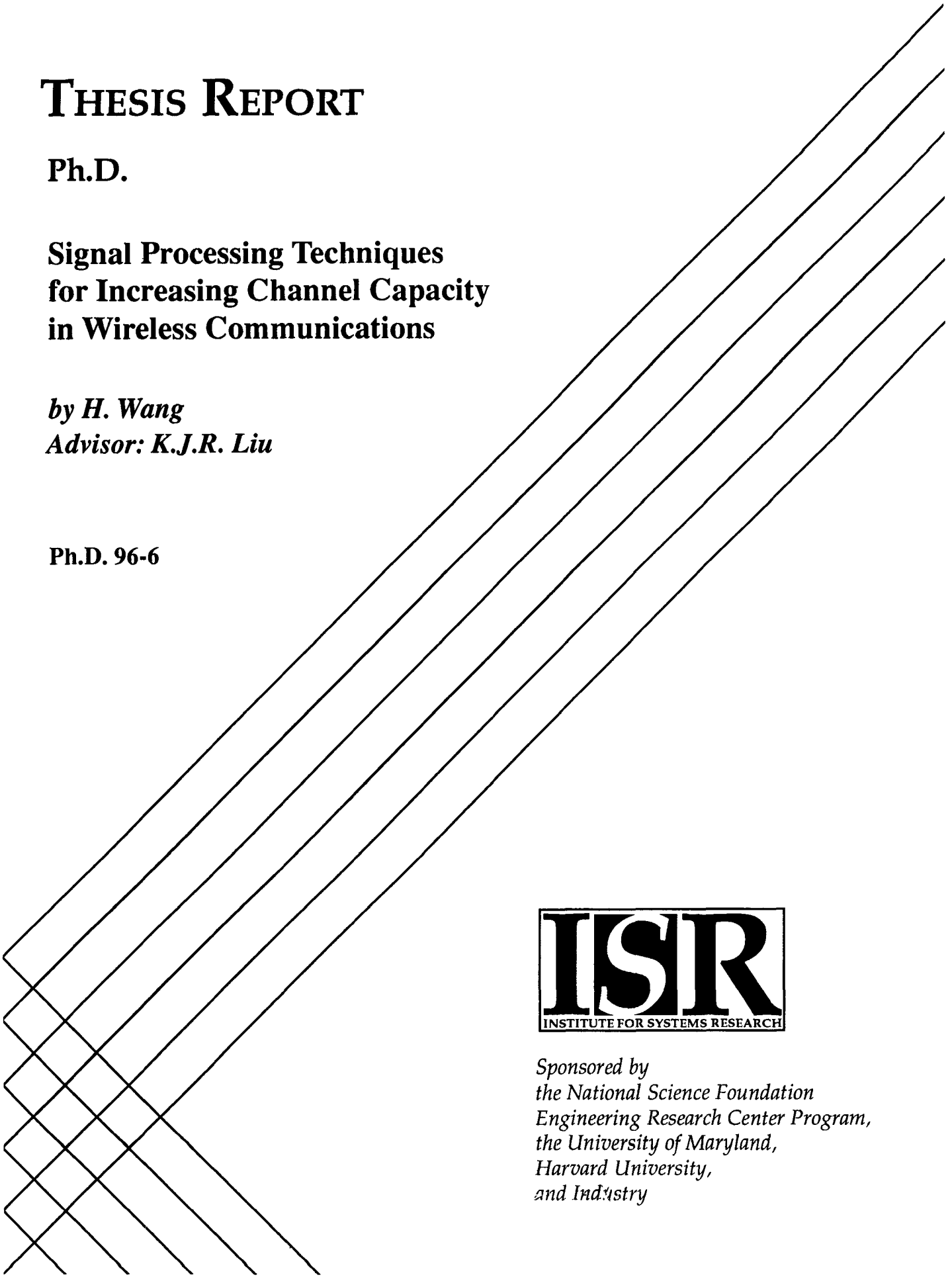
THESIS REPORT

Ph.D.

Signal Processing Techniques for Increasing Channel Capacity in Wireless Communications

by H. Wang
Advisor: K.J.R. Liu

Ph.D. 96-6



Sponsored by
the National Science Foundation
Engineering Research Center Program,
the University of Maryland,
Harvard University,
and Industry

ABSTRACT

Title of Dissertation: Signal Processing Techniques for Increasing Channel Capacity in Wireless Communications

Hongyi Wang, Doctor of Philosophy, 1996

Dissertation directed by: Professor K. J. Ray Liu
Department of Electrical Engineering

As the digital signal processing technology advances, the use of adaptive arrays to combat multipath fading and to reduce interference becomes increasingly valuable as a means of adding capacity to mobile communications. This dissertation addresses the major obstacles encountered in applying the two most applicable adaptive array algorithms to time division multiple access(TDMA) wireless communication systems.

We first investigated the reference signal based adaptive diversity combining algorithm, which conventionally relies on feedback symbols in the absence of reference signals. Our computer simulation revealed that on a fast time varying fading channel, error propagation in the decision directed tracking mode severely degrades the performance. We developed a simultaneous diversity combining and

decoding technique which incorporates QR decomposition-based recursive least-square parallel weights tracking and M-D decoding algorithms. In contrast to the conventional system where only one set of array weights is kept and updated, in our system, we update M sets of candidate weights. Thus we are able to make a more reliable symbol decision based on D symbols without compromising weights tracking speed. The M-D algorithm was first developed for the binary convolutional codes and then extended to Trellis-coded modulation. This technique significantly reduces error propagation. Simulation results showed that about 8 to 10dB improvement in the total interference suppression at low ISR and about 5dB improvement at high ISR can be achieved with a moderate increase in complexity.

In the next part of the dissertation, we proposed and studied the use of the constrained adaptive array algorithm for extracting signals from interferences at separable directions. This algorithm requires direction-of-arrival(DOA) information and does not need reference signals. However, most of the high resolution DOAs estimation methods are only effective for noncoherent signals, while in mobile radio channels, coherent signals are inevitable. We developed a general spatial smoothing(SS) technique and a forward backward spatial smoothing technique for two dimensional arrays to decorrelate coherent signals from arbitrary directions. We found and proved the necessary and sufficient conditions on an array configuration for applying SS. This array must have an orientational invariance structure with an ambiguity free center array, and the number of subarrays must be larger than or equal to the size of the largest group of coherent signals. We also studied the causes of ambiguities and found some ambiguity free array manifolds. We expanded the application of our SS to several high resolution DOA estimation and constrained adaptive beamforming algorithms. All the predicted results were

verified by simulations.

In the last part of the dissertation, we investigated the applications of adaptive array technique in DS/CDMA systems. We applied reference-signal-based simultaneous diversity combining and decoding to reduce fading and suppress interference caused by poor synchronization and power control.

Signal Processing Techniques for Increasing Channel Capacity in Wireless Communications

by

Hongyi Wang

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland at College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
1996

Advisory Committee:

Professor K. J. Ray Liu, Chairman/Advisor
Professor Thomas E. Fuja
Professor Steve Tretter
Professor Leandros Tassiulas
Professor Dianne O'Leary

Dedication

To my daughter, my husband and my parents

Acknowledgements

I am truly grateful to my advisor, Professor K.J.Ray Liu for his continuous inspiration, insightful guidance, and boundless support. What I learned from him – his dedication, enthusiasm and methodology, will be a life long asset.

I acknowledge my gratitude to committee members, Professors Dianne O'Leary, Thomas E. Fuja, Steve Tretter and Leandros Tassiulas for their time and advice on my dissertation.

It was a great pleasure to work with Dr.Jack Winters who helped me understand throughly the diversity combining project during my four months at Bell Labs. I sincerely thank Dr.Lee-Fang Wei for his valuable advice in my research. I benefited from stimulating discussions with Carol Martin and others at the Bell Labs. I also like to thank Henry Anderson of Watkins-Johnson Company who contributed to the research on adaptive arrays.

Special thanks are due to Ut-Va Koc and Andy Wu, who provided me their expertise on computer systems. I also like to thank other members in the DSP group who contributed to my learning and research. I enjoyed the comradery tremendously.

I am deeply indebted to my daughter Linda who brought so much joy and happiness but did not get my fullest attention, and to my husband Yihong for his infinite love, support and companionship in my pursuit for Ph.D. A debt of gratitude is owed to my parents for their sacrifice and most needed help and for all they have given me in my life.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

In recent years, there has been an explosive increase in the demand of radio channels in cellular communication systems. Since the frequency spectrum is limited, the total number of available channels is bounded. To increase the number of users in a given bandwidth becomes the dominant goal of much of today's intense research in wireless communication systems. Currently, digital speech transmission and Time Division Multiple Access (TDMA) [1] have been adopted in most of the second generation cellular systems. Accordingly, efficient speech coding, error-control coding, and bandwidth efficient modulation have been used. On the other hand, Code Division Multiple Access (CDMA) has been proposed for the third generation wireless information systems. It is believed that a substantial additional gain in system capacity can be achieved by exploiting the space dimension no matter which system is used.

The overall objectives of this thesis are in the framework of increasing wireless channel capacity and improving the quality of transmitted speech. The basis of our approaches is to suppress interferences in the space domain using array signal processing techniques.

1.1 Previous Schemes to Increase Capacity in Space Domain

To increase frequency efficiency by exploiting the space dimension, there are three existing approaches based on the co-channel interference reduction factor (CIRF), q_s, which is defined as $q_s = \frac{D_s}{R}$ where R is the cell radius and D_s is the minimum required distance between any two co-channel cells in a cellular system corresponding to the required carrier-to-interference ratio (C/I) received at both the cell site and the mobile unit in a cell.

- **Split the cells**

  Capacity can be increased by reducing R but keeping q_s unchanged, that is, by rescaling the system. A conventional microcell follows this approach. In theory, the cell splitting process may be carried out infinitum. In practice, however, there are some obvious constraints [2]: (1) With cells becoming smaller and smaller, it is increasingly difficult to place base stations at the locations that offer the necessary radio coverage. The problem of establishing a new cell cite at an optimum location is particularly acute in large congested cities where capacity requirements are most pressing. Finding a suitable location for the base station may become a difficult task. (2) The total cost of the system is increased because the number of required base stations is increased. (3) The trunking efficiency is degraded. (4) As the distance between cells reduces, the cochannel interference increases, although the same repeat pattern is kept.

- **Sectorization**

This technique basically divides the cell into a number of sectors, each served by a different set of channels and illuminated by a directional antenna. The sector can therefore be considered as a new cell. The use of directional antennas substantially cuts down the cochannel interference, thus allowing the cocells to be more closely spaced. The problem for the sectorization is that it reduces the covered area [2], and the trunking efficiency of the utilized channels decreases [3]. The hand-offs occur as the vehicle passes across the sectors, same way as across cells.

- **Reducing the required D/R by a multiple-zone microcell approach**

  Generally a new microcell [3] consists of three zones. Every zone site physically shares the same radio equipment installed at the base. To serve a vehicle from a zone site, an 800-MHz cellular signal can be converted up to a microwave or optical signal at the base and then converted down back to the 800-MHz signal at the zone site to serve the vehicle in that zone as if the vehicle were located at the base. In this case not only the cell radius is reduced, but also the CIRF is reduced and there is no degradation in trunking efficiency. However, this method needs additional zone sites and additional infrastructure. It also introduces additional noise and time delay.

In short, wireless communication industry has encountered practical limits on reducing the cell size to increase spectral efficiency.

1.2 Adaptive Arrays in Wireless Communications —- Background

To increase the capacity beyond the limits by cell size reduction, adaptive array has been proposed for increasing the carrier to interference ratio (C/I) at both the base station and the mobile terminal. If this can be achieved, then more RF channels can be added to the base stations in the area without driving the C/I ratio below an acceptable limit.

An adaptive array consists of an array of spatially distributed antennas and a real time adaptive digital signal processor. By properly combining the antenna outputs with adaptive array signal processing algorithms, the array beam pattern can be automatically adjusted to extract the individual signal waveform from received signals corrupted by cochannel interference and noise. Compared to a conventional array with fixed beam, an adaptive array is more flexible and more accurate. The continued decrease in the cost of digital signal processors makes adaptive arrays even more attractive now.

There are many optimum adaptive array combining algorithms. Two of them are most applicable in TDMA wireless communications and have attracted a lot of attention recently. They are briefly introduced in the following and will be discussed in depth in the later chapters.

1.2.1 Optimum Diversity Combining

A significant increase in system capacity can be achieved by the use of reference signal based optimum spatial diversity array combining [5]-[9]. In a spatial diversity array combining system, multiple antennas are spaced at least one half

of a wavelength apart, and the array weights are chosen to minimize the mean square error between the reference signals and array output. Optimum diversity combining is capable of cancelling the interference produced by users who are occupying the same frequency band and time slots. Theoretically, for independent flat-Rayleigh fading wireless systems with N mutually interfering users and $K+N$ antennas, $N-1$ interferers can be nulled out and $K+1$ path diversity improvement can be achieved by each of the N users [9]. The interferers can also be users in other radio systems, or even other types of radiating devices. Thus interference cancellation also allows radio systems to operate in high interference environments. These theoretical results provide a solid basis for assessing the improvement that can be achieved by antenna diversity with optimum combining.

In practice, a mobile radio environment is characterized by (i) path loss, (ii) long-term log-normal shadowing determined by the terrain, (iii) short-term Rayleigh fading caused by local structures and obstacles in the immediate vicinity and (iv) Doppler shift caused by vehicle mobility. The average duration of fades, as well as the rate of level crossings are functions of vehicle speed and wavelength. The power of the interfering signal fluctuates. Hence the adaptive array signal processing algorithm should be able to track the variation in the signals. In IS-136 digital cellular standard, we have only 14 symbols at the begining of each time slot that can be used as reference signals. A numerically stable and fast converging adaptive array signal processing algorithm has to be chosen to get an initial estimate of array weights in the training mode. After training mode, we have to make a symbol by symbol decision, and feedback the decided symbol every time to update the array weights. To track a fast time-varying fading channel, the updating window size has to be small. If we make a decision error, this error weights heavily in the

estimation of the next weights and will result in an erroneous weights estimation and next decision error. This error will further propagate and cause subsequent decision errors.

We conducted simulations to evaluate the performance of an adaptive array under different situations: ideal case, training mode and tracking mode.

- In the ideal case when we perfectly know the channel interfering and fading conditions, our simulation results show that interference that is 50dB stronger than the desired signal can be suppressed at all vehicle speeds with optimum combining of four antenna elements.

- In the training mode, reference signals are available. We observe some loss in tracking at high vehicle speed. But, still, interference about 40dB stronger than the desired signals can be suppressed at vehicle speed of 60miles/hr.

- In the tracking mode, we feed back currently decided symbol to further update the array weights. This results in error propagation. We observed severe degradation in BER performance.

Therefore, the effectiveness of optimum diversity combining on a fast time-varying fading channel depends on the tracking speed of the adaptive algorithms and on the level of a decision error propagation.

1.2.2 Constrained Adaptive Beamforming

The other suitable adaptive array combining scheme is based on high resolution direction finding followed by constrained adaptive beamforming. The array response is constrained so that signals from the direction of interest are passed with specified gain. The array weights are chosen to minimize array output power under

these constraints. By actively tracking mobile units and directionally transmitting information to and receiving information from these units, a significant increase in channel capacity of current wireless communication system is possible [10]- [13]. The key to making this technique effective in multipath fading mobile environment is accurate DOA's estimation.

There are several DOA estimation methods. The classical Fourier-based low resolution approach basically searches for those array response vectors that have the highest correlations with the received array vector. High resolution DOA estimation methods, both multiple signal classification (MUSIC) and estimation of signal parameters via rotational invariance techniques (ESPRIT), are based on the fact that the received signal spans the two disjoint spaces: signal plus noise space and noise space. The space spanned by the array response corresponding to the incoming signals is the same as the signal plus noise space. Angles are identified by searching for those array response vectors that are orthogonal to the noise subspace. These approaches can exceed the conventional Rayleigh resolution limit. So they are called high resolution methods. However, these methods are valid only when there are no coherent signals in the incoming signals, i.e., the correlation matrix of the signal is full rank, whereas in reality, coherent signals are inevitable in wireless communications. When there are coherent signals, a preprocessing SS can be used to decorrelate coherent signals. However, such a scheme was only applicable to uniformly spaced linear arrays. For two dimensional arrays, no computationally efficient method was available in the published literature. Multidimensional subspace fitting algorithms such as deterministic maximum likelihood (DML) [14], multidimensional (MD)-MUSIC [15], and recently proposed weighted subspace fitting (WSF) [16], [17], are effective in both coherent and noncoherent

environment and can be applied to arrays of nonlinear geometry. However, all these algorithms involve some searching procedures used to solve nonlinear equations. They are computationally intensive and are impractical in real-time applications. The direction finding techniques using spatial smoothing with interpolated arrays [13] [18] map the signal received by the array to a virtual array, but these methods all need approximations and have limited applications.

1.3 Motivation

As the demand for radio spectrum grows and DSP technology advances, the ability of using sophisticated adaptive array algorithms to combat multipath fading and to reduce interference becomes increasingly valuable as means of adding capacity to mobile communication systems. This motivates us to solve or alleviate the major problems we identified in the applications of adaptive array to mobile radio communications.

The most difficult problem in realizing optimum diversity combining on a fading channel is error propagation in the decision directed tracking mode. There are several possible solutions to eliminate or reduce error propagations.

- Blind equalization avoids the use of reference signals, and therefore has no error propagation problem. However, generally speaking, blind equalizers converge too slowly. Recently, various enhancements of the CMA have been investigated. They offer improved convergence rates with increased complexity, but the convergence rates are still slower than that can be achieved by RLS algorithms.

- Delayed decision feedback. Instead of make instantaneous premature de-

cision, we may use convolutional code and feedback more reliable delayed tentative decision from a convolutional decoder. This was used in the slowly time-varying channel. However, in a fast time-varying channel, a decision delay results in poor tracking performance.

To reduce error propagation on a fast fading channel, the challenge is to delay symbol decision without losing tracking speed. Our approach is to use simultaneous weights tracking and decoding.

The diversity optimum combining technique is applicable to both mobile handset and base station, located in a metropolitan area as well as rural and suburban area.

In rural or suburban areas, or in the metropolitan areas with high rise antennas, we have the option of using constrained adaptive array based on DOA's information at a base station. This technique does not require reference signals. Error propagation is avoided.

In a DOA based constrained adaptive array combining system, the presence of coherent signals renders accurate DOA estimation difficult. To make constrained adaptive array beamforming effective in a coherent interference mobile channel environment, and to achieve robustness in DOA estimation, we developed techniques to apply SS to two dimensional arrays to decorrelate the coherent signals and to make high resolution DOA estimation methods MUSIC and ESPRIT operative in a coherent interference environment.

1.4 Major Contributions

To examine the feasibility of diversity combining based adaptive arrays in different wireless systems, we investigate their performance under various fading rates, various SNR and ISR environment, under both training mode and tracking mode. We found that the error propagation in the decision directed tracking mode severely degraded the BER performance.

We developed a novel adaptive diversity combining technique with QRD-RLS based parallel weights tracking and M-D decoding algorithms with moderate increase in complexity, this system significantly reduces error propagation in the decision directed array systems while maintaining the same tracking speed. The M-D algorithm was first developed for convolutional code and then extended to the Trellis coded modulation (TCM). Computer simulation has confirmed that this technique has much better performance than that of the conventional technique.

We studied and evaluated different weights tracking algorithms. Based on our study, we recommend the numerically stable QRD-RLS algorithm which, when implemented in the data domain, can almost double the tolerable interference level compared to the standard RLS algorithms under low fading rate situations. We also studied the AGC and A/D effects on the signals and array weights extraction and revealed an unstability problem caused by lower bits A/D converter. We developed an exact initialization method for complex QRD-RLS algorithms to address this problem and to reduce the total computational complexity.

To significantly improve the robustness of DOA estimation and of beamforming and to estimate both azimuth and elevation angles in a 3D multipath mobile radio environment, we developed techniques for applying SS to arrays of nonlinear geometry. We found and proved the necessary and sufficient conditions on an array

configuration for applying SS. This array must have an orientational invariance structure with an ambiguity free center array, and the number of subarrays must be larger than or equal to the size of the largest group of coherent signals. We also studied the cause of ambiguities in a multipath environment. We found the necessary and sufficient conditions for a three-sensor array manifold to be ambiguity free and identified several higher order ambiguity situations. If an array is also central symmetric, the forward/backward spatial smoothing can be used to improve the resolution. Finally, we expanded the application of our technique to MUSIC and adaptive beamforming algorithms as well as to ESPRIT algorithm. All the predicted results have been verified by simulations.

We applied DOA based adaptive array technique to TDMA mobile communications. We showed the effectiveness of our two dimensional spatial smoothing and DOA based adaptive array on suppressing interference under some multipath fast fading mobile environments.

For DS/CDMA system, we developed a reference-signal-based simultaneous diversity combining and decoding technique to reduce fast fading and suppress interference caused by poor synchronization and power control.

1.5 Outline of the Dissertation

In the next four chapters, we investigate the two kinds of adaptive arrays that are most applicable to the TDMA wireless system. The adaptive array based on optimum diversity combining is covered in Chapter 2 and 3. The adaptive array based on DOA is covered in Chapter 4 and 5. The application of adaptive arrays in DS/CDMA system is discussed in Chapter 6.

In Chapter 2, we introduce the principle of optimum diversity combining and several conventionally used weights tracking algorithms. We present a new exact initialization method for the complex QRD-RLS algorithm. Simulation results are presented to compare different weights tracking algorithm and to study different effects on the performance of adaptive array. We find that error propagation in the decision directed tracking mode causes severe performance degradation.

In Chapter 3, the idea of simultaneous diversity combining and decoding is introduced. To realize it, computationally efficient D-symbol delay algorithm and M-D algorithm are then developed. QRD based parallel weights tracking techniques are presented. The performance and the computational complexity of our adaptive diversity combining system are analyzed. The D-symbol delay and M-D algorithms are extended to the TCM signals. Computer simulation results are provided to demonstrate the significantly improved performance of our techniques.

In Chapter 4, we introduce MUSIC and SS, and state the limitation of linear arrays. We prove the necessary and sufficient conditions on a two dimensional array for applying SS, and consider the FBSS technique for applications in two dimensional array. We study the cause of ambiguities in a multipath signal environment. We expand our results to ESPRIT.

In Chapter 5, we discuss some practical issues of using SS and verify the theoretical results obtained in Chapter 4 using computer simulations. We demonstrate the effectiveness of our SS under multipath multiuser TDMA wireless communication environment. We present our techniques for suppressing interference and achieving capacity increase using DOA based adaptive array.

In Chapter 6, we discuss the application of combined weights tracking and decoding technique in DS/CDMA system. We developed reference-signal-based

diversity combining schemes with predicted weights and orthogonal convolutional codes.

Chapter 7 summarizes the results obtained in this dissertation and proposes some suggestions for future research.

Chapter 2

Diversity Combining

2.1 Introduction

Previous theoretical studies, based upon the ideal case of knowing complete information about the fading channel characteristics showed that antenna arrays with optimum combining can reduce multipath fading of the desired signal and suppress interfering signals [9], thereby increasing both the performance and capacity of wireless systems. In the simulation study of adaptive array in the digital Mobile Radio System IS-54 with flat fading [8], no error propagation in the decision directed tracking mode was assumed and a computational intensive DMI algorithm was used. The purpose of the work presented in this chapter is to evaluate the performance of adaptive arrays in different situations: ideal case, training mode and tracking mode, and to choose a best available adaptive algorithm for array weights acquisition and tracking.

To achieve optimum combining in fast fading environment, we need to have a fast convergence adaptive signal combining algorithm which can track the fading channel. The DMI algorithm has been shown to have a fast convergence rate

and good tracking capability compared to the low complexity LMS algorithm. In this work, we investigate several RLS algorithms which are mathematically equivalent to the DMI algorithm but computationally more efficient for symbol by symbol weight updating. We completed a simulation study using different algorithms to determine the maximum tolerable interference for a desired signal to achieve a 0.01 BER performance. We found that, at a low fading rate, QRD-RLS (QR decomposition based recursive least square) algorithm provides a much larger dynamic range, and can almost double the tolerable interference level compared to other standard RLS algorithms and modified DMI algorithm. Its computational complexity is, nevertheless, no more than the others.

Through computer simulations, we found that the decision errors in the tracking mode cause error propagation and cause severe degradation in BER performance. On the other hand, in the training mode, an interference that is 40dB stronger than the desired signal can be suppressed at a fading rate 80Hz and symbol rate 24.3ksym/sec by using QRD-RLS for weights acquisition and tracking. This fading rate is corresponding to a vehicle speed of 60 miles/hr in IS-136 system.

Using QRD-RLS for weights acquisition and tracking, we found that in the training mode, an interference that is 40dB stronger than the desired signal can be suppressed at a fading rate 80Hz and symbol rate 24.3ksym/sec. This fading rate is corresponding to a vehicle speed of 60 miles/hr in IS-136 system. However, in the tracking mode, decision errors cause error propagation and cause severe degradation in BER performance.

We also studied the AGC (Automatic Gain Control) and A/D effects on the signals and array weights extraction and revealed an unstability problem caused by a lower bits A/D converter. We developed an exact initialization method for

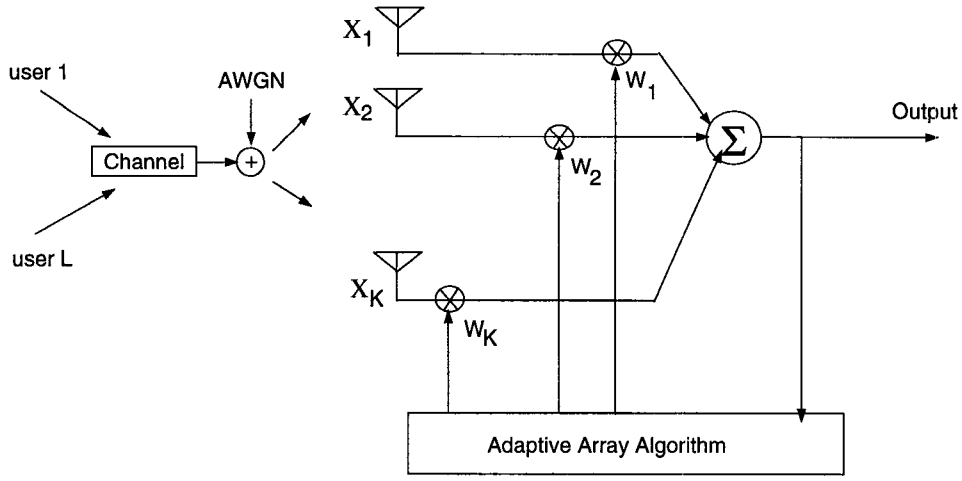


Figure 2.1: Block diagram of adaptive array

complex QRD-RLS algorithms to address this problem and to reduce the total computational complexity.

This chapter is organized as follows: In section 2.2, we introduce the principle of optimum diversity combining and several conventionally used weights tracking algorithms. In section 2.3, we present QRD-RLS weights tracking algorithm and a new exact initialization of the complex QRD-RLS algorithm. Simulation results are provided in section 2.4 to compare different weights tracking algorithm and to study different effects on the performance of adaptive array. In section 2.5, we analyze the error propagations in the decision directed tracking mode. We show the severe performance degradations caused by that. We will discuss the algorithms of reducing of error propagation in the next section.

2.2 Diversity Combining

2.2.1 Ideal Optimum Combiner

Fig. 2.1 shows a block diagram of a M-element adaptive array handset. The received signal vector consists of the desired signal vector $\mathbf{x}_d$, thermal noise vector $\mathbf{x}_n$, and interference vectors $\mathbf{x}_j, j = 1, 2, \cdots, L$ and therefore, can be expressed as

$$\mathbf{x} = \mathbf{x}_d + \mathbf{x}_n + \sum_{j=1}^{L} \mathbf{x}_j \tag{2.1}$$

$$= \mathbf{u}_d s_d(k) + \mathbf{x}_n + \sum_{j=1}^{L} \mathbf{u}_j s_j(k) \tag{2.2}$$

where $\mathbf{u}_d$ and $\mathbf{u}_j$ are the desired and jth interfering signal channel distortion vectors, respectively and $s_d(k)$ and $s_j(k)$ are the desired and jth interfering signals. Assuming the desired signal, noise, and interfering signals are uncorrelated, and

$$E[s_d^2(k)] = 1 \tag{2.3}$$

$$E[s_j^2(k)] = P_j \tag{2.4}$$

Then the received signal correlation matrix can be expressed as

$$\mathbf{R}_{xx} = E(\mathbf{x}\mathbf{x}^H) = \mathbf{u}_d\mathbf{u}_d^H + \sigma^2 I + \sum_{j=1}^{L} P_j \mathbf{u_j}\mathbf{u_j}^H \tag{2.5}$$

where σ^2 is the noise power and I is the identity matrix.

The difference between the desired array response and the actual array output signal defines an error signal $\epsilon(n)$:

$$\epsilon(n) = d(n) - \mathbf{w}^H\mathbf{x}(n) \tag{2.6}$$

The equation for the weights that minimize the mean-square error is [19]

$$\mathbf{w}_{opt} = \mathbf{R}_{xx}^{-1}\mathbf{r}_{xd} = \mathbf{R}_{xx}^{-1}\mathbf{u}_d^* \tag{2.7}$$

In the following simulations, the weights in the ideal case are calculated by Eq. 2.7.

2.2.2 Weight Generation

In practice, only a finite number of data are available for processing. MSE is practically impossible to be calculated exactly. So we use the method of least squares instead. We choose the tap weights of the adaptive array $\mathbf{w}$, so as to minimize the cost function that consists of the sum of error squares:

$$\mathcal{E}(n) = \sum_{i=n_0}^{n} \mid e(i) \mid^2 \tag{2.8}$$

Here, we employ the sliding window as the data weighting function. In the method of exponentially weighted least squares, we use an exponential window as the data weighting function. We minimize the cost function

$$\mathcal{E}(n) = \sum_{i=1}^{n} \lambda^{n-i} |e(i)|^2 \tag{2.9}$$

where λ is a forgetting factor between 0 and 1. The optimum value of the tap-weight vector, $\mathbf{w}(n)$, at which the cost function $\mathcal{E}(n)$ attains its minimum value is defined by the normal equation written in a matrix form:

$$\hat{\mathbf{R}}_{xx}\hat{\mathbf{w}}(n) = \hat{\mathbf{r}}_{xd} \tag{2.10}$$

In this work, we only consider the use of an exponential window. The use of a sliding window will be considered in the future. By using the exponential window, $\hat{\mathbf{R}}_{xx}$ is defined by

$$\hat{\mathbf{R}}_{xx} = \sum_{i=1}^{n} \lambda^{n-i} \mathbf{x}(i)\mathbf{x}^H(i) \tag{2.11}$$

and

$$\hat{\mathbf{r}}_{xd} = \sum_{i=1}^{n} \lambda^{n-i} \mathbf{x}(i)d^*(i) \tag{2.12}$$

To track time-varying signals, we update the weights upon the arrival of new data at time n and the estimate of the weights at time $n-1$. Several methods can be used for updating:

2.2.3 The modified DMI method

In the modified DMI method [8], we use the following recursion to update the value of correlation matrix:

$$\hat{\mathbf{R}}_{xx}(n) = \lambda \hat{\mathbf{R}}_{xx}(n-1) + \mathbf{x}(n)\mathbf{x}^H(n) \tag{2.13}$$

Similarly, we use the following recursion for updating the cross-correlation vector between the received signal and the desired response:

$$\hat{\mathbf{r}}_{xd}(n) = \lambda \hat{\mathbf{r}}_{xd}(n-1) + \mathbf{x}(n)d^*(n) \tag{2.14}$$

We calculate $\mathbf{w}$ on a symbol-by-symbol basis as follows:

$$\mathbf{w(n)} = \hat{\mathbf{R}}_{xx}(n)^{-1}\hat{\mathbf{r}}_{xd}(n) \tag{2.15}$$

In Eq.(2.15), we assumed that $\hat{\mathbf{R}}_{xx}$ is nonsingular so that $\hat{\mathbf{R}}_{xx}$ exists. If not, we can use SVD(Singular Value Decomposition) to find $\mathbf{w}$.

This method needs to determine the inverse of the correlation matrix for each symbol interval. It can be numerically unstable and can be time consuming, particularly if the number of weights is large.

2.2.4 The standard recursive least squares(RLS) algorithm

To avoid performing matrix inverse every time, and to compute the weights recursively, the standard RLS algorithm uses a matrix inversion lemma to update the inverse of the correlation matrix.

According to the *matrix inversion lemma* [20], we express the inverse of the correlation matrix as follows:

$$\hat{\mathbf{R}}_{xx}^{-1}(n) = \lambda^{-1}\hat{\mathbf{R}}_{xx}^{-1}(n-1) - \frac{\lambda^{-2}\hat{\mathbf{R}}_{xx}^{-1}(n-1)\mathbf{x}(n)\mathbf{x}^H(n)\hat{\mathbf{R}}_{xx}^{-1}(n-1)}{1+\lambda^{-1}\mathbf{x}^H(n)\hat{\mathbf{R}}_{xx}^{-1}(n-1)\mathbf{x}(n)} \tag{2.16}$$

We can then update $\mathbf{w}$ accordingly.

The standard RLS algorithm requires an initialization of $\hat{\mathbf{R}}_{xx}$. Two methods in the literature can be used to perform such initialization. The soft-constraint initialization has proved statistical performance and was shown to be error bounded [21]. We adopt this method in our simulation. We set $\hat{\mathbf{R}}_{xx}(0) = \delta\mathbf{I}$. The recommended choice of δ is that it should be small compared to $0.01\sigma_u^2$, where σ_u^2 is the variance of a data sample $\mathbf{x}(n)$ [21]. However, without a *priori* information about the interference and the Gaussian noise, an arbitrary set of the initial value may cause a bias in the weight estimation.

Another problem with the standard RLS algorithm is that as DMI algorithms, standard RLS is implemented in the covariance domain. As we will see in the next section, it is numerically unstable compared to the one implemented in the data domain.

2.3 QRD-RLS algorithm

An equivalent cost function of Eq.(2.9) in matrix notation is given as

$$\mathcal{E}(n) = \|\mathbf{\Lambda}^{1/2}(n)\epsilon(n)\|^2 \tag{2.17}$$

where $\mathbf{\Lambda}$ is the exponential weighting matrix given by

$$\mathbf{\Lambda}(n) = diag[\lambda^{n-1}, \lambda^{n-2}, \cdots, 1] \tag{2.18}$$

and the error vector ϵ equals to

$$\epsilon(n) = \mathbf{d}(n) - \mathbf{A}(n)\mathbf{w}(n) \tag{2.19}$$

where the vector $\mathbf{d}(n)$ denotes the desired data vector, and $\mathbf{A}(n)$ denotes the data matrix. The norm of a vector is unaffected by premultiplication by a unitary

matrix. Hence, we may express the cost function $\mathcal{E}$ as

$$\mathcal{E}(n) = \|\mathbf{Q}(n)\mathbf{\Lambda}^{1/2}(n)\epsilon(n)\|^2 \tag{2.20}$$

where $\mathbf{Q}(n)$ is an unitary matrix. It can be deduced [20] that $\mathcal{E}(n)$ is minimized when $\hat{\mathbf{w}}(n)$ satisfies the condition

$$\mathbf{R}(n)\mathbf{w}(n) = \mathbf{p}(n) \tag{2.21}$$

where $\mathbf{Q}(n)$ is generated in such a way that it applies an orthogonal triangularization to the weighted data matrix $\mathbf{\Lambda}^{1/2}(n)\mathbf{A}(n)$, as shown by

$$\mathbf{Q}(n)\mathbf{\Lambda}^{1/2}(n)\mathbf{A}(n) = \begin{bmatrix} \mathbf{R}(n) \\ \mathbf{0} \end{bmatrix} \tag{2.22}$$

$\mathbf{R}$ is an upper triangular matrix, and $\mathbf{0}$ is an null matrix. $\mathbf{p}(n)$ is a vector defined by

$$\mathbf{p}(n) = \mathbf{F}(n)\mathbf{\Lambda}^{1/2}(n)\mathbf{d}(n) \tag{2.23}$$

where $\mathbf{F}(n)$ consists of the first M rows of $\mathbf{Q}(n)$. Thus Eq.(3.12) may be readily solved for the optimum weight vector $\mathbf{w}(n)$ by a process of back substitution. To solve the least squares problem recursively, we use a sequence of Givens rotations to annihilate all M elements in the new incoming data $\mathbf{x}(n)$ one by one. This procedure is shown as follows:

$$\begin{bmatrix} \mathbf{R}(n) \\ \mathbf{0} \end{bmatrix} = \mathbf{T}(n) \begin{bmatrix} \lambda^{1/2}\mathbf{R}(n-1) \\ \mathbf{0} \\ \mathbf{u}^H(n) \end{bmatrix} \tag{2.24}$$

where $\mathbf{T}(n)$ is the unitary matrix denoting the combined effect of a sequence of Givens rotations:

$$\mathbf{T}(n) = \mathbf{J}_M(n), \cdots, \mathbf{J}_2(n)\mathbf{J}_1(n) \tag{2.25}$$

The QRD-RLS algorithm has good numerical properties. In our simulations, we use a single precision floating point format. We found that QRD-RLS can almost double the tolerable interference power level compared to a standard RLS algorithm. This is because the QRD-RLS algorithm is implemented directly in the data domain $\mathbf{A}(n)$, while the standard RLS algorithm or modified DMI is implemented in the covariance domain $\mathbf{R}_{xx}$. Suppose, there is only one interference, if I/S ratio is P_1 in the covariance domain, then I/S is $\sqrt{P_1}$ in the data domain. Under the same finite precision, this results in a higher signal to quantization noise ratio in the data domain than in the covariance domain. Therefore, a higher dynamic range is obtained. Equivalently, in order to suppress the same amount of interference, the QRD-RLS algorithm requires a significantly shorter word-length than the standard RLS algorithm.

2.3.1 Exact Initialization of the complex QRD-RLS Algorithm

We have seen in the previous section that the orthogonal triangularization of the data matrix is recursively updated using a sequence of Givens rotation as each new set of data enters the computation. A Givens rotation used to annihilate the *kth* element of a new data vector $\mathbf{x}$ is given as:

$$\mathbf{J}_k(n) = \begin{bmatrix} \mathbf{I} & & & \mathbf{0} \\ & c & \mathbf{0}^t & s^* \\ & \mathbf{0} & I & \mathbf{0} \\ & -s & \mathbf{0}^t & c \end{bmatrix} \tag{2.26}$$

The real cosine parameter c and the complex sine parameter s have to satisfy the following conditions:

$$c^2 + |s|^2 = 1 \tag{2.27}$$

$$x_k(n) \rightarrow -sr_{kk} + cx_k(n) = 0 \tag{2.28}$$

where r_{kk} is the kth diagonal element of the matrix $\mathbf{R}(n)$. For a complex input $\mathbf{x}(n)$, c and s are given by

$$c = \frac{|r_{kk}|}{\sqrt{|r_{kk}|^2 + |x_k(n)|^2}} \tag{2.29}$$

$$s = \frac{x_k(n)}{r_{kk}} c \tag{2.30}$$

Now suppose that we begin the time recursions at $n = 2$ when we have two vectors $\mathbf{x}(1)$ and $\mathbf{x}(2)$. We transpose and stack these two vectors to form a matrix with two rows. Then we apply a 2×2 Givens rotation matrix to eliminate the first element of the row vector $\mathbf{x}(2)$:

$$\mathbf{G}(2) \begin{bmatrix} \sqrt{\lambda}\mathbf{x}_1(1) & \sqrt{\lambda}\mathbf{x}_2(1) & \cdots & \sqrt{\lambda}\mathbf{x}_M(1) \\ \mathbf{x}_1(2) & \mathbf{x}_2(2) & \cdots & \mathbf{x}_M(2) \end{bmatrix} = \begin{bmatrix} r_{11}(1) & r_{12}(1) & \cdots & r_{1M}(1) \\ 0 & x_2^{(2)}(2) & \cdots & x_M^{(2)}(2) \end{bmatrix} \tag{2.31}$$

In our simulation, we found that after going through the AGC and A/D converter, low noise can be totally quantized to zero, and $\mathbf{x}(1)$ and $\mathbf{x}(2)$ can be identical at very low fading rates. This results in $r_{11}(1) = 0$. Thus we cannot further append vector $\mathbf{x}(3)$ to the matrix at the right side of Eq.(2.31) and apply the Givens rotation to eliminate the first two elements of $\mathbf{x}(3)$ using Eq.(2.30). Actually, suppose all the diagonal elements of R are real, the cosine and sine in the Givens rotation can be calculated as follows:

$$c = \frac{r_{kk}}{\sqrt{|r_{kk}|^2 + |x_k(n)|^2}} \tag{2.32}$$

$$s = \frac{x_k(n)}{\sqrt{|r_{kk}|^2 + |x_k(n)|^2}} \tag{2.33}$$

In particular, if we start with a real r_{kk}, then the transformation maintains that property. This also results in reduced computational complexity for further updating. In the following, we developed an exact initialization algorithm for QR decomposition which can eliminate the problem of dividing by zero and guarantee the realization of all the diagonal elements of R in each updating step.

We begin the time recursions at $n = 1$. We multiply a complex number which takes the complex conjugate of the first element of data vector $\mathbf{x}(1)$ and normalize it to 1. This number can be expressed as $e^{-j\theta_1}$ where $\theta_1 = arctan[Im(x_1(1))/Re(x_1(1))]$. At $n = 2$, We append $\mathbf{x}(2)$ to form a matrix with two rows. We apply a 2×2 Givens rotation matrix $G(2)$ with c and s calculated by Eq. 2.32 and 2.33 to eliminate the first element of the row vector $\mathbf{x}(2)$.

$$\begin{bmatrix} c & s^* \\ -s & c \end{bmatrix} \begin{bmatrix} \sqrt{\lambda}e^{-j\theta_1}x_1(1) & \sqrt{\lambda}e^{-j\theta_1}x_2(1) & \cdots & \sqrt{\lambda}e^{-j\theta_1}x_M(1) \\ x_1(2) & x_2(2) & \cdots & x_M(2) \end{bmatrix} \tag{2.34}$$

$$= \begin{bmatrix} r_{11}^{(2)}(1) & r_{12}^{(2)}(1) & \cdots & r_{1M}^{(2)}(1) \\ 0 & x_2^{(1)}(2) & \cdots & x_M^{(1)}(2) \end{bmatrix} \tag{2.35}$$

where

$$c = \frac{\sqrt{\lambda}e^{-j\theta_1}x_1(1)}{\sqrt{(\sqrt{\lambda}e^{-j\theta_1}x_1(1))^2 + |x_1(2)|^2}} \tag{2.36}$$

and

$$s = \frac{x_1(2)}{\sqrt{(\sqrt{\lambda}e^{-j\theta_1}x_1(1))^2 + |x_1(2)|^2}} \tag{2.37}$$

We then multiply a unitary matrix to make the $x_2^{(1)}(2)$ real.

$$\begin{bmatrix} 1 & \\ & e^{-j\theta_2} \end{bmatrix} \begin{bmatrix} r_{11}^{(2)} & r_{12}^{(2)} & \cdots & r_{1M}^{(2)} \\ 0 & x_2^{(1)}(2) & \cdots & x_M^{(1)}(2) \end{bmatrix} = \begin{bmatrix} r_{11}^{(2)} & r_{12}^{(2)} & \cdots & r_{1M}^{(2)} \\ 0 & r_{22}^{(2)} & \cdots & r_{2M}^{(2)} \end{bmatrix} \tag{2.38}$$

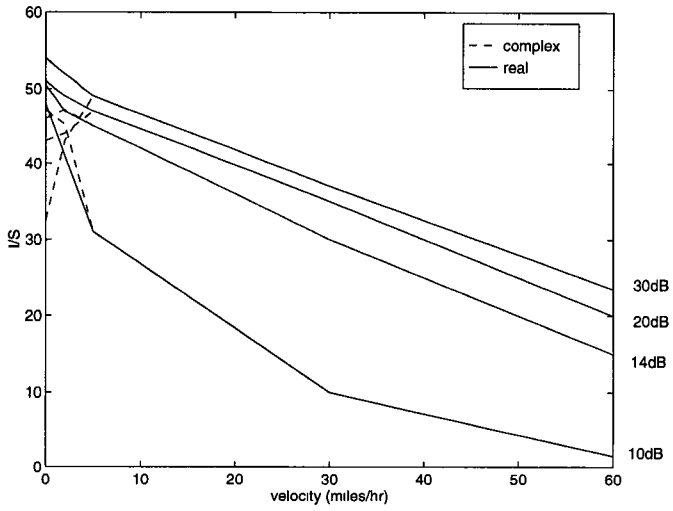


Figure 2.2: Comparison of different QRD-RLS initialization schemes

where $\theta_2 = arctan[Im(x_2^{(1)}(2))/Re(x_2^{(1)}(2))]$. At $n = 3$, we append vector $\mathbf{x}(3)$ to the matrix at the right side of Eq. 2.38 to form a three-row matrix. We eliminate the first two elements of $\mathbf{x}(3)$ using two Givens rotations $J_2(3)$ and $J_1(3)$.

$$J_2(3)J_1(3)\begin{bmatrix} r_{11}^{(2)} & r_{12}^{(2)} & \cdots & r_{1M}^{(2)} \\ 0 & r_{22}^{(2)} & \cdots & r_{2M}^{(2)} \\ x_1(3) & x_2(3) & \cdots & x_M(3) \end{bmatrix} = \begin{bmatrix} r_{11}^{(3)} & r_{12}^{(3)} & r_{13}^{(3)} & \cdots & r_{1M}^{(3)} \\ 0 & r_{22}^{(3)} & r_{23}^{(3)} & \cdots & r_{2M}^{(3)} \\ 0 & 0 & x_3^{(2)}(3) & \cdots & x_M^{(2)}(3) \end{bmatrix} \tag{2.39}$$

We then multiply a unitary matrix to make the $x_3^{(2)}(3)$ real.

$$\begin{bmatrix} 1 & & \\ & 1 & \\ & & e^{-j\theta_3} \end{bmatrix}\begin{bmatrix} r_{11}^{(3)} & r_{12}^{(3)} & r_{13}^{(3)} & \cdots & r_{1M}^{(3)} \\ 0 & r_{22}^{(3)} & r_{23}^{(3)} & \cdots & r_{2M}^{(3)} \\ 0 & 0 & x_3^{(2)}(3) & \cdots & x_M^{(2)}(3) \end{bmatrix} = \begin{bmatrix} r_{11}^{(3)} & r_{12}^{(3)} & r_{13}^{(3)} & \cdots & r_{1M}^{(3)} \\ 0 & r_{22}^{(3)} & r_{23}^{(3)} & \cdots & r_{2M}^{(3)} \\ 0 & 0 & r_{33}^{(3)} & \cdots & r_{3M}^{(2)} \end{bmatrix} \tag{2.40}$$

By continuing this process, we can construct a triangular matrix at $n = M$ with all diagonal elements real.

Fig. 2.2 shows simulation results of different initialization schemes. With the one without realization, we can see an obvious performance degradation caused by the problem described earlier in this section. In contrast, our method shows no evidence of numerical instability.

2.4 BER Performance

5 In this section, we present our simulation results to compare different algorithms and to evaluate the performance of adaptive arrays quantitatively.

2.4.1 Radio Channel Specifications

In the adaptive array system we simulated, the transmitted signal is modulated by QPSK. The carrier frequency is 2 GHz. The modulation data rate is 16kb/s. The channel is time-division-multiplexed. There are 162 symbols in each time slot. BER were obtained based on an average over 1000 slots.

A useful index of the rate at which the mobile radio channel varies with time is the symbol-normalized fade rate [22] $\gamma = f_D T_s$, where f_D is the maximum Doppler frequency and T_s is the inverse of the channel symbol transmission rate. Here, we use an equivalent index **symbol vary period**, denoted as N_{fd}, which is given by $N_{fd} = 1/\gamma = 1/f_D T_s$. Table 2.1 shows this index for different systems for vehicle speed at 60 miles/hr.

According to this index, we can establish a relationship between different systems. This relationship is shown in Fig 2.3. Although our simulations are based on PCS system, a corresponding performance index can be found for other TDMA systems.

System	N_s	T_s	f_D at 60miles/hr	N_{fd}
PCS	162	62.5μsec	178.8 Hz	89.5
IS-136	162	41.5μsec	80 Hz	301.2
GSM	156.25	3.692μsec	100 Hz	2708.6

Table 2.1: Adaptive Array Design Parameters for TDMA Systems

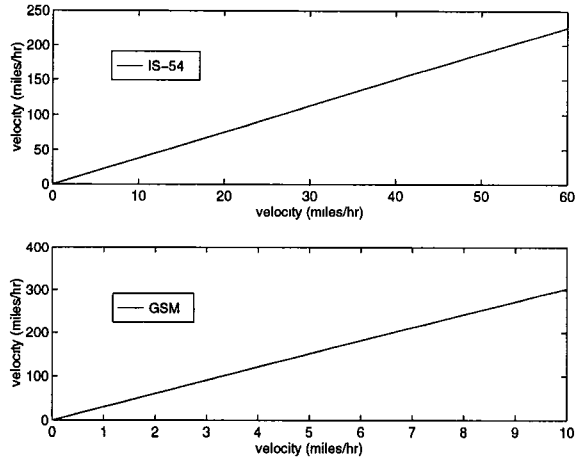


Figure 2.3: Equivalent Fading Velocities

2.4.2 Algorithms Comparison

The computer simulation results shown below illustrate the effect of finite precision on the performance of adaptive array, based on different algorithms. A single precision floating point format is used in all these approaches. Fig. 2.4 depicts the boundary conditions to achieve 0.01 BER. Here, we assume

- The reference signal is the transmitted signal.

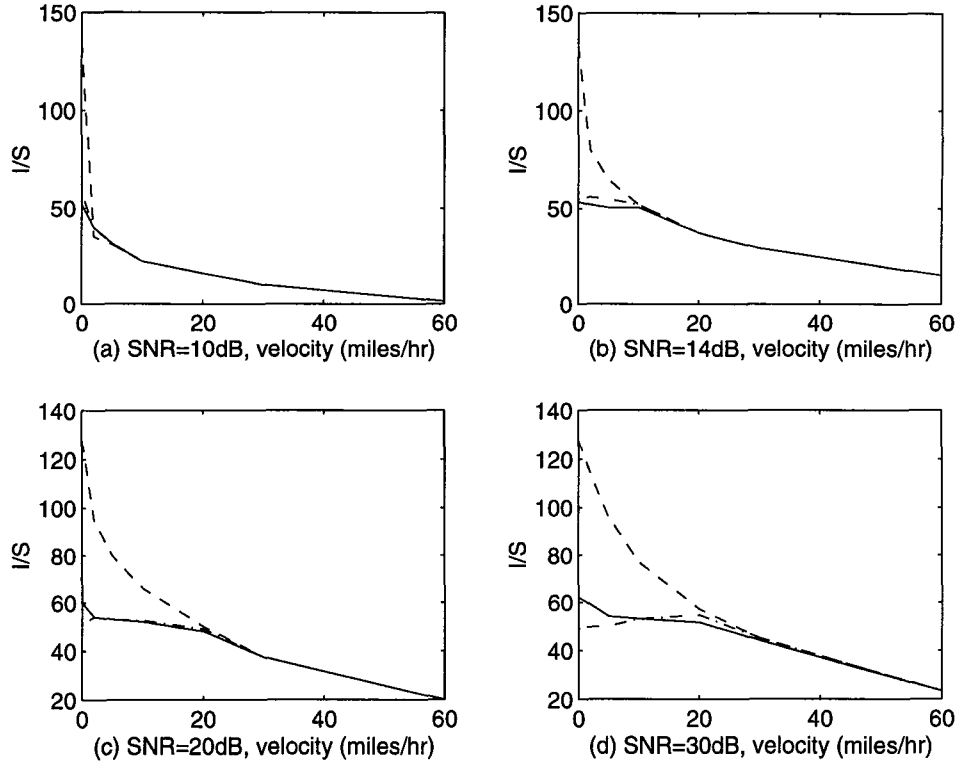


Figure 2.4: The boundary conditions for achieving 0.01 BER, without considering the finite precision A/D effect. Both signal and interference move at the same speed, QRD-RLS('- -'), DMI('-') and standard RLS('-.')

- Both vehicles move at the same speed.
- A/D converter has more than 32 bits precision and has negligible effect compared to the single precision effect in DSP processing.
- The forgetting factor λ was optimized at speed of 60 miles/hr and was found to be 0.74.

In our simulation we found that the performance is almost stable for a small change of λ, e.g, from 0.72 to 0.78.

In Fig. 2.4, interference to noise ratio is plotted as a function of the vehicle velocity. From Fig. 2.4 we can make the following observations:

- Although RLS has a faster convergence rate comparing to other adaptive algorithms such as LMS and CMA, the tracking ability is still limited and loss of tracking is observed at a higher fading rate.
- At a lower fading rate, QRD-RLS shows supreme performance, can suppress interference that is much stronger than the desired signal. The DMI or the standard RLS has a limited interference suppressing capability because of a much more severe finite precision effect in the covariance domain which distorts the desired signal at a high interference level.

The complexity of the different RLS algorithms in terms of the number of additions, multiplications, divisions per iteration is shown in Table 2.2.

In general, addition and multiplication operations can be achieved within one instruction cycle by most state-of-the-art digital signal processors such as TMS320C40, one division operation can be finished within 7 cycles, and one inverse square root operation can be finished within 10 cycles. Thus both RLS algorithms have about

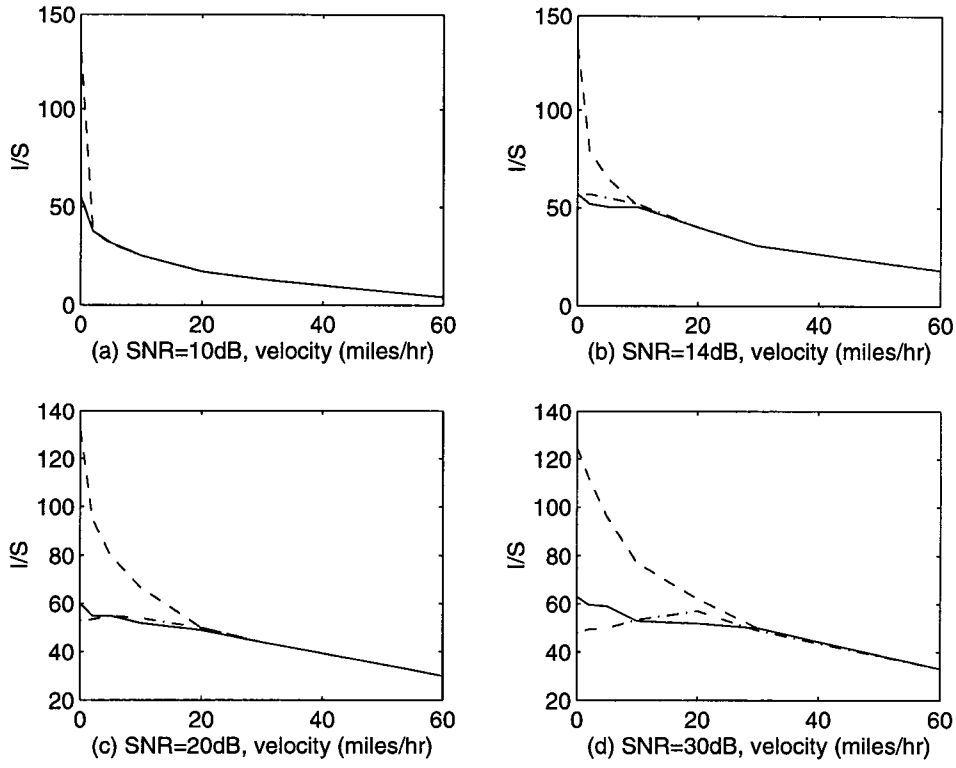


Figure 2.5: The boundary conditions for achieving 0.01 BER, without considering the finite precision A/D effect. The desired signal moves at 5 miles/hr, QRD-RLS('- -'), DMI('-') and standard RLS('-.')

RLS agorithm	QRD-RLS	m=4	Standard RLS Version 2(Version 1)	m=4
Multiplication (1 cycle)	$9m^2 + 12m$	156	$14(10)m^2 + 18m + 2$	298(234)
Addition	$6m^2 + 4m$	112	$12m^2 + 10m$	232
Division (7 cycle)	$2m$	56	2	14
Reciprocal Square Root (10 cycle)	m	40		
Approximate Total Cycle		252		312(248)

Table 2.2: Complexity Comparison Between Standard RLS and QRD-RLS used for Adaptive Array Weights Tracking

the same computational complexity. Table 2.3 lists the number of instruction per second required in QRD-RLS algorithm for different systems. We can see that for PCS and IS-136, this level of MIPS is well within the capabilities of TMS320C40. For a GSM system, this complexity is not acceptable using a single DSP processor, allowing However, in a GSM system, the symbol fade rate is low even for a vehicle speed at about 60 miles/hr, it allows us to update weights every 10 to 20 symbols. We can also adopt block processing techniques [56]. Thus the complexity is again within the capabilities of modern programmable DSP processors.

PCS	4 MIPS
IS-136	6 MIPS
GSM	68 MIPS

Table 2.3: MIPs required in different systems

2.4.3 Several Factors in Tracking

1. Tracking relative to velocity

   The fading rate of the desired signal and of the interference are generally not the same. The fading rates can be even more different when an indoor user is interfered by an outdoor user. In Fig. 2.5 an interference to noise ratio is plotted as a function of the interference handset moving velocity. Here, we assume that the desired handset moves at 5 miles/hr. Comparing this result with the one shown in Fig. 2.4, we observe that the BER is lower in Fig.5 when interference velocity is faster than 5 miles/hr. This is consistent with common sense that it is more difficult to track two faster moving vehicles than to track one. The relative variation rate of signal to interference ratio depends both on the variation rate of the signal and the variation rate of the interference. This relationship is shown below:

$$\frac{\frac{\partial}{\partial t}\left(\frac{P_s(t)}{P_I(t)}\right)}{\frac{P_s}{P_I}} = \frac{\frac{\partial P_s(t)}{\partial t}P_I + P_s\frac{\partial P_I(t)}{\partial t}}{P_I^2}\frac{P_I}{P_s} = \frac{\frac{\partial P_s}{\partial t}}{P_s} + \frac{\frac{\partial P_I}{\partial t}}{P_I} \tag{2.41}$$

   In the following simulation, we assume that both the desired signal and interference have the same fading rate.

2. Tracking relative to SNR

Fig. 2.4 and Fig. 2.5 show that the interference suppressing level increases nonlinearly with SNR. When SNR increases from 10dB to 14dB, the improvement is significant. However, when the SNR increases from 20dB to 30dB, the improvement is less obvious. At a very low fading rate, an adaptive array can suppress interference that is much stronger than the desired signal even when the SNR is low. In this situation, the signal and noise level change very slowly in one time slot. Gaussian noise is the dominant effect. However, for fast fading channels, the joint effect of high level Gaussian noise and fast changing channel distortion make it difficult for all the adaptive algorithms to extract the phase information of the desired signal. When the noise level is relatively high e.g., 20dB, the fast changing channel distortion is the dominant effect on weight tracking. In indoor communication, signal levels are usually lower. But cellular phones(car phones) usually get signal power several dB higher relative to indoor [3] signals. In most cases, we don't have to be concerned with fast fading coupled with low SNR.

3. Tracking relative to the forgetting factor

   In our simulation, the forgetting factor λ is optimized when both vehicles move at 60 miles/hr. The smaller the λ, the faster the convergence rate but the higher the misadjustment. At a higher fading rate, we need to choose a smaller λ for fast tracking. At a lower fading rate a larger λ can improve the performance. In most cases, we don't know the vehicle velocity in advance. Therefore, we use the λ optimized for 60miles/hr, for all velocities. We are most interested in the adaptive array performance at 60 miles/hr.

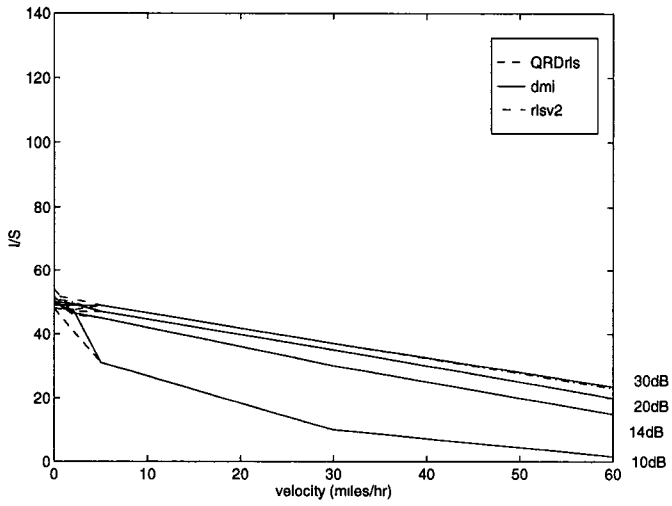


Figure 2.6: The boundary conditions for achieving 0.01 BER, with a 10 bits A/D converter

2.4.4 AGC and A/D Effect

In a receiver, the received power variation of each branch is regulated by an AGC. The gain is determined by the maximum power of the four branches, and is used on all branches (common AGC). The A/D converter we simulated has 10-bits accuracy.

Fig. 2.6 shows the boundary of 0.01 BER in the system described above. In this simulation we did not consider the effect of decision errors on the reference signal. We have the following observations:

- The lower bit A/D converter dominates the finite precision effect and causes the performance deterioration.

- Due to the finite precision A/D effect, the difference of using different algorithms is not discernable. The single precision floating point format used

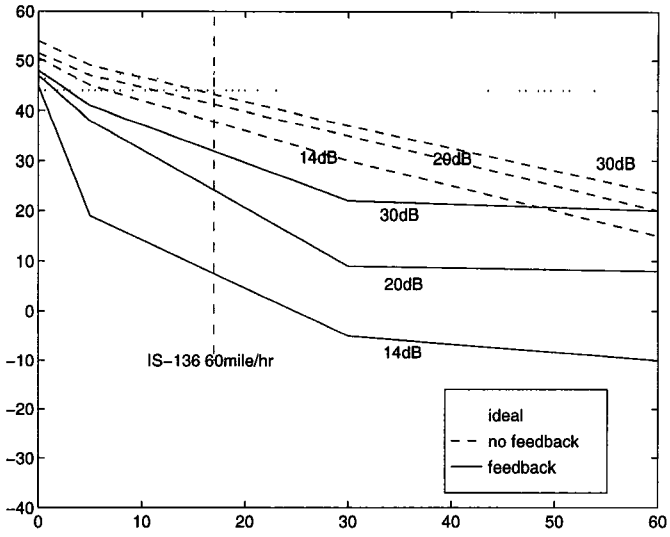


Figure 2.7: The boundary conditions for achieving 0.01 BER

in our DSP processor has more than enough precision compared to the 10 bit precision used in the A/D converter. We should be able to use much lower precision for QRD-RLS than for the standard RLS to achieve the same performance.

- With the assumption that a correct reference signal is always available, an interference which is 20dB higher than the desired signal can be suppressed at SNR of as low as 10dB for a vehicle speed up to 17 miles/hr. This fading rate is equivalent to a fading rate for a vehicle speed up to about 60 miles/hr in the IS-54 system. Meanwhile, we also see a performance loss at a high fading rate.

2.5 Error Propagation Problem

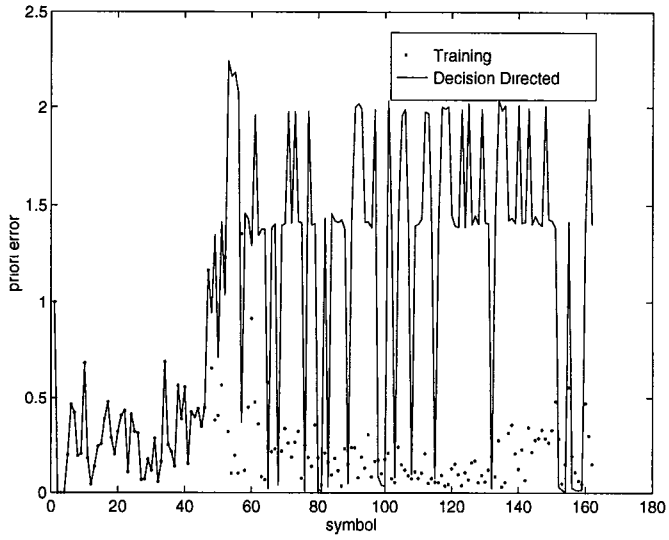


Figure 2.8: Error propagation phenomenon when SNR=20dB, Velocity=60miles/hr I/S=20dB

In the following simulation, we use QRD-RLS algorithm for adaptive array processing. In a system like IS-54, there are only 14 known symbols available for synchronization and training. We have to change to the decision directed mode to further track the fading channel. To obtain the solid line in Fig 2.7, we used the 1-14 symbols as the training sequence and later changed to decision directed mode. It is necessary to change to a decision directed mode to further track the fading channel. We see a severe performance loss compared to our previous results. This is caused by the error propagation in the tracking, i.e., when an decision error is made, this error remains in the estimation of the next set of weights. In our simulation we use a forgetting factor of 0.74, the corresponding window size is small. The decision error weights heavily in the subsequent estimation and will most likely cause a large bias leading to wrong decisions. The further

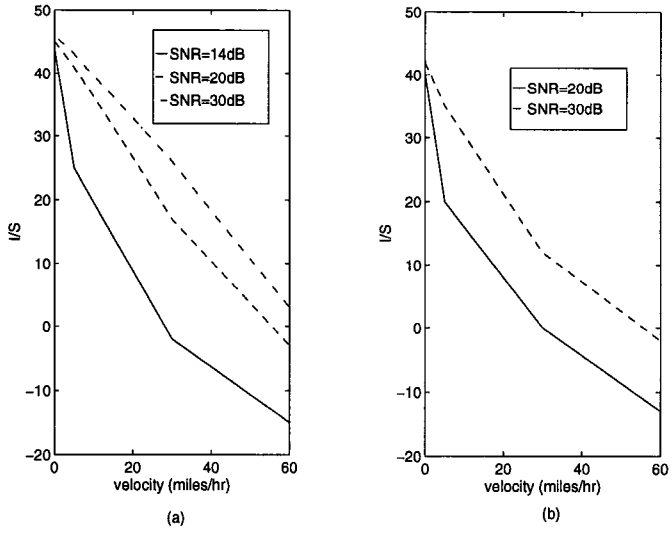


Figure 2.9: The boundary conditions for achieving (a) 0.1 frame error rate (b) 0.01 frame error rate

propagation of errors causes eventually the failing of the subsequent decisions. This phenomenon is shown in Fig. 2.8 which measured the absolute error between the transmited signals and the signals after array combining. We also measured the frame error rate. Fig 2.9 (a) shows the 10% frame error rate boundary. Fig 2.9 (b) shows the 1% frame error rate boundary.

Chapter 3

Simultaneous Diversity Combining and Decoding

3.1 Introduction

In the previous chapter, we have shown that in slowly time-varying mobile radio channels, adaptive diversity combining can reduce multipath fading of desired signal and suppress interfering signals. The effectiveness of optimum adaptive diversity combining on a fast time-varying channel, however, depends on the tracking speed of the adaptive algorithms and on the control of a decision error propagation.

Adaptive optimum diversity combining weights tracking and adaptive channel equalization are two closely related problems. Most of the techniques used for equalization can also be applied to adaptive optimum diversity combining. The majority of previous studies on channel equalization or adaptive array combining have concentrated on slowly time-varying fading channels.

In a slowly time-varying fading channel, LMS algorithm is used most often for channel equalization. A small change in the equalizer weights is capable of

tracking the fading variation. Error propagation is not a severe problem in this situation. Bit Error Rate (BER) performance can be further improved with the combination of an equalizer and a Viterbi decode [53]- [55]. A tentative decision with small delay or no delay from a Viterbi decoder has been used for channel tracking. In contrary, in a fast time-varying fading channel, a decision delay results in poor tracking performance and a premature tentative decision will cause error propagation.

Also, in a slowly time-varying fading channel, blind equalization techniques such as constant modulus algorithm (CMA) can be used to avoid error propagation in the decision-directed channel equalization [56]. Unfortunately, blind equalization algorithms converge slowly and are not capable of tracking fast time-varying fading channel. Although various improved schemes of CMA have been investigated [57], most of them involve significant increases in complexity or computational costs, and the convergence rates are still lower than that can be achieved by RLS algorithms.

Recently, respective-states channel estimation (RCE) [58][59] was proposed, and was reported to have better performance over conventional decision-directed channel equalization on fast time varying fading channels. Nevertheless, the continued use of the Viterbi algorithm (VA) in RCE tends to introduce error when cochannel interferences are strong. Moreover, the complexity of this approach is high. This method has not yet been studied on adaptive array systems.

To effectively perform diversity combining on a fast time-varying fading channel, we developed an adaptive diversity combining system using a M-D decoder. The array weights are tracked by using the QRD-RLS algorithm along each of M surviving paths selected by using an M-D decoding algorithm. M and D are to

be selected according to RLS updating window length and interference to signal ratio. This system significantly reduces error propagation in the decision directed array system while maintaining the same tracking speed. Our technique is first developed for the convolutional encoded signals and then extended to trellis-coded modulation (TCM) signals to increase information bit rate.

This chapter is organized as follows: In section 3.2, the idea of simultaneous diversity combining and decoding is introduced. To realize it, computationally efficient D-symbol delay algorithm and M-D algorithm are developed. QRD based parallel weights tracking techniques are presented. The performance and the computational complexity of our adaptive diversity combining system are analyzed. In section 3.3, the D and M-D algorithms are extended to the TCM signals. In section 3.4, computer simulation results are provided to demonstrate the significantly improved performance of our techniques.

3.2 Simultaneous Weights Tracking and Decoding

We have shown in the previous chapter that a premature symbol by symbol decision is not reliable and will cause error propagation. One solution is to adopt the blind equalization technique, which avoids the use of reference signals; however, its convergence rate is too slow. The other solution is to simply combine array weights updating with convolutional decoding, and feed back more reliable delayed decision from a convolutional decoder. However, a decision delay may cause poor channel tracking.

To reduce error propagation, and to make a more reliable delayed decision

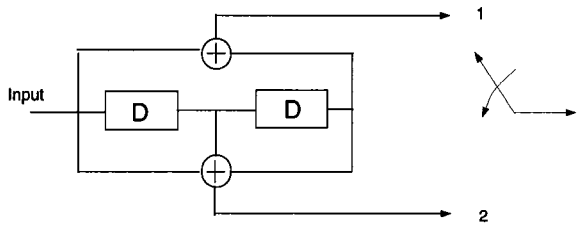


Figure 3.1: (2,1,2) convolutional encoder

but without lossing weights tracking speed, we propose the following simultaneous array weights tracking and decoding technique.

3.2.1 More reliable decision based on D symbols

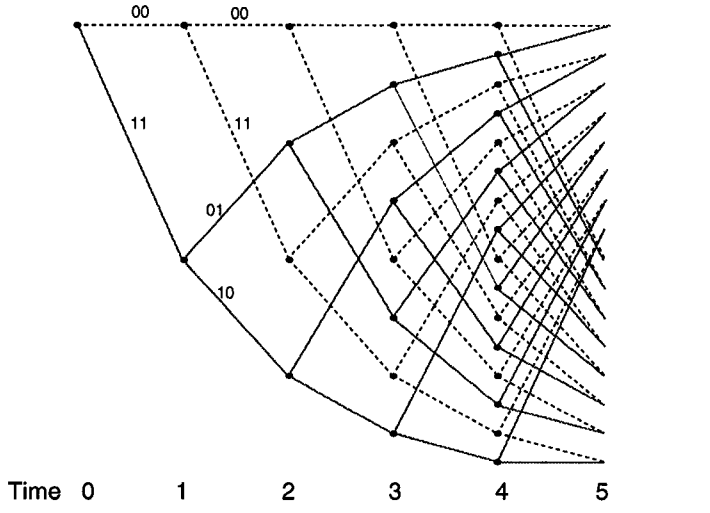
We use a convolutional code shown in Fig. 3.1 to encode the transmitted information bit and use QPSK to modulate the transmitted signals. At each state, two possible QPSK symbols might be transmitted based upon a "0" or a "1" input. Instead of making an immediate bit decision based upon one symbol we make a more reliable symbol decision based upon D symbols.

We elucidate our thinking using the example shown in Fig. 3.2. In Fig. 3.2(a), $D = 5$. At each symbol interval, all the possible sequences in the next D stages are saved. On each path, we perform the following operations:

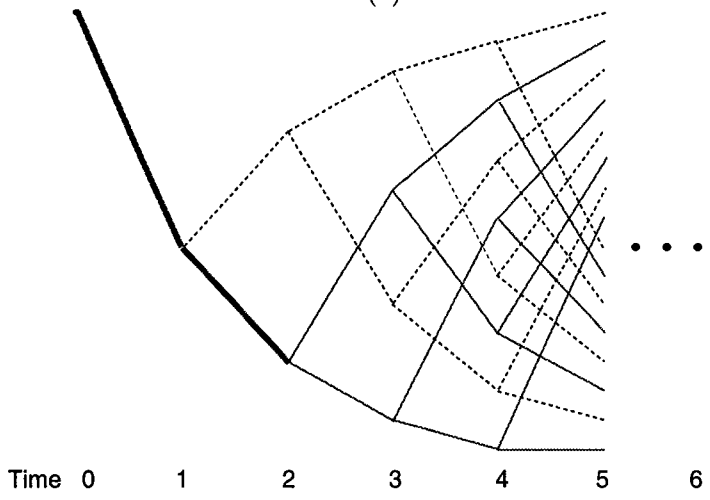
- Update diversity weights using each path's own reference signals, i.e. its own path outputs.

- Calculate the Euclidean distance between array output and path output for each branch as follows:

$$b_i(n) = [\mathbf{w}_i^H(n-1)\mathbf{x}(n) - s_i(n)]^2 \tag{3.1}$$



(a)



(b)

Figure 3.2: An example of delayed bit decision based on 5 symbols

where $\mathbf{w}_i(n-1)$ is a previously updated weighting vector, $\mathbf{x}(n)$ is a currently received array data and $s_i(n)$ is the reference signal which is the path output on each branch.

- Calculate and save the accumulated Euclidean distance which is evaluated by $\sum_{n=1}^{D} b_i(n)$ along its own path.

In Fig. 3.2 (a), all the solid lines originate from a "1" input bit at stage "0" and all the dashed line originate from a "0" input bit. If at stage 0, the transmitted information bit is "1" which corresponds to path output "11", then a correct path must be a solid line path. After D symbol interval, for one half of the solid paths, weights are updated based upon one correct reference signals and $D-1$ wrong reference signals. One fourth of the solid line paths are updated based upon two correct reference signals and $D-2$ wrong reference signals and so on. On the other hand, on each dashed line path, weights are updated erroneously along D branches and will result in a large accumulated path metric. The smallest accumulated path metric of all the solid line paths should be smaller than that of all the dashed line paths. We then decide the input bit back by D stages. That input bit should be the bit that leads to the selected smallest accumulated path metric.

Once the input bit back by D stages is chosen, we save all the branches originating from it and discard the rest. In Fig. 3.2 (a), we discard all the dashed line paths. At the next stage, we repeat this process to choose the bit at stage 1, which is shown in Fig. 3.2 (b).

Along the correct path, weights are updated symbol by symbol. There is no loss of tracking. Moreover, there is no error propagation. We use the correct path's own path output as the reference signal. There is no need to feedback any decided symbol. This way, we make a more reliable symbol decision based on D symbols.

Therefore, we reduce error propagation while keeping the same tracking speed.

3.2.2 D-symbol Delay Algorithm

We introduced our idea for reducing error propagation based upon convolutional code and accumulated path metrics along $D+1$ symbols.

In the following, we will present an efficient D-delay algorithm to perform the simultaneous weights tracking and decoding.

In Fig 3.2 (a), at stage 5, two paths merge into each state before we make decision. After the decision, we discard one path in every pair. We will see in the following that each state only needs to remember one $\mathbf{w}$ and one accumulated path metric. It then contains all the information on the past D symbols. Each time, we only need to update the information contained in the current states as shown in Fig. 3.3.

The decoding algorithm relies on the following key properties of a trellis diagram with 2^D states.

- **Property 1:** The states in the trellis are in the order from 0 to $2^D - 1$. The state of the encoder is set to 0 at the begining of each frame.

  An example is shown in Fig. 3.4 where $D = 4$.

- **Property 2:** The ith bit of each state corresponds to the input bit i stages back.

  For example, in Fig. 3.4, at stage $N-1$, all the states with "0" as the last bit are from a "0" input 4 stages back at stage $N-5$.

  From Fig. 3.4, we observe that a state $X_1X_2\cdots X_D$ can only be reached from the previous state that is either $X_2\cdots X_{D-1}0$ or $X_2\cdots X_{D-1}1$. This

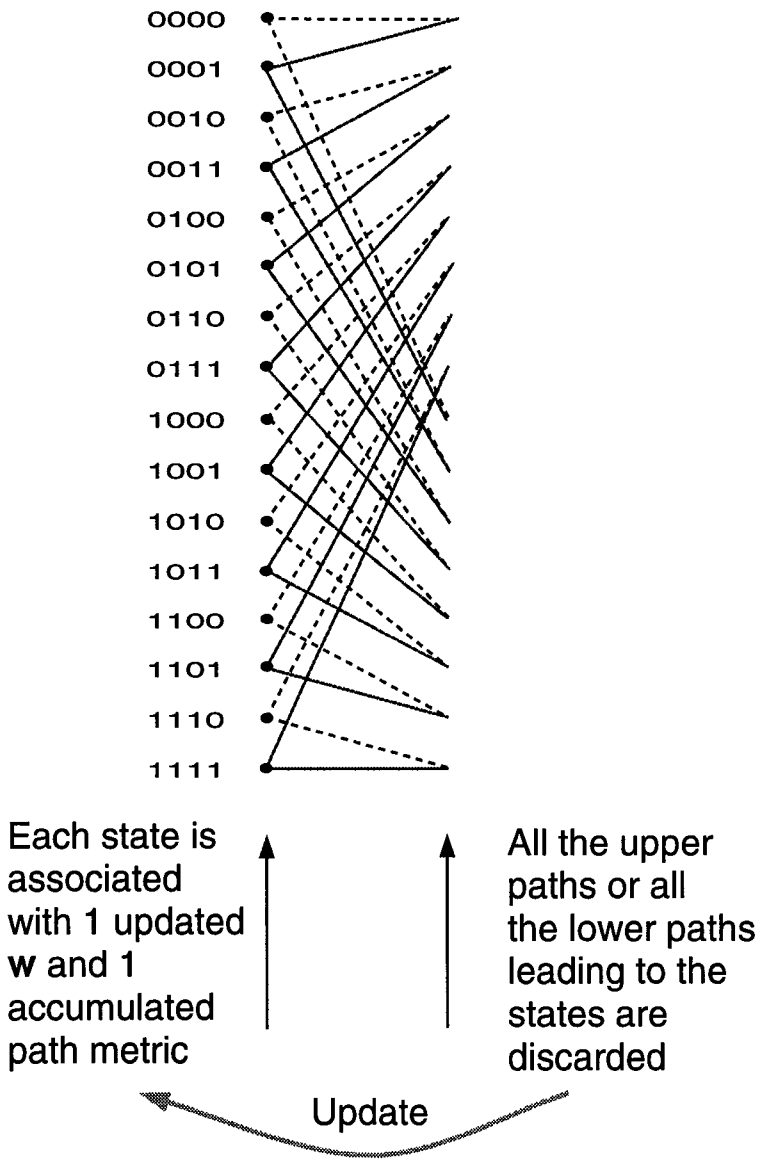


Figure 3.3: D-delay algorithm

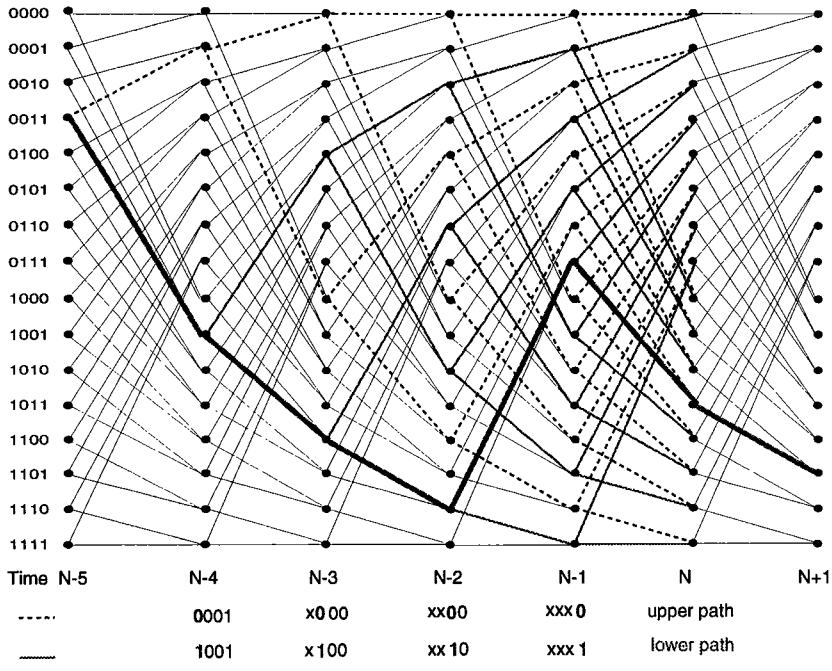


Figure 3.4: 4-symbol delay decoding trellis diagram

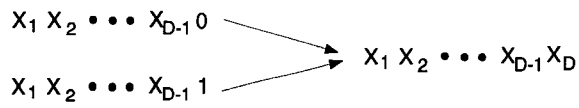


Figure 3.5: States change relationship between consecutive stages

is demonstrated more clearly in Fig. 3.5. From this point on, we will refer to the path from state $X_2 \cdots X_{D-1}0$ to $X_1X_2 \cdots X_D$ as an upper path and the path from the state $X_2 \cdots X_{D-1}1$ to $X_1X_2 \cdots X_{D-1}X_D$ as a lower path. State $X_2 \cdots X_{D-1}0$ is always above state $X_2 \cdots X_{D-1}1$. Therefore, the upper path that merges to $X_1X_2 \cdots X_D$ must be from $X_2 \cdots X_{D-1}0$ and the lower path must be from $X_2 \cdots X_{D-1}1$. According to Property 2, $X_2 \cdots X_{D-1}0$ is from a "0" input bit D stages back and $X_2 \cdots X_{D-1}1$ is from a "1" input bit D stages back. This leads to property 3.

- **Property 3:** Every upper path of a pair that merges to each state corresponds to a "0" input bit $D+1$ stages back. Every lower path of a pair that merges to each state corresponds to a "1" input bit $D+1$ stages back.

Based on these properties, we developed the following D-symbol delay algorithm.

- At each stage, calculate the next accumulated path metrics of all the paths that are generated from all the current states. Compare their accumulated path metrics. If the path with the smallest accumulated path metric is an upper path, then every upper path at each state is kept and every lower path is discarded. The information bit D stages back is decided to be 0.

Otherwise, all the lower paths are kept and all the upper path are discarded, the information bit D stages back is decided to be 1.

- At the current stage, a set of array weights $\mathbf{w}_s(n)$ at each state s, $s = 1, \cdots, 2^D$, are updated by using $\mathbf{w}_{pre-s}(n-1)$, which is a set of array weights obtained at the state prior to "s" along the surviving branch, and by using array data $\mathbf{x}(n)$ and the modulated output signal of the path from state "$pre - s$" to "s".

  In Fig. 3.4 at stage $N-1$, the path from state 0111 to state 1011 has the smallest accumulated path metric. All the lower paths, i.e. all the solid line paths, are saved as the surviving paths. The information bit 4 stages back is 1 which is the input bit that leads to the path from state 0011 at time $N-5$ to state 1001 at time $N-4$. We then update the array weights along the surviving paths. For example, $\mathbf{w}_{1011}(n)$ is updated based on $\mathbf{w}_{0111}(n-1)$, the QPSK modulated signal $(\frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2})$ of path output 00 and received array data. Finally, we update the information at the current states with new updated array weights and accumulated path metrics of the surviving paths. The whole updating and symbol decision process is completed by only using the information at current time slot as shown in Fig. 3.3.

- At the end of each time slot, the last D input bits are decided based upon the final state of the selected path. The last ith bit is equal to the first ith bit of the selected state. In Fig. 3.4 the last state is 1101, the last four input bits are 1011.

3.2.3 Performance Analysis

In this section, we will prove that the reliability of bit decision increases with an increase of D in the D-symbol delay algorithm when there is no strong cochannel interference. But when there is strong cochannel interference, D has to be appropriately chosen to achieve an optimal BER.

In general, the branch metrics measured by Eq.(3.1) can be divided into three categories:

- Case 1: $\mathbf{w}$ is updated using n desired symbols, and $s_i(n)$ is a desired symbol. The expected value of $b_i(n)$ is the mean square error of the estimated array output $\mathbf{w}_i^H(n-1)\mathbf{x}$. We denote it as mse. $\frac{b_i(n)}{mse} \sim \chi_1^2$. The mse is determined by the adaptive algorithm used, such as RLS or LMS and also determined by the channel conditions such as SNR, fading rate and cochannel interference to signal ratio.

- Case 2: $\mathbf{w}$ is updated using n desired symbols, $s_i(n)$ is an undesired symbol, $\frac{b_i(n)}{mse}$ is from noncentral χ_1^2 with 1 degree of freedom and noncentral parameter d_1, where d_1 is the Euclidean distance between two output signals from two branches that stem from the same node.

- Case 3: $\mathbf{w}$ is updated using $n-m$ desired symbols and m undesired symbols. $s_i(n)$ is an undesired symbol, The expected value of $b_i(n)$ is large in this case and is denoted by MSE. If there is no strong interference in the received array signals, along a wrong path the path outputs are randomly connected with respect to the received array data, and the updated array weights are getting more and more divergent. As a result, we have

$$MSE_D \geq MSE_{D-1} \geq MSE_{D-2} \geq \cdots \geq MSE_1 > mse. \qquad (3.2)$$

Accordingly, we evaluated the path metrics $P_i(n), i = 1, \cdots 2^{D+1}$ by $P_i(n) = \sum_{n=1}^{D+1} b_i(n), i = 1, \cdots 2^{D+1}$, there are three classes of path metrics:

- Class 1: P_c is a path metric of a correct path.
- Class 2: The first m branches belong to Case 1, the $(m+1)$th branch belongs to case 2, the last $D-m$ branches belong to Case 3, $m = 1, \cdots, D$
- Class 3: The first branch metric belongs to Case 2, the rest branches belong to Case 3.

In a D-delay decoder, we make a choice between two subsets of equal size based on the smallest P_i at each stage. Note that all the paths in the undesired subset belong to Class 3, and the paths in the desired subset belong to either Class 1 or Class 2. Obviously, the average difference between a P_u in the undesired subset and the P_d corresponding to the correct path will decrease with an increase of D. The average difference between a P_u and a P_i of a false path in the desired subset is

$$E(P_u - P_i) = \sum_{i=D-m+1}^{D} MSE_i - m \cdot mse \quad m = 1, 2, \cdots, D, \tag{3.3}$$

The overall average difference E_D between a path in Class 3 and a path in Class 2 for a D-symbol fixed-delay decoder is given by

$$\begin{aligned} E_D &= \overline{E(P_u - P_i)} && (3.4)\\ &= \frac{1}{2^D - 1} \sum_{m=1}^{D} 2^{D-m} \Big(\sum_{i=D-m+1}^{D} MSE_i - m \cdot mse \Big) && (3.5)\\ &= \frac{1}{2^D - 1} \Big[\sum_{m=1}^{D-1} 2^{D-m} \Big(\sum_{i=D-m+1}^{D} MSE_i - m \cdot mse \Big) + \Big(\sum_{i=1}^{D} MSE_i - D \cdot mse \Big) \Big] && (3.6)\\ &= \frac{1}{2^D - 1} \Big[2 \cdot \sum_{m=1}^{D-1} 2^{D-1-m} \Big(\sum_{i=D-m+1}^{D} MSE_i - m \cdot mse \Big) + \Big(\sum_{i=1}^{D} MSE_i - D \cdot mse \Big) \Big] && (3.7) \end{aligned}$$

$$\geq \frac{1}{2^D-1}[2\cdot\sum_{m=1}^{D-1}2^{D-1-m}(\sum_{i=D-1-m+1}^{D-1}MSE_i - m\cdot mse) + (\sum_{i=1}^{D}MSE_i - D\cdot mse)] \quad (3.8)$$

$$= \frac{1}{2^D-1}[2\cdot(2^{D-1}-1)E_{D-1} + (\sum_{i=1}^{D}MSE_i - D\cdot mse)] \quad (3.9)$$

$$= E_{D-1} + [(\sum_{i=1}^{D}MSE_i - D\cdot mse) - E_{D-1}] \quad (3.10)$$

$$> E_{D-1} \quad (3.11)$$

The bigger of the decision delay, the larger the average Euclidean distance between the correct path and a false path in the other subset and also the larger the average Euclidean distance between paths in the two different subsets. Therefore, the BER decrease with the increase of the decoding decision delay. However, if the output symbols of one of the 2^D surviving paths happen to be the same as the transmitted symbols from the interference, the further increase of D will cause the weights to converge along the interference path and will result in an increase of BER. The relationship in Eq.(3.2) no longer holds. MSE_i increases initially and then decreases. When D is large enough, MSE_i finally converges to a mean square error between (i) the combined array outputs using the weights that are trained by the interference signals and (ii) the interference signals. We denote this mean square error by mse_I. When the Ith interference is stronger than the desired signal, we have $mse_I < mse$. Thus P_I may become smaller than P_d when D is large enough. Our simulation results also showed that the increase of D will decrease the BER when the interference signals are not as strong as the desired signal. When an interference signal is stronger than the desired signal, BER decreased as we increase the decision delay from 1 to 4 symbols. The improvement becomes smaller and smaller. As we further increase the decision delay, we observe a slight increase of BER. Therefore, a proper delay length needs to be decided based on

I/S ratio and RLS updating window size to minimize BER.

3.2.4 M-D algorithm

The use of the D-symbol delay algorithm reduces the error propagation in the conventional decision directed array weights tracking algorithm. However, the complexity would increase drastically if we use a D-symbol delay algorithm, because now we have to update 2^D weighting vector instead of 1 weighting vector.

To reduce complexity, we exploit the fact that the correct path should have much smaller accumulated path metric than most of the other selected paths. So, we should be able to discard a majority of the selected paths and keep those most likely ones without compromising the performance. This results in the following M-D algorithm.

Based on the three properties we discussed in the previous section, we find that if all the binary representation of the surviving states have a common last bit, they must be from a unique state D stages back, and the input bit to that state is equal to this last bit. Otherwise, they are from different states D stages back. From this finding and the D-delay algorithm presented in previous sections, we developed the following M-D algorithm.

- At each stage, calculate the next branch metrics and accumulated path metrics of all the paths that are generated from the surviving states.

- When the binary form of the surviving states have a common last bit, either "1" or "0", the input bit D stages back is decided to be equal to this common last bit. Select M paths that have the smallest accumulated path metrics from all the surviving paths from the surviving states.

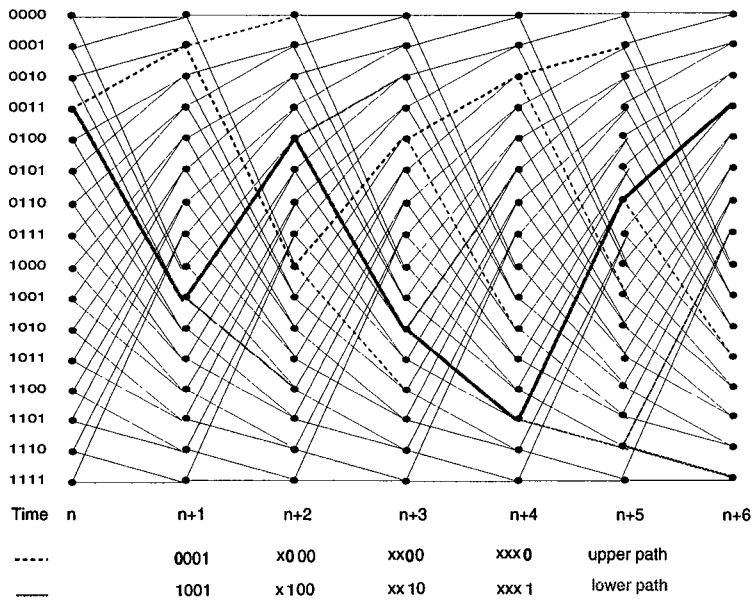


Figure 3.6: An example of M-D decoding trellis diagram, $M = 2, D = 4$

- When the surviving states have different last bits, compare all the path metrics from these states. If the smallest path metric is from an upper path, keep all the upper paths and discard all the lower paths. The input bit D stages back is decided to be "0". Vice versa. Select M paths that have the smallest accumulated path metrics among the surviving paths.

- A set of array weights $\mathbf{w}_s(n)$ at each surviving state is updated by using $\mathbf{w}_{pre-s}(n-1)$ obtained at the state prior to 's' along the surviving branch, and by using array data $x(n)$ and the modulated output signal of the path from state "$pre - s$" to "s".

- At the end of each time slot, the last D input bits are decided based on the final state of the selected path. The last ith bit is equal to the first ith bit of the final selected state.

In Fig. 3.6, the two surviving states 0010 and 1101 at stage $n+4$ have different last bits. They are from two different input bits back by D stages. Using the weights obtained at these two surviving states, and the received array signal, we calculate the branch metrics of the branches generated from these two states using its own path output. We update accumulated path metrics of the four corresponding paths. The path that enters state 0110 at stage $n+5$ has the smallest path metric. It is a lower path. The input bit 4 stages back is decided to be "1" which is the bit that generates the path from state 0011 at stage n to the state 1001 at stage $n+1$. We then discard all the upper paths at stages $n+5$ and select 2 paths from all the lower paths. In this case, both lower paths are selected and kept. We then update the weights of these two surviving paths using its own path output and save the updated weights separately at these two surviving states. At the next stage $n+5$, we compare the last bit of the two surviving states 0110 and 1110. They have a common last bit "0". The input bit 4 stages back is decided to be "0" which is the bit that generates the path from 1001 at stage $n+1$ to the state 0100 at stage $n+2$. We then continue to update the accumulated path metrics for the paths generated from these two surviving states, find the one with the smallest path metric, choose the next two surviving paths, and update the next two weights.

The use of M-D algorithm reduces the number of surviving paths from 2^D to M. Our simulation results showed that an M as small as two results in only a slightly degraded performance while $D = 5$. Although, compared to the decision directed weights tracking algorithm, M-D algorithm still increases the number of updating weighting sets from 1 to M, the complexity does not increase by M times. We will demonstrate this relation in the following sections.

3.2.5 Diversity Weights Tracking

To achieve MMSE optimum combining on time varying fading channels, recursive algorithms such as LMS or RLS can be applied. RLS has been known to have a fast convergence rate and good tracking capability compared to the low complexity LMS algorithm [60] or blind adaptive algorithms such as CMA. Among different RLS algorithms, QRD-RLS has much better numerical stability. Moreover, it is also computationally more efficient. We can obtain the branch metrics required in M-D algorithm directly using the modified QRD-RLS algorithm without calculating the $\mathbf{w}$ explicitly. The QRD-RLS algorithm is thus adopted in our system.

The weights in QRD-RLS algorithm satisfy the following equation

$$\mathbf{R}(n)\mathbf{w}(n) = \mathbf{p}(n) \tag{3.12}$$

which is another form of solution to minimizing Eq.2.9. $\mathbf{R}$ is an upper triangular matrix obtained in the following equation, in which the weighted data matrix $\mathbf{\Lambda}^{1/2}(n)\mathbf{A}(n)$ is trianglarized through a unitary matrix $\mathbf{Q}(n)$.

$$\mathbf{Q}(n)\mathbf{\Lambda}^{1/2}(n)\mathbf{A}(n) = \begin{bmatrix} \mathbf{R}(n) \\ \mathbf{0} \end{bmatrix} \tag{3.13}$$

where $\mathbf{0}$ is a null matrix. $\mathbf{p}(n)$ is a vector defined by

$$\mathbf{p}(n) = \mathbf{F}(n)\mathbf{\Lambda}^{1/2}(n)\mathbf{d}(n) \tag{3.14}$$

and $\mathbf{F}(n)$ consists of the first M rows of $\mathbf{Q}(n)$.

To use the QRD-RLS algorithm for parallel residual error (square root of branch metric) calculation, M sets of array weights $\mathbf{w}_i(n), i = 1, \cdots M$, which are associated with the M surviving paths, must satisfy the following equation:

$$\mathbf{R}(n)[\mathbf{w}_1(n), \mathbf{w}_2(n), \cdots, \mathbf{w}_M(n)] = [\mathbf{p}_1(n), \mathbf{p}_2(n), \cdots, \mathbf{p}_M(n)] \tag{3.15}$$

To solve the least square problem recursively, we use a sequence of Givens rotations to annihilate all M elements in the new incoming data $\mathbf{x}(n)$ one by one. This procedure is shown as follows:

$$\begin{bmatrix} \mathbf{R}(n) \\ \mathbf{0} \end{bmatrix} = \mathbf{T}(n) \begin{bmatrix} \lambda^{1/2}\mathbf{R}(n-1) \\ \mathbf{0} \\ \mathbf{x}^H(n) \end{bmatrix} \tag{3.16}$$

where $\mathbf{T}(n)$ is the unitary matrix denoting the combined effect of a sequence of Givens rotations:

$$\mathbf{T}(n) = \mathbf{J}_M(n), \cdots, \mathbf{J}_2(n)\mathbf{J}_1(n) \tag{3.17}$$

$\mathbf{p}_k(n)$ is updated based upon the output $d_k^H(n)$ of the kth surviving branch at time n and the vector $\mathbf{p}_k(n-1)$ generated at the departing state of the branch.

$$\begin{bmatrix} \mathbf{p}_k(n) \\ \mathbf{v}_k(n) \end{bmatrix} = \mathbf{T}(n) \begin{bmatrix} \lambda^{1/2}\mathbf{p}_k(n-1) \\ \lambda^{1/2}\mathbf{v}_k(n-1) \\ \mathbf{d}_k^H(n) \end{bmatrix}. \tag{3.18}$$

Without calculating $\mathbf{w}$ explicitly, the residual error (or the branch metric) of the kth path can be obtained through $T(n)$ and the last element of $\mathbf{v}_k(n)$ [20].

3.2.6 Computational Complexity

In the QRD-RLS updating algorithm, the complexity of updating R is $O(K^2)$, The complexity of updating p is $O(K)$. The total complexity of calculating $2 \times M$ branch metrics is $O(K^2 + 2MK)$. While in the decision directed algorithm, in addition to updating R and p, we need to perform back substitution to calculate $\mathbf{w}$ in order to get estimated array output. The complexity of back substitution is $O(K^2)$. Overall, compared to decision directed algorithm, the complexity of updating M paths is only moderately increased when M is less than K.

If we keep M surviving paths, we need to calculate $2M$ branch metrics and compare $2M$ accumulated path metrics.

The cost of getting M surviving paths in this M-D decoder is evaluated in the following. One can find the best path out of $2M$ paths with $2M-1$ comparisons [62]. Let $\bar{C}_t(n)$ denote the average number of comparisons required to find the tth largest of n elements. Then an algorithm exists [62] for which

$$\bar{C}_t(n) = n + t + f(n) \quad \text{where} \lim_{n\to\infty} f(n)/n = 0. \tag{3.19}$$

To find the Mth best path out of $2M$ paths, we need $2M+M+f(2M)$ comparisons. Once the metric of the Mth path is known, we can choose the rest $M-1$ best paths with at most $M-1$ comparisons [63]. In total, the cost of finding the M best paths out of $2M$ paths is

$$4M - 1 + f(2M) \quad \text{comparisons/branch released} \tag{3.20}$$

The complexity is $O(M)$. The memory required is also $O(M)$. Thus, this part adds a small amount of complexity when M is less than K.

3.2.7 Proposed Adaptive Diversity Combining System

A block diagram shown in Fig. 3.7 summarizes our proposed adaptive diversity combining system. Each source is encoded with a binary convolutional code. This system combines our proposed M-D decoder and QRD-RLS algorithm for surviving paths selection and symbol decision. At each symbol interval, M surviving paths are selected. Among them, only one is left after D symbol intervals. Each transmitted symbol is automatically decided after a D symbol delay. This system reduces error propagation, with moderate increase in computational complexity.

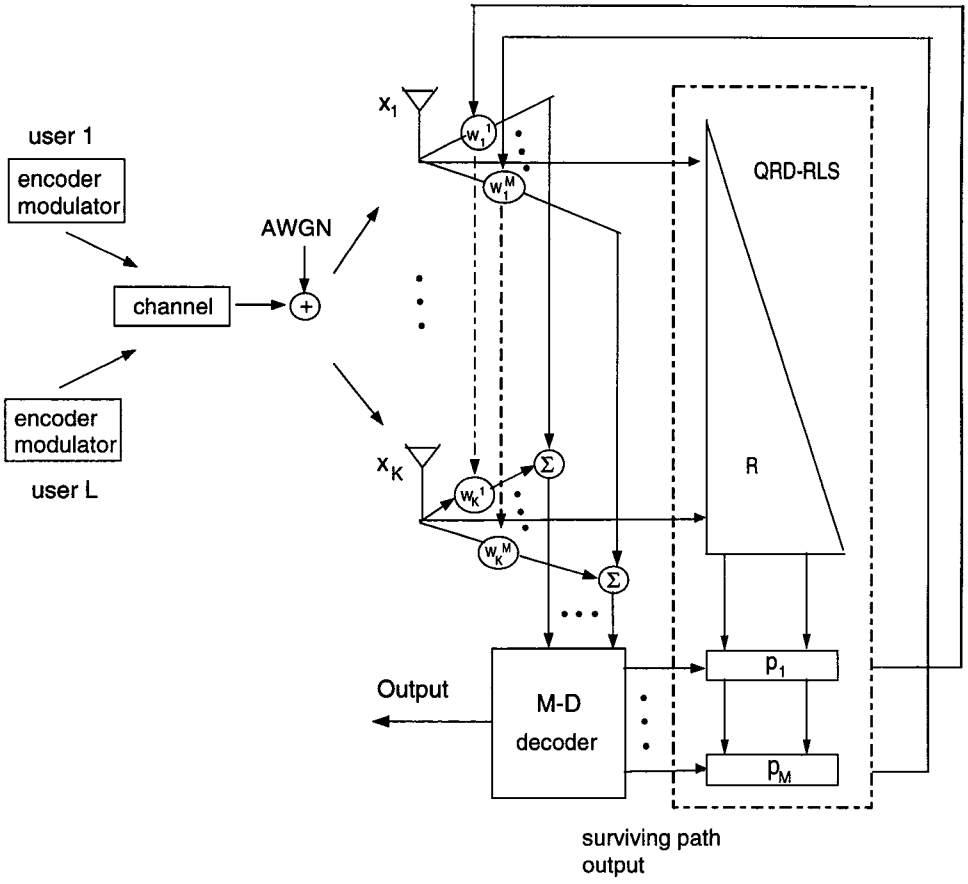


Figure 3.7: The proposed adaptive diversity combining system

3.3 M-D decoding of a Trellis coded 8-PSK code

In some situations such as wireless video transmission, we need to transmit high speed data through some limited frequency bandwidth. A use of 8-PSK signal constellation in conjunction with trellis codes can double the transmitted information bit rate compared to a binary 1/2 convolutional coded 4-PSK signal used in the previous examples. Therefore, we modified the D and M-D decoding algorithm of convolutional code and applied it to TCM [61].

In our example of TCM, we partition the eight-phase signal constellation shown in Fig. 3.8 into subsets of increasing minimum Euclidean distance. We use the rate of 1/2 convolutional code to encode one information bit while the second information bit is left uncoded. The encoder is shown in Fig 3.8. The coded bits are used to select one of the four subsets that contain two signal points each, while the uncoded bit is used to select one of the two signal points within each subset. The Euclidean distance between parallel paths is $2\sqrt{\epsilon}$, where ϵ is the energy of the signal.

The decoding algorithm is given as follows:

- Select the branch having the smaller path metric among the parallel branches. If an upper path is selected, the uncoded bit at current time is decided to be "0", otherwise, it is decided to be "1".

- The coded bit is decoded using the D or M-D algorithm presented in the previous sections.

Fig. 3.9 shows a 2-symbol delay 8-PSK TCM decoder. Each time, one branch is selected between each pair of parallel branches in favor of the one having a smaller branch metric. Then we choose surviving paths among these selected paths

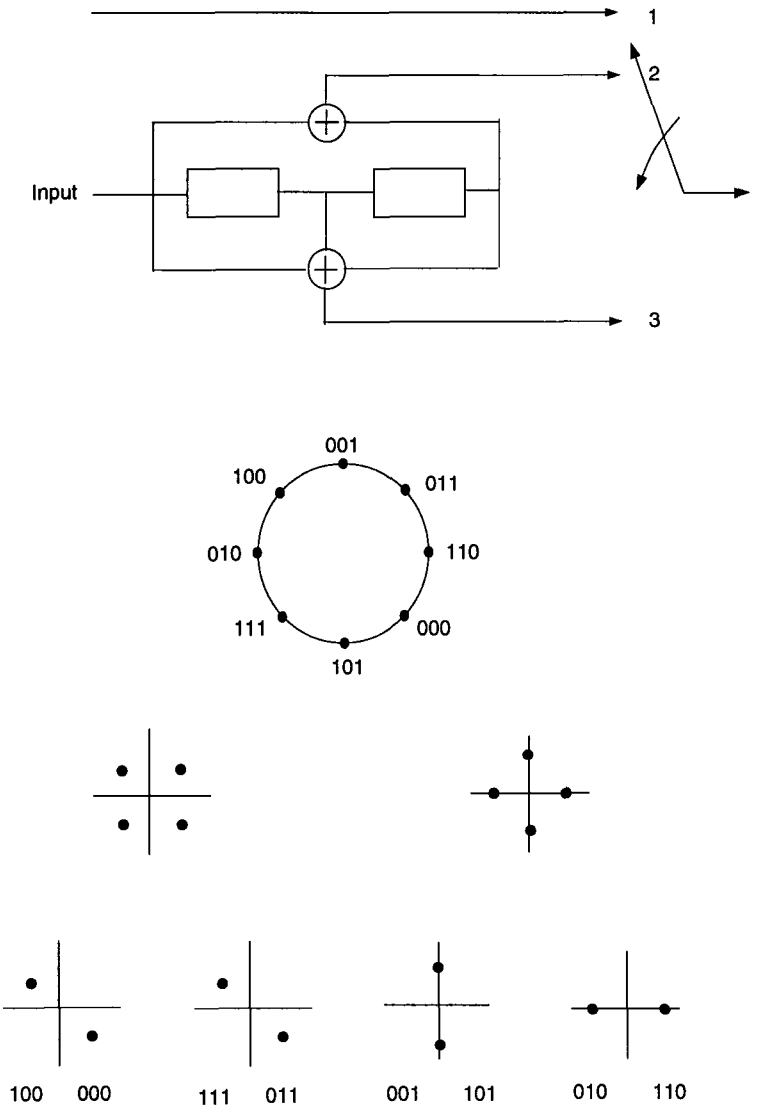


Figure 3.8: Set partitioning of an 8-PSK signal set

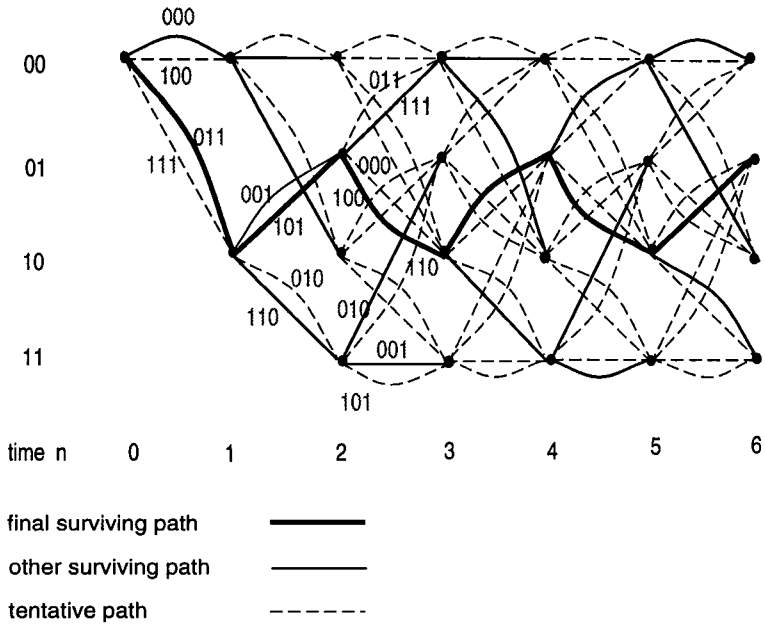


Figure 3.9: 2-symbol delay 8-PSK TCM decoding trellis diagram

following the same procedure that is used in the convolutional decoder having a 2-symbol delay. At time 3, one path is selected from each pair of branches that are generated from surviving states 01 or 11 at stage 2. Within these four selected paths, the path that enters state 10 at stage 3 has the smallest accumulated path metric. All the lower paths in the surviving paths are saved, i.e., the paths enter state 00 and state 10 at stage 3.

3.4 Simulation Results

In our simulations, four antennas are used. The channel is time-division-multiplexed. There are 162 symbols in each time slot. The first 14 symbols are from the training sequence. The carrier frequency is 900 MHz. The modulated data rate is

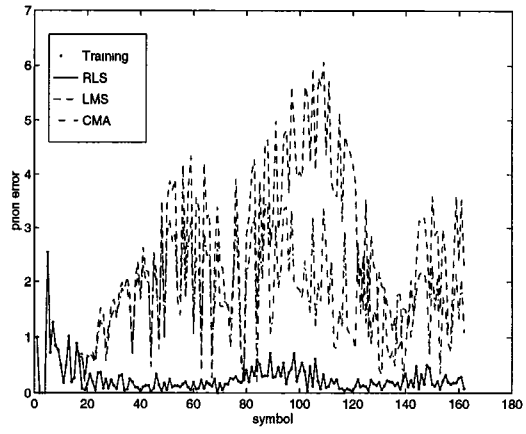


Figure 3.10: Learning curve of LMS, CMA and RLS

24.3ksym/s, which is the same as in IS-136 standard. The SNR is 15dB.

It is well know that the RLS algorithm has faster initial convergence rate and better tracking ability than the LMS algorithm, and the reference dependent RLS and LMS algorithms have faster convergence rate and better tracking ability than the blind adaptive algorithms such as CMA. [20]. With 14 training symbols (in IS-54), it is not possible for weights to converge to optimal values using either LMS or CMA [60]. RLS algorithm is adopted in our system for weights initialization. RLS algorithm is however computationally more expensive than the other algorithms. In most slowly time-varying channels, people still prefer LMS. To see the feasibility of these algorithms on the tracking of fast time-varying channels, we did the following simulations.

In Fig. 3.10, RLS is used in the training mode. RLS, LMS and CMA are used individually in the tracking mode. The SNR=20dB, INR=10dB. The weights updating of the RLS algorithm is based upon the algorithm given in section IV. The weights updating of the LMS and CMA algorithm is based upon the following

equation:

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \mu \mathbf{x}(n) e^*(n) \tag{3.21}$$

where μ is the stepsize, $e(n)$ is estimation error. For LMS, $e(n)$ is given by

$$e(n) = d(n) - \hat{\mathbf{w}}^{\mathbf{H}}(n)\mathbf{x}(n) \tag{3.22}$$

where $d(n)$ is a reference signal. For CMA, $e(n)$ is given by

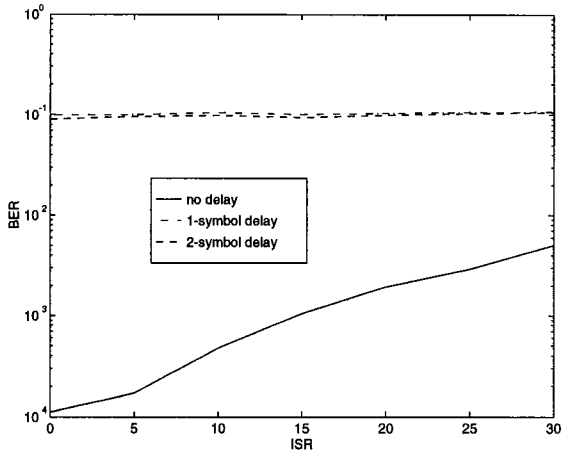
$$e(n) = \hat{\mathbf{w}}^{\mathbf{H}}(n)\mathbf{x}(n)(|\hat{\mathbf{w}}^{\mathbf{H}}(n)\mathbf{x}(n)|^2 - 1)^2 \tag{3.23}$$

Notice that there are other kinds of blind equalizing algorithms such as Sato's algorithm. Here, the most widely used CMA algorithm is used in our demo.

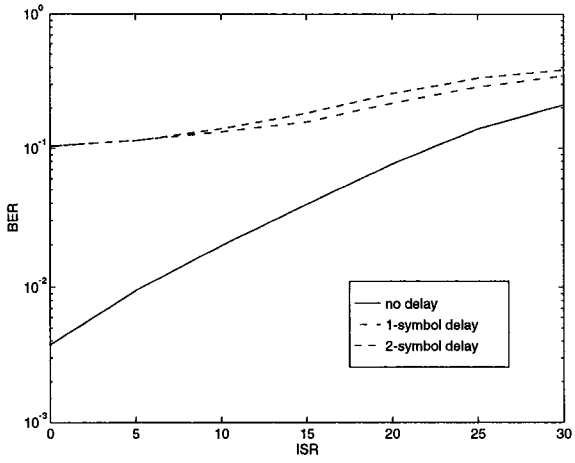
We observe a severe performance loss in fast fading channel tracking by using LMS or CMA. Different stepsizes have been tried. Similar or even worse results have been observed when LMS and CMA are used. Therefore the RLS algorithms is adopted in our system for both weights acquisition and tracking on fast time-varying fading channels.

A frequently used approach in a slowly fading channel equalization with convolutional or Trellis coded signal is to use a delayed temporal decision from a viterbi decoder to update the equalizer weights. However, when adopting such techniques to our system, we found a loss in channel tracking and poor BER performance. Especially, for the fast time-varying fading channel as shown in Fig. 3.11(b), a larger decision delay results in a worse performance.

Computer simulation results provided in Fig. 3.12 give a quantitative examination of the BER improvement from using D-symbol delay algorithm and the M-D algorithm, respectively. Three sources are all encoded with the convolutional encoder shown in Fig 3.1. The desired vehicle is moving at 60 miles/hr. The other two are moving at 30miles/hr and have the same interference to signal ratio.

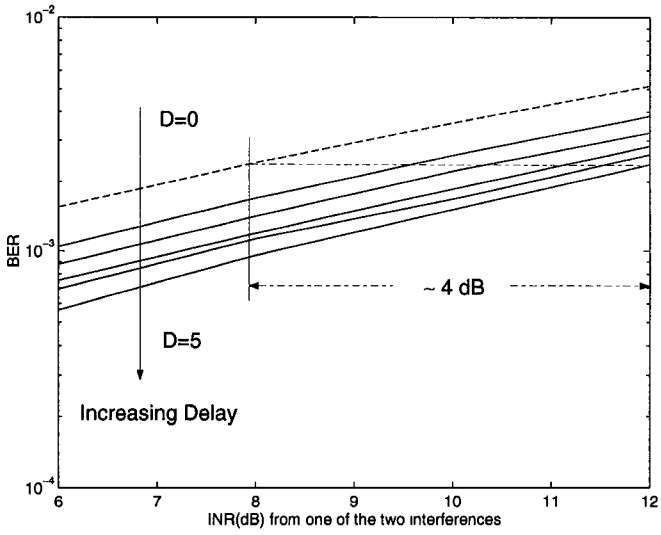


(a) SNR=20dB crf=900MHz symbol rate=24.3 Ksymbol/s

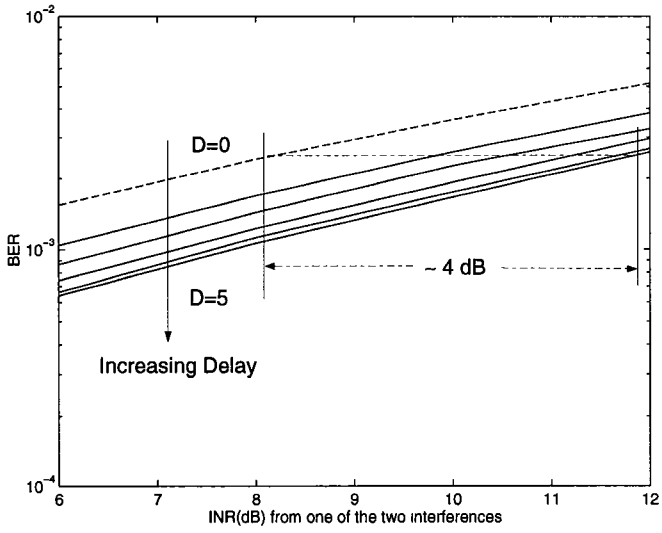


(b) SNR=20dB crf=2GHz symbol rate=16 Ksymbol/s

Figure 3.11: Delayed decision-directed weights tracking for coded QPSK

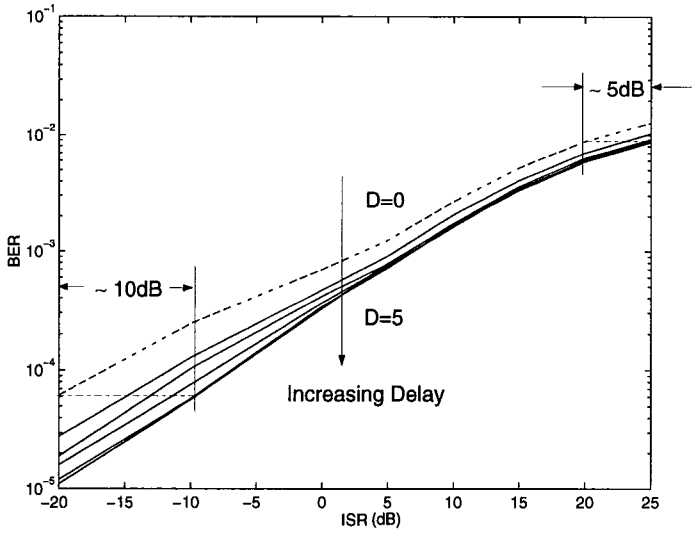


(a) D-delay

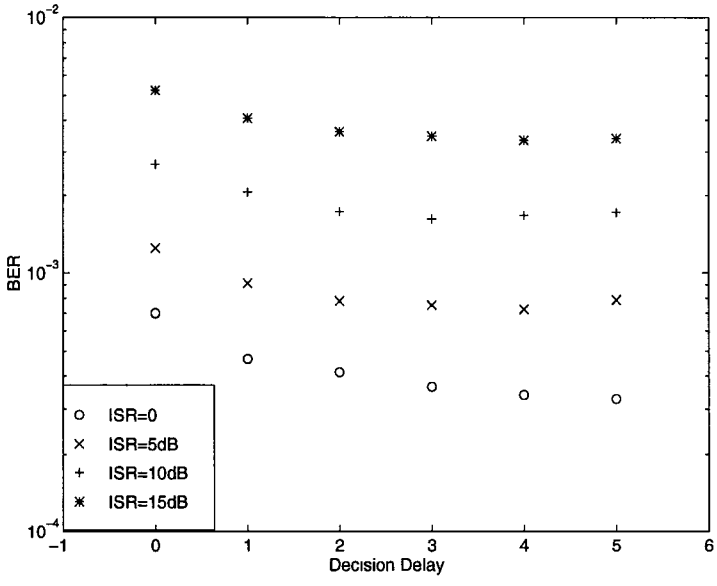


(b) M-D

Figure 3.12: Influence of D on the performance of D-delay and M-D decoding algorithms



(a) Influence of D at low ISR and high ISR



(b) Influence of D at high ISR

Figure 3.13: More significant improvement at low ISR (M-D algorithm, M=2)

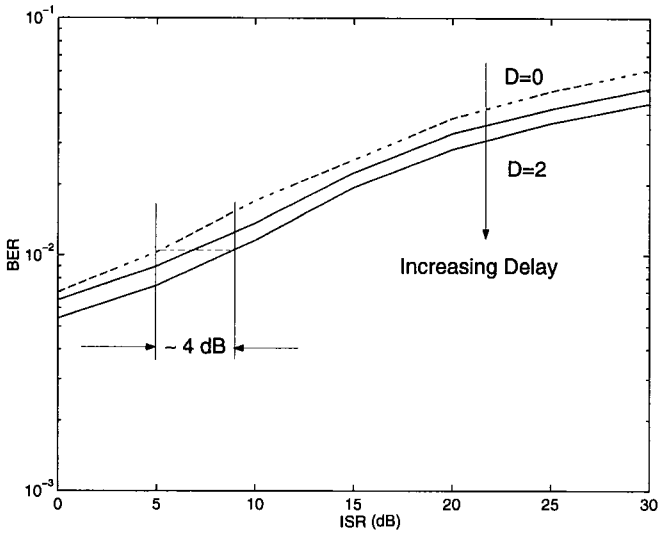


Figure 3.14: BER performance of D-symbol delay decoding of a TCM code

In Fig. 3.12, we observe improved BER performance as we increase D from 0 to 5: about 4dB improvement in one interference suppression and a total of 8dB improvement in both cases (a) and (b). With M-D algorithm, the improvement is slightly less compared to D algorithm, but the complexity of M-D algorithm is greatly reduced.

In Fig. 3.13, the M-D algorithm is used and M is set to 2. In Fig. 3.13(a), at low ISR, BER performance is gradually improved as D is increased. A total of 10dB improvement with $D = 5$ over $D = 0$. At high ISR, BER performance is improved as D is increased from 1 to 4 and is slightly decreased as D is further increased. This is because when the interference is stronger than a desired signal, weights may converge along a wrong surviving path. So D should be appropriately chosen at high ISR to achieve the optimal BER.

An extension of D-delay and M-D algorithms to the TCM code is demonstrated

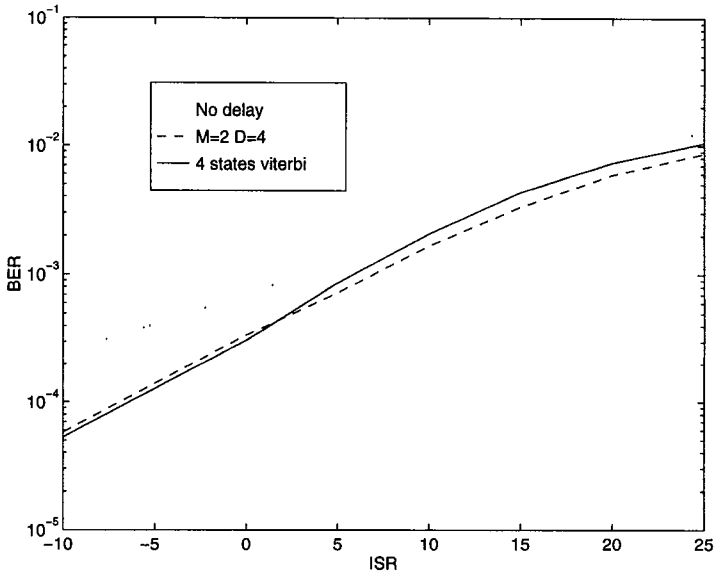


Figure 3.15: Comparison between M-D algorithm and Viterbi algorithm

in the following example. A TCM encoder shown in Fig. 3.8 is used. In Fig. 3.14, the dotted line represents the BER performance under decision directed QRD-RLS for weights tracking. Coherent demodulation is made on an uncoded QPSK signal sequence. The solid line represent the BER under 1-symbol and 2-symbol delay algorithms. Using 2-symbol delay algorithm, 3 to 5 dB improvement in the interference suppression over the conventional decision directed algorithm is observed.

The BER performance achieved by using M-D algorithm with $M = 2, D = 4$ are compared to that achieved by using Viterbi algorithm [59] with 4 states. Infinite memory length is used in the Viterbi algorithm. Fig. 3.15 shows that more improvement in BER is achieved by using M-D algorithm at high ISR. The complexity of M-D($M = 2$) algorithm is also lower. This is because the length of two parallel surviving paths in the Viterbi decoding algorithm is not fixed and can

not be controled. Therefore, using Viterbi decoding algorithm, we can not avoid the converging of weights along a wrong path at high ISR.

Chapter 4

Two-Dimensional Spatial Smoothing for Multipath Coherent Signal Identification and Separation

4.1 Introduction

In Chapter 2 and 3, adaptive diversity combining based on reference signals is used to suppress interference and to reduce the effects of multipath fading of desired signals. We find that the lack of accurate tracking of the fast changing mobile channel and the error propagation in the decision directed tracking mode degrade the BER performance. Combined diversity weights tracking and decoding can greatly reduce the error propagation, but still can not eliminate error propagation. The diversity optimum combining technique is applicable to both mobile handset and base station, located in a metropolitan area as well as rural and suburban areas. In rural or suburban areas, or in the metropolitan areas with high rise antennas, we have the option of using constrained adaptive array based on DOA's information at base station. This technique does not require reference signals.

Error propagation is avoided.

The basic idea of this technique is to constrain the response of array beamformer so that signals from the direction of interest are passed with specified gain. The weights are chosen to minimize array output power under these constraints. Accurate DOA estimation and effective adaptive beamforming are the two key steps. The existing methods for high resolution DOA estimation and optimum adaptive beamforming include the well known MUSIC [25] algorithm and ESPRIT [26] algorithm for DOA estimation and MVDR and LCMV algorithms [27], [28], [29] for beamforming. However, an important drawback of these techniques is the severe degradation of the estimation accuracy in DOA estimation [30] or signal cancellation [31] in adaptive beamforming, in the presence of highly correlated or coherent signals.

To counter the deleterious effects due to some coherent signals, a pre-processing scheme referred to as spatial smoothing (SS) proposed by Evans et al. [32] and further developed by Shan et al. [30], [33] has been shown to be effective in decorrelating coherent signals. However, such a scheme is only applied to uniformly spaced linear arrays. Linear arrays are known to be limited to estimating azimuth angles within 180^0, and practically effective only for signals from the broadside direction. The degree of SS using a uniformly spaced linear array is also sensitive to DOA's [34]. As a result, a linear array is not very effective in radar, sonar, or especially in cellular communications where users can never predict the incoming directions of the moving targets.

In the past decade, research has been carried out in developing algorithms for coherent interference using arrays of arbitrary geometry. In the area of DOA estimation, multidimensional subspace fitting algorithms such as deterministic max-

imum likelihood (DML) [14], multidimensional (MD)-MUSIC [15], and recently proposed weighted subspace fitting (WSF) [16], [17], are effective in both coherent and noncoherent environment and can be applied to arrays of arbitrary geometry. However, all these algorithms involve some searching procedures used to solve nonlinear equations. They are computationally intensive and are not practical in real-time applications. Direction finding using spatial smoothing with interpolated arrays technique [13] [18] maps the signal received by the array to a virtual array, but these methods all need approximations and have restricted applications. In the area of narrow-band adaptive beamforming, the coherent interference suppression using null constraint with an array of arbitrary geometry was addressed in [35]. This approach still requires pre-estimation of arrival angles of coherent interferences. The SPT-LCMV beamforming algorithm applicable to arrays of arbitrary geometry was considered in [36]. This algorithm requires increased computational complexity compared to LCMV. Recently, diversity combining [9] and blind adaptive beamforming [56] have been proposed to combat multipath fading and cochannel interference. However, both techniques have limitations on tracking fast fading channels with strong cochannel interference.

In this work, we develop a general SS technique for arrays of arbitrary geometry to make MUSIC, ESPRIT and optimum adaptive beamforming algorithms operative in a coherent interference environment and meanwhile achieve robustness in performance. Compared with the aforementioned methods for arrays of arbitrary geometry, this SS technique can be easily implemented. It does not increase the computational complexity of either MUSIC, ESPRIT, or adaptive beamforming. It allows us to work on a data domain [37], and thus enables us to incorporate the recently developed URV [38] [39] algorithm to DOA estimation and updating

and enables us to implement MVDR beamforming algorithm using systolic arrays. Therefore it has great potential in mobile radio communication where coherent and cochannel multipath interference is a major problem. Also, it can be used in conjunction with MUSIC or ESPRIT algorithm to provide an initialization for the WSF method to get a more accurate DOA estimation [17].

Specifically, we discovered and proved the necessary and sufficient conditions on an array geometry for applying SS. They are: (1) such an array must have an orientational invariance structure; (2) its center array has an ambiguity free array manifold; and (3) the number of subarrays is larger than or equal to the largest number of mutually coherent signals. By working on a smoothed data matrix obtained from SS, we can use MUSIC and optimum adaptive beamformers effectively in a coherent interference environment. To further increase efficiency and estimation resolution, we found that the forward/backward spatial smoothing [40] (FBSS), when applied to a nonlinear array of central symmetry, can reduce the number of sensors required and improve the estimation resolution for closely spaced incoming signals. Finally we expand the application of our results to ESPRIT.

In all the papers cited above that dealt with DOA estimation with arrays of arbitrary geometry, ambiguity free array manifolds were assumed. In [25] Schmidt discovered and defined the rank-n ambiguity in an array manifold. In [43], Lo and Marple proved the conditions for a rank-2 ambiguity. In [41] ambiguities of linear arrays were studied. However, constructing a nonlinear array free of up to rank-k ambiguities using only $(k+1)$ sensors remains a challenging problem [42]. In this chapter, we report a more thorough study on this issue. We proved the necessary and sufficient conditions for a three-sensor array manifold to be ambiguity free. We then identified several situations, for higher order sensor array manifolds, in

which ambiguity may arise. Thus we get corresponding necessary conditions to design ambiguity free center arrays and subarrays.

This chapter is divided into five sections. In section 4.2, we introduce MUSIC and SS, and state the limitation of linear array. In section 4.3, we prove the necessary and sufficient conditions on a two dimensional array for applying SS, and consider the FBSS technique for applications in two dimensional array. In section 4.4, we study the cause of ambiguities in a multipath signal environment. In section 4.5 we expand our results to ESPRIT.

4.2 The Problem Statements

In this section, we will briefly describe the array model for DOA estimation and beamforming and explain the principles of various techniques developed under the assumptions of noncoherent interference environment and how it was applied to a coherent interference environment in the specific case of uniformly spaced linear array by using SS. We will then demonstrate the undesired sensitivity of the linear array to the DOA estimation.

Fig. 4.1 shows an equally spaced linear array model for DOA estimation and beamforming. In this example, there are four array elements, each has its own receiver. The array beam pattern is adjusted automatically through the array weights $\mathbf{w}$. Array weights $\mathbf{w}$ are chosen according to DOA estimation information and beamforming algorithms. DOA estimation and beamforming are implemented in a real time DSP processor. Different incoming signals result in different phase delays at each array element. In Fig. 4.1, the phase delay of the signal from angle θ is characterized by steering vector $[1, e^{-j\tau_1(\theta)}, e^{-j\tau_2(\theta)}, ..., e^{-j\tau_p(\theta)}]^T$. $\tau_j(\theta_i)$

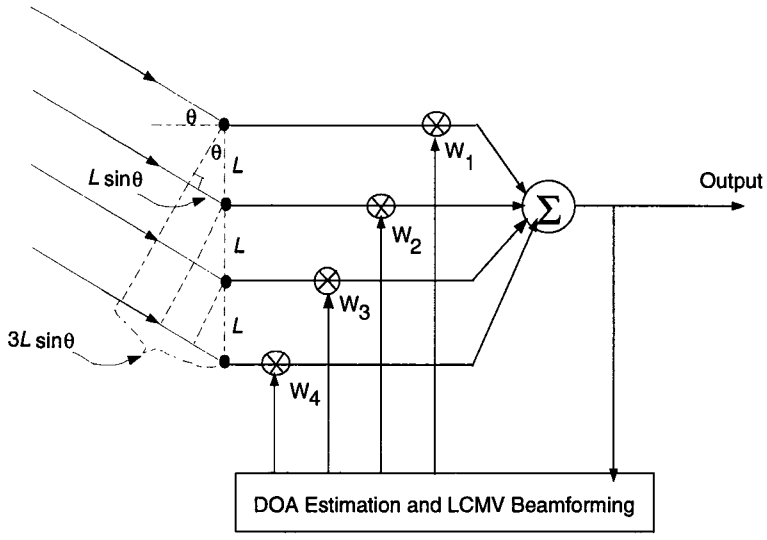


Figure 4.1: Linear array model

is the phase delay of the signal from the first sensor to the jth sensor. $\tau_j(\theta_i) = 2\pi\frac{L}{\lambda}sin(\theta)$. λ is the signal wavelength. High resolution DOA estimation methods (MUSIC and ESPRIT) are based upon the fact that the received array signal vectors span the two disjoint spaces signal plus noise space and noise space. These two spaces are orthogonal and complement to each other. The space spanned by the steering vectors associated with the incoming angles is the same as the signal plus noise space. Incoming signal angles are identified by searching for those steering vectors that are orthogonal to the noise space. After the estimation of DOA's, the array weights are chosen by LCMV or MVDR beamforming algorithm such that the signal energy at the desired directions are passed and that signals at the interference directions are suppressed and that the total array output power is minimized. Therefore, the noise effect is minimized.

The mathematical model is given as follows. Consider an array of p sensors. Let

d narrow-band plane waves $s_1(t), s_2(t), ..., s_d(t)$ impinge on the array at incident angles $\theta_1, ..., \theta_d$. There is also an additive white Gaussian noise vector $\mathbf{n}(t)$, where $\mathbf{n}(t) = [n_1(t), \cdots, n_p(t)]^T$, and $n_i(t), i = 1, \cdots, p$ have zero mean and variance σ^2. The noise received by any sensor is assumed to be uncorrelated with signals and with noise received by any other sensors. The received signals of the array can be expressed as

$$\mathbf{r}(t) = A\mathbf{s}(t) + \mathbf{n}(t) \tag{4.1}$$

where $\mathbf{r}(t) = [r_1(t), ..., r_p(t)]^T$, and $r_i(t)$ is the received signal at the ith sensor, and A is a $p \times d$ matrix, $p > d$,

$$A = [\mathbf{a}(\theta_1), ..., \mathbf{a}(\theta_d)] \tag{4.2}$$

where $\mathbf{a}(\theta_i)$ is the *steering vector* associated with the arrival angle θ_i,

The array output covariance matrix has the form:

$$R = E(\mathbf{r}(t)\mathbf{r}^H(t)) = AR_sA^H + \sigma^2 I, \tag{4.3}$$

where $R_s = E(\mathbf{s}(t)\mathbf{s}^H(t))$. Let $\{\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_p\}$ and $\{\nu_1, \nu_2, ..., \nu_p\}$ denote the eigenvalues and corresponding eigenvectors of R. When the d incoming signals are noncoherent, and the matrix A is of full column rank, the MUSIC algorithm can be used to estimate the angles of the incoming signals as the peaks of the MUSIC estimates

$$\hat{S}_{MUSIC}(\theta) = \frac{1}{\sum_{l=d+1}^{p} |\mathbf{a}(\theta)^H \nu_l|^2}. \tag{4.4}$$

However, when the signals are coherent, R_s is then singular, and the MUSIC algorithm is no longer applicable. In the case of a uniformly spaced linear array, with a sensor spacing Δ, the SS [30] algorithm can be applied to achieve the nonsingularity of the modified covariance matrix of the signals. This technique begins by

dividing a uniform linear array with L sensors into K overlapping subarrays of size p, with sensors $\{1, ..., p\}$ forming the first subarray, sensors $\{2, ..., p+1\}$ forming the second subarray, etc. It was shown that [30]

$$A_k = A_1 E^{(k-1)}, \tag{4.5}$$

where A_i, $i = 1, ..., K$, is a $p \times d$ matrix consisting of steering vectors associated with the ith subarray, $E^{(k)}$ denotes the kth power of the $d \times d$ diagonal matrix E, where $E = diag\{e^{-j\frac{\Delta}{c}\sin(\theta_1)}, ..., e^{-j\frac{\Delta}{c}\sin(\theta_d)}\}$, c is the speed of the incoming signals.

The covariance matrix of the kth subarray is therefore given by

$$R_k = A_1 E^{(k-1)} R_s E^{H(k-1)} A_1^H + \sigma^2 I. \tag{4.6}$$

The spatially smoothed covariance matrix is defined as the average of the subarray covariances:

$$\bar{R} = \frac{1}{K}\sum_{k=1}^{K} R_k = A_1 \bar{R}_s A_1^H + \sigma^2 I, \tag{4.7}$$

where R_k is the covariance matrix associated with the kth subarray, $\bar{R}_s$ is the modified covariance matrix of the signals, and has been proved [30] to be full rank when $K \geq d$. The signals are thus progressively decorrelated [34]. However, linear arrays have limitations in the domain of estimable DOA's. It has been shown in [44] that $\bar{R}_s$ can be decomposed as follows:

$$\bar{R}_s = \frac{1}{K} C C^H \tag{4.8}$$

where $C = PA^T$ with $P = diag(p_1, p_2, \cdots, p_d)$, and

$$A = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ e^{-j2\pi\frac{\Delta}{\lambda}\sin(\theta_1)} & e^{-j2\pi\frac{\Delta}{\lambda}\sin(\theta_2)} & \cdots & e^{-j2\pi\frac{\Delta}{\lambda}\sin(\theta_d)} \\ e^{-j4\pi\frac{\Delta}{\lambda}\sin(\theta_1)} & e^{-j4\pi\frac{\Delta}{\lambda}\sin(\theta_2)} & \cdots & e^{-j4\pi\frac{\Delta}{\lambda}\sin(\theta_d)} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-j2K\pi\frac{\Delta}{\lambda}\sin(\theta_1)} & e^{-j2K\pi\frac{\Delta}{\lambda}\sin(\theta_2)} & \cdots & e^{-j2K\pi\frac{\Delta}{\lambda}\sin(\theta_d)} \end{bmatrix}. \tag{4.9}$$

When incoming signals are closely spaced, the columns of both A and A_1 become almost linearly dependent [44]. The dependency increases drastically when some of $sin(\theta_i), i = 1, \cdots, d$ approach 1 for DOA's near 90^o. As a result, the performance of a linear array deteriorates quickly when some of DOA's approach 90^o. The highly directional sensitivity of the linear array causes the lack of performance robustness to the DOA's and limits the domain of estimable angles to azimuth angles from broadside direction of the array. The lack of performance robustness of a linear array is even more severe when the SS technique is applied, because in the smoothed covariance matrix, not only the steering matrix A_1, but also A is ill-conditioned in the situation described above. A general SS technique that is robust and can be applied to directionally independent arrays is thus more desirable.

4.3 SS for Array of Arbitrary Geometry

4.3.1 Orientational Invariance Structure

It is apparent that the mapping relation between A_k and A_1 is the key to successful application of the SS technique. In general, we can divide an arbitrary array into K subarrays which may overlap. There is not always a steering matrix A to map all the steering matrices A_k, for $k = 1, \cdots, K$ to A. In this section, we will develop necessary and sufficient conditions on array geometries for implementing the general SS. First, we give the following lemmas.

Lemma 4.3.1.1 *For steering matrices A and B, given by $A = [\mathbf{a}(\theta_1), \cdots, \mathbf{a}(\theta_d)]$ and $B = [\mathbf{b}(\theta_1), \cdots, \mathbf{b}(\theta_d)]$, there exists a mapping relation $B = AC$ if and only if C is a diagonal matrix.*

Proof of Lemma 4.3.1.1:

If part:

The proof is obvious and is omited.

Only If part:

If $B = AC$, $A = [a(\theta_1), \cdots, a(\theta_d)]$ and $B = [b(\theta_1), \cdots, b(\theta_d)]$ and also assume C is not a diagonal matrix, i.e. it has non-zero element c_{lm} for $l \neq m$, then the steering vector $b(\theta_m)$ is

$$b(\theta_m) = \sum_{i=1}^{d} c_{im} a(\theta_i) = c_{lm} a(\theta_l) + \sum_{i=1, i \neq l}^{d} c_{im} a(\theta_i). \tag{4.10}$$

This means that $b(\theta_m)$ is a function of variable θ_l, which contradicts to the definition that $b(\theta_m)$ is only a function of θ_m. Thus the assumption that C is a non-diagonal matrix is false. C has to be a diagonal matrix with $c_{ii} = \frac{b(\theta_i)}{a(\theta_i)}$.

Lemma 4.3.1.2 *For K steering matrices $A_1, A_2, \cdots, A_K$, each A_i can be mapped to a steering matrix B if and only if there exists a mapping relation $A_j = A_i C_{ij}$ for any i and j.*

Proof of Lemma 4.3.1.2:

If part:

Obviously, B can be any of $\{A_1, A_2, \cdots, A_K\}$.

Only if part:

If each A_i can be mapped to a steering matrix B, by definition there exist C_i, C_j such that $A_i = BC_i$, $A_j = BC_j$. By Lemma 4.3.1.1, C_i is a diagonal matrix. So C_i^{-1} exists and is also a diagonal matrix. We have $A_j = A_i C_i^{-1} C_j$. Let $C_{ij} = C_i^{-1} C_j$, C_{ij} is the product of two diagonal matrices. So C_{ij} is also an diagonal matrix. $A_j = A_i C_{ij}$.

Consider an array that is divided into K subarrays. Suppose A_i and A_j are the steering matrices associated with the ith and the jth subarrays, and there are

d signals with incoming angles $\theta_1, \cdots, \theta_d$. A_i can be written as

$$A_i = [\mathbf{a}_i(\theta_1), \mathbf{a}_i(\theta_2), \cdots, \mathbf{a}_i(\theta_d)], \tag{4.11}$$

where $\mathbf{a}_i^T(\theta_k) = [e^{-j\phi_{i1}(\theta_k)}, e^{-j\phi_{i2}(\theta_k)}, \cdots, e^{-j\phi_{ip}(\theta_k)}], k = 1, \cdots, d$, is the steering vector associated with the ith subarray, and $\phi_{il}(\theta_k)$, $l \in \{1, \cdots, p\}$, is the phase delay of the kth signal at the lth sensor of the ith subarray from the first sensor of the first subarray. We refer to the sensor of an array associated with the lth row of a steering matrix of the array as the lth sensor of the array.

Let Δ_{ijl}, $1 \leq l \leq p$, represent the distance between the lth sensor in the ith subarray and the lth sensor in the jth subarray. Let β_{ijl} represent the angle of the line on which these two sensors are located. If the ith and the jth subarrays are identical and have the same orientation, i.e. all Δ_{ijl} for $l = 1, \cdots, p$ are equal and all $\beta_{ijl}, l = 1, \cdots, p$ are equal, then the phase delay of a signal with an incoming angle θ_k from each sensor in the ith subarray to the corresponding sensor in the jth subarray is the same according to the far field assumption. We denote this phase delay by $\Phi_{ij}(\theta_k)$. For any $l \in \{1, \cdots, p\}$, we have

$$\Phi_{ij}(\theta_k) = \phi_{jl}(\theta_k) - \phi_{il}(\theta_k) = 2\pi \frac{\Delta_{ijl}}{\lambda} \sin(\beta_{ijl} - \theta_k + \frac{\pi}{2}), \tag{4.12}$$

then $A_j = A_i C_{ij}$, where C_{ij} is a diagonal matrix with the mth diagonal element $e^{-j\Phi_{ij}(\theta_m)}$. The identical and orientational invariance properties between two subarrays guarantee a mapping relation between their steering matrices.

On the other hand, if $A_j = A_i C$, by Lemma 4.3.1.1, C should be a diagonal matrix and can be represented by $C = diag\{c_{11}(\theta_1), c_{22}(\theta_2), \cdots, c_{dd}(\theta_d)\}$. It requires that

$$e^{-j\phi_{il}(\theta_k)} c_{kk}(\theta_k) = e^{-j\phi_{jl}(\theta_k)} \quad \text{for} \quad l = 1, \cdots, p, \tag{4.13}$$

which can be simplified to

$$\phi_{jl}(\theta_k) - \phi_{il}(\theta_k) = \Phi'_{ij}(\theta_k) + 2\pi n, \quad \text{for } l = 1, \cdots, p \tag{4.14}$$

where n can be any integer. The relation in (4.14) holds for all θ_k in $[0, 360)$ only if $\Delta_{ij1} = \Delta_{ij2} = \cdots = \Delta_{ijp}$ and $\beta_{ij1} = \beta_{ij2} = \cdots = \beta_{ijp}$, i.e. the ith and the jth subarrays must be identical and have the same orientation. Thus, we have Lemma 4.3.1.3:

Lemma 4.3.1.3 *Suppose A_i and A_j are steering matrices associated with the ith and the jth subarrays. The sensors in each subarray are numbered in the same sequence. There exists a mapping relation $A_j = A_i C_{ij}$ if and only if the ith and the jth subarrays are identical and have the same orientation.*

From Lemmas 4.3.1.2 and 4.3.1.3, we have:

Theorem 4.3.1.1 *Suppose an array can be divided into K subarrays, each having a $p \times d$ steering matrix A_i, $i = 1, 2, \cdots, K$. All $A_1, A_2, \cdots, A_K$ can be mapped to a $p \times d$ steering matrix B by $A_i = BD_i$ if and only if all these subarrays are identical and have the same orientation.*

We call the array structure held by an array satisfying conditions in Theorem 4.3.1.1 the orientational invariance structure. A more rigorous definition is given as follows:

Definition 4.3.1.1 (Orientational Invariance Structure) *An array has an orientational invariance structure if it can be divided into subarrays that are identical and have the same orientation.*

For an array with orientational invariance structure, we can consider each subarray as one element located at its first sensor. Then all these elements form a center array. A more rigorous definition for center array is given as follows:

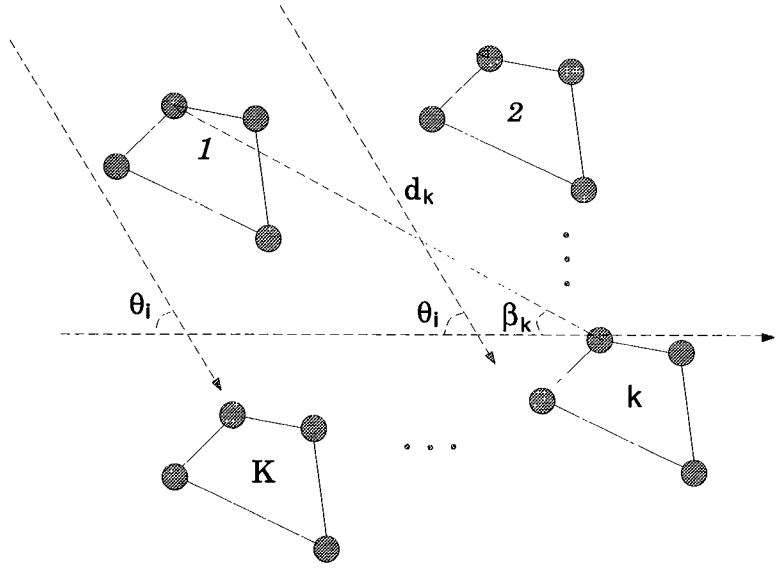


Figure 4.2: Orientational invariance sensor array geometry

Definition 4.3.1.2 (Center Array) *an array with orientational invariance structure is divided into subarrays (which can have overlap), then the collection of all the first sensors of these subarrays form a center array.*

4.3.2 Necessary and Sufficient Conditions

Suppose an array has an orientational invariance structure. Moreover, its center array has an ambiguity free structure and the number of subarrays is larger than or equal to the largest number of mutually coherent signals. The $p \times d$ steering matrices $A_1, A_2, \cdots, A_K$ are associated with the subarrays $1, 2, \cdots, K$, respectively, and d_k is the distance between the first sensor in the first subarray and the first sensor in the kth subarray. The angle β_k represents the direction of the line on which the first sensor in the first subarray and the first sensor in the kth subarray

are located (see Fig.4.2). We have

$$A_k = A_1 D_k, \quad k = 2, \cdots, K \tag{4.15}$$

where

$$D_k = \begin{bmatrix} e^{-j\frac{2\pi d_k}{\lambda}\sin(\beta_k - \theta_1 + \frac{\pi}{2})} & & & \\ & e^{-j\frac{2\pi d_k}{\lambda}\sin(\beta_k - \theta_2 + \frac{\pi}{2})} & & \\ & & \ddots & \\ & & & e^{-j\frac{2\pi d_k}{\lambda}\sin(\beta_k - \theta_d + \frac{\pi}{2})} \end{bmatrix}. \tag{4.16}$$

The covariance matrix of the kth subarray is thus given by

$$R_k = A_1 D_k R_s D_k^H A_1^H + \sigma^2 I, \tag{4.17}$$

where R_s is the covariance matrix of the source. The spatially smoothed covariance matrix is defined as the average of the subarray covariances

$$\bar{R} = \frac{1}{K}\sum_{k=1}^{K} R_k = A_1 \bar{R}_s A_1 + \sigma^2 I, \tag{4.18}$$

where $\bar{R}_s$ is the modified covariance matrix of the signal given by

$$\bar{R}_s = \frac{1}{K}\sum_{k=1}^{K} D_k R_s D_k^H. \tag{4.19}$$

We will show in the following that $\bar{R}_s$ is nonsingular. First, $\bar{R}_s$ can be written as

$$\bar{R}_s = [I \; D_2 \; \cdots D_K] \begin{bmatrix} \frac{1}{K}R_s & & & \\ & \frac{1}{K}R_s & & \\ & & \ddots & \\ & & & \frac{1}{K}R_s \end{bmatrix} \begin{bmatrix} I \\ D_2^H \\ \vdots \\ D_K^H \end{bmatrix}. \tag{4.20}$$

Let C denote the Hermitian square root of $\frac{1}{K}R_s$, i.e.

$$CC^H = \frac{1}{K}R_s. \tag{4.21}$$

It follows that

$$\bar{R}_s = GG^H \tag{4.22}$$

where G is a $d \times Kd$ block matrix given by

$$G = [C \; D_2C \; \cdots \; D_KC]. \tag{4.23}$$

Clearly, the rank of $\bar{R}_s$ is equal to the rank of G. Suppose there are q groups of signals in d incoming signals, with l_i, $i = 1, \cdots, q$, correlated signals in each group. R_s must be a block diagonal matrix with block size l_i, $i = 1, \cdots, q$. We can thus get a corresponding block diagonal matrix C. If we exchange the columns of a matrix, the rank of the matrix does not change. By grouping columns of similar elements, we can verify that

$$rank(G) = \rho \begin{bmatrix} c_{1,1}\mathbf{b}_1 & \cdots & c_{1,l_1}\mathbf{b}_1 & & & & \\ \vdots & \cdots & \vdots & & & & \\ c_{l_1,1}\mathbf{b}_{l_1} & \cdots & c_{l_1,l_1}\mathbf{b}_{l_1} & & & & \\ & & & \ddots & & & \\ & & & & c_{d-l_q+1,d-l_q+1}\mathbf{b}_{d-l_q+1} & \cdots & c_{d-l_q+1,d}\mathbf{b}_{d-l_q+1} \\ & & & & \vdots & \cdots & \vdots \\ & & & & c_{d,d-l_q+1}\mathbf{b}_d & \cdots & c_{d,d}\mathbf{b}_d \end{bmatrix} \tag{4.24}$$

c_{ij} is the ijth element of matrix C, and $\mathbf{b}_i$ $(i = 1, \cdots, d)$ is the $1 \times K$ row vector given by

$$\mathbf{b}_i = [1 \; e^{-j\frac{2\pi d_2}{\lambda} sin(\beta_2 - \theta_i + \frac{\pi}{2})} \; e^{-j\frac{2\pi d_3}{\lambda} sin(\beta_3 - \theta_i + \frac{\pi}{2})} \; \cdots e^{-j\frac{2\pi d_K}{\lambda} sin(\beta_K - \theta_i + \frac{\pi}{2})} \;]. \tag{4.25}$$

Each row of matrix C has at least one nonzero element because the energy of each signal is nonzero. It is observed that $\mathbf{b}_i$ is the transpose of the steering

vector associated with the center array. Since the center array is assumed to have ambiguity free array manifold, when $K \geq \max\{l_1, l_2, \cdots, l_q\}$, all the $\mathbf{b}$ vectors associated with all the signals within a group of coherent signals are thus linearly independent. Therefore, G is of full row rank and the modified covariance matrix $\bar{R}_S$ is of full rank. Otherwise, if $K < \max\{l_1, l_2, \cdots, l_q\}$, we will see that $\bar{R}_s$ is rank deficient. We assume that R_{si} is the correlation matrix associated with the ith group of coherent signals. Thus, R_{si} has rank 1 and can be expressed as $\lambda_i \mathbf{h}_i \mathbf{h}_i^H$ where λ_i and $\mathbf{h}_i$ are the corresponding eigenvalue and eigenvector of R_{si}. We have

$$R_s = \begin{bmatrix} R_{s1} & & & \\ & R_{s2} & & \\ & & \ddots & \\ & & & R_{sq} \end{bmatrix} = \begin{bmatrix} \lambda_1 \mathbf{h}_1 \mathbf{h}_1^H & & & \\ & \lambda_2 \mathbf{h}_2 \mathbf{h}_2^H & & \\ & & \ddots & \\ & & & \lambda_q \mathbf{h}_q \mathbf{h}_q^H \end{bmatrix} \tag{4.26}$$

and

$$\bar{R}_s = \begin{bmatrix} \sum_{k=1}^{K} \lambda_1 (D_{k_1} \mathbf{h}_1)(D_{k_1} \mathbf{h}_1)^H & & & \\ & \sum_{k=1}^{K} \lambda_2 (D_{k_2} \mathbf{h}_2)(D_{k_2} \mathbf{h}_2)^H & & \\ & & \ddots & \\ & & & \sum_{k=1}^{K} \lambda_q (D_{k_q} \mathbf{h}_q)(D_{k_q} \mathbf{h}_q)^H \end{bmatrix} \tag{4.27}$$

where D_{k_i} is a diagonal matrix consisting of l_i diagonal elements of D_k which are associated with all the DOA's from ith group of coherent signals. Since

$$rank(\sum_{k=1}^{K} \lambda_i (D_{k_i} \mathbf{h}_i)(D_{k_i} \mathbf{h}_i)^H) \leq min(K, l_i) \tag{4.28}$$

and

$$dim(\sum_{k=1}^{K} \lambda_i (D_{k_i} \mathbf{h}_i)(D_{k_i} \mathbf{h}_i)^H) = l_i \times l_i, \tag{4.29}$$

thus $\bar{R}_s$ is rank deficient if $K \leq max(l_1, l_2, \cdots, l_q)$.

If the center array is not ambiguity free, then all the **b** vectors associated with all the signals within a group of coherent signals can be linearly dependent, G cannot be ensured to be of full row rank, and neither can R_s.

From Theorem 4.3.1.1 and the proof above, we get the following theorem.

Theorem 4.3.2.1 *SS can be applied to an array of arbitrary geometry to obtain a full rank smoothed signal covariance matrix if and only if an array has an orientational invariance structure, its center array has an ambiguity free structure, and the number of subarrays is larger than or equal to the size of the largest group of coherent signals.*

4.3.3 Further Improvement

To get a smoothed nonsingular covariance matrix $\bar{R}_s$ by using the SS technique, we need $K \geq max\{l_1, l_2, \cdots, l_q\}$. We can further reduce the number of subarrays by getting another K backward subarrays similar to the case in a linear array [40]. Although, the Forward-Backward Spatial Smoothing (FBSS) [40] can always be applied in a uniformly spaced linear array. For arrays of arbitrary geometry, there is some requirements on the geometry for successful implementation of the backward method. We first give the definition of central symmetry:

Definition 4.3.3.1 (Central Symmetry) *The array is central symmetric if it is identical before and after rotating* 180^0 *about its center of mass.*

If an array is central symmetric, we can get K additional backward subarrays by reversing the order of the subarrays and the order of the sensors within each subarray. Let $\mathbf{r}_k^b(t)$ denote the complex conjugate of the output of the kth backward

subarray for $k = 1, \cdots, K$. We have

$$\mathbf{r}_k^b(t) = A_1 D_k (D_L \mathbf{s}(t))^* + \tilde{\mathbf{n}}^*(t) \tag{4.30}$$

where $\tilde{\mathbf{n}}(t)$ is an additive white Gaussian noise vector, D_L is a diagonal matrix with the ith diagonal element given by $e^{-j2\pi \frac{d_{Kp}}{\lambda} \sin(\beta_{Kp} - \theta_i + \frac{\pi}{2})}$ and d_{Kp} is the distance between the first sensor in the first forward subarray and the first sensor in the first backward subarray. The angle β_{Kp} represents the direction of the line on which the two sensors are located.

The covariance matrix of the kth backward subarray is given by

$$R_k^b = A_1 D_k R_s^b D_k^H A_1^H + \sigma^2 I \tag{4.31}$$

with

$$R_s^b = E(D_L^* \mathbf{s}^*(t) \mathbf{s}^T(t) D_L^T) = D_L^* R_s^* D_L^T. \tag{4.32}$$

Define the spatially smoothed backward subarray covariance matrix $\bar{R}^b$ as the average of these subarray covariance matrices, i.e.,

$$\bar{R}^b = \frac{1}{K} \sum_{k=1}^{K} R_k^b = A_1 \bar{R}_s^b A_1^H + \sigma^2 I, \tag{4.33}$$

where

$$\bar{R}_s^b = \frac{1}{K} \sum_{k=1}^{K} D_k R_s^b D_k^H, \tag{4.34}$$

and define the forward/backward smoothed covariance matrix $\tilde{R}$ as the average of $\bar{R}$ in (4.18) and $\bar{R}^b$, i.e.,

$$\tilde{R} = \frac{\bar{R} + \bar{R}^b}{2} = A_1 \tilde{R}_s A_1^H + \sigma^2 I. \tag{4.35}$$

It follows that

$$\tilde{R}_s = \frac{\bar{R}_s + \bar{R}_s^b}{2}. \tag{4.36}$$

We can show, in a similar way as in the case of a linear array [40], that the modified source covariance matrix $\tilde{R}_s$ is nonsingular as long as $2K \geq \max\{l_1, l_2, \cdots, l_q\}$.

4.4 Ambiguity Free Array Structure

To perform SS, we need an ambiguity free center array manifold. Also, to perform MUSIC, we further require ambiguity free subarray manifolds. Ambiguity arises when a steering vector can be expressed as a linear combination of other steering vectors in an array manifold [25]. For a uniformly spaced linear array, rank-1 ambiguity [25] cannot be avoided since the DOA's which are "mirror images" with respect to the array line, have the same steering vector. This limits the range of DOA's estimable by a uniformly spaced linear array to within 180^0. Suppose an array has p elements, then rank-p [25] ambiguities cannot be avoided. In this chapter, an ambiguity free array manifold of an array of p sensors refers to rank-(p-1) ambiguity free. Generally, to avoid ambiguity, an array used for high-resolution DOA estimation must have a proper structure. An ambiguity free array manifold has been assumed in several papers [26], [17], [15]. Our attempt is to identify all the situations in which ambiguity may arise. One of our guidelines in designing arrays is to avoid these identified ambiguities.

Theorem 4.4.1 *In an azimuth only system, the necessary and sufficient condition for an ambiguity free three-sensor array manifold is that all these three sensors are not on one line and that the distance between any two sensors is less than or equal to* $\frac{\lambda}{2}$.

Proof of Theorem 4.4.1:

If part:

If sensors A, B and C are not on one line and their mutual distance is less than $\frac{\lambda}{2}$, without loss of generality, we let sensor A be the first sensor in the array, B the

second and C the third. The steering matrix of the array has the form

$$V = \begin{bmatrix} 1 & 1 & 1 \\ e^{j\phi_1(\theta_1)} & e^{j\phi_1(\theta_2)} & e^{j\phi_1(\theta_3)} \\ e^{j\phi_2(\theta_1)} & e^{j\phi_2(\theta_2)} & e^{j\phi_2(\theta_3)} \end{bmatrix} \tag{4.37}$$

where ϕ denotes phase delay. If the distance between any two sensors is $< \frac{\lambda}{2}$, the phase delay $\phi_1(\theta_i)$ and $\phi_2(\theta_i), i = 1, 2, 3$, are real numbers from $(-\pi, \pi)$.

Note that the steering matrix of the array corresponding to three incoming signals at different angles is a special case of the general array in Lemma 4.3.1.2 of [43]. By Lemma 4.3.1.2 in [43], V is nonsingular with possible exception in one of the following three situations:

(1) When $\phi_1(\theta_1)$=$\phi_1(\theta_2)$, i.e. the two incoming signals are symmetric with respect to the line on which sensors A and B are located. Note that

$$det(V) = \begin{vmatrix} 0 & 1 & 0 \\ e^{j\phi_1(\theta_1)} - e^{j\phi_1(\theta_2)} & e^{j\phi_1(\theta_2)} & e^{j\phi_1(\theta_3)} - e^{j\phi_1(\theta_2)} \\ e^{j\phi_2(\theta_1)} - e^{j\phi_2(\theta_2)} & e^{j\phi_2(\theta_2)} & e^{j\phi_2(\theta_3)} - e^{j\phi_2(\theta_2)} \end{vmatrix} \tag{4.38}$$

$$= - \begin{vmatrix} e^{j\phi_1(\theta_1)} - e^{j\phi_1(\theta_2)} & e^{j\phi_1(\theta_3)} - e^{j\phi_1(\theta_2)} \\ e^{j\phi_2(\theta_1)} - e^{j\phi_2(\theta_2)} & e^{j\phi_2(\theta_3)} - e^{j\phi_2(\theta_2)} \end{vmatrix}. \tag{4.39}$$

When $\phi_1(\theta_1) = \phi_1(\theta_2)$, $\det(V) = 0$ if and only if $\phi_2(\theta_1) = \phi_2(\theta_2)$ or $\phi_1(\theta_3) = \phi_1(\theta_2)$. Since these sensors are not on one line, if $\phi_1(\theta_1) = \phi_1(\theta_2)$, we have $\phi_2(\theta_1) \neq \phi_2(\theta_2)$. Since θ_1, θ_2 and θ_3 are three different angles, when θ_1 and θ_2 are symmetric with respect to the line, θ_3 and θ_2 cannot be symmetric to the line, i.e. if $\phi_1(\theta_1) = \phi_1(\theta_2)$, then we get $\phi_1(\theta_3) \neq \phi_1(\theta_2)$. Thus, when $\phi_1(\theta_1) = \phi_1(\theta_2)$, the matrix V is nonsingular.

(2) Similarly, we can prove that when $\phi_2(\theta_1) = \phi_2(\theta_2)$, the matrix V is nonsingular.

(3) When $\phi_1(\theta_1) - \phi_1(\theta_2) = \phi_2(\theta_1) - \phi_2(\theta_2)$, i.e. $\phi_1(\theta_1) - \phi_2(\theta_1) = \phi_1(\theta_2) - \phi_2(\theta_2)$, θ_1 and θ_2 are symmetric with respect to the line connecting sensors B and C. Note that

$$det(V) = \begin{vmatrix} e^{-j\phi_2(\theta_1)} & e^{-j\phi_2(\theta_2)} & e^{-j\phi_2(\theta_3)} \\ e^{j\phi_1(\theta_1)-j\phi_2(\theta_1)} & e^{j\phi_1(\theta_2)-j\phi_2(\theta_2)} & e^{j\phi_1(\theta_3)-j\phi_2(\theta_3)} \\ 1 & 1 & 1 \end{vmatrix} \quad (4.40)$$

$$= (e^{-j\phi_2(\theta_1)} - e^{-j\phi_2(\theta_2)})(e^{j\phi_1(\theta_2)-j\phi_2(\theta_2)} - e^{j\phi_1(\theta_3)-j\phi_2(\theta_3)}) \quad (4.41)$$

When $\phi_1(\theta_1) - \phi_2(\theta_1) = \phi_1(\theta_2) - \phi_2(\theta_2)$, $\det(V) = 0$ if and only if $e^{-j\phi_2(\theta_1)} = e^{-j\phi_2(\theta_2)}$ or $e^{j\phi_1(\theta_2)-j\phi_2(\theta_2)} = e^{j\phi_1(\theta_3)-j\phi_2(\theta_3)}$. Since the mutual distance between A, B and C are less than $\frac{\lambda}{2}$, $\phi_2(\theta_1)$, $\phi_2(\theta_2)$, $\phi_1(\theta_2) - \phi_2(\theta_2)$ and $\phi_1(\theta_3) - \phi_2(\theta_3)$ are all real numbers in $(-\pi, \pi)$. $e^{-j\phi_2(\theta_1)} = e^{-j\phi_2(\theta_2)}$ if and only if $\phi_2(\theta_1) = \phi_2(\theta_2)$. $e^{j\phi_1(\theta_2)-j\phi_2(\theta_2)} = e^{j\phi_1(\theta_3)-j\phi_2(\theta_3)}$ if and only if $\phi_1(\theta_2) - \phi_2(\theta_2) = \phi_1(\theta_3) - \phi_2(\theta_3)$.

Since A, B and C are not on one line, if θ_1 and θ_2 are symmetric to the line connecting B and C, they can not be symmetric to the line connecting A and B or A and C. i.e., if $\phi_1(\theta_1) - \phi_2(\theta_1) = \phi_1(\theta_2) - \phi_2(\theta_2)$, we have $\phi_2(\theta_1) \neq \phi_2(\theta_2)$. Since θ_1, θ_2 and θ_3 are three different incoming angles, if θ_1 and θ_2 are symmetric to the line connecting B and C, θ_2 and θ_3 can not be symmetric to the line. i.e., if $\phi_1(\theta_1) - \phi_2(\theta_1) = \phi_1(\theta_2) - \phi_2(\theta_2)$, we have $\phi_1(\theta_2) - \phi_2(\theta_2) \neq \phi_1(\theta_3) - \phi_2(\theta_3)$. Thus, when $\phi_1(\theta_1) - \phi_2(\theta_1) = \phi_1(\theta_2) - \phi_2(\theta_2)$, the matrix V is nonsingular.

Therefore, we conclude that all the three situations which cause the singularity of the matrix in Lemma 4.3.1.2 of [43] will not cause the singularity of three-sensor steering matrix if three sensors are not on one line and their mutual distance is less than $\frac{\lambda}{2}$. Therefore the matrix V is full rank.

If the spacing between any two of the three sensors is not larger than $\frac{\lambda}{2}$, and there is at least one pair in these three sensors with a spacing of $\frac{\lambda}{2}$, then the only situation that the phase delay $\phi_1(\theta_i)$ and $\phi_2(\theta_i), i = 1, 2, 3$, are not all in $(-\pi, \pi)$ is when one of the incoming signals is from the direction parallel to a line on which the two sensors with spacing $\frac{\lambda}{2}$ are located. The other two signals can be either from the opposite direction or from other directions. If one of the other two signals is from the opposite direction, it can be easily proved that the corresponding steering matrix is full rank. If the other two signals are from the two other different directions, then one of $\phi_n(\theta_i)$, $n = 1, 2$, $i = 1, 2, 3$ is equal to π and the rest are real numbers from $(-\pi, \pi)$. Similarly, we can prove that the matrix V is of full rank.

Only if part:

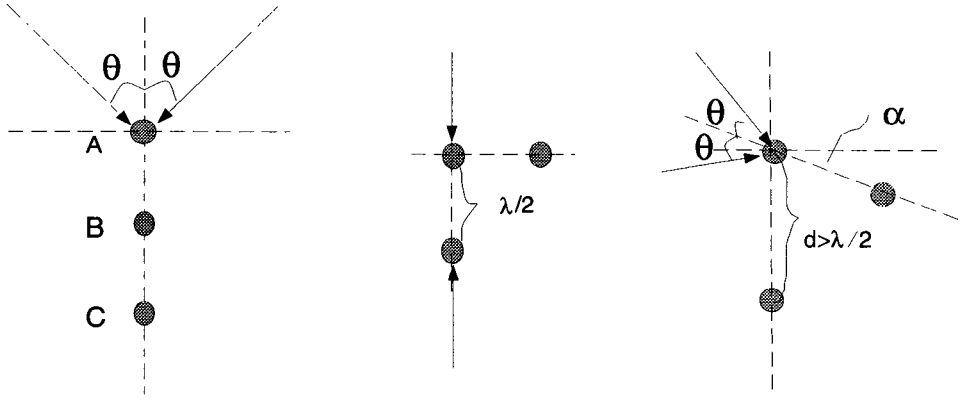
If the conditions in Theorem 4.3.1.1 are not satisfied, rank-1 or rank-2 ambiguity occurs for some incoming signals. These situations are shown schematically in Fig.4.3(a)(b). In Fig.4.3(a), the relation between θ and α is

$$2\pi\frac{d}{\lambda}sin(\theta - \alpha) + k2\pi = 2\pi\frac{d}{\lambda}sin(\theta + \alpha) \quad k \in \{1, 2, \cdots\}. \tag{4.42}$$

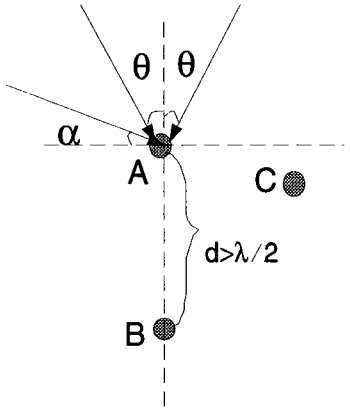
In Fig.4.3(b), the relation between θ and α is

$$2\pi\frac{d}{\lambda}sin(\alpha) + k2\pi = 2\pi\frac{d}{\lambda}sin(\frac{\pi}{2} - \theta), \quad k \in \{1, 2, \cdots\}. \tag{4.43}$$

We can see in general that (a) rank-1 ambiguity occurs not only in uniformly spaced linear arrays but also in rectangular arrays with sensors having a uniform spacing of $\frac{\lambda}{2}$ along either x-axis or y-axis, (b) rank-2 ambiguity occurs in an array that consists of two parallel linear arrays with an identical uniform sensor spacing that is larger than $\frac{\lambda}{2}$, (c) rank-3 ambiguity occurs in an array that consists of three parallel linear arrays with an identical uniform sensors spacing that is larger

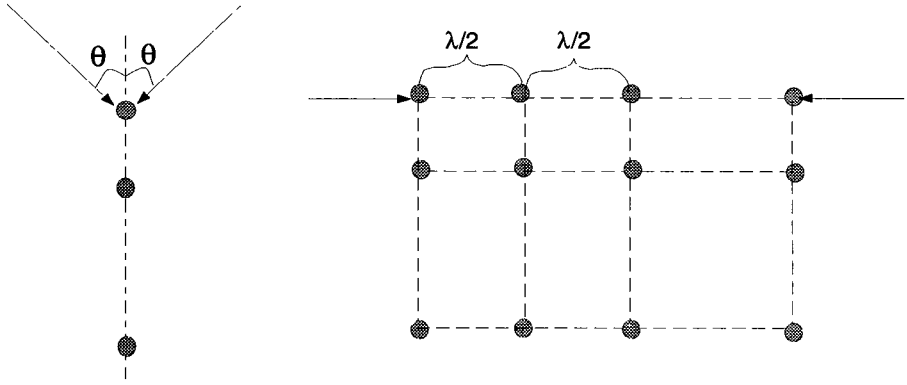


(a) rank-1 ambiguity

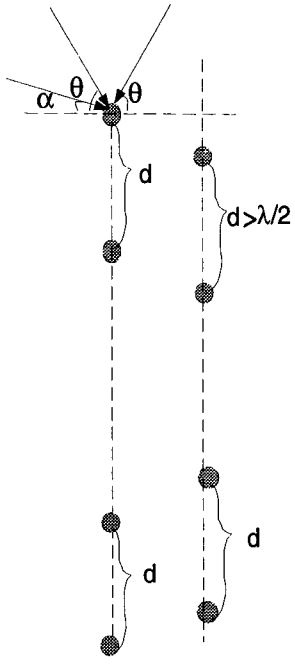


(b) rank-2 ambiguity

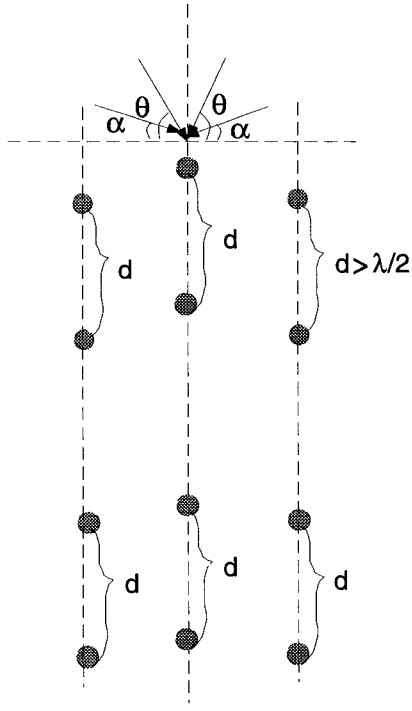
Figure 4.3: Three-sensor array structures that can cause ambiguities



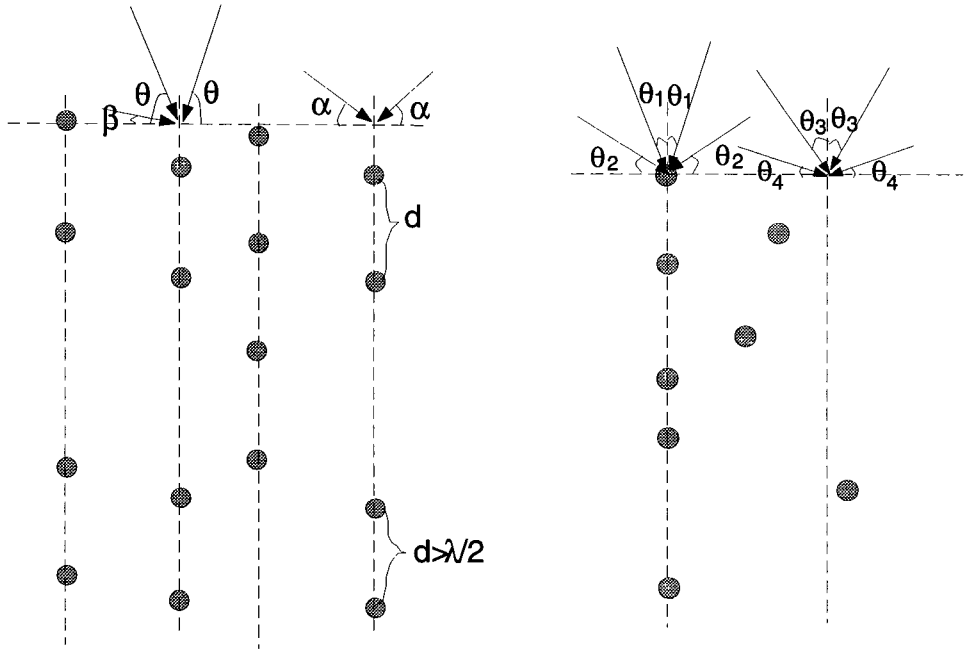
(a) rank-1 ambiguity



(b) rank-2 ambiguity



(c) rank-3 ambiguity



(d) high order ambiguity

Figure 4.4: high order array structures that can cause ambiguities

than $\frac{\lambda}{2}$, and (d) higher order ambiguity occurs if more than $\lceil \frac{k}{2} \rceil$ sensors are on one line in a k sensor array or if an array consists of m parallel linear arrays with an identical uniform sensor spacing that is larger than $\frac{\lambda}{2}\lfloor \frac{m}{2} \rfloor$. These situations are shown schematically in Fig.4.4(a)-(d). In Fig.4.4 (b) and (c), the angles θ and α satisfy the following constraint:

$$2\pi \frac{d}{\lambda} \sin(\alpha) + 2k\pi = 2\pi \frac{d}{\lambda} \sin(\theta), \quad k \in \{1, 2, \cdots\}. \tag{4.44}$$

In Fig.4.4 (d), the angles θ, β and α satisfy the following constraint:

$$2\pi \frac{d}{\lambda} \sin(\alpha) + 2k_1\pi = 2\pi \frac{d}{\lambda} \sin(\beta) + 2k_2\pi = 2\pi \frac{d}{\lambda} \sin(\theta), \tag{4.45}$$

where k_1,$k_2 \in \{1, 2, \cdots\}$ and $k_1 \neq k_2$.

To get an ambiguity free array manifold, it is necessary to avoid these identified situations.

4.5 Spatial Smoothing for ESPRIT

Similar to MUSIC, the ESPRIT algorithm [26] is an approach to signal parameter estimation. It exploits an underlying data model at significant computational savings. The ESPRIT algorithm is also limited to estimating parameters in noncoherent incoming signals. The conventional SS can be incorporated into ESPRIT [47], but it requires the center array to be a uniformly spaced linear array. In this section, we show that our scheme also works for the ESPRIT algorithm to estimate parameters in a coherent interference environment.

In the ESPRIT algorithm, we consider d narrow-band plane waves with incident angles $\theta_1, \cdots \theta_d$, and wavelength λ, impinge on a planar array of m sensors (m is even), arranged in $\frac{m}{2}$ doublet pairs. The displacement vector is the same for each

doublet pair, but the location of each pair is arbitrary. The sensor output $x(t)$ is given by

$$\mathbf{x}(t) = \begin{pmatrix} A \\ A\Phi \end{pmatrix} \mathbf{s}(t) + \mathbf{n}(t) \tag{4.46}$$

where $\mathbf{n}(\mathbf{t})$ is a white Gaussian noise vector. A and $A\Phi$ are the steering matrices corresponding to the first sensors and the second sensors in all pairs, respectively. The matrix Φ is a diagonal $d \times d$ matrix of phase delays between the doublet sensors for the d signals. The sensor output covariance matrix R_x is thus measured by

$$R_x = \begin{pmatrix} A \\ A\Phi \end{pmatrix} R_s \begin{pmatrix} A \\ A\Phi \end{pmatrix}^H + \sigma^2 I. \tag{4.47}$$

A full rank matrix R_s is assumed when the ESPRIT algorithm is performed. If some of the incoming signals are coherent, R_s will not be a full rank matrix and the ESPRIT will fail. The spatial smoothing technique we introduced in the previous sections can then be applied here to get a modified full rank signal covariance matrix.

We consider each doublet sensor pair in the array used by ESPRIT algorithm as one element. Then the array consists $\frac{m}{2}$ elements. If this array has an orientational invariance structure with K subarrays and the corresponding center array has an ambiguity free structure, the sensor output at the kth subarray is given by

$$\mathbf{x}_k(t) = \begin{pmatrix} A_1 \\ A_1\Phi \end{pmatrix} D_k \mathbf{s}(t) + \mathbf{n}(t). \tag{4.48}$$

Matrix D_k is a diagonal $d \times d$ matrix of the phase delays in the form given in (4.16). The corresponding covariance matrix R_{x_k} is given by

$$R_{x_k} = \begin{pmatrix} A_1 \\ A_1\Phi \end{pmatrix} D_k R_s D_k^H \begin{pmatrix} A_1 \\ A_1\Phi \end{pmatrix}^H + \sigma^2 I. \tag{4.49}$$

A smoothed output covariance matrix $\bar{R}_x$ can thus be defined as

$$\bar{R}_x = \frac{1}{K}\sum_{i=1}^{K} R_{x_k} = \begin{pmatrix} A_1 \\ A_1\Phi \end{pmatrix} \bar{R}_s \begin{pmatrix} A_1 \\ A_1\Phi \end{pmatrix}^H + \sigma^2 I \tag{4.50}$$

where $\bar{R}_s$ is the modified signal covariance matrix as defined in (4.19). As proved in Section II, $\bar{R}_s$ is of full rank if K is larger than or equal to the size of the largest group of coherent signals. We can now successfully perform ESPRIT based on $\bar{R}_x$. We can also use FBSS to further reduce the number of sensors required and to improve the estimation resolution if the array of $\frac{m}{2}$ element is central symmetric.

Although SS enables ESPRIT to estimate DOA's in a coherent interference environment, the estimation is still limited to identifying DOA's within 180^o in an azimuth only system. Hence, in terms of performance robustness to DOA's, our SS is more effective for MUSIC than for ESPRIT.

Chapter 5

Capacity Increase with Constrained Adaptive Beamforming

5.1 Introduction

An alternative to using reference signal based adaptive array combining is to use constrained adaptive array combining. It is based on high-resolution direction finding followed by optimal combination of the antenna outputs. By actively tracking mobile units and directionally transmitting information to and receiving information from these units, significant improvements are achieved without any modification to the mobile units and without using reference signals. It is independent of the particular type of signal modulation and is therefore compatible with current and future modulation schemes in wireless communication systems. It allows a reduction in transmitter power at the base station by directive transmission while still improving signal quality by increasing amount of power received by the mobile unit.

The DOA based array approach is limited to some environment where angle

spread is small and SS and DOA estimation is possible. In this chapter, we will demonstrate the effectiveness of our SS in decorrelating coherent signals under these environments.

This chapter is organized as follows: In section 5.2, we will discuss some practical issues of using SS. In section 5.3 we will verify the theoretical results obtained in Chapter 4 using computer simulations. In section 5.4, we will demonstrate the effectiveness of our SS under multipath multiuse TDMA wireless communication environment. In section 5.5, we will present the techniques of suppressing interference and achieving capacity increase using DOA based adaptive array.

5.2 Some Practical Considerations

5.2.1 Performing SS in the Data Domain

In practice, we can perform FBSS by setting up a special data matrix. Specifically, for the nth snapshot we set up the data matrix

$$A^H(n) = \begin{bmatrix} u(p,1,n) & \cdots & u(p,K,n) & u^*(1,K,n) & \cdots & u^*(1,1,n) \\ u(p-1,1,n) & \cdots & u(p-1,K,n) & u^*(2,K,n) & \cdots & u^*(2,1,n) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ u(1,1,n) & \cdots & u(1,K,n) & u^*(p,K,n) & \cdots & u^*(p,1,n) \end{bmatrix} \tag{5.1}$$

where $u(i,j,n)$ denotes the sample taken at the ith sensor of the jth subarray. For the totality of N snapshots, we can define the overall data matrix

$$A^H = \left[A^H(1), A^H(2), \cdots, A^H(N)\right]. \tag{5.2}$$

It follows that the averaged smoothed correlation matrix $\tilde{R}$ (as defined in (4.35)) can be estimated as follows:

$$\tilde{R} = \frac{1}{2NK} A^H A. \tag{5.3}$$

As we know, more robust results can be obtained from data domain rather than from covariance domain [37]. We can proceed with MUSIC [37] algorithm or MVDR [28] [37] beamforming algorithm based on A instead of $\tilde{R}$.

An array needs to be chosen for applying SS. Such an array should satisfy all conditions aforementioned. An omnidirectional circular array has been a conventional choice for mobile communications [10] [11], and there have been active research efforts to find a pre-processing scheme for the circular array to handle the coherent interference [46]. However, we can see clearly from our discussion that a single circular array is not orientational invariant. Therefore it does not satisfy the necessary condition for applying SS. This implies that the circular array cannot overcome the coherent interference by using the SS technique. For some circular arrays with central symmetric, we can apply FBSS to handle two coherent signals. To handle more than two coherent signals, several parallel circular arrays have to be used.

5.2.2 Selecting Orientational Invariance Structure

In this section, we study some guidelines for designing an optimal sensor array for SS. We found that the sensor utilization rate is an important factor for estimating DOA's of coherent signals with SS.

Definition 5.2.2.1 (Sensor Utilization Rate (SUR))

$$SUR = \frac{\sum n_{subarray}}{n_{array}} \tag{5.4}$$

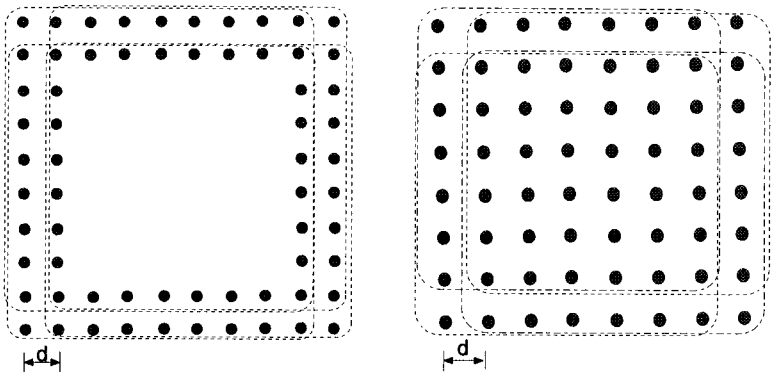


Figure 5.1: (a) A sixty four-sensor dense square array with four overlapping dense square subarrays of forty nine-sensors (b)A sixty four-sensor hollow square array with four overlapping hollow square subarrays of thirty two-sensors

where $\sum n_{subarray}$ is the sum of the number of sensors in each subarray, and n_{array} is the total number of sensors in the whole array. Obviously $SUR \geq 1$, because of possible overlap of subarrays.

Example 5.2.2.1: We perform simulations on two 64-element arrays: (1) a hollow square array as shown in Fig.5.1(a), which has a low SUR for a given number of sensors, and (2) a dense square array, as shown in Fig.5.1(b), which has a high SUR. The dense square array contains 4 subarrays each having 49 sensors. The spacing between two neighboring sensors is 0.45λ. The SUR of the array is approximately 3. The hollow square array contains 4 subarrays, each having 32 sensors. The spacing between two neighboring sensors is 0.45λ. The SUR is 2. Both structures are used to estimate the DOA's of two coherent signals. The input SNR is 20dB. We use the FBSS method. The simulation results are shown in Fig.5.2. In cases (a) and (b), the two coherent signals are at 40^o and 50^o. Both arrays can clearly

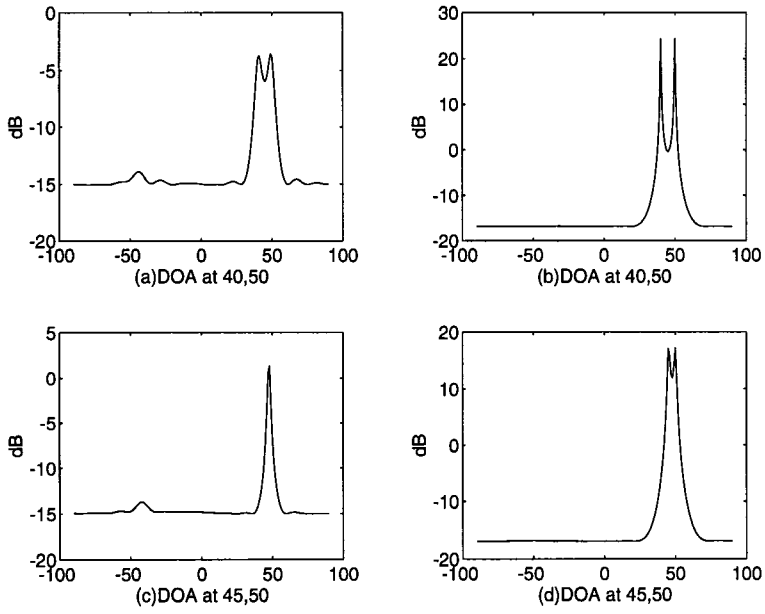


Figure 5.2: A sixty four-sensor hollow square array is used in (a) and (c), A sixty four-sensor dense square array is used in (b) and (d)

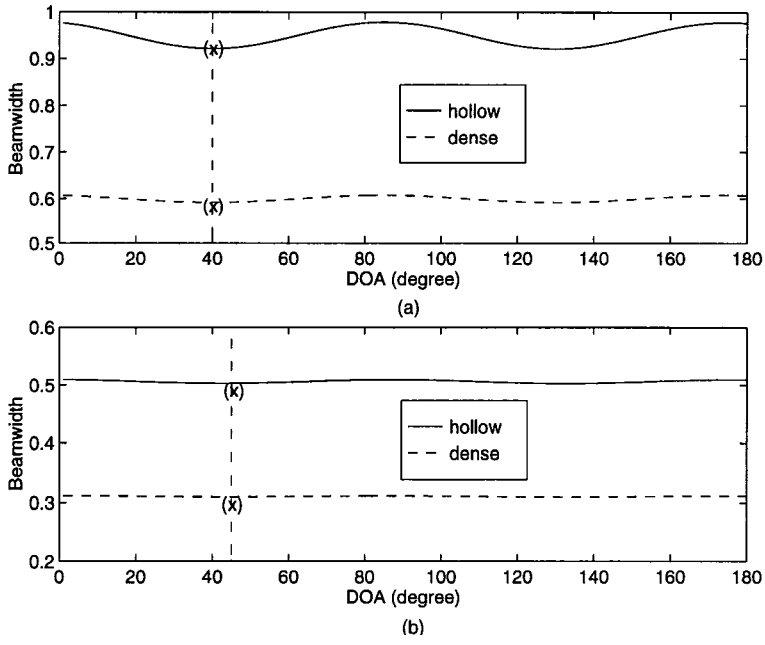


Figure 5.3: Beamwidth comparison between a dense square array and a hollow square array (a) two DOAs are 10^o apart (b) two DOAs are 5^o apart

identify the DOA's. In cases (c) and (d), the two coherent signals are at 45^o and 50^o, only the dense square array can determine the DOA's. Our results show that a dense square array structure is better than a hollow square array structure.

For a uniformly spaced linear array, the larger the physical aperture, the narrower the beamwidth, and the higher the resolution. For a nonlinear array, the DOA resolution is however decided by an effective array beamwidth. The beamwidth $b_{1,2}$ of two signals, with DOA at θ_1 and θ_2, arriving at an array with steering vector $\mathbf{v}(\theta)$ is evaluated by the following equation.

$$b_{1,2} = \frac{2}{\pi} cos^{-1}\left(\frac{|\mathbf{v}(\theta_1)^H \mathbf{v}(\theta_2)|}{|\mathbf{v}(\theta_1)||\mathbf{v}(\theta_2)|}\right) \tag{5.5}$$

We compared the beamwidths between a dense square subarray and a hollow square

subarray. In Fig. 5.3, we plot the beamwidths of two signals, that are 10^o and 5^o apart in (a) and (b), respectively, versus the angle of arrival of the signal that has a smaller DOA values. We see that although the dense square array has a smaller physical array aperture, it has a narrower beamwidth. Since both arrays have the same number of elements, we infer that SUR is an important factor and needs to be maximized in the array design. An array consisting of several parallelly positioned circular arrays has a low SUR and is not recommended for SS applications.

5.3 Simulation Results

In this section, we present some simulation results on MUSIC algorithm to show the applications and effectiveness of our SS and FBSS. We choose a square array, which has an orientational invariance structure, central symmetric, and a sensor spacing less than $\frac{\lambda}{2}$.

Example 5.3.1: To compare the performance of a square array to that of a linear array of the same complexity, we use a nine-sensor linear array and a nine-sensor square array as shown in Fig.5.4, both having a spacing of 0.45λ between neighboring sensors. We divide the both arrays into four overlapped subarrays. We get six and four sensors in each subarray of the linear array and the square array respectively. The resolution of DOA estimations is measureed by the beamwidth of the subarrays. The beamwidth $b_{1,2}$ of two signals, with DOA at θ_1 and θ_2, arriving at an array with steering vector $\mathbf{v}(\theta)$ is evaluated by the following equation.

$$b_{1,2} = \frac{2}{\pi} cos^{-1}\left(\frac{|\mathbf{v}(\theta_1)^H \mathbf{v}(\theta_2)|}{|\mathbf{v}(\theta_1)||\mathbf{v}(\theta_2)|}\right) \tag{5.6}$$

We consider two narrow-band coherent signals $s_1(t) = e^{j2\pi ft}$, and $s_2(t) = e^{j2\pi f(t+\delta t)}$, with DOA's at 70^o and 85^o. where $f = 0.2$, $\delta t = 0.1$. The SNR is 20 dB. A total

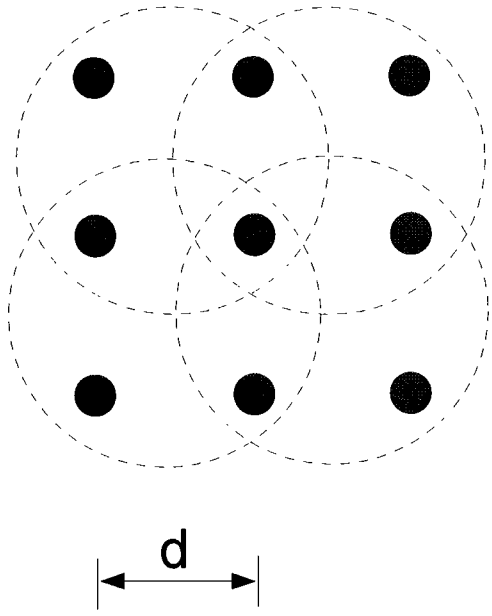


Figure 5.4: A nine-sensor square array with spacing d

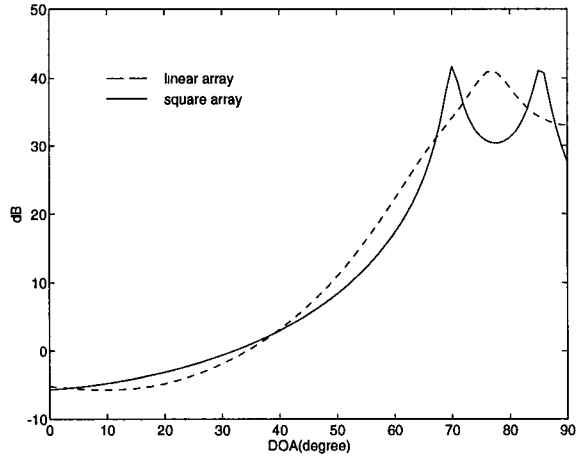


Figure 5.5: SS and MUSIC for DOA estimation of two coherent signals at 70^o and 85^o

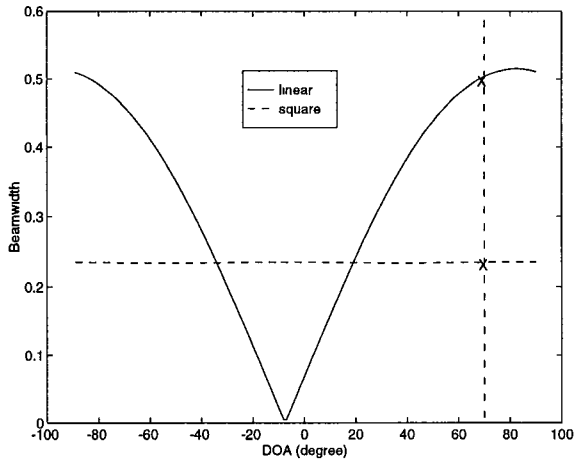


Figure 5.6: Beamwidth comparison between a six-sensor linear array and a four-sensor square array

of 500 samples ("snapshots") ($t = 1, \cdots, 500$) are taken from the array. We use SS as a pre-processing scheme for MUSIC. Fig.5.5 shows that the DOA's of the two coherent signals are not resolved using a linear array, whereas the square array gives a satisfactory result. In Fig. 5.6, we plot the beamwidths of two signals, that are 15^o apart, versus the angle of arrival of the signal that has a smaller DOA values. For the linear array, actual beamwidth varies greatly with the DOA. The square array, however, shows a relatively stable beamwidth for different DOA's. When DOA's are at 70^o and 85^o, the square array provides a narrower beamwidth and a higher resolution.

Example 5.3.2: We use the square array shown in Fig.5.4 to receive two coherent signals that are described in Example 1. with DOA's at 75^o and 100^o. The SNR is 20dB. A total of 500 samples are taken from the array each time. We apply SS and FBSS separately. Fig.5.7 shows that the DOA estimation resolution achieved

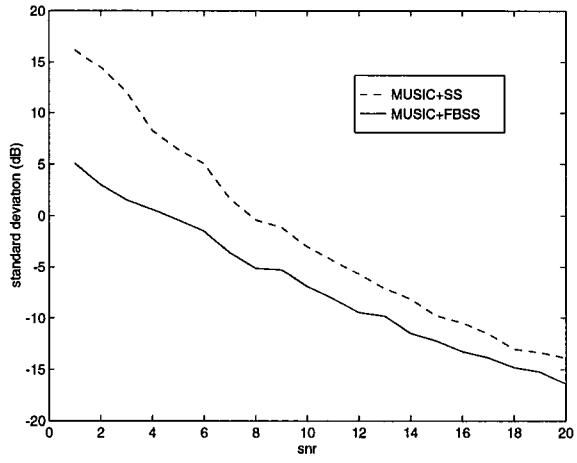


Figure 5.7: Standard deviation of DOA estimation using SS and FBSS

by a central symmetric array is significantly improved by using the FBSS method. The standard deviation is averaged over 200 estimated DOA's.

Example 5.3.3: We use a dense square array of sixty-four sensors as shown in Fig.5.1(b). The array contains 4 subarrays each of 49 sensors. The spacing between two neighboring sensors is 0.45λ. The array receives signals from four mobile users: four coherent signals at $20^o, 65^o, 150^o$ and 200^o from user 1, three at $230^o, 250^o$ and 280^o from user 2, two at 30^o and 300^o from user 3 and one at 320^o from user 4. The SNR is 20 dB. A total of 500 samples are used. First, we apply FBSS and then apply MUSIC. Simulation results are shown in Fig.5.8.

Example 5.3.4: We use the square array shown in Fig.5.4 to receive two coherent signals, one at an azimuth of 40^o and an elevation of 30^o, and the other at an azimuth of 50^o and an elevation of 60^o. The SNR is 20dB. There is a random phase delay from $(0, 2\pi)$ between these two signals at each snapshot. The number of samples taken is 500. By using FBSS and MUSIC, we obtain the result in Fig.5.9. It demonstrates that a planar array enables us to perform FBSS and

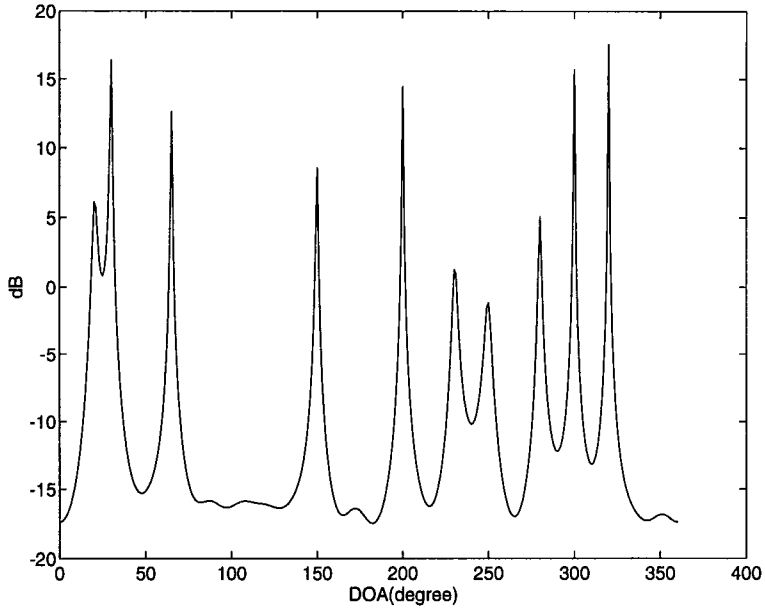


Figure 5.8: DOA estimation of four groups of coherent signals at (20^o,65^o,150^o,200^o), (230^o,250^o,280^o), (30^o,300^o) and 320^o based on a sixty four-sensor square array

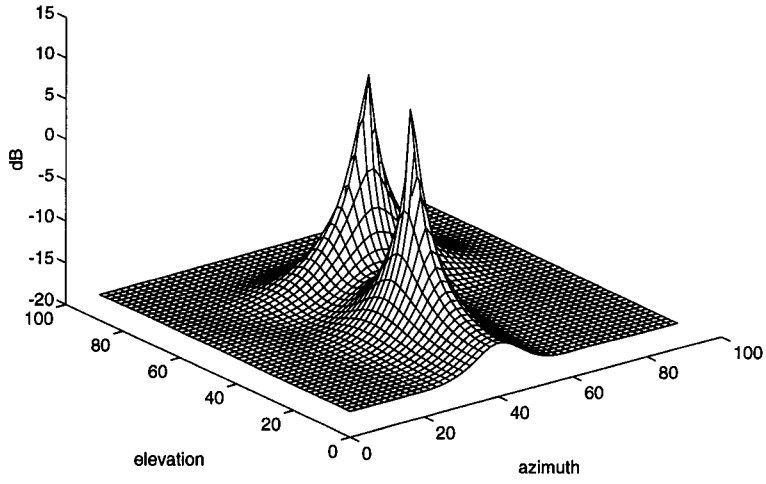


Figure 5.9: DOA estimation of two coherent signals at an azimuth of 40^o and an elevation of 30^o, and at an azimuth of 50^o and an elevation of 60^o, respectively

DOA's estimation in a 3D domain while a linear array or two crossed linear arrays are not capable of doing that.

Example 5.3.5: A twelve sensor array shown in Fig.5.10 is used in this example to receive two coherent signals at 70^o and 80^o. This array consists of two overlapping nine-sensor square arrays. Each sensor in one square array and its counterpart in another form a doublet pair. These nine doublet pairs form an array which has orientational invariance structure and is central symmetric. The spacing between two neighboring sensors is 0.45λ. The doublet spacing for ESPRIR is 0.45λ. The SNR is 20dB. A total of 2000 trials are run. A histogram of the results is given in Fig.5.11. We apply FBSS first and then applied the ESPRIT. The two angles are clearly identified.

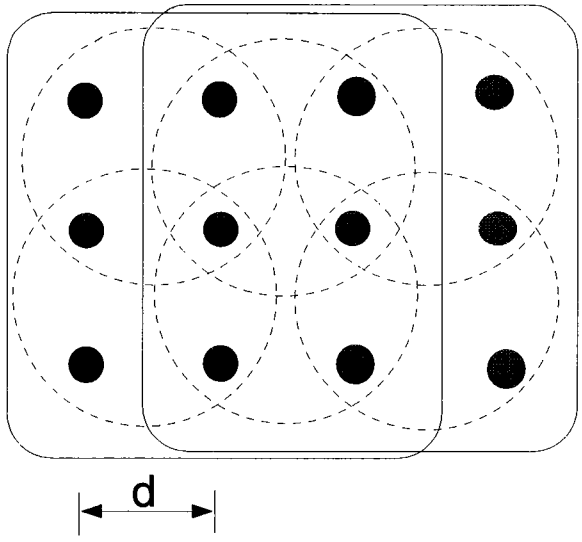


Figure 5.10: A twelve-sensor rectangle array with spacing d

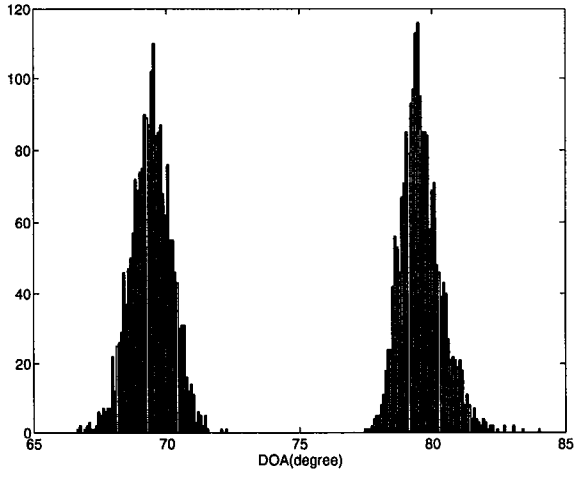


Figure 5.11: FBSS and ESPRIT for DOA estimation of two coherent signals at 70^o and 80^o

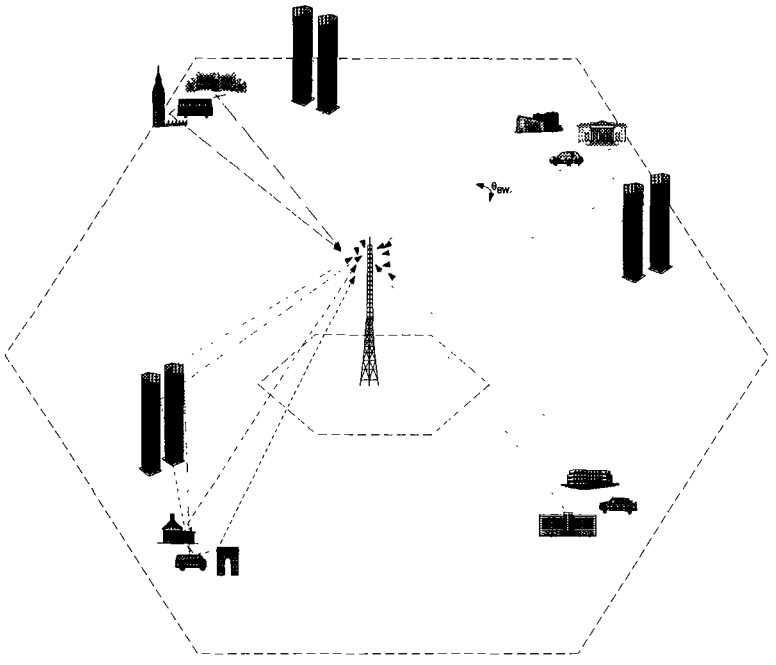


Figure 5.12: Multiuser Frequency Selective Channel

5.4 DOA Estimation in Multipath Fading and Cochannel Interference Environment

In a fading wireless channel, coherent signals are inevitable. Fig. 5.12 shows a typical example of the distribution of scatterers in a mobile radio environment [3]. We can treat all the reflected signals from local scatterers as coming from a super position. those reflected signals from far away scatterers such as high rise buildings or mountains are from another super position. Each group of local scatterers span a small angle with respect to the base station. There are many reflected signals within each small angle spread. These angles cannot be resolved even with high resolution MUSIC algorithm. They will be treated as one signal that suffers from flat fading. Thus there are seven angles to be estimated in Fig. 5.12. There are

two groups of coherent fading signals with three and two coherent signals in each. The others are noncoherent fading signals.

We use a dense square array of sixteen sensors as the base station antenna. The array contains 4 subarrays each of 9 sensors. The spacing between two neighboring sensors is 0.45λ. The wireless communication environment in Fig. 5.12 is simulated. The array receives signals from four mobile users. All of them move at 60miles/hr. The carrier frequency is 900 MHz. The modulated data rate is 24.3 ksym/s, which is the same as in IS-136 standard (an Electronic Industry Association Interim Standard). The maximum Doppler frequency $fd_{max} = 80Hz$. There are 40 reflected signals from each local scatterers or each distant scatterers. These signals uniformly span a 4^o angle. The combined signals to noise ratio is 10dB.

The received array data is given as

$$\mathbf{r}(t) = A * \mathbf{s}(t) + \mathbf{n}(t) \tag{5.7}$$

where the steering matrix $A = [A_1, A_2, \cdots, A_7]$; $A_i = [a_{i1}, a_{i2}, \cdots, a_{i40}]$.

$$a_{ij} = \begin{pmatrix} e^{-j2\pi(f_o+f_d)\cdot\frac{0.45\lambda}{c}} \\ \vdots \\ e^{-j2\pi(f_o+f_d)\cdot\frac{0.45\lambda}{c}\cdot(3\cdot cos(\theta_{i,j})} \\ \vdots \\ e^{-j2\pi(f_o+f_d)\cdot\frac{0.45\lambda}{c}\cdot(3\cdot sin(\theta_{i,j}))} \\ \vdots \\ e^{-j2\pi(f_o+f_d)\cdot\frac{0.45\lambda}{c}\cdot(3\cdot cos(\theta_{i,j})+3\cdot sin(\theta_{i,j}))} \end{pmatrix} \tag{5.8}$$

is a 16×1 steering vector.

$$\mathbf{s}(t) = [s_{1,1}(t), \cdots, s_{1,40}(t), s_{2,1}(t), \cdots, s_{2,40}(t), \cdots, s_{7,1}(t), \cdots, s_{7,40}(t)]^H \tag{5.9}$$

is a signal vector, $s_{n,k}(t)$ is the kth reflected signal from the nth group of scatterers.

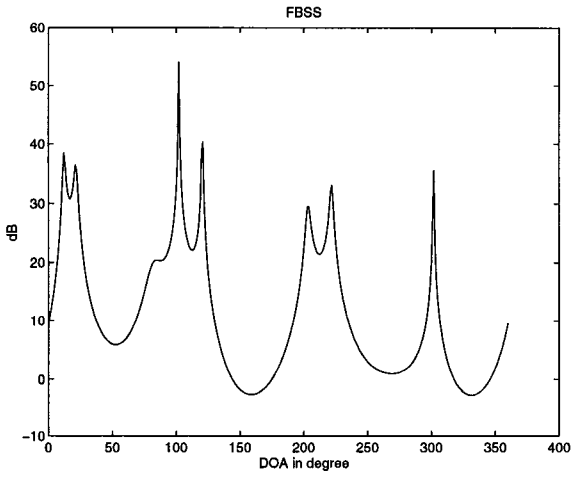


Figure 5.13: DOA estimation of four groups of coherent signals at (10^o,20^o,100^o), (120^o), (200^o,220^o) and (300^o) based on a sixteen sensor square array

$$s_{n,k}(t) = \frac{1}{40} e^{j2\pi(fd_{n,k} \cdot t + \phi_{n,k})} \tag{5.10}$$

The phase distorsion $\phi_{n,k}$ is uniformly distributed in $[0, 2\pi]$. The doppler frequency $fd_{n,k} = fd_{max} \times cos(\frac{2\pi(k-1)}{40})$, $k = 1, \cdots 40$.

$$\mathbf{n}(t) = [n_1(t), \cdots, n_{16}(t)]^H \tag{5.11}$$

is a noise vector.

Reflected signal for user 1 is from 10^o to 14^o, 20^o to 24^o and 100^o to 104^o. Reflected signal for user 2 is from 120^o to 124^o. Reflected signal for user 3 is from 200^o to 204^o and 220^o to 224^o. Reflected signal for user 4 is from 300^o to 304^o.

A total of 162 samples are used. We apply FBSS first and then MUSIC. Simulation results are shown in Fig. 5.13. According to the DOA's information, we can further perform constrained beamforming and thus can achieve Spatial Division

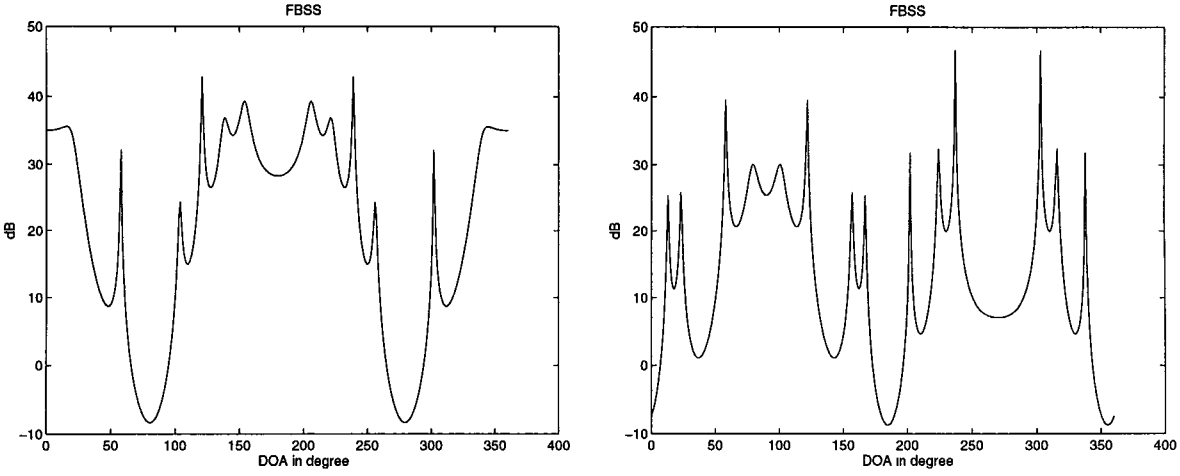
Multiple Access(SDMA) [12] in a multipath environment.

A square array is efficient in terms of the sensor reuse rate. One can also use any other kind of planar arrays which satisfy the necessary and sufficient conditions stated in chapter III.

An alternative way of using planar arrays to cover all the azimuth angles is to use two crossed uniformly spaced linear arrays. Spatial smoothing and DOA's estimation can be done on each linear array. Ambiguities related to linear arrays can be largely avoided by searching for common estimated DOA's from both arrays. It is effective when there are only a few DOA's and these DOA's are not close to the end fire of either of the linear arrays. For example, to resolve two coherent signals, we need two linear arrays each with four sensors. Using square arrays, we need nine sensors. The complexity is reduced by using two crossed linear arrays. Better DOA's estimation resolution can be achieved for some angles as shown in Fig. 5.14. But we will show in the following that in wireless communication environments when there are a lot of DOA's, the use of two crossed linear arrays may increase the complexity and may cause ambiguities for signal incoming angles close to the end fire of either of the linear arrays.

In the previews example, if we rearrange the sixteen sensors to two crossed linear arrays, each of eight sensors, we will not be able to perform FBSS on each linear array to estimate seven incoming signals. The condition that the number of sensors in each subarray is larger than the number of incoming signals was not satisfied.

We increased the number of sensors. We used two crossed linear arrays each of nine sensors. We divided each linear array to two subarrays each of eight sensors. The spacing between two neighboring sensors is 0.45λ. We use these two crossed



(a) horizontally positioned linear array (b) vertically positioned linear array

Figure 5.14: DOA estimation of four groups of coherent signals at $(10^o,20^o,100^o), (120^o)$, $(200^o,220^o)$ and (300^o) based on a two acrossed linear arrays

linear array to received the same transmitted signals described in Example 3. Fig. 5.14 (a) shows the DOA estimation using a horizontally positioned linear array. For incoming signals close to the array end fire, the DOA estimation resolution is poor. Two signals at 10^o and 20^o cannot be resolved. Fig. 5.14 (b) shows the result of DOA estimation using vertically positioned linear array. The final decisions on the DOA's is decided based upon the common peaks in these two figures. The unresolved angles at 10^o and 20^o in Fig 5.14 (a) make the DOA's estimation difficult. For those incoming signals that are not close to array end fire, we can get more accurate DOA's estimation. The DOA's estimation resolution of two acrossed linear arrays is sensitive to the DOA's. This approach also increases the computational complexity. We need to search for peaks of DOA's twice. Therefore, compared to the square array used in Example 3, the two crossed linear arrays require high hardware and computational complexity.

5.5 Capacity Increase in TDMA Wireless System with Constrained Adaptive Beamforming

This method is based on high-resolution direction finding followed by optimal combination of the antenna outputs. The basic idea is to constrain the response of array beamformers so that signals from the direction of interest are passed with specified gain. The weights are chosen to minimize array output power under these constraints. For example, to get the cochannel signals from different directions A, B and C separately, we use three beamformers. We constrain the response of the

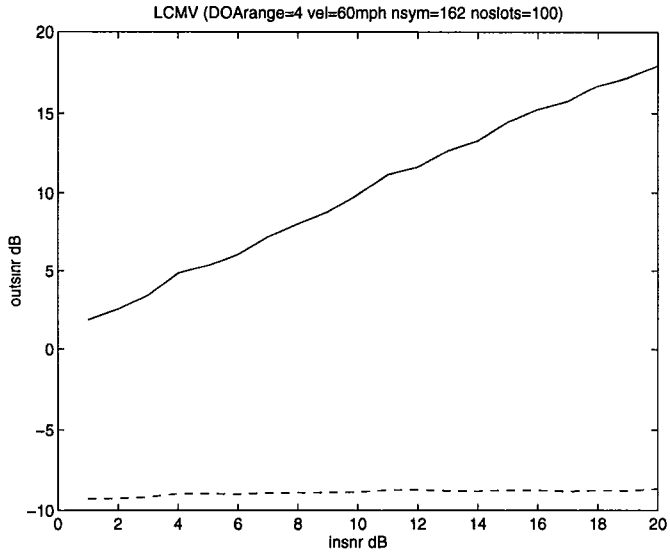


Figure 5.15: Interference suppression with LCMV

first beamformer so that signals from the direction A are passed with no attenuation while signals from directions B, C and noise arriving from other directions are minimized. Similarly, we can get signals from B and C simultaneously by using two other beamformers. In such a case, we achieved 3-fold capacity increase. By actively tracking mobile units and directionally transmitting information to and receiving information from these units, significant capacity increases are achieved without any modification to the mobile units.

To reduce multipath fading, we use angle diversity. We optimally combine the reflected signals from both distant scatterers and local scatterers.

Fig. 5.15 shows the result of LCMV. Weights are chosen such that the signals at one desired direction are passed and all the signals at the other interference directions are all suppressed, and the signal energy at the rest angles are all minimized. We observed that output SNR is almost equal to the input SNR. All the

interferences are successfully suppressed.

To suppress interference at the mobile handset, we use data independent beamforming algorithms in the transmitter beamformer. The weights in a data independent beamformer are designed so that the beamformer response approximates a desired response independent of the array data or data statistics. In a slow fading channel where all users are still or moving slowly, the beamforming weights in a beamformer of transmitter can be chosen to be the complex conjugate of the receiver beamformer weights if the forward and the downward link share the same frequency band. When mobile terminals move at some fast speeds, we try to steer the main beam lobe of each beamformer to each mobile and minimize sidelobe at other direction. So we chooses the weights of each beamformer to be the steering vector corresponding to the DOA of each targeted mobile. This results in increased gain in the direction of the mobile and decreased gain in the direction of any interferers. Thus, cochannel interference from the base station to mobiles in other cells is reduced, and each mobile receives its own signal with very little cochannel interference.

Chapter 6

Adaptive Array in CDMA

6.1 Introduction

In the previous chapters, we presented the applications of adaptive arrays in TDMA mobile communication system. In this chapter, we will investigate their applications in CDMA systems.

In (DS) CDMA spread spectrum systems, the same frequency band is shared by many users. The receiver uses a matched filter corresponding to the user's spreading chip sequence to extract the desired signal components from the composite signal comprised of the desired interference signals.

CDMA system possesses many advantages such as security, resistance to interference, but it also has an inherent disadvantage of having the near-far problem. Overcoming this problem holds the key to achieving high-capacity CDMA mobile communication systems. Power control is often used to solve this problem, but the rapid fluctuation in the signal strength caused by multipath Rayleigh fading remains in spite of power control. A hybrid diversity scheme combining space diversity and path diversity [67] provides an effective way to reduce of fast fading

effect and to further reduce the near-far effect. Conventionally, equal gain combining is used to combine diversity paths. However, when there are remaining interferences after despreading, caused by poor power control or poor synchronization, then equal gain combining degrades the performance [68]. In such situations, an adaptive diversity combining technique that can suppress residual interference and reduce fast fading is beneficial.

In this chapter, we propose a diversity combining technique with weights obtained based on previous information along surviving paths of an orthogonal convolutional code. We develop the reference signal based diversity combining techniques for O-CDMA system with orthogonal Hadamard-Walsh (HW) codes to suppress remaining interferences after demodulation.

6.2 DS/CDMA Mobile Signal with HW Modulation

In the proposed CDMA standard IS-95, the downlink employs coherent demodulation with assistance of pilot signals. Multiple transmitted signals are synchronously combined. Its performance in a single cell system will be much superior to that of the reverse link. No power control is required, since for each subscriber, any interference caused by other subscriber's signals remains at the same level relative to the desired signal. There are little degradations due to fading.

In this work, we will only consider the diversity combining in the most difficult uplink. In the following, we will briefly present the signal characteristic at the uplink.

Orthogonal Hardmard-Walsh modulations are used on the uplink in IS-95. The

baseband equivalent of the $l-th$ DS/SS K-ary Hardmard-Walsh(HW) modulated user signal can be expressed as follows:

$$\tilde{s}^{(k)}(t) = \alpha^k(t)\sqrt{E_c^{(k)}} \sum_{i=-\infty}^{\infty} (c_{p,|i|_L}^{(k)} + c_{q,|i|_L}^{(l)}) \cdot d_h^{(k)} \cdot g_T(t - iT_c) \tag{6.1}$$

where $|i|_L \triangleq i \bmod L$, the superscript $\tilde{}$ denotes a complex-valued signal, $E_c^{(k)}$ is the average transmitted power of the k-th user, α is Rayleigh distributed, $g_T(t)$ is the impulse response of the transmit filter, $c_{p,|i|_L}^{(k)}$ and $c_{q,|i|_L}^{(k)} \in \pm 1$ are the so called chips of the spreading sequence (with i ticking at the chip rate $1/T_c$), $d_h^{(k)} \in \pm 1$ is the h-th HW binary symbols (with h ticking at the HW symbol rate H/T_s), L is the spreading code repetition period. The chip pulse shape $g_T(t)$ may be a standard square-root Nyquist raised-cosine with roll-off factor α with respect to the chip signaling rate $1/T_c$. The baseband equivalent of the DS/SS-CDMA signal received is thus a multiplex of K different signals in the form of Eq. 6.1 as follows(no channel distortion is assumed):

$$\begin{aligned}
\tilde{r}(t) &= \sum_{k=1}^{K} \tilde{s}^{(k)}(t - \tau_k) e^{j(\theta_k + \Delta w_k t)} + \tilde{\nu}(t) && (6.2)\\
&= \sum_{k=1}^{K} \alpha^k(t)\sqrt{E_c^{(k)}} e^{j(\theta_k + \Delta w_k t)} && (6.3)\\
&\quad \sum_{i=-\infty}^{\infty} (c_{p,|i|_L}^{(k)} + c_{q,|i|_L}^{(k)}) \cdot d_h^{(k)} g_T(t - iT_c - \tau_k) + \tilde{\nu}(t) && (6.4)
\end{aligned}$$

where τ_k and θ_k are the unknown delay and carrier phase shift, respectively, and Δw_k is the carrier frequency offset of the local oscillator. The white Gaussian noise process $\tilde{\nu}(t) \triangleq \nu_p(t) + j\nu_q(t)$ has one-sided power spectral density I_0.

Let us now look at the signals after the demodulator. After noncoherent I and Q demodulation and chip matched filtering, the basedband components of received signal are correlated each with both I and Q signature codes and appropriately combined, to give a noise-corrupted replica of the generic binary symbol

of currently transmitted HW function. Those samples are taken H at a time and fed to correlate with each of the possible HW functions. The observable vectors for in phase and quadrature outputs may be taken as $\mathbf{y}^I = (y_1^{(I)}, y_2^{(I)}, \cdots, y_N^{(I)})$ and $\mathbf{y}^Q = (y_1^{(Q)}, y_2^{(Q)}, \cdots, y_N^{(Q)})$. $N = T_s/T_c$ is the spreading factor. When a HW function corresponding to a correct signal is used,

$$p_C(\mathbf{y}^{(I)}, \mathbf{y}^{(Q)}|\alpha, \phi) = \prod_{n=1}^{N} exp[-(y_n^{(I)}) - \alpha\sqrt{E_c}cos\phi)^2/I_0]exp[-(y_n^{(Q)} - \alpha\sqrt{E_c}sin\phi)^2/I_0]/\pi I_0 \tag{6.5}$$

Otherwise

$$p_I(\mathbf{y}^{(I)}, \mathbf{y}^{(Q)}) = \prod_{n=1}^{N} exp[-(y_n^{(I)})^2/I_0]exp[-(y_n^{(Q)})^2/I_0]/\pi I_0 \tag{6.6}$$

Each square modulus Z defined by $Z = Y^{(I)^2} + Y^{(Q)^2}$ is then calculated to get rid of phase errors, where $Y^{(I)} = \sum_{n=1}^{N} y_n^{(I)}$, $Y^{(Q)} = \sum_{n=1}^{N} y_n^{(Q)}$ The estimated HW sequence is the one with the largest Z. The probability density function $p_C(Z)$ of Z refers to the presence of a (correct) signal in the given correlator and is given by

$$p_C(Z) = \int_0^{\infty} \frac{2\alpha}{\sigma^2} e^{-\frac{\alpha^2}{\sigma^2}} \frac{exp(-(Z + \alpha^2 M^2)/V)}{V} \varphi_0(\frac{2\sqrt{\alpha^2 M^2 Z}}{V}) d\alpha \tag{6.7}$$

$$= \frac{1}{V_F} e^{-Z/V_F} \tag{6.8}$$

The probability density function $p_I(Z)$ refers to the absence of a (correct) signal and is given by

$$p_I(Z) = \frac{1}{V} e^{-Z/V} \tag{6.9}$$

$V = NI_0$, $M^2 = N^2 E_c$, $V_F = V + M^2\sigma^2$

More generally, there are L independent multipaths. When these paths have independent Rayleigh fading with equal-average energy, equal gain combining is usually used. The probability density function of Z which is the sum of the squares

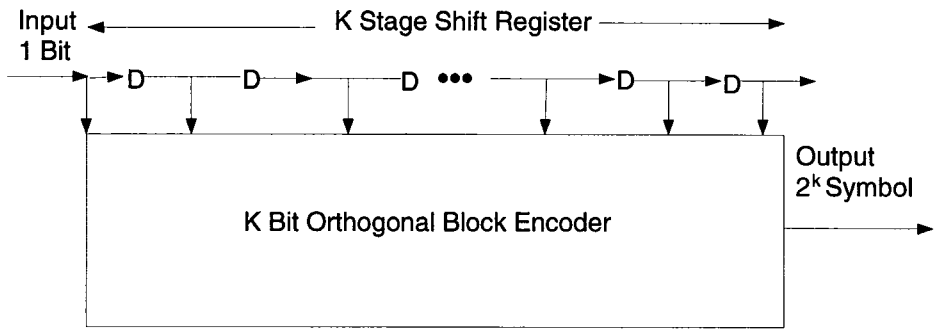


Figure 6.1: Orthogonal convolutional encoder

of L independent Rayleigh observations, is given as follows: (i) when signal is present,

$$p_C(Z) = \frac{z^{L-1}e^{-Z/V_F}}{(L-1)!V_F^L} \tag{6.10}$$

(ii) when signal is not present,

$$p_I(Z) = \frac{z^{L-1}e^{-Z/V}}{(L-1)!V^L} \tag{6.11}$$

6.3 Multipath Selection Based on Orthogonal Convolutional Code

If the orthogonal convolutional code shown in Fig. 6.1 is used, we can achieve better performance by using combining based on the predicted weights than by using equal gain combining. The orthogonal convolutional encoder of Fig. 6.1 produces for each bit time one of $H = 2^K$ Hardmard-Walsh sequences of length 2^K, $X_1, X_2, \cdots, X_H$. Fig. 6.2 shows an example of paths of the code trellis diagram over their unmerged span, for $K = 6$, $H = 64$, labeled according to the Hardmard-Walsh sequence generated for each branch. In a slowly fading situation, on a correct path marked by solid line in Fig. 6.2, the noncoherent demodulator output Z at

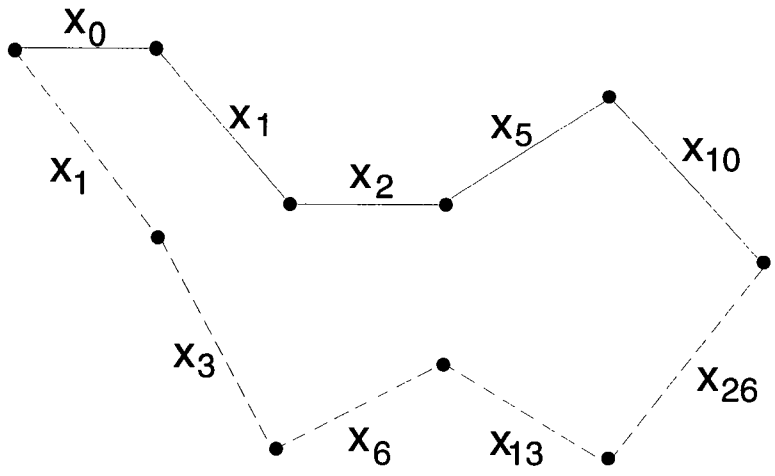


Figure 6.2: Example of two unmerged paths of orthogonal convolutional code

two consecutive branches are highly correlated. Along the correct path, when the desired signal from the *ith* fading path suffers from deep fading, it will still suffer from deep fading at the next branch. The signal energy distribution on a correct branch can be predicted by the noncoherent demodulator output at the previous branches. In contrast, on a wrong path marked by dashed line, the demodulator output at two consecutive branches are uncorrelated.

Suppose that an incorrect path is unmerged from the correct path for d branches, in which case all corresponding branch pairs are orthogonal. Let the branch metrics on the correct and incorrect paths be denoted by $Z_1, Z_2, \cdots, Z_d$ and $Z'_1, Z'_2, \cdots, Z'_d$ respectively.

When equal gain combining is used, the probability of pairwise error upon remerging is just

$$P_d = Pr(\sum_{i=1}^{d} Z'_i > \sum_{i=1}^{d} Z_i). \tag{6.12}$$

We then have the Chernoff bound,

$$P_d < E[e^{\rho \sum_{i=1}^{d}(Z_i' - Z_i)}] = (E[e^{\rho(Z'-Z)}])^d, \tag{6.13}$$

with

$$E[e^{\rho(Z'-Z)}] = \int_0^\infty e^{\rho Z'} p_I(Z') dZ' \int_0^\infty e^{-\rho Z} p_C(Z) dZ \tag{6.14}$$

$$= \frac{1}{(1-\rho)^L [1+\rho(1+U)]^L}, \ \rho > 0. \tag{6.15}$$

where $U = (\sigma^2 E_b/I_0)/L = (\bar{E}_b/I_0)/L$. The minimum with respect to ρ occurs at $\rho = (U/2)/(1+U)$. Therefore,

$$P_d < W_0^d, \tag{6.16}$$

where

$$W_0 = [\frac{1+U}{(1+U/2)^2}]^L, \tag{6.17}$$

The excess of E_b/I_0 over $2ln(1/W_0)$ for this noncoherent L-path Rayleigh channel is

$$\frac{\bar{E}_b/I_0}{2ln(1/W_0)} = \frac{U/2}{ln[(1+U/2)^2(1+U)]}, \tag{6.18}$$

$$\frac{2ln(1/W_0)}{L} = 2ln[(1+U/2)^2/(1+U)]. \tag{6.19}$$

This excess of $\bar{E}_b/I_0$ over $2ln(1/W_0)$ in decibel is plotted as a function of $2ln(1/W_0)/L$ in decibels in Fig. 6.3.

In our effort to improve combining performance, we assume that, on a correct path, we can predict the path of the strongest received signal energy. The probability function of the largest square modulus Z is given by

$$Pr_C(max\{Z_i, i = 1, \cdots, L\} < Z) = [\int_0^Z \frac{1}{V_F} e^{-z/V_F} dz]^L \tag{6.20}$$

Its density function is given by

$$p_C(Z) = L[\int_0^Z \frac{1}{V_F} e^{-z/V_F} dz]^{L-1} \frac{1}{V_F} e^{-Z/V_F} \quad (6.21)$$

$$= L[1 - e^{-Z/V_F}]^{L-1} \frac{e^{-Z/V_F}}{V_F} \quad (6.22)$$

$$= L[1 - e^{-Z/(1+U)}]^{L-1} \frac{e^{-Z/(1+U)}}{1+U} \quad (6.23)$$

Then we obtain the Chernoff bound:

$$P_d = Pr(\sum_{i=1}^{d} Z_i' > \sum_{i=1}^{d} Z_i) \quad (6.24)$$

$$< E[e^{\rho \sum_{i=1}^{d} (Z_i' - Z_i)}] = (E[e^{\rho(Z'-Z)}])^d, \quad (6.25)$$

with

$$E[e^{\rho(Z'-Z)}] = \int_0^\infty e^{\rho Z'} p_I(Z') dZ' \int_0^\infty e^{-\rho Z} p_C(Z) dZ \quad (6.26)$$

$$= \int_0^\infty e^{(\rho-1)Z'} dZ' \int_0^\infty \frac{L[1 - e^{-Z/(1+U)}]^{L-1}}{1+U} e^{-Z[\rho + \frac{1}{1+U}]} dZ \quad (6.27)$$

$$= \frac{1}{(1-\rho)} \frac{L!}{\prod_{i=1}^{L} [i + \rho(1+U)]} \quad (6.28)$$

$$= W_0. \quad (6.29)$$

The corresponding required excess E_b/I_0 in decibel is plotted in dashed line in Fig. 6.3. We observe a reduced E_b/I_0 compared to equal gain combining.

6.4 Weighted Diversity Combining

In practice, due to the noise effect, we may not always know the diversity path with the strongest signal every time. We can nevertheless predict the suitable weights for the path diversity combining at next branch based on orthogonal convolutional code.

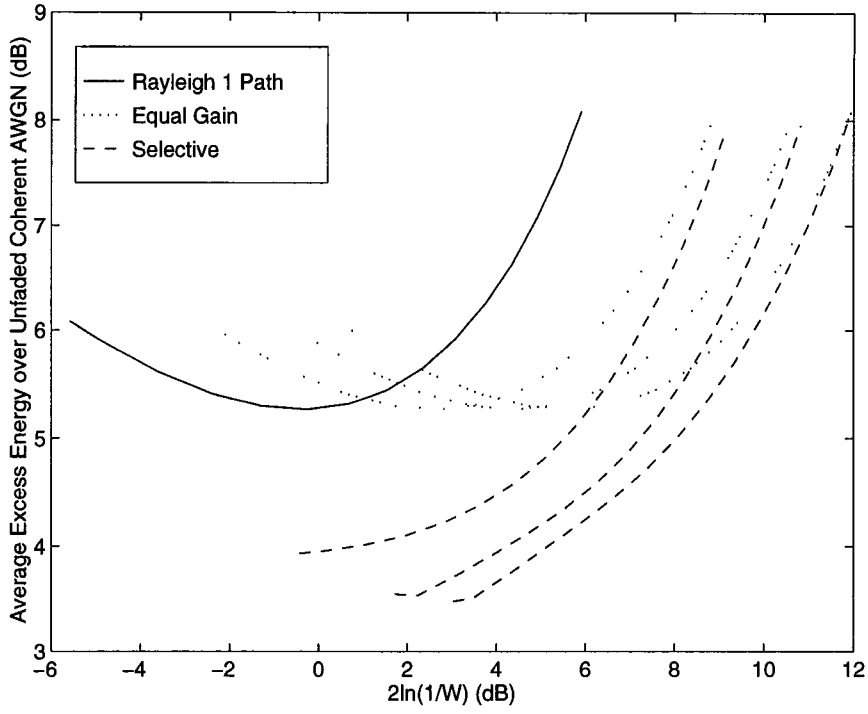


Figure 6.3: Required excess E_b/I_0 for noncoherent Rayleigh fading with orthogonal convolutional codes

In the case when we have perfect synchronization and power control, the signals after CMF consist of the desired signal, Gaussian noise, and summation of many residual interference signals with low energy. The summation of all these interference signals can be approximated by Gaussian noise. The characteristic of noncoherent demodulator output has been analyzed in the previous sections. The weights prediction algorithm in this case is given in the following.

- We update the weights along each surviving path in the code trellis diagram.

- On each path, the weights are updated based upon updated smoothed branch metrics. The reason to use the smoothed branch metric is to reduce the Gaussian noise effect. We define the smoothed branch metric B_l^s of the sth surviving path as follows:

$$B_l^s = \sum_{i=1}^{n-1} \lambda^{n-i} Z_l^s(n-i) + Z_l^s(n) \tag{6.30}$$

  l denotes the lth diversity channel, n denotes the nth symbol period and λ is a forgetting factor. The predicted weights for the next connected branch is given as follows

$$w_l^s = \frac{B_l^s}{\sum_{l=1}^{L} B_l^s}. \tag{6.31}$$

  Along a correct path, the weights approximately reflect the noncoherent demodulator outputs from different fading paths at the next branch. On the other hand, along a wrong path, such weights do not provide any information about the noncoherent outputs at the next branch.

- We then correlate the weighting vector with the noncoherent demodulator output vector $\mathbf{Z}$, $\mathbf{Z} = [Z_1, Z_2, \cdots Z_L]$ (L is the total fading paths) at the next

branch, and use the correlated result as the final branch metric. Use the final branch metric to update the accumulated path metric as follows

$$P^s(n) = \sum_{i=1}^{n-1} (\mathbf{W}^s(n-i-1))^T \cdot \mathbf{Z}^s(n-i) + (\mathbf{W}^s(n-1))^T \mathbf{Z}^s(n) \qquad (6.32)$$

- Select the surviving path from the code trellis diagram based on the accumulated path metrics obtained in the above step.

The accumulated path metrics calculated based on equal gain combining is given by

$$P^s(n) = \sum_{i=1}^{n-1} (\sum_{l=1}^{L} Z_l^s(n-i)) + \sum_{l=1}^{L} Z_l^s(n) \qquad (6.33)$$

P^s obtained from (6.32) is enlarged compared to that obtained from (6.33), on a correct path, and is reduced on a wrong path. Therefore, we increase the difference between the path metric of a correct path and that of a wrong path. This results in a reduced BER as shown in the following example.

In our simulation example, the channel is code-division-multiplexed. There are 384 symbols in each frame. The symbol rate is 19.2ksym/s. The carrier frequency is 17 GHz. There are totally eight diversity paths (including both path diversity and antenna diversity) The desired user is moving at 5miles/hr. Orthogonal convolutional code and Hardmard Walsh modulation($K = 3$, $H = 8$) are used. Perfect synchronization and effective power control are assumed. After CMF, the signal to noise ratio is 6dB. $\lambda = 0.8$. In Fig. 6.4 the BER obtained with predicted weights is compared to that with equal gain combining. We observe 0.2 to 0.3dB SNR improvements from the use of predicted weights.

In the case when there is no effective power control and no perfect synchronization, after noncoherent demodulation, there will be some strong interference signals remaining in some diversity paths. Similar to the diversity combining in

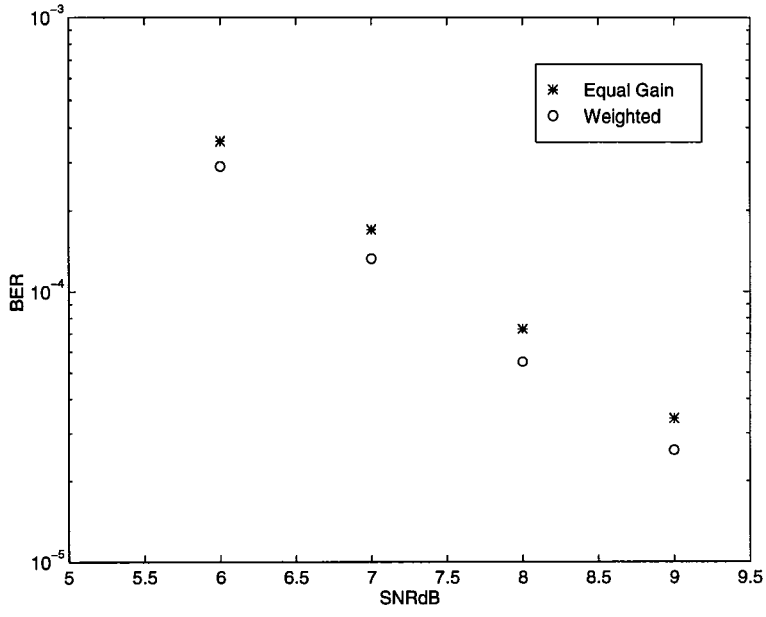


Figure 6.4: BER comparison between equal gain combining and combining based on the predicted weights

TDMA, the desired signal can not be extracted from strong interferences and noise without using reference signals. Here, we also need to insert training sequences at the begining of each frame and use them as the reference signals to get an initial estimate of the multipath weights. The use of orthogonal convolutional codes allows us to further update the weights to track the fading variations in the channel. Without the use of orthogonal convolutional code, we have to use the initial weights for the rest of the diversity combining for the whole frame period. To prevent weights from converging along a wrong path which is associated with an interfering signal sequence, weights have to be updated under certain constraints. As we have explained previously, the initial estimated weights should be highly correlated with the noncoherent demodulator output vector at the next branch along a correct path and not correlated along a wrong path. Therefore, we evaluate this correlation first on each path before updating the weights. Our algorithm with the use of reference signals is given below:

- We normalize Z_i as follows:

$$z_i^s(n) = \frac{Z_i^s(n)}{\sum_{i=1}^{L} Z_i^s(n)}. \tag{6.34}$$

- We then find the correlation coefficient ρ between $\mathbf{z}$ and $\mathbf{w}(n-1)$ obtained in Eq. 6.31,

$$\rho = \frac{\sum_{i=1}^{L} w_i^s(n-1) z_i^s(n)}{\sqrt{\sum_{i=1}^{L} (w_i^s(n-1))^2} \sqrt{\sum_{i=1}^{L} (z_i^s(n))^2}}. \tag{6.35}$$

- On the sth path, if ρ is greater than a preset Threshold, we update the weights following the procedure in the previous algorithm. Otherwise, we keep the previous weights.

In the following simulation example, we assume five strong interference signals remain after CMF, each's $INR = -3 \sim 1dB$. $SNR = 6dB$, Threshold is set

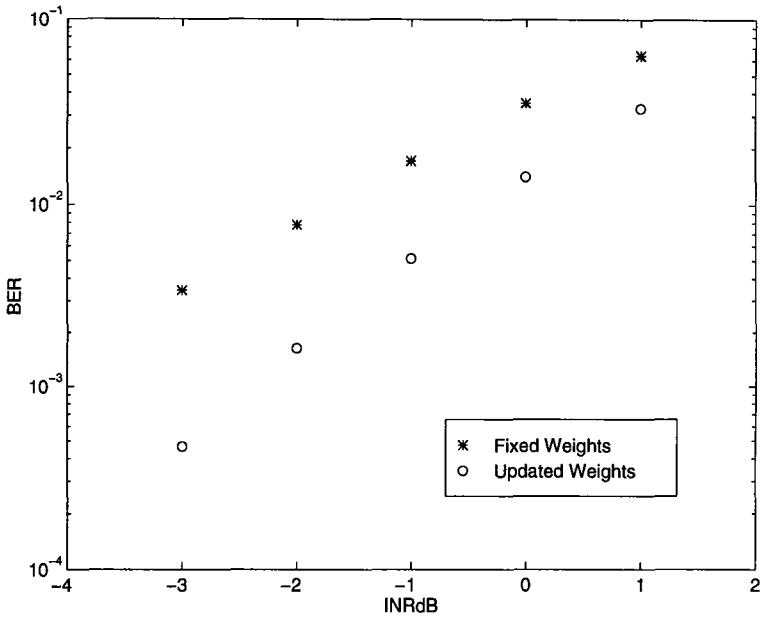


Figure 6.5: BER comparison between diversity combining with fixed weights and predicted weights

to be 0.9. $\lambda = 0.8$. The channel conditions are the same as in the previous example except that all the desired vehicle and interfering vehicles are moving at 10 miles/hr. Fig. 6.5 shows $1 \sim 1.5$dB improvement in interference suppression using adapted weights over using fixed weights obtained in the training mode.

Chapter 7

Conclusions and Future Research

Our dissertation is devoted to applying the adaptive array signal processing techniques to wireless communications to increase channel capacity. We identified and addressed issues related to two major kinds of adaptive arrays suitable for TDMA systems. One is optimum adaptive diversity combining based on reference signals and the other is constrained adaptive beamforming based on the DOAs.

From computer simulation studies, we discovered that the effectiveness of reference signal based array combining on a fast time-varying channel depends on the tracking speed of the adaptive signal processing algorithm used and on the level of a decision error propagation.

To reduce the decision error propagation in the conventional optimum diversity combining system, we developed a simultaneous diversity weights updating and decoding technique which incorporates QRD-RLS based parallel weights tracking and a M-D decoding algorithm. The M-D decoding algorithm was first developed for the binary convolutional codes and then extended to TCM codes. It provides instantaneously a set of candidate reference signals for weights tracking and makes a final symbol decision with a D symbol delay based on more reliable accumulated

path metrics. The memory required by the M-D algorithm is only O(M). The computational complexity required is $O(K^2 + M \cdot K)$ for K weights updating and $O(M)$ for decoding, which is not much increased from $O(K^2)$ in the conventional decision directed adaptive array system when M is smaller than K. Simulation results showed that about 8 to 10dB improvement in total interference suppression at low ISR and about 3 to 5dB improvement at high ISR can be achieved in our system using M-D algorithm with $M = 2$. This system significantly reduces error propagation in the decision directed array systems with moderate increase in complexity.

To find the best weights acquisition and tracking algorithm, we conducted comparisons on applicable adaptive array algorithms. We found that QRD-RLS algorithm has good numerical stability, can double the tolerable interference level compared to the DMI or other RLS algorithms if a high precision A/D converter is available. Its complexity is no more than that of the others and is within the capability of a modern programmable DSP processor. We revealed an unstability problem caused by a low bits A/D converter, and developed a new exact initialization approach for the QRD-RLS algorithm to solve the problem.

To effectively use DOA based adaptive array in a multipath TDMA mobile radio channel, a robust DOA estimation algorithm applicable to coherent signals is crucial. We developed a general SS technique for two dimensional arrays to decorrelate coherent signals, and to make MUSIC, ESPRIT and adaptive beamformers operative in a coherent interference environment. In order to apply SS to an array of nonlinear geometry, this array must have an orientational invariance structure and its center array must be ambiguity free. Also the number of subarrays must be greater than or equal to the largest number of mutually coherent signals. To

apply SS in conjunction with MUSIC, all the subarrays must also be ambiguity free, and the number of sensors in each subarrays must be larger than the number of incoming signals. For ESPRIT, two identical arrays (or subarrays) separated by a displacement vector are used each satisfying the conditions for applying SS and MUSIC. When a nonlinear array is central symmetric, the FBSS can be used and it outperforms the regular SS in terms of improved efficiency and estimation resolution.

We proved the necessary and sufficient conditions for a three-sensor array manifold to be ambiguity free. We identified several situations, for higher order sensor array manifolds, in which ambiguity may arise. It is necessary to avoid the identified ambiguities in designing ambiguity free center arrays and subarrays.

We conducted simulations to verify the effectiveness of our SS in decorrelating multipath coherent signals in TDMA wireless communications environment. The successful interference suppressing and capacity increase based on SS, high resolution DOA estimation and LCMV beamforming have also been verified by computer simulation.

The combined diversity weights tracking and decoding previously developed for TDMA system is applied to DS/CDMA system. We proposed a diversity combining technique using predicted weights and orthogonal convolutional code. Better BER performance can be achieved compared to conventional equal gain combining especially when there is lack of perfect synchronization and power control.

With regard to possible future research direction, the simultaneous weights updating and decoding may be extended to various decision directed algorithms for tracking time-varying environment with cochannel interference, for example, the application to decision feedback equalization(DFE). In the frequency selective

fading environment, fading, intersymbol interference and cochannel interference occur at the same time. Diversity-DFE techniques [70] [72] [73] have been proposed to improve the channel conditions. Error propagation in DFE can be even more severe. If we make a decision error, not only the reference signals but also the input signals are erroneous in the next equalizer weights updating. Bidirectional equalization has been proposed to reduce error propagation, but this method is valid only for those TDMA systems that have reference signals at both the begining and the end of each time slot. A precoding technique [71] has been proposed to reduce error propagation in DFE, but this technique is only applicable when the actual system response varies moderately. Reducing error propagation in DFE under fast time-varying frequency select environment should be an interesting and research rewarding topic.

Other possible applications of this combined technique include joint phase synchronization and decoding, joint channel and data estimation.

Several interesting research topics can be pursued in CDMA.

For DPSK DS/CDMA system, an adaptive optimum path diversity combining scheme is proposed in [68] to reduce interference only path effect. RLS was used for diversity weights tracking. We can apply simultaneous optimum diversity weights tracking and decoding in this system to reduce error propagation. Fig. 7.1 is a proposed block diagram. More quantitative studies in this area are needed.

Adaptive linear and decision feedback receiver structures for coherent demodulation in asynchronous code division multiple access (CDMA) system were considered in [69]. The error propagation in the decision feedback receiver may be even more severe. The combined equalizer weights updating and decoding can also be applied here.

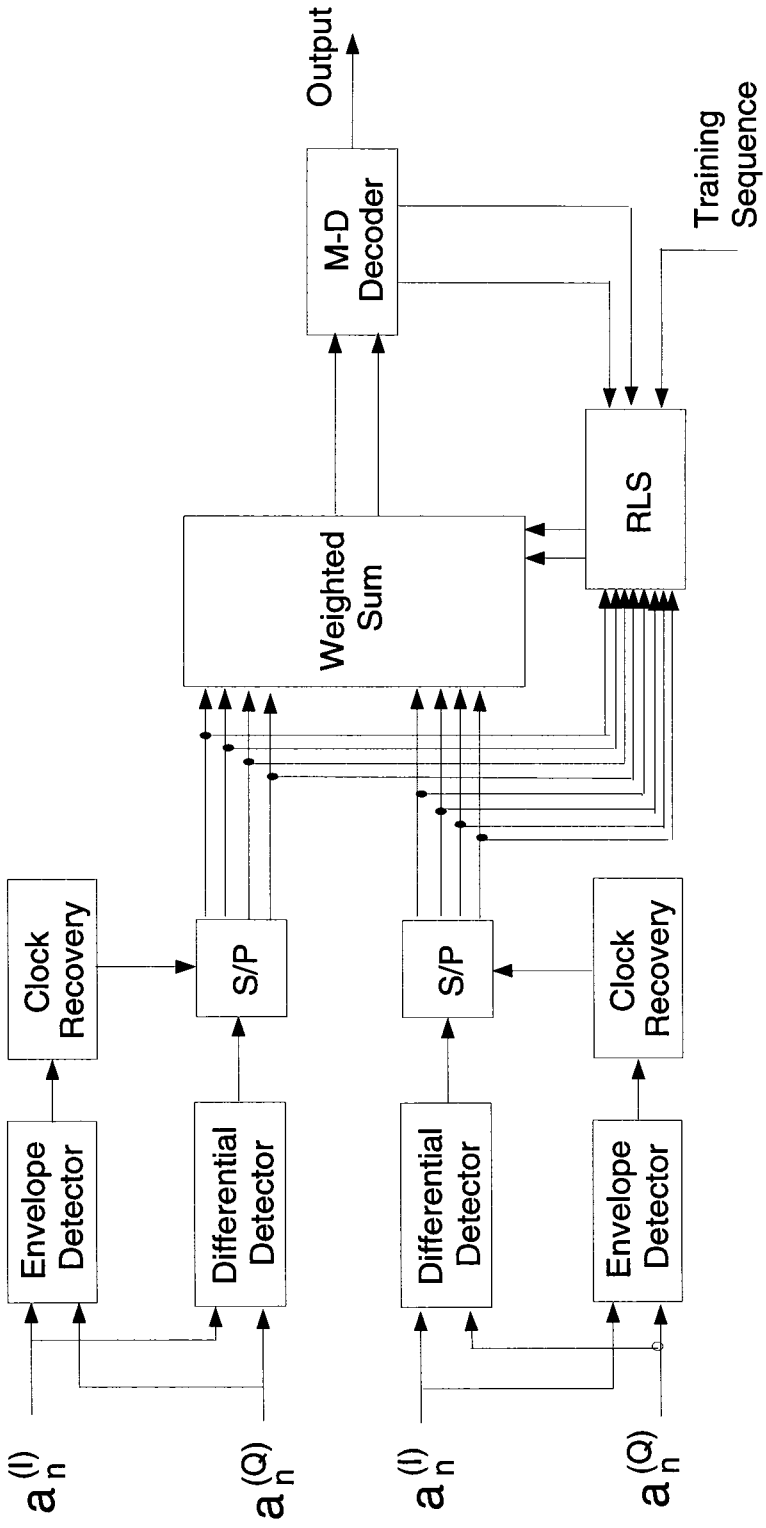


Figure 7.1: Simultaneous diversity combining and decoding for DPSK DS/CDMA system

Bibliography

[1] D. J. Goodman "Trends in Cellular and Cordless Communications" *IEEE Communications Magazine*, pp. 31-40. June 1991.

[2] M. D. Yacoub "Foundations of Mobile Radio Engineering" CRC Press, Inc. 1993.

[3] W. C. Y. Lee, "Mobile Communications Design Fundamentals," 2nd Edition, New York: Wiley International, 1993.

[4] D. J. Goodman "Second Generation Wireless Information Networks" *IEEE Trans. on Vehicle Technology*, vol. 40, pp. 366-374, May. 1991.

[5] R. G. Vaughan, "On optimum combining at the mobile," *IEEE Trans. on Vehicle Technology*, vol. 37, pp. 181-188, Nov. 1988.

[6] J. H. Winters, "Optimum combining in digital mobile radio with cochannel interference," *IEEE J. Select Area in Communications*, vol. SAC-2, pp. 528-539, July. 1984.

[7] J. H. Winters, "On the capacity of radio communication system with diversity in Rayleigh fading environment," *IEEE J. Select Area in Communications*, vol. SAC-5, pp. 871-878, June. 1987.

[8] J. H. Winters, "Signal Acquisition and Tracking with Adaptive Arrays in the Digital Mobile Radio System IS-54 with Flat Fading," *IEEE Trans. on Vehicular Technology*, vol. 42, No. 4, pp. 377-384, November 1993.

[9] J. H. Winters, J. Salz and R. D. Gitlin, "The Impact of Antenna Diversity on the Capacity of Wireless Communication Systems," *IEEE Trans. on Communications*, vol. 42, No. 2/3/4, pp. 1740-1751, February/March/April 1994.

[10] S. C. Swales, M. A. Beach, D. J. Edwards and J. P. McGeehan, "The performance enhancement of multibeam adaptive base-station antennas for cellular land mobile radio systems," *IEEE Trans. on Vehicular Technology*, vol. 39, No.1, pp. 56-67, Feb. 1990.

[11] S. Anderson, M. Millnert, M. Viberg and B. Wahlberg,, "An adaptive array for mobile communication systems" *IEEE Trans. on Vehicular Technology*, vol. 40, No.1, pp. 230-236, Feb. 1991.

[12] M. Cooper and R. Roy, "SDMA technology-overview and development status," ArrayComm-ID-010, ArrayComm, Inc., CA.

[13] B. Friedlander and A. J. Weiss, "Direction Finding Using Spatial Smoothing With Interpolated Arrays" *IEEE Trans. on Aerospace and Electronic System* vol. 28, No. 2, pp. 574-586, April, 1992.

[14] Y. Brester and A. Macovski, "Exact maximum likelihood parameter estimation of superimposed exponential signals in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol.34, pp. 1081-1089, Oct. 1986.

[15] J. A. Cadzow, "A high resolution direction-of-arrival algorithm for narrow-band coherent and incoherent sources," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 965-979, July 1988.

[16] M. Viberg and B. Ottersten, "Sensor array processing based on subspace fitting," *IEEE Trans. Signal Processing* vol. 39, No. 5, pp. 1110-1121, May 1991.

[17] M. Viberg, B. Ottersten and T. Kailath "Detection and estimation in sensor arrays using weighted subspace fitting" *IEEE Trans. Signal Processing* vol. 39, No. 11, pp. 2436-2449, Nov. 1991.

[18] A. J. Weiss and B. Friedlander, "Performance Analysis of Spatial Smoothing with Interpolated Arrays" *IEEE Trans. on Signal Processing* vol. 41, No. 5, pp. 1881-1892, May, 1993.

[19] R. A. Monzigo and T. W. Miller, "Introduction to Adaptive Arrays," New York: Wiley, 1980.

[20] S. Haykin, "Adaptive Filter Theory," Second Edition, Prentice Hall Information and System Sciences Series, 1991

[21] N. E Hubing, S. T. Alexander, "Statistical Analysis of Initialization Methods for RLS Adaptive Filter," *IEEE Trans. on Signal Processing*, vol. 39, No. 8, pp. 1793-1804, August 1991.

[22] Y. Liu, M. Wallace, and J. W. Ketchum, "A Soft-Output Bidirectional Decision Feedback Equalization Technique for TDMA Cellular Radio," *IEEE Trans. on Selected Areas in Communications*, vol. 11, No. 7, pp. 1034-1045, September 1993.

[23] T. Ohgane, T. Shimura, N. Matsuzawa, and H. Sasaoka, "An Implementation of a CMA Adaptive Array for High Speed GMSK Transmission in Mobile Communications," *IEEE Trans. on Vehicular Technology*, vol. 42, No. 3, pp. 282-288, August 1993.

[24] W. C. Y. Lee, "Applying the Intelligent Cell Concept to PCS," *IEEE Trans. on Vehicular Technology*, vol. 43, No.3, pp. 672-679, August, 1994.

[25] R. O. Schmidt, "A signal subspace approach to multiple source location and spectral estimation," Ph.D. dissertation, Stanford University, Stanford, CA, May 1981.

[26] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, No. 7, pp. 984-995, July 1989.

[27] J. Capon, "High-resolution frequency-wavenumber spectrum analysis," *Proc. IEEE*, vol.57, pp. 1408-1418, Aug. 1969.

[28] J. G. Mcwhirter and T. J. Shepherd "Systolic array processor for MVDR beamforming," *IEE Proc.-F* (London), vol.136, pp. 75-80, April, 1989.

[29] O. L. Frost, "An algorithm for linearly constrained adaptive array processing," *Proc.IEEE*, vol.60, pp. 926-935, Aug. 1972.

[30] T. J. Shan, M. Wax and T. Kailath, "On spatial smoothing for Direction-of-Arrival estimation of coherent signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, No. 4, pp. 806-811, Aug. 1985.

[31] B. Widrow, K .M. Duval, R. P. Gooch and W. C. William, "Signal cancellation phenomena in adaptive antennas: causes and cures" *IEEE Trans. Acoust., Speech, Signal Processing*, vol. AP-30, No. 3, pp. 469-478, May 1982.

[32] J. E. Evans, J. R. Johnson and D. F. Sun, "High resolution angular spectrum estimation techniques for terrain scattering analysis and angle of arrival estimation", in *1st ASSP Workshop Spectral Estim.*, Hamilton (CANADA), 1981.

[33] T. J. Shan and T. Kailath, "Adaptive beamforming for coherent signals and interference", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, No. 3, pp. 527-536, June 1985.

[34] V. U. Reddy, A. Paulraj and T. Kailath, "Performance analysis of the optimum beamformer in the presence of correlated sources and its behavior under spatial smoothing", *IEEE Trans. Signal Processing*, vol. 35, No. 7, pp. 927-936, July 1987.

[35] C. C. Yeh and W. D. Wang, "Coherent interference suppression by an antenna array of arbitrary geometry" *IEEE Trans. on AP* vol. 37, No. 10, pp. 1317-1322, Oct. 1989.

[36] M. Lu and Z. Y. He, "Adaptive beamforming for coherent interference suppression" in *Proc. ICASSP 93*, Minnesota, USA, 1993, pp. (I-301)-(I-304).

[37] S. Haykin, *Adaptive Filter Theorem.* Pretice-Hall, Information and System sciences Series, 1991.

[38] G. W. Stewart, "An updating algorithm for subspace tracking," *IEEE Trans. Signal Processing* , vol. 40, No. 6, pp. 1535-1541, June 1992.

[39] K. J. R. Liu, D. P. O'Leary, G. W. Stewart, and Y.-J. Wu, "URV ESPRIT for Tracking Time-Varying Signals", *IEEE Trans. on Signal Processing*, Vol 42, No 12, pp.3441-3448, Dec. 1994.

[40] S. U. Pillai and B. H. Kwon, "Forward/backward spatial smoothing techniques for coherent signal identification," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, No. 1, pp. 8-15, Jan. 1989.

[41] C. Proukakis and A. Manikas, "Study of Ambiguities of Linear Arrays," Proc of ICASSP pp.IV-(549-552), 1994

[42] K. Tan and Z. Goh, "A construction of arrays free of rank ambiguities," Proc of ICASSP pp.IV-(545-548), 1994

[43] J. T.-H. Lo and S. L. Marple, "Observability conditions for multiple signal direction finding and array sensor localization," *IEEE Trans. Signal Processing*, vol. 40, No. 11, pp. 2641-2650, Nov. 1992.

[44] J. Li, "Improved angular resolution for spatial smoothing techniques," *IEEE Trans. Signal Processing*, vol. 40, pp. 3078-3081, Dec. 1992.

[45] T. Shan, A. Paulraj and T. Kailath, "On Smoothed Rank Profile Tests in Eigenstructure Methods for Directions-of-Arrival Estimation," *IEEE Trans. Acoustic, Speech, and Signal Processing*, vol. 35, pp. 1377-1385, Oct. 1987.

[46] J. Litva and M. Zeytinoghi, "Application of high-resolution direction-finding algorithms to circular arrays with mutual coupling present" Final report prepared for DREO under terms of Contract No. W7714-9-9132/01-SZ, July 1990.

[47] A. L. Swindlehurst, B. Ottersten, R. Roy and T. Kailath, "Multiple invariance ESPRIT" *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 40, No. 4, pp. 867-881, April 1992.

[48] J. Proakis, "Digital Communications," Second Edition, McGraw-Hill, Series, 1989.

[49] A. J. Viterbi, "CDMA Principles of Spread Spectrum Communication" Addison-Wesley Publishing Company 1995

[50] J. Kennedy, M. C. Sullivan, "Direction Finding and "Smart Antennas" Using Software Radio Architectures *IEEE Communications Magazine*, pp. 62-68, May 1995.

[51] M. V. Clark, L. J. Greenstein, W. K. Kennedy and M. Shafi, "MMSE Diversity Combining for Wide-Band Digital Cellular Radio," *IEEE Trans. on Communications*, vol. 40, No.6, pp. 1128-1135, June 1992.

[52] P. Jung, "Performance Evaluation of a Novel M-Detector for Coherent Receiver Antenna Diversity in a GSM-Type Mobile Radio System," *IEEE Journal on Selected Areas in Communications*, vol. 13, No.1, pp. 80-88, January 1995.

[53] P. R. Chevillat and E. Eleftheriou, "Decoding of Trellis-Encoded Signals in the Presence of Intersymbol Interference and Noise," *IEEE Trans. on Communications*, vol. 37, No.7, pp. 669-676, July 1989.

[54] K. Zhou, J. G. Proakis and F. Ling, "Decision-Feedback Equalization of Time-Dispersive Channels with Coded Modulation," *IEEE Trans. on Communications*, vol. 38, No.1, pp. 18-24, January 1990.

[55] W. Sheen and G. L. Stuber, "MLSE Equalization and Decoding for Multipath-Fading Channels," *IEEE Trans. on Communications*, vol. 39, No.10, pp. 1455-1464, October 1991.

[56] T. Ohgane, T. Shimura, N. Matsuzawa and H. Sasaoka, "An Implementation of a CMA Adaptive Array for High Speed GMSK Transmission in Mobile Communication," *IEEE Trans. on Vehicular Technology*, vol. 42, No. 3, pp. 282-288, August 1993

[57] T. Schirtzinger, X. Li, and W. K. Jenkins, "A Comparison Of Three Algorithms For Blind Equalization Based On The Constant Modulus Error Criterion," *Proceedings of the 1995 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Detroit, Michigan, May 9-12, 1995, pp. 1049-1052.

[58] H. Kubo, K. Murakami and T. Fujino, "An Adaptive Maximum-Likelihood Sequence Estimation for Fast Time-Varying Intersymbol Interference Channels" *IEEE Journal on Selected Areas in Communications*, vol. 42, No. 2/3/4, pp. 1872-1880, Feb./March/April 1994.

[59] H. Kubo, K. Murakami and T. Fujino, "Adaptive Maximum-Likelihood Sequence Estimation by Means of Combined Equalization and Decoding in Fading Environments," *IEEE Journal on Selected Areas in Communications*, vol. 13, No. 1, pp. 102-109, January 1995.

[60] J. Proakis, "Adaptive Equalization for TDMA Digital Mobile Radio," *IEEE Trans. on Vehicular Technology*, vol. 40, No.2, pp. 333-341, May 1991.

[61] G. Ungerboeck, "Trellis-Coded Modulation with Redundant Signal Sets Part I: Introduction," *IEEE Communication Magazine*, vol. 25, No.2, pp. 5-21, May 1987.

[62] D. E. Knuth, "The Art of Computer Programming," Vol. III: Sorting and Searching. Reading, MA: Addison-Wesley. 1973

[63] J. B. Anderson and S. Mohan, "Sequential Coding Algorithms: A survey and Cost Analysis," *IEEE Trans. on Communications*, vol. 32, No.2, pp. 169-176, Feb. 1984.

[64] A. F. Naguib, A. Paulraj and T. Kailath "Capacity Improvement with Base-Station Antenna Arrays in Cellular CDMA," *IEEE Trans. on Vehicular Technology*, vol. 43, No. 3, August 1994.

[65] K. S. Gilhousen, I. Jacobs, R. Padovani, A. J. Viterbi, L. A. Weaver. Jr. and C. E. Wheatley "On the Capacity of a Cellular CDMA System," *IEEE Trans. on Vehicular Technology*, vol. 40, No. 2, pp. 303-312, May 1991.

[66] W. C. Y. Lee "Overview of Cellular CDMA," *IEEE Trans. on Vehicular Technology*, vol. 40, No. 2, pp. 291-302, May 1991.

[67] H. I. Takayasu and Y. Karasawa "An Investigation of Space-Path Hybrid Diversity Scheme for Base Station Reception in CDMA Mobile Radio" *IEEE Journal on Selected Areas in Communications*, vol. 12, No. 5, pp. 962-969, June 1994.

[68] A. Higashi and T. Matsumoto "Combined Adaptive RAKE Diversity (ARD) and Coding for DPSK DS/CDMA Mobile Radio" *IEEE Journal on Selected Areas in Communications*, vol. 11, No. 7, pp. 1076-1084, Sep. 1993.

[69] P. B. Rapajic and B. S. Vucetic "Adaptive Receiver Structures for Asynchronous CDMA systems" *IEEE Journal on Selected Areas in Communications*, vol. 12, No. 4, pp. 685-697, May. 1994.

[70] H. Suzuki "Performance of a New Adaptive Diversity-Equalization for Digital Mobile Radio" *Electronics Letteyyr*, vol. 26, No. 10, pp. 626-627, May. 1990.

[71] M. Russell and J. W. M. Bergmans "A Technique to Reduce Error Propagation in M-ary Decision Feedback Equalization" *IEEE Trans. on Communications*, vol. 43, No.12, pp. 2878-2881, Dec. 1995.

[72] P. Balaban and J. Salz "Optimum Diversity Combining and Equalization in Digital Data Transmission with Applications to Cellular Mobile Radio–Part I: Theoretical Considerations" *IEEE Trans. on Communications*, vol. 40, No.5, pp. 885-894, May. 1992.

[73] P. Balaban and J. Salz "Optimum Diversity Combining and Equalization in Digital Data Transmission with Applications to Cellular Mobile Radio–Part II: Numerical Results" *IEEE Trans. on Communications*, vol. 40, No.5, pp. 895-907, May. 1992.