FANO MULTIPLE-SYMBOL DIFFERENTIAL DETECTORS

by

Kuok Meng (Patrick) Pun

B.A.Sc (Computer Engineering), Simon Fraser University, 2003

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

In the School
of
Engineering Science

SIMON FRASER UNIVERSITY

Summer 2005

APPROVAL

Name:	**Kuok Meng (Patrick) Pun**
Degree:	**Master of Applied Science**
Title of Thesis:	**Fano Multiple-Symbol Differential Detectors**

Examining Committee:

Chair: **Dr. Bozena Kaminska**
Professor, School of Engineering Science

____

Dr. Paul Ho
Senior Supervisor
Professor, School of Engineering Science

____

Dr. James Cavers
Supervisor
Professor, School of Engineering Science

____

Dr. Dong In Kim
Internal Examiner
Associate Professor, School of Engineering Science

Date Defended/Approved: July 12, 2005

ABSTRACT

Multiple-symbol differential detection (MSDD) is a robust maximum-likelihood receiver for frequency-nonselective fast Rayleigh fading channels. However, its complexity grows exponentially with the block size. Recently, multiple-symbol-differential sphere decoder (MSDSD) is developed to alleviate this problem but its complexity at low signal-to-noise ratio (SNR) grows exponentially. This work investigates the possibility of using a Fano decoder as an efficient MSDD. The detector, namely Fano-MSDD, is an "intelligent" decision-feedback detector (DFD) that uses a running threshold and an accumulated path metric as navigation tools when it roams the decoding tree. Our results indicate that Fano-MSDD is more attractive than DFD from the perspectives of error-performance and complexity. When compared to MSDSD, our best Fano-MSDD suffers a small degradation in power efficiency. However, its complexity is a stable function of SNR. Furthermore, with the extension of Fano-MSDD to differential space-time modulation, our receivers become more remarkable when compared to other reduced complexity techniques.

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude and appreciation to my senior supervisor, Dr. Paul Ho, for his encouragement and guidance in the preparation of this research. Special thanks are due to Dr. James Cavers and Dr. Dong In Kim for examining and reviewing the thesis.

I am also indebted to my family for many years of support and encouragement throughout my studies.

Finally, I am thankful to my lab-mates and friends for providing me a pleasant working environment and always being there when I needed them.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOTATIONS

$\delta[\bullet]$	discrete-time impulsive function
$J_0(\bullet)$	zero-order Bessel function of first kind
$\log_2(\bullet)$	based-2 logarithm function
$round(x)$	rounding operator to round x to the nearest integer
$(\bullet)^*$	complex conjugate of a number
$[\bullet]^T$	transpose of a matrix
$[\bullet]^H$	Hermitian transpose or conjugate transpose of a matrix
$\vert \bullet \vert$	determinant of a matrix
$diag(\mathbf{x})$	diagonal matrix with the components of $\mathbf{x}$ on its diagonal
$\mathbf{I}_K$	identity matrix of size K
$\otimes$	Kronecker product operator
$E[\bullet]$	expectation operator

GLOSSARY

APM	accumulated path metric
AWGN	additive white Gaussian noise
BEP	bit-error probability
DD	differential detection
DFDD	decision-feedback differential detection
EA	eigen-assisted
i.i.d.	independent and identically distributed
ML	maximum likelihood
MDPSK	*M*-ary differential phase shift keying
MPSK	*M*-ary phase shift keying
MIMO	multiple-input-multiple-output
MSDD	multiple-symbol differential detection
MSDSD	multiple-symbol differential sphere decoding
OSTBC	orthogonal space-time block code
p.d.f.	probability density function
p.s.d.	power spectral density
PSAM	pilot-symbol assisted modulation
SISO	single-input-single-output
SNR	signal-to-noise ratio
SQRC	square-root raised cosine
ST	space-time
ST-MSDD	space-time multiple-symbol differential detection
VA	Viterbi algorithm

CHAPTER 1: INTRODUCTION

In wireless communication, a frequently occurring problem with signal propagation is the multipath effect. This multipath effect occurs when the transmitted signal arrives at the receiver via multiple propagation paths at different delays. All of these delayed signal components may add up destructively so that the signal arrived at the receiver may suffer from both amplitude and phase distortions. This phenomenon is termed fading. Since the transmitter commonly modulates its information in the amplitude and phase of the transmitted signal, this multipath effect becomes an issue to many communication systems. In order to achieve reliable reception in such a hostile environment, these amplitude and phase distortions are required to be estimated and tracked. It can be achieved through the adoption of a pilot-symbol assisted modulation (PSAM) [1]. Pilot symbols are sent at a regular interval so that the receiver is able to estimate the channel accurately. However, channel estimation may be too difficult and costly in a fast fading environment. Consider a mobile travelling on a train at a speed of 250 km/hr (approximately 70 m/s) and operating at a carrier frequency of 1.9GHz, the Doppler frequency shift is approximately f_d = 450Hz. If we assume a baud rate $1/T_s$ of 9600Hz, the normalized Doppler frequency f_dT_s then turns out to be about 0.05 and, according to the Nyquist rate, the largest allowable frame size can only be ten symbols long. In other words, one pilot symbol is required to be sent for every nine transmitted symbols (i.e. 90% transmission efficiency). This can be even worse when we consider a space-time (ST) system [2]-[4]. Say we have an Alamouti-type ST system [3] with two transmit antennas and one receive antenna, now it requires channel estimation on two different channels to estimate so twice as many pilot symbols are sent. Therefore, it is natural to consider non-coherent detection techniques that do not require any explicit channel estimation at the receiver.

In the thesis, we present two novel non-coherent detection techniques, termed Fano multiple-symbol differential detection (MSDD) and Fano space-time multiple-symbol differential detection (ST-MSDD), for the single antenna systems and the Alamouti-type ST systems respectively. Before getting into the details of our detectors, we begin this chapter by reviewing some of the major contributions in non-coherent detections.

1.1 Non-Coherent Detections in Single-Input-Single-Output Systems

In a single-input-single-output (SISO) system, differential detection (DD) of M-ary phase shift keying transmission with differential encoding (MDPSK) is a well-known non-coherent detection technique for frequency-nonselective Rayleigh fading environment because of its simple receiver structure. The main idea of MDPSK transmission is that information-bearing symbols are embedded in the phase differences of two consecutive transmitted symbols so that, when the channel condition is quasi-static or is changing very slowly, the first transmitted symbol can be served as a noisy reference symbol providing the receiver a rough estimate of the channel gain. However, if the channel gain changes substantially from one symbol to the next, this reference symbol may no longer be accurate and, subsequently, the error performance of this conventional DD may suffer from an irreducible error floor.

Divsalar and Simon have solved this problem by introducing the concept of MSDD [5] for the additive white Gaussian noise (AWGN) channel. Then, Ho and Fung [6] apply the same concept for Rayleigh fading channel and their results show that MSDD is very robust in combating the irreducible error floor in fast fading environment. The bit-error probability (BEP) improves with increasing block size N and the ideal coherent result can be achieved when N is infinitely large. However, the complexity of a brute-force MSDD receiver grows exponentially with N.

In solving the issue of complexity, Mackenthun [7] has developed an efficient MSDD receiver that can be used in both the AWGN channel and the static fading channel. The complexity of this receiver only depends on the efficiency of its sorting routine, which is only in the order of $N \log(N)$. Unfortunately, adopting this algorithm in a fast fading environment also leads to an irreducible error floor.

Kam and Teh [8] and Schober and Gerstaker [9] have developed a decision-feedback differential detection (DFDD) receiver with moderate complexity to solve the time-varying channel issue. DFDD is based on the concept of linear prediction which makes a decision on the current received signal based on the previous N-1 detected symbols. However, it is apparent that this receiver can suffer from error propagation, especially when the modulation has a large constellation size.

Recently, multiple-symbol differential sphere decoding[1] (MSDSD) [10] is introduced as an optimal MSDD receiver in the fast fading environment. However, the receiver is only suitable while operating at high signal-to-noise ratio (SNR) and, at low

[1] Due to the nature of the MSDD, the terms decoding and detection are used interchangeably in the thesis.

SNR, its complexity grows almost exponentially with decreasing SNR. This will be further verified by the simulation results presented in Chapter 3.

1.2 Non-Coherent Detections in Space-Time Systems

In a multiple-input-multiple-output (MIMO) system, ST codes have received a lot of interest from the wireless communications community because of its ability to provide dramatic improvement to the radio link quality. In particular, orthogonal space-time block code (OSTBC) [4] has been extensively studied in the past few years. Its simplest form, namely the Alamouti-type ST code [3], has been adopted in the third generation systems [11]. However, analogous to single antenna transmission, explicit channel estimation is required at the receiver for reliable performance. An Alamouti-type ST system, which is composed of two transmit antennas and one receive antenna, requires twice the number of pilot symbols as in the SISO channel. In solving the issue of transmission overhead, researchers [12]-[15] have proposed a number of differential ST codes with differential detections. Unfortunately, all of the proposed schemes show a loss of 3 dB in power efficiency when compared to an ideal coherent detector with differential encoding. In order to narrow this performance gap, Gao et al. [16] has extended the concept of MSDD in SISO systems to ST systems and developed a non-coherent ST maximum likelihood (ML) detector. We will refer to this MSDD detector as a ST-MSDD. Intuitively, the BEP of the ST-MSDD detector can improve with the detection window size *N* and can achieve the performance of an ideal coherent detection with differential encoding when *N* is infinity. In exchange, its computational complexity grows exponentially with *N*, severely limiting the benefits it can deliver in practice.

Riediger and Ho [17] have resolved this complexity issue by introducing the Eigen-Assisted (EA) receiver, which has a linear complexity in *N*, the observation window size. However, its application is restricted to the static fading channel. Several sub-optimal, reduced-complexity approaches [18]-[21] have been proposed for fast fading channels but, since these detection techniques are based on the ideas of decision feedback and iterative processing, they are subjected to the problems of error propagation and slow convergence. This leads us to the work of investigating a practical MSDD receiver with good balance in BEP performance and complexity.

1.3 Thesis Outline

The organization of the thesis is the following. Chapter 2 presents the signal and system model for the SISO system and the Alamouti-type ST system. The primary purpose of this chapter is to provide reader with an overview of our communication system and to introduce some necessary notations that are used throughout the thesis.

We present in Chapter 3 the derivation of the MSDD receivers in the SISO channel, with special emphasis on the Fano-MSDD receivers. Also included in this chapter are the two reference detectors, the DFDD and the MSDSD. They are used as benchmarks when we evaluate the performance of our Fano receivers.

In Chapter 4, the focus is shifted to the multiple antenna system and the ST-MSDD sequence detectors are derived. The ST versions of the Fano-MSDD and MSDSD, termed Fano ST-MSDD and ST-MSDSD respectively, are introduced. A new branch sorting/ordering algorithm, which resolves the problem of an exhaustive sort in the Fano ST-MSDD and ST-MSDSD receivers, is also presented. A similar version of the SISO decision-feedback detector, termed ST-DFDD, is used as performance guidance.

The last chapter summarizes the work in the thesis and provides suggestions on the possible improvements of our Fano receivers.

1.4 Contributions of the Thesis

Of all the SISO non-coherent detectors discussed in Section 1.1, none of them can provide simultaneously the attributes of 1) low implementation complexity, 2) robustness against fast fading environment, and 3) attaining near ML performance. In the thesis, we explore the potential of the Fano algorithm as a computationally efficient MSDD receiver. The detector, termed a Fano-MSDD, is capable of delivering excellent error performance at moderate implementation complexity over a wide range of SNR and fading rates. Our simulation results indicate that Fano-MSDD only suffers from a moderate degradation in power efficiency relative to the MSDSD while its computational complexity is very steady over a wide range of SNR.

We also propose a bi-directional Fano-MSDD, termed Bi-Fano MSDD, which searches in both the forward and the reverse directions, and provide a mechanism for error detection and error correction. An edge-cut variant of this bi-directional algorithm, Edge-Bi-Fano, is introduced to further improve the power efficiency. Simulation results indicate that both of the bidirectional detectors have near ML bit-error performance and there is no noticeable irreducible error floor. The complexity of the Bi-Fano MSDD approximately doubles that of a unidirectional Fano-MSDD and it is also a very stable function of SNR.

By adopting the concepts of the Fano, Bi-Fano, and Edge-Bi-Fano algorithms to the Alamouti-type ST system, we arrive at an efficient implementation of the ST-MSDD receivers. They are inherently referred to as the Fano ST-MSDD, the Bi-Fano ST-MSDD, and the Edge-Bi-Fano ST-MSDD. As benchmarks, we present two ST variants of the MSDSD and the DFDD, namely the ST-MSDSD and the ST-DFDD respectively.

In addition, we discover two refinements that can further reduce the computational complexity of our receivers. The first refinement is the adoption of an improved threshold adjustment scheme for the basic Fano algorithm in order to avoid the unnecessary back-and-forth node transitions in the decoding process. The other refinement is an efficient branch sorting routine proposed for both the Fano ST-MSDD receivers and the ST-MSDSD receiver. By adopting this branch sorting algorithm, these detectors do not need to go through the tedious process of calculating and sorting all branch metrics on each node transition and now they can perform sorting on demand. The author would like to emphasize that this sorting algorithm can be applied to any ST sequence detectors and the same concept can be easily extended to other MIMO systems with more transmit and receive antennas.

CHAPTER 2: SIGNAL AND SYSTEM MODEL

In this chapter, we present the signal and system model for the communication systems in the SISO channel and the Alamouti-type ST channel. We begin with a brief description of the baseband representation of bandpass systems. Then the channel and signal model for SISO systems is presented. We conclude with the description of the Alamouti-type ST modulation and its received signal representation.

2.1 Baseband Representation of Bandpass Systems

In digital communication systems, information is sent by modulating a carrier with a signal, whose bandwidth is much smaller than the carrier frequency. These modulated signals are called narrowband bandpass signals and they can be expressed in the form

$$\begin{aligned} \tilde{s}(t) &= A(t)\cos\left(2\pi f_c t + \theta(t)\right) \\ &= \mathrm{Re}\left\{ s(t) e^{j2\pi f_c t} \right\}, \end{aligned} \tag{1}$$

where $A(t)$ and $\theta(t)$ are the amplitude and phase of the information signal respectively, f_c is the carrier frequency, and $s(t) = A(t)e^{j\theta(t)}$ is the complex baseband equivalent of the bandpass signal $\tilde{s}(t)$. For mathematical convenience, sometimes it is more desirable to represent our modulated signals and channels by their baseband equivalents. Therefore, we adopt the complex baseband representation throughout the thesis and we only deal with the transmission of the equivalent low-pass signals through the equivalent low-pass channels without any loss of generality.

2.2 Transmission Model in Single-Input-Single-Output Systems

As described in Chapter 1, when the channel condition is changing too rapidly, PSAM [1] is inefficient in transmitting information over the channel as extra bandwidth is allocated for the pilot symbols. Consequently, non-coherent modulation is a more plausible transmission scheme in this hostile channel condition. MDPSK is a common non-coherent transmission technique and we adopt this transmission scheme for our SISO system.

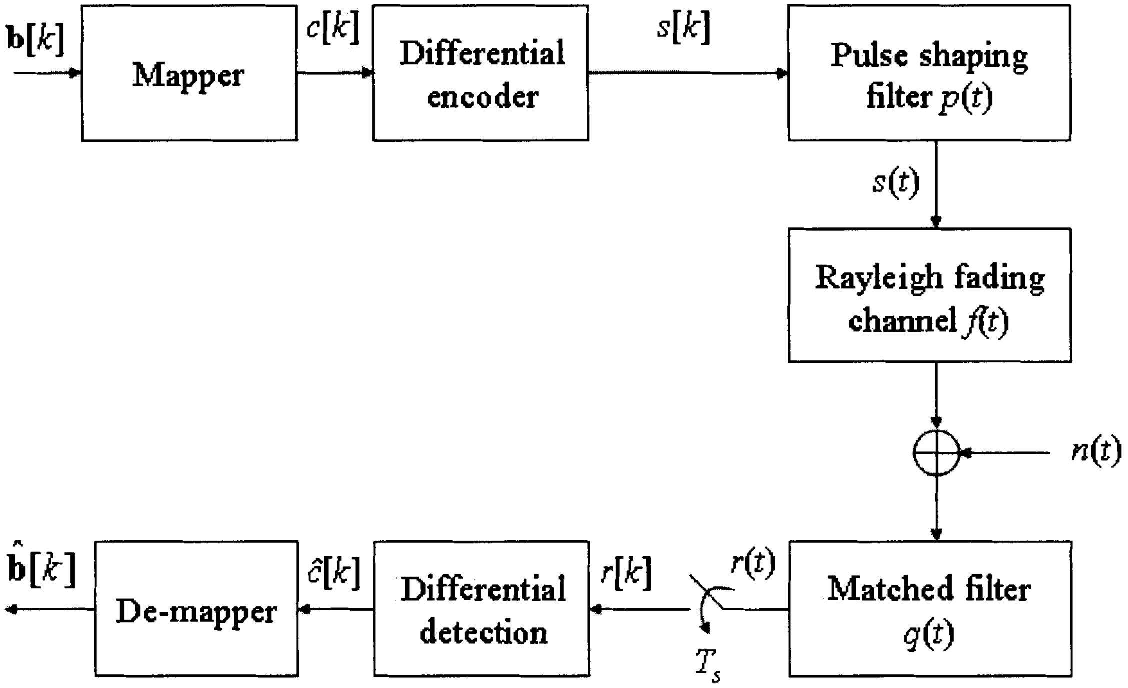

Figure 2-1. Transmission model for a SISO system.

Consider the SISO transmission model illustrated in Figure 2-1. At discrete time k, the raw information bit sequence $\mathbf{b}[k]$ of length $n_b = \log_2(M)$ is passed to a mapping device which maps the information bits using Gray mapping into a data symbol $c[k]$ from the MPSK constellation $S = \{\exp(j2\pi m / M) : m = 0,\ 1,\ ...,\ M-1\}$. Then, the differential encoder encodes this data symbol to form a transmitted symbol $s[k]$ according to the differential encoding formula

$$s[k] = c[k]s[k-1]. \tag{2}$$

The symbol $s[k-1]$ in the above equation is the previous transmitted symbol and the value of the reference transmitted symbol $s[0]$ is set to unity for convenience. Without loss of generality, $s[k]$ is also from the same MPSK set S. After the transmitted symbol is modulated by a square-root raised cosine (SQRC) pulse shaping filter $p(t)$ with a one-sided bandwidth of $1/T_s$ and unity energy $\int p^2(t)dt = 1$, the transmitted signal is represented as

$$s(t) = \sum_k s[k]p(t - kT_s). \tag{3}$$

We assume the transmission channel is a frequency-nonselective Rayleigh fading channel with AWGN. Subsequently, the received signal can be mathematically modelled as

$$r(t) = f(t)s(t) + n(t), \tag{4}$$

where the fading gain $f(t)$ and the noise term $n(t)$ are both zero-mean complex Gaussian random processes with a variance of $\sigma_f^2 = \frac{1}{2}E\left[|f(t)|^2\right]$ and a power spectral density (p.s.d.) of $N_0 = 1$ respectively. The fading gains $f(t)$'s are assumed constant within each symbol interval T_s but can vary from interval to interval. Under the assumptions of isotropic scattering and vertical polarized antenna, the fading gain $f(t)$ has a Jakes' power spectrum or, in other words, its autocorrelation function is

$$\phi_f(\tau)=\frac{1}{2}E\left[f(t)f^*(t-\tau)\right]=\sigma_f^2 J_0(2\pi f_d \tau), \tag{5}$$

where $J_0(\bullet)$ is the zero-order Bessel function of the first kind and f_d is the Doppler frequency.

An SQRC matched filter $q(t)=p(-t)$ is used at the receiver to maximize the SNR and the matched filter output is sampled at a rate of $1/T_s$. With perfect sampling, a normalized Doppler frequency of $f_d T_s < 0.05$, and no channel/processing delays, the received sample at discrete time k can be written as [22]

$$r[k]=f[k]s[k]+n[k]. \tag{6}$$

Note that the autocorrelation function of the fading sample $f[k]$ is now

$$\phi_f[v]=\sigma_f^2 J_0(2\pi f_d T_s v)=\sigma_f^2 J_v, \tag{7}$$

where $J_v=J_0\left(2\pi f_d T_s v\right)$, and the variance of the noise sample $n[k]$ is $N_0=1$. Based on the channel model in (6), the average received bit SNR is

$$\Gamma_b=\frac{\sigma_f^2}{n_b}. \tag{8}$$

The received samples are then passed onto a differential detector for estimating their information-bearing symbols $c[k]$'s and these estimated information symbols $\hat{c}[k]$'s are further converted into their corresponding information bit sequences $\hat{\mathbf{b}}[k]$'s. In

Chapter 3, we will describe the design of the Fano-MSDD receiver as an efficient generalized differential detector and provide BEP and complexity comparisons against other reduced-complexity techniques.

2.3 Transmission Model in Space-Time Systems

Alamouti-type ST code, which is used for a MIMO system with two transmit antennas and one or more receive antennas, is a well-known ST modulation technique for transmitting signals across both space and time in order to achieve transmit diversity. Because of its relatively simple structure, the ST system considered in the thesis is confined to Alamouti-type ST system. However, the same concept can be easily extended to other OSTBC [4] with more transmit and receive antennas.

As mentioned in Chapter 1, the downside of Alamouti-type ST system is the loss of transmission efficiency because twice the number of pilot symbols is required. This motivates the investigation on the design of differential ST codes [12]-[14]. However, these differential schemes exhibit a 3-dB performance deficit in power efficiency. The differential scheme proposed by Tarokh and Jafarkhani [12] is of our particular interest and we adopt this transmission scheme for the ST systems discussed throughout the thesis. Figure 2-2 illustrates the generalization of our ST transmission model.

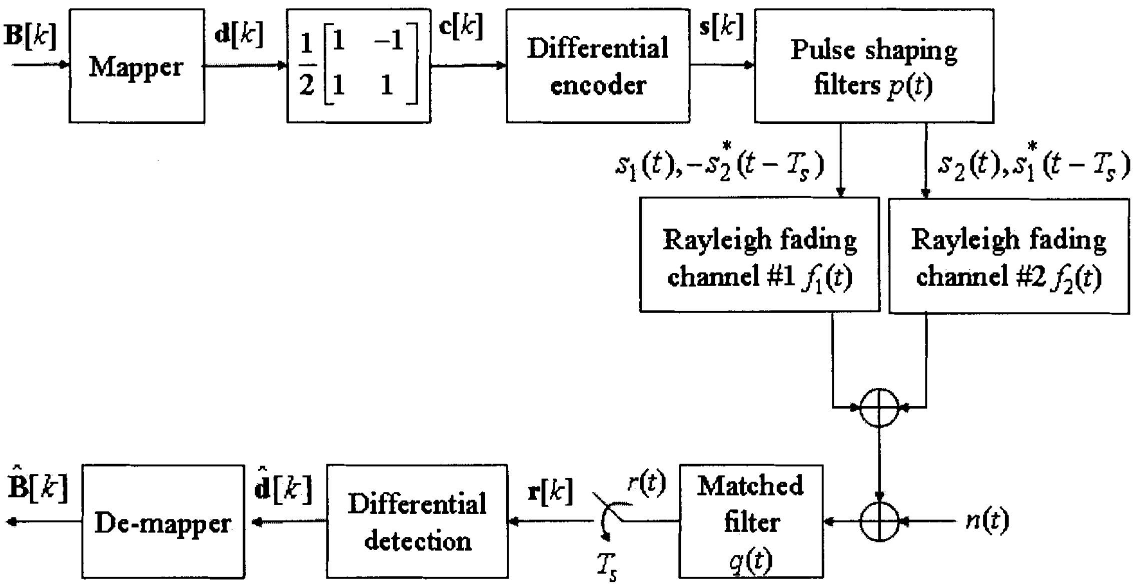

Figure 2-2. Transmission model for a ST system.

The transmission model for our ST system is similar to the one in the SISO channel except that there are now two transmit antennas and the differential encoder is following the ST differential encoding rule specified in [12]. In Alamouti-type ST system, the encoding/modulation period is two-symbol long. In order to transmit at the same rate of n_b bits per T_s seconds as in the SISO case, Tarokh and Jafarkhani [12] suggest the following scheme.

At the k-th ST interval (of length $T_{ST} = 2T_s$), the incoming information bit sequence $\mathbf{B}[k] = \left[b_1[k],\ b_2[k],\ \ldots,\ b_{2n_b}[k]\right]$ is encoded using Gray mapping as a data vector $\mathbf{d}[k] = \left[d_1[k] \quad d_2[k]\right]$ where $d_1[k]$ and $d_2[k]$ are holding the first and second n_b information bits respectively and they are both MPSK $\left(M = 2^{n_b}\right)$ symbols from the signal set S. The data vector $\mathbf{d}[k]$ is then transformed into a ST data symbol with the Alamouti-type ST structure

$$\mathbf{D}[k]=\frac{1}{\sqrt{2}}\begin{bmatrix} d_1[k] & d_2[k] \\ -d_2^*[k] & d_1^*[k] \end{bmatrix}. \tag{9}$$

This ST data symbol $\mathbf{D}[k]$ is then multiplied by a transformation matrix

$$\mathbf{A}[k]=\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \tag{10}$$

to form a rotated ST data symbol

$$\mathbf{c}[k]=\begin{bmatrix} c_1[k] & c_2[k] \\ -c_2^*[k] & c_1^*[k] \end{bmatrix}=\mathbf{D}[k]\mathbf{A}[k]. \tag{11}$$

This rotated ST data symbol $\mathbf{c}[k]$ is further differentially encoded into the k-th transmitted symbol

$$\mathrm{s}[k]=\mathbf{c}[k]\mathrm{s}[k-1]=\begin{bmatrix} s_1[k] & s_2[k] \\ -s_2^*[k] & s_1^*[k] \end{bmatrix}, \tag{12}$$

where $\mathrm{s}[k-1]$ is the symbol transmitted in the previous interval. It is worth mentioning that the transformation matrix $\mathbf{A}[k]$ is used to obtain the same set of rotated ST data symbols $\mathbf{c}[k]$'s defined in [12], [21], and [23]. This special transformation is not necessarily required in the differential encoding process and the original ST data symbol $\mathbf{D}[k]$ can be directly encoded to form the transmitted symbol $\mathrm{s}[k]$ (i.e. $\mathrm{s}[k]=\mathbf{D}[k]\mathrm{s}[k-1]$).

The initial transmitted symbol, $\mathrm{s}[0]$, has the Alamouti-type ST structure and is unitary, i.e. $\mathrm{s}[0]\mathrm{s}^H[0]=\mathbf{I}_2$. As a result, all subsequent $\mathrm{s}[k]$'s also have the Alamouti-type

ST structure and are unitary matrices. The constant $1/\sqrt{2}$ in (9) and (10) is introduced so that the transmitted power in $\mathbf{s}[k]$ is kept constant at all time. Note that the individual $s_i[k]$'s are not necessarily MPSK symbols due to constellation expansion in the differential encoding process.

The matrix shown in (12) can be interpreted as follows: the individual symbol in the *i*-th row and *j*-th column of the matrix represents the symbol transmitted at time $t = kT_{ST} + (i-1)T_s$ by the *j*-th antenna. For example, the symbols $s_1[k]$ and $s_2[k]$ in (12) correspond to the transmitted symbols sent at time $t = kT_{ST}$ by the first and the second antenna respectively. Similarly, the symbols $-s_2^*[k]$ and $s_1^*[k]$ in the second row are sent at time $t = kT_{ST} + T_s$. The individual transmitted symbols at the two antennas are modulated by a unit-energy SQRC pulse shaping filter $p(t)$, and they are sent over a two-link Rayleigh flat fading channel with AWGN. The fading gains $f_1(t)$ and $f_2(t)$ of the two links are independent and identically distributed (i.i.d.) zero-mean complex Gaussian processes with the autocorrelation function defined in (5). The fading gains are assumed to be constant over the entire ST interval T_{ST} but they can vary from one ST interval to the other. The noise term $n(t)$ is a zero-mean complex white Gaussian random process with a variance of $N_0 = 1$. At the receiver, an SQRC matched filter $q(t) = p(-t)$ is used and the filtered signal is sampled at a rate of $1/T_s$. With the assumption of no processing/channel delays, two received samples, $r_1[k]$ and $r_2[k]$, are obtained for the *k*-th ST interval. Their relationship with $\mathbf{s}[k]$, in vector form, is

$$\mathbf{r}[k] = \begin{bmatrix} r_1[k] \\ r_2[k] \end{bmatrix} = \mathbf{s}[k]\mathbf{f}[k] + \mathbf{n}[k], \tag{13}$$

where

$$\mathbf{f}[k] = \begin{bmatrix} f_1[k] \\ f_2[k] \end{bmatrix} \tag{14}$$

and

$$\mathbf{n}[k] = \begin{bmatrix} n_1[k] \\ n_2[k] \end{bmatrix} \tag{15}$$

are respectively the fading gain vector and the noise vector. $f_1[k]$ and $f_2[k]$, the fading gains in the two links, are also i.i.d. zero-mean complex Gaussian random sequences with an autocorrelation function

$$\phi_f[v] = \tfrac{1}{2}E\left[f_i^*[k]f_i[k+v]\right] = \sigma_f^2\xi_v, \tag{16}$$

where $\xi_v = J_0\left(2\pi f_d T_{ST} v\right)$. It means the autocorrelation of the fading gain vector $\mathbf{f}[k]$ is

$$\mathbf{\Omega}_f[v] = \tfrac{1}{2}E\left[\mathbf{f}[k+v]\mathbf{f}^H[k]\right] = \phi_f[v]\mathbf{I}_2. \tag{17}$$

As for the noise terms $n_i[k]$, $i = 1,2$, they are i.i.d. zero-mean complex Gaussian sequence and their autocorrelation function, at the vector level, is

$$\mathbf{\Omega}_n[v] = \tfrac{1}{2}E\left[\mathbf{n}[k+v]\mathbf{n}^H[k]\right] = \delta[v]\mathbf{I}_2. \tag{18}$$

where $\delta[v]$ is the discrete time impulsive function. Note also that the fading gains and the channel noise are mutually independent. Given that the transmitted ST symbol $\mathbf{s}[k]$

has a unitary structure, i.e. $\mathbf{s}[k]\mathbf{s}^H[k] = \mathbf{I}_2$, we can deduce, based on the channel model in (13) that the received bit SNR is

$$\Upsilon_b = \frac{\sigma_f^2}{2n_b} = \frac{\Gamma_b}{2}, \tag{19}$$

where Γ_b is the bit SNR defined in (8). Note that the bit SNR of the ST system is half that of the SISO system. This stems from the fact that now the total energy transmitted per ST interval is split into two individual transmitted symbols $s_1[k]$ and $s_2[k]$ in two sub-intervals T_s's, and they have the energy constraint of unity (i.e. $|s_1[k]|^2 + |s_2[k]|^2 = 1$).

Finally, the received samples are passed onto differential detectors that will be described in Chapter 4 for estimating the information pattern $\mathbf{d}[k]$. The estimated pattern, $\hat{\mathbf{d}}[k]$, is then mapped to a corresponding information bit sequence $\hat{\mathbf{B}}[k]$.

CHAPTER 3: SINGLE-INPUT-SIGNAL-OUTPUT MULTIPLE-SYMBOL DIFFERENTIAL DETECTIONS

As described earlier in Chapter 1, MSDD is a robust and effective detection strategy for achieving coherent performance in both AWGN and Rayleigh fading channels but at a cost of an exponential growth in complexity with the block size *N*. In this chapter, we consider three reduced-complexity sequence detectors in a single antenna system. Of the three, the Fano algorithm is the focal point of our discussion.

The organization of this chapter is the following. We present in Section 3.1 the derivation of an ML sequential decoding metric. Two reduced-complexity sequence detectors, the DFDD and the MSDSD, are reviewed in Section 3.2 and Section 3.3 respectively. The former detector is suboptimal while the latter is optimal. Section 3.4 introduces our novel detector with the Fano algorithm as its decoding engine. In addition, an improved version of the Fano algorithm, which makes use of a new threshold adjustment scheme, is presented. Also included in this section are two bi-directional variants of the Fano algorithm, the Bi-Fano and the Edge-Bi-Fano algorithms. At the end, we conclude this chapter with the simulation results.

3.1 Derivation of the ML Decoding Metric

In a fast fading environment, a detector needs to exploit the statistics of the fading channel in order to compensate the fading effects on the transmitted signals and to suppress the irreducible error floor. This can be accomplished by processing simultaneously a block of N received samples at a time. This receiver is referred to as the MSDD receiver. The received samples are divided into overlapping blocks as shown in Figure 3-1, with the i-th block being

$$\mathbf{r}_i = \begin{bmatrix} r[i(N-1)] \\ r[i(N-1)+1] \\ \vdots \\ r[(i+1)(N-1)] \end{bmatrix} = \mathbf{s}_i \mathbf{f}_i + \mathbf{n}_i \tag{20}$$

where $\mathbf{s}_i = \operatorname{diag}\left(s[i(N-1)]\ s[i(N-1)+1]\ \ldots\ s[(i+1)(N-1)]\right)$ is a diagonal matrix representing the transmitted pattern, $\mathbf{f}_i = \left[f[i(N-1)], f[i(N-1)+1], \ldots, f[(i+1)(N-1)]\right]^T$ and $\mathbf{n}_i = \left[n[i(N-1)],\ n[i(N-1)+1],\ \ldots,\ n[(i+1)(N-1)]\right]^T$ denote the corresponding fading gain vector and the noise vector respectively. Note that the overlapping symbol $r[i(N-1)]$ is being used implicitly as a reference symbol. In the following discussion, only the first observation window (i.e. $i = 0$) is assumed and for brevity, we drop the block index i in all the expressions above.

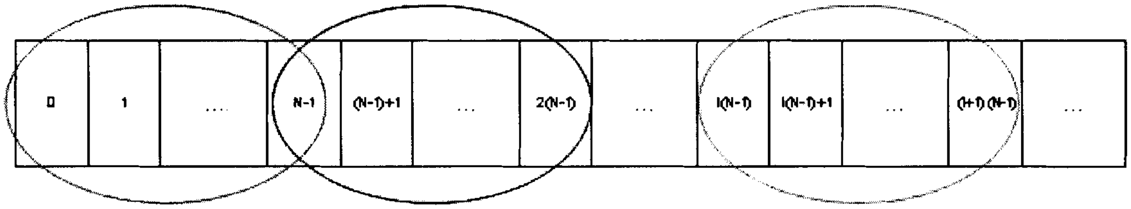

Figure 3-1. Arrangement of the overlapping blocks in a MSDD receiver

Since both the fading gain $f[k]$ and the noise term $n[k]$ are zero-mean complex Gaussian random variables, (20) implies that the received block $\mathbf{r}$ has a joint probability density function (p.d.f.) [24]

$$p_{\mathbf{r}}(\mathbf{r}\,|\,\mathbf{s}) = \frac{1}{(2\pi)^N \left|\mathbf{\Phi}_{rr}\right|} \exp\left\{ -\frac{1}{2}\left[\mathbf{r}^H \mathbf{\Phi}_{rr}^{-1} \mathbf{r} \ \right] \right\}, \tag{21}$$

where

$$\mathbf{\Phi}_{rr} = \frac{1}{2} E\left[\mathbf{r}\mathbf{r}^H \right] = \mathbf{s}\mathbf{\Phi}_{ff}\mathbf{s}^H + \mathbf{\Phi}_{nn} \tag{22}$$

is the covariance matrix of the received vector $\mathbf{r}$, $\mathbf{\Phi}_{ff} = \frac{1}{2} E\left[\mathbf{f}\mathbf{f}^H \right]$ is the covariance matrix of the fading gain vector $\mathbf{f}$, and $\mathbf{\Phi}_{nn} = \frac{1}{2} E\left[\mathbf{n}\mathbf{n}^H \right] = \mathbf{I}_N$ is the covariance matrix of the noise vector $\mathbf{n}$. Since $\mathbf{s}$ is a unitary matrix $\left(\text{i.e. } \mathbf{s}\mathbf{s}^H = \mathbf{I}_N \right)$ and $\mathbf{\Phi}_{nn}$ is a diagonal matrix, (22) can be rewritten as

$$\mathbf{\Phi}_{rr} = \mathbf{s}\left(\mathbf{\Phi}_{ff} + \mathbf{\Phi}_{nn} \right)\mathbf{s}^H = \mathbf{s}\mathbf{\Phi}_{gg}\mathbf{s}^H , \tag{23}$$

where

$$\mathbf{\Phi}_{gg} = \mathbf{\Phi}_{ff} + \mathbf{\Phi}_{nn} = \begin{bmatrix} \sigma_f^2 + 1 & \sigma_f^2 J_{-1} & \cdots & \sigma_f^2 J_{-N+1} \\ \sigma_f^2 J_1 & \sigma_f^2 + 1 & \cdots & \sigma_f^2 J_{-N+2} \\ \vdots & \vdots & \cdots & \vdots \\ \sigma_f^2 J_{N-1} & \sigma_f^2 J_{N-2} & \cdots & \sigma_f^2 + 1 \end{bmatrix} \tag{24}$$

is the covariance matrix of the fading-plus-noise vector $\mathbf{g} = \mathbf{f} + \mathbf{n}$. Now (21) can be expressed as

$$
\begin{aligned}
p_{\mathbf{r}}(\mathbf{r}\,|\,\mathbf{s}) &= \frac{1}{(2\pi)^N \left|\mathbf{\Phi}_{rr}\right|}\exp\left\{-\frac{1}{2}\left[\mathbf{r}^H \mathbf{\Phi}_{rr}^{-1}\mathbf{r}\ \right]\right\} \\
&= \frac{1}{(2\pi)^N \left|\mathbf{s}\ \mathbf{\Phi}_{gg}\mathbf{s}^H\right|}\exp\left\{-\frac{1}{2}\left[\mathbf{r}^H\left(\mathbf{s}\ \mathbf{\Phi}_{gg}\mathbf{s}^H\right)^{-1}\mathbf{r}\ \right]\right\} \\
&= \frac{1}{(2\pi)^N \left|\mathbf{s}\ \mathbf{\Phi}_{gg}\mathbf{s}^H\right|}\exp\left\{-\frac{1}{2}\left[\mathbf{r}^H\mathbf{s}\ \mathbf{\Phi}_{gg}^{-1}\mathbf{s}^H\mathbf{r}\ \right]\right\}.
\end{aligned}
\tag{25}
$$

Since the determinant term

$$\left|\mathbf{s}\ \mathbf{\Phi}_{gg}\mathbf{s}^H\right| = \left|\ \mathbf{s}\ \right| \times \left|\mathbf{\Phi}_{gg}\right| \times \left|\mathbf{s}^H\right| = \left|\mathbf{\Phi}_{gg}\right| \tag{26}$$

is independent of the transmitted pattern $\mathbf{s}$, a joint ML decision on all the $N-1$ information symbols

$$\mathbf{c} = \begin{bmatrix} c[1] & c[2] & \cdots & c[N-1] \end{bmatrix}^T \tag{27}$$

contained in $\mathbf{r}_i$ is made by minimizing the ML metric

$$M\left(\hat{\mathbf{s}}\right) = \mathbf{r}^H\hat{\mathbf{s}}\ \mathbf{\Phi}_{gg}^{-1}\hat{\mathbf{s}}^H\mathbf{r}\ , \tag{28}$$

where $\hat{\mathbf{s}} = \operatorname{diag}\left(\hat{s}[0],\ \hat{s}[1],\ \ldots,\ \hat{s}[N-1]\right)$ is a diagonal matrix representing a possible transmitted pattern. The term $\hat{\mathbf{s}}^H\mathbf{r}$ in (28) represents a hypothesized fading-plus-noise pattern. The multiplication of this pattern and its conjugate transpose $\mathbf{r}^H\hat{\mathbf{s}}$ by $\mathbf{\Phi}_{gg}^{-1}$ in the metric determines how likely the trajectory of the hypothesized fading-plus-noise samples is on the complex plane. Let us define the n-th hypothesized fading-plus-noise sample as $\hat{g}[n] = \hat{s}^*[n]r[n]$ and let $\hat{\mathbf{g}}[n] = \left[\hat{g}[0]\ \hat{g}[1]\ \ldots\ \hat{g}[n]\right]^T$ be the vector containing all such samples up to time n, where $n = 1, 2, \ldots (N - 1)$. Furthermore, let the covariance matrix of the actual fading-plus-noise samples $\mathbf{g}[n] = \left[g[0]\; g[1] \ldots g[n]\right]^T$ be

$$\mathbf{\Phi}_{gg}[n] = \tfrac{1}{2}E\left[\mathbf{g}[n]\mathbf{g}^H[n]\right]. \tag{29}$$

This covariance matrix can be interpreted as an $(n + 1)$ by $(n + 1)$ sub-matrix of the covariance matrix in (24). Now the operation $\hat{\mathbf{g}}^H[n]\mathbf{\Phi}_{gg}^{-1}[n]\hat{\mathbf{g}}[n]$ can be written as

$$\hat{\mathbf{g}}^H[n]\mathbf{\Phi}_{gg}^{-1}[n]\hat{\mathbf{g}}[n] = \left[\hat{\mathbf{g}}^H[n-1] \quad \hat{g}^*[n]\right]\begin{bmatrix}\mathbf{\Phi}_{gg}[n-1] & \boldsymbol{\varphi}_g[n-1] \\ \boldsymbol{\varphi}_g^H[n-1] & \phi[0]\end{bmatrix}^{-1}\begin{bmatrix}\hat{\mathbf{g}}[n-1] \\ \hat{g}[n]\end{bmatrix}, \tag{30}$$

where $\phi[0] = \sigma_f^2 + 1$ is the variance of fading-plus-noise, and

$$\boldsymbol{\varphi}_g[n-1] = \frac{1}{2}E\left[\mathbf{g}[n-1]g^*[n]\right] = \sigma_f^2 \cdot \left[J_{-n}, J_{1-n}, \ldots, J_{-1}\right]^T \tag{31}$$

is the correlation vector between the current and past fading-plus-noise samples. By applying the iterative approach given in [25] and [26], the inverse of the covariance matrix in (29) can be determined iteratively as

$$\begin{aligned}\mathbf{\Phi}_{gg}^{-1}[n] &= \begin{bmatrix}\mathbf{\Phi}_{gg}[n-1] & \boldsymbol{\varphi}_g[n-1] \\ \boldsymbol{\varphi}_g^H[n-1] & \phi[0]\end{bmatrix}^{-1} \\ &= \begin{bmatrix}\mathbf{I}_n & -\mathbf{\Phi}_{gg}^{-1}[n-1]\boldsymbol{\varphi}_g[n-1] \\ \mathbf{0} & 1\end{bmatrix}\begin{bmatrix}\mathbf{\Phi}_{gg}^{-1}[n-1] & \mathbf{0} \\ \mathbf{0} & \dfrac{1}{\varepsilon^2[n]}\end{bmatrix}\begin{bmatrix}\mathbf{I}_n & \mathbf{0} \\ -\boldsymbol{\varphi}_g^H[n-1]\mathbf{\Phi}_{gg}^{-1}[n-1] & 1\end{bmatrix},\end{aligned} \tag{32}$$

where $\mathbf{I}_n$ is an n by n identity matrix and

$$\varepsilon^2[n] = \phi[0] - \boldsymbol{\varphi}_g^H[n-1]\mathbf{\Phi}_{gg}^{-1}[n-1]\boldsymbol{\varphi}_g[n-1]. \tag{33}$$

Consequently, (30) can be simplified to

$$\begin{aligned}\hat{\mathbf{g}}^H[n]\mathbf{\Phi}_{gg}^{-1}[n]\hat{\mathbf{g}}[n] &= \hat{\mathbf{g}}^H[n-1]\mathbf{\Phi}_{gg}^{-1}[n-1]\hat{\mathbf{g}}[n-1] \\ &\quad + \frac{\left|\hat{g}[n] - (\boldsymbol{\varphi}_g^H[n-1]\mathbf{\Phi}_{gg}^{-1}[n-1])\hat{\mathbf{g}}[n-1]\right|^2}{\varepsilon^2[n]},\end{aligned} \tag{34}$$

and applying this recursive relationship to (28) implies the ML metric (after a small change in notation) can now be rewritten as

$$M\left(\hat{\mathbf{g}}[N-1]\right) = \sum_{n=1}^{N-1} \frac{\left|\hat{g}[n] - \left(\boldsymbol{\varphi}_g^H[n-1]\mathbf{\Phi}_{gg}^{-1}[n-1]\right)\hat{\mathbf{g}}[n-1]\right|^2}{\varepsilon^2[n]} = \sum_{n=1}^{N-1} m[n], \tag{35}$$

where

$$m[n] = \frac{\left|\hat{g}[n] - \left(\boldsymbol{\varphi}_g^H[n-1]\mathbf{\Phi}_{gg}^{-1}[n-1]\right)\hat{\mathbf{g}}[n-1]\right|^2}{\varepsilon^2[n]} \tag{36}$$

is the branch metric at time index n. Note that the constant term $|\hat{g}[0]|^2/\phi[0]$ is omitted in the ML metric. The term $\boldsymbol{\varphi}_g^H[n-1]\mathbf{\Phi}_{gg}^{-1}[n-1]$ in (36) can be interpreted as an n-th order linear predictor that predicts the next fading-plus-noise sample based on the previous n hypothesized samples $\hat{\mathbf{g}}[n-1]$. When the hypothesized sequence is the same as the transmitted pattern, then the prediction errors, $\hat{g}[n]-\left(\boldsymbol{\varphi}_g^H[n-1]\mathbf{\Phi}_{gg}^{-1}[n-1]\right)\hat{\mathbf{g}}[n-1]$, $n=1,\ 2,\ \ldots,\ (N-1)$, are independent, zero mean complex Gaussian random variables with variances $\varepsilon^2[n]$. Thus (36) represents the magnitude square of the n-th normalized prediction error and its expected value is thus 1. Since the ML metric in (35) is the sum of $N-1$ such terms, its expected value is simply $N-1$. When the hypothesized sequence differs from the transmitted one, the prediction errors can be very large, owing to the abrupt transitions in the hypothesized fading-plus-noise pattern. Consequently, (35) will be substantially larger than the expected value of $N-1$ for the correct sequence. This property specifically helps the Fano decoder, which will be discussed in Section 3.4, to differentiate the most likely pattern from the erroneous ones.

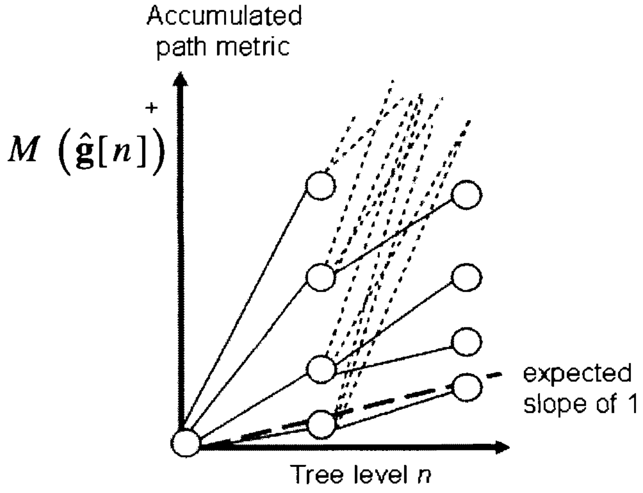

Figure 3-2. Accumulated path metric of a distance tree.

The main reason why we want to manipulate the original metric (28) into (35) is that the metric of the entire hypothesized fading-plus-noise pattern is now expressed as a sum of branch metrics. Furthermore, the accumulated path metric (APM) up to time n (equivalent to a tree level in a distance tree as illustrated in Figure 3-2)

$$a[n] = M\left(\hat{\mathbf{g}}[n]\right) = \sum_{k=1}^{n} m[k] \tag{37}$$

is a non-decreasing function of n and depends only on the received information up to time n. These properties are required for performing a sequential search in the distance tree for both the sphere decoder (in Section 3.3) and the Fano decoder (in Section 3.4).

Finally, the structure of (35) enables us to arrive quickly to the efficient tree search algorithms, the sphere decoding and the Fano algorithms, as we shall see in Sections 3.3 and 3.4 respectively.

3.2 DFDD

Compared to the sphere decoder and the Fano decoder, DFDD [8]-[9] is a simple sequential decoding technique in which its implementation is the least complicated. Unlike the original MSDD, which makes a joint decision on $N-1$ data symbols simultaneously, the DFDD makes a decision one symbol at a time based on the past N-1 decisions. In fact, the decoding process of DFDD with an observation window of size N employs an $(N-1)$-th order linear predictor. In order to maintain consistency with our usual notations, the decision metric of the DFDD in [8]-[9] is represented as

$$M\left(\hat{g}[n]\right)=\frac{\left|\hat{g}[n]-\mathbf{p}_N\hat{\mathbf{g}}_N[n]\right|^2}{\varepsilon_N^2}, \tag{38}$$

where $\mathbf{p}_N=\boldsymbol{\varphi}_g^H[N-2]\boldsymbol{\Phi}_{gg}^{-1}[N-2]$ is the $(N-1)$-th order linear predictor that predicts the next fading-plus-noise sample $g[n]=s^*[n]r[n]$ at discrete time n, $\varepsilon_N^2=\phi[0]-\boldsymbol{\varphi}_g^H[N-2]\boldsymbol{\Phi}_{gg}^{-1}[N-2]\boldsymbol{\varphi}_g[N-2]$ is the corresponding mean square prediction error, $\hat{\mathbf{g}}_N[n]=\left[\hat{g}[n-N+1]\ \hat{g}[n-N+2]\ ...\ \hat{g}[n-1]\right]^T$, $\hat{g}[n]=\hat{s}^*[n]r[n]$, is the vector containing the past $(N-1)$ samples, and $\hat{s}[n]$ is the decision on $s[n]$. Given that the mean square prediction error ε_N^2 is independent of the hypothesis, the estimated transmitted symbol $\hat{s}[n]$ can be easily computed by calculating the phase angle

$\theta[n] = \arg\left\{\mathbf{p}_N \hat{\mathbf{g}}_N[n] r^*[n]\right\}$ and locating the point in the MPSK constellation that is closest in phase to $\theta[n]$. In other words, the estimated transmitted symbol can be computed by

$$\hat{s}[n] = \exp\left(j\frac{2\pi}{M}\rho[n] \right), \tag{39}$$

where

$$\rho[n] = round\left(\frac{\theta[n] \times M}{2\pi} \right) \tag{40}$$

is the phase index of the estimated transmitted symbol $\hat{s}[n]$ and $round(x)$ is the rounding operator which rounds x to the nearest integer.

After the estimated transmitted symbol $\hat{s}[n]$ is calculated, the decoder then slides the observation window forward by one symbol and makes a decision on the next transmitted symbol $s[n+1]$. Figure 3-3 depicts the arrangement of the overlapping blocks in a DFDD receiver. We will see in the simulation results that the complexity of a DFDD receiver is independent of either the SNR or the fading condition, but it can suffer from poor BEP performance due to error propagation.

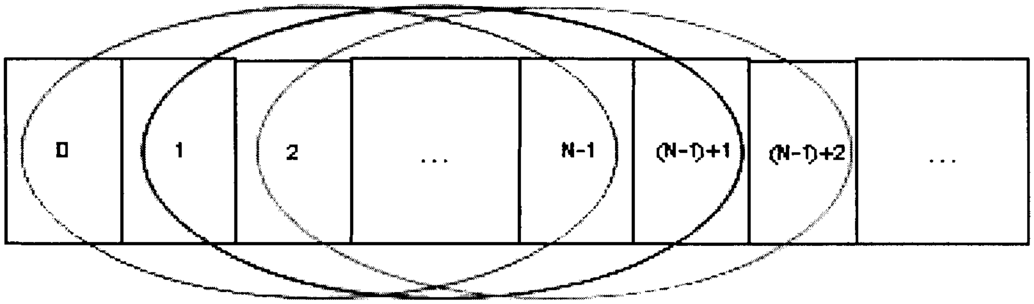

Figure 3-3. Arrangement of the overlapping blocks in a DFDD receiver.

3.3 MSDSD

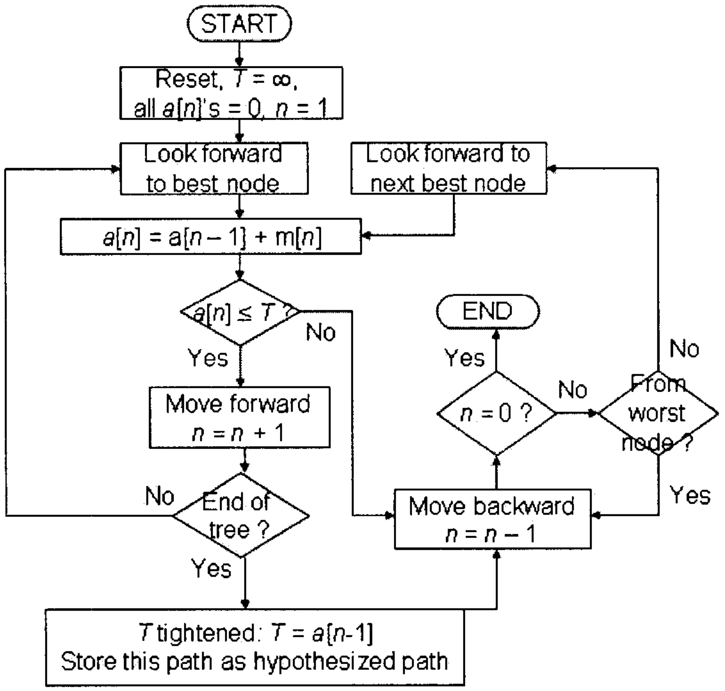

Figure 3-4. Flowchart of a sphere decoder.

The sphere decoding algorithm is originally proposed as an efficient searching algorithm for solving the closest lattice point problem [27]. Recently, this decoding strategy has been extensively adopted in solving many MIMO communication problems [28]-[29]. The fundamental idea behind sphere decoding algorithm is that the decoder only examines the candidate vectors $\hat{\mathbf{c}} \in S^{N-1}$ that are lying within a sphere with a certain radius, rather than searches over all the candidate vectors in the entire vector space. Figure 3-4 illustrates the flowchart of a sphere decoder.

As discussed earlier, this vector space can be visualized as an M-ary search tree with a depth of $N - 1$. When the sphere decoding process begins, the decoder starts at the root node with a threshold T initially set to infinity[2]. This threshold T is only updated when the sphere decoder reaches to the last tree level. If the sphere decoder is examining a path which has its APM, $a[n]$, greater than the threshold, the decoder immediately stops searching along that path and starts a new search again. The process continues until no new search is possible. The way the sphere decoder terminates its search makes it optimal. However, it will be shown in Section 3.5 that the computational complexity of the sphere decoder grows exponentially with decreasing SNR while all of our Fano-MSDD receivers can maintain a steady complexity versus SNR characteristics.

Note that the implementation of the sphere decoder in Figure 3-4 is different from the one in [10]. Our MSDSD receiver is implemented by employing the same tricks and assumptions as we implement the Fano-MSDD receiver. Now the sphere decoder adopts a sum of the branch metrics shown in (35) as its navigation tool rather than the one given by a triangular matrix [10]. The reason for the change is that the computational complexity of a receiver strongly depends on the programming tricks and other resources (such as look-up tables). This ensures that a fair assessment is made while comparing the complexity of the sphere decoder with the Fano detectors. It will be shown in the simulation that our implementation of the MSDSD requires only half as many computations as that reported in [10].

3.4 The Fano-MSDDs

The Fano decoding algorithm [30]-[32] is a well-established sequential decoding algorithm commonly used in decoding convolutional codes. Other applications include

[2] In order to maintain consistency with the Fano algorithm, the term threshold is analogous to the radius of a sphere decoder.

linear block code decoding [32]-[34], channel coefficient estimation [35], and blind equalization [36]. Although suboptimal, it is a more suitable choice than the optimal Viterbi algorithm (VA) when the constraint length is large. While the complexity of the VA grows exponentially with the constraint length, the complexity of the Fano algorithm grows only linearly with the depth of the decoding tree N (i.e. block size in our case). The fundamental idea behind the Fano decoding algorithm is that in the decoding process, only the most promising paths are explored. If a branch to a node looks bad, the algorithm can back up to the ancestor nodes and try different paths. This is accomplished by monitoring the APM against a running threshold T that can be varied in steps of Δ. The running threshold T can be tightened (i.e. decreased) or relaxed (i.e. increased) depending on the results of the comparison. The ability to backtrack is one of the key features that distinguish the Fano decoder from the DFDD, and the one that allows the decoder to deliver close to ML performance with a modest computational complexity.

In the subsequent sub-sections, we first present the original Fano search algorithm and then propose an improved version, which utilizes a new threshold adjustment scheme that can significantly reduce the computational complexity. Furthermore, two variants of the Fano algorithm, the Bi-Fano and the Edge-Bi-Fano algorithms, are introduced.

3.4.1 Original Fano search

Due to the nature of the Fano algorithm, it is more convenient to describe the search procedure by adopting the tilted or biased metric

$$M_b\left(\hat{\mathbf{g}}[N-1]\right) = \sum_{n=1}^{N-1} m_b[n], \tag{41}$$

where

$$m_b[n] = m[n] - b \tag{42}$$

is the biased branch metric at time index n, and b is a bias that has to be chosen properly. A larger bias translates into a larger tolerance to the statistical fluctuation in the metric. The biased APM up to time n is defined as

$$v[n] = \sum_{k=1}^{n} m_b[k] = v[n-1] + m_b[n], \tag{43}$$

where $v[N-1]$ is the same as $M_b(\hat{\mathbf{g}}[N-1])$. When the Fano decoder is tracking the correct transmitted pattern, each branch metric $m[n]$ has an expected value of unity as discussed. Consequently, the average slope of $v[n]$ is $(1-b)$, where b is set to a number greater than 1 so that the slope is negative. In other words, for the correct path, $v[n]$ tends to decrease linearly as n increases. On the other hand for an incorrect path, some of the $m[n]$'s can be quite large because the erroneous symbols in the incorrect path cause abrupt transitions in the hypothesized fading-plus-noise pattern, causing large prediction errors. As a result, the slope of biased APM is positive and $v[n]$ tends to increase with n. Figure 3-5 depicts the transformation of the original positive-distance tree into a tilted distance tree by adopting a bias b greater than unity. Note that the Fano decoder is now searching for the data pattern with the most negative biased path metric.

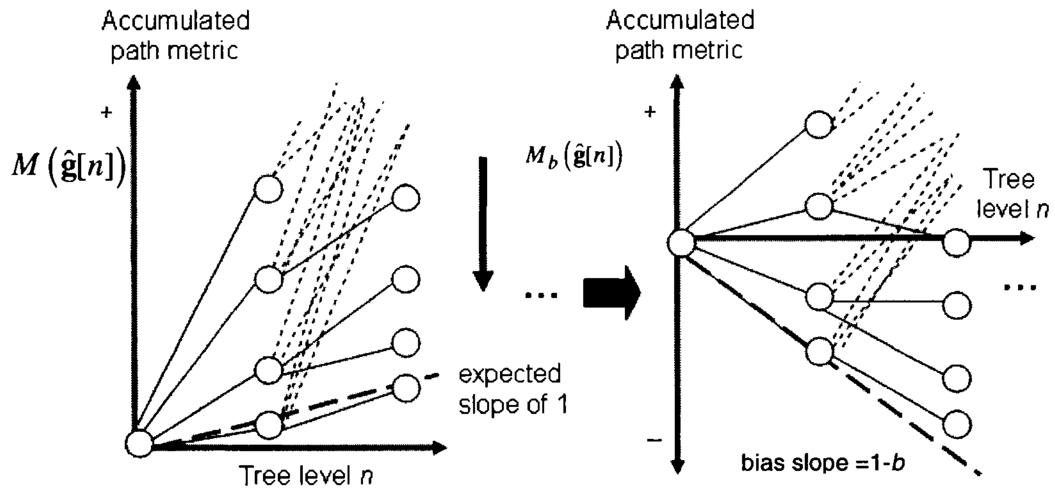

Figure 3-5. Tilted tree as a result of a bias slope.

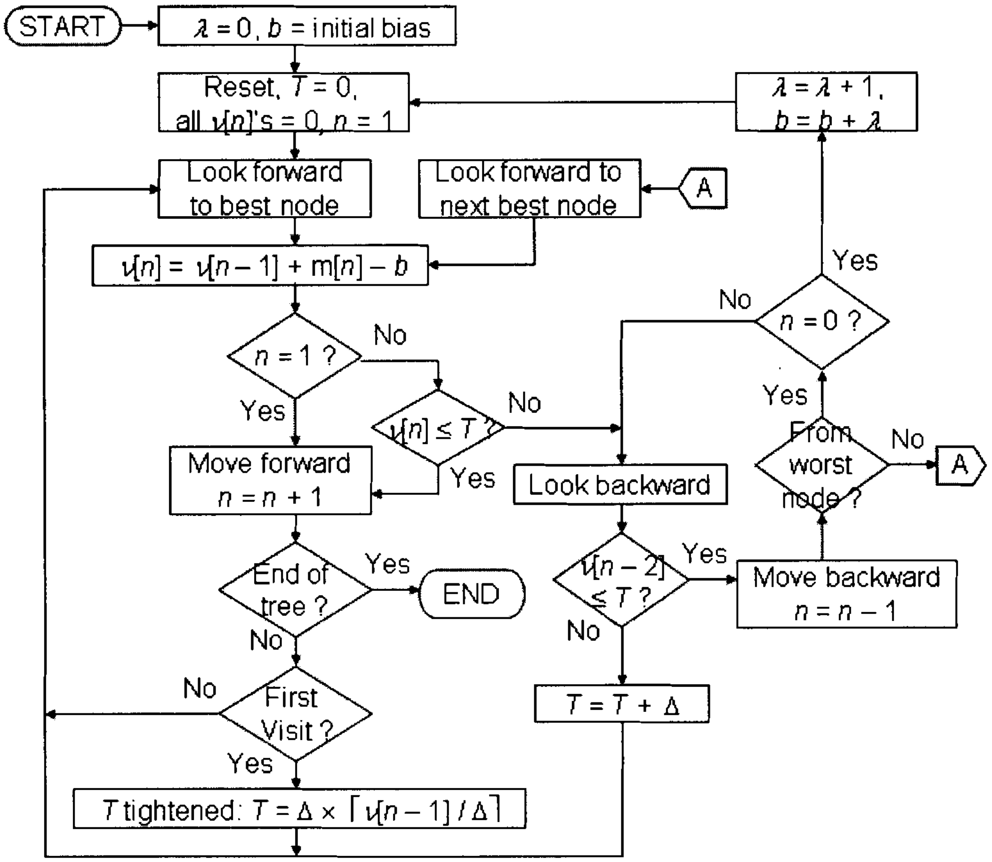
START
$\lambda = 0$, b = initial bias
Reset, $T = 0$, all $v[n]$'s = 0, $n = 1$
$\lambda = \lambda + 1$, $b = b + \lambda$
Look forward to best node
Look forward to next best node
A
$v[n] = v[n-1] + m[n] - b$
Yes
No
$n = 0$?
$n = 1$?
No
Yes
$v[n] \leq T$?
No
Yes
Yes
From worst node ?
No
A
Move forward $n = n + 1$
Look backward
End of tree ?
Yes
END
No
$v[n-2] \leq T$?
Yes
No
Move backward $n = n - 1$
No
First Visit ?
Yes
$T = T + \Delta$
T tightened: $T = \Delta \times \lceil v[n-1] / \Delta \rceil$

Figure 3-6. Flowchart for the original Fano algorithm.

The decoding process begins at the root node (i.e. tree level 0 or equivalently n = 1) and terminates at one of the nodes at the end of the tilted tree (i.e. tree level $N - 2$). Say the decoder is now located at some node N_D at tree level L (i.e. $n = L + 1$). From that node the decoder *looks*[3] forward to its M succeeding nodes at tree level n. If the best succeeding node has a biased APM $v[n]$ that is less than or equal to the running threshold T, the decoder will *move* to that node. Furthermore, if that successor node is visited the first time, the threshold will be tightened to the quantized level just above $v[n]$. On the other hand if $v[n]$ is greater than T, the decoder *looks* backward to its ancestor node at tree level $L - 1$ and checks if its biased APM $v[n-2]$ violates T. If it is also greater than T, T is loosened by a step size of Δ and the decoder *looks* forward again in an attempt to *move* forward. If $v[n-2]$ is less than T, the decoder recognizes an unlikely path and *moves* backward to the ancestor node. If node N_D was the worst node when the decoder *moved* forward from its ancestor node at some early decoding stage, the decoder continues to *move* backward. Otherwise, the decoder searches along another path by *looking* forward to the next best node. If the decoder reaches the end of the tree, the decoding process terminates and the path is taken as the decoded path (or estimated sequence). Figure 3-6 illustrates the overall operation of a Fano decoder.

[3] The two actions *looking* and *moving* are needed to be distinguished in the Fano algorithm. When the decoder *looks* at a node, it means evaluating the metric at that node. The decoder *moves* to that node only if certain criteria are met (or not met).

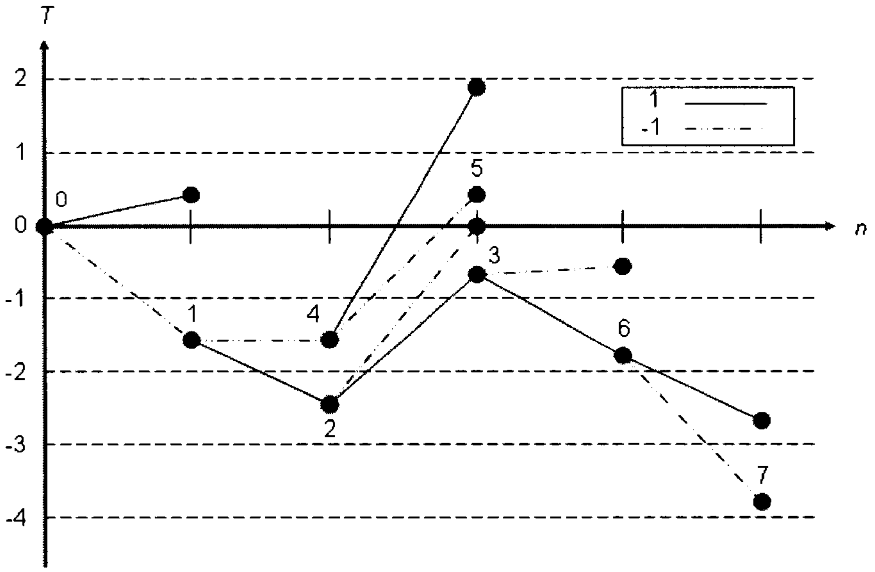

Figure 3-7. A Fano search for BPSK symbols with $b = 3$, $\Delta = 1$, and $N = 6$.

Table 3-1. Actions taken for the example shown in Figure 3-7.

Node	n	T	m[n]	$v[n\text{-}2]$	$v[n\text{-}1]$	$v[n]$	Action (x indicates violation)				
0: [1]	1	0	[-1]: 1.5	***	0	-1.5	look at 1		point to 1		set T = -1
1: [1,-1]	2	-1	[1]: 2.1	0	-1.5	-2.4	look at 2		point to 2		set T = -2
2: [1,-1,1]	3	-2	[1]: 4.8	-1.5	-2.4	-0.6	look at 3	x	look at 1	x	set T = -1
2: [1,-1,1]	3	-1	[1]: 4.8	-1.5	-2.4	-0.6	look at 3	x	look at 1		point to 1
1: [1,-1]	2	-1	[-1]: 3	0	-1.5	-1.5	look at 4		point to 4		set T = -1
4: [1,-1,-1]	3	-1	[-1]: 5	-1.5	-1.5	0.5	look at 5	x	look at 1		point to 1
1: [1,-1]	2	-1	***	0	-1.5	***	look at 0	x	set T = 0		
1: [1,-1]	2	0	[1]: 2.1	0	-1.5	-2.4	look at 2		point to 2		
2: [1,-1,1]	3	0	[1]: 4.8	-1.5	-2.4	-0.6	look at 3		point to 3		set T = 0
3: [1,-1,1,1]	4	0	[1]: 1.8	-2.4	-0.6	-1.8	look at 6		set T = -1		
6: [1,-1,1,1,1]	5	-1	[-1]: 1	-0.6	-1.8	-3.8	look at 7		output [1,-1,1,1,1,-1]		

An illustrative example of the search algorithm is provided in Figure 3-7 and Table 3-1 for BPSK. The first column of the table contains not only the current node where the decoder pointer is located, but also the array of tentative decisions up to that point in the decoding tree. Without loss of generality, the first tentative decision is always 1 because of the way our simulation is set up. The fourth column of the table contains the branch metric as well as the corresponding transmitted symbol. The output pattern in the last entry of the table will be differentially decoded to yield the corresponding information pattern. The rest of the table and the figure are self-explanatory.

In fact, the tree search in MSDSD is very similar to the one in the Fano-MSDD except that the threshold T of the MSDSD can only be tightened. Another difference is that the sphere decoder searches in a non-decreasing tree as shown in Figure 3-2 instead of a biased tree as in the Fano algorithm. Furthermore, the sphere decoder updates its threshold only when it reaches to the last tree level while the Fano decoder updates its threshold each time it moves to another node and the threshold can be either tightened or loosened dynamically. Consequently, MSDSD is guaranteed to find the ML solution over the block. Fano-MSDD does not provide any guarantee, but allows indefinite length and continuous overlap.

For both the sphere and the Fano decoders, when the decoder is penetrating a tree, it always looks to the best succeeding node to move forward to. If that turns out to be futile, it will examine the next best node, and so on. This means the decoder needs to sort the M outgoing branches of the current node from best to worst. Due to the nature of the MPSK constellation, this can be easily accomplished by first computing the phase angle $\theta[n] = \arg\left\{\left(\boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\right)\hat{\mathbf{g}}[n-1]r^*[n]\right\}$ and then substituting this angle to equations (39) and (40) to calculate the estimated transmitted symbol $\hat{s}[k]$ with the minimum branch metric $m[n]$. After this best outgoing branch is found, the other branches can be sorted quickly by zigzagging through the remaining angles, in phase to $\theta[n]$, based on the quotient and remainder of the rounding operation in (40). There is no need to calculate all M branch metrics in this process. Figure 3-8 illustrates an example of the zigzagging operation for a QPSK constellation.

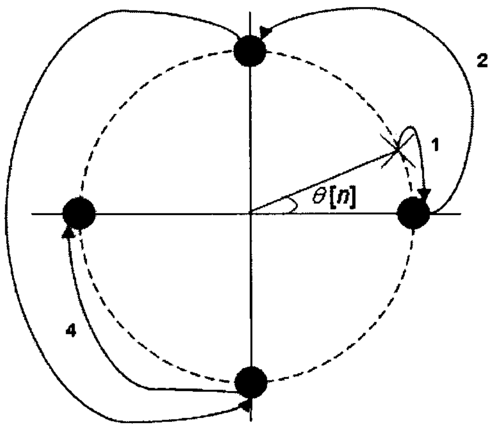

Figure 3-8. Zigzagging operation for a QPSK constellation.

In order to extract good performance at a reasonable complexity from the Fano decoder, the bias b as well as the step size Δ used to adjust the running threshold need to be sufficiently large to keep backtracking as infrequent as possible. Note that backtracking is usually caused by a deep fade in the detection block, as the APM will increase sharply when this occurs. Even with careful choices, there are still chances that some biased branch metrics $m_b[n]$ along the correct path are so large that the receiver is not able to get to the end of the tree, ending up eventually at the tree's root node with no path to move forward to. This event corresponds to a search failure.

Specifically, search failures happen when some biased APMs $v[n]$'s, $n = 1..N-1$, along the correct path are greater than that of the root node (i.e. $v[n] > 0$). For example, as shown in Figure 3-9, the sharp increase in the biased branch metric of the correct path at tree level $n = 2$ causes the corresponding biased APM $v[2]$ to be greater than zero. Based on the Fano algorithm shown in Figure 3-6, the decoder will eventually stay at the root node with no other paths to penetrate forward. When this happens, another search, using a larger bias b, is required. Our Fano algorithm in Figure 3-6 first assigns b an initial value and gradually increases it while search failures are detected. The bias increment is intuitively set to the number of search failures λ. Note that, without any loss of generality, the number of bias re-adjustment is equal to the number of search failures λ.

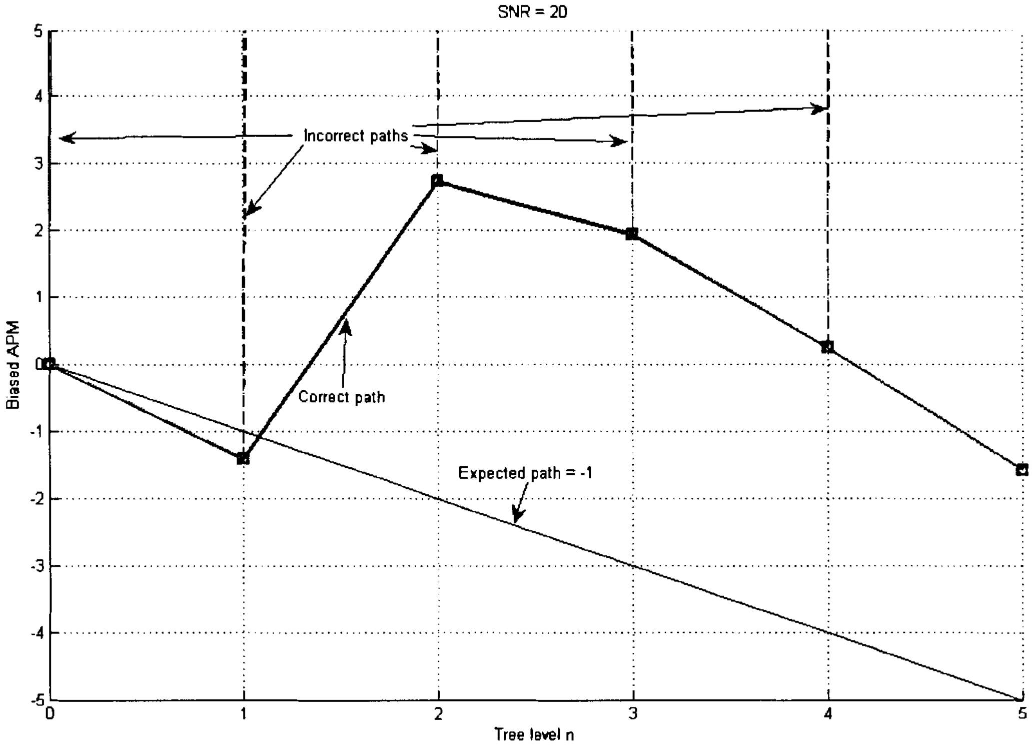

Figure 3-9. A deep fade happened at the beginning of a block.

Intuitively, we can deduce that search failures happen more often when deep fades occur at the beginning of the decoding block. This stems from the fact that, at the root portion of the tree, there is less room for statistical fluctuation and backtracking. A deep fade happened near the root node can cause the biased APMs in that neighbourhood to increase dramatically that the biased APMs can be greater than zero (see Figure 3-9) and, hence, search failures can occur. However, when deep fades happen at the end of the decoding block, the decoder is able to penetrate deep enough to allow more room for statistical fluctuations. Figure 3-10 illustrates an example where a deep fade occurs at the end of the observation window. In sum, we can conclude that the probability of a search failure is closely related to the probability of a deep fade occurred at the beginning of the observation window and it will be verified by the simulation results shown in Section 3.5.3.

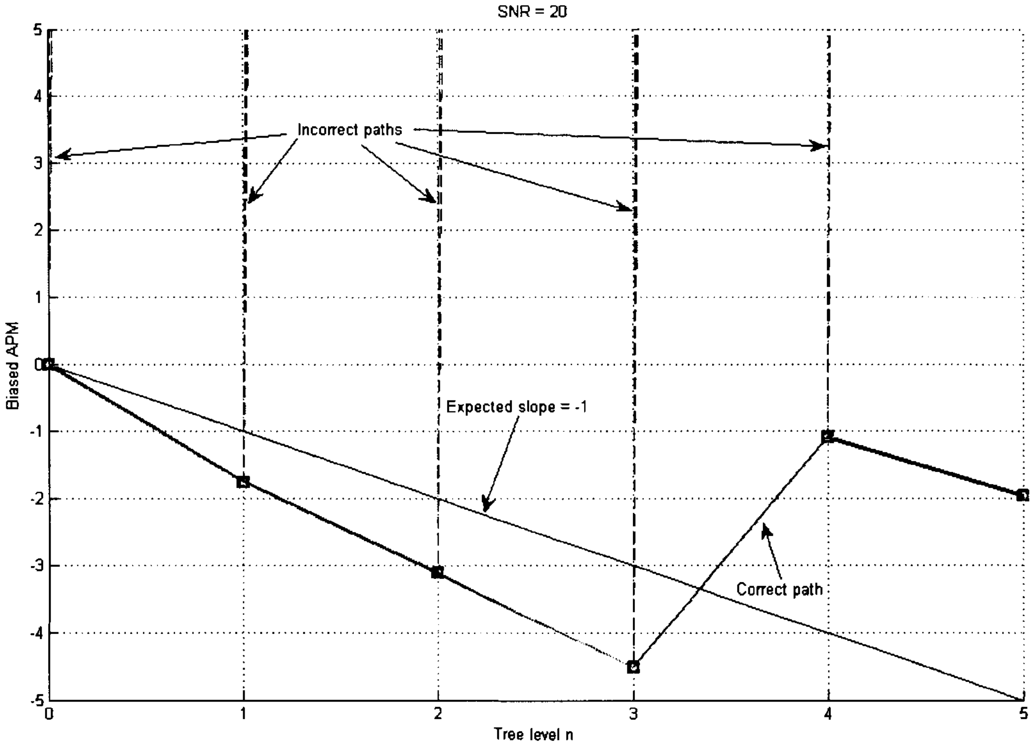

Figure 3-10. A deep fade happened at the end of a block.

In addition, the number of search failures is dependent on SNR. At large SNR, the correct path becomes more prominent in terms of its APM (see Figure 3-11) and, in absence of deep fades, the Fano decoder can always penetrate through this path without any backtracking. However, when deep fades occur, some branch metrics along this path can become so large that some backtracking is required. Since other paths look even worse in terms of APM (see Figure 3-9), the decoder eventually happens to be travelling back-and-forth exclusively along this path and, in the worst case, the decoder may end up staying at the root node. However, when SNR is low, search failures are not as frequent because all paths lie close to each other (see Figure 3-12). When the Fano decoder is searching along a path that has a biased APM too large to penetrate through, the decoder can simply backtrack to its preceding nodes and then try other paths. Subsequently, the decoder has a higher probability to pick a path, which can lead to the end of the tree. Note that this decoded path may not necessarily be the correct path.

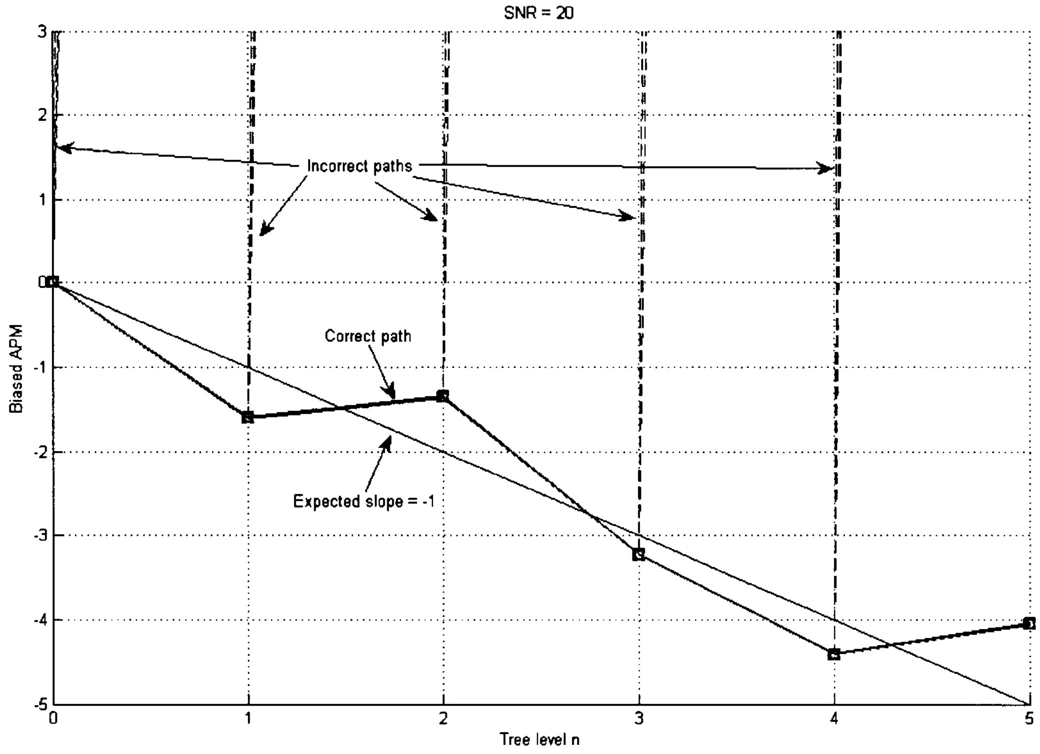

Figure 3-11. Example of the tilted tree at SNR = 20 dB.

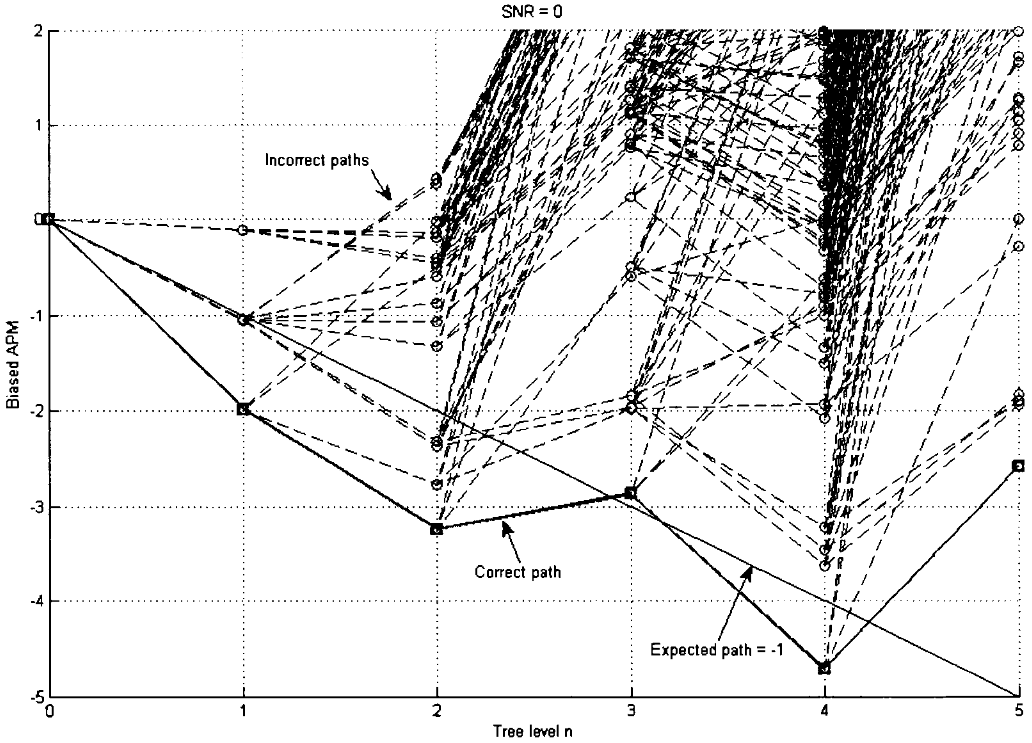

Figure 3-12. Example of the tilted tree at SNR = 0 dB.

3.4.2 Improved Fano search

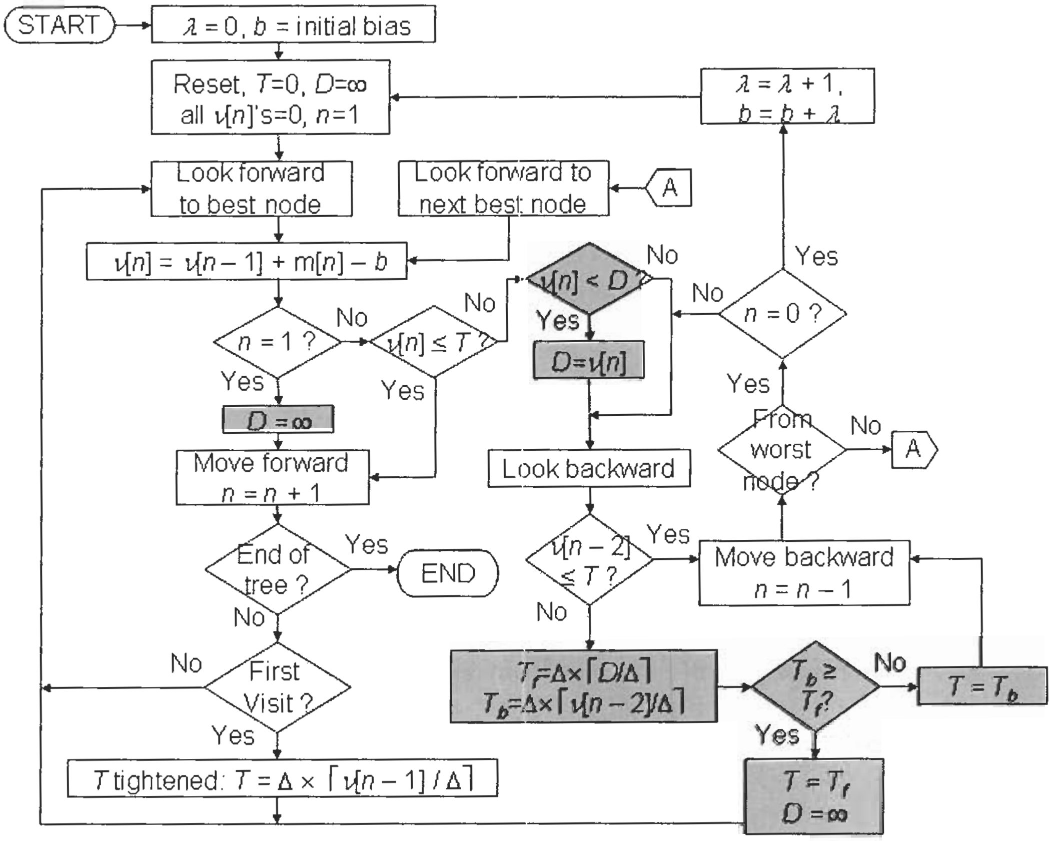

Figure 3-13. Flowchart of the improved Fano MSDD with a step size of Δ.

The flowchart of an improved Fano-MSDD is shown in Figure 3-13. It is similar to the original algorithm in Figure 3-6, except that new measures are introduced to reduce the computation complexity. These changes/additions are highlighted in the figure. One of the key changes to the original algorithm is the introduction of the parameter D. This parameter stores the smallest biased APM (43) that had previously violated the running threshold T. It helps to avoid unnecessary back-and-forth node transitions when the decoder is looking for the minimal threshold that allows it to penetrate through the decoding tree. It also helps in reducing the number of steps involved in relaxing the threshold when a branch violates the threshold by more than one step size Δ during a forward movement. These advantages are most apparent when Δ is relatively small.

An illustrative example of the improved Fano search strategy is provided in Figure 3-14 and Table 3-2. In addition, Table 3-3 lists the actions that the decoder would take if the original algorithm (Section 3.4.1) was employed instead. From the perspective of the improved algorithm, Steps 5-7 in Table 3-3 are unnecessary node transitions. Step 12 in the same table shows a situation when a forward search results in a biased APM that violates the running threshold by two step sizes. Thus for this particular example, the improved Fano algorithm saves a total of four steps in metric calculations and comparisons. In actual simulations, the improved algorithm also provides very significant saving in complexity.

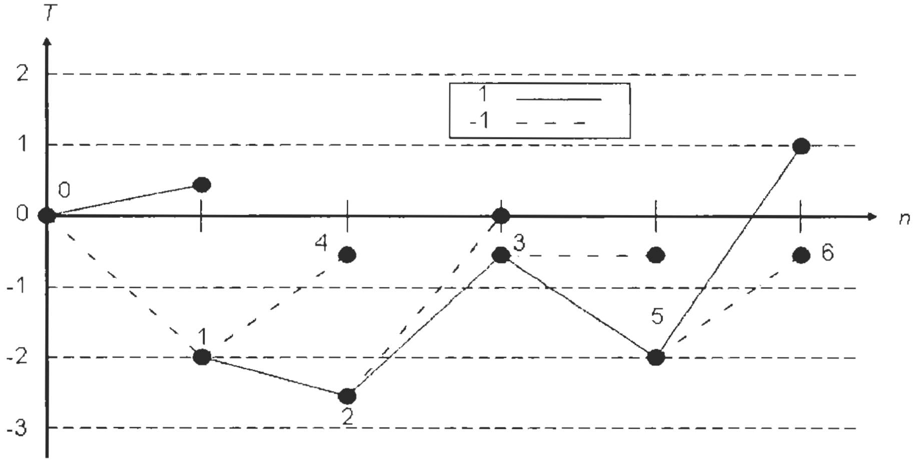

Figure 3-14. A Fano search for BPSK symbols with $b = 3$, $\Delta = 1$, and $N = 6$.

Table 3-2. Actions taken by the improved Fano search.

	Node	n	T	m[n]	$v[n-2]$	$v[n-1]$	$v[n]$	D	Tf	Tb	Action (x indicates threshold violation)
1	0: [1]	1	0	[-1]: 1	***	0	-2	∞	***	***	look at 1 point to 1 set T = -2
2	1: [1,-1]	2	-2	[1]: 2.5	0	-2	-2.5	∞	***	***	look at 2 point to 2 set T = -2
3	2: [1,-1,1]	3	-2	[1]: 5	-2	-2.5	-0.5	-0.5	0	-2	look at 3 x look at 1 point to 1
4	1: [1,-1]	2	-2	[-1]: 4.5	0	-2	-0.5	-0.5	0	0	look at 4 x look at 0 x set T = 0, set D = ∞
5	1: [1,-1]	2	0	[1]: 2.5	0	-2	-2.5	∞	***	***	look at 2 point to 2
6	2: [1,-1,1]	3	0	[1]: 5	-2	-2.5	-0.5	∞	***	***	look at 3 point to 3 set T = 0
7	3: [1,-1,1,1]	4	0	[1]: 1.5	-2.5	-0.5	-2	∞	***	***	look at 5 point to 5 set T = -2
8	5: [1,-1,1,1,1]	5	-2	[-1]: 4.5	-0.5	-2	-0.5	-0.5	0	0	look at 6 x look at 3 x set T = 0, set D = ∞
9	5: [1,-1,1,1,1]	5	0	[-1]: 4.5	-0.5	-2	-0.5	∞	***	***	look at 6 output [1,-1,1,1,1,-1]

Table 3-3. Actions taken by the original Fano search.

	Node	n	T	m[n]	$v[n-2]$	$v[n-1]$	$v[n]$	Action (x indicates threshold violation)
1	0: [1]	1	0	[-1]: 1	***	0	-2	look at 1 point to 1 set T = -2
2	1: [1,-1]	2	-2	[1]: 2.5	0	-2	-2.5	look at 2 point to 2 set T = -2
3	2: [1,-1,1]	3	-2	[1]: 5	-2	-2.5	-0.5	look at 3 x look at 1 point to 1
4	1: [1,-1]	2	-2	[-1]: 4.5	0	-2	-0.5	look at 4 x look at 0 x set T = -1
5	**1: [1,-1]**	**2**	**-1**	**[1]: 2.5**	**0**	**-2**	**-2.5**	**look at 2 point to 2**
6	**2: [1,-1,1]**	**3**	**-1**	**[1]: 5**	**-2**	**-2.5**	**-0.5**	**look at 3 x look at 1 point to 1**
7	**1: [1,-1]**	**2**	**-1**	**[-1]: 4.5**	**0**	**-2**	**-0.5**	**look at 4 x look at 0 x set T = 0**
8	1: [1,-1]	2	0	[1]: 2.5	0	-2	-2.5	look at 2 point to 2
9	2: [1,-1,1]	3	0	[1]: 5	-2	-2.5	-0.5	look at 3 point to 3 set T = 0
10	3: [1,-1,1,1]	4	0	[1]: 1.5	-2.5	-0.5	-2	look at 5 point to 5 set T = -2
11	5: [1,-1,1,1,1]	5	-2	[-1]: 4.5	-0.5	-2	-0.5	look at 6 x look at 3 x set T = -1
12	**5: [1,-1,1,1,1]**	**5**	**-1**	**[-1]: 4.5**	**-0.5**	**-2**	**-0.5**	**look at 6 x look at 3 x set T = 0**
13	5: [1,-1,1,1,1]	5	0	[-1]: 4.5	-0.5	-2	-0.5	look at 6 output [1,-1,1,1,1,-1]

3.4.3 Bi-Fano MSDD

Consider the path and branch metrics in (35) and (36). They were derived under the implicit assumption that the decoder searches forward in time (i.e. from time $n = 1$ to $n = N - 1$) for the data pattern that best matches the received pattern. As far as an ML MSDD is concerned, it makes no difference whether the decoder searches forward or backward (from time $n = N - 2$ to $n = 0$) in time, since by definition, this receiver evaluates all possible data patterns within the detection window. However, for a Fano-MSDD, a forward search and a backward search may not yield identical results, since the algorithm is suboptimal. Another reason for the potential discrepancy is that different predictors (and different observations) are being used to predict the fading gains affecting the same data symbols when searching in different directions. For example, the fading-plus-noise sample $g[n]$ is implicitly estimated by an n-th order predictor during the forward search, and by an $(N - 1 - n)$-th predictor during the backward search. The reader is referred to Appendix A for the derivation of the ML metric in the form of backward linear predictions.

The above observation leads us to introduce a bi-directional Fano-MSDD (Bi-Fano MSDD) that uses the outcomes of the forward and backward searches to perform error detection and error correction. Let $\left(\hat{c}_f[1], \hat{c}_f[2], ..., \hat{c}_f[N-1]\right)$ and $\left(\hat{c}_b[1], \hat{c}_b[2], ..., \hat{c}_b[N-1]\right)$ be the decisions provided by the forward and the backward decoders respectively. If the two sequences are identical, then they become the final decision. On the other hand, if discrepancies occur at positions $\{n_1, n_2, \, ... \, , n_d\}$, then the decoder examines all possible sequences of the form $\{\tilde{c}_K[1],\ \tilde{c}_K[2],\ \ldots,\ \tilde{c}_K[N-1]\}$, $K = 0,\ 1,\ \ldots,\ 2^{n_d}-1$, where

$$\tilde{c}_K[n] = \begin{cases} x[n], & n = n_i \\ \hat{c}_f[n], & n \neq n_i, \end{cases} \tag{44}$$

$$x[n_i] = \begin{cases} \hat{c}_f[n_i], & b_i = 0 \\ \hat{c}_b[n_i], & b_i = 1, \end{cases} \tag{45}$$

and $\{b_1,\ b_2,\ \ldots,\ b_{n_d}\}$ is the binary equivalent of the integer K. There are altogether 2^{n_d} such sequences and the one with the smallest metrics (41) is released by the decoder as the final decision. The simulation results show that this bi-directional search technique provides a substantial improvement to the bit-error performance when compared to the forward Fano-MSDD. The computational complexity, on the other hand, is approximately doubled at large SNR.

3.4.4 Edge-Bi-Fano MSDD

It is observed that for the forward Fano-MSDD, data estimates near the end of the block (i.e. $n = N-1$) have a higher error probability than those at the beginning. This stems from the fact that the decoder will not be able to discover soon enough when it makes a wrong turn in the decoding tree while searching in that neighbourhood. The opposite is also true for a backward Fano-MSDD. To mitigate the problem, we propose an edge-cut version of the Bi-Fano algorithm with an effective block size of $N_e = N + E$, where $N-1$ is the actual number of symbols the algorithm wants to decode, and $E = 2H$, is the number of symbols whose decisions will be discarded. Figure 3-15 depicts the arrangement of the overlapping blocks for the new detector. This improved detector, termed an Edge-Bi-Fano MSDD, first searches in both the forward and the backward directions and arrives at the tentative decisions $\{\hat{c}_f[1],\ \hat{c}_f[2],\ \dots,\ \hat{c}_f[N_e-1]\}$ and $\{\hat{c}_b[1],\ \hat{c}_b[2],\ \dots,\ \hat{c}_b[N_e-1]\}$. Then the first and the last H symbols in each of these sequences are dropped. The surviving sequences, $\{\hat{c}_f[H+i]\}_{i=1}^{N-1}$ and $\{\hat{c}_b[H+i]\}_{i=1}^{N-1}$ are then used to construct the final decision space as per (44) and (45). After the transmitted pattern of the current observation window has been decoded, the decoder slides its window forward by $N-1$ symbols and starts another decoding process with the same effective windows size $N_e = N + E$. It will be shown in the next section that the Edge-Bi-Fano detector provides a substantial improvement over a Bi-Fano detector with the same block size.

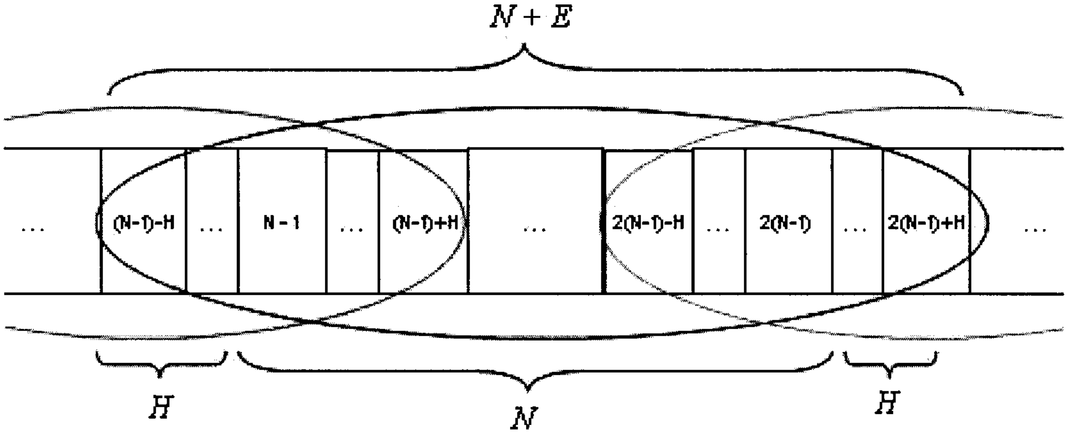

Figure 3-15. Arrangement of overlapping blocks in an Edge-Bi-Fano receiver.

3.5 Simulation Results and Discussion

We present in this section the bit-error performance and the implementation complexity of the three Fano sequence detectors described in the preceding sections. The BEPs and complexities of the MSDSD and the DFDD are used as performance benchmarks. Other statistical measures such as the average number of node transitions, the average number of search failures (or bias adjustments), and the probability of a differed symbol in the forward and the backward detected patterns, are also discussed. All simulation results are based on the QDPSK and the 8DPSK transmissions with detection block sizes of $N = 6$ and $N = 10$. Both fast $(f_d T_s = 0.03)$ and static $(f_d T_s = 0)$ fading are considered. The bit SNR used in our simulation follows the definition in (8) of Section 2.2.

All the Fano MSDD schemes are simulated using a bias of $b = 2$ and a step size of $\Delta = 3$. Note that we did not attempt to optimize these parameters because, in general, the more stringent search criteria (in terms of tolerance to metric fluctuation) yields better performance results but, in exchange, with higher complexity. In addition, we would like to caution the reader that the computational complexity can vary substantially from programmer to programmer because complexity strongly depends on implementation tricks and other assumptions.

3.5.1 Bit-Error Performance

The BEPs of the Fano-MSDDs, the MSDSD, and the DFDD are plotted in Figure 3-16 and Figure 3-17 for the fast fading channel, and in Figure 3-19 and Figure 3-20 for the static fading channel. Among the three Fano schemes, the forward Fano detector has the worst performance, the Bi-Fano detector is ranked second, while the Edge-Bi-Fano detector is the best. In addition, all of our Fano detectors are able to track the coherent bound whereas the DFDD detectors begin to exhibit a noticeable irreducible error floor at high SNR; see Figure 3-17 and Figure 3-18. Note that the initial transmitted pattern $\left[s[0]\ s[1]\ \ldots\ s[N-2]\right]^T$ is assumed to be known to the DFDD detector so the error curves shown in the plots can be referred to as the performance lower bounds for the DFDD.

In the fast fading channel, the Edge-Bi-Fano detector has almost identical performance as the MSDSD when the block size equals to $N = 6$. At a BEP of 10^{-3}, it is approximately 0.6 dB more power efficient than the Bi-Fano detector, 2 dB more efficient than the forward Fano detector, and 2.5 dB more efficient than the DFDD. At the same BEP of 10^{-3}, when we increase the block size to $N = 10$, the degradation experienced by the Edge-Bi-Fano, Bi-Fano, forward Fano, and DFDD detectors, relative to the MSDSD, are 0.6 dB, 0.9 dB, 2.5 dB, and 3 dB respectively. In an 8DPSK system, similar degradations are observed except the forward Fano and the DFDD detectors are respectively 3 dB and 3.5 dB less power efficient than the MSDSD.

Although our primary interest is on the fast fading environment, to demonstrate the robustness of the Fano detectors, we show in Figure 3-19 and Figure 3-20 the bit-error performance of the sequence detectors in the static fading environment. As observed in the figures, the Fano-MSDDs are almost as good as the MSDSD. With a block size of $N = 6$, all of the sequence detectors have practically the same BEP, except that the forward Fano-MSDD is approximately 0.2 dB less power efficient than the others. When we increase the block size to $N = 10$, the BEP of the forward Fano detector improves by 0.2 dB while the other non-coherent detectors show only little improvement in power efficiency due to the diminishing return of an MSDD detector. The reader is reminded that in a static fading environment, one can simply use the Mackenthun's algorithm [7] because the computational complexity of this algorithm is independent of the SNR and it is only in the order of $N \log(N)$.

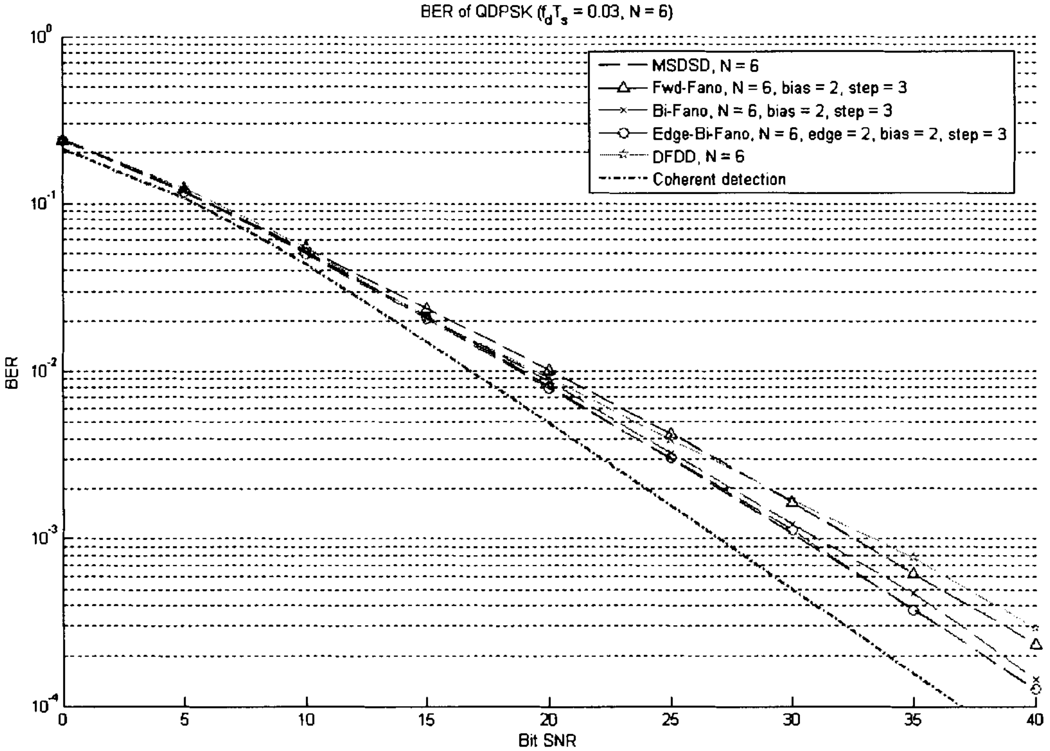

Figure 3-16. BEP versus bit SNR for QDPSK at $N = 6$ and $f_dT_s = 0.03$.

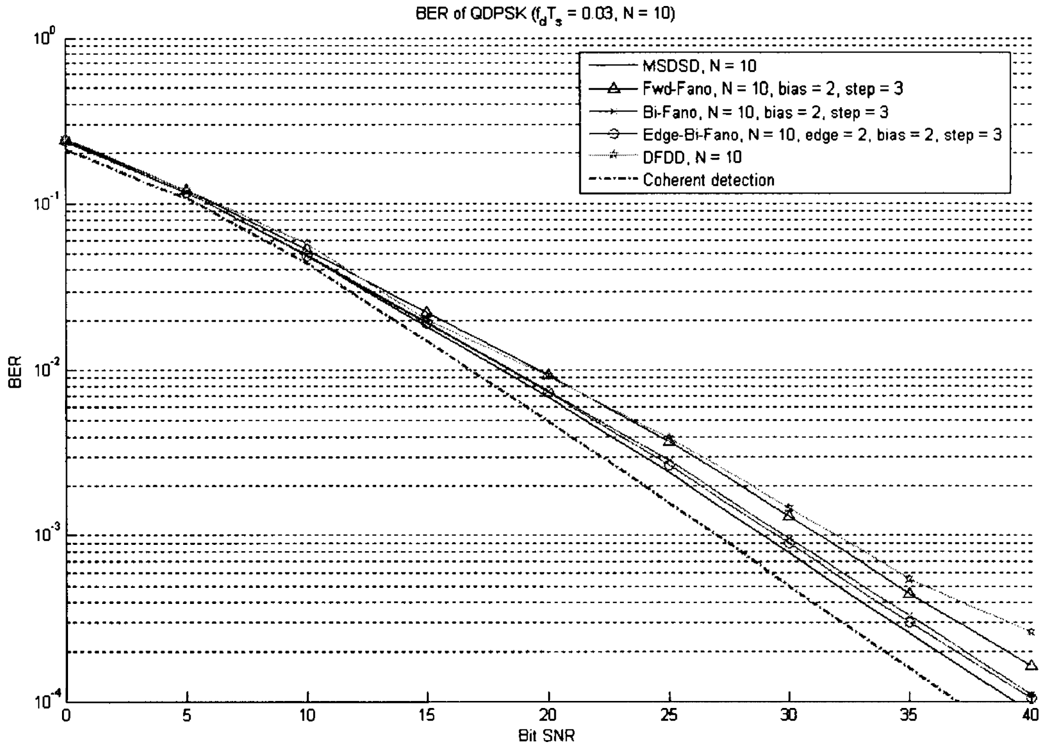

Figure 3-17. BEP versus bit SNR for QDPSK at N = 10 and f_dT_s = 0.03.

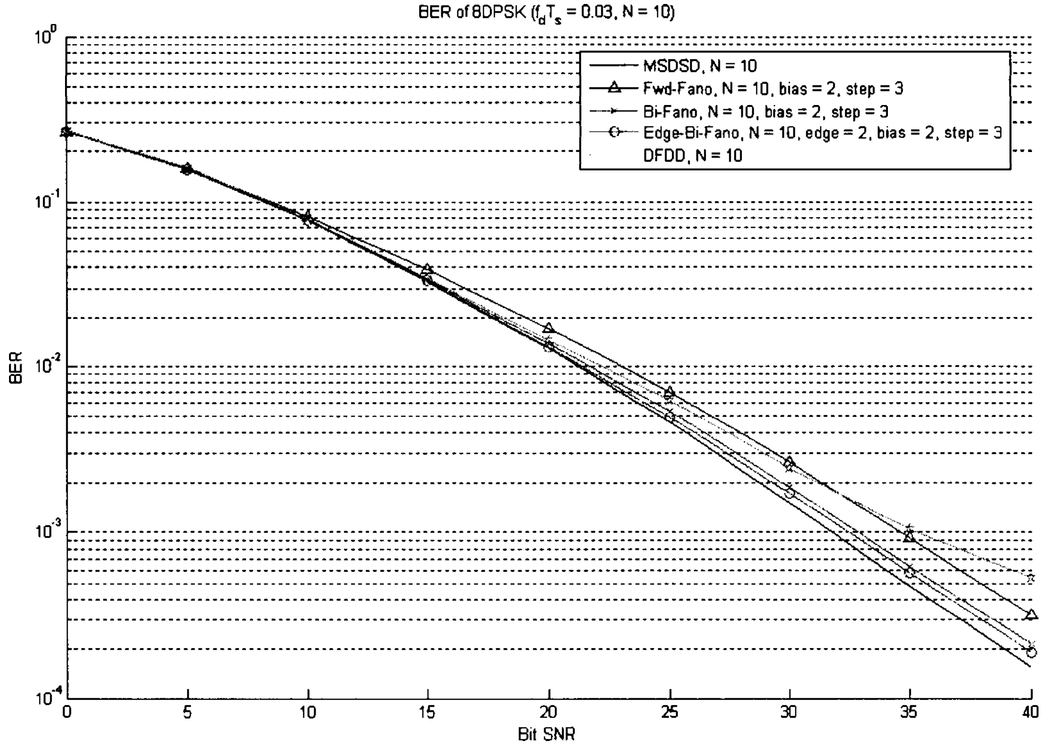

Figure 3-18. BEP versus bit SNR for 8DPSK at N = 10 and f_dT_s = 0.03.

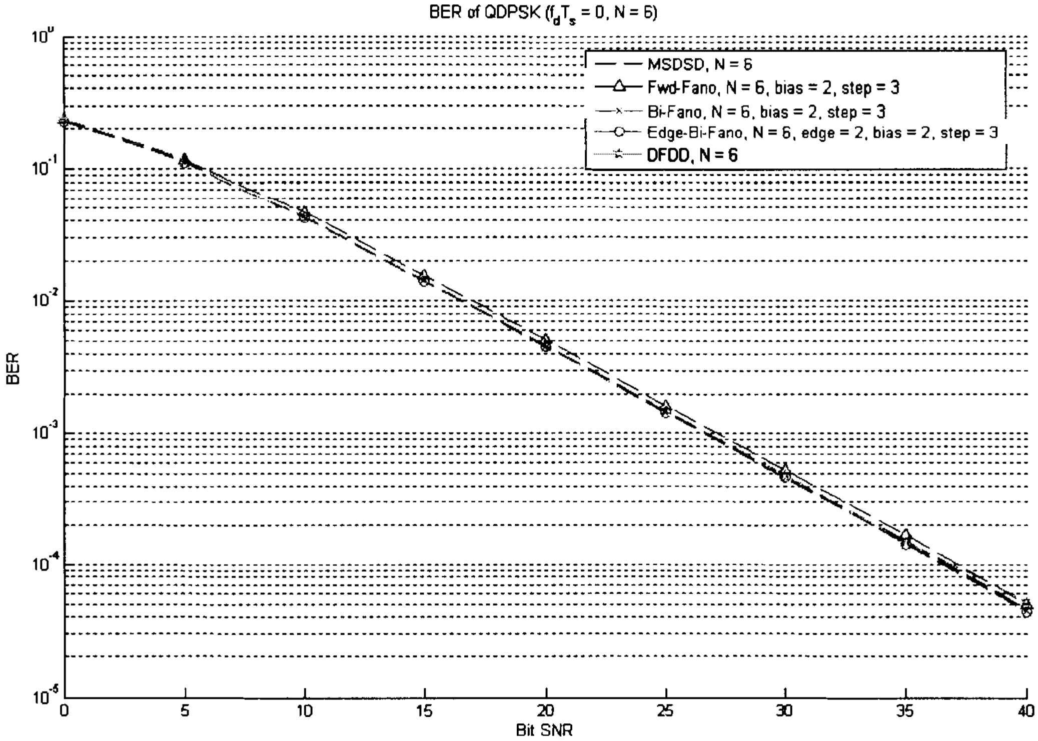

Figure 3-19. BEP versus bit SNR for QDPSK at N = 6 and static fading.

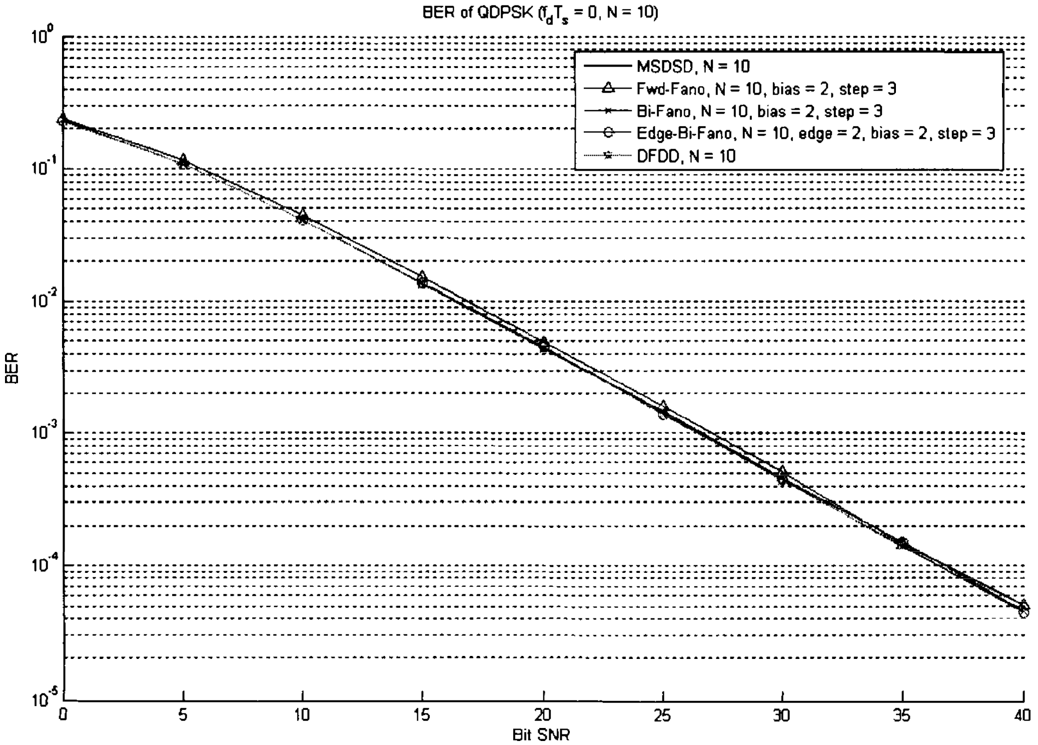

Figure 3-20. BEP versus bit SNR for QDPSK at N = 10 and static fading.

3.5.2 Computational Complexity

The average number of real multiplications and comparisons per block of N symbols are used as measures for the complexity and they are plotted in Figure 3-21 and Figure 3-23 for QDPSK and in Figure 3-22 and Figure 3-24 for 8DPSK. All of the complexity versus SNR curves are simulated under the assumption of a fast fading environment $(f_d T_s = 0.03)$ and the resultant curves are obtained by inserting counters at various place in the simulation programs. Note that the complexity of the MSDSD measured in the plots is only half of that reported in [10]. This stems from the fact that the MSDSD in the thesis is implemented based on the same programming tricks and resources we used to implement the Fano MSDD.

In general, we observe that our proposed Fano-MSDD receivers are computationally stable even at an SNR of 0 dB. They do not exhibit the exponential growth in complexity as seen in MSDSD when SNR decreases. For QDPSK transmission and a block size of $N = 10$, our Bi-Fano MSDD has lower complexity than the MSDSD when the SNR is below 20 dB. In terms of power efficiency, at the same SNR of 20 dB, the Bi-Fano MSDD is only 0.2 dB less efficient than the MSDSD. If we consider an 8DPSK transmission, the cross-over point is moved to 25 dB and the same performance loss of 0.2 dB is observed.

Also shown in the plots are the complexity curves for the DFDD. At large SNR, the forward Fano MSDD is very similar to a DFDD because backtracking does not happen very often and the decoder spends most of the time moving forward. From the perspective of the sphere decoder, the initial pass most likely provides a radius, which is small enough to narrow its search and hence to avoid any additional roaming in the tree.

In addition, since our Bi-Fano MSDD searches in both the forward and the reverse directions, it is observed that, at high SNR, our bi-directional decoder requires twice as many computations as the DFDD.

Note that when SNR is low, the Bi-Fano and the Edge-Bi-Fano both exhibit a growth in the number of multiplications. This is due to the increased number of discrepancies between the tentative decisions provided by the forward and the backward Fano detectors. At low SNR, all paths may look equally bad and there is insufficient information in the APM that enables the decoder to discriminate the transmitted pattern from the rest. As a consequence, more errors are made in the initial searches, which leads to more computations in the final decision making process.

In summary, the Bi-Fano detector is most suitable for channels at medium to low SNR as it is able to achieve close to ML performance, while its complexity is still computationally stable. However, in order to obtain reliable data transmission, medium to high SNR is usually of interest. At large SNR, the Bi-Fano detector suffers from a modest degradation in power efficiency when compared to the sphere decoder, but it requires twice as many computations. It will be shown in the next section that, for Bi-Fano and Edge-Bi-Fano, a backward search (or error correction) is not always required, as the forward and the backward searches may yield identical results, especially when SNR is large. We believe that further improvement can be made on the complexity of the Bi-Fano and Edge-Bi-Fano detectors if a smart way of performing bi-directional search on demand can be found. We will present in Chapter 5 some suggestions that can done to achieve this goal.

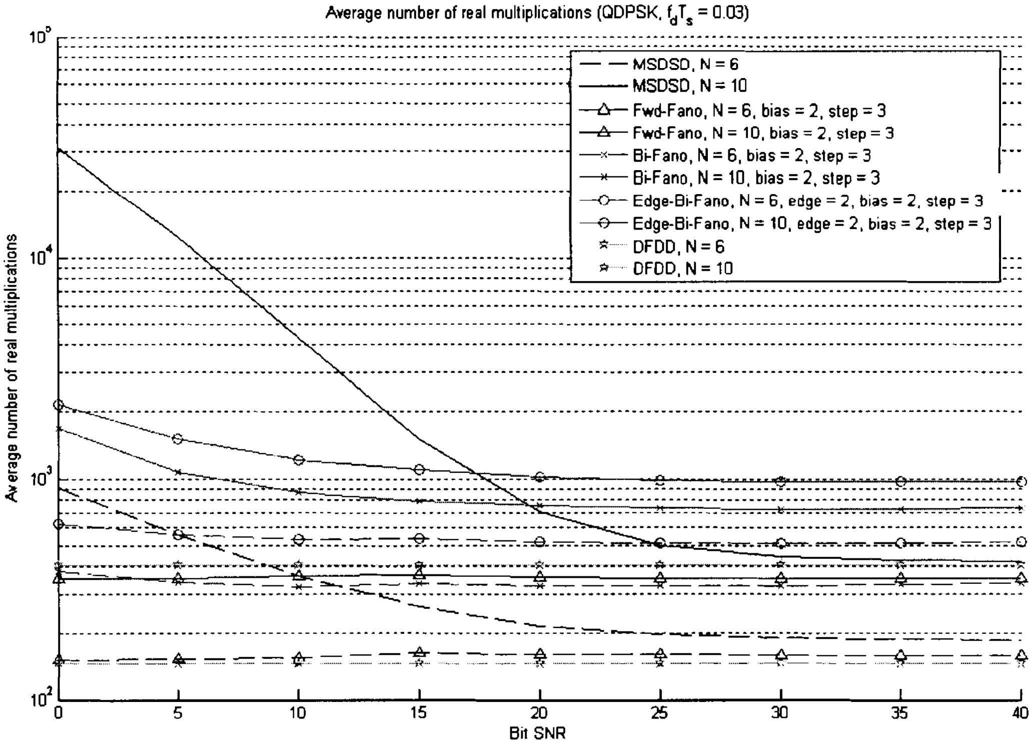

Figure 3-21. Average number of real multiplications versus bit SNR for QDPSK at f_dT_s = 0.03.

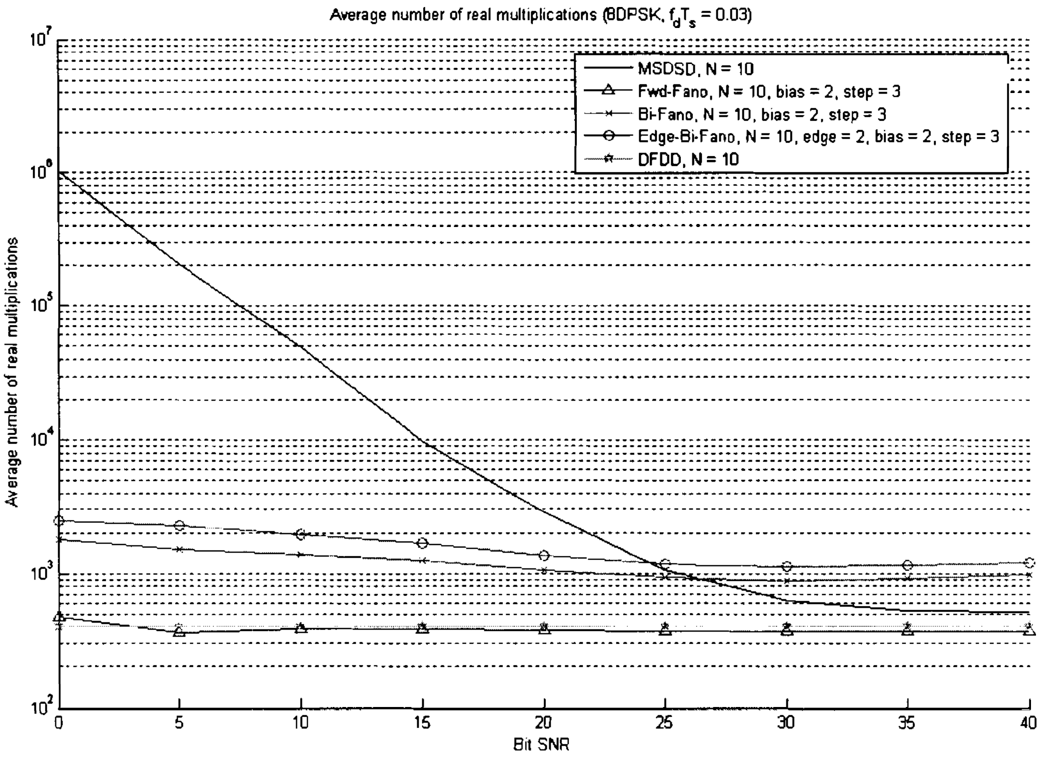

Figure 3-22. Average number of real multiplications versus bit SNR for 8DPSK at f_dT_s = 0.03.

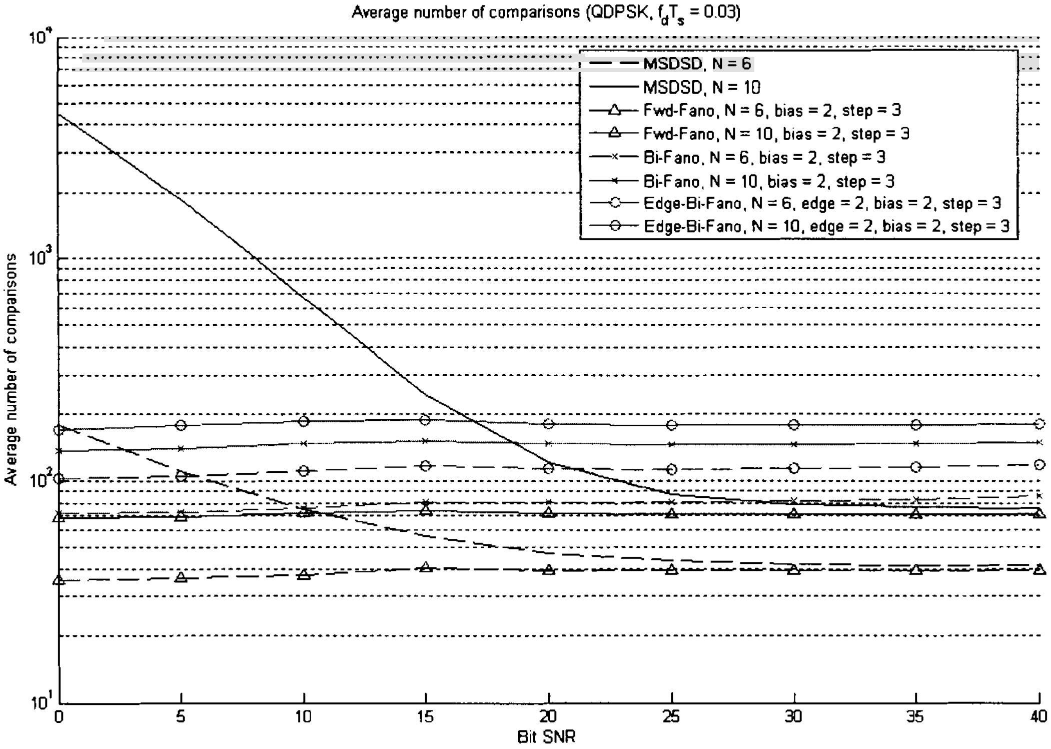

Figure 3-23. Average number of comparisons versus bit SNR for QDPSK at f_dT_s = 0.03.

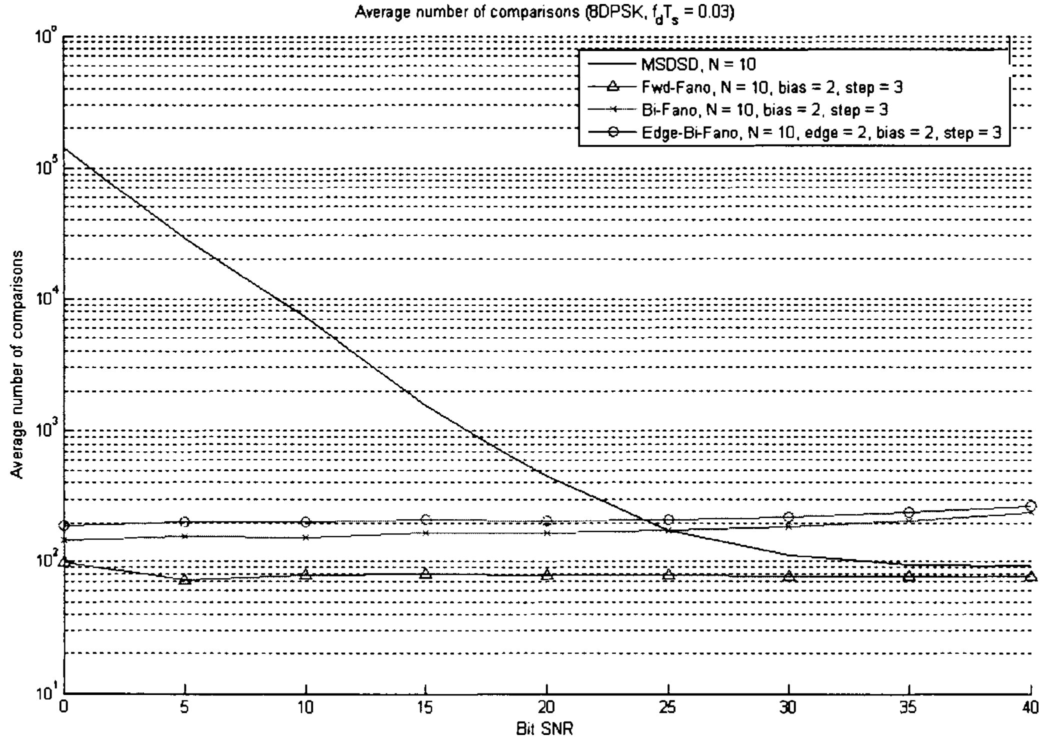

Figure 3-24. Average number of comparisons versus bit SNR for 8DPSK at f_dT_s = 0.03.

3.5.3 Other Statistical Measurements for Fano MSDD

Other than the BEP and the computational complexity, we have also taken other statistical measurements for our Fano-MSDDs. The average numbers of node transitions, search failures (or bias re-adjustments), and the probability of a differed symbol in the forward and the backward tentative decisions are plotted in Figure 3-25, Figure 3-27, and Figure 3-29 for QDPSK and in Figure 3-26, Figure 3-28, and Figure 3-30 for 8DPSK respectively. All of the results are obtained based on the fast fading scenario $(f_d T_s = 0.03)$.

It is interesting to see that, in both Figure 3-27 and Figure 3-28, there is a substantial growth in the number of bias re-adjustments for both the Bi-Fano and the Edge-Bi-Fano detectors. As recalled from Section 3.4.1, bias re-adjustments are needed only when some biased branch metrics $m_B[n]$ along the correct path are so large that the receiver is not able to get to the end of the tree and ending up at the root node with no path to move forward. For the forward Fano decoder, this incident often happens when a deep fade occurs at the beginning of the decoding block. For the Bi-Fano and the Edge-Bi-Fano detectors, search failures happen when deep fades appear at both ends of the decoding block. Subsequently, the number of search failures is closely related to both the probability of deep fades and their positions within the observation window. This assumption is justified in Figure 3-27 and Figure 3-28 that the number of search failures is almost independent of the window size *N*. The plots also confirm our explanation in Section 3.4.1 that the number of search failures increases with SNR. In spite of this, when we look at the plots in Figure 3-25 and Figure 3-26, the number of node transitions is still a stable function of SNR for the forward Fano and only shows a small growth for the Bi-Fano and Edge-Bi-Fano.

The probability of a differed symbol in the tentative decisions provided by the forward and the backward Fano detectors are plotted in Figure 3-29 for QDPSK and in Figure 3-30 for 8DPSK. The error curves are obtained by dividing the average number of differed symbols in the two tentative decisions $E[n_d]$ (see Section 3.4.3) by the number of data symbols per block, i.e. $E[n_d]/(N-1)$. Figure 3-29 and Figure 3-30 show that the Edge-Bi-Fano detector is making more accurate preliminary decisions than the Bi-Fano detector. This coincides with our assumption that the data symbols at the end of the block are more prone to errors. As an example, for a block size of $N = 10$ and an SNR of 40 dB, the Bi-Fano has a probability of 3.3×10^{-3} for QDPSK and 1.1×10^{-2} for 8DPSK whereas the Edge-Bi-Fano has a probability of 1.3×10^{-3} for QDPSK and 4.4×10^{-3} for 8DPSK. In other words, most of the time error corrections are not required at the final decision making process, especially when SNR is large. This suggest that further improvement can be made on the complexity of the Bi-Fano and Edge-Bi-Fano detectors if an intelligent way of performing bi-directional search on demand is found. We will present in Chapter 5 that this bi-directional search on demand can be performed by monitoring the instantaneous energy level of the received signals.

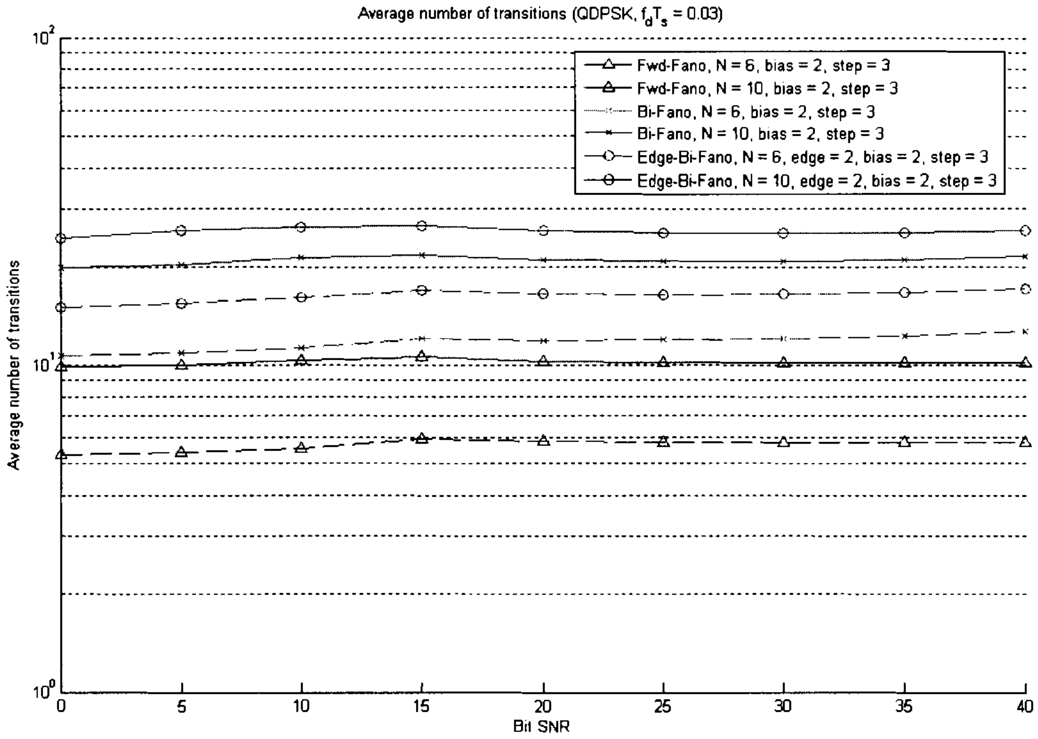

Figure 3-25. Average number of node transitions versus bit SNR for QDPSK at f_dT_s = 0.03.

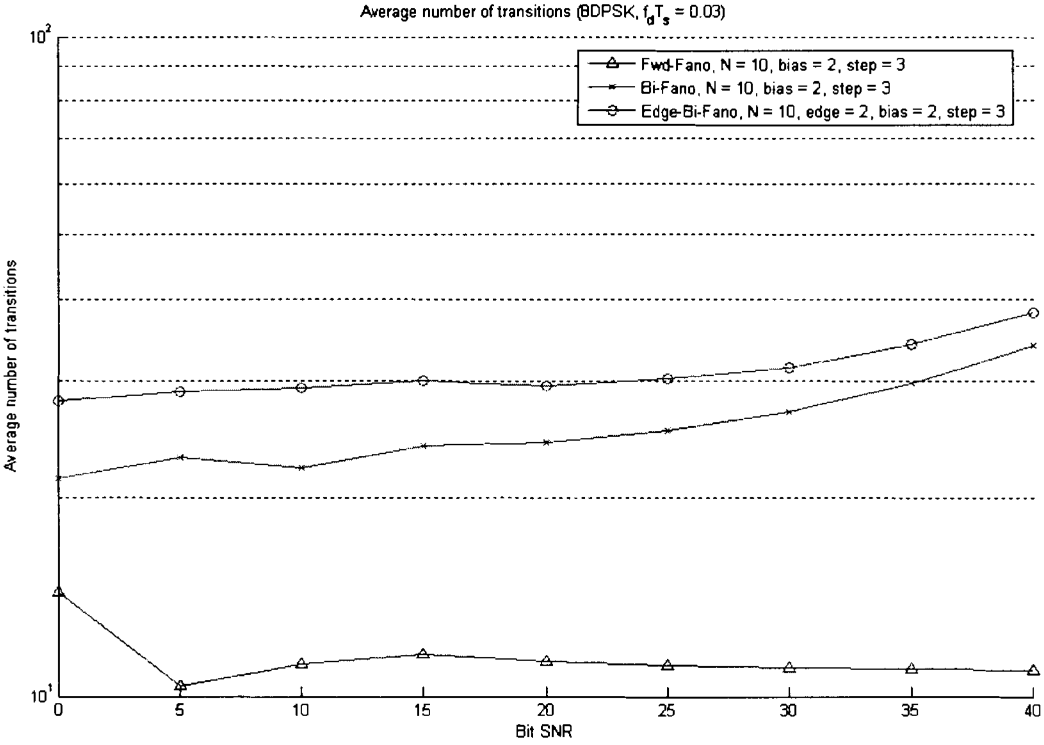

Figure 3-26. Average number of node transitions versus bit SNR for 8DPSK at f_dT_s = 0.03.

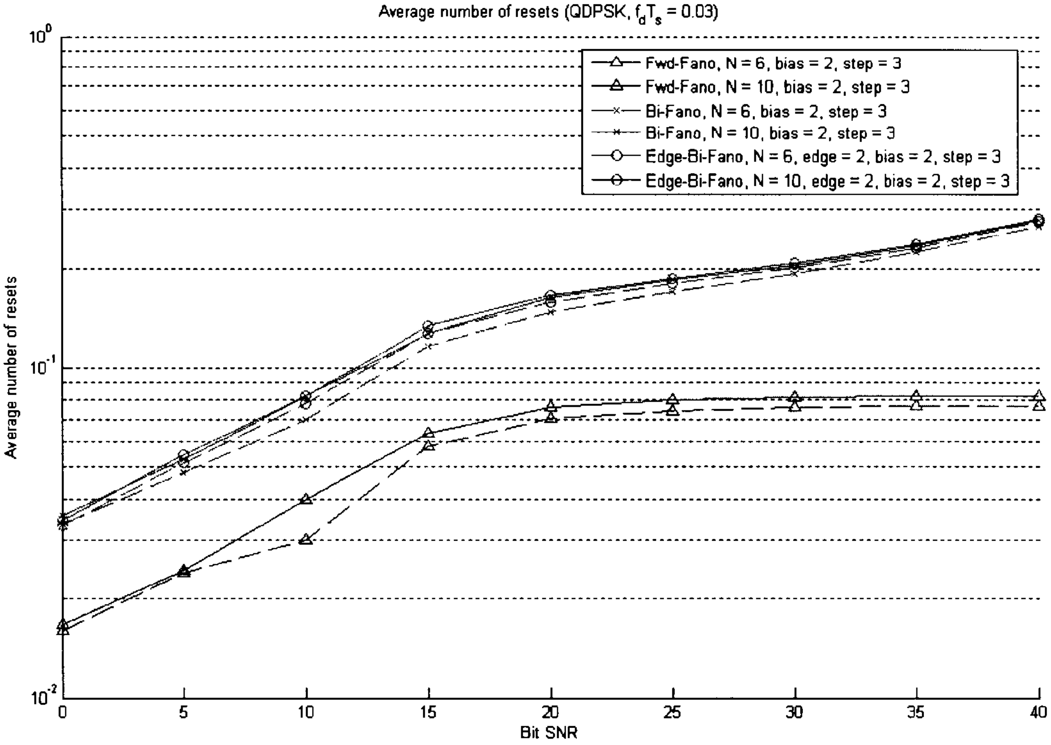

Figure 3-27. Average number of search failures versus bit SNR for QDPSK at f_dT_s = 0.03.

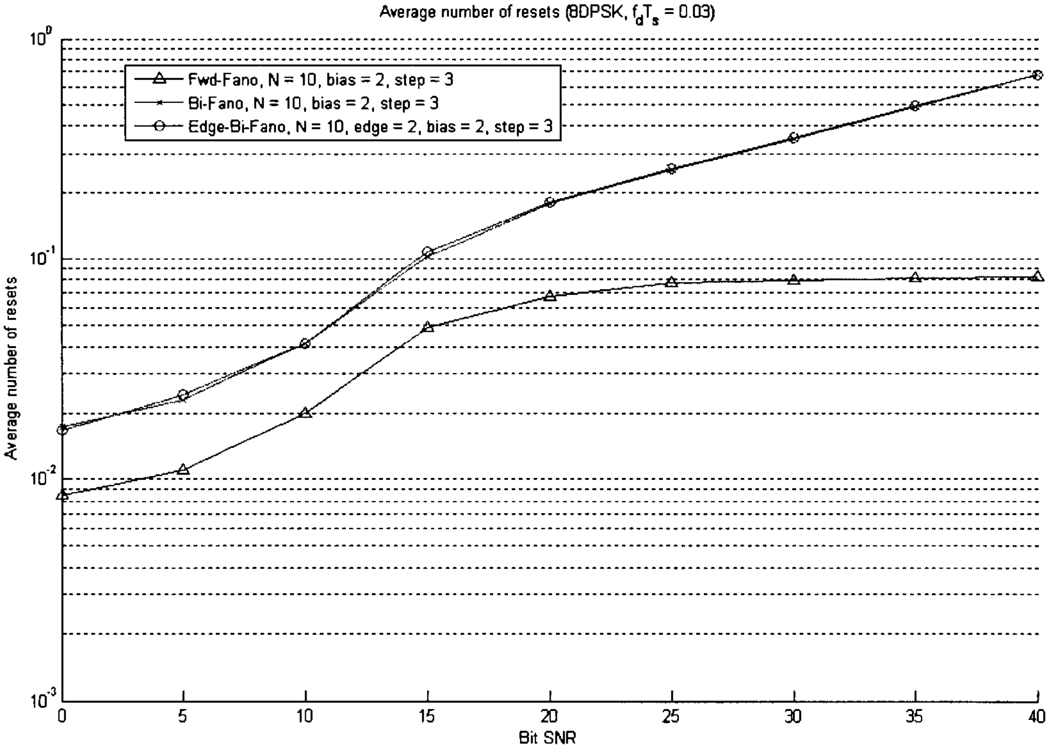

Figure 3-28. Average number of search failures versus bit SNR for 8DPSK at f_dT_s = 0.03.

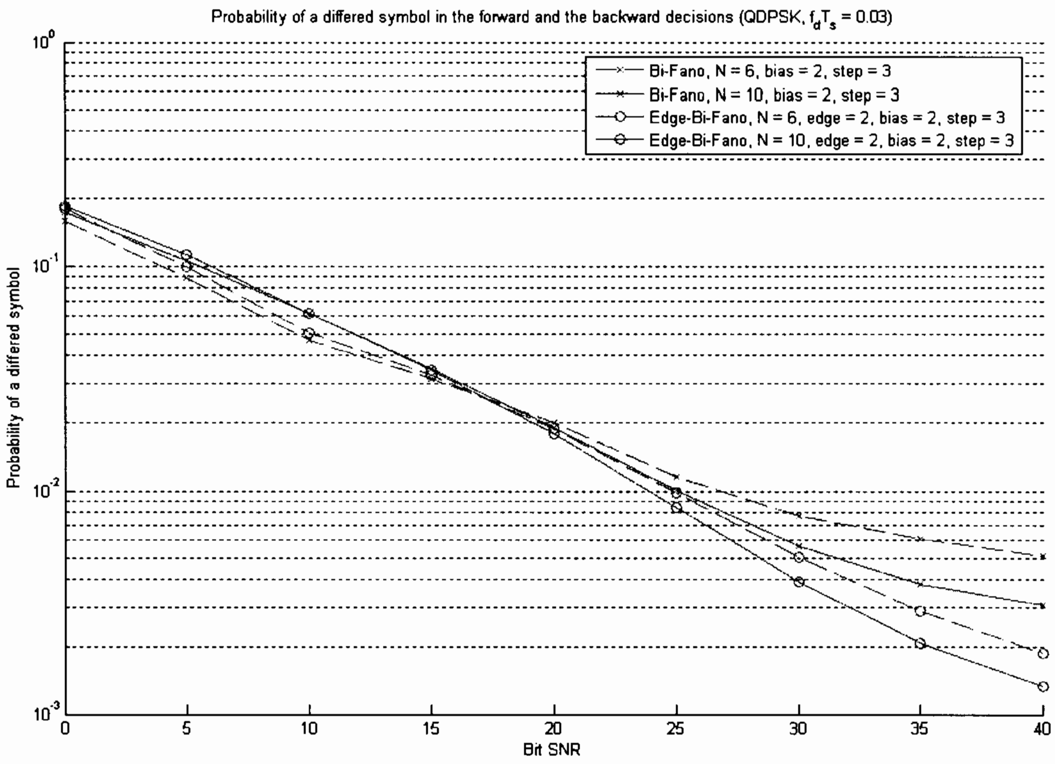

Figure 3-29. Probability of a different detected symbol versus bit SNR for QDPSK at f_dT_s = 0.03.

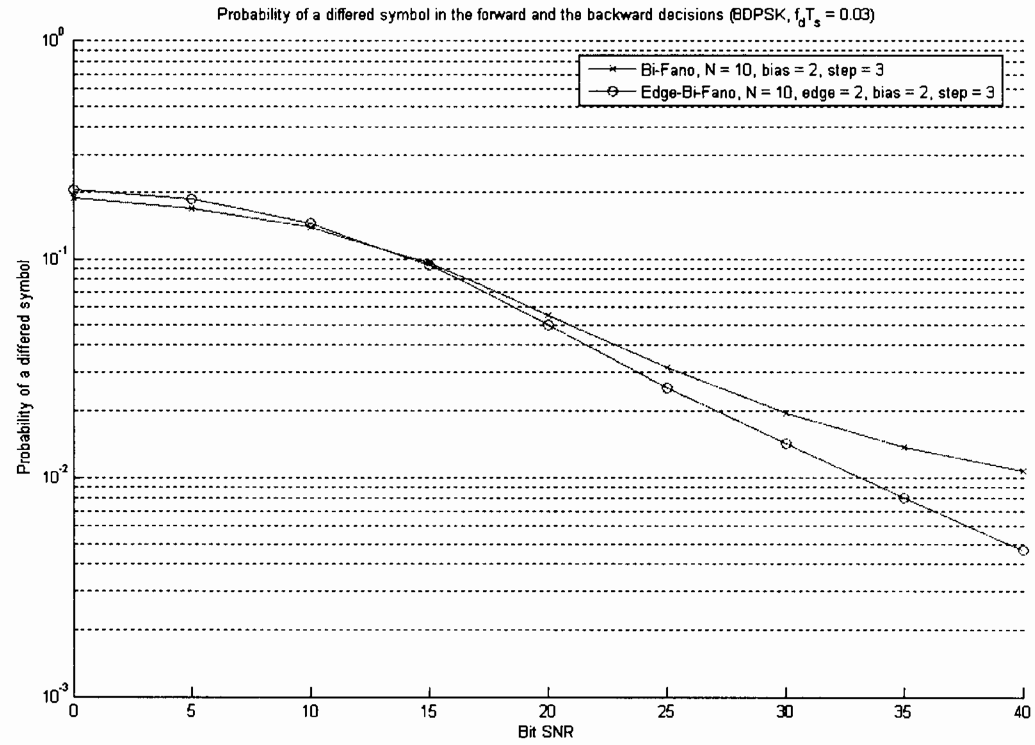

Figure 3-30. Probability of a different detected symbol versus bit SNR for 8DPSK at f_dT_s = 0.03.

CHAPTER 4: SPACE-TIME MULTIPLE-SYMBOL DIFFERENTIAL DETECTIONS

As observed in Chapter 3, the Fano decoder is capable of striking a good balance between bit-error performance and complexity, and robust enough to operate over a wide range of fading rates. It has been shown that, for QDPSK, the complexity of this novel decoding strategy reduces by one to two orders of magnitude when it is compared with the one of the sphere decoder at low SNR. The results become more remarkable when the number of outgoing branches per node is increased to eight (i.e. 8DPSK). In order to take the full advantage of the Fano algorithm, we consider in this chapter a higher dimensional system. We intuitively extend the application of the Fano algorithm to a ST-MSDD receiver for an Alamouti-type ST system with differential encoding. Similarly, simulations and discussions are presented at the end of the chapter. Two other reduced-complexity techniques, ST-DFDD and ST-MSDSD, are derived and used as performance benchmarks.

This chapter is organized as follows: Section 4.1 derives the ML sequence decoding metric. Section 4.2 presents a new two-dimensional branch sorting algorithm for ranking the candidate ST symbols in each decoding interval. This algorithm is used in both the Fano ST-MSDD and ST-MSDSD receivers to avoid the tedious process of calculating and sorting of the branch metrics at each node transition. Then, ST-DFDD and ST-MSDSD are derived in Sections 4.3 and 4.4 respectively. In Section 4.5, the ST versions of the three Fano-MSDD receivers, forward Fano, Bi-Fano, and Edge-Bi-Fano

ST-MSDDs, are presented. Finally, this chapter is concluded with simulation results and some follow-up discussions.

4.1 Derivation of the ML Decoding Metric

In an Alamouti-type ST system, the receive antenna receives two received samples at each ST-symbol interval T_{ST}. In order to maintain consistency throughout the thesis, a ST-MSDD receiver with block size of N is referred to as a block detector which processes $2N$ received samples simultaneously at a time. In other words, an N-symbol long ST-MSDD receiver makes a joint ML decision on the $2(N-1)$ information symbols $\hat{d}_i[k]$'s, $i=1,2,\ k=1..(N-1)$, contained in the received pattern

$$\mathbf{R}=\begin{bmatrix}\mathbf{r}[0]\\ \mathbf{r}[1]\\ \vdots\\ \mathbf{r}[N-1]\end{bmatrix}=\mathbf{SF}+\mathbf{N}, \tag{46}$$

where $\mathbf{S}=diag\{\mathbf{s}[0],\ \mathbf{s}[1],\ \dots,\ \mathbf{s}[N-1]\}$ is a block diagonal matrix representing the transmitted pattern, and $\mathbf{F}=\left[\mathbf{f}^T[0],\mathbf{f}^T[1],\dots,\mathbf{f}^T[N-1]\right]^T$ and $\mathbf{N}=\left[\mathbf{n}^T[0],\mathbf{n}^T[1],\dots,\mathbf{n}^T[N-1]\right]^T$ are the corresponding fading and noise patterns. Conditioned on $\mathbf{S}$, the received pattern is a zero-mean complex Gaussian vector with a covariance matrix

$$\mathbf{\Psi}_{\mathrm{RR}}=\tfrac{1}{2}E\left[\mathbf{RR}^H\right]=\mathbf{S\Psi}_{\mathrm{FF}}\mathbf{S}^H+\mathbf{\Psi}_{\mathrm{NN}}, \tag{47}$$

where $\mathbf{\Psi}_{\mathrm{FF}}=\mathbf{\Phi}_{ff}\otimes\mathbf{I}_2$, $\mathbf{\Psi}_{\mathrm{NN}}=\mathbf{\Phi}_{nn}\otimes\mathbf{I}_2$, and $\mathbf{\Phi}_{ff}$ and $\mathbf{\Phi}_{nn}$ are the SISO channel's fading and noise covariance matrices defined in Chapter 3. Specifically, $\mathbf{\Phi}_{nn}=\mathbf{I}_N$, and the (n,m)-th element of $\mathbf{\Phi}_{ff}$, n, $m = 0,1, \ldots, N-1$, equals $\phi_f[n-m] = \phi_f[v] = \sigma_f^2 J_v$ as shown in (7). Based on these results, we can show that $\mathbf{\Psi}_{\mathrm{RR}}$ can be rewritten as

$$\mathbf{\Psi}_{\mathrm{RR}} = \mathbf{S}\left(\mathbf{\Phi}_{gg} \otimes \mathbf{I}_2\right)\mathbf{S}^H, \tag{48}$$

where $\mathbf{\Phi}_{gg} = \mathbf{\Phi}_{ff} + \mathbf{\Phi}_{nn}$ is equivalent to the SISO channel's fading-plus-noise covariance matrix in (24). Furthermore, we can deduce from (48) that the conditional p.d.f. of the received pattern $\mathbf{R}$ is

$$p(\mathbf{R}\,|\,\mathbf{S}) = \frac{1}{(2\pi)^{2N}\left|\mathbf{\Psi}_{RR}\right|}\exp\left(-\mathbf{R}^H\mathbf{\Psi}_{RR}^{-1}\mathbf{R}\right). \tag{49}$$

Since $\left|\mathbf{\Psi}_{RR}\right|$ is independent of $\mathbf{S}$, the ML ST sequence detector can make a joint decision on the $N-1$ ST information symbols $\mathbf{c}[1], \mathbf{c}[2], \ldots, \mathbf{c}[N-1]$ in (11) by minimizing the metric

$$M\left(\hat{\mathbf{G}}\right) = \mathbf{R}^H\hat{\mathbf{S}}\left(\mathbf{\Phi}_{gg} \otimes \mathbf{I}_2\right)^{-1}\hat{\mathbf{S}}^H\mathbf{R} = \hat{\mathbf{G}}^H\left(\mathbf{\Phi}_{gg} \otimes \mathbf{I}_2\right)^{-1}\hat{\mathbf{G}}, \tag{50}$$

where

$$\hat{\mathbf{G}} = \hat{\mathbf{S}}^H\mathbf{R} = \begin{bmatrix} \hat{\mathbf{g}}[0] \\ \hat{\mathbf{g}}[1] \\ \vdots \\ \hat{\mathbf{g}}[N-1] \end{bmatrix}, \tag{51}$$

$\hat{\mathbf{g}}[k] = \left[\hat{g}_1[k], \hat{g}_2[k]\right]^T = \hat{\mathbf{s}}^H[k]\mathbf{r}[k]$, is the sequence of hypothesized fading-plus-noise samples associated with the hypothesized transmitted pattern $\hat{\mathbf{S}} = diag\left\{\hat{\mathbf{s}}[0], \hat{\mathbf{s}}[1], \ldots, \hat{\mathbf{s}}[N-1]\right\}$.

Although the metric in (50) is well defined, its block-oriented format is not compatible with the ML sequential detections. However, by applying the similar iterative procedure described in Section 3.1 on the matrix $\mathbf{\Phi}_{gg} \otimes \mathbf{I}_2$, we can manipulate (50) into a sum of branch metrics. To begin, let

$$\mathbf{G}[n] = \begin{bmatrix} \mathbf{g}[0] \\ \vdots \\ \mathbf{g}[n] \end{bmatrix} = \begin{bmatrix} \mathbf{s}^H[0]\mathbf{r}[0] \\ \vdots \\ \mathbf{s}^H[n]\mathbf{r}[n] \end{bmatrix} \tag{52}$$

be the equivalence of the fading-plus-noise process. Its covariance matrix is

$$\mathbf{\Psi}_{GG}[n] = \frac{1}{2}E\left[\mathbf{G}[n]\mathbf{G}^H[n]\right] = \mathbf{\Phi}_{gg}[n] \otimes \mathbf{I}_2 \tag{53}$$

where $\mathbf{\Phi}_{gg}[n]$ is the $(n+1)$ by $(n+1)$ covariance matrix defined in (29). Similarly, let

$$\hat{\mathbf{G}}[n] = \begin{bmatrix} \hat{\mathbf{g}}[0] \\ \vdots \\ \hat{\mathbf{g}}[n] \end{bmatrix} \tag{54}$$

be a column vector containing $2(n+1)$ hypothesized fading-plus-noise samples up to ST interval *n*, where *n* = 1, 2, ... (*N* – 1). Now the quadratic form $\hat{\mathbf{G}}^H[n]\mathbf{\Psi}_{GG}^{-1}[n]\hat{\mathbf{G}}[n]$ can be written as

$$\begin{aligned} &\hat{\mathbf{G}}^H[n]\mathbf{\Psi}_{GG}^{-1}[n]\hat{\mathbf{G}}[n] \\ &\quad = \left[\hat{\mathbf{G}}^H[n-1] \quad \hat{\mathbf{g}}^H[n]\right] \begin{bmatrix} \mathbf{\Psi}_{GG}[n-1] & \boldsymbol{\xi}_G[n-1] \\ \boldsymbol{\xi}_G^H[n-1] & \phi[0]\mathbf{I}_2 \end{bmatrix}^{-1} \begin{bmatrix} \hat{\mathbf{G}}[n-1] \\ \hat{\mathbf{g}}[n] \end{bmatrix} \\ &\quad = \left[\hat{\mathbf{G}}^H[n-1] \quad \hat{\mathbf{g}}^H[n]\right] \begin{bmatrix} \mathbf{\Phi}_{gg}[n-1] \otimes \mathbf{I}_2 & \boldsymbol{\varphi}_g[n-1] \otimes \mathbf{I}_2 \\ \boldsymbol{\varphi}_g^H[n-1] \otimes \mathbf{I}_2 & \phi[0] \otimes \mathbf{I}_2 \end{bmatrix}^{-1} \begin{bmatrix} \hat{\mathbf{G}}[n-1] \\ \hat{\mathbf{g}}[n] \end{bmatrix}, \end{aligned} \tag{55}$$

where

$$\xi_G[n-1]=\frac{1}{2}E\left[\mathbf{G}[n-1]\mathbf{g}^H[n]\right]=\boldsymbol{\varphi}_g[n-1]\otimes\mathbf{I}_2, \tag{56}$$

and $\phi[0]=\sigma_f^2+1$. $\boldsymbol{\Phi}_{gg}[n-1]$ and $\boldsymbol{\varphi}_g[n-1]$ were defined in (29) and (31) respectively. By taking the inverse of the covariance matrix in (53) using the iterative technique described in [25] and [26], (55) can be arrived to

$$\begin{aligned}
&\hat{\mathbf{G}}^H[n]\boldsymbol{\Psi}_{GG}^{-1}[n]\hat{\mathbf{G}}[n]\\
&\quad=\hat{\mathbf{G}}^H[n-1]\boldsymbol{\Psi}_{GG}^{-1}[n-1]\hat{\mathbf{G}}[n-1]+\hat{\mathbf{g}}^H[n]\mathbf{K}^{-1}[n]\hat{\mathbf{g}}[n]\\
&\quad-\hat{\mathbf{G}}^H[n-1]\boldsymbol{\Psi}_{GG}^{-1}[n-1]\left(\boldsymbol{\varphi}_g[n-1]\otimes\mathbf{I}_2\right)\mathbf{K}^{-1}[n]\hat{\mathbf{g}}[n]\\
&\quad-\hat{\mathbf{g}}^H[n]\mathbf{K}^{-1}[n]\left(\boldsymbol{\varphi}_g^H[n-1]\otimes\mathbf{I}_2\right)\boldsymbol{\Psi}_{GG}^{-1}[n-1]\hat{\mathbf{G}}[n-1]\\
&\quad+\hat{\mathbf{G}}^H[n-1]\boldsymbol{\Psi}_{GG}^{-1}[n-1]\left(\boldsymbol{\varphi}_g[n-1]\otimes\mathbf{I}_2\right)\mathbf{K}^{-1}[n]\left(\boldsymbol{\varphi}_g^H[n-1]\otimes\mathbf{I}_2\right)\boldsymbol{\Psi}_{GG}^{-1}[n-1]\hat{\mathbf{G}}[n-1],
\end{aligned} \tag{57}$$

where

$$\begin{aligned}
\mathbf{K}[n]&=\phi[0]\mathbf{I}_2-\left(\boldsymbol{\varphi}_g^H[n-1]\otimes\mathbf{I}_2\right)\left(\boldsymbol{\Phi}_{gg}[n-1]\otimes\mathbf{I}_2\right)^{-1}\left(\boldsymbol{\varphi}_g[n-1]\otimes\mathbf{I}_2\right)\\
&=\phi[0]\mathbf{I}_2-\left(\boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\boldsymbol{\varphi}_g[n-1]\right)\otimes\mathbf{I}_2\\
&=\varepsilon^2[n]\mathbf{I}_2,
\end{aligned} \tag{58}$$

with $\varepsilon^2[n]$, defined earlier in (33), being the mean-square error of an n-th order linear predictor of the fading process. Substituting (58) into (57) leads to

$$\begin{aligned}
&\hat{\mathbf{G}}^{H}[n]\boldsymbol{\Psi}_{GG}^{-1}[n]\hat{\mathbf{G}}[n] \\
&= \hat{\mathbf{G}}^{H}[n-1]\boldsymbol{\Psi}_{GG}^{-1}[n-1]\hat{\mathbf{G}}[n-1] + \frac{1}{\varepsilon^{2}[n]}\hat{\mathbf{g}}^{H}[n]\hat{\mathbf{g}}[n] \\
&- \frac{1}{\varepsilon^{2}[n]}\hat{\mathbf{G}}^{H}[n-1]\left\{\left(\boldsymbol{\Phi}_{gg}^{-1}[n-1]\boldsymbol{\varphi}_{g}[n-1]\right)\otimes\mathbf{I}_{2}\right\}\hat{\mathbf{g}}[n] \\
&- \frac{1}{\varepsilon^{2}[n]}\hat{\mathbf{g}}^{H}[n]\left\{\left(\boldsymbol{\varphi}_{g}^{H}[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\right)\otimes\mathbf{I}_{2}\right\}\hat{\mathbf{G}}[n-1] \\
&+ \frac{1}{\varepsilon^{2}[n]}\hat{\mathbf{G}}^{H}[n-1]\left\{\left(\boldsymbol{\Phi}_{gg}^{-1}[n-1]\boldsymbol{\varphi}_{g}[n-1]\right)\otimes\mathbf{I}_{2}\right\}\left\{\left(\boldsymbol{\varphi}_{g}^{H}[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\right)\otimes\mathbf{I}_{2}\right\}\hat{\mathbf{G}}[n-1].
\end{aligned} \tag{59}$$

The second term in the above equation is simply

$$\frac{1}{\varepsilon^{2}[n]}\hat{\mathbf{g}}^{H}[n]\hat{\mathbf{g}}[n] = \frac{1}{\varepsilon^{2}[n]}\left(|\hat{g}_{1}[n]|^{2} + |\hat{g}_{2}[n]|^{2}\right). \tag{60}$$

Furthermore, since

$$\left\{\left(\boldsymbol{\varphi}_{g}^{H}[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\right)\otimes\mathbf{I}_{2}\right\}\hat{\mathbf{G}}[n-1] = \begin{bmatrix} \boldsymbol{\varphi}_{g}^{H}[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\hat{\mathbf{G}}_{1}[n-1] \\ \boldsymbol{\varphi}_{g}^{H}[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\hat{\mathbf{G}}_{2}[n-1] \end{bmatrix}, \tag{61}$$

where

$$\hat{\mathbf{G}}_{1}[n-1] = \begin{bmatrix} \hat{g}_{1}[0] \\ \hat{g}_{1}[1] \\ \vdots \\ \hat{g}_{1}[n-1] \end{bmatrix} \tag{62}$$

and

$$\hat{\mathbf{G}}_{2}[n-1] = \begin{bmatrix} \hat{g}_{2}[0] \\ \hat{g}_{2}[1] \\ \vdots \\ \hat{g}_{2}[n-1] \end{bmatrix}, \tag{63}$$

the last two terms in (59) can be rewritten as

$$\begin{aligned}&\frac{1}{\varepsilon^2[n]}\hat{\mathbf{g}}^H[n]\left\{\left(\boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\right)\otimes\mathbf{I}_2\right\}\hat{\mathbf{G}}[n-1]\\&=\frac{1}{\varepsilon^2[n]}\left\{\hat{g}_1^*[n]\left(\boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\hat{\mathbf{G}}_1[n-1]\right)+\hat{g}_2^*[n]\left(\boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\hat{\mathbf{G}}_2[n-1]\right)\right\}\end{aligned}\tag{64}$$

and

$$\begin{aligned}&\frac{1}{\varepsilon^2[n]}\hat{\mathbf{G}}^H[n-1]\left\{\left(\boldsymbol{\Phi}_{gg}^{-1}[n-1]\boldsymbol{\varphi}_g[n-1]\right)\otimes\mathbf{I}_2\right\}\left\{\left(\boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\right)\otimes\mathbf{I}_2\right\}\hat{\mathbf{G}}[n-1]\\&=\frac{1}{\varepsilon^2[n]}\hat{\mathbf{G}}_1^H[n-1]\left(\boldsymbol{\Phi}_{gg}^{-1}[n-1]\boldsymbol{\varphi}_g[n-1]\boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\right)\hat{\mathbf{G}}_1[n-1]\\&+\frac{1}{\varepsilon^2[n]}\hat{\mathbf{G}}_2^H[n-1]\left(\boldsymbol{\Phi}_{gg}^{-1}[n-1]\boldsymbol{\varphi}_g[n-1]\boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\right)\hat{\mathbf{G}}_2[n-1].\end{aligned}\tag{65}$$

Finally, substituting (60)-(65) into (59) enables us to arrive at the recursive equation

$$\begin{aligned}\hat{\mathbf{G}}^H[n]\boldsymbol{\Psi}_{GG}^{-1}[n]\hat{\mathbf{G}}[n]&=\hat{\mathbf{G}}^H[n-1]\boldsymbol{\Psi}_{GG}^{-1}[n-1]\hat{\mathbf{G}}[n-1]\\&+\frac{1}{\varepsilon^2[n]}\sum_{i=1}^{2}\left|\hat{g}_i[n]-\left(\boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\right)\hat{\mathbf{G}}_i[n-1]\right|^2.\end{aligned}\tag{66}$$

In comparing (66) with its counterpart in the SISO channel in Chapter 3, we notice an additional square distance term in the branch metric. This coincides with the fact that there are now two links between the transmitter and the receiver. The total APM can be written as

$$\begin{aligned}&\hat{\mathbf{G}}^H[N-1]\boldsymbol{\Psi}_{GG}^{-1}[N-1]\hat{\mathbf{G}}[N-1]\\&=m[0]+\sum_{n=1}^{N-1}\frac{1}{\varepsilon^2[n]}\sum_{i=1}^{2}\left|\hat{g}_i[n]-\left(\boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\right)\hat{\mathbf{G}}_i[n-1]\right|^2,\end{aligned}\tag{67}$$

where

$$m[0]=\frac{1}{\varepsilon^2[0]}\sum_{i=1}^{2}\left|\hat{g}_i[0]\right|^2\tag{68}$$

is the branch metric of the reference fading-plus-noise samples. It should be pointed out that $m[0]$ is common to all hypotheses and hence it will be omitted from the path metric in the subsequent formulations.

Now the metric in (50) can be manipulated into

$$M\left(\hat{\mathbf{G}}\right) = \sum\nolimits_{n=1}^{N-1} m[n], \tag{69}$$

where

$$m[n] = \frac{1}{\varepsilon^2[n]} \sum_{i=1}^{2} \left| \hat{g}_i[n] - \left(\boldsymbol{\varphi}_g^H[n-1] \boldsymbol{\Phi}_{gg}^{-1}[n-1] \right) \hat{\mathbf{G}}_i[n-1] \right|^2 \tag{70}$$

is the n-th branch metric. As recalled, $\hat{\mathbf{G}}_i[n-1] = \left[\hat{g}_i[0] \quad \hat{g}_i[1] \quad \cdots \quad \hat{g}_i[n-1] \right]^T$ is the sequence of past fading-plus-noise estimates in the i-th link, and $\boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]$ is an n-th order fading-plus-noise predictor, $\varepsilon^2[n] = \phi[0] - \boldsymbol{\varphi}_g^H[n-1]\boldsymbol{\Phi}_{gg}^{-1}[n-1]\boldsymbol{\varphi}_g[n-1]$ is the corresponding mean-square prediction error, $\boldsymbol{\varphi}_g[n-1] = \sigma_f^2 \cdot \left[J_{-n}, J_{1-n}, \ldots, J_{-1} \right]^T$, defined in (31),is the correlation vector between the current fading-plus-noise sample $g_i[n] = f_i[n] + n_i[n]$ and the past samples $\mathbf{g}_i[n-1] = \left[g_i[0] \quad g_i[1] \quad \cdots \quad g_i[n-1] \right]^T$, and $\boldsymbol{\Phi}_{gg}[n-1]$, the covariance matrix of $\mathbf{g}_i[n-1]$, is a matrix obtained from the first n rows and n columns of $\boldsymbol{\Phi}_{gg}$ in (24).

As discussed earlier in Chapter 3, in order for the Fano decoder to move forward from a node that it visits the first time, it must determine which is the most promising branch emanating from that node. If that turns out to be futile, it will examine the next best node, and so on. This means the decoder needs to sort all of the outgoing branches of the current node from best to worst. The same sorting is required for the sphere decoder. In the SISO system, this sorting can be easily accomplished by zigzagging through the hypothesized phase angles (refer to Section 3.4.1). However, in the ST system, the branch metric $m[n]$ now is composed of two square distance terms and, in a brute-force implementation, the sorting of all branch metrics requires M^2 branch metric calculations and comparisons. If unfortunately the decoder has to backtrack from a node, then the next most promising branch from the ancestor node must also be known. Thus in the worst case, the decoder needs to go through the tedious process of calculating and sorting the branch metrics for every node that it has visited. While this scenario is unlikely to happen, it points out the importance of calculating the branch metrics and sorting them efficiently. We will provide in the next section a sorting routine that can achieve this objective.

4.2 Branch Sorting Algorithm for ST-MSDD

Consider the branch metric in (70). It can be rewritten in the form

$$\begin{aligned} m[n] &= \frac{1}{\varepsilon^2[n]} \sum_{i=1}^{2} \left| \hat{g}_i[n] - \left(\boldsymbol{\varphi}_g^H[n-1] \boldsymbol{\Phi}_{gg}^{-1}[n-1] \right) \hat{\mathbf{G}}_i[n-1] \right|^2 \\ &= \frac{1}{\varepsilon^2[n]} \sum_{i=1}^{2} \left| \hat{g}_i[n] - \tilde{g}_i[n] \right|^2 \\ &= \frac{1}{\varepsilon^2[n]} \left\{ \left| \hat{g}_1[n] \right|^2 + \left| \hat{g}_2[n] \right|^2 + \left| \tilde{g}_1[n] \right|^2 + \left| \tilde{g}_2[n] \right|^2 - 2\,\mathrm{Re}\left\{ \tilde{g}_1[n] \hat{g}_1^*[n] \right\} - 2\,\mathrm{Re}\left\{ \tilde{g}_2[n] \hat{g}_2^*[n] \right\} \right\}, \end{aligned} \tag{71}$$

where

$$\tilde{g}_i[n] = \left(\boldsymbol{\varphi}_g^H[n-1] \boldsymbol{\Phi}_{gg}^{-1}[n-1] \right) \hat{\mathbf{G}}_i[n-1], \qquad i = 1,2, \tag{72}$$

are the predicted fading-plus-noise samples. Given that the transmitted ST symbol $\mathbf{s}[n]$ is unitary and $\hat{\mathbf{g}}[n] = \hat{\mathbf{s}}^H[n]\mathbf{r}[n]$, we have $\hat{\mathbf{g}}^H[n]\hat{\mathbf{g}}[n] = \mathbf{r}^H[n]\hat{\mathbf{s}}[n]\hat{\mathbf{s}}^H[n]\mathbf{r}[n] = \mathbf{r}^H[n]\mathbf{r}[n]$, or equivalent

$$\left|\hat{g}_1[n]\right|^2 + \left|\hat{g}_2[n]\right|^2 = \left|r_1[n]\right|^2 + \left|r_2[n]\right|^2, \tag{73}$$

a result that is independent of the hypothesis at the n-th ST interval. Similarly, the sum of the predicted fading-plus-noise samples, $\left|\tilde{g}_1[n]\right|^2 + \left|\tilde{g}_2[n]\right|^2$, is also independent of the branch hypothesis. These imply that during the decoding process, the branch metrics can be sorted based on the last two terms in (71). Let us define the sum of these two terms as

$$\begin{aligned} q[n] &= 2\,\mathrm{Re}\left\{\tilde{g}_1[n]\hat{g}_1^*[n]\right\} + 2\,\mathrm{Re}\left\{\tilde{g}_2[n]\hat{g}_2^*[n]\right\} \\ &= 2\,\mathrm{Re}\left\{\tilde{g}_1^*[n]\hat{g}_1[n]\right\} + 2\,\mathrm{Re}\left\{\tilde{g}_2^*[n]\hat{g}_2[n]\right\} \\ &= 2\,\mathrm{Re}\left\{\tilde{\mathbf{g}}^H[n]\hat{\mathbf{g}}[n]\right\}, \end{aligned} \tag{74}$$

where $\tilde{\mathbf{g}}[n] = \left[\tilde{g}_1[n] \quad \tilde{g}_2[n]\right]^T$ is the predicted fading-plus-noise vector at the n-th ST interval and it depends solely on the path history in the decoding tree. Since $\hat{\mathbf{s}}[n] = \hat{\mathbf{c}}[n]\hat{\mathbf{s}}[n-1]$ and $\hat{\mathbf{g}}[n] = \hat{\mathbf{s}}^H[n]\mathbf{r}[n] = \hat{\mathbf{s}}^H[n-1]\hat{\mathbf{c}}^H[n]\mathbf{r}[n]$, (74) can be rewritten as

$$\begin{aligned} q[n] &= 2\,\mathrm{Re}\left\{\left(\tilde{\mathbf{g}}^H[n]\hat{\mathbf{s}}^H[n-1]\right)\hat{\mathbf{c}}^H[n]\mathbf{r}[n]\right\} \\ &= \mathrm{Re}\left\{\mathbf{a}^H[n]\hat{\mathbf{d}}^H[n]\mathbf{r}[n]\right\}, \end{aligned} \tag{75}$$

where

$$\mathbf{a}[n] = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \hat{\mathbf{s}}[n-1]\tilde{\mathbf{g}}[n] = \begin{bmatrix} a_1[n] \\ a_2[n] \end{bmatrix}. \tag{76}$$

Similar to the predicted fading-plus-noise vector $\tilde{\mathbf{g}}[n]$, the vector $\mathbf{a}[n]$ also depends solely on the path history. Furthermore, (75) can be further simplified to

$$\begin{aligned} q[n] &= \mathrm{Re}\left\{ \begin{bmatrix} a_1^*[n] & a_2^*[n] \end{bmatrix} \begin{bmatrix} \hat{d}_1^*[n] & -\hat{d}_2[n] \\ \hat{d}_2^*[n] & \hat{d}_1[n] \end{bmatrix} \begin{bmatrix} r_1[n] \\ r_2[n] \end{bmatrix} \right\} \\ &= \mathrm{Re}\left\{ r_1[n]a_1^*[n]\hat{d}_1^*[n] + r_1[n]a_2^*[n]\hat{d}_2^*[n] - r_2[n]a_1^*[n]\hat{d}_2[n] + r_2[n]a_2^*[n]\hat{d}_1[n] \right\} \\ &= \mathrm{Re}\left\{ r_1^*[n]a_1[n]\hat{d}_1[n] + r_1^*[n]a_2[n]\hat{d}_2[n] - r_2[n]a_1^*[n]\hat{d}_2[n] + r_2[n]a_2^*[n]\hat{d}_1[n] \right\} \\ &= \mathrm{Re}\left\{ \left(r_1^*[n]a_1[n] + r_2[n]a_2^*[n] \right)\hat{d}_1[n] + \left(r_1^*[n]a_2[n] - r_2[n]a_1^*[n] \right)\hat{d}_2[n] \right\} \\ &= \mathrm{Re}\left\{ y_1^*[n]\hat{d}_1[n] \right\} + \mathrm{Re}\left\{ y_2^*[n]\hat{d}_2[n] \right\} \\ &= q_1[n] + q_2[n], \end{aligned} \tag{77}$$

where

$$\begin{aligned} q_1[n] &= \mathrm{Re}\left\{ y_1^*[n]\hat{d}_1[n] \right\} \\ q_2[n] &= \mathrm{Re}\left\{ y_2^*[n]\hat{d}_2[n] \right\}. \end{aligned} \tag{78}$$

$\hat{d}_i[n]$, $i = 1, 2$, are hypotheses of the data symbols in (11), and

$$\begin{aligned} y_1^*[n] &= r_1^*[n]a_1[n] + r_2[n]a_2^*[n] \\ y_2^*[n] &= r_1^*[n]a_2[n] - r_2[n]a_1^*[n] \end{aligned} \tag{79}$$

are the fading-compensated samples in the individual links. The important points conveyed by these equations are 1) all parameters, except $\hat{d}_1[n]$ and $\hat{d}_2[n]$, depend on the path history, and 2) the most promising branch is the one with the largest $q[n] = q_1[n] + q_2[n]$. These properties suggest that the M^2 values of $q[n]$ can be efficiently sorted by first creating the sorted lists

$$Q_i[n] = \left\{ q_{i,(1)}[n], q_{i,(2)}[n], \ldots, q_{i,(M)}[n] \right\}, \quad i = 1, 2, \tag{80}$$

for the individual $q_i[n]$'s, where $q_{i,(1)}[n]$ is the largest and $q_{i,(M)}[n]$ is the smallest. As described in Section 3.4.1, the sequence $Q_i[n]$ and the corresponding data estimates $\left\{\hat{d}_{i,(m)}[n]\right\}_{m=1}^{M}$ can be generated by first calculating the phase angle $\theta_i[n] = \arg\{y_i[n]\}$ and then sorting the points in the MPSK constellation based on their distance from $\theta_i[n]$. The best data estimate $\hat{d}_{i,(1)}[n]$ can be computed by

$$\hat{d}_{i,(1)}[n] = \exp\left(j\frac{2\pi}{M} \times round\left(\frac{\theta_i[n] \times M}{2\pi} \right) \right), \tag{81}$$

and the rest of the data estimates can be computed by zigzagging through the rest of the phase angles based on the quotient and remainder of the rounding operation described in Section 3.4.1.

Next, we form the $M \times M$ matrix

$$\mathbf{M}_S[n] = \begin{bmatrix} S_{11}[n] & S_{12}[n] & \cdots & S_{1M}[n] \\ S_{21}[n] & S_{22}[n] & \cdots & S_{2M}[n] \\ \vdots & \vdots & & \vdots \\ S_{M1}[n] & S_{M2}[n] & \cdots & S_{MM}[n] \end{bmatrix}, \tag{82}$$

where the (i, j)-th element of this matrix is $S_{ij}[n] = q_{1,(i)}[n] + q_{2,(j)}[n]$. This matrix has the following properties 1) the $S_{ij}[n]$'s are decreasing from left to right and from top to bottom, 2) any sub-matrix of $\mathbf{M}_S[n]$ also possesses the previous property, and 3) the matrix is symmetrical in the sense that if one of its elements takes on a value of *v*, then there must be another element with a value of -*v*. Based on these properties, we can see that $S_{11}[n]$ is, by default, the largest value of the sum $q[n]$. For the second largest value, we only need to choose between $S_{12}[n]$ and $S_{21}[n]$ as shown in Figure 4-1. Suppose $S_{12}[n]$ is the second largest, then the only candidates for the third largest sum are $S_{13}[n]$ and $S_{21}[n]$ (see Figure 4-2).

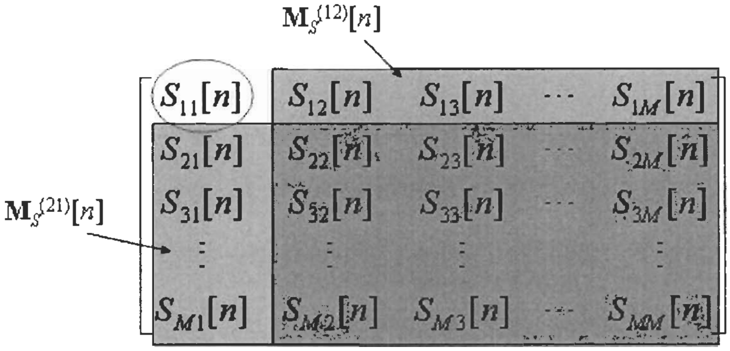
$\mathbf{M}_S^{(12)}[n]$
$\mathbf{M}_S^{(21)}[n]$
$S_{11}[n]$ $S_{12}[n]$ $S_{13}[n]$ $\cdots$ $S_{1M}[n]$
$S_{21}[n]$ $S_{22}[n]$ $S_{23}[n]$ $\cdots$ $S_{2M}[n]$
$S_{31}[n]$ $S_{32}[n]$ $S_{33}[n]$ $\cdots$ $S_{3M}[n]$
$S_{M1}[n]$ $S_{M2}[n]$ $S_{M3}[n]$ $\cdots$ $S_{MM}[n]$

Figure 4-1. Decision between $S_{12}[n]$ and $S_{21}[n]$.

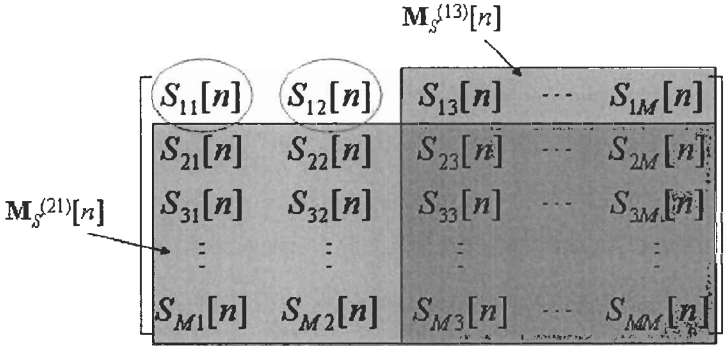
$\mathbf{M}_S^{(13)}[n]$
$\mathbf{M}_S^{(21)}[n]$
$S_{11}[n]$ $S_{12}[n]$ $S_{13}[n]$ $\cdots$ $S_{1M}[n]$
$S_{21}[n]$ $S_{22}[n]$ $S_{23}[n]$ $\cdots$ $S_{2M}[n]$
$S_{31}[n]$ $S_{32}[n]$ $S_{33}[n]$ $\cdots$ $S_{3M}[n]$
$S_{M1}[n]$ $S_{M2}[n]$ $S_{M3}[n]$ $\cdots$ $S_{MM}[n]$

Figure 4-2. Decision between $S_{13}[n]$ and $S_{21}[n]$.

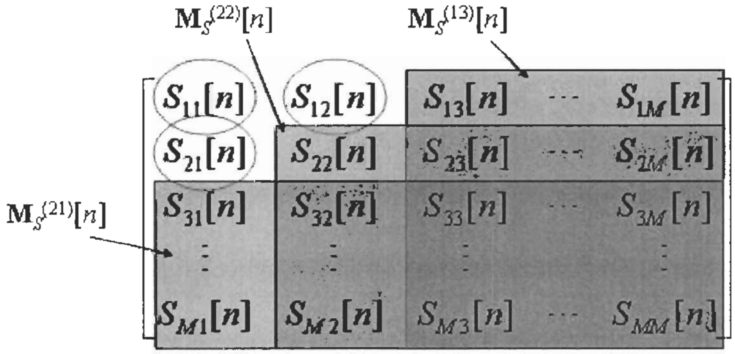
$\mathbf{M}_S^{(22)}[n]$
$\mathbf{M}_S^{(13)}[n]$
$\mathbf{M}_S^{(21)}[n]$
$S_{11}[n]$ $S_{12}[n]$ $S_{13}[n]$ $\cdots$ $S_{1M}[n]$
$S_{21}[n]$ $S_{22}[n]$ $S_{23}[n]$ $\cdots$ $S_{2M}[n]$
$S_{31}[n]$ $S_{32}[n]$ $S_{33}[n]$ $\cdots$ $S_{3M}[n]$
$S_{M1}[n]$ $S_{M2}[n]$ $S_{M3}[n]$ $\cdots$ $S_{MM}[n]$

Figure 4-3. Decision between $S_{13}[n]$, $S_{22}[n]$, and $S_{31}[n]$.

If $S_{21}[n]$ is greater than $S_{13}[n]$, then the candidates for the fourth largest sum are $S_{31}[n]$, $S_{22}[n]$, and $S_{13}[n]$. The procedure continues until the first $M^2/2$ largest sums are found. At this point, we make use of the symmetry in the matrix to sort the rest of the sums. We provide in Figure 4-4 the flow chart of this sorting algorithm. It should be emphasized that we only need to perform sorting on demand, because of the probabilistic nature of sequential decoding. In the worst case scenario, the number of comparisons/additions required is upper-bounded by

$$\begin{aligned}\Lambda &= 2^2 + 3^2 + \ldots + (M-1)^2 + \frac{M^2}{2} \\ &< \sum_{i=2}^{M} i^2 \\ &\leq \frac{M^3}{2}, \quad \text{for } M \geq 2.\end{aligned} \tag{83}$$

The statistical average, however, is much less than this number. It should also be emphasized that the application of this sorting algorithm is not restricted to the sphere and the Fano decoders. The same concept can be easily extended to other ordered tree searching algorithms and even to ST systems with higher dimensions.

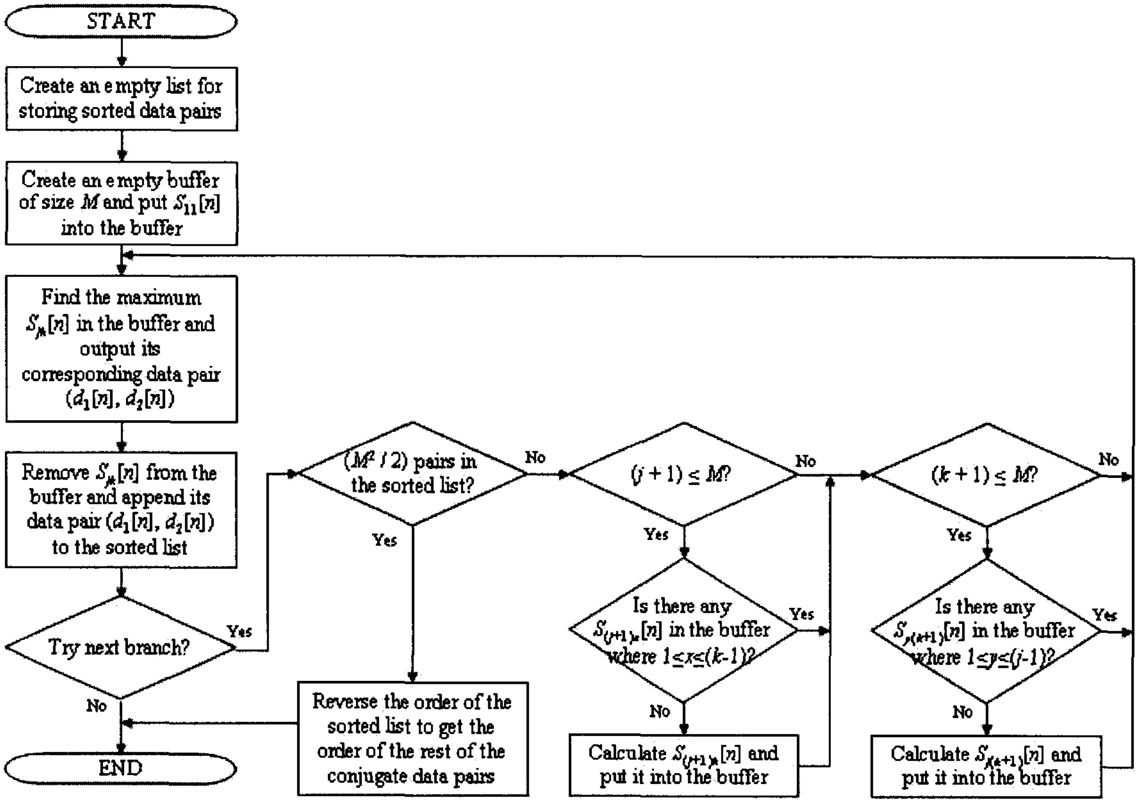
START
Create an empty list for storing sorted data pairs
Create an empty buffer of size M and put $S_{11}[n]$ into the buffer
Find the maximum $S_{jk}[n]$ in the buffer and output its corresponding data pair $(d_1[n], d_2[n])$
Remove $S_{jk}[n]$ from the buffer and append its data pair $(d_1[n], d_2[n])$ to the sorted list
Try next branch?
Yes
No
END
$(M^2/2)$ pairs in the sorted list?
Yes
No
Reverse the order of the sorted list to get the order of the rest of the conjugate data pairs
$(j+1) \leq M$?
Yes
No
Is there any $S_{(j+1)x}[n]$ in the buffer where $1 \leq x \leq (k-1)$?
Yes
No
Calculate $S_{(j+1)k}[n]$ and put it into the buffer
$(k+1) \leq M$?
Yes
No
Is there any $S_{y(k+1)}[n]$ in the buffer where $1 \leq y \leq (j-1)$?
Yes
No
Calculate $S_{j(k+1)}[n]$ and put it into the buffer

Figure 4-4. Flowchart of the sum sorting algorithm.

4.3 ST-DFDD

For the sake of completeness, a decision-feedback detector is derived in this chapter for the Alamouti-type ST system. This detector, termed ST-DFDD, follows the fundamental structure of the DFDD receiver used in the SISO channel. Inherently, the ST-DFDD receiver is a forward linear prediction receiver and it makes decision on one ST symbol at a time, based on the $N-1$ previously detected ST symbols. As mentioned in the preceding section, the sequence decoding metric now has two square distance terms because there are two links between the transmitter and the receiver. The decoding metric of a ST-DFDD receiver now becomes

$$M\left(\hat{g}[n]\right) = \frac{1}{\varepsilon_N^2}\sum_{i=1}^{2}\left|\hat{g}_i[n] - \mathbf{p}_N \hat{\mathbf{G}}_N^{(i)}[n]\right|^2 \tag{84}$$

where $\mathbf{p}_N = \boldsymbol{\varphi}_g^H[N-2]\boldsymbol{\Phi}_{gg}^{-1}[N-2]$ and $\varepsilon_N^2 = \phi[0] - \boldsymbol{\varphi}_g^H[N-2]\boldsymbol{\Phi}_{gg}^{-1}[N-2]\boldsymbol{\varphi}_g[N-2]$ are the same $(N-1)$-th order linear predictor and mean square prediction error as in Chapter 3, and $\hat{\mathbf{G}}_N^{(i)}[n] = \left[\hat{g}_i[n-N+1]\ \hat{g}_i[n-N+2]\ ...\ \hat{g}_i[n-1]\right]^T$, $i = 1,2,$ is the vector containing the past $(N-1)$ fading-plus-noise samples of the i-th link. Without any loss of generality, minimizing the metric in (84) is equivalent to maximizing the term

$$\begin{aligned} q_N[n] &= 2\mathrm{Re}\left\{\tilde{g}_N^{(1)}[n]\hat{g}_1^*[n]\right\} + 2\mathrm{Re}\left\{\tilde{g}_N^{(2)}[n]\hat{g}_2^*[n]\right\} \\ &= 2\mathrm{Re}\left\{\tilde{\mathbf{g}}_N^H[n]\hat{\mathbf{g}}[n]\right\}, \end{aligned} \tag{85}$$

where $\tilde{g}_N^{(i)}[n] = \mathbf{p}_N\hat{\mathbf{G}}_N^{(i)}[n]$, $i = 1,2,$ is the predicted fading-plus-noise sample of the i-th link, $\tilde{\mathbf{g}}_N[n] = \left[\tilde{g}_N^{(1)}[n] \quad \tilde{g}_N^{(2)}[n]\right]^T$ is the corresponding vector at the n-th ST interval, and $\hat{\mathbf{g}}[n] = \left[\hat{g}_1[n] \quad \hat{g}_2[n]\right]^T$ is the hypothesized fading-plus-noise pattern. By performing further simplification, (85) can be rewritten as

$$q_N[n] = q_N^{(1)}[n] + q_N^{(2)}[n], \tag{86}$$

where

$$\begin{aligned} q_N^{(1)}[n] &= \mathrm{Re}\left\{w_1^*[n]\hat{d}_1[n]\right\} \\ q_N^{(2)}[n] &= \mathrm{Re}\left\{w_2^*[n]\hat{d}_2[n]\right\}, \end{aligned} \tag{87}$$

$\hat{d}_i[n]$, $i = 1,2,$ are hypotheses of the data symbols in (11),

$$
\begin{aligned}
w_1^*[n] &= r_1^*[n]a_N^{(1)}[n] + r_2[n]a_N^{(2)*}[n] \\
w_2^*[n] &= r_1^*[n]a_N^{(2)}[n] - r_2[n]a_N^{(1)*}[n]
\end{aligned}
\tag{88}
$$

are the conjugate of the fading-compensated samples, and

$$
\mathbf{a}_N[n] = \begin{bmatrix} a_N^{(1)}[n] \\ a_N^{(2)}[n] \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \hat{\mathbf{s}}[n-1]\tilde{\mathbf{g}}_N[n]. \tag{89}
$$

Now the estimated data symbols $\hat{d}_1[n]$ and $\hat{d}_2[n]$ can be easily computed by calculating the phase angles $\theta_i[n] = \arg\{w_i[n]\}$, $i = 1,2,$ and then locating the MPSK constellation points that are closest in phase to $\theta_i[n]$'s. The estimated data symbols can be calculated by

$$
\hat{d}_i[n] = \exp\left(j\frac{2\pi}{M} \times round\left(\frac{\theta_i[n] \times M}{2\pi} \right) \right), \quad i = 1,2. \tag{90}
$$

After the estimated data symbols are found, the decoder slides its window forward by one ST symbol and starts another forward prediction on the next received vector $\mathbf{r}[n+1] = [r_1[n+1], r_2[n+1]]^T$.

4.4 ST-MSDSD

The sphere decoding algorithm used for our Alamouti-type ST system is exactly the same as the one used for the SISO channel except that the decoder now makes use of the new branch sorting algorithm introduced in Section 4.2. Figure 4-5 highlights the portion where the new sorting routine is implanted.

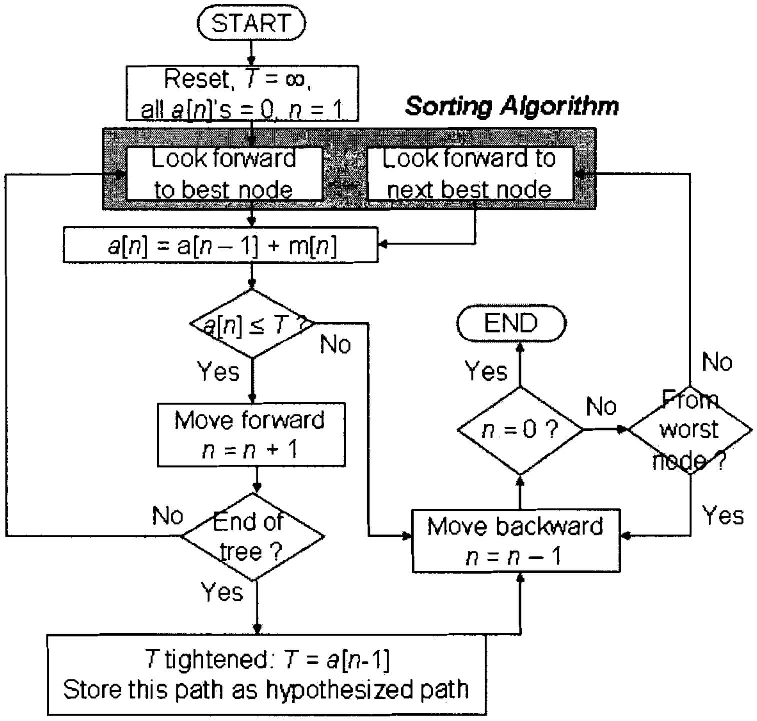

Figure 4-5. Flowchart of the ST-MSDSD.

Recalled from the simulation results presented in the previous chapter, we notice that, at low SNR, the more outgoing branches a node has, the longer time the sphere decoder spends on roaming (or more paths are being explored by the decoder). We may deduce that the sphere decoding process in ST systems requires more computations than the one in SISO systems. This stems from the fact that there are now M^2 branches emerging from each node. This huge increase in complexity leads us to the investigation of the Fano algorithm as an alternative decoding strategy for a ST-MSDD receiver.

4.5 The Fano ST-MSDDs

This section deals with the Fano algorithm applied to the Alamouti-type ST system. It can be seen as a natural extension of Chapter 3. Recalled from the expression (70) in Section 4.1, the branch metrics represent the normalized square prediction errors.

When the hypothesized pattern $\hat{\mathbf{S}}$ equals the transmitted pattern $\mathbf{S}$, then these errors will be small. Furthermore, each branch metric has an expected value of 2 and the APM

$$M\left(\hat{\mathbf{G}}[n]\right) = \sum_{i=1}^{n} m[i] \tag{91}$$

will fluctuate around a straight line with a positive slope of 2. On the other hand when $\hat{\mathbf{S}} \neq \mathbf{S}$, the branch metrics can be quite large because of the abrupt transitions in the hypothesized fading-plus-noise sequence. In this case, the APM will, at some point, deviate suddenly from the straight line mentioned above. This difference in the APM's characteristics in the two cases enables the Fano ST-MSDD to make accurate decisions on the transmitted data without the need to search exhaustively through the decoding tree.

As introduced in Section 3.4, the Fano decoder is essentially an intelligent decision feedback detector that employs a dynamic threshold to check if it is tracking the correct path in the decoding tree. In order to maintain consistency with our usual notions of threshold tightening/relaxation when describing the Fano ST-MSDD, we adopt a similar biased path metric

$$M_b\left(\hat{\mathbf{G}}[N-1]\right) = \sum_{n=1}^{N-1} m_b[n], \tag{92}$$

where the biased branch metric $m_b[n] = m[n] - b$ is given in (42). The bias b is chosen such that the average value of $m_b[n]$ is negative when conditioned on $\hat{\mathbf{S}} = \mathbf{S}$. From the above discussion, we can deduce that $b > 2$. With the introduction of the bias, the transmitted pattern will have the most negative metric amongst all the candidates (most of the time), and the average slope of its biased APM, $v[n] = \sum_{k=1}^{n} m_b[k]$ in (43), is negative and is equal to $2-b$.

4.5.1 Fano ST-MSDD

After the biased APM has been formulated, the Fano decoder can now use this APM along with the running threshold T to help in decoding the distance tree. As described earlier, the Fano decoder can be viewed as an intelligent decision feedback detector, which knows probabilistically if it is following a correct decision path. It can also be viewed as a device that performs an initial pass of a sphere decoder. However, the initial pass performed by a Fano decoder is more promising than the one performed by a sphere decoder because of its ability to backtrack and update its running threshold dynamically. Furthermore, numerical results show that, if an additional pass is performed in the backward direction, this bidirectional Fano decoder can achieve a BEP performance close to the one of a sphere decoder. This further illustrates the practically usefulness of the Fano algorithm in a ST-MSDD receiver.

The Fano algorithm used by our Fano ST-MSDD receiver is the same as the improved Fano algorithm discussed in Section 3.4.2 except that the new sorting routine described in Section 4.2 is engaged. Figure 4-6 depicts the area where the sorting algorithm is adopted.

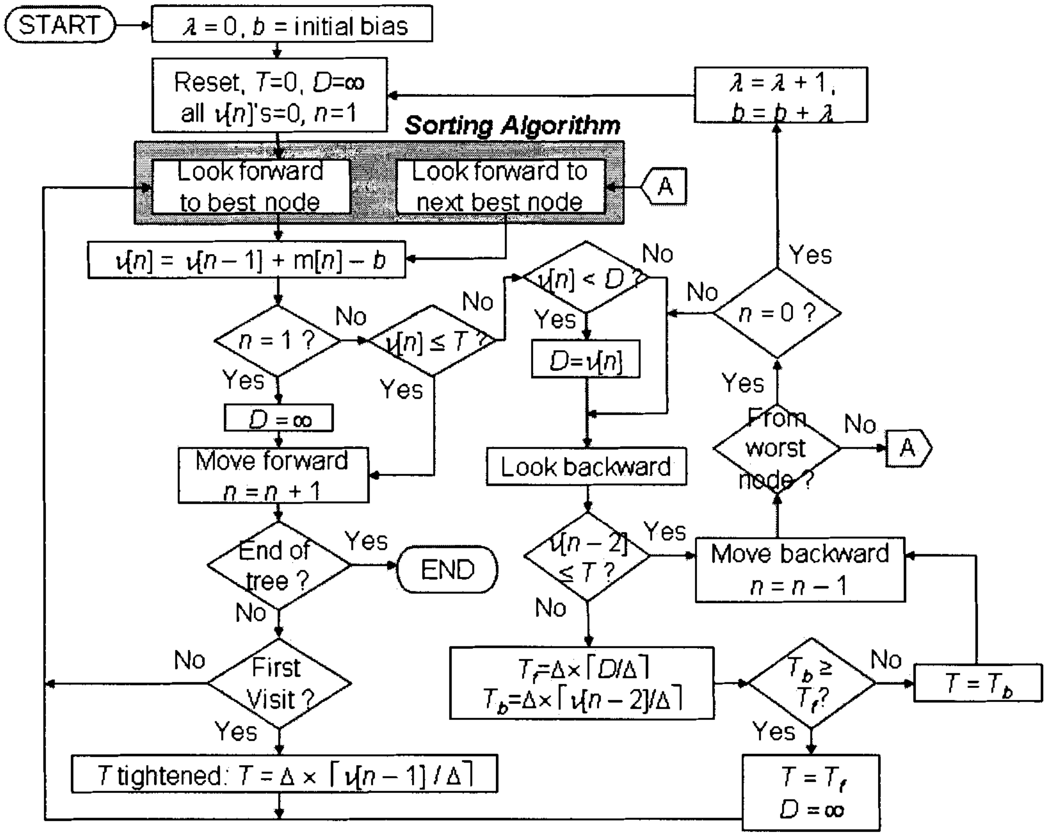

Figure 4-6. Flowchart of the Fano ST-MSDD.

4.5.2 Bi-Fano ST-MSDD

The Bi-Fano ST-MSDD is no different from the Bi-Fano MSDD in the SISO channel except there are some little changes in notation. The outcomes of the forward and backward searches are used to perform error detection and error correction. The decisions made by the forward decoder $\left(\hat{\mathbf{d}}_f[1],\hat{\mathbf{d}}_f[2],...,\hat{\mathbf{d}}_f[N-1]\right)$ and by the backward decoder $\left(\hat{\mathbf{d}}_b[1],\hat{\mathbf{d}}_b[2],...,\hat{\mathbf{d}}_b[N-1]\right)$ are compared to form a final decoded pattern. Similarly, if the two sequences are identical, then the final decoded pattern is equal to either one of them and no error correction is needed. However, if there are discrepancies at the positions

$\{n_1,\ n_2,\ \ldots,\ n_d\}$, the decoder explores all possible sequence of the form $\{\tilde{\mathbf{d}}_K[1],\ \tilde{\mathbf{d}}_K[2],\ \ldots,\ \tilde{\mathbf{d}}_K[N-1]\}$, $K=0,\ 1,\ \ldots,\ 2^{n_d}-1$, where

$$\tilde{\mathbf{d}}_K[n]=\begin{cases}\mathbf{z}[n], & n=n_i\\ \hat{\mathbf{d}}_f[n], & n\neq n_i,\end{cases} \tag{93}$$

$$\mathbf{z}[n_i]=\begin{cases}\hat{\mathbf{d}}_f[n_i], & b_i=0\\ \hat{\mathbf{d}}_b[n_i], & b_i=1,\end{cases} \tag{94}$$

and $\{b_1,\ b_2,\ \ldots,\ b_{n_d}\}$ is the binary equivalent of the integer K. Inherently, 2^{n_d} sequences are compared and the sequence with the smallest metric (92) is chosen as the final decoded pattern.

4.5.3 Edge-Bi-Fano ST-MSDD

As recalled, an Edge-Bi-Fano receiver is an edge-cut version of the Bi-Fano algorithm with an effective block size of $N_e=N+E$, where $N-1$ is the actual number of ST symbols the algorithm wants to decode, and $E=2H$ is the number of discarded ST symbols. Similarly, the Edge-Bi-Fano ST-MSDD receiver makes tentative decisions $\{\hat{\mathbf{d}}_f[1],\ \hat{\mathbf{d}}_f[2],\ \ldots,\ \hat{\mathbf{d}}_f[N_e-1]\}$ and $\{\hat{\mathbf{d}}_b[1],\ \hat{\mathbf{d}}_b[2],\ \ldots,\ \hat{\mathbf{d}}_b[N_e-1]\}$ on both the forward and the backward directions, respectively, and the first and the last H ST symbols in these two sequences are discarded. The surviving sequences, $\{\hat{\mathbf{d}}_f[H+i]\}_{i=1}^{N-1}$ and $\{\hat{\mathbf{d}}_b[H+i]\}_{i=1}^{N-1}$, are then used to form the final decoded pattern by carrying out the same procedure as described in the preceding Section 4.5.2.

4.6 Simulation Results and Discussion

We present in this section the simulation results for the forward Fano, Bi-Fano, and the Edge-Bi-Fano ST-MSDDs. In addition to the BEP and the computational complexity, we also provide the average numbers of node transitions, bias adjustments, and the probability of a differed ST-symbol, just like in the SISO case. All simulation results are obtained based on ST-QDPSK transmission in both the fast $(f_d T_{ST} = 0.03)$ and the static fading scenarios. The bit SNR used in our simulation is defined earlier in (19); see Section 2.3. In order to maintain consistency with our results for the SISO channel, we adopt a bias of $b = 3$ and a step size of $\Delta = 3$ so that the overall expected slope of the biased APM becomes -1. On the other hand, the BEPs and the implementation complexities of the ST-MSDSD and the ST-DFDD are served as performance benchmarks. It is worth mentioning that the ST-DFDD has perfect knowledge of the initial transmitted pattern $[s_i[0]\ s_i[1]\ \dots\ s_i[N-2]]^T$, $i = 1,2,$ and its error curves shown in the plots essentially represent the BEP lower bounds of the ST-DFDD detector.

4.6.1 Bit-Error Performance

The bit-error performance of the Fano, the Bi-Fano, and the Edge-Bi-Fano ST-MSDDs are plotted in Figure 4-7 and Figure 4-8 for the fast fading channel, and in Figure 4-9 and Figure 4-10 for the static fading channel. Also shown in the figures are the BEPs of the ST-MSDSD, the ST-DFDD, and the coherent detector with ST-QPSK transmission. In a fast fading environment, the Fano detectors do not show any observable irreducible error floor. At a BEP of 10^{-4}, the N_e=8 (i.e. $N = 6$ and $E = 2$) Edge-

Bi-Fano detector outperforms the ST-MSDSD, the Bi-Fano, the forward Fano, and the ST-DFDD by 0.6 dB, 1 dB, 2 dB, and 2 dB respectively. In addition, the BEP of a Bi-Fano detector with the same effective block size of $N = 8$ is plotted to show that the Edge-Bi-Fano detector can provide substantial improvement over a Bi-Fano detector with the same effective block size. It is observed that, at the same BEP of 10^{-4}, the Edge-Bi-Fano is 0.5 dB more power efficient than the Bi-Fano counterpart. When we increase the block size to $N = 10$, all of the ST-MSDDs have an additional 1 dB improvement in power efficiency.

In the static fading environment, Figure 4-9 shows that, at a BEP of 10^{-5}, the $N_e = 8$ (i.e. $N = 6$ and $E = 2$) Edge-Bi-Fano detector is 0.5 dB and 1.5 dB more power efficient than the Bi-Fano and the forward Fano detectors respectively while it is only 0.2 dB and 2.3 dB away from the ST-MSDSD and the coherent detector. When we compare the Edge-Bi-Fano with the ST-DFDD at the same BEP of 10^{-5}, we observe a 1.5 dB improvement in power efficiency. When the block size is increased to $N = 10$, all of the non-coherent detectors offer little improvement over their $N = 6$ counterparts in Figure 4-9 due to the diminishing return as observed in the SISO channel.

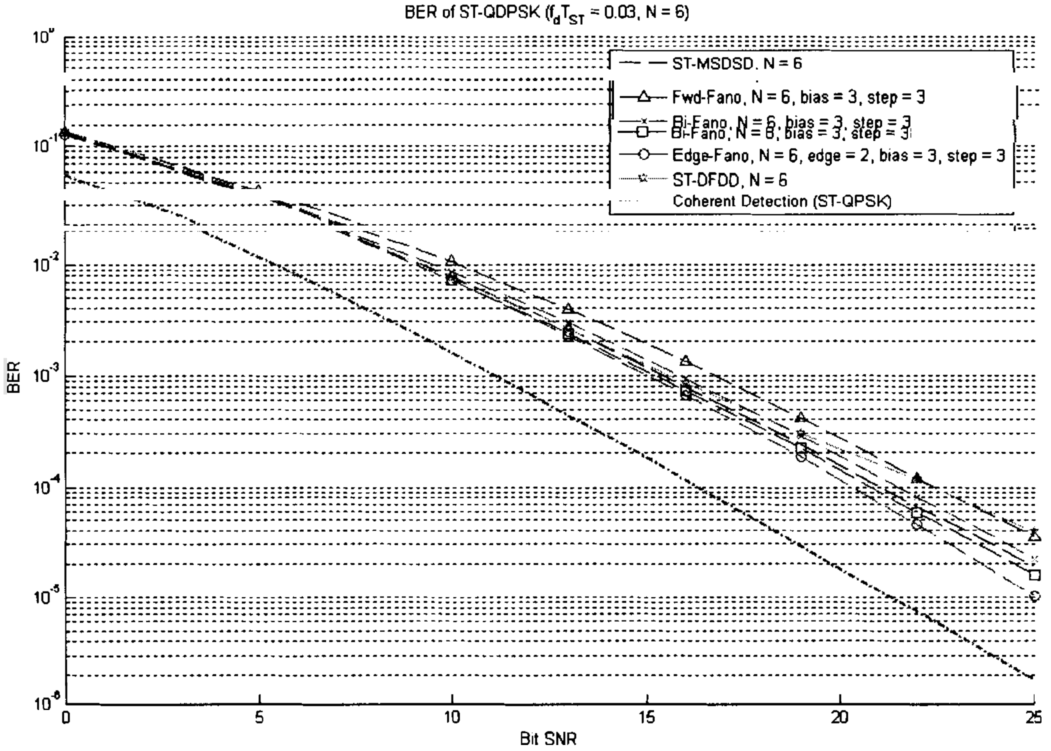

Figure 4-7. BEP versus bit SNR for ST-QDPSK at N = 6 and f_dT_{ST} = 0.03.

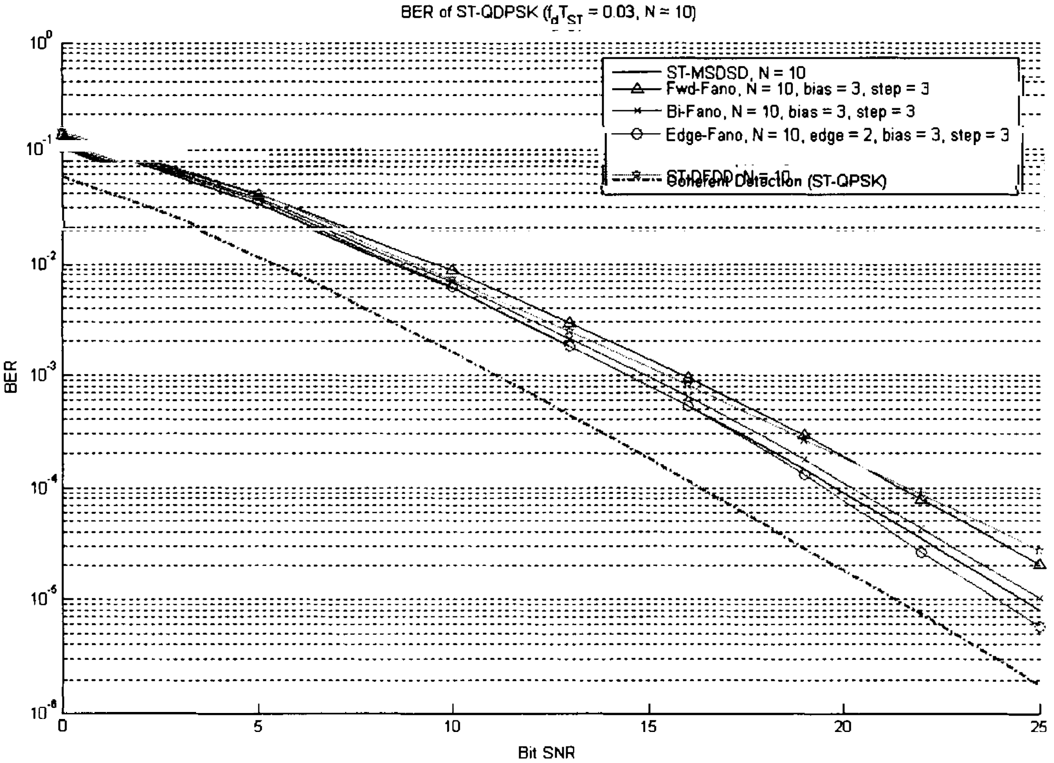

Figure 4-8. BEP versus bit SNR for ST-QDPSK at N = 10 and f_dT_{ST} = 0.03.

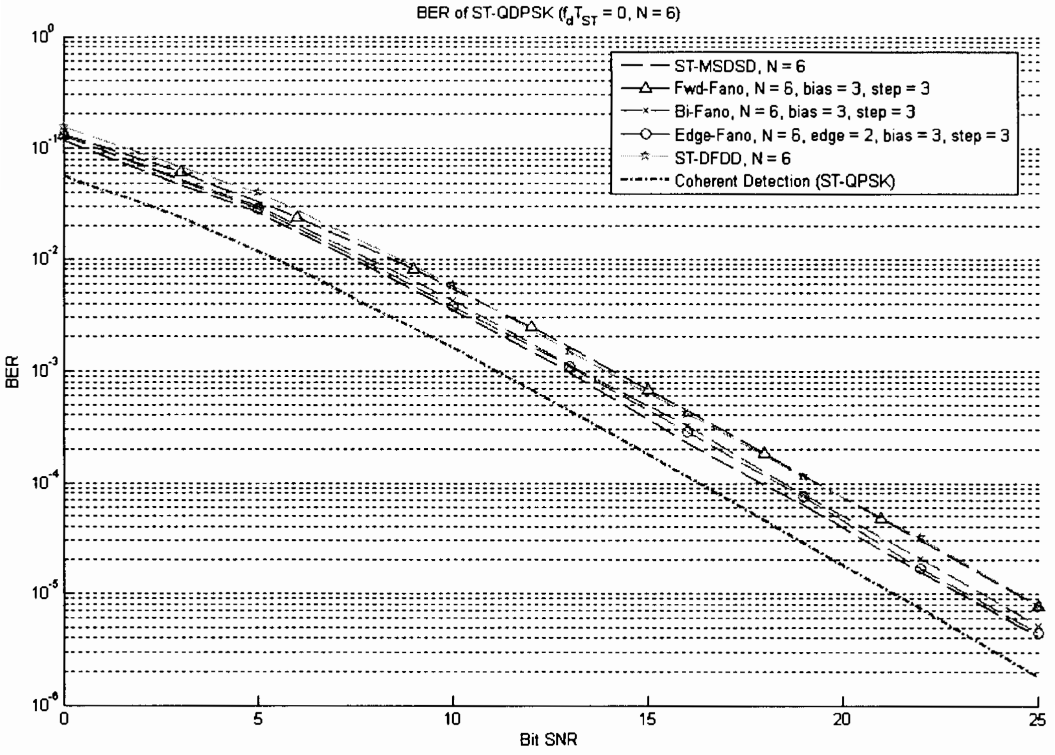

Figure 4-9. BEP versus bit SNR for ST-QDPSK at *N* = 6 and static fading.

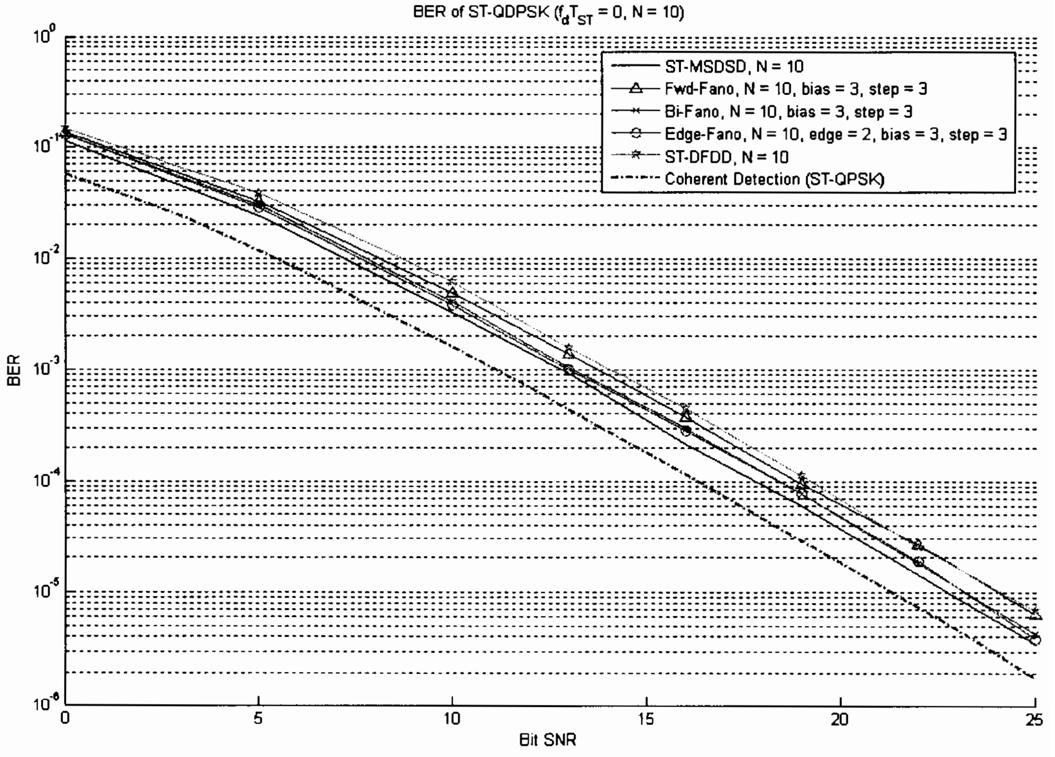

Figure 4-10. BEP versus bit SNR for ST-QDPSK at *N* = 10 and static fading.

4.6.2 Computational Complexity

For computational complexity, we consider both the average number of real multiplications and comparisons per block of N ST symbols and these results are plotted in Figure 4-11 and Figure 4-12 respectively for the fast fading scenario $(f_d T_{ST} = 0.03)$. The computational complexity curves of the ST-DFDD are also plotted as benchmarks.

As shown in both plots, the complexities of our Fano ST-MSDDs are more manageable than the ST-MSDSD. For example, at an SNR of 0 dB, the $N = 10$ Bi-Fano detector requires approximately 5 x 10^4 multiplications and 820 comparisons whereas the sphere decoder requires 2 x 10^7 multiplications and 2 x 10^6 comparisons. This translates into savings of 2.6 and 3.4 orders of magnitude in multiplications and comparisons respectively. Furthermore, the Bi-Fano ST-MSDD has lower complexity than the ST-MSDSD when the SNR is below 17 dB and, at this SNR, the Bi-Fano detector is only 0.3 dB less efficient than the sphere decoder in power efficiency. It is more interesting to see that, in Figure 4-12, the $N = 10$ Bi-Fano detector has less comparisons than the $N = 6$ sphere decoder when SNR is blow 7.5 dB.

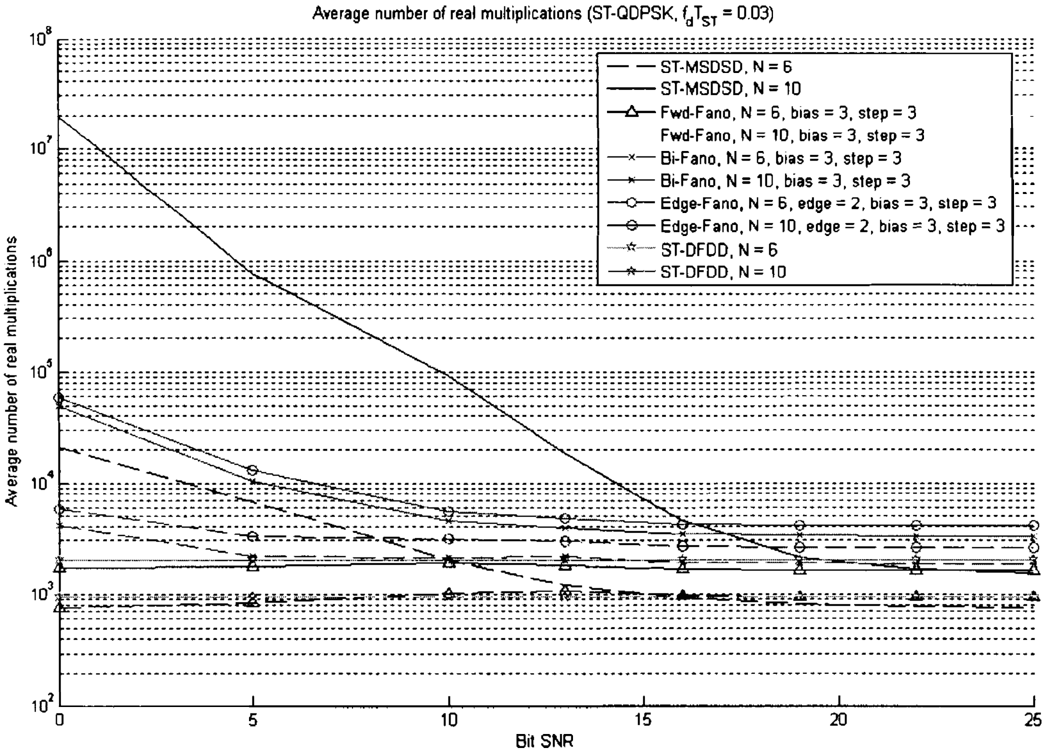

Figure 4-11. Average number of real multiplications versus bit SNR for ST-QDPSK at f_dT_{ST} = 0.03.

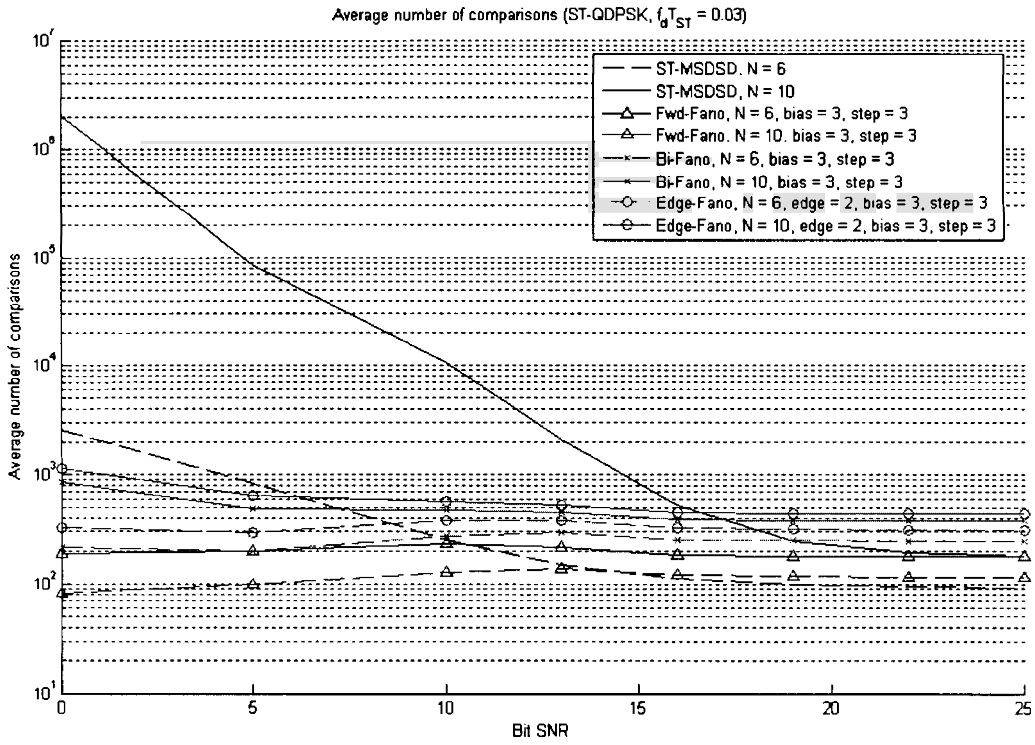

Figure 4-12. Average number of comparisons versus bit SNR for ST-QDPSK at f_dT_{ST} = 0.03.

4.6.3 Other Statistical Measurements for Fano ST-MSDD

Similar to what we have illustrated in Chapter 3, the average numbers of node transitions, search failures (or bias re-adjustments), and the probability of a differed ST-symbol are simulated and they are plotted in Figure 4-13, Figure 4-14, and Figure 4-15 respectively. As shown in Figure 4-14, we observe a growth in the number of search failures as the SNR is increasing. However, when we compare it with the one for QDPSK in Figure 3-27, the curves shown in Figure 4-14 seem to reach their asymptotic bounds whereas the SISO curves still keep on growing even at an SNR of 40 dB. As recalled, the growth is associated with the frequency of deep fades occurring at both ends of the decoding block. Since our ST system has two independent transmission links, the probability that both links are in deep fades is relatively small and hence the number of search failures for our Fano ST-MSDDs is expected to be lower than that of the SISO detectors.

In addition, when we compare Figure 4-15 with its SISO counterpart in Figure 3-29, both plots show that the Edge-Bi-Fano detector requires less error corrections than that of the Bi-Fano detector in the final decision process. With a block size of $N = 10$ and an SNR of 25 dB, the probability of error correction required per symbol is 4.6 x 10^{-5} for the Edge-Bi-Fano and 9.2 x 10^{-5} for the Bi-Fano. These results once again suggest that the complexity of the Bi-Fano and the Edge-Bi-Fano detectors can be reduced if an intelligent way of performing bidirectional search on demand is found. We will suggest in Chapter 5 that it can be accomplished by monitoring the instantaneous energy level of the received signals.

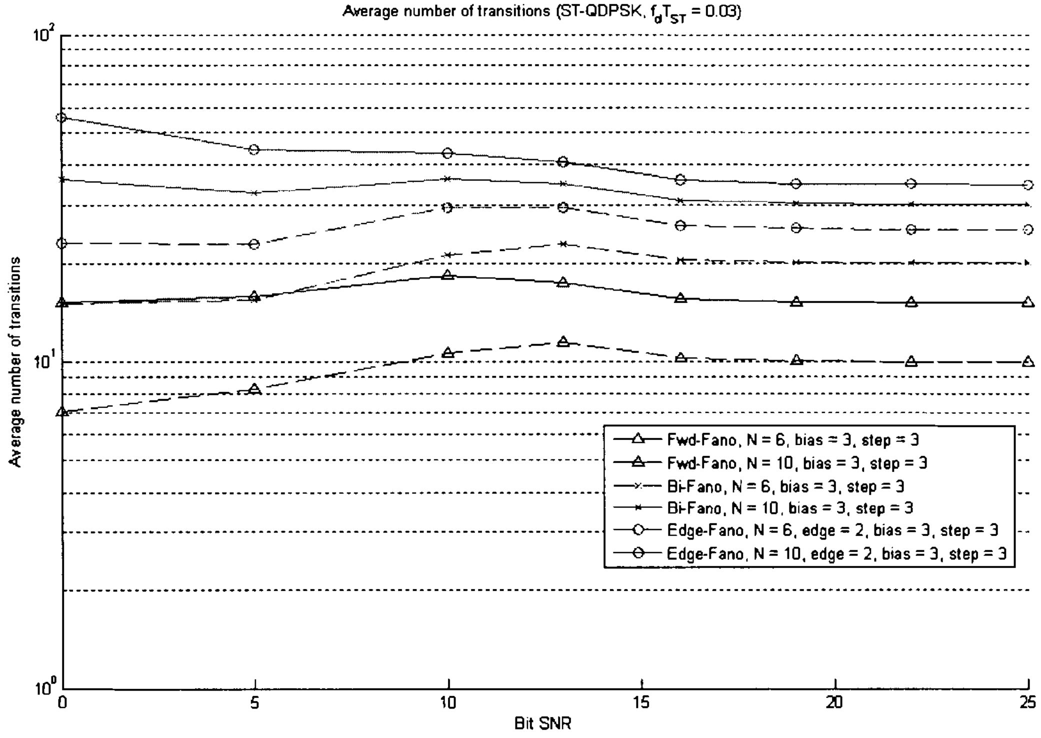

Figure 4-13. Average number of node transitions versus bit SNR for ST-QDPSK at f_dT_{ST} = 0.03.

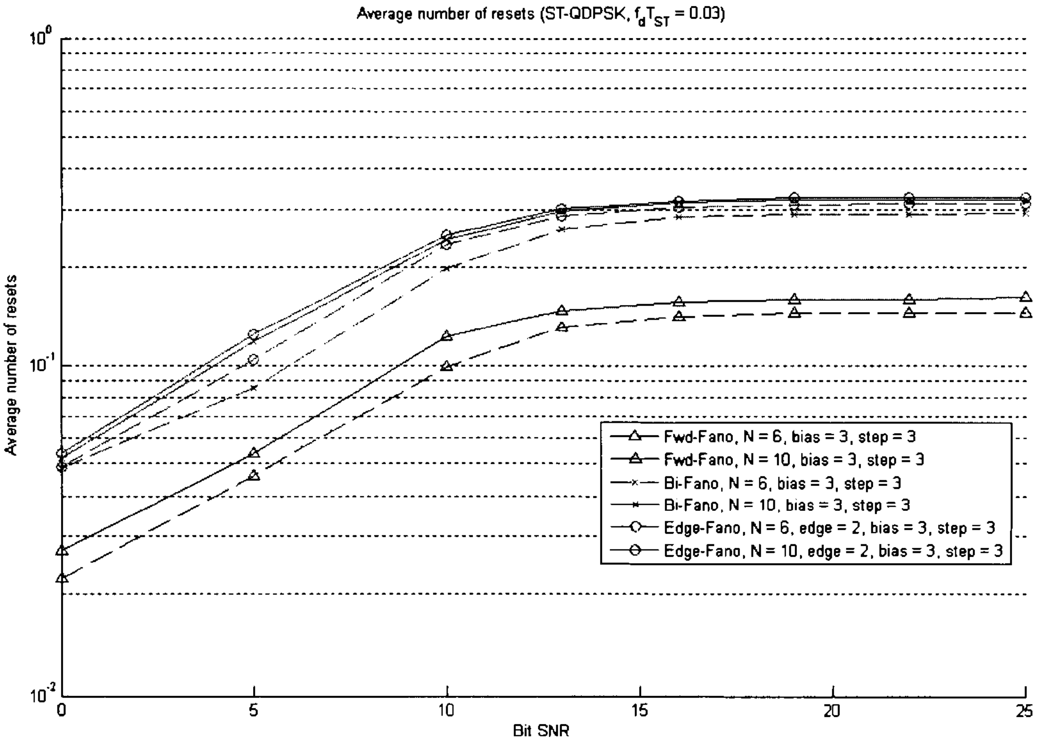

Figure 4-14. Average number of search failures versus bit SNR for ST-QDPSK at f_dT_{ST} = 0.03.

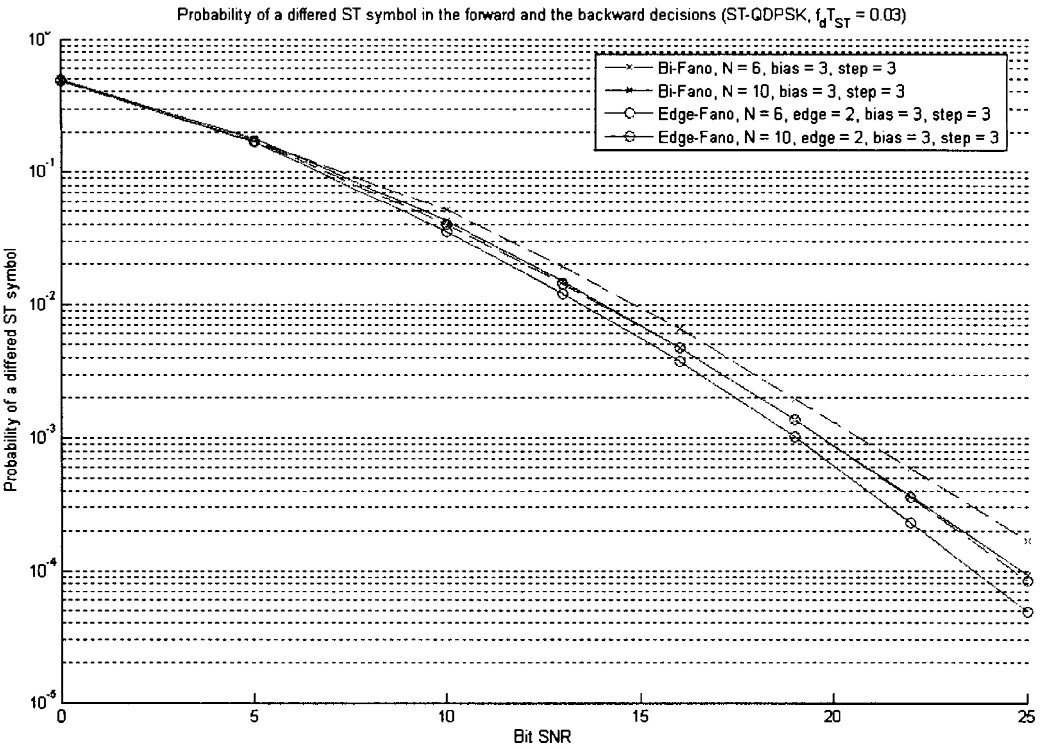

Figure 4-15. Probability of a different detected symbol for ST-QDPSK at $f_d T_{ST} = 0.03$.

CHAPTER 5: CONCLUSION

5.1 Conclusion

In the thesis, the Fano-MSDDs and its ST variants, Fano ST-MSDDs, are discussed and analyzed. Their counterparts, the MSDSD and the DFDD, are reviewed and used as performance benchmarks. Numerical results show that our Fano detectors are capable of delivering near-ML performance at modest implementation complexity and robust enough to operate over a wide range of fading rates. When compared to the sphere decoders, our Bi-Fano detectors reduce the number of computations by two to three orders of magnitude when SNR is extremely low. For example, at an SNR of 0 dB, the $N = 10$ Bi-Fano detector requires ten times less computations than the sphere decoder for QPSK and almost 100 to 1000 times less for ST-QDPSK. Unfortunately, at large SNR, the Bi-Fano detector requires twice as many computations as that of the sphere decoder while it still suffers from a 0.5 dB to 1 dB degradation in power efficiency. However, the Bi-Fano detector is an attractive alternative to the DFDD detector. It is observed that, with a normalized Doppler frequency of 0.03, the Bi-Fano detector can provide up to 4 dB gain in power efficiency for QDPSK and 2.5 dB for ST-QDPSK, and most importantly, it does not exhibit any noticeable irreducible error floor.

In addition, we observe that the computational complexity gap between the Edge-Bi-Fano and the Bi-Fano shrinks as the block size N increases. In other words, the Edge-Bi-Fano can be more computationally efficient as N is large. In fact, the decoding complexity of the Edge-Bi-Fano can be roughly estimated to be $C = (N + E - 1)/(N - 1)$ times that of the Bi-Fano and C can approach unity when N is infinitely large. However, we would like to emphasize that the performance gain provided by the Edge-Bi-Fano over the Bi-Fano can decrease with the block size *N* due to the diminishing return of an MSDD receiver. Consequently, the Edge-Bi-Fano is most suitable when a moderate window size is used.

Besides fading, the instantaneous SNR at the receiver is subject to fluctuation while the receiver is operating in a multiuser environment. Since the error performance of our receivers strongly depends on the accuracy of the channel statistics, the sudden change in instantaneous SNR due to multiuser interference may result in a decoding metric that is mismatched to the actual channel statistics. This mismatch in channel statistics can result in larger prediction error that makes the decoder harder to differentiate the correct path from the erroneous paths. Consequently, more backtracking may happen in the decoding process. However, this mismatch does not affect our receiver in the statistical sense because, in Rayleigh fading environment, the BEP of our receiver is a linear function of SNR and the complexity is almost independent of it. If we average the instantaneous values of the BEP and the complexity over the range of fluctuating SNR, their expected values does not change, and hence, on average, multiuser interference does not constitute to any performance degradation in our receivers.

In addition, our simulation results suggest several improvements for our Fano detectors. We present in the next section some possible improvements that can further enhance the BEP performance and reduce the implementation complexity of our receivers.

5.2 Suggestions for Future Work

As discussed earlier, the computational complexity of the bi-directional Fano decoders, Bi-Fano and Edge-Bi-Fano, can be reduced significantly if the decoders can perform bi-directional search on demand. Numerical results show that the probability of having a differed symbol in the forward and the backward tentative decisions is very small, especially in the MIMO channel and at large SNR. It suggests that, almost always, a forward search is enough and no error correction is required. However, it comes to the question on how the decoder knows whether an additional backward search is needed. In fact, it can be easily resolved by 1) examining the number of backtracking in the forward decoding process, and/or 2) monitoring the instantaneous energy level of the received signals. It is apparent that, when deep fades happen, the instantaneous received energy can drop tens of dB below its mean value. As a result, the estimated transmitted pattern provided by the forward Fano decoder is more prone to error due to low instantaneous SNR. To mitigate this problem, error detection and error correction can be made by performing a bi-directional search only when the received energy level falls below a certain threshold. By adopting this adaptive approach, we believe that the complexity gap between the sphere decoder and the Bi-Fano can be tightened considerably.

Another possible refinement is the optimization of the search parameters (b, Δ). In general, a large bias b and a large step size Δ provide the decoder more tolerance to the sudden change in APM when a branch is struck by deep fades. Consequently, backtracking can be less frequent and hence the complexity can be much lower. In exchange, the BEP performance can be substantially degraded. However, tight search criteria usually results in large number of search failures or bias re-adjustments, especially when deep fades occur at both ends of the decoding block. In fact, this problem can be easily resolved by adopting the similar fade monitoring mechanism as discussed earlier. When deep fades are identified at the branches that are in the neighbourhood of the root node, the decoder can dynamically enlarge the biases b's of those branches in order to give more cushion against the sudden change in APM. However, we would like to emphasize that the biases have to be chosen very carefully because too large biases can result in the loss of power efficiency.

In addition, the bit-error performance of our Fano detectors can be further enhanced by fixing the order of their linear predictors, just like how the DFDD is implemented. As recalled, the DFDD receiver uses a finite order predictor $\mathbf{p}_N$ in the decoding process and slides its observation window by only one symbol at a time, instead of $N-1$ symbols. This receiver, termed the approximate ML sequence detection [25], is capable of delivering considerable amount of improvement in power efficiency because the decision on each transmitted symbol $s[k]$ is based on the same $(N-1)$-th order of linear prediction. Since the Fano decoder is considered as an intelligent decision-feedback detector, which shares many common features of the DFDD, it is straightforward to adopt this sequence detection strategy in our Fano detector. Note that this receiver structure is not compatible with the sphere decoder due to its nature of searching.

APPENDIX A: DERIVATION OF THE MAXIMUM LIKELIHOOD DECODING METRIC FOR BACKWARD LINEAR PREDICTIONS

We present in this appendix the derivation of an ML metric in the backward decoding direction for both the SISO channel and the Alamouti-type ST channel. The reader, who has already gone through the derivation of the ML metric in Chapter 3 and Chapter 4, can treat this appendix as a natural extension to the forward linear prediction with little changes in notations. All of the work shown below is applied in the implementation of the Bi-Fano and Edge-Bi-Fano detectors.

A.1 Backward ML Metric for SISO Systems

For a backward search, we first define a backward discrete time index $m = (N-2),\ (N-1),\ \ldots,\ 0,$ where m can be related to the forward time index $n = 1,\ 2,\ \ldots,\ (N-1)$ by $m = N - n - 1$. Let us define the m-th hypothesized fading-plus-noise pattern $\hat{\mathbf{g}}^{(b)}[m] = \left[\hat{g}[m]\ \hat{g}[m+1]\ \ldots\ \hat{g}[N-1]\right]^T$, where $\hat{g}[m] = \hat{s}^*[m]r[m]$ is the m-th hypothesized fading-plus-noise sample. The covariance matrix of the actual fading-plus-noise pattern $\mathbf{g}^{(b)}[m] = \left[g[m]\ g[m+1]\ \ldots\ g[N-1]\right]^T$ can be written as

$$\mathbf{\Phi}_{gg}[N-m-1] = \tfrac{1}{2}E\left[\mathbf{g}^{(b)}[m]\mathbf{g}^{(b)H}[m]\right] = \mathbf{\Phi}_{gg}[n]. \tag{95}$$

Note that the covariance matrix $\mathbf{\Phi}_{gg}[N-m-1]$ is interpreted as an $(N-m-1)$ by $(N-m-1)$ sub-matrix of the covariance matrix defined in (24) of Section 3.1. Now the quadratic form $\hat{\mathbf{g}}^{(b)H}[m]\mathbf{\Phi}_{gg}^{-1}[N-m-1]\hat{\mathbf{g}}^{(b)}[m]$ can be written as

$$\begin{aligned}&\hat{\mathbf{g}}^{(b)H}[m]\mathbf{\Phi}_{gg}^{-1}[N-m-1]\hat{\mathbf{g}}^{(b)}[m]\\&=\left[\hat{g}^{*}[m]\quad \hat{\mathbf{g}}^{(b)H}[m+1]\right]\begin{bmatrix}\phi[0] & \boldsymbol{\varphi}_g^{(b)}[N-m-2]\\ \boldsymbol{\varphi}_g^{(b)H}[N-m-2] & \mathbf{\Phi}_{gg}[N-m-2]\end{bmatrix}^{-1}\begin{bmatrix}\hat{g}[m]\\ \hat{\mathbf{g}}^{(b)}[m+1]\end{bmatrix},\end{aligned} \tag{96}$$

where $\phi[0]=\sigma_f^2+1$ is the variance of fading-plus-noise, and

$$\boldsymbol{\varphi}_g^{(b)}[N-m-2]=\frac{1}{2}E\left[g[m]\mathbf{g}^{(b)H}[m+1]\right]=\sigma_f^2\cdot\left[J_{-1},J_{-2},\ldots,J_{-(N-m-1)}\right] \tag{97}$$

is the correlation vector between the current and past fading-plus-noise samples in the backward direction. Note that this correlation vector is a row vector whereas the one in (31) of Section 3.1 is a column vector. The order of the Bessel coefficients is also reversed. By applying the same iterative procedure shown in Section 3.1, (96) can be simplified to

$$\begin{aligned}&\hat{\mathbf{g}}^{(b)H}[m]\mathbf{\Phi}_{gg}^{-1}[N-m-1]\hat{\mathbf{g}}^{(b)}[m]\\&\qquad=\hat{\mathbf{g}}^{(b)H}[m+1]\mathbf{\Phi}_{gg}^{-1}[N-m-2]\hat{\mathbf{g}}^{(b)}[m+1]\\&\qquad+\frac{\left|\hat{g}[m]-(\boldsymbol{\varphi}_g^{(b)}[N-m-2]\mathbf{\Phi}_{gg}^{-1}[N-m-2])\hat{\mathbf{g}}^{(b)}[m+1]\right|^2}{\varepsilon^2[N-m-1]},\end{aligned} \tag{98}$$

where

$$\varepsilon^2[N-m-1]=\phi[0]-\boldsymbol{\varphi}_g^{(b)}[N-m-2]\mathbf{\Phi}_{gg}^{-1}[N-m-2]\boldsymbol{\varphi}_g^{(b)H}[N-m-2] \tag{99}$$

is the mean square prediction error of an $(N-m-1)$-th order backward linear predictor. Note that $\varepsilon^2[N-m-1]$ is equivalent to $\varepsilon^2[n]$ as defined in (33). This stems from the fact that, on average, the square error of an i-th order forward linear prediction is the same as the one of the backward linear prediction with the same order.

Now the ML decoding metric can be written as

$$M\left(\hat{\mathbf{g}}^{(b)}[0]\right)=\sum_{m=N-2}^{0}\frac{\left|\hat{g}[n]-\left(\boldsymbol{\varphi}_g^{(b)}[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\right)\hat{\mathbf{g}}^{(b)}[m+1]\right|^2}{\varepsilon^2[N-m-1]}. \tag{100}$$

Note that the constant term $|\hat{g}[N-1]|^2/\phi[0]$ is omitted in the decoding process. The term $\boldsymbol{\varphi}_g^{(b)}[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]$ can be interpreted as an $(N-m-1)$-th order backward linear predictor.

A.2 Backward ML Metric for ST Systems

Let us define

$$\hat{\mathbf{G}}^{(b)}[m]=\begin{bmatrix}\hat{\mathbf{g}}[m]\\ \vdots\\ \hat{\mathbf{g}}[N-1]\end{bmatrix} \tag{101}$$

as the vector containing all hypothesized fading-plus-noise samples from the m-th ST interval up to the (N – 1)-th ST interval where $m=(N-2),\ (N-1),\ \ldots,\ 0$. Note that m is used as backward ST interval index whereas n is used as forward ST interval index and their relationship is $m=N-n-1$. The covariance matrix of the actual fading-plus-noise samples

$$\mathbf{G}^{(b)}[m]=\begin{bmatrix}\mathbf{g}[m]\\ \vdots \\ \mathbf{g}[N-1]\end{bmatrix}=\begin{bmatrix}\mathbf{s}^{H}[m]\mathbf{r}[m]\\ \vdots \\ \mathbf{s}^{H}[N-1]\mathbf{r}[N-1]\end{bmatrix} \tag{102}$$

can be defined as

$$\boldsymbol{\Psi}_{GG}[N-m-1]=\frac{1}{2}E\left[\mathbf{G}^{(b)}[m]\mathbf{G}^{(b)H}[m]\right]=\boldsymbol{\Phi}_{gg}[N-m-1]\otimes\mathbf{I}_2 \tag{103}$$

where $\boldsymbol{\Phi}_{gg}[N-m-1]$ is an $(N-m-1)$ by $(N-m-1)$ sub-matrix of the covariance matrix defined in equation (24) of Section 3.1. Now the quadratic form $\hat{\mathbf{G}}^{(b)H}[m]\boldsymbol{\Psi}_{GG}^{-1}[N-m-1]\hat{\mathbf{G}}^{(b)}[m]$ can be written as

$$\begin{aligned}
&\hat{\mathbf{G}}^{(b)H}[m]\boldsymbol{\Psi}_{GG}^{-1}[N-m-1]\hat{\mathbf{G}}^{(b)}[m]\\
&=\begin{bmatrix}\hat{\mathbf{g}}^{H}[m] & \hat{\mathbf{G}}^{(b)H}[m+1]\end{bmatrix}\begin{bmatrix}\phi_0\mathbf{I}_2 & \boldsymbol{\xi}_G^{(b)}[N-m-2]\\ \boldsymbol{\xi}_G^{(b)H}[N-m-2] & \boldsymbol{\Psi}_{GG}[N-m-2]\end{bmatrix}^{-1}\begin{bmatrix}\hat{\mathbf{g}}[m]\\ \hat{\mathbf{G}}^{(b)}[m+1]\end{bmatrix}\\
&=\begin{bmatrix}\hat{\mathbf{g}}^{H}[m] & \hat{\mathbf{G}}^{(b)H}[m+1]\end{bmatrix}\begin{bmatrix}\phi_0\otimes\mathbf{I}_2 & \boldsymbol{\varphi}_g^{(b)}[N-m-2]\otimes\mathbf{I}_2\\ \boldsymbol{\varphi}_g^{(b)H}[N-m-2]\otimes\mathbf{I}_2 & \boldsymbol{\Phi}_{gg}[N-m-2]\otimes\mathbf{I}_2\end{bmatrix}^{-1}\begin{bmatrix}\hat{\mathbf{g}}[m]\\ \hat{\mathbf{G}}^{(b)}[m+1]\end{bmatrix},
\end{aligned} \tag{104}$$

where $\phi_0=\sigma_f^2+1$. The correlation vector of the fading-plus-noise vectors $\mathbf{g}[m]$ and $\mathbf{G}^{(b)}[m+1]$ is defined as

$$\boldsymbol{\xi}_G^{(b)}[N-m-2]=\frac{1}{2}E\left[\mathbf{g}[m]\mathbf{G}^{(b)H}[m+1]\right]=\boldsymbol{\varphi}_g^{(b)}[N-m-2]\otimes\mathbf{I}_2, \tag{105}$$

where $\boldsymbol{\varphi}_g^{(b)}[N-m-2]$ is the correlation vector defined in (97).

Applying the same iterative procedure shown in Section 4.1 yields

$$\begin{aligned}
&\hat{\mathbf{G}}^{(b)H}[m]\boldsymbol{\Psi}_{GG}^{-1}[N-m-1]\hat{\mathbf{G}}^{(b)}[m] \\
&\quad = \hat{\mathbf{G}}^{(b)H}[m+1]\boldsymbol{\Psi}_{GG}^{-1}[N-m-2]\hat{\mathbf{G}}^{(b)}[m+1] + \hat{\mathbf{g}}^{H}[m]\mathbf{H}^{-1}[m]\hat{\mathbf{g}}[m] \\
&\quad - \hat{\mathbf{G}}^{(b)H}[m+1]\boldsymbol{\Psi}_{GG}^{-1}[N-m-2]\left(\boldsymbol{\varphi}_{g}^{(b)H}[N-m-2]\otimes\mathbf{I}_2\right)\mathbf{H}^{-1}[m]\hat{\mathbf{g}}[m] \\
&\quad - \hat{\mathbf{g}}^{H}[m]\mathbf{H}^{-1}[m]\left(\boldsymbol{\varphi}_{g}^{(b)}[N-m-2]\otimes\mathbf{I}_2\right)\boldsymbol{\Psi}_{GG}^{-1}[N-m-2]\hat{\mathbf{G}}^{(b)}[m+1] \\
&\quad + \hat{\mathbf{G}}^{(b)H}[m+1]\boldsymbol{\Psi}_{GG}^{-1}[N-m-2]\mathbf{D}[m]\boldsymbol{\Psi}_{GG}^{-1}[N-m-2]\hat{\mathbf{G}}^{(b)}[m+1],
\end{aligned} \tag{106}$$

where

$$\mathbf{D}[m] = \left(\boldsymbol{\varphi}_{g}^{(b)H}[N-m-2]\otimes\mathbf{I}_2\right)\mathbf{H}^{-1}[m]\left(\boldsymbol{\varphi}_{g}^{(b)}[N-m-2]\otimes\mathbf{I}_2\right) \tag{107}$$

and

$$\begin{aligned}
\mathbf{H}[m] &= \phi_0\mathbf{I}_2 - \left(\boldsymbol{\varphi}_{g}^{(b)}[N-m-2]\otimes\mathbf{I}_2\right)\left(\boldsymbol{\Phi}_{gg}[N-m-2]\otimes\mathbf{I}_2\right)^{-1}\left(\boldsymbol{\varphi}_{g}^{(b)H}[N-m-2]\otimes\mathbf{I}_2\right) \\
&= \phi_0\mathbf{I}_2 - \left(\boldsymbol{\varphi}_{g}^{(b)}[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\boldsymbol{\varphi}_{g}^{(b)H}[N-m-2]\right)\otimes\mathbf{I}_2.
\end{aligned} \tag{108}$$

Since the inverse covariance sub-matrices $\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]$ and $\boldsymbol{\Phi}_{gg}^{-1}[n-1]$ are equivalent for $n = N-m-1$, and the backward correlation vector $\boldsymbol{\varphi}_{g}^{(b)}[N-m-2]$ in (97) is just the reversed order of the forward correlation vector $\boldsymbol{\varphi}_{g}[n-1]$ as defined in (31), (108) can be re-written as

$$\mathbf{H}[m] = \mathbf{K}[N-m-1] = \varepsilon^2[N-m-1]\mathbf{I}_2, \tag{109}$$

where $\varepsilon^2[N-m-1]$ is defined in (99). Substituting (109) into (106) yields

$$\begin{aligned}
&\hat{\mathbf{G}}^{(b)H}[m]\boldsymbol{\Psi}_{GG}^{-1}[N-m-1]\hat{\mathbf{G}}^{(b)}[m] \\
&\quad= \hat{\mathbf{G}}^{(b)H}[m+1]\boldsymbol{\Psi}_{GG}^{-1}[N-m-2]\hat{\mathbf{G}}^{(b)}[m+1] + \frac{1}{\varepsilon^2[N-m-1]}\hat{\mathbf{g}}^H[m]\hat{\mathbf{g}}[m] \\
&\quad- \frac{1}{\varepsilon^2[N-m-1]}\hat{\mathbf{G}}^{(b)H}[m+1]\left\{\left(\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\boldsymbol{\varphi}_g^{(b)H}[N-m-2]\right)\otimes\mathbf{I}_2\right\}\hat{\mathbf{g}}[m] \\
&\quad- \frac{1}{\varepsilon^2[N-m-1]}\hat{\mathbf{g}}^H[m]\left\{\left(\boldsymbol{\varphi}_g^{(b)}[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\right)\otimes\mathbf{I}_2\right\}\hat{\mathbf{G}}^{(b)}[m+1] \\
&\quad+ \frac{1}{\varepsilon^2[N-m-1]}\hat{\mathbf{G}}^{(b)H}[m+1]\left(\mathbf{E}[m]\otimes\mathbf{I}_2\right)\hat{\mathbf{G}}^{(b)}[m+1],
\end{aligned} \tag{110}$$

where

$$\mathbf{E}[m] = \boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\boldsymbol{\varphi}_g^{(b)H}[N-m-2]\boldsymbol{\varphi}_g^{(b)}[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]. \tag{111}$$

The second term in (110) is simply

$$\frac{1}{\varepsilon^2[N-m-1]}\hat{\mathbf{g}}^H[m]\hat{\mathbf{g}}[m] = \frac{1}{\varepsilon^2[N-m-1]}\left(\left|\hat{g}_1[m]\right|^2 + \left|\hat{g}_2[m]\right|^2\right). \tag{112}$$

Let

$$\hat{\mathbf{G}}_1^{(b)}[m+1] = \begin{bmatrix} \hat{g}_1[m+1] \\ \vdots \\ \hat{g}_1[N-2] \\ \hat{g}_1[N-1] \end{bmatrix} \tag{113}$$

and

$$\hat{\mathbf{G}}_2^{(b)}[m+1] = \begin{bmatrix} \hat{g}_2[m+1] \\ \vdots \\ \hat{g}_2[N-2] \\ \hat{g}_2[N-1] \end{bmatrix} \tag{114}$$

be the estimated fading-plus-noise vectors from transmit antennas 1 and 2 respectively.

Since

$$\left\{\left(\boldsymbol{\varphi}_g^H[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\right)\otimes\mathbf{I}_2\right\}\hat{\mathbf{G}}^{(b)}[m+1] \\ =\begin{bmatrix}\boldsymbol{\varphi}_g^H[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\hat{\mathbf{G}}_1^{(b)}[m+1]\\ \boldsymbol{\varphi}_g^H[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\hat{\mathbf{G}}_2^{(b)}[m+1]\end{bmatrix}, \tag{115}$$

the last two terms in (110) can be written as

$$\frac{1}{\varepsilon^2[N-m-1]}\hat{\mathbf{g}}^H[m]\left\{\left(\boldsymbol{\varphi}_g^{(b)}[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\right)\otimes\mathbf{I}_2\right\}\hat{\mathbf{G}}^{(b)}[m+1] \\ =\frac{1}{\varepsilon^2[N-m-1]}\left\{\hat{g}_1^*[m]\left(\boldsymbol{\varphi}_g^H[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\hat{\mathbf{G}}_1^{(b)}[m+1]\right)\right\} \\ +\frac{1}{\varepsilon^2[N-m-1]}\left\{\hat{g}_2^*[m]\left(\boldsymbol{\varphi}_g^H[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\hat{\mathbf{G}}_2^{(b)}[m+1]\right)\right\} \tag{116}$$

and

$$\frac{1}{\varepsilon^2[N-m-1]}\hat{\mathbf{G}}^{(b)H}[m+1]\left(\mathbf{E}[m]\otimes\mathbf{I}_2\right)\hat{\mathbf{G}}^{(b)}[m+1] \\ =\frac{1}{\varepsilon^2[N-m-1]}\left\{\hat{\mathbf{G}}_1^{(b)H}[m+1]\mathbf{E}[m]\hat{\mathbf{G}}_1^{(b)}[m+1]\right\} \\ +\frac{1}{\varepsilon^2[N-m-1]}\left\{\hat{\mathbf{G}}_2^{(b)H}[m+1]\mathbf{E}[m]\hat{\mathbf{G}}_2^{(b)}[m+1]\right\}. \tag{117}$$

After substituting (112)-(117) in (110), we arrive at a backward recursive equation

$$\hat{\mathbf{G}}^{(b)H}[m]\boldsymbol{\Psi}_{GG}^{-1}[N-m-1]\hat{\mathbf{G}}^{(b)}[m] \\ =\hat{\mathbf{G}}^{(b)H}[m+1]\boldsymbol{\Psi}_{GG}^{-1}[N-m-2]\hat{\mathbf{G}}^{(b)}[m+1] \\ +\frac{1}{\varepsilon^2[N-m-1]}\sum_{i=1}^{2}\left|\hat{g}_i[m]-\left(\boldsymbol{\varphi}_g^{(b)}[N-m-2]\boldsymbol{\Phi}_{gg}^{-1}[N-m-2]\right)\hat{\mathbf{G}}_i^{(b)}[m+1]\right|^2. \tag{118}$$

Note that the recursive equation of the backward prediction is very similar to the one of forward prediction except that m is counting backward from $(N-2)$ to 0. Furthermore, the total accumulated metric for the backward prediction with a block size of N can be written as

$$\begin{aligned}&\hat{\mathbf{G}}^{(b)H}[0]\mathbf{\Psi}_{GG}^{-1}[N-1]\hat{\mathbf{G}}^{(b)}[0] \\ &= m[N-1] + \sum_{m=N-2}^{0} \frac{1}{\varepsilon^2[N-m-1]} \sum_{i=1}^{2} \left| \hat{g}_i[m] - \left(\boldsymbol{\varphi}_g^{(b)}[N-m-2]\mathbf{\Phi}_{gg}^{-1}[N-m-2] \right) \hat{\mathbf{G}}_i^{(b)}[m+1] \right|^2 ,\end{aligned} \tag{119}$$

where

$$m[N-1] = \frac{1}{\varepsilon^2[0]} \sum_{i=1}^{2} \left| \hat{g}_i[N-1] \right|^2 \tag{120}$$

is the branch metric of the reference fading-plus-noise sample for backward prediction. Since this term is independent of the hypothesized pattern, it can be omitted in the decoding process.

Now let us define the branch metric as

$$\begin{aligned} m^{(b)}[m] &= \frac{1}{\varepsilon^2[N-m-1]} \sum_{i=1}^{2} \left| \hat{g}_i[m] - \left(\boldsymbol{\varphi}_g^{(b)}[N-m-2]\mathbf{\Phi}_{gg}^{-1}[N-m-2] \right) \hat{\mathbf{G}}_i^{(b)}[m+1] \right|^2 \\ &= \frac{1}{\varepsilon^2[N-m-1]} \sum_{i=1}^{2} \left| \hat{g}_i[m] - \tilde{g}_i^{(b)}[m] \right|^2 \\ &= \frac{1}{\varepsilon^2[N-m-1]} \left\{ \begin{aligned} &\left|\hat{g}_1[m]\right|^2 + \left|\hat{g}_2[m]\right|^2 + \left|\tilde{g}_1^{(b)}[m]\right|^2 + \left|\tilde{g}_2^{(b)}[m]\right|^2 \\ &\qquad - 2\,\mathrm{Re}\left\{ \tilde{g}_1^{(b)}[m]\hat{g}_1^*[m] \right\} - 2\,\mathrm{Re}\left\{ \tilde{g}_2^{(b)}[m]\hat{g}_2^*[m] \right\} \end{aligned} \right\}, \end{aligned} \tag{121}$$

where

$$\tilde{g}_i^{(b)}[m] = \left(\boldsymbol{\varphi}_g^{(b)}[N-m-2]\mathbf{\Phi}_{gg}^{-1}[N-m-2] \right) \hat{\mathbf{G}}_i^{(b)}[m+1] \tag{122}$$

is the predicted gain based on path history up to time $m+1$. Similar to the forward prediction case, the first four terms of (121) are independent of the branch hypothesis in the current interval so the backward branch hypothesis can be sorted by the last two terms. The sum of the last two terms is represented as

$$
\begin{aligned}
q^{(b)}[m] &= 2\operatorname{Re}\left\{\tilde{g}_1^{(b)}[m]\hat{g}_1^*[m]\right\} + 2\operatorname{Re}\left\{\tilde{g}_2^{(b)}[m]\hat{g}_2^*[m]\right\} \\
&= 2\operatorname{Re}\left\{\left(\tilde{g}_1^{(b)}[m]\right)^* \hat{g}_1[m]\right\} + 2\operatorname{Re}\left\{\left(\tilde{g}_2^{(b)}[m]\right)^* \hat{g}_2[m]\right\} \\
&= 2\operatorname{Re}\left\{\left(\tilde{\mathbf{g}}^{(b)}[m]\right)^H \hat{\mathbf{g}}[m]\right\},
\end{aligned}
\tag{123}
$$

where

$$
\tilde{\mathbf{g}}^{(b)}[m] = \begin{bmatrix} \tilde{g}_1^{(b)}[m] \\ \tilde{g}_2^{(b)}[m] \end{bmatrix}
\tag{124}
$$

is the vector of backward predicted gains at time *m*. Since $\hat{\mathbf{s}}[m] = \hat{\mathbf{c}}^H[m+1]\hat{\mathbf{s}}[m+1]$ and $\hat{\mathbf{g}}[m] = \hat{\mathbf{s}}^H[m]\mathbf{r}[m] = \hat{\mathbf{s}}^H[m+1]\hat{\mathbf{c}}[m+1]\mathbf{r}[m]$, (123) can be rewritten as

$$
\begin{aligned}
q^{(b)}[m] &= 2\operatorname{Re}\left\{\left(\hat{\mathbf{s}}[m+1]\tilde{\mathbf{g}}^{(b)}[m]\right)^H \hat{\mathbf{c}}[m+1]\mathbf{r}[m]\right\} \\
&= \operatorname{Re}\left\{\left(\mathbf{a}^{(b)}[m]\right)^H \hat{\mathbf{d}}[m+1]\mathbf{y}[m]\right\}
\end{aligned}
\tag{125}
$$

where

$$
\mathbf{y}[m] = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}\mathbf{r}[m] = \begin{bmatrix} y_1[m] \\ y_2[m] \end{bmatrix}
\tag{126}
$$

is a rotated ST received vector, and

$$
\mathbf{a}^{(b)}[m] = \hat{\mathbf{s}}[m+1]\tilde{\mathbf{g}}^{(b)}[m] = \begin{bmatrix} a_1^{(b)}[m] \\ a_2^{(b)}[m] \end{bmatrix}.
\tag{127}
$$

If we further simplify $q^{(b)}[m]$, (125) can be rewritten as

$$\begin{aligned}
q^{(b)}[m] &= \mathrm{Re}\left\{\begin{bmatrix}\left(a_1^{(b)}[m]\right)^* & \left(a_2^{(b)}[m]\right)^*\end{bmatrix}\begin{bmatrix}\hat{d}_1[m+1] & \hat{d}_2[m+1] \\ -\hat{d}_2^*[m+1] & \hat{d}_1^*[m+1]\end{bmatrix}\begin{bmatrix}y_1[m] \\ y_2[m]\end{bmatrix}\right\} \\
&= \mathrm{Re}\left\{\begin{aligned} & y_1[m]\hat{d}_1[m+1]\left(a_1^{(b)}[m]\right)^* - y_1[m]\hat{d}_2^*[m+1]\left(a_2^{(b)}[m]\right)^* \\ & \qquad + y_2[m]\hat{d}_2[m+1]\left(a_1^{(b)}[m]\right)^* + y_2[m]\hat{d}_1^*[m+1]\left(a_2^{(b)}[m]\right)^* \end{aligned}\right\} \\
&= \mathrm{Re}\left\{\begin{aligned} & y_1[m]\hat{d}_1[m+1]\left(a_1^{(b)}[m]\right)^* - y_1^*[m]\hat{d}_2[m+1]a_2^{(b)}[m] \\ & \qquad + y_2[m]\hat{d}_2[m+1]\left(a_1^{(b)}[m]\right)^* + y_2^*[m]\hat{d}_1[m+1]a_2^{(b)}[m] \end{aligned}\right\} \\
&= \mathrm{Re}\left\{\begin{aligned} & \left(y_2^*[m]a_2^{(b)}[m] + y_1[m]\left(a_1^{(b)}[m]\right)^* \right)\hat{d}_1[m+1] \\ & \qquad + \left(y_2[m]\left(a_1^{(b)}[m]\right)^* - y_1^*[m]a_2^{(b)}[m] \right)\hat{d}_2[m+1] \end{aligned}\right\} \\
&= \mathrm{Re}\left\{\hat{d}_1[m+1]\left(\tilde{d}_1^{(b)}[m]\right)^*\right\} + \mathrm{Re}\left\{\hat{d}_2[m+1]\left(\tilde{d}_2^{(b)}[m]\right)^*\right\} \\
&= q_1^{(b)}[m] + q_2^{(b)}[m], \qquad (128)
\end{aligned}$$

where

$$\begin{aligned}
\left(\tilde{d}_1^{(b)}[m]\right)^* &= y_2^*[m]a_2^{(b)}[m] + y_1[m]\left(a_1^{(b)}[m]\right)^* \\
\left(\tilde{d}_2^{(b)}[m]\right)^* &= y_2[m]\left(a_1^{(b)}[m]\right)^* - y_1^*[m]a_2^{(b)}[m]
\end{aligned} \qquad (129)$$

and

$$\begin{aligned}
q_1^{(b)}[m] &= \mathrm{Re}\left\{\hat{d}_1[m+1]\left(\tilde{d}_1^{(b)}[m]\right)^*\right\} \\
q_2^{(b)}[m] &= \mathrm{Re}\left\{\hat{d}_2[m+1]\left(\tilde{d}_2^{(b)}[m]\right)^*\right\}.
\end{aligned} \qquad (130)$$

For the backward Fano detection, the sequence of the paths to be searched is also determined by sorting the sums in (128). Thus, the decoder can use the same sorting algorithm as described in Section 4.2 to sort the M^2 branches from largest to smallest at every discrete time *m*.

REFERENCE LIST

[1] J.K. Cavers, "An Analysis of Pilot Symbol Assisted Modulation for Rayleigh Fading Channels," *IEEE Transactions on Vehicular Technology*, vol. 40, issue 4, pp. 686-693, Nov 1991.

[2] V. Tarokh, N. Seshadri and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criterion and code construction," *IEEE Transactions on Information Theory*, vol. 44, pp. 744–765, Mar 1998.

[3] S. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 1451–1458, Oct 1998.

[4] V. Tarokh, H. Jafarkhani and A. R. Calderbank, "Space-time block codes from orthogonal designs" *IEEE Transactions on Information Theory*, vol. 45, pp. 1456–1467, Jul 1999.

[5] D. Divsalar and M.K. Simon, "Multiple-Symbol Differential Detection of MPSK," *IEEE Transactions on Communications*, vol. 38, issue 3, pp. 300-308, Mar 1990.

[6] P. Ho and D. Fung, "Error Performance of Multiple-Symbol Differential Detection of PSK Signals Transmitted Over Correlated Rayleigh Fading Channels," *IEEE Transactions on Communications*, vol. 40, issue 10, pp. 1566-1569, Oct 1992.

[7] K.M. Mackenthun, "A Fast Algorithm for Multiple-Symbol Differential Detection of MPSK," *IEEE Transactions on Communications*, vol. 42, issue 2/3/4, pp. 1471-1474, Feb/Mar/Apr 1994.

[8] P.Y. Kam and C.H. Teh, "Reception of PSK Signals over Fading Channels via Quadrature Amplitude Estimation", *IEEE Transactions on Communications*, vol. 31, issue 8, pp. 1024-1027, Aug 1983.

[9] R. Schober and W.H. Gerstacker, "Decision-Feedback Differential Detection Based on Linear Prediction for MDPSK Signals Transmitted over Ricean Fading Channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, issue 3, pp. 391-402, Mar 2000.

[10] L. Lampe, R. Schober, V. Pauli and C. Windpassinger, "Multiple-Symbol Differential Sphere Decoding," in *Proceedings, IEEE International Conference on Communications, Paris*, vol. 2, pp. 787-791, 20-24 Jun 2004.

[11] R.T. Derryberry, "Transmit Diversity in 3G CDMA Systems," *IEEE Communications Magazine*, vol. 40, issue 4, pp. 68-75, Apr 2002.

[12] V. Tarokh and H. Jafarkhani, "A differential detection scheme for transmit diversity," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 1169–1174, Jul 2000.

[13] B. Hughes, “Differential space-time modulation,” *IEEE Transactions on Information Theory*, vol. 46, pp. 2567–2578, Nov 2000.

[14] B. Hochwald and W. Sweldens, “Differential unitary space time modulation,” *IEEE Transactions on Communications*, vol. 48, pp. 2041–2052, Dec 2000.

[15] H. Jafarkhani and V. Tarokh, “Multiple transmit antenna differential detection from generalized orthogonal designs,” *IEEE Transactions on Information Theory*, vol. 47, pp. 2626–2631, Sept 2001.

[16] C. Gao, A.M. Haimovich, and D. Lao, “Multiple-symbol differential detection for space-time block codes,” in *Proceedings, the 36th Annual Conference on Information Science and Systems*, Princeton, NJ, Mar 2002.

[17] M. Riediger and P. Ho, "Non-Coherent Detection of Alamouti-Type Space-Time Codes for PSK Signal Sets", in *Proceedings, IEEE Global Telecommunications Conference*, Dallas, Texas, vol. 5, pp. 2956-2961, Nov 29 - Dec 3, 2004.

[18] Y. Li, C.N. Georghiades and G. Huang, “Iterative maximum-likelihood sequence estimation for space-time coded systems,” *IEEE Transactions on Communications*, vol. 49, pp. 948–951, Jun 2001.

[19] R. Schober and L. H.-J. Lampe, “Noncoherent receivers for differential space-time modulation,” *IEEE Transactions on Communications*, vol. 50, pp. 768–777, May 2002.

[20] C. Ling, H.K. Li and A.C. Kot, “Noncoherent sequence detection of differential space-time modulation,” *IEEE Transactions on Information Theory*, vol. 49, pp. 2727–2734, Oct 2003.

[21] P. Tarasak and V.K. Bhargava, “Improved Approximate Maximum-Likelihood Receiver for Differential Space –Time Block Codes Over Rayleigh-Fading Channels”, *IEEE Transactions on Vehicular Technology*, vol. 53, pp. 461-468, Mar 2004.

[22] J.K. Cavers, “On the Validity of the Slow and Moderate Fading Models for Matched Filter Detection of Rayleigh Fading Signals,” *Canadian Journal of Electrical and Computer Engineering*, vol. 17, issue 4, pp. 183-189, Oct 1992.

[23] E. Chiavaccini and G.M. Vitetta, “Further results on Tarokh’s space-time differential technique,“ in *Proceedings, IEEE International Conference on Communications*, vol. 3, pp. 1778-1782, Apr 2002.

[24] J.G. Proakis, Digital Communications. *McGraw-Hill*, 4th edition, pp. 48-50, 2001.

[25] J.H. Lodge and M.L. Moher, “Maximum Likelihood Sequence Estimation of CPM Signals Transmitted Over Rayleigh Flat-Fading Channels,” *IEEE Transactions on Communications*, vol. 38, issue 6, pp. 787-794, Jun 1990.

[26] D.G. Manolakis, V.K. Ingle and S.M. Kogon, Statistical and Adaptive Signal Processing, Spectral Estimation, Signal Modeling, Adaptive Filtering and Array Processing. *McGraw-Hill*, Appendix A, pp. 745-746, 2000.

[27] M. Pohst, "On the Computation of Lattice Vectors of Minimal Length, Successive Minima and Reduced Bases with Applications," *ACM SIGSAM Bulletin*, vol. 15, pp. 37-44, Feb 1981.

[28] O. Damen, A. Chkeif, and J.-C. Belfiore, "Lattice Code Decoder for Space-Time Codes," *IEEE Communications Letters*, vol. 4, issue 5, pp. 161-163, May 2000.

[29] H. Vikalo and B. Hassibi, "Modified Fincke-Pohst Algorithm for Low-Complexity Iterative Decoding over Multiple Antenna Channels," in *Proceedings*, *IEEE International Symposium on Information Theory*, pp. 390, Jul 2002.

[30] R.M. Fano, "A Heuristic Discussion of Probabilistic Decoding," *IEEE Transactions on Information Theory*, vol. 9, issue 2, pp. 64-74, Apr 1963.

[31] J.L. Massey, "Variable-Length Codes and the Fano Metric," *IEEE Transactions on Information Theory*, vol. 18, issue 1, pp. 196-198, Jan 1972.

[32] J.M. Wozencraft and I.M. Jacobs, Principles of Communication Engineering. *John Wiley & Sons*, pp. 431, 1965.

[33] V. Sorokine and F.R. Kschischang, "A Sequential Decoder for Linear Block Codes with a Variable Bias-Term Metric," *IEEE Transactions on Information Theory*, vol. 44, issue 1, pp. 410-416, Jan 1998.

[34] J. Chen, R. Sun and X. Wang, "Optimal Threshold Sequential Decoding Algorithms for Linear Block Codes," in *Proceedings, Vehicular Technology Conference*, *Tokyo*, vol. 1, pp. 605-607, 15-18 May 2000.

[35] E.M. Long and A.M. Bush, "Decision-Aided Sequential Intersymbol Interference Sequence Estimation for Channels," in *Proceedings*, *IEEE International Conference on Communications*, *Boston*, vol. 2, pp. 841-845, 11-14 Jun 1989.

[36] E. Zervas and J. Proakis, "A Sequential Algorithm for Blind Equalization," in *Proceedings*, *Military Communications Conference*, vol. 1, pp. 231-235, 11-14 Oct 1992.