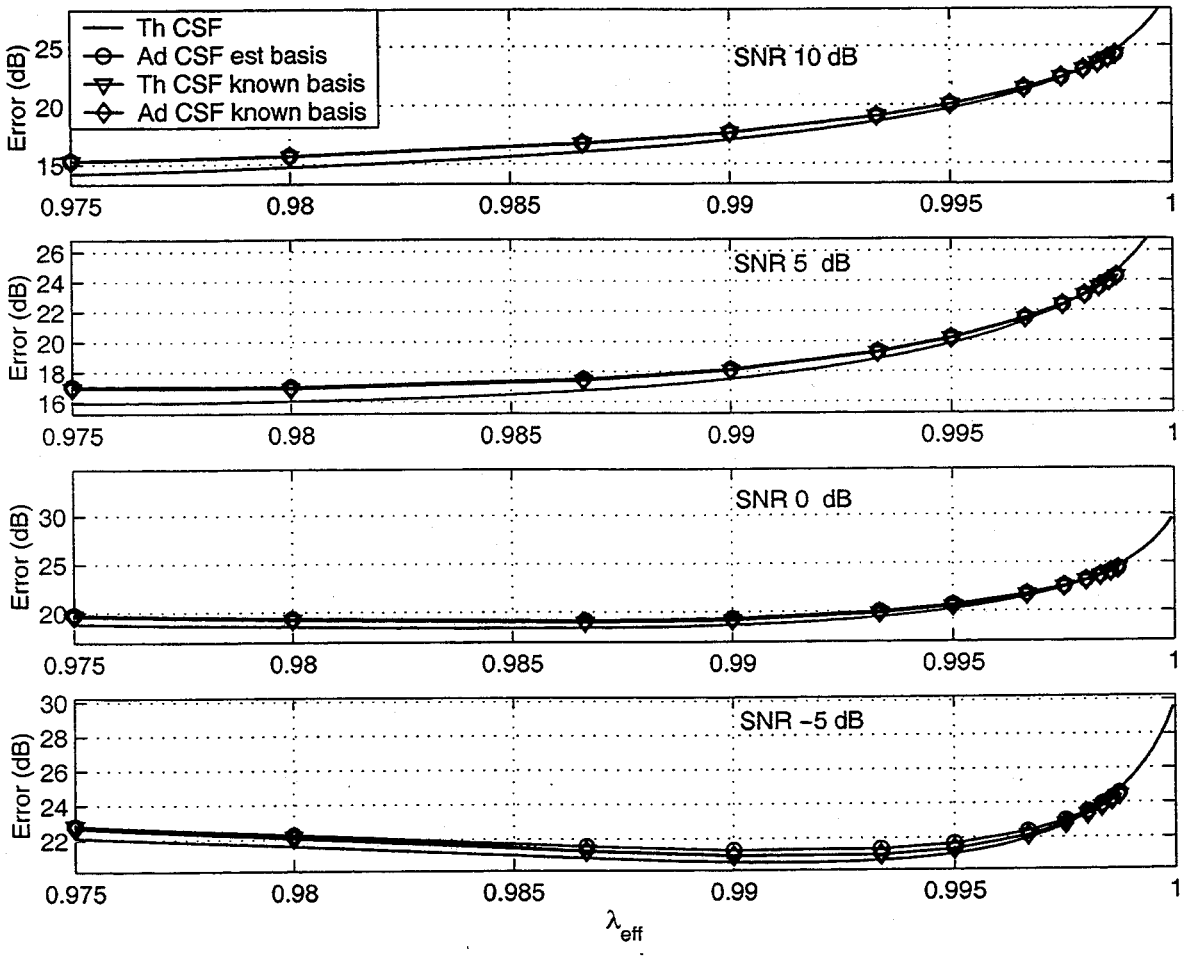

Figure 4-3: Theoretically computed CSF with known eigenvectors vs Adaptive CSF using known and estimated eigenvectors for EW-RLS algorithm

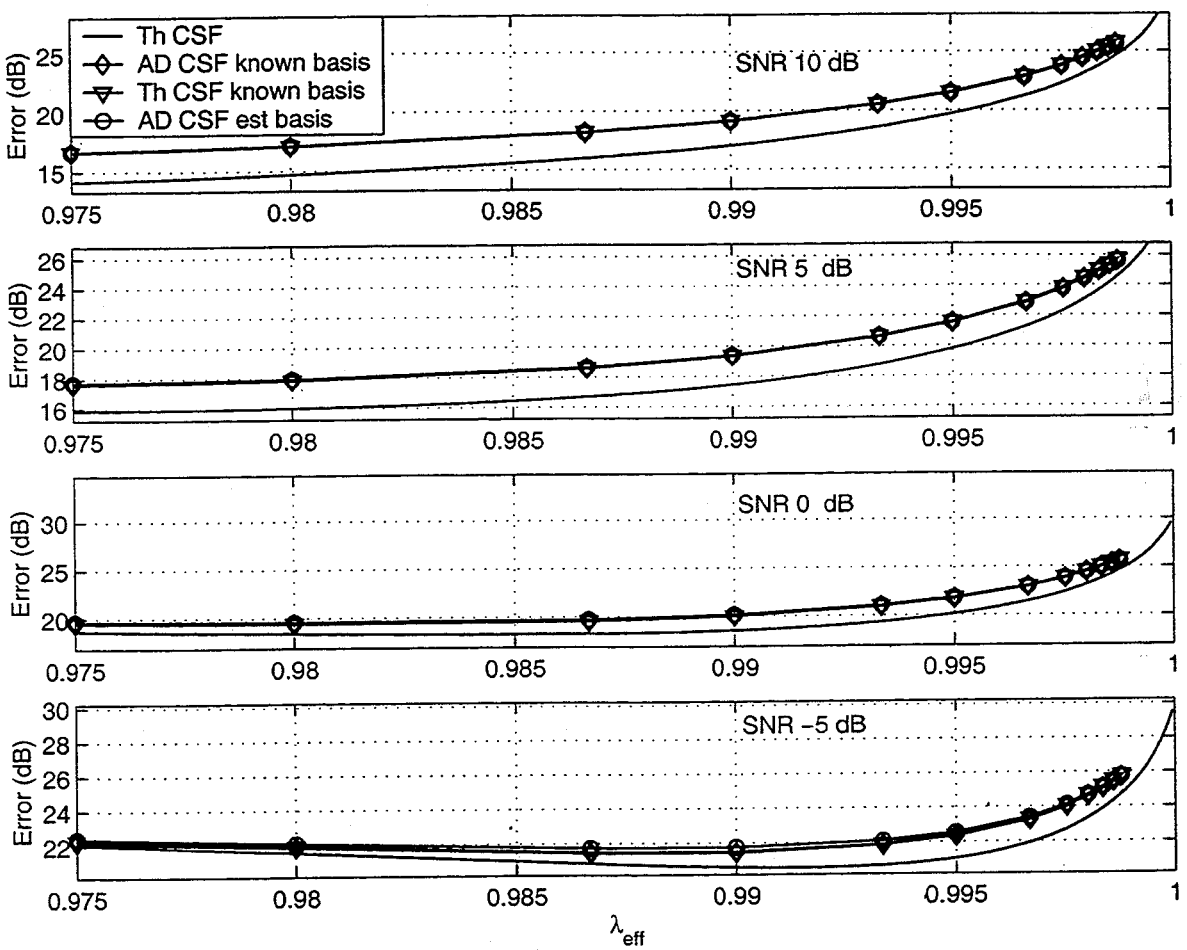

Figure 4-4: Theoretically computed CSF with known eigenvectors vs Adaptive CSF using known and estimated eigenvectors for SW-RLS algorithm

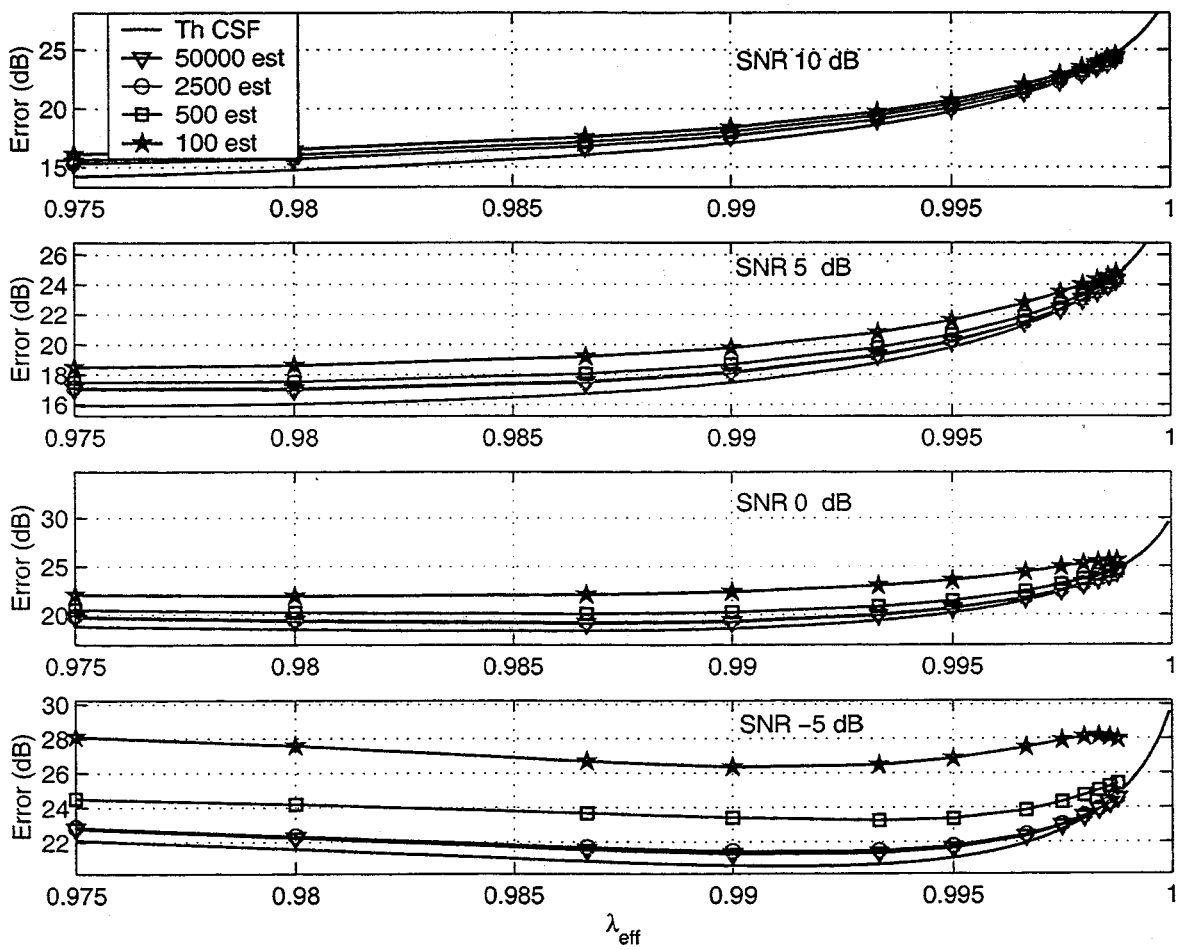

Figure 4-5: Effect of errors in CSF coefficient estimation on CSF performance for EW-RLS algorithm

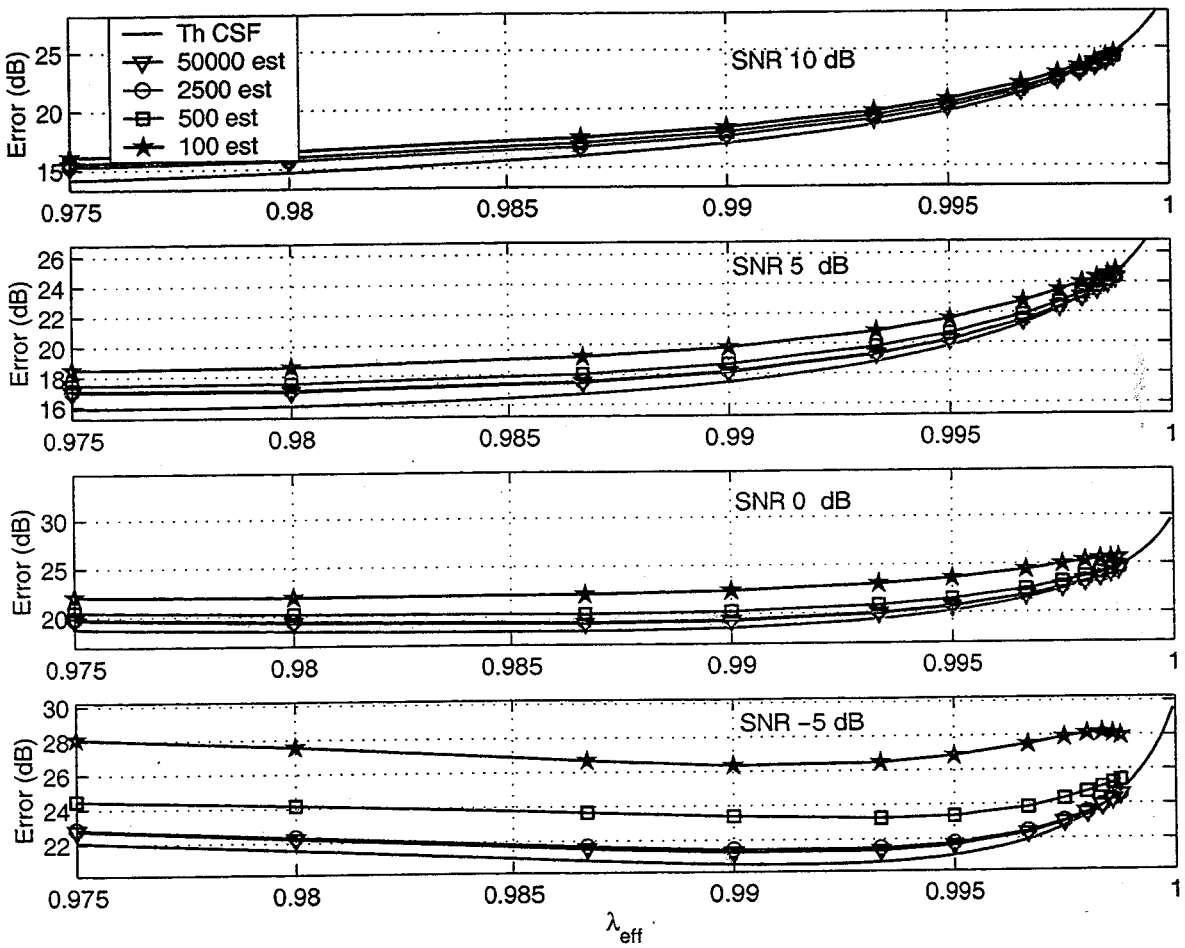

Figure 4-6: Effect of errors in eigenvector and CSF coefficient estimation error on CSF performance for EW-RLS algorithm

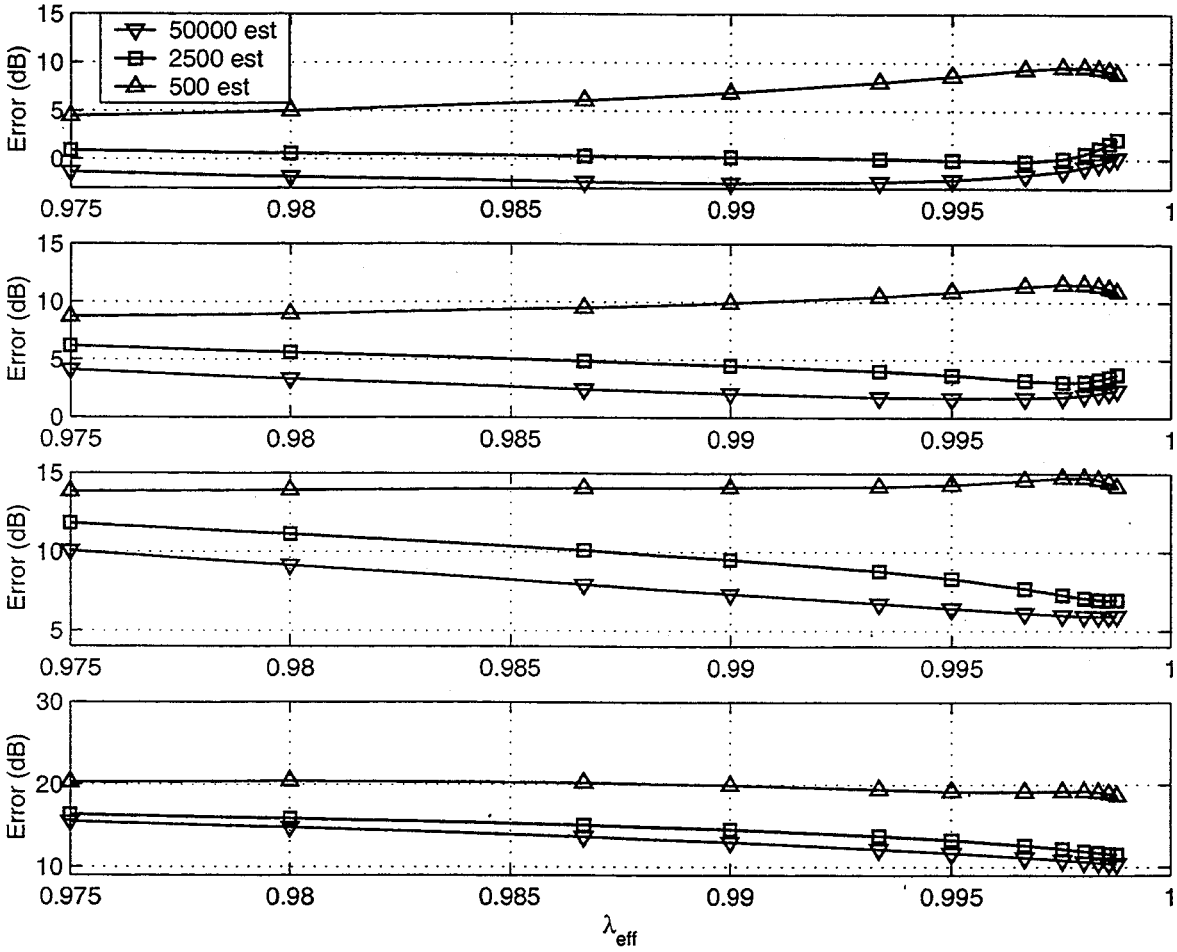

Figure 4-7: Effect of finite data used to estimate eigenvectors on $tr(\mathbf{R}_{\delta h_p \delta h_p})$ for EW-RLS algorithm

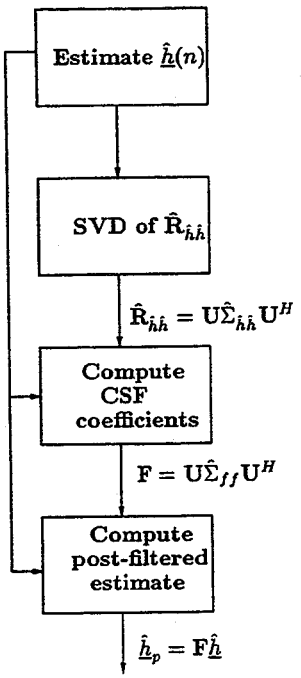

Figure 4-8: Methodology for adaptive CSF algorithm

are estimated using 42,000 channel estimate samples. Since the eigenvector estimation error due the use of that many samples is fairly negligible, figure 4-6) reaffirms the relatively minimal loss in performance over the variant of the Ad-CSF algorithm when the eigenvectors are known perfectly. Figure 4-7 shows the increase in the error term $tr(\mathbf{R}_{\delta h_p \delta h_p})$ as fewer number of channel estimates are used to compute the eigenvectors. The cross correlation term $\mathbf{R}_{\delta h_p \epsilon_f}$ was computed using simulations to be negligible, as assumed earlier.

4.4 Performance on experimental data

The adaptive CSF algorithm presented above was used to process experimental data collected in actual acoustic communication trials. The following sections present the results of processing this data over a range of operating channel conditions. This will serve to demonstrate that, even though the a simplified theoretical framework was initially used to formulate the algorithm, an adaptive CSF algorithm is able to exploit the channel subspace structure to improve the performance of the traditional

RLS algorithms. For the sake of simplicity but more importantly consistency in presentation, the performance of the SW-RLS algorithm has been used as a benchmark in evaluating any performance improvements. In selecting the performance of the SW-RLS algorithm as the benchmark, implicitly the assumption has been made that any performance improvement exhibited could be analogously demonstrated for the EW-RLS algorithm as well. Figure 4-8 shows the methodology of using the adaptive CSF algorithm on experimental data.

For the experimental results presented, the channel estimates are obtained using an SW-RLS algorithm. The channel estimate time series is used to compute the channel estimate correlation matrix as in (4.4) using either causal or non-causal channel estimates. The eigenvalue-decomposition of the channel estimate correlation matrix is then computed to yield the eigenvectors $\hat{\mathbf{U}}$. The adaptive CSF algorithm presented is used to compute the channel subspace filtering coefficients based on the conjectured rank r. The post-processed channel estimate $\underline{h}_p(n)$ is used to compute the n_0-step prediction error. In determining the optimal rank of the channel, performance curves of prediction error, averaged over a large enough data set, vs. rank are generated for the adaptive CSF. The minimum point of this performance curve corresponds to the optimal rank r^*. For comparison, a sub-optimal filter referred to as the abrupt rank reduction (ARR) filter $\mathbf{F}_{ARR} = \hat{\mathbf{U}}\Sigma_{ARR}\hat{\mathbf{U}}^H$ based on the same eigenvectors is also generated as a function of rank r such that:

$$\Sigma_{ARR} = \begin{pmatrix} \mathbf{I}_r & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \tag{4.47}$$

where $\mathbf{I}_r$ is an $r \times r$ identity matrix. Hence the ARR filter merely exploits the reduced dimensionality of the channel and does not exploit the correlation between the channel estimate and the channel estimation error. The following sections present the results of processing experimental data representing a wide range of channel conditions. The experimental data used is downsampled, basebanded, symbol matched and then processed using the SW-RLS and the CSF algorithms. The $n_0 = 3$ step

prediction error reflects the improvement in performance due to CSF.

4.4.1 NOA data

Figure 4-9 shows the channel estimate time series for the NOA continental shelf data. At 2 samples/symbol, the samples shown represent 2.5 kHz sampling frequency while the carrier frequency was 2.25 kHz. A 175 tap model is used to represent the time-varying system. Figure 4-10 shows the improvement in performance obtained by using a CSF approach. A rank 7 channel gives the best performance. Figure 4-11 shows the entire performance vs tracking parameter curve and demonstrates the dramatic improvement due to CSF. It can also be seen that both the causal and non-causal processing results in this improved performance. A comparison with the performance of the ARR filter, also demonstrates that it is the combination of exploiting the reduced dimensionality as well as the Wiener filtering of the subspaces that gives the improvement in performance. For subsequent results on experimental data, the adaptive CSF post-filter is computed non-causally. This is done for the sake of reducing computational complexity since the causal AD-CSF algorithm is enormously more complex than the non-causal AD-CSF algorithm. Even though the performance improvement thus realized might seem spurious, processing of NOA data both casually and non-causally has shown that the causal AD-CSF algorithm can achieve comparable performance as the non-causal version.

4.4.2 AOSN data

Figure 4-12 shows the channel estimate time series for the AOSN data. At 2 samples/symbol, the samples shown represent a 3.1 kHz sampling frequency while the carrier frequency was 15.5 kHz. A 180 tap model is used to represent the time-varying system. Figure 4-13 shows the improvement in performance obtained by using a CSF approach. A rank 7 channel gives the best performance. A comparison with the performance of the ARR filter, also demonstrates that it is the combination of exploiting the reduced dimensionality as well as the Wiener filtering of the subspaces that gives

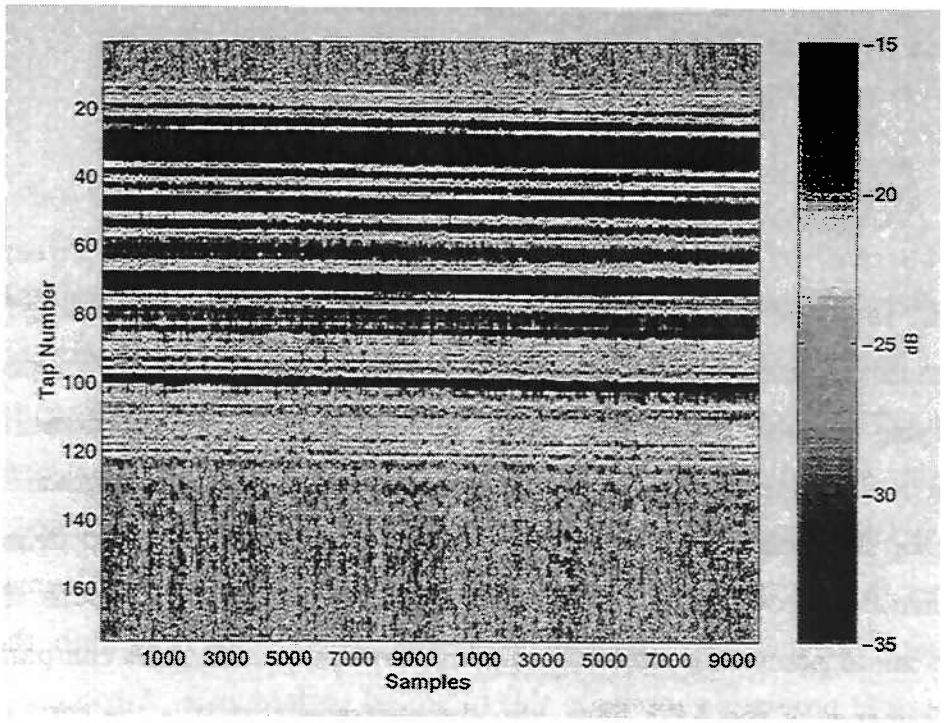

Figure 4-9: Channel estimate time series for NOA data

the improvement in performance.

4.4.3 SZATE data

Figure 4-14 shows the channel estimate time series for the SZATE surf zone data. At 2 samples/symbol, the samples shown represent a 24 kHz sampling frequency while the carrier frequency was 13.5 kHz. A 144 tap model is used to represent the time-varying system. Figure 4-15 shows the improvement in performance obtained by using a CSF approach. A rank 30 channel gives the best performance. A comparison with the performance of the ARR filter, also demonstrates, once again that it is the combination of exploiting the reduced dimensionality as well as the Wiener filtering of the subspaces that gives the improvement in performance.

4.5 Summary

An adaptive CSF algorithm was presented that was shown to be equivalent to the theoretical CSF algorithm in a stationary environment. It was shown that finite data

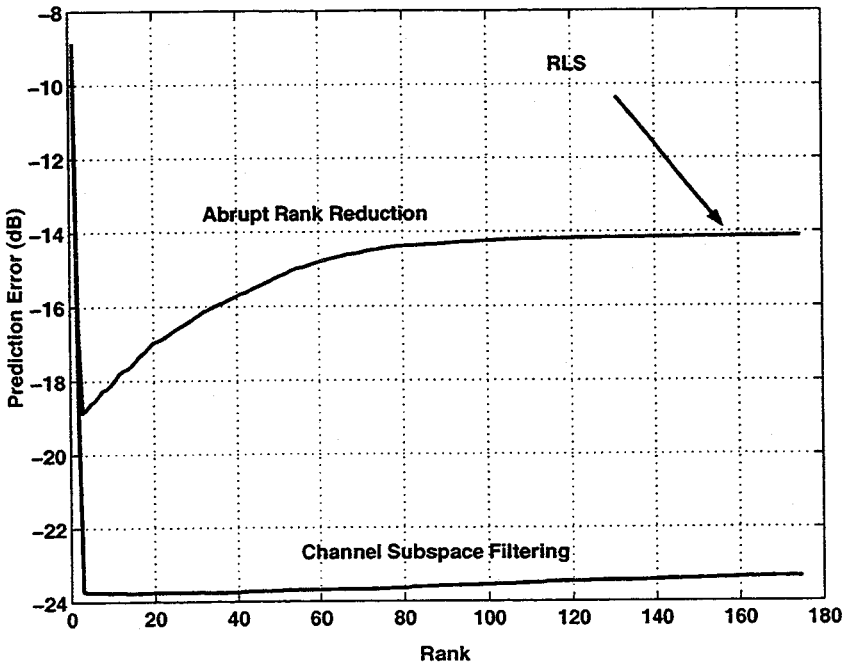

Figure 4-10: Performance vs rank for NOA data

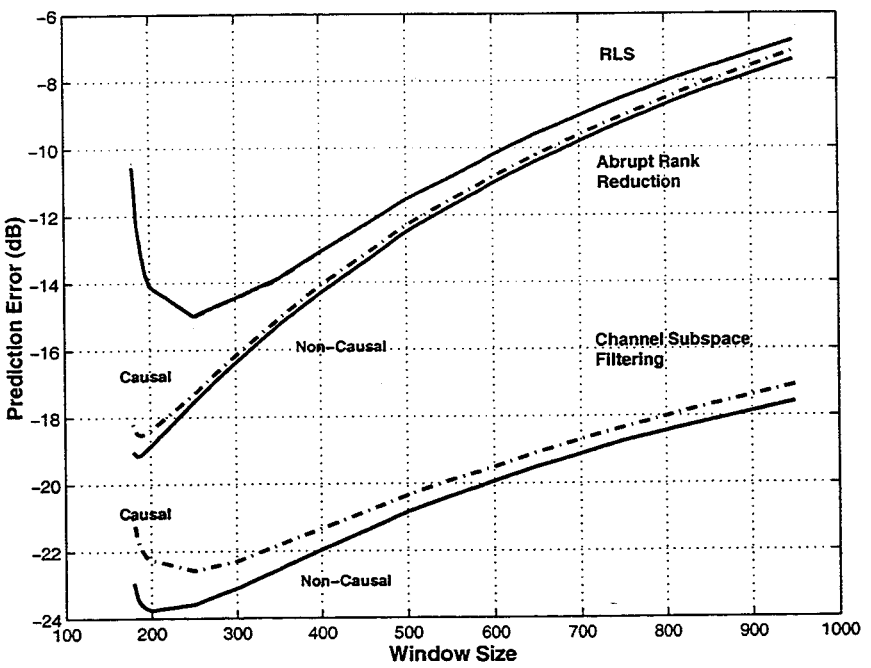

Figure 4-11: Performance vs Window Size for NOA data

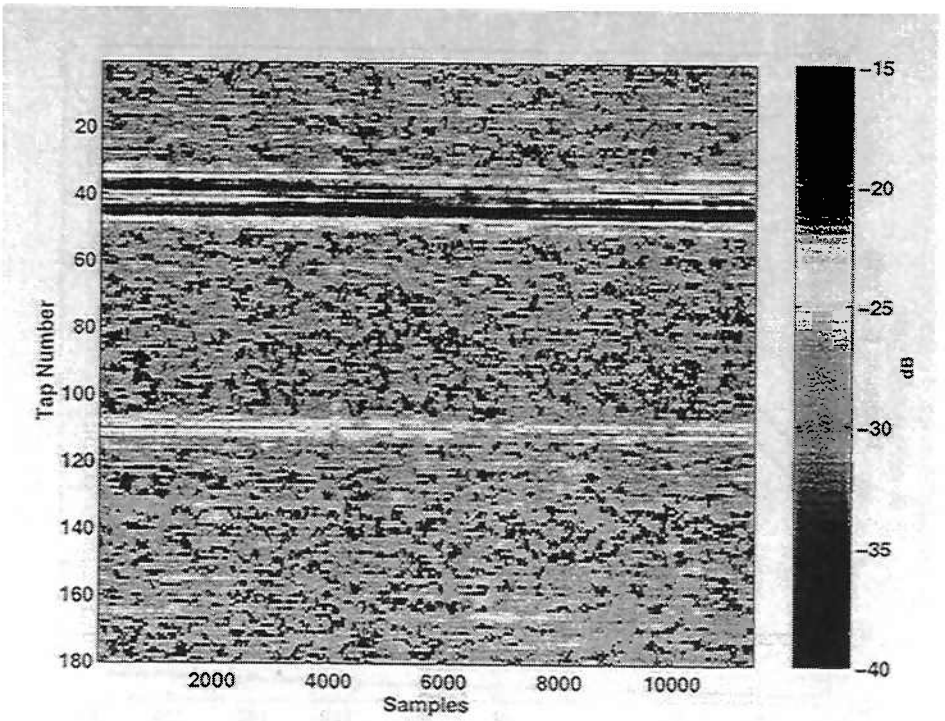

Figure 4-12: Channel estimate time series for AOSN data

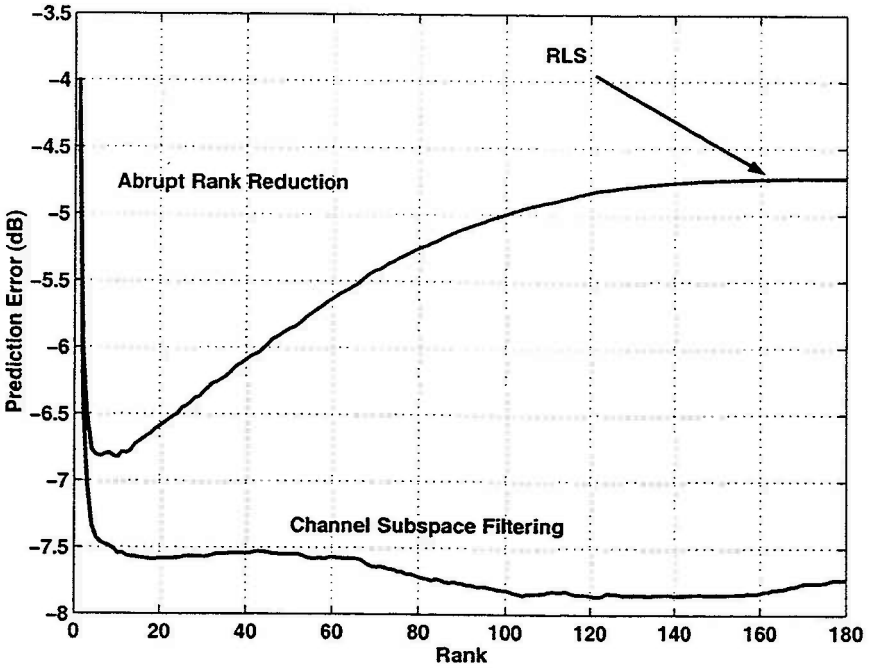

Figure 4-13: Performance vs rank for AOSN data

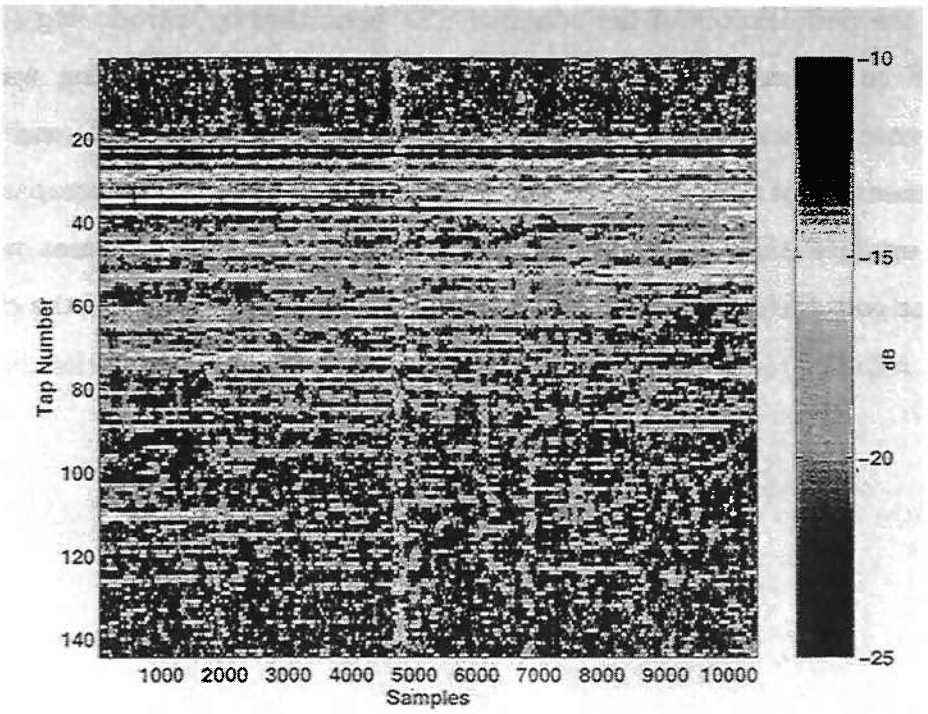

Figure 4-14: Channel estimate time series for SZATE data

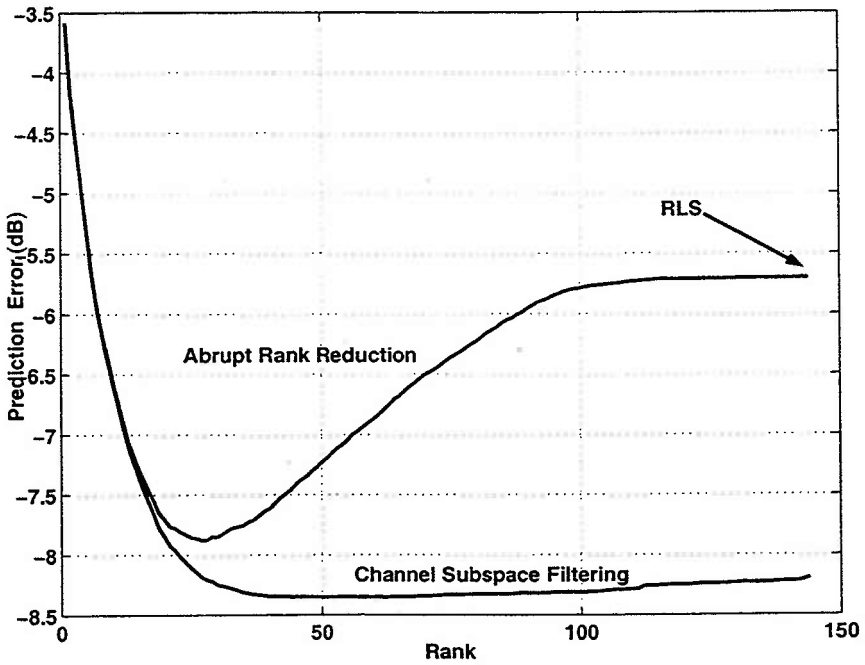

Figure 4-15: Performance vs rank for SZATE data

affected the performance of the adaptive CSF algorithm by introducing eigenvector and CSF coefficient estimation errors. The adaptive CSF algorithm was used to process experimental data for three different, representative channel conditions. An improvement in performance was achieved in all three cases. A comparison with the sub-optimal ARR filter showed that the performance improvement was due to exploiting *both* the reduced dimensionality and subspace structure of the channel.

Chapter 5

Channel estimate based adaptive equalization

The theory and the practice of the channel subspace filtering (CSF) approach was introduced in the previous chapters. It was shown that channel subspace filtering of the least squares channel estimate resulted in a lower channel estimation error. This was corroborated using both simulated data and a range of experimental data. While, intuitively, improved system identification should result in improved equalizer performance, previous chapters have not made this relationship explicit. The following sections introduce the channel estimate based decision feedback equalizer (DFE), assess the impact of imperfect channel estimation on its performance and finally, explicitly, demonstrate the improvement in performance due to the use of a CSF post-filtered RLS channel estimate.

5.1 Channel and DFE system model

Consider the following model, presented earlier in (2.2), for the channel output:

$$y(n) = \underline{h}^H(n)\underline{x}(n) + w(n) \tag{5.1}$$

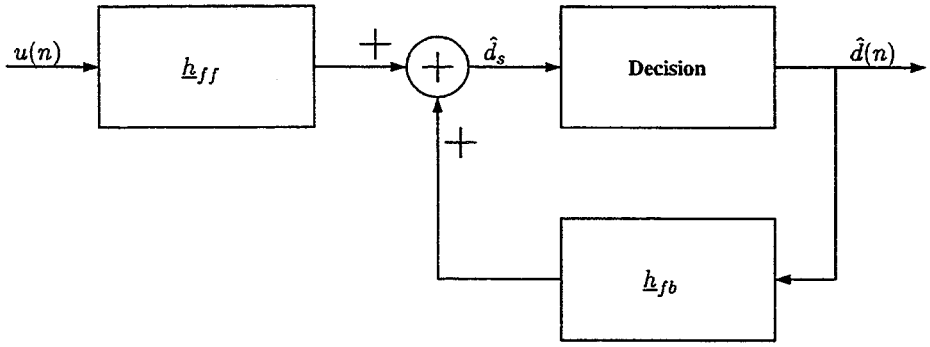

Figure 5-1: Channel estimate based DFE structure

The transmitted data vector is given by $\underline{x}(n) = [d(n), \ldots d(n-N+1)]^T$, where N is the number of taps used to represent the channel impulse response $\underline{h}(n)$ and $d(n)$ is the symbol transmitted at time n. Let N_a and N_c be the number of anti-causal and causal taps in the same representation of the channel impulse response. Causality or anti-causality in this context is with reference to the tap with the most energy. Accordingly the post-cursor of the pre-cursor taps are referred to as the causal or the anti-causal taps. As shown in figure 5-1, the DFE consists of a feedforward section $\underline{h}_{ff}$, a feedback section $\underline{h}_{fb}$ and a detector. The parameters of the DFE define the number of taps needed in the feedforward and the feedback section. Let L_{fb} be the number of taps in the feedback section and L_a and L_c be the number of anti-causal and causal taps in the feedforward section. Figure 5-2 shows how a channel estimate based DFE may be configured based on the variables that have just been defined. The number of filter taps chosen to represent the feedforward and the feedback sections affects the performance of the equalizer. The tradeoff between the number of taps thus chosen and the equalizer performance will be discussed more explicitly in subsequent sections.

Let $\mathbf{G}(n)$ be a matrix with the i^{th} row equal to $[\underline{0}_{1\times(i-1)}, \underline{h}_f^H(n-Lc+i-1), \underline{0}_{1\times(L_a+Lc-i+1)}]$. The symbol data vector $\underline{d}(n)$ is given by:

$$\underline{d}(n) = [d(n-L_c-N_c), \ldots, d(n+L_a+N_a)]^T \tag{5.2}$$

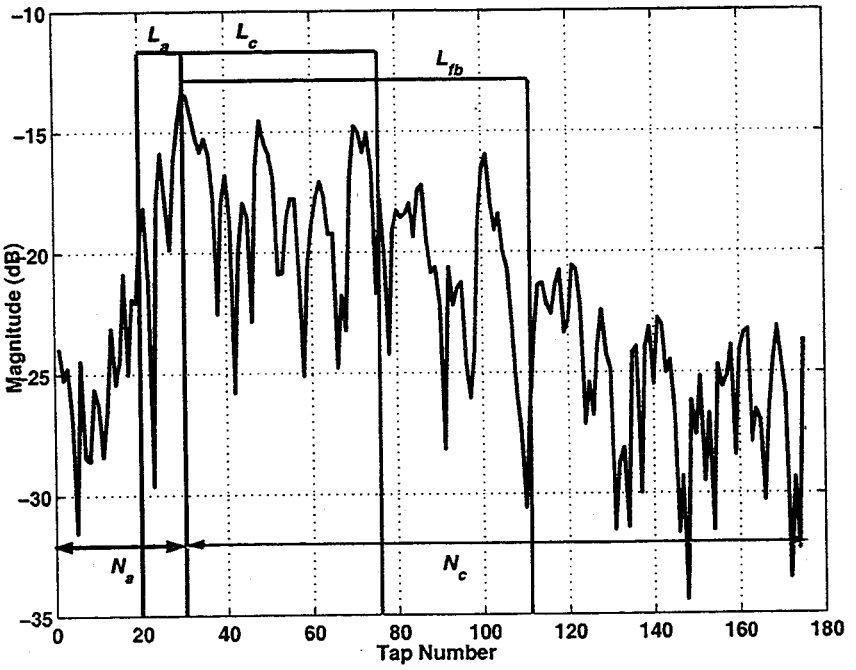

Figure 5-2: Configuring a channel estimate based DFE

The feedback data vector $\underline{\hat{d}}_{fb}(n)$ is given by:

$$\underline{\hat{d}}_{fb}(n) = [\hat{d}_{fb}(n - L_{fb}), \ldots, \hat{d}_{fb}(n-1)]^T \tag{5.3}$$

while the corresponding received data vector and the additive observation noise vector are given by:

$$\underline{u}(n) = [y(n - L_c), \ldots, y(n + L_a)]^T \tag{5.4}$$

$$\underline{w}(n) = [w(n - L_c), \ldots, w(n + L_a)]^T \tag{5.5}$$

respectively. Using the channel model in (2.1), an expression can be formed relating the above vectors such that:

$$\underline{u}(n) = \mathbf{G}(n)\underline{d}(n) + \underline{w}(n) \tag{5.6}$$

As shown in figure 5-1, given the above vectors and based on the choice for the feedforward filter $\underline{h}_{ff}(n)$ and the feedback filter $\underline{h}_{fb}(n)$, the input to the decision

device $\hat{d}_s(n)$ is given by:

$$\hat{d}_s(n) = [\underline{h}_{ff}^H(n)\underline{h}_{fb}^H(n)] \begin{bmatrix} \underline{u}(n) \\ \underline{\hat{d}}_{fb}(n) \end{bmatrix} \tag{5.7}$$

The equalizer tap-weights are chosen to minimize the mean-squared error $E[\| d(n) - \hat{d}_s(n) \|^2]$ at the input to the decision device, assuming that the ISI due to previously detected symbols has been canceled by the feedback section, and subject to the fact that only the knowledge of the channel estimates and channel statistics is available. In practice, incorrect symbol decisions will affect the performance of both the equalizer and the channel estimator in the decision-directed mode of operation. The induced symbol error propagation then has to be limited by periodically inserting the training sequences into the data stream. Assuming that reliable operation has been achieved in this way, the effect of symbol errors is neglected in determining the optimal parameter values for the DFE equalizer tap weights in the following section.

5.2 DFE parameter optimization

The output of the feedforward section is given as:

$$y_{ff}(n) = \underline{h}_{ff}^H(n)\underline{u}(n) \tag{5.8}$$

and the output of the feedback section is given by:

$$y_{fb}(n) = \underline{h}_{fb}^H(n)\underline{\hat{d}}_{fb}(n) \tag{5.9}$$

where $\underline{\hat{d}}_{fb}(n)$ given by (5.3) represents the decision made on the past symbols. The input to the decision device is then:

$$\hat{d}(n) = \underline{h}_{ff}^H(n)\underline{u}(n) + \underline{h}_{fb}^H(n)\underline{\hat{d}}_{fb}(n) \tag{5.10}$$

Rewriting (5.10) in terms of (5.6) results in:

$$\hat{d}(n) = \underline{h}_{ff}^{H}(n)\Big(\mathbf{G}(n)\underline{d}(n) + \underline{w}(n)\Big) + \underline{h}_{fb}^{H}(n)\hat{\underline{d}}_{fb}(n) \tag{5.11}$$

If the channel matrix $\mathbf{G}(n)$ is expressed in terms of its column vectors as:

$$\mathbf{G}(n) = [\underline{f}(n - L_c - N_c), \ldots, \underline{f}(n + L_a + N_a)] \tag{5.12}$$

then, (5.11) can be rewritten explicitly in terms of the causal and anti-causal decisions as:

$$\begin{aligned}\hat{d}(n) = \underline{h}_{ff}^{H}(n)\Big(&\underbrace{\sum_{m=1}^{L_c+N_c} \underline{f}(n-m)d(n-m)}_{Causal} \\ &+ \underline{f}(n)d(n) + \underbrace{\sum_{m=-1}^{-L_a-N_a} \underline{f}(n-m)d(n-m)}_{Anticausal} + \underline{w}(n)\Big) + \underline{h}_{fb}^{H}(n)\hat{\underline{d}}_{fb}(n)\end{aligned} \tag{5.13}$$

When the decisions are correct with high probability, it is reasonable to assume that the past symbols are in fact correct. Thus the substitution $\hat{\underline{d}}_{fb}(n) = \underline{d}_{fb}(n)$ can be made in (5.13). Furthermore, in an ideal DFE, if the true channel impulse response were known perfectly, the the feedback filter could ideally completely cancel the ISI due to past symbols. The ideal feedback filter would thus be given by:

$$\underline{h}_{fb}^{H}(n) = -\underline{h}_{ff}^{H}(n)\Big(\sum_{m=1}^{L_{fb}} \underline{f}(n-m)\Big) \tag{5.14}$$

In realistic situations however, only the channel estimates are available. Thus the feedback taps of the DFE are chosen as:

$$\underline{h}_{fb}^{H}(n) = -\underline{h}_{ff}^{H}(n)\Big(\sum_{m=1}^{L_{fb}} \hat{\underline{f}}(n-m)\Big) \tag{5.15}$$

where $\underline{\hat{f}}(n-m)$ is the corresponding column of the estimated channel matrix $\hat{\mathbf{G}}(n)$ constructed using the channel estimates $\underline{\hat{h}}(n)$ as shown in (5.12). Given this choice for the feedback filter coefficients, the input to the decision device can once again be expressed as:

$$\begin{aligned}\hat{d}(n) = \underline{h}_{ff}^{H}(n)\Big(&\sum_{m=L_{fb}+1}^{L_c+N_c} \underline{f}(n-m)d(n-m) + \underline{f}(n)d(n) \\ &+ \sum_{m=-1}^{-L_a-N_a} \underline{f}d(n-m) + \sum_{m=1}^{L_{fb}} \underline{\delta f}d(n-m) + \underline{w}(n)\Big)\end{aligned} \tag{5.16}$$

where $\underline{\delta f}(n-m) = \underline{f}(n) - \underline{\hat{f}}(n)$ is the channel estimation error. The individual terms in the above expression can be grouped as:

$$\hat{d}(n) = \underline{h}_{ff}^{H}(n)\Big(\underline{f}(n)d(n) + \tilde{\mathbf{G}}(n)\underline{\tilde{d}}(n) + \underline{\epsilon}_f(n) + \underline{w}(n)\Big) \tag{5.17}$$

$$\tilde{\mathbf{G}}(n) = \Big(\sum_{m=L_{fb}+1}^{L_c+N_c} \underline{f}(n-m)d(n-m) + \sum_{m=-1}^{-L_a-N_a} \underline{f}d(n-m)\Big) \tag{5.18}$$

$$\underline{\epsilon}_f(n) = \sum_{m=1}^{L_{fb}} \underline{\delta f}(n-m)d(n-m) \tag{5.19}$$

The term $\underline{\epsilon}_f(n)$ results from channel estimation errors in the feedback taps of the equalizer. The term $\tilde{\mathbf{G}}(n)\underline{\tilde{d}}(n)$ represents the noise due to the residual ISI as a result of past and future symbols. The optimization of the feedforward filter through minimization of the mean-squared error, $MSE = E[\| d(n) - \hat{d}(n) \|^2]$, is made difficult by the fact that the noise terms are data-dependent. In order to avoid this dependence on the unknown symbols, the covariances of the data-dependent terms are substituted for their expected values. Additionally, since it is assumed that the symbol detection probability is fairly high, the equalizer may be assumed to be operating in a regime of high SNR, so that for all practical purposes, the effect of $\underline{\epsilon}(n)$ in determining the optimal feedforward equalizer coefficients is ignored. The impact of feedback errors on the feedback equalizer coefficients is an active area of research that shall not be considered in this thesis. For the case of independent, unit-variance data symbols,

the MMSE solution for the feedforward equalizer coefficients is obtained as:

$$\underline{h}_{ff}(n) = \mathbf{B}^{-1}(n)\underline{\hat{f}}(n) \tag{5.20}$$

where:

$$\mathbf{B}(n) = \left(\tilde{\mathbf{G}}(n)\tilde{\mathbf{G}}^{H}(n) + \sigma^2\mathbf{I}\right) \tag{5.21}$$

5.3 Impact of imperfect channel estimation on DFE performance

For a channel estimate based equalizer, as described above, it can be seen, from (5.15), that the optimal feedback filter can be expressed as a function of the feedforward filter. The feedforward filter can be computed using (5.20). Based on these expressions for the feedforward and the feedback filter, the resulting error criterion $\sigma_d^2 = E[\| d(n) - \hat{d}(n) \|^2]$ can be expressed as:

$$\sigma_d^2 = \sigma_o^2 + \sigma_m^2 \tag{5.22}$$

where σ_o^2 is the optimal equalizer error and σ_m^2 is the misadjustment error. The optimal equalizer error σ_o^2 is given by:

$$\sigma_o^2 = E[d^2(n)] - \underline{f}^{H}(n)\mathbf{B}^{-1}(n)\underline{f}(n) \tag{5.23}$$

The optimal equalizer error depends on only the channel realization and is independent of the channel estimation error. The channel misadjustment can be computed using matrix perturbation theory and keeping terms up to the second-order [26] so that:

$$\sigma_m^2 = \underline{h}_{ff}^{H}(n)E[\delta\mathbf{G}_{fb}\delta\mathbf{G}_{fb}^{H}]\underline{h}_{ff}(n) + E[(\delta\underline{f}(n) - \Delta(n)\underline{h}_{ff}(n))^{H}\mathbf{B}(n)(\delta\underline{f}(n) - \Delta(n)\underline{h}_{ff}(n))] \tag{5.24}$$

where:

$$\Delta(n) = \Delta\tilde{\mathbf{G}}(n)\tilde{\mathbf{G}}^H(n)$$

and

$$\Delta\tilde{\mathbf{G}}(n) = \hat{\tilde{\mathbf{G}}}(n) - \tilde{\mathbf{G}}(n) \tag{5.25}$$

The term $\Delta\tilde{\mathbf{G}}(n)$ represents the error in constructing the $\tilde{\mathbf{G}}(n)$ when imperfect channel estimates are used instead of the true channel impulse response. For a channel estimate based equalizer, σ_m^2 has terms that depend on the projection of the channel estimate error correlation matrix $\mathbf{R}_{\epsilon\epsilon}$ on $\mathbf{B}$. (5.23) and (5.24) implicitly express the tradeoff between equalizer performance, in terms of the error metric σ_d^2, and the number of taps used to represent the feedback and the feedforward sections of the channel estimate based DFE. The optimal equalizer error σ_o^2 from (5.23) does not depend on the channel estimation error. As the number of feedback taps in the equalizer filter increases, the eigenvalues of $\mathbf{B}(n)$ increase resulting in a decrease in σ_o^2. Similarly, increasing the number of feedforward taps, L_a and L_c increases the eigenvalues of $\mathbf{B}(n)$ thereby reducing the optimal equalizer error σ_o^2. This seems to suggest that increasing the number of taps used to represent the feedforward filter should result in improved equalizer performance as indicated. This would certainly be true if perfect channel estimates were available.

Since estimates of the channel impulse response are used to compute the optimal feedback filter, the misadjustment error σ_m^2 affects the performance of the equalizer as well. For a channel estimate based equalizer, the misadjustment error σ_m^2, as noted earlier, has terms that depend on the projection of the channel estimation error $\mathbf{R}_{\epsilon\epsilon}$ on $\mathbf{B}$ and on other functions of the channel impulse response vector $\underline{f}(n)$. As the number of taps in the equalizer increases, the misadjustment error increases as well. As a result the optimal equalizer configuration and hence the choice for the number of feedforward and feedback taps involves balancing the optimal equalizer error σ_o^2

and the misadjustment error σ_m^2. It is certainly apparent that channel subspace post-filtering of the RLS estimates results in decreased misadjustment error and hence, conceivably, improved equalizer performance. Choosing the number of taps for the feedforward and feedback sections either adaptively or statically is an area of active research and shall not be addressed in this thesis. In discussing, quantitatively or qualitatively, any improvements in equalizer performance due to the use of channel subspace post-filtered channel estimates while determining the channel estimate based DFE coefficients, the DFE configuration used is implicitly assumed as being optimal.

5.4 Performance on simulated data

A 35 tap channel with a rank 5 subspace profile having channel subspace energies of $[0.6056, 0.1831, 0.1217, 0.0608, 0.0303]$, the parameter $\alpha = 0.9995$ and a range of SNR's from 10 dB to 20 dB were used to simulate the system in (2.1). The high SNR's were used so that the bit error probabilities were relatively low and the impact of feedback errors on the equalizer coefficient selection would be minimal. The taps were modeled as uncorrelated so that $\mathbf{U} = \mathcal{P}(\mathbf{I})$ with the non-zero tap locations permuted randomly, and the system was evolved according to (2.1) and (2.2). The channel subspace post-filter was computed apriori as $\mathbf{F} = \mathbf{U}\Sigma_{ff}\mathbf{U}^H$ where the channel subspace filtering coefficients, the diagonal elements of Σ_{ff}, were calculated as in (2.70) based on knowledge of the channel model parameters. Channel estimates were generated using the EW-RLS algorithm and subsequently post-processed using the channel subspace filter $\mathbf{F}$ computed apriori. The post-filtered channel estimate was used to determine the feedback and the feedforward filter coefficient values using (5.15) and (5.20). The performance of the CSF post-filtered channel estimate based DFE was compared to that of the unconstrained RLS channel estimate based DFE. Figure 5-3 demonstrates the improvement in equalizer performance due to CSF of the RLS channel estimates.

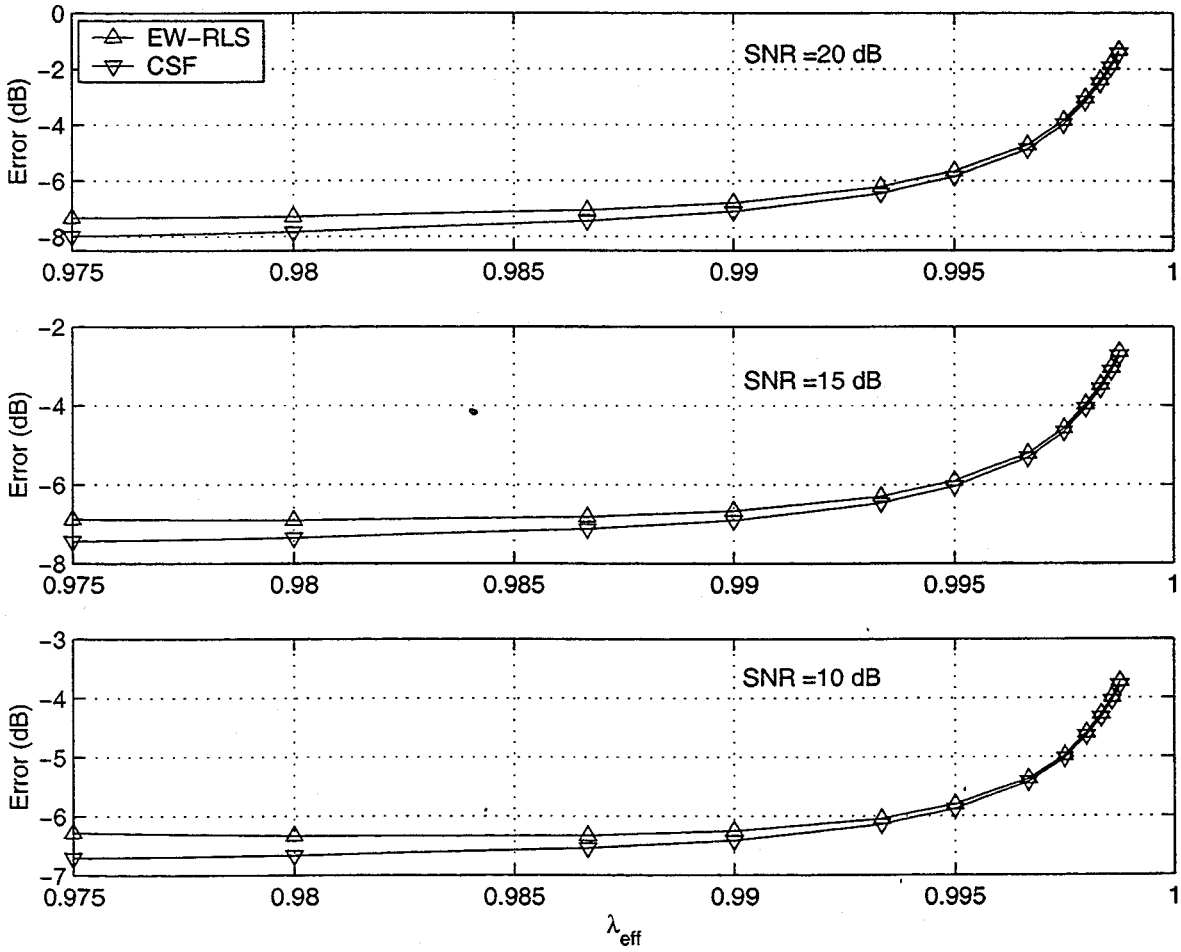

Figure 5-3: Equalizer performance for simulated data

5.5 Performance on experimental data

The channel estimate based DFE algorithm presented above was used to process experimental data collected in actual acoustic communication trials. The following sections present the results of processing this data over a range of operating channel conditions. For the sake of simplicity but more importantly consistency in presentation, the performance of the SW-RLS algorithm has again been used as a benchmark in evaluating any performance improvements. For the experimental results presented, the channel estimates are obtained using an SW-RLS algorithm. The channel estimate time series is used to compute the channel estimate correlation matrix as in (4.4) using either causal or non-causal channel estimates. The eigenvalue decomposition of the estimated channel estimate correlation matrix is then computed to yield the eigenvectors $\hat{\mathbf{U}}$. The adaptive CSF algorithm is used to compute the channel subspace filtering coefficients using the rank r that result in the least channel estimation error from the previous chapter . The post-processed channel estimate $\underline{h}_p(n)$ is used to compute the coefficients of the feedback and the feedforward filter using (5.15) and (5.20).

5.5.1 NOA data

Figure 5-4 compares the equalizer performance of a channel estimate based DFE for post-filtered channel estimates and conventional RLS channel estimates. Figure 5-5 compares the probability of error for the same. Figure 5-7 shows the temporal improvement in prediction error due to CSF while figure 5-6 shows the temporal improvement in equalizer performance as a result of improved channel estimation. As can be seen, the performance of the CSF algorithm is superior to the conventional RLS algorithm for both the causal and the non-causal variants of the algorithm. Henceforth, the non-causal variant of the AD-CSF algorithm shall be used in discussing the experimental data.

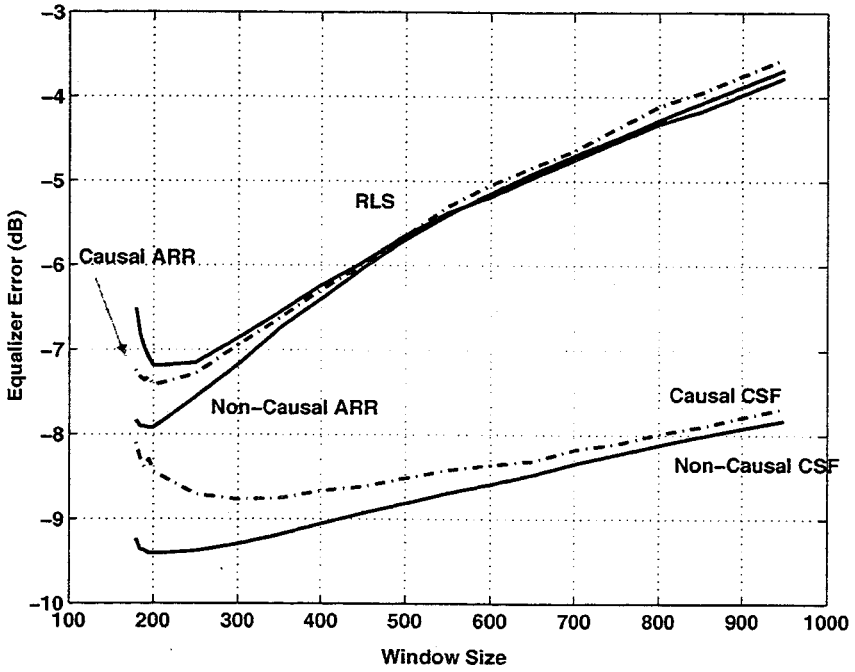

Figure 5-4: Equalizer performance vs window size for NOA data

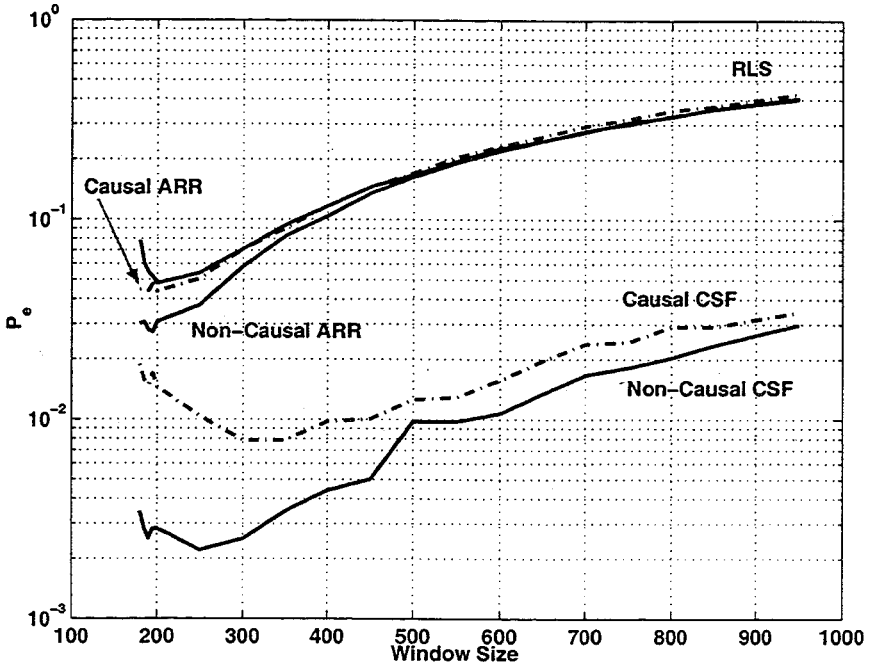

Figure 5-5: Probability of error vs window size for NOA data

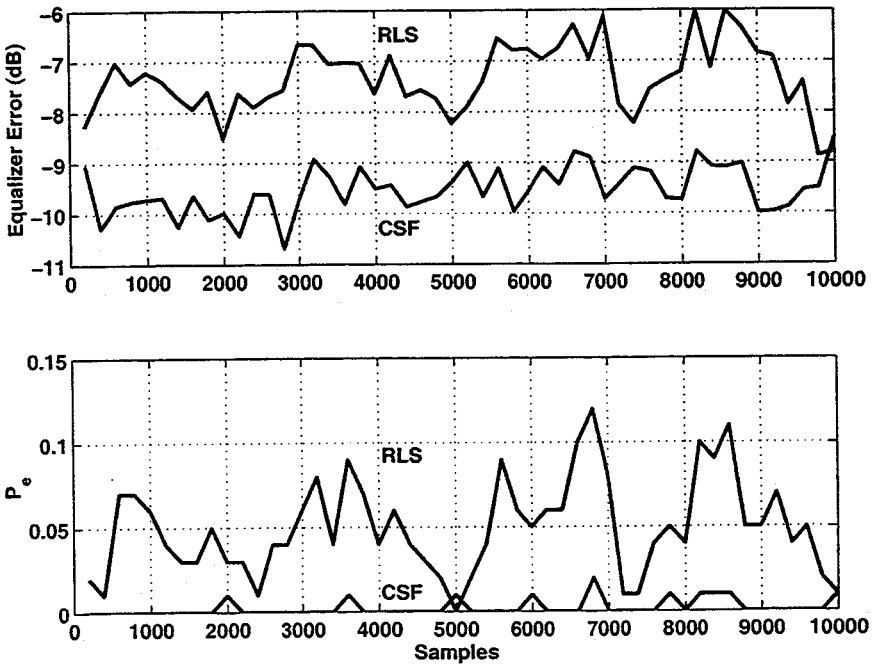

Figure 5-6: Equalizer performance for NOA data

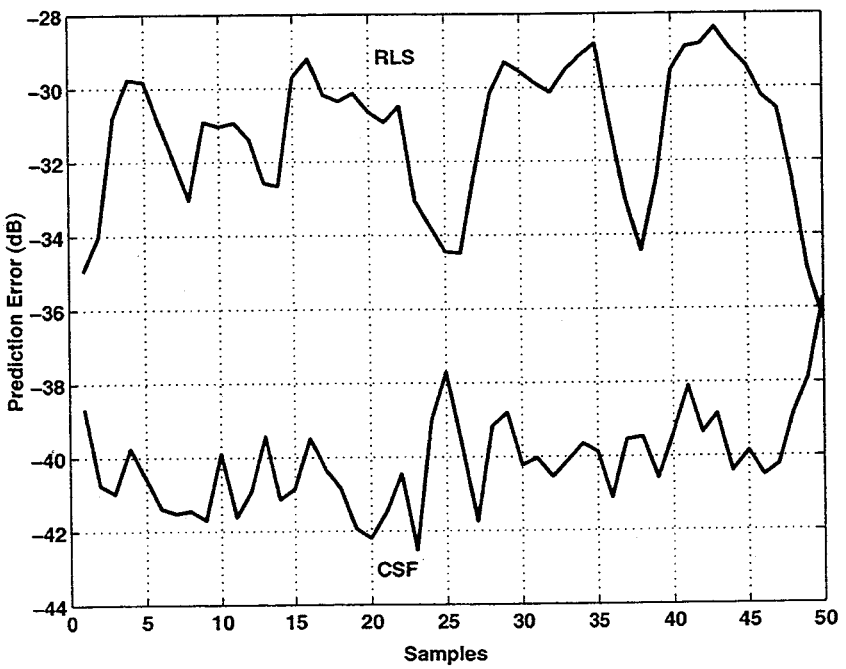

Figure 5-7: Channel estimator performance for NOA data

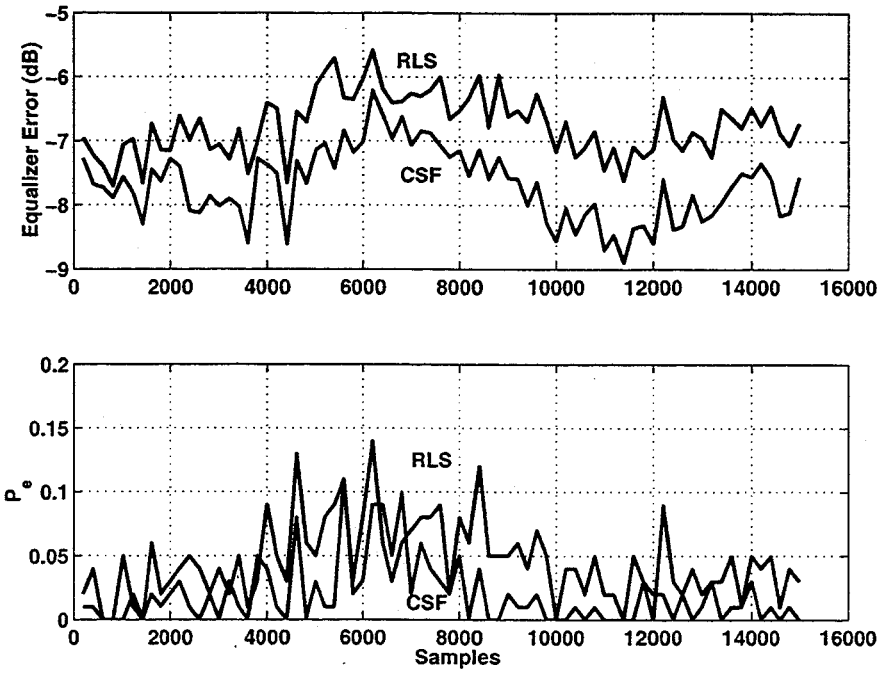

Figure 5-8: Equalizer performance for AOSN data

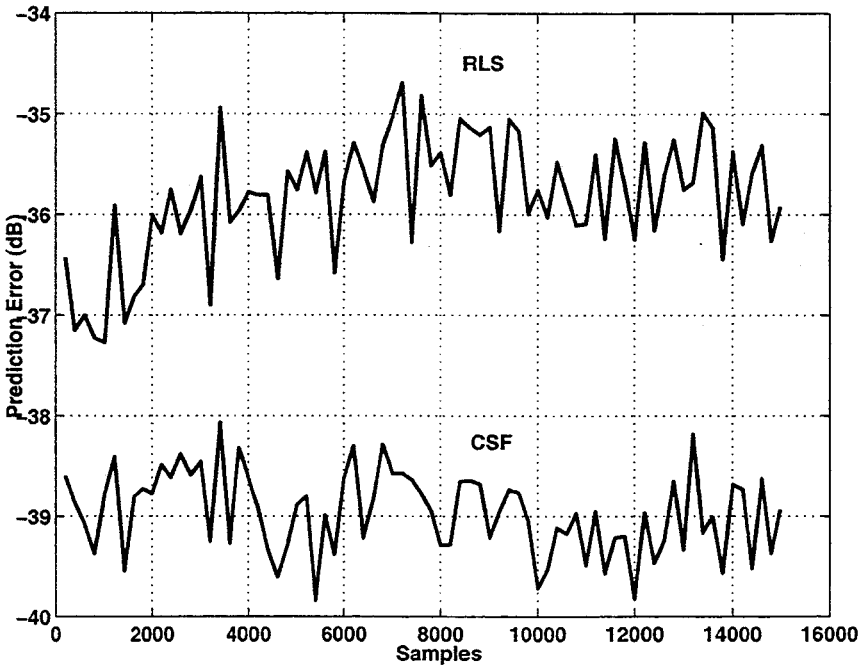

Figure 5-9: Channel estimator performance for AOSN data

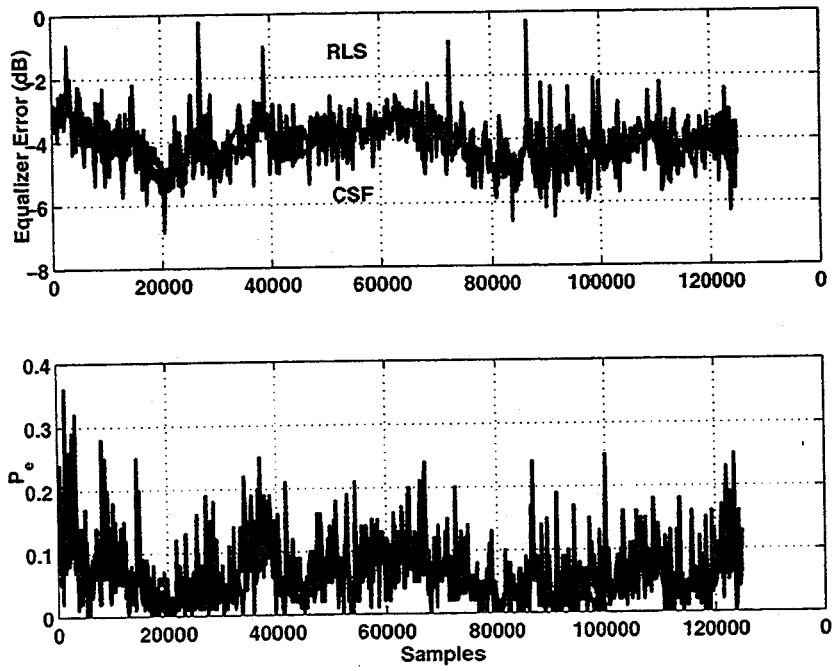

Figure 5-10: Equalizer performance for SZATE data

5.5.2 AOSN data

Figure 5-9 shows the temporal improvement in prediction error due to CSF while figure 5-8 shows the temporal improvement in equalizer performance as a result of improved channel estimation. As can be seen, the performance of the CSF algorithm is superior to that of the conventional RLS algorithm.

5.5.3 SZATE data

Figure 5-11 shows the temporal improvement in prediction error due to CSF while figure 5-10 shows the temporal improvement in equalizer performance as a result of improved channel estimation. As can be seen, the performance of the CSF algorithm is superior to that of the conventional RLS algorithm.

5.6 Summary

In this chapter, a channel estimate based DFE structure was introduced. It was analytically shown that improved channel estimation led to improved equalizer perfor-

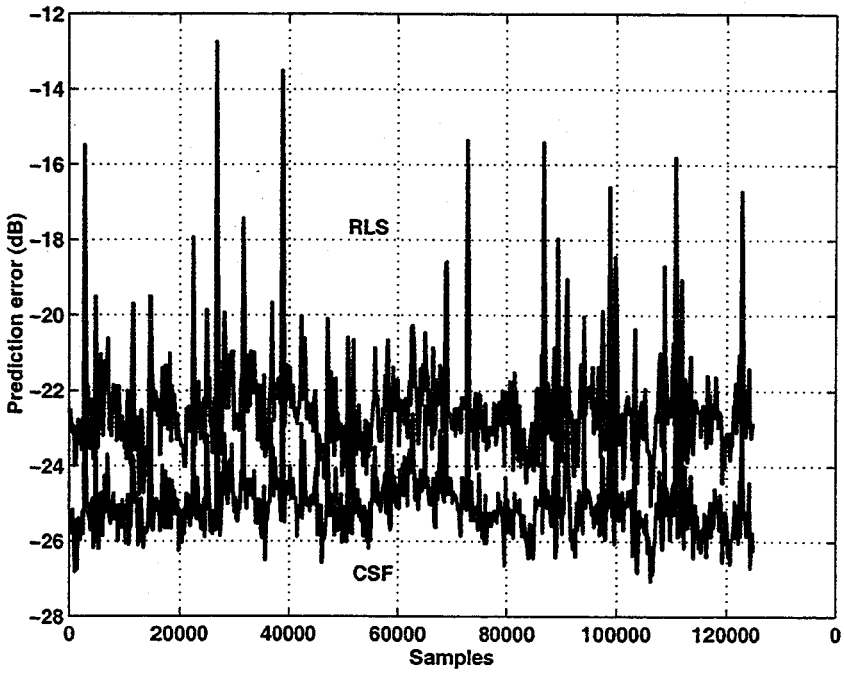

Figure 5-11: Channel estimator performance for SZATE data

mance. This improvement was demonstrated using both simulated and experimental data. Hence, it was established that improved equalizer performance due to CSF resulted in a corresponding improvement in equalizer performance.

Chapter 6

Conclusion

This chapter summarizes the contributions of this thesis and suggests possible directions for future work.

6.1 Thesis overview

In Chapter 1, the low-rank nature of the channel was alluded to and the inability of traditional least-squares tracking algorithms to exploit this characteristic was mentioned. Chapter 2 introduced least squares metrics for tracking performance and developed a methodology for analyzing the tracking performance of common recursive least squares (RLS) algorithms. The channel subspace filtering approach was introduced. For low rank channels, it was shown that tracking performance of a post-filtered unconstrained RLS algorithm was similar to that of a post-filtered uncorrelated parameter reduced rank RLS algorithm. This suggested that efforts made to precisely identify the reduced rank eigenvectors of a system and to formulate an appropriate reduced rank RLS algorithm do not result in significantly improved performance when the CSF approach is used. The analytical predictions made in Chapter 2 were verified using simulations in Chapter 3 where it was indeed demonstrated that CSF of low-rank channels could indeed result in improved tracking performance. The effect of finite data on eigenvector estimation errors and correspondingly on the tracking performance of the post-filtered channel estimate was derived analytically

and demonstrated using simulated data. Chapter 4 introduced an adaptive CSF algorithm for use in an experimental scenario. The equivalence between the theoretical CSF algorithm and the adaptive CSF algorithm was explicitly shown and verified using simulations. Analytically, it was shown that eigenvector estimation errors due to finite data affected the computation of the channel subspace filtering coefficients and hence the performance of the post-filtered channel estimate. These analytical results were corroborated, once again, using simulations.

6.2 Future work

The adaptive CSF algorithm presented in this thesis has been shown to significantly outperform the conventional RLS algorithms. This would justify its implementation in an actual underwater acoustic communication system. However, the causal AD-CSF algorithm is computationally very inefficient which is a major disadvantage for the applications that require UWA communication capability. Future work should concentrate on reducing the numerical complexity of the AD-CSF algorithm so that it may be implemented in an UWA communication system. Furthermore, if the numerical complexity of the algorithm were reduced to levels comparable to an RLS algorithm, then it would be possible to implement a multichannel CSF post-filter. This could potentially lead to dramatically enhanced performance of the UWA communication systems.

It would also be interesting to consider a modified DFE equalizer structure that exploits the low rank of the channel in updating its parameters. The channel estimate based DFE presented in this thesis indirectly exploits the low rank nature of the channel. A directly constrained reduced rank equalizer structure would be an interesting area of future work. Another area of future work would involve linking the physical characteristics of the acoustic channel to the motivation behind the CSF approach.

Appendix A

The complex Wishart distribution

The Wishart distribution plays an important role in statistical signal processing. In this appendix, a summary [15] of some important properties of the Wishart distribution for complex valued data is presented. In particular, a result that is pivotal to a rigorous analysis of the convergence behavior of the standard RLS algorithm is presented.

A.1 Definition

Consider an $N \times N$ time-averaged sample correlation matrix $\Phi(n)$, defined by:

$$\Phi(n) = \sum_{i=1}^{n} \underline{u}(i)\underline{u}^H(i) \tag{A.1}$$

where

$$\underline{u}(i) = \Big[u_1(i), u_2(i), \ldots, u_N(i)\Big]^T \tag{A.2}$$

It is assumed that $\underline{u}(1), \underline{u}(2), \ldots, \underline{u}(n)(n > N)$ are independent, identically distributed random vectors. If $\{u_1(i), u_2(i), \ldots, u_N(i) \mid i = 1, 2, \ldots, n\}, n \geq N$, is a sample from the N-dimensional Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{R}_{uu})$, and if $\Phi(n)$ is the time-averaged correlation matrix defined in (A.1), then the elements of $\Phi(n)$ have the complex Wishart distribution $\mathcal{W}_N(n, \mathbf{R}_{uu})$, which is characterized by the parameters

N, n, and $\mathbf{R}_{uu}$. The probability density function of Φ is given by:

$$f(\Phi) = \frac{1}{2^{\frac{Nn}{2}}\Gamma_N\left(\frac{1}{2}n\right)(det(\mathbf{R}_{uu}))^{\frac{n}{2}}} etr\left(-\frac{1}{2}\mathbf{R}_{uu}^{-1}\Phi\right)(det(\Phi))^{\frac{n-N-1}{2}} \tag{A.3}$$

where $det(\cdot)$ denotes the determinant of the enclosed matrix, $etr(\cdot)$ denotes the exponential raised to the trace of the enclosed matrix, and $\Gamma_N(a)$ is the multivariate gamma function defined by

$$\Gamma_N(a) = \int_{\mathbf{A}} etr(\mathbf{A})(det(\mathbf{A}))^{a-(N+1)/2}\, d\mathbf{A} \tag{A.4}$$

where $\mathbf{A}$ is a positive definite matrix.

A.2 The chi-square distribution as a special case

For the special case of the univariate distribution, that is, $N = 1$, (A.1) reduces to the scalar form [15]:

$$\varphi(n) = \sum_{i=1}^{n} \mid u(i) \mid^2 \tag{A.5}$$

Correspondingly, the correlation matrix $\mathbf{R}_{uu}$ reduces to the variance σ^2. Let

$$\chi^2(n) = \frac{\varphi(n)}{\sigma^2} \tag{A.6}$$

then, using (A.3) the normalized probability density function of the normalized random variable $\chi^2(n)$ may be defined as:

$$f(\chi^2) = \frac{(\frac{\chi^2}{2})^{n/2-1}e^{-\chi^2/2}}{2^{n/2}\Gamma(\frac{1}{2}n)} \tag{A.7}$$

where $\Gamma(n/2)$ is the scalar gamma function. The variable $\chi^2(n)$, defined above, is said to have a *chi-squared distribution with n degrees of freedom* [15]. Thus the complex Wishart distribution may be viewed as a generalization of the univariate chi-square distribution.

A useful property of a chi-square distribution with n degrees of freedom is the fact that is *reproductive with respect to 1/2n* [38]. That is, the r^{th} moment of $\chi^2(n)$ is given by:

$$E[\chi^{2r}(n)] = \frac{2^r \Gamma(\frac{n}{2}+r)}{\Gamma(\frac{n}{2})} \tag{A.8}$$

Thus, the mean, mean-square, and variance of $\chi^2(n)$ are as follows, respectively:

$$E[\chi^2(n)] = n \tag{A.9}$$

$$E[\chi^4(n)] = n(n+2) \tag{A.10}$$

$$var[\chi^4(n)] = n(n+2) - n^2 = 2n \tag{A.11}$$

Moreover, from (A.8), the mean of the reciprocal of $\chi^2(n)$ is given by:

$$\begin{aligned} E\left[\frac{1}{\chi^2(n)}\right] &= \frac{1}{2}\frac{\Gamma(\frac{n}{2}-1)}{\Gamma(\frac{n}{2})} \\ &= \frac{1}{2}\frac{\Gamma(\frac{n}{2}-1)}{(\frac{n}{2}-1)\Gamma(\frac{n}{2})} = \frac{1}{n-2} \end{aligned} \tag{A.12}$$

where the property $\Gamma(g) = (g-1)\Gamma(g-1)$ enables the above simplification.

A.3 Expectation of the inverse correlation matrix

An additional property of the complex Wishart distribution is that if $\mathbf{\Phi}$ is $\mathcal{W}_N(n, \mathbf{R}_{uu})$ and $\underline{a}$ is any $N \times 1$ fixed non-zero vector in $\mathcal{R}^N$, then $\underline{a}^H \mathbf{R}_{uu}^{-1} \underline{a} / \underline{a}^H \mathbf{\Phi}^{-1} \underline{a}$ is chi-square distributed with $n - N + 1$ degrees of freedom. Let $\chi^2(n-N+1)$ denote this ratio. Then, using the result described in (A.12),

$$\begin{aligned} E[\underline{a}^H \mathbf{\Phi}^{-1}(n) \underline{a}] &= \underline{a}^H \mathbf{R_{uu}}^{-1} \underline{a}\; E\left[\frac{1}{\chi^2(n-N+1)}\right] \\ &= \frac{1}{n-N-1}\; \underline{a}^H \mathbf{R_{uu}}^{-1} \underline{a}, \qquad n > N+1 \end{aligned} \tag{A.13}$$

which in turn implies that

$$E[\Phi^{-1}(n)] = \frac{1}{n-N-1}\mathbf{R}_{uu}^{-1}. \qquad n > N+1 \tag{A.14}$$

Bibliography

[1] K. Abed-Meraim, P. Loubaton, and E. Moulines. A subspace algorithm for certain blind identification problems. *IEEE Trans. Information Theory*, 43(2):499–511, Mar. 1997.

[2] A. Baggeroer. Acoustic telemetry - an overview. *IEEE Journal of Oceanic Engineering*, OE-9:229–235, Oct. 1984.

[3] P. Bello. Characterization of randomly time-variant linear channels. *IEEE Trans. Communication Systems*, CS-11:360–393, Dec. 1963.

[4] L. Berkhovskikh and Y. Lysonov. *Fundamentals of Ocean Acoustics.* Springer, New York, 1982.

[5] J. A. Bucklew, T. Kurtz, and W. A. Sethares. Weak convergence and local stability properties of fixed stepsize recursive algorithms. *IEEE Trans. Information Theory*, 39(3):966–978, May 1993.

[6] Y. F. Cheng and D. M. Etter. Analysis of an adaptive technique for modeling sparse systems. *IEEE Trans. Signal Processing*, 37(2):254–264, Feb. 1989.

[7] R. Coates. *Underwater Acoustic Systems.* Wiley, New York, 1989.

[8] T. Curtin, J. Bellingham, J. Catipovic, and D. Webb. Autonomous oceanographic sampling networks. *Oceanography*, 6(3):86–94, 1993.

[9] Z. Ding. Matrix outer-product decomposition method for blind multiple channel identification. *IEEE Trans. Signal Processing*, 45(12):3053–3061, Dec. 1997.

[10] E. Eleftheriou and D. D. Falconer. Tracking properties and steady state performance of rls adaptive filter algorithms. *IEEE Trans. Acoust. Speech Signal Processing*, ASSP-34:1097–1110, 1986.

[11] I. J. Fevrier, S. B. Gelfand, and M. P. Fitz. Reduced complexity decision feedback equalization for multipath channels with large delay spreads. *IEEE Trans. Communications*, 47(6):927–937, June 1999.

[12] S. M Flatte. Wave propagation through random media: Contributions from ocean acoustics. *Proc. of the IEEE*, 71:1267–1294, Nov. 1983.

[13] S. M. Flatte, J. Colosi, T. Duda, and G. Rovner. *Ocean variability and acoustic propagation*, chapter Impulse-response analysis of ocean acoustic properties, pages 161–172. Kluwer Academic Publishers, 1991.

[14] R. Galvin and R. F. W. Coates. Analysis of the performance of an underwater acoustic communication system and comparison with a stochastic model. In *Proc. OCEANS '94*, pages III.478–III.482, Brest, France, Sep. 1994.

[15] S. Haykin. *Adaptive Filter Theory.* Prentice Hall, Englewood Cliffs, NJ, 1996.

[16] G. S. Howe, P. Tarbit, O. Hinton, B. Sharif, and A. Adams. Sub-seq acoustic remote communications utilizing and adaptive receiving beam former for multipath suppression. In *Proc. OCEANS '94*, pages I.313–I.316, Brest, France, Sep. 1994.

[17] W. C. Jakes, editor. *Microwave Mobile Communications.* IEEE Press, Piscataway, NJ, 1974.

[18] D. B. Kilfoyle and A. B. Baggeroer. The state of the art in underwater acoustic telemetry. *IEEE Journal of Oceanic Engineering*, 25(1):4–27, Jan 2000.

[19] D. Kocic, D. Brady, and M. Stojanovic. Sparse equalization for real-time digital underwater acoustic communications. In *Proc. OCEANS'95*, number 3, pages 133–139. MTS/IEEE, 1995.

[20] M. Kristensson and B. Ottersten. Statistical analysis of a subspace method for blind channel identification. In *Proc. ICASSP*, number 17, pages 2437–2440, Atlanta, GA, May 1996.

[21] H. J. Kushner. *Approximation and weak convergence methods for random processes with applications to stochastic system theory.* MIT Press, Cambridge, MA, 1984.

[22] P. Monsen. Feedback equalization for fading dispersive channels. *IEEE Trans. Information Theory*, Jan. 1971.

[23] E. Moulines and P. Duhamel. Subspace methods for the blind identifications of multichannel fir filters. *IEEE Transactions of Signal Processing*, 43(2):516–525, Feb. 1995.

[24] M. Niedzwecki. First order tracking properties of weighted least squares estimator. *IEEE Trans. Automatic Control*, 33(1):94–96, Jan. 1988.

[25] H. V. Poor and G. W. Wornell, editors. *Wireless communications: Signal processing perspectives.* Prentical Hall, Upper Sadle River, NJ, 1998.

[26] J. C. Preisig. Personal communication.

[27] J. G. Proakis, H. Lev-Ari, J. Lin, and F. Ling. Optimal tracking of time-varying channels: A frequency domain approach for known and new algorithms. *IEEE Trans. Selected Areas in Communications*, 13(1):141–154, Jan. 1995.

[28] S. U. H. Qureshi. Adaptive equalization. In *Proc. of the IEEE*, number 9, pages 1349–1387. IEEE, Sep. 1985.

[29] I. S. Reed, J. D. Mallett, and L. E. Brennan. Rapid convergence rate in adaptive arrays. *IEEE Trans. Aerospace and Electronic Systems*, AES-10(6):853–863, Nov. 1974.

[30] L. L. Scharf and D. W. Tufts. Rank reduction for modeling stationary signals. *IEEE Trans. Acoustic, Speech and Signal Processing*, ASSP-35:350–355, Mar. 1987.

[31] P. Shukla and L. Turner. Channel estimation based adaptive dfe for fading multipath radio channels. *IEE Proceedings*, 138(6):525–543, dec 1991.

[32] M. Stojanovic, J. A. Catipovic, and J. G. Proakis. Adaptive multi-channel combining and equalization for underwater acoustic communications. *Journal of Acoustic Society of America*, 94:1621–1631, 1993.

[33] M. Stojanovic, J. A. Catipovic, and J. G. Proakis. Phase-coherent digital communication for underwater acoustic channels. *IEEE Journal of Oceanic Engineering*, 19:100–111, 1994.

[34] M. Stojanovic, L. Freitag, and M. Johnson. Channel-estimate based adaptive equalization of underwater acoustic signals. In *Proc. Oceans '99*, Seattle, Washington, 1999.

[35] P. Strobach. Low rank adaptive filters. *IEEE Trans. Signal Processing*, 44(12):2932–2947, Dec. 1996.

[36] L. Tong, G. Xu, and T. Kailath. A new approach to blind identification and equalization of multipath channels. In *Proc. 25th Asilomar Conference*, pages 856–860, Pacific Grove, CA, Nov. 1991.

[37] B. Widrow and S. D. Stearns. *Adaptive Signal Processing.* Prentice Hall, Englewood Cliffs, NJ, 1985.

[38] S. S. Wilks. *Mathematical statistics.* Wiley, New York, 1962.