Reduced complexity decoding for multiple antenna wireless communications

Philippe Bergeron-Burns B.A.Sc.

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

April 2005

A thesis submitted to McGill University in partial fulfilment of the requirements of the degree of Master of Engineering (M.Eng.) in Electrical Engineering.

2005/04/22

Abstract

One way to achieve high data rate and bandwidth efficient wireless communications is to employ multiple transmit and receive antennas creating a Multiple-Input Multiple-Output (MIMO) system, combined with Linear Dispersion (LD) codes. Sphere Decoder (SD) is a low complexity Maximum Likelihood (ML) method of decoding LD codes.

Using Subspace Matched Filtering principles, two complexity reducing front-ends to any variant of the SD, are developed. These two-stage decoders are designed to only have marginally worse performance than SD.

Computer simulations confirm the lower complexity and close to ML performance of the two-stage decoders. For a two transmit and two receive antenna LD coded MIMO system at an SNR of 22dB, the two-stage decoder reduces the average size of the SD search tree by a factor of 5. The BER performance of the two-stage decoder is within 0.25dB of the ML performance.

Sommaire

Dans les systèmes de communications sans fil, une façon d'atteindre des débits de transmission élevés et une haute efficacité spectrale consiste à utiliser des antennes multiples du côté de l'émetteur et du récepteur (créant un canal Multi-Entrées Multi-Sorties (MEMS)) ainsi que des codes à dispersion linéaire (DL). Le Décodage Sphérique (DS) est une méthode à basse complexité pour décoder les codes DS avec une performance semblable à celle du maximum de vraisemblance (MV).

Utilisant des principes du Filtrage Adapté Sous-espace et pour toute variante du DS, un module précédant le DS est développé pour deux systèmes différents dans le but de réduire la complexité. Cependant, ces méthodes de décodage introduisent une faible pénalité sur la performance.

Des simulations numériques confirment que les codes proposés réduisent la complexité et ont une performance proche de celle du MV. Pour un système DL MEMS avec deux antennes de transmission et deux antennes de réception, à 22dB, le décodage proposé réduit la complexité moyenne de DS par un facteur de 5. En termes de taux d'erreur binaire la performance du décodage proposé est à 0.25dB de la performance du MV.

Acknowledgments

I am very thankful to the Natural Sciences and Engineering Research Council of Canada and Professor Harry Leib for their financial support. I would also like to thank Professor Harry Leib for his very helpful feedback. Lastly, I would like to thank my family and friends, for your support and encouragement.

List of Acronyms

AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
GLR	Generalized Likelihood Ratio
HT	Householder Transform
KZ	Korkine-Zolotareff
LD	Linear Dispersion
LLL	Lenstra, Lenstra, and Lovasz
MIMO	Multiple-Input Multiple-Output
ML	Maximum Likelihood
MLE	Maximum Likelihood Estimate
MMSE	Minimum Mean Square Estimate
O-STBC	Orthogonal Space-Time Block Code
PAM	Pulse Amplitude Modulation
PBC-ML	Projection Based Conditional Maximum Likelihood
PR-ML	PRojection based Maximum Likelihood
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase Shift Keying
SD	Sphere Decoder
SE	Schnorr-Euchner
SE/SD	Schnorr-Euchner variant of the SD
SER	Symbol Error Rate
SFC	Statistical Full Coverage
SMF	Subspace Matched Filtering
SNR	Signal to Noise Ratio
SPC	Statistical Partial Coverage
STBC	Space-Time Block Code
V-BLAST	Vertical Bell labs LAyered Space-Time
ZF	Zero-Forcing

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Future wireless communication systems will require much higher throughputs in order to deliver high voice quality, reliable video, and advanced mobile applications [1,2]. All these services compete for a limited spectrum which, in many cases, must be licensed from the government. The main obstacle to reliable wireless communications is the multipath time-varying channel that causes fading [3,4,5].

Diversity, a technique to combat fading, can be realized with no bandwidth expansion by increasing the number of receive antennas or, more recently developed, by increasing the number of transmit antennas [6]. These techniques are referred to as receive diversity and transmit diversity. Increasing the number of transmit antennas is more practical in the downlink of many wireless communication systems since it only increases hardware complexity at the base station rather than at each mobile unit. One way of achieving diversity using multiple transmit antennas involves sending the same symbol from each sufficiently spaced transmit antenna so that the channel experienced by each transmitted symbol undergoes independent fading. Interestingly, the random scatterer filled environment can be used to create multiple effective channels which can, besides being used for diversity, be exploited to simultaneously transmit

different symbols in order to dramatically increase throughput without increasing system bandwidth [7]. At the same time multiple receive antennas can still be used for diversity. For this reason Multiple-Input Multiple-Output (MIMO) antenna systems have a much greater capacity than single antenna systems [8].

When the transmit diversity order is equal to the number of transmit antennas the system is said to have achieved full transmit diversity. Likewise, when the number of transmit antennas is equal to the transmission rate, in symbols per channel use, the system is said to achieve full spatial multiplexing. The Alamouti scheme for two transmit antennas [6] and Tarokh's Orthogonal Space-Time Block Codes (O-STBCs) [9] for more than two transmit antennas are techniques for achieving full transmit diversity that also allow for low complexity symbol-by-symbol Maximum-Likelihood (ML) detection. The O-STBCs and the Alamouti scheme can only achieve at most a rate of one symbol per channel use. Another technique, V-BLAST [7] (uncoded transmission), does not use the transmit antennas for diversity but can achieve full spatial multiplexing, a throughput of N_{TX} (the number of transmit antennas) symbols per channel-use. A limitation, imposed by the V-BLAST detection scheme, is that the number of transmit antennas must be less than the number of receive antennas [10]. Another limitation of V-BLAST transmission is its lack of any spatial or temporal coding and their resulting error resilience [10].

A class of codes that subsumes both O-STBCs as well as V-BLAST, are known as Linear Dispersion (LD) codes [10], are designed by optimizing the mutual information between the transmitted and received signals while maintaining a linear structure in the transmitted symbols. LD codes achieve diversity from coding as well as spectral efficiency from spatial multiplexing. Linearity aids in decoding and allows the transmission rate to increase at the expense of losing orthogonality between transmit antennas. The LD codes are shown to outperform both O-STBCs and V-BLAST at a

given transmission rate for a wide range of SNRs, number of receive antennas (N_{RX}), and number of transmit antennas (N_{TX}). However, the LD codes are neither guaranteed to achieve full transmit diversity nor full spatial multiplexing but normally achieve some intermediate level of both. Performance is boosted by spatial multiplexing since, at a fixed rate, this allows for the use of a smaller symbol constellation. LD codes can be designed for any N_{RX} and N_{TX} and therefore can be used in the downlink where N_{TX} is normally greater than N_{RX}.

Of major concern for all of these techniques is ML detection computational complexity. O-STBCs, due to their forced orthogonal structure, allow for linear complexity ML detection. In contrast, ML detection of V-BLAST and LD codes has exponential computational complexity in both N_{TX} and the size of the symbol constellation. There are other suboptimal lower complexity detection methods for V-BLAST and LD codes such as zero-forcing (ZF) detection, minimum mean squared error (MMSE) detection, and decision feedback detection. These methods, however, result in significant performance degradation compared to ML detection.

Recently, it has been shown that ML decoding with a lower than exponential complexity is possible for certain linear, non-orthogonal, codes using the sphere decoder (SD) [10,11]. Rather than considering all possible symbol vectors, as is done for the exhaustive search ML detector, the SD searches for symbol vectors within a spheroid centered by an estimate of the transmitted symbol vector. SD achieves ML performance detecting either an uncoded symbol vector or LD codewords since both are linear in terms of the transmitted symbols [10]. The computational complexity of SD is dependant on the realization of the channel matrix with the expected value of its complexity being a third degree polynomial in the number of transmit antennas [12].

Computational complexity gains over the original SD can be achieved by beginning the search for symbol vectors in the center of the spheroid rather than at the edge

[13,14], a technique known as the Schnorr-Euchner (SE) algorithm [15]. Another modification, Statistical Pruning [16], also offers a reduction in computational complexity at only a small compromise in performance. Statistical Pruning and the SE algorithm both extend the number of dimensions that the SD can handle before it reverts to exponential complexity.

Like LD codes, layered space-time codes [17] (also called D-BLAST) can achieve both diversity and spectral efficiency. A benefit to using a multi-antenna D-BLAST code is its simplicity since it is made up of many of single-antenna code blocks. D-BLAST decoding, however, always contains nulling and canceling and so ML performance cannot be achieved.

Another technique, lattice basis reduction, was first proposed in the context of detection for MIMO systems as an optional front-end to the SD to improve its speed and numerical stability [13]. Various lattice basis reduction algorithms exist each with its own criteria for finding the equivalent basis whose elements are reasonably short and orthogonal. One such algorithm, known as Korkine-Zolotareff (KZ) reduction [13], has fairly high complexity but good performance and would therefore be appropriate for detection in a slow fading environment where the channel matrix is constant over many channel-uses and lattice reduction is done infrequently. Otherwise, in situations where KZ reduction would be too slow, LLL (Lenstra, Lenstra, and Lovasz) reduction [18], offering lower performance but at a much reduced complexity, is recommended [13].

Lattice basis reduction was also considered as a front-end to both linear and non-linear suboptimal detectors [19]. Using a modified form of LLL reduction for two dimensions it was shown that LLL lattice-basis-reduced suboptimal detection has better performance with minor extra complexity than lattice-basis-unreduced suboptimal detection. This work was then extended to more than two dimensions using the LLL

algorithm [20]. A drawback of the lattice-basis-reduction front-end is that it distorts the symbol constellation. This drawback, besides adding complexity to the detector, also results in decreased performance that becomes less as the size of the QAM constellation increases since the symbol constellation begins to approximate an infinite lattice [13,19]. It is suggested in [12] that the drawbacks of lattice basis reduction make it an unpromising approach to reduced complexity decoding.

The contributions of this thesis are:

- Using Subspace Matched Filtering (SMF) principles [21,22], two complexity reducing front-ends to any variant of the SD, were invented. Either of the resulting two-stage decoders can achieve lower complexity decoding of either LD codes or uncoded (V-BLAST) transmission. The performance of these two-stage decoders, unlike those based on estimation and cancelation techniques, can be made arbitrarily close to ML.

- Developed computer simulations to show the tradeoff between performance and complexity for these two two-stage decoders. These simulations confirm the complexity advantage and close to ML performance of both two-stage decoders.

This thesis is organized as follows. Chapter 2 presents the system assumptions and both the uncoded and coded frameworks. Then, certain pertinent mathematical results and existing decoding methods are explained. In chapter 3 the Statistical Partial Coverage (SPC) front-end to the SD is described, analyzed, and simulated for both frameworks. Then, in chapter 4, another low complexity front-end, the Statistical Full Coverage (SFC) method, is described. Finally, in chapter 5, a computer simulation comparison of the detection schemes and their combinations is presented. Chapter 6 provides the conclusions and overall observations. Appendix A is an overview and user guide to the source code used to generate the computer simulations. An attached CD

contains all the C++ source code.

Chapter 2

System, Mathematical Preliminaries, and Detection Methods

2.1 System Assumptions

We consider a multiple-antenna wireless communication system with N_{TX} transmit antennas and N_{RX} receive antennas. For simplicity, several assumptions are made to focus attention on the proposed decoding schemes rather than on the model.

We assume that channel training allows the receiver but not the transmitter to know all of the effective channels between each transmit and receive antenna. Moreover, we assume sufficient antenna spacing, narrowband transmission, and flat fading in both frequency and time so that the channel matrix, $\tilde{\mathbf{H}} \in \mathcal{C}^{N_{RX} \times N_{TX}}$, is comprised of independent circular complex Gaussian 0-mean variance-1 random variables. We also assume that the transmitted vector, $\tilde{\mathbf{u}} \in \mathcal{C}^{N_{TX}}$, is a vector of squared M-QAM complex symbols. The symbols have average power one and M is the number con-

stellation points. The information bits in each transmitted symbol follow a Gray code assignment. The noise vector, $\tilde{\mathbf{n}} \in \mathcal{C}^{N_{RX}}$, is assumed to be made up of independent circular complex additive white gaussian noise (AWGN) of 0-mean and variance-1 random variables.

We let ρ be the average SNR at each receive antenna. Each transmitted symbol is multiplied by $\sqrt{\frac{\rho}{N_{TX}}}$ since the signal power at each receive antenna is the sum of the transmit power from all N_{TX} transmit antennas.

2.2 System Framework

Since both the SD and its lower-complexity detection variant, the SE variant of the SD (SE/SD) [13,14], as well as an important aspect of the two proposed front-ends, the Subspace Matched Filter (SMF) [21,22], are presented in the literature using a real system model we transform the complex system model into a real system model. Furthermore, the linearity of the LD codes in terms of the transmitted real symbols (taking the real and imaginary components of each complex symbol), which led to LD codes being considered in the literature using a real system model [10], is another motivation for the adoption the real system model. In fact, nearly all publications working in this area adopt a real system model. Our assumption of a squared M-QAM complex constellation enables each real transmitted symbol to be split into two $\sqrt{M}$-PAM symbols with an identical constellation. Each real symbol is scaled so as to have an average power of $1/2$. M-PSK symbols can also be split into two real symbols but are not considered in this work. We choose to focus instead on highly bandwidth efficient systems requiring large M where M-PSK is less appropriate than M-QAM. The valid symbols for four simple PAM constellations are presented in Table 2.1. The next two subsections show how both an uncoded real and LD coded real system model

can be obtained from the underlying complex model.

Constellation	b/s/Hz	distance (d)	Valid real symbols of average power 1/2
PAM-2	1	$\frac{2}{\sqrt{2}} = 1.414$	$-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}$
PAM-4	2	$\frac{2}{\sqrt{10}} = 0.632$	$-\frac{3}{\sqrt{10}}, -\frac{1}{\sqrt{10}}, \frac{1}{\sqrt{10}}, \frac{3}{\sqrt{10}}$
PAM-8	3	$\frac{2}{\sqrt{42}} = 0.309$	$-\frac{7}{\sqrt{42}}, -\frac{5}{\sqrt{42}}, -\frac{3}{\sqrt{42}}, -\frac{1}{\sqrt{42}}, \frac{1}{\sqrt{42}}, \frac{3}{\sqrt{42}}$ $\frac{5}{\sqrt{42}}, \frac{7}{\sqrt{42}}$
PAM-16	4	$\frac{2}{\sqrt{170}} = 0.153$	$-\frac{15}{\sqrt{170}}, -\frac{13}{\sqrt{170}}, -\frac{11}{\sqrt{170}}, -\frac{9}{\sqrt{170}}, -\frac{7}{\sqrt{170}},$ $-\frac{5}{\sqrt{170}}, -\frac{3}{\sqrt{170}}, -\frac{1}{\sqrt{170}}, \frac{1}{\sqrt{170}}, \frac{3}{\sqrt{170}}, \frac{5}{\sqrt{170}},$ $\frac{7}{\sqrt{170}}, \frac{9}{\sqrt{170}}, \frac{11}{\sqrt{170}}, \frac{13}{\sqrt{170}}, \frac{15}{\sqrt{170}}$

Table 2.1 Four simple PAM constellations with $E[|u_i|^2] = 1/2$.

2.2.1 Uncoded Framework

The uncoded framework is the framework for V-BLAST transmission. It is referred to as uncoded transmission since there is no coding method introducing either spatial or temporal diversity.

The baseband equivalent model of the received uncoded complex signal vector, $\tilde{\mathbf{r}} \in \mathcal{C}^{N_{RX}}$, at the sampling instant, which is assumed to be synchronous, can be represented as [10]:

$$\tilde{\mathbf{r}} = \sqrt{\frac{\rho}{N_{TX}}}\tilde{\mathbf{H}}\tilde{\mathbf{u}} + \tilde{\mathbf{n}}. \tag{2.1}$$

The decoder processes $\tilde{\mathbf{r}}$ to determine $\tilde{\mathbf{u}}$, the transmitted vector of squared M-QAM symbols. A block diagram of the uncoded system is shown below.

The transmission rate, R_u, of an uncoded system depends on the number of symbols sent in one channel-use, N_{TX}, and the size of the constellation, M. The uncoded rate in bits/channel-use is [10]:

$$R_u = N_{TX} \log_2 M. \tag{2.2}$$

For the reasons listed above we choose to express the complex system model as the

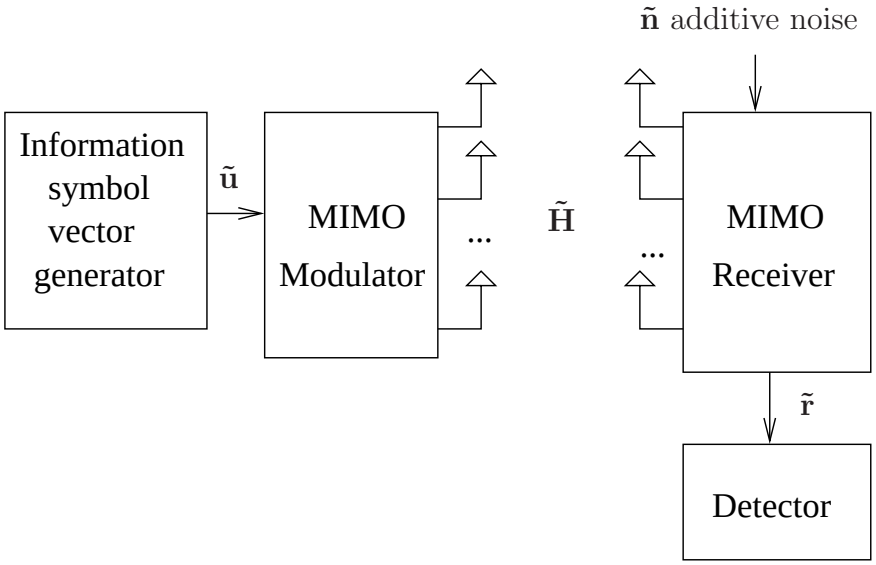

Fig. 2.1 Uncoded system block diagram

real system model:

$$\mathbf{r} = \sqrt{\frac{\rho}{N_{TX}}}\mathbf{H}\mathbf{u} + \mathbf{n}. \tag{2.3}$$

Where $\mathbf{r} = \begin{bmatrix} Re\{\tilde{\mathbf{r}}\} \\ Im\{\tilde{\mathbf{r}}\} \end{bmatrix}$, $\mathbf{u} = \begin{bmatrix} Re\{\tilde{\mathbf{u}}\} \\ Im\{\tilde{\mathbf{u}}\} \end{bmatrix}$, $\mathbf{n} = \begin{bmatrix} Re\{\tilde{\mathbf{n}}\} \\ Im\{\tilde{\mathbf{n}}\} \end{bmatrix}$ and

$$\mathbf{H} = \begin{bmatrix} Re\{\tilde{\mathbf{H}}\} & -Im\{\tilde{\mathbf{H}}\} \\ Im\{\tilde{\mathbf{H}}\} & Re\{\tilde{\mathbf{H}}\} \end{bmatrix} \tag{2.4}$$

When transmitting uncoded symbols it is assumed that $N_{TX} \leq N_{RX}$.

2.2.2 Linear Dispersion (LD) Coded Framework

LD codes are STBCs that improve on the performance of uncoded transmission by introducing diversity. Defining L as the block length of the code, LD codes require

that the channel be constant for L channel-uses.

The transmission rate, R_c, of a LD coded system depends on the number of M-QAM symbols sent, Q, L, and the size of the constellation, M. Any comparisons between coded and uncoded systems will be done at equal transmission rates. The coded rate in bits/channel-use is [10]:

$$R_c = \frac{Q}{L} \log_2 M. \tag{2.5}$$

If the channel is constant for at least L channel-uses then:

$$\tilde{\mathbf{r}}_t = \sqrt{\frac{\rho}{N_{TX}}} \tilde{\mathbf{H}} \tilde{\mathbf{s}}_\mathbf{t} + \tilde{\mathbf{n}}_\mathbf{t}, \quad t = 1, \ldots, L \tag{2.6}$$

where $\tilde{\mathbf{s}}_\mathbf{t} \in \mathcal{C}^{N_{TX} \times 1}$ represents the coded vector transmitted at time $t \in [1, L]$. A block diagram of the LD coded system can be seen below.

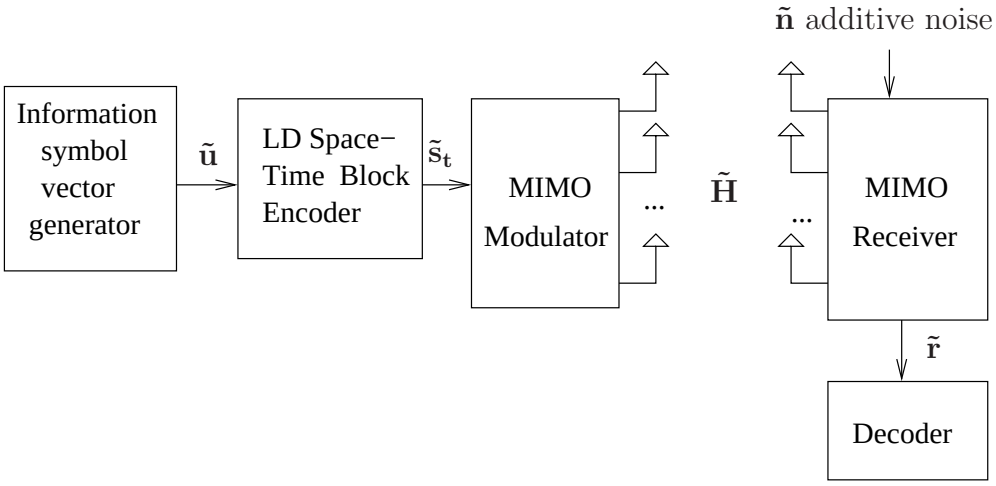

Fig. 2.2 LD coded system block diagram

We define the following three matrices:

$$\tilde{\mathbf{R}} = \begin{bmatrix} \tilde{\mathbf{r}}_1 & \tilde{\mathbf{r}}_2 & \dots & \tilde{\mathbf{r}}_L \end{bmatrix}^T, \tag{2.7}$$

$$\tilde{\mathbf{S}} = \begin{bmatrix} \tilde{\mathbf{s}}_1 & \tilde{\mathbf{s}}_2 & \dots & \tilde{\mathbf{s}}_L \end{bmatrix}^T, \tag{2.8}$$

$$\tilde{\mathbf{N}} = \begin{bmatrix} \tilde{\mathbf{n}}_1 & \tilde{\mathbf{n}}_2 & \dots & \tilde{\mathbf{n}}_L \end{bmatrix}^T. \tag{2.9}$$

We can then write the system equation in terms of these matrices:

$$\tilde{\mathbf{R}} = \sqrt{\tfrac{\rho}{N_{TX}}}\tilde{\mathbf{S}}\tilde{\mathbf{H}}^{\mathbf{T}} + \tilde{\mathbf{N}}. \tag{2.10}$$

LD codes are of the form:

$$\tilde{\mathbf{S}} = \sum_{q=1}^{Q} (\alpha_q \mathbf{A_q} + j\beta_q \mathbf{B_q}) \tag{2.11}$$

where $\tilde{\mathbf{S}} \in \mathcal{C}^{L \times N_{TX}}$ is the codeword matrix with α_q and β_q being real information scalars such that $\tilde{u}_q = \alpha_q + j\beta_q$, $q \in [1, Q]$. The two sets of matrices: $\mathbf{A_q}$, $\mathbf{B_q} \in \mathcal{C}^{L \times N_{TX}}$, $q \in [1, Q]$, completely describe the code. Equation (2.11) shows the linearity of the code matrix in the real transmitted symbols.

Expressing the LD coded system framework in terms of their real and imaginary components yields:

$$\begin{aligned} \tilde{\mathbf{R}}_{\mathbf{R}} + j\tilde{\mathbf{R}}_{\mathbf{I}} &= \sqrt{\frac{\rho}{N_{TX}}} \sum_{q=1}^{Q} [\alpha_q(\mathbf{A_{q,R}} + j\mathbf{A_{q,I}}) + j\beta_q(\mathbf{B_{q,R}} + j\mathbf{B_{q,I}})] \, (\tilde{\mathbf{H}}_{\mathbf{R}}^{\mathbf{T}} + j\tilde{\mathbf{H}}_{\mathbf{I}}^{\mathbf{T}}) \\ &\quad + \tilde{\mathbf{N}}_{\mathbf{R}} + j\tilde{\mathbf{N}}_{\mathbf{I}} \end{aligned} \tag{2.12}$$

with $\tilde{\mathbf{R}}_{\mathbf{R}} = Re(\tilde{\mathbf{R}})$, $\tilde{\mathbf{R}}_{\mathbf{I}} = Im(\tilde{\mathbf{R}})$, $\mathbf{A_{q,R}} = Re(\mathbf{A_q})$, $\mathbf{A_{q,I}} = Im(\mathbf{A_q})$, $\mathbf{B_{q,R}} = Re(\mathbf{B_q})$,

$\mathbf{B_{q,I}} = Im(\mathbf{B_q})$, $\tilde{\mathbf{H}}^{\mathbf{T}}_{\mathbf{R}} = Re(\tilde{\mathbf{H}}^{\mathbf{T}})$, $\tilde{\mathbf{H}}^{\mathbf{T}}_{\mathbf{I}} = Im(\tilde{\mathbf{H}}^{\mathbf{T}})$, $\tilde{\mathbf{N}}_{\mathbf{R}} = Re(\tilde{\mathbf{N}})$, $\tilde{\mathbf{N}}_{\mathbf{I}} = Im(\tilde{\mathbf{N}})$.

We define the following two matrices.

$$\mathcal{A}_{\mathbf{q}} = \begin{pmatrix} \mathbf{A_{q,R}} & -\mathbf{A_{q,I}} \\ \mathbf{A_{q,I}} & \mathbf{A_{q,R}} \end{pmatrix}, \quad \mathcal{B}_{\mathbf{q}} = \begin{pmatrix} -\mathbf{B_{q,I}} & -\mathbf{B_{q,R}} \\ \mathbf{B_{q,R}} & -\mathbf{B_{q,I}} \end{pmatrix}. \tag{2.13}$$

Likewise, we define the c^{th} column of $\tilde{\mathbf{R}}_{\mathbf{R}}$, $\tilde{\mathbf{R}}_{\mathbf{I}}$, $\tilde{\mathbf{H}}^{\mathbf{T}}_{\mathbf{R}}$, $\tilde{\mathbf{H}}^{\mathbf{T}}_{\mathbf{I}}$, $\tilde{\mathbf{N}}_{\mathbf{R}}$, and $\tilde{\mathbf{N}}_{\mathbf{I}}$, respectively, as $\mathbf{r_{c,R}}$, $\mathbf{r_{c,I}}$, $\mathbf{h_{c,R}}$, $\mathbf{h_{c,I}}$, $\mathbf{n_{c,R}}$, and $\mathbf{n_{c,I}}$. We define:

$$\mathbf{h_c} = \begin{bmatrix} \mathbf{h_{c,R}} \\ \mathbf{h_{c,I}} \end{bmatrix}. \tag{2.14}$$

Next, expressing the received matrix as a vector and expressing the real information scalars as a vector we can translate (2.10) to the following form [10]:

$$\begin{bmatrix} \mathbf{r_{1,R}} \\ \mathbf{r_{1,I}} \\ \vdots \\ \mathbf{r_{N_{RX},R}} \\ \mathbf{r_{N_{RX},I}} \end{bmatrix} = \sqrt{\frac{\rho}{N_{TX}}} \begin{bmatrix} \mathcal{A}_1\mathbf{h_1} & \mathcal{B}_1\mathbf{h_1} & \dots & \mathcal{B_Q}\mathbf{h_1} \\ \vdots & \vdots & \dots & \vdots \\ \mathcal{A_N}\mathbf{h_N} & \mathcal{B_N}\mathbf{h_N} & \dots & \mathcal{B_Q}\mathbf{h_N} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \beta_1 \\ \vdots \\ \alpha_Q \\ \beta_Q \end{bmatrix} + \begin{bmatrix} \mathbf{n_{1,R}} \\ \mathbf{n_{1,I}} \\ \vdots \\ \mathbf{n_{N_{RX},R}} \\ \mathbf{n_{N_{RX},I}} \end{bmatrix}$$

$$\mathbf{r} = \sqrt{\frac{\rho}{N_{TX}}}\mathbf{G}\mathbf{u} + \mathbf{n}. \tag{2.15}$$

Therefore $\mathbf{G} \in \mathcal{C}^{2N_{RX}L \times 2Q}$ for the LD coded system acts like $\mathbf{H}$ for the uncoded system. Under the assumption that the decoder has perfect knowledge of $\mathbf{H}$ and the LD code (and thus all the $\mathbf{A_q}$, $\mathbf{B_q} \in \mathcal{C}^{L \times N_{TX}}$, $q \in [1, Q]$ matrices) then it will also know $\mathbf{G}$. Furthermore, the LD coded form allows for any combination of N_{TX} and N_{RX}. It is only required that $Q \leq N_{RX}L$ so that the system is consistent.

2.3 Mathematical Preliminaries

2.3.1 Projections

The manipulation of projections and statistics of projections are concepts used to develop the two proposed SD front-ends. The channel matrix $\mathbf{H} = [\mathbf{h_1} \dots \mathbf{h_{N_{TX}}}]$ has a corresponding subspace $\langle \mathbf{H} \rangle$, called the signal subspace, which is the span of $\{\mathbf{h_i}\}_{i=1}^{N_{TX}}$ [22]. Letting the columns of $\mathbf{Q}$ be an orthonormal basis of the $\langle \mathbf{H} \rangle$ subspace then

$$\mathbf{P_H} = \mathbf{QQ^T} \tag{2.16}$$

is the unique orthogonal projection matrix onto the $\langle \mathbf{H} \rangle$ subspace [23]. Taking the QR factorization of $\mathbf{H} = \mathbf{QR}$ and substituting $\mathbf{Q} = \mathbf{HR}^{-1}$ into (2.16) we obtain:

$$\begin{aligned} \mathbf{P_H} &= \mathbf{HR}^{-1}(\mathbf{HR^{-1}})^T \\ &= \mathbf{HR}^{-1}(\mathbf{R}^T)^{-1}\mathbf{H}^T \\ &= \mathbf{H}(\mathbf{R^TR})^{-1}\mathbf{H}^T, \end{aligned} \tag{2.17}$$

$$\begin{aligned} \mathbf{H^TH} &= (\mathbf{QR})^T\mathbf{QR} \\ &= \mathbf{R^TQ^TQR} \\ &= \mathbf{R^TR}. \end{aligned} \tag{2.18}$$

Therefore, the unique orthogonal projection matrix onto the $\langle \mathbf{H} \rangle$ subspace can be found in terms of its non-orthonormal component vectors.

$$\mathbf{P_H} = \mathbf{H}(\mathbf{H^TH})^{-1}\mathbf{H}^T. \tag{2.19}$$

In order to project a point, such as the received vector, $\mathbf{r}$, onto the $\langle \mathbf{H} \rangle$ subspace it is pre-multiplied by the orthogonal projection matrix, $\mathbf{P_H r}$. The projection matrix for the subspace orthogonal to the $\langle \mathbf{H} \rangle$ subspace is [22]:

$$\mathbf{P_H^{\perp}} = \mathbf{I} - \mathbf{P_H}. \tag{2.20}$$

Other important projection properties which follow directly include [22]:

$$\mathbf{P_H} = \mathbf{P_H^T} = \mathbf{P_H^2}, \tag{2.21}$$

$$\mathbf{P_H P_H^{\perp}} = \mathbf{0}. \tag{2.22}$$

The channel matrix, $\mathbf{H}$, can be spliced into $\mathbf{S_i}$, defined as $\mathbf{H}$ without the i^{th} column, and $\mathbf{h_i}$, the i^{th} column of $\mathbf{H}$.

$$\mathbf{v_i} = \mathbf{P_{S_i}^{\perp} h_i}. \tag{2.23}$$

$\langle \mathbf{v_i} \rangle$ represents the subspace of $\mathbf{h_i}$ orthogonal to the $\mathbf{S_i}$ subspace. The $\mathbf{P_H}$ projection matrix can then be decomposed between these subspaces.

$$\mathbf{P_H} = \mathbf{P_{v_i}} + \mathbf{P_{S_i}}. \tag{2.24}$$

The $\mathbf{P_H}$ projection matrix can also be decomposed into the oblique projections matrices $\mathbf{E_{h_i S_i}}$ and $\mathbf{E_{S_i h_i}}$ that have range spaces $\langle \mathbf{h_i} \rangle$ and $\langle \mathbf{S_i} \rangle$ respectively and null spaces $\langle \mathbf{S_i} \rangle$ and $\langle \mathbf{h_i} \rangle$ respectively [21].

$$\mathbf{E_{S_i h_i}} = \mathbf{S_i}(\mathbf{S_i^T P_{h_i}^{\perp} S_i})^{-1} \mathbf{S_i^T P_{h_i}^{\perp}}. \tag{2.25}$$

$$\mathbf{E_{h_i S_i}} = \mathbf{h_i}(\mathbf{h_i^T P_{S_i}^{\perp} h_i})^{-1} \mathbf{h_i^T P_{S_i}^{\perp}}. \tag{2.26}$$

$$\mathbf{P_H} = \mathbf{E_{h_i S_i}} + \mathbf{E_{S_i h_i}}. \tag{2.27}$$

Figure 2.3 shows the relationship between two decompositions of the $\langle \mathbf{H} \rangle$ subspace. $\mathbf{y}$ is a vector that is being projected onto this subspace.

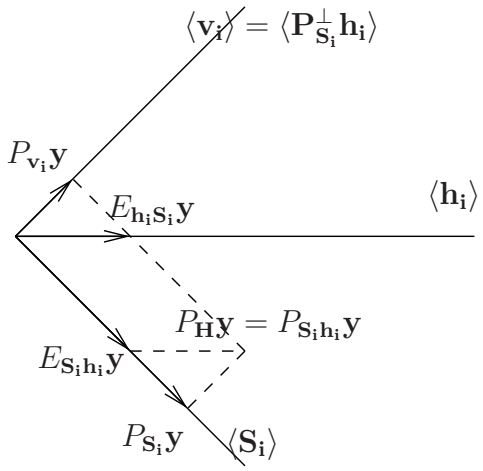

Fig. 2.3 Decompositions of the $\langle \mathbf{H} \rangle$ subspace

If $\mathbf{x}$ has a $\mathcal{N}[\mathbf{0}, \mathbf{I}]$ distribution and $\mathbf{P}$ is a rank r projection matrix then the distribution of $\mathbf{y} = \mathbf{Px}$ is $\mathcal{N}[\mathbf{0}, \mathbf{P}]$ [22]. This means that the quadratic form $\mathbf{y^T y} = \mathbf{x^T P x}$ has a central χ_r^2 distribution. If $\mathbf{x}$ has a $\mathcal{N}[\mathbf{m}, \mathbf{I}]$ distribution then the quadratic form, $\mathbf{y^T y}$, has a noncentral χ_r^2 distribution with noncentrality parameter $\lambda^2 = \mathbf{m^T m}$ [21].

2.3.2 Maximum-Likelihood Detection

The ML detected vector $\mathbf{u_{ML}}$ is the vector which maximizes the likelihood function $f_{\mathbf{r}}(\mathbf{r}|\mathbf{u_s})$ when $\mathbf{u_s} = \mathbf{u_{ML}}$ [24]. Here $\mathbf{u_s}$ can be any vector of symbols from a given constellation. The set of all symbol vectors is called $\mathcal{S}$.

$$\mathbf{u_{ML}} = \arg \max_{\mathbf{u_s} \in \mathcal{S}} f_{\mathbf{r}}(\mathbf{r}|\mathbf{u_s}). \tag{2.28}$$

When the channel is fixed, the randomness in the received vector, $\mathbf{r}$, comes from

the Gaussian noise vector so that the conditional probability density function (pdf), $f_{\mathbf{r}}(\mathbf{r}|\mathbf{u_s})$, has a jointly Gaussian distribution. Assuming the noise elements have variance $N_0/2$ this distribution is [24]:

$$f_{\mathbf{r}}(\mathbf{r}|\mathbf{u_s}) = \frac{1}{(\pi N_0)^{N_{RX}/2}} \exp\{-\frac{1}{N_0}\|\mathbf{r} - \sqrt{\frac{\rho}{N_{TX}}}\mathbf{H}\mathbf{u_s}\|_2^2\}. \tag{2.29}$$

To maximize $f_{\mathbf{r}}(\mathbf{r}|\mathbf{u_s})$ over the set of all possible symbol vectors, $\mathbf{u_s} \in \mathcal{S}$, it is necessary to minimize $\|\mathbf{r}-\sqrt{\frac{\rho}{N_{TX}}}\mathbf{H}\mathbf{u_s}\|_2^2$ which is equivalent to minimizing $\|\mathbf{r}-\sqrt{\frac{\rho}{N_{TX}}}\mathbf{H}\mathbf{u_s}\|_2$. Then, independent of the noise variance, the ML decision rule is to find the symbol vector $\mathbf{u_s}$ that is closest to the received vector in the euclidean sense.

$$\mathbf{u_{ML}} = \arg\min_{\mathbf{u_s}\in\mathcal{S}} \|\mathbf{r} - \sqrt{\frac{\rho}{N_{TX}}}\mathbf{H}\mathbf{u_s}\|_2. \tag{2.30}$$

The complexity of ML detection, using an exhaustive search over $\mathcal{S}$, is exponential both in the size of the signal constellation (M), and in the number columns of $\mathbf{H}$, the number of transmit antennas (N_{TX}). The exhaustive search complexity is $O(M^{N_{TX}})$.

2.3.3 Subspace Matched Filtering

SMF is a more general version of the well known Matched Filter. The SMF is able to deal with multirank (subspace) interference and multirank signals [21]. In this work we will only consider rank-1 signals but the SMF's ability to deal with multirank interference will be important.

Suppose we have an N-dimensional measurement vector, $\mathbf{y}$, which is composed as follows:

$$\mathbf{y} = \mathbf{x}\lambda + \mathbf{S}\phi + \mathbf{n}. \tag{2.31}$$

The rank-1 signal is $\mathbf{x}\lambda$ and the multirank interference is $\mathbf{S}\phi$. It is also assumed, in this detection problem, that $\mathbf{S}$ and $\mathbf{x}$ are known but that λ and ϕ are not known. The unknown ϕ is dealt with as an unknown parameter. Our detection goal on measuring $\mathbf{y}$ is to decide whether $\lambda = 0$, the null hypothesis (H_0), or $\lambda \neq 0$, the alternative hypothesis (H_1). The dimensions of the various quantities are:

$$\begin{aligned} \mathbf{y} &\in \mathcal{R}^{N\times 1} \qquad & \mathbf{x} &\in \mathcal{R}^{N\times 1} \\ \mathbf{S} &\in \mathcal{R}^{N\times p} \qquad & \phi &\in \mathcal{R}^{p\times 1}. \end{aligned}$$

Furthermore, it is assumed that $p < N$ and the noise vector, $\mathbf{n}$, is assumed to have a $\mathcal{N}[0, \sigma^2\mathbf{I}]$ distribution. The detection problem tests between the following two probability distributions of $\mathbf{y}$:

$$\begin{aligned} H_0 : \mathbf{y} : \mathcal{N}\left[\mathbf{S}\phi, \sigma^2\mathbf{I}\right] \qquad & vs \\ H_1 : \mathbf{y} : \mathcal{N}\left[\mathbf{x}\lambda + \mathbf{S}\phi, \sigma^2\mathbf{I}\right]. & \end{aligned} \tag{2.32}$$

The subspaces $\langle\mathbf{x}\rangle$ and $\langle\mathbf{S}\rangle$ are not assumed to be orthogonal but are assumed to be linearly independent. The pdf of $\mathbf{y}$ is:

$$f(\mathbf{y}; \lambda, \phi, \sigma^2) = (2\pi\sigma^2)^{-N/2} \exp\{-\tfrac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{x}\lambda - \mathbf{S}\phi\|_2^2\}. \tag{2.33}$$

The likelihood function of $\mathbf{y}$ is:

$$l(\lambda, \phi, \sigma^2; \mathbf{y}) = (2\pi\sigma^2)^{-N/2} \exp\{-\tfrac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{x}\lambda - \mathbf{S}\phi\|_2^2\}. \tag{2.34}$$

For the two sets of values $(\lambda_1, \phi_{\mathbf{1}}, \sigma^2)$ and $(\lambda_0, \phi_{\mathbf{0}}, \sigma^2)$, under hypothesis H_1 and H_0,

the likelihood ratio for this problem is:

$$\begin{aligned}\Lambda(\mathbf{y}) &= \frac{l(\lambda_1, \phi_{\mathbf{1}}, \sigma^2; \mathbf{y})}{l(\lambda_0, \phi_{\mathbf{0}}, \sigma^2; \mathbf{y})} \qquad (2.35)\\ &= \frac{(2\pi\sigma^2)^{-N/2} \exp\{-\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{x}\lambda_1 - \mathbf{S}\phi_{\mathbf{1}}\|_2^2\}}{(2\pi\sigma^2)^{-N/2} \exp\{-\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{S}\phi_{\mathbf{0}}\|_2^2\}} \\ &= \exp\{\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{S}\phi_{\mathbf{0}}\|_2^2 - \frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{x}\lambda_1 - \mathbf{S}\phi_{\mathbf{1}}\|_2^2\}. \qquad (2.36)\end{aligned}$$

SMF uses the likelihood ratio to decide between H_0 and H_1. A threshold, t, is chosen, and H_0 is selected if $\Lambda(\mathbf{y}) < t$ and H_1 is selected if $\Lambda(\mathbf{y}) > t$.

2.4 Zero-Forcing (ZF) Detection

The ZF detector is a low complexity linear detector with sub-ML performance. In ZF detection the received vector, $\mathbf{r}$, is first pre-multiplied by a ZF matrix, $\mathbf{F_{ZF}}$.

$$\mathbf{F_{ZF}} = (\mathbf{H^T H})^{-1}\mathbf{H^T}. \qquad (2.37)$$

The ZF matrix is a linear transformation from an N_{RX}-dimensional vector, $\mathbf{r}$, to an N_{TX}-dimensional vector, ϖ. The ZF matrix is designed to completely cancel the effect of the channel, $\mathbf{H}$, for a noiseless system.

$$\begin{aligned}\mathbf{F_{ZF}H} &= (\mathbf{H^T H})^{-1}\mathbf{H^T H} \\ &= \mathbf{I}. \qquad (2.38)\end{aligned}$$

Although the channel, $\mathbf{H}$, is random, each realization is assumed to be known by the detector. The received symbol SNR, $\sqrt{\frac{\rho}{N_{TX}}}$, is also assumed to be known by the

detector.

$$\begin{aligned} \varpi &= \mathbf{F_{ZF}}\mathbf{r} && (2.39)\\ &= \sqrt{\frac{\rho}{N_{TX}}}(\mathbf{H^T H})^{-1}\mathbf{H^T H u} + (\mathbf{H^T H})^{-1}\mathbf{H^T n} \\ &= \sqrt{\frac{\rho}{N_{TX}}}\mathbf{u} + (\mathbf{H^T H})^{-1}\mathbf{H^T n}. && (2.40) \end{aligned}$$

A decision is made for each component of ϖ; and this vector is called $\hat{\mathbf{u}}$.

$$\hat{u}_i = \arg\min_{\tilde{u}_i} \| \frac{\varpi_i}{\sqrt{\frac{\rho}{N_{TX}}}} - \tilde{u}_i \|_2. \quad (2.41)$$

2.4.1 Noise Colouring

Pre-multiplying $\mathbf{r}$ by the ZF matrix, $\mathbf{F_{ZF}}$, causes noise colouring. The covariance matrix of the additive noise vector is transformed from $\mathbf{I}$ to $(\mathbf{H^H H})^{-1}$ [25].

$$E\{(\mathbf{H^T H})^{-1}\mathbf{H^T n}\} = \mathbf{0}. \quad (2.42)$$

$$\begin{aligned} E\{[(\mathbf{H^T H})^{-1}\mathbf{H^T n}][(\mathbf{H^T H})^{-1}\mathbf{H^T n}]^T\} &= (\mathbf{H^T H})^{-1}\mathbf{H^T} E\{\mathbf{n n^T}\}\mathbf{H}(\mathbf{H^T H})^{-T} \\ &= (\mathbf{H^T H})^{-1}\mathbf{H^T I H}(\mathbf{H^T H})^{-1} \\ &= (\mathbf{H^T H})^{-1}\mathbf{H^T H}(\mathbf{H^T H})^{-1} \\ &= (\mathbf{H^T H})^{-1}. && (2.43) \end{aligned}$$

This noise colouring causes a degradation in performance in ZF detection with respect to ML detection. Following multiplication by the ZF matrix the pdf of the coloured noise, $(\mathbf{H^T H})^{-1}\mathbf{H^T n}$, no longer has the spherical pdf geometry of the undistorted noise, $\mathbf{n}$, used by the ML detector [25]. The symbol decision regions are designed for

white noise and so the amplified, coloured, noise leads to incorrect decisions and a higher probability of error.

2.4.2 Analysis of the i^{th} output of the ZF Detector ($\hat{u}_i$)

The following matrix identity is useful in the analysis of $\hat{u}_i$ [26]:

$$[\mathbf{F_{ZF}r}]_i = \frac{\mathbf{h_i^T P_{S_i}^{\perp} r}}{\|\mathbf{v_i}\|_2^2}. \tag{2.44}$$

Therefore, in ZF detection, the following decision is made for the i^{th} symbol where $\mathcal{U}$ is the set of all symbols in the constellation.

$$\begin{aligned}
\hat{u}_i &= \arg\min_{\tilde{u}_i \in \mathcal{U}} \| \frac{[\mathbf{F_{ZF}r}]_i}{\sqrt{\frac{\rho}{N_{TX}}}} - \tilde{u}_i \|_2^2 \\
&= \arg\min_{\tilde{u}_i \in \mathcal{U}} \| \frac{\mathbf{h_i^T P_{S_i}^{\perp} r}}{\sqrt{\frac{\rho}{N_{TX}}}\|\mathbf{v_i}\|_2^2} - \tilde{u}_i \|_2^2 \\
&= \arg\min_{\tilde{u}_i \in \mathcal{U}} \| \frac{\mathbf{h_i^T P_{S_i}^{\perp}} (\sqrt{\frac{\rho}{N_{TX}}}\mathbf{Hu} + \mathbf{n})}{\sqrt{\frac{\rho}{N_{TX}}}\mathbf{v_i^T v_i}} - \tilde{u}_i \|_2^2 \\
&= \arg\min_{\tilde{u}_i \in \mathcal{U}} \| \frac{\sqrt{\frac{\rho}{N_{TX}}}\mathbf{v_i^T v_i}}{\sqrt{\frac{\rho}{N_{TX}}}\mathbf{v_i^T v_i}} u_i - \tilde{u}_i + \frac{(\mathbf{P_{S_i}^{\perp} h_i})^{\mathbf{T}} \mathbf{n}}{\sqrt{\frac{\rho}{N_{TX}}}\mathbf{v_i^T v_i}} \|_2^2 \\
&= \arg\min_{\tilde{u}_i \in \mathcal{U}} \| u_i - \tilde{u}_i + \frac{(\mathbf{v_i^T v_i})^{-1}\mathbf{v_i^T n}}{\sqrt{\frac{\rho}{N_{TX}}}} \|_2^2.
\end{aligned} \tag{2.45}$$

Since $\mathbf{v_i}$ is a known vector we can also say that:

$$\begin{aligned}
\hat{u}_i &= \arg\min_{\tilde{u}_i \in \mathcal{U}} (\| u_i - \tilde{u}_i + \frac{(\mathbf{v_i^T v_i})^{-1}\mathbf{v_i^T n}}{\sqrt{\frac{\rho}{N_{TX}}}} \|_2^2 \times \| \sqrt{\frac{\rho}{N_{TX}}}\mathbf{v_i} \|_2^2) \\
&= \arg\min_{\tilde{u}_i \in \mathcal{U}} \| \sqrt{\frac{\rho}{N_{TX}}}\mathbf{v_i}(u_i - \tilde{u}_i) + \mathbf{P_{v_i} n} \|_2^2.
\end{aligned} \tag{2.46}$$

Equation (2.46) shows the relation between the i^{th} output of the ZF detector, $\hat{u}_i$, and the $\mathbf{v_i}$ subspace. This relation shows that the two factors that lead to a ZF detection error at the i^{th} symbol are a large $\mathbf{P}_{\mathbf{v_i}}\mathbf{n}$ and a small $\|\mathbf{v_i}\|_2$.

2.5 Sphere Decoder (SD)

The algorithm underlying the SD was discovered by Pohst [27] and then simplified and applied for communication systems to become known as the SD by Viterbo and Boustros [11]. The Schnorr-Euchner (SE) algorithm [15], a lower complexity variant of the SD, was first presented in a communications system context by Agrell et. al. [13] and by Chan and Lee [14]. The basic idea behind the SD, regardless of the variant, is the same. Instead of exhaustively searching through all symbol vectors only those found within a spheroid, of given radius θ, centered at the equalized received vector, ϖ, are searched. This strategy is able to lower complexity because it uses extra information, an approximation of the solution, that is not used by the exhaustive search detector. Similar to the exhaustive search detector, the SD solves the least square minimization:

$$\min_{\bar{\mathbf{u}}\in\mathcal{G}} \|\mathbf{r} - \sqrt{\frac{\rho}{N_{TX}}}\mathbf{H}\bar{\mathbf{u}}\|_2^2 = \min_{\bar{\mathbf{u}}\in\mathcal{G}} \|\sqrt{\frac{\rho}{N_{TX}}}\mathbf{H}(\mathbf{u}-\bar{\mathbf{u}}) + \mathbf{n}\|_2^2. \tag{2.47}$$

We let $\mathcal{G}$ be the set of symbol vectors in the spheroid. While still producing the ML solution the SD has much less complexity than the exhaustive search since it only considers a small subset of the symbol vectors.

We begin the SD search process by pre-multiplying $\mathbf{r}$ by $\mathbf{F_{ZF}}$. This is done to find an estimate of the solution which will be used as the center of the sphere.

$$\varpi = \mathbf{F_{ZF}}\mathbf{r}. \tag{2.48}$$

Although the calculation of ϖ causes noise colouring this doesn't effect the least square calculation since the SD still uses $\mathbf{r}$ in equation (2.59). We define the vector $\mathbf{k}$ as:

$$\mathbf{k} = \varpi - \sqrt{\tfrac{\rho}{N_{TX}}}\bar{\mathbf{u}}. \tag{2.49}$$

The SD detector uses ϖ as the center of the spheroid.

$$\begin{aligned} \min_{\bar{\mathbf{u}}\in\mathcal{G}} \|\mathbf{Hk}\|_2^2 &= \min_{\bar{\mathbf{u}}\in\mathcal{G}} \|\mathbf{H}(\varpi - \sqrt{\frac{\rho}{N_{TX}}}\bar{\mathbf{u}})\|_2^2 && (2.50)\\ &= \min_{\bar{\mathbf{u}}\in\mathcal{G}} \|\mathbf{H}(\sqrt{\frac{\rho}{N_{TX}}}\mathbf{u} + (\mathbf{H^T H})^{-1}\mathbf{H^T n} - \sqrt{\frac{\rho}{N_{TX}}}\bar{\mathbf{u}})\|_2^2 \\ &= \min_{\bar{\mathbf{u}}\in\mathcal{G}} \|\sqrt{\frac{\rho}{N_{TX}}}\mathbf{H}(\mathbf{u} - \bar{\mathbf{u}}) + \mathbf{P_H n}\|_2^2. && (2.51) \end{aligned}$$

The minimization of (2.47) and (2.51) both achieve the same result and so it is valid to use ϖ as the center of the spheroid, in spite of the introduced noise colouring. Both these equations achieve the same result because equation (2.51) only excludes noise orthogonal the subspace over which the minimization occurs.

Now we can work out a method for finding those lattice points that lie within the search spheroid.

$$\|\mathbf{Hk}\|_2^2 = \mathbf{k^T H^T Hk} \leq \theta^2. \tag{2.52}$$

Performing a cholesky factorization of $\mathbf{H^T H} = \mathbf{R^T R}$ where $\mathbf{R}$ is an upper triangular matrix yields:

$$\mathbf{k^T R^T Rk} = \|\mathbf{Rk}\|_2^2 = \|\mathbf{a}\|_2^2 \leq \theta^2. \tag{2.53}$$

We next assume that every element of $\mathbf{a}$, except the last one, is zero, making it possible to find a range of $\bar{u}_{N_{TX}}$.

$$r_{N_{TX}N_{TX}}(\varpi_{N_{TX}} - \sqrt{\tfrac{\rho}{N_{TX}}}\bar{u}_{N_{TX}}) \leq \|\theta\|_2 \tag{2.54}$$

$$\frac{\varpi_{N_{TX}}}{\sqrt{\frac{\rho}{N_{TX}}}} + \frac{\theta}{\sqrt{\frac{\rho}{N_{TX}}} r_{N_{TX}N_{TX}}} \geq \bar{u}_{N_{TX}} \geq \frac{\varpi_{N_{TX}}}{\sqrt{\frac{\rho}{N_{TX}}}} - \frac{\theta}{\sqrt{\frac{\rho}{N_{TX}}} r_{N_{TX}N_{TX}}}. \tag{2.55}$$

This inequality gives the range of $\bar{u}_{N_{TX}}$ so that $\bar{\mathbf{u}}$ remains in the spheroid. The order in which the symbols in the range are considered is an important distinction between the Viterbo-Boustros algorithm [11], implementations of which we refer to as the SD, and the lower complexity Schnorr-Euchner algorithm [15]. The Viterbo-Boustros algorithm considers the symbols in the range ordered from smallest to largest. The main advantage of using this symbol ordering is its simplicity.

To find the range of $\bar{u}_{N_{TX}-1}$ all elements of $\mathbf{a}$ except the last two are assumed to be zero. This range is found for each symbol, $\bar{u}_{N_{TX}}$, in the range defined by equation (2.55). This allows for the assumption, for the purposes of finding a range of $\bar{u}_{N_{TX}-1}$, that $\bar{u}_{N_{TX}}$ is known.

$$(r_{N_{TX}-1\,N_{TX}-1}k_{N_{TX}-1} + r_{N_{TX}-1\,N_{TX}}k_{N_{TX}})^2 + (r_{N_{TX}N_{TX}}k_{N_{TX}})^2 \leq \theta^2$$

$$r_{N_{TX}-1\,N_{TX}-1}k_{N_{TX}-1} + r_{N_{TX}-1\,N_{TX}}k_{N_{TX}} \leq \|\theta^2 - (r_{N_{TX}N_{TX}}k_{N_{TX}})^2\|_2 = g$$

$$\frac{\varpi_{N_{TX}-1}}{\sqrt{\frac{\rho}{N_{TX}}}} + \frac{g - r_{N_{TX}N_{TX}}k_{N_{TX}}}{\sqrt{\frac{\rho}{N_{TX}}} r_{N_{TX}-1\,N_{TX}-1}} \geq \bar{u}_{N_{TX}-1} \geq \frac{\varpi_{N_{TX}-1}}{\sqrt{\frac{\rho}{N_{TX}}}} - \frac{g - r_{N_{TX}N_{TX}}k_{N_{TX}}}{\sqrt{\frac{\rho}{N_{TX}}} r_{N_{TX}-1\,N_{TX}-1}} \tag{2.56}$$

$$k_{N_{TX}} = \varpi_{N_{TX}} - \sqrt{\frac{\rho}{N_{TX}}}\bar{u}_{N_{TX}}. \tag{2.57}$$

This is how the range of $\bar{u}_{N_{TX}-1}$ is calculated for one particular value of $\bar{u}_{N_{TX}}$. In this way, a search tree is built until a range for the first element, $\bar{u}_1$, is found.

The range for the i^{th} element of $\bar{\mathbf{u}}$, $\bar{u}_i$, is found by assuming that all elements of $\mathbf{a}$ prior to the i^{th} element are zero and all elements after the i^{th} element are known.

$$\frac{\varpi_i + \sum_{j=i+1}^{N_{TX}} r_{ij}k_j}{\sqrt{\frac{\rho}{N_{TX}}}} + \frac{\theta^2 - \sum_{j=i+1}^{N_{TX}} a_j^2}{\sqrt{\frac{\rho}{N_{TX}}} r_{i\,i}} \geq \bar{u}_i \geq \frac{\varpi_i + \sum_{j=i+1}^{N_{TX}} r_{ij}k_j}{\sqrt{\frac{\rho}{N_{TX}}}} - \frac{\theta^2 - \sum_{j=i+1}^{N_{TX}} a_j^2}{\sqrt{\frac{\rho}{N_{TX}}} r_{i\,i}}. \tag{2.58}$$

When a symbol vector is found within the spheroid, its distance from $\mathbf{r}$ is calculated.

$$\|\mathbf{r} - \sqrt{\tfrac{\rho}{N_{TX}}}\mathbf{H}\bar{\mathbf{u}}\|_2^2. \tag{2.59}$$

2.5.1 Complexity and Search Tree Representation of the SD

The complexity of SD is defined as the average number of loops required to complete the SD algorithm. During each loop a range of symbols is calculated and this represents a node visitation when the SD is thought of as a search tree. An example of a SD search tree realization is shown in figure 2.4 below. The bottom node in the tree is the search starting point when a range of $\bar{u}_{N_{TX}}$ is determined. The nodes in the first level up from the bottom represent the possible values of $\bar{u}_{N_{TX}}$. In this example there are three possible values of $\bar{u}_{N_{TX}}$: $\{-1, 0, 1\}$. When $\bar{u}_{N_{TX}} = -1$ the only possible value of $\bar{u}_{N_{TX}-1}$ is -1. Likewise, when $\bar{u}_{N_{TX}} = 0$ there are no possible values of $\bar{u}_{N_{TX}-1}$ and when $\bar{u}_{N_{TX}} = 1$ the two possible values of $\bar{u}_{N_{TX}-1}$ are $\{2, 3\}$. In this example the SD

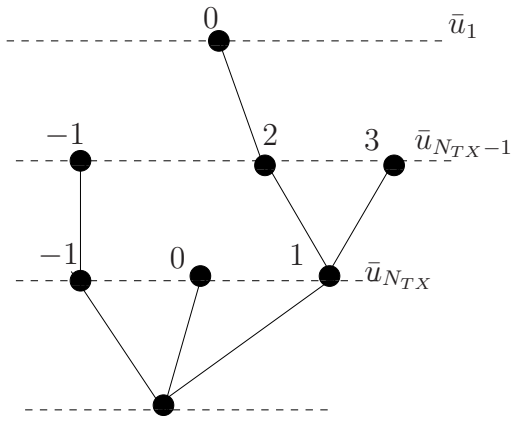

Fig. 2.4 Example of SD as a search tree

finds only one valid lattice point $\bar{\mathbf{u}} = (\bar{u}_1, \bar{u}_{N_{TX}-1}, \bar{u}_{N_{TX}})^T = (0, 2, 1)^T$ in the spheroid but there are 7 node visitations. The simulations will measure the complexity of any SD variant as the average number nodes visited during the search process.

2.5.2 QR Factorization with the Householder Transform (HT)

The SD requires a Cholesky factorization of the Gram matrix, $\mathbf{G_H}$, of the channel matrix, $\mathbf{H}$. It has been observed, however, that calculating the Gram matrix causes a loss in precision [28]. One method of performing the Cholesky factorization without having to calculate the Gram matrix is with a QR factorization of $\mathbf{H}$. Then, a Cholesky factorization of the Gram matrix is $\mathbf{R^T R}$, where $\mathbf{R}$ is the upper triangular matrix obtained from the QR factorization of $\mathbf{H}$.

$$\begin{aligned} \mathbf{G_H} &= \mathbf{H^T H} && (2.60) \\ &= (\mathbf{QR})^T \mathbf{QR} \\ &= \mathbf{R^T Q^T QR} \\ &= \mathbf{R^T R}. && (2.61) \end{aligned}$$

A stable way to perform the QR factorization of $\mathbf{H}$ is by repeated application of the Householder Transform (HT) [28]. The HT allows for the annihilation to 0 of several entries in a column at the same time.

The HT, expressed as a matrix, $\mathbf{\Theta}$, pre-multiplies a given vector, say $\mathbf{x}$, and the result is the first basis vector of the same length, $\alpha\mathbf{e_0}$, with $\mathbf{e_0} = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}^T$. Therefore $\alpha = \|\mathbf{x}\|_2$ and

$$\mathbf{\Theta x} = \alpha\mathbf{e_0}. \quad (2.62)$$

We show that $\mathbf{\Theta} = \mathbf{I} - \mathbf{P_g}$ by considering an isosceles triangle with equal length sides $\mathbf{x}$ and $\alpha\mathbf{e_0}$ and base $\mathbf{g} = \mathbf{x} - \alpha\mathbf{e_0}$.

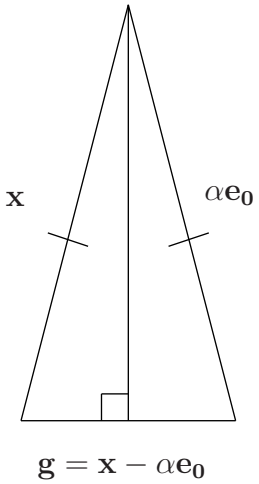

Fig. 2.5 Isosceles triangle used to derive the Householder Transform

$$\begin{aligned} \alpha\mathbf{e_0} &= \mathbf{x} - 2\mathbf{P_g}\mathbf{x} \\ &= (\mathbf{I} - \mathbf{P_g})\mathbf{x} \\ &= \mathbf{\Theta}\mathbf{x}. \end{aligned} \tag{2.63}$$

We begin the QR factorization process by finding the HT for the first column of $\mathbf{H}$, $\mathbf{h_1}$, which we call $\mathbf{\Theta_1}$. Then we create $\mathbf{\Theta_2}$ from the second column of the product $\mathbf{\Theta_1 H}$ without the first element. In this way we annihilate to 0 all elements below the diagonal of $\mathbf{H}$.

$$\mathbf{R} = \mathbf{\Theta}_{\mathbf{N_{TX}}} \ldots \mathbf{\Theta_2 \Theta_1 H}. \tag{2.64}$$

2.5.3 Practical Selection of the SD Radius

Of great practical importance to the SD is the selection of the spheroid radius. In the original SD paper [11] no method of radius selection is given but it is suggested that the radius can be dynamically adjusted. For instance, if no points are found within an initially selected radius then this radius should be increased and the search repeated

until such time as at least one symbol vector is discovered within the spheroid.

When the Schnorr-Euchner algorithm is incorporated into the SD [13,14,29] the computational complexity of this modified SD is seen to be less dependant on the chosen radius [14]. Therefore, for ease of comparison, all of the SD implementations in this work will use the same static radius selection method. A practical benefit of this method is that at least one symbol vector is always located within it's selected radius.

All of the implemented SDs pick the spheroid radius, θ, as follows, using the output of the ZF detector.

$$\begin{aligned} \theta^2 &= \|\mathbf{H}(\mathbf{F_{ZF}r} - \sqrt{\frac{\rho}{N_{TX}}}\hat{\mathbf{u}})\|_2^2 \\ &= \|\mathbf{R}(\varpi - \sqrt{\frac{\rho}{N_{TX}}}\hat{\mathbf{u}})\|_2^2. \end{aligned} \tag{2.65}$$

This ensures that at least one symbol vector, $\hat{\mathbf{u}}$, is in the spheroid of radius θ.

2.5.4 Schnorr-Euchner Variant of the SD (SE/SD)

The SE/SD has lower complexity, as defined in section 2.5.1, than the SD. The SE/SD begins searching for symbol vectors at the center of the spheroid rather than at the edge, as is done in the original SD [11]. This altered search ordering allows the ML solution to be found more quickly since the ML solution is most likely to be found closest to the center of spheroid [14]. Except for this altered search ordering, SE/SD and SD algorithms are the same, and both have the same ML performance.

The range of the i^{th} symbol in the symbol vector, $\bar{u}_i$, contained in the spheroid of radius θ, is given by equation (2.58). SE/SD sorts the symbols in this range in ascending order of the following metric, where $y \in \mathcal{H}$ and $\mathcal{H}$ is the set of symbols in

the range of the i^{th} symbol.

$$|y - \frac{\varpi_i + \sum_{j=i+1}^{N_{TX}} r_{ij} k_j}{\sqrt{\frac{\rho}{N_{TX}}}}|^2. \tag{2.66}$$

The following computer simulation confirms the lower complexity of the SE/SD compared to the SD. The relative lower complexity of the SE/SD is seen to decrease

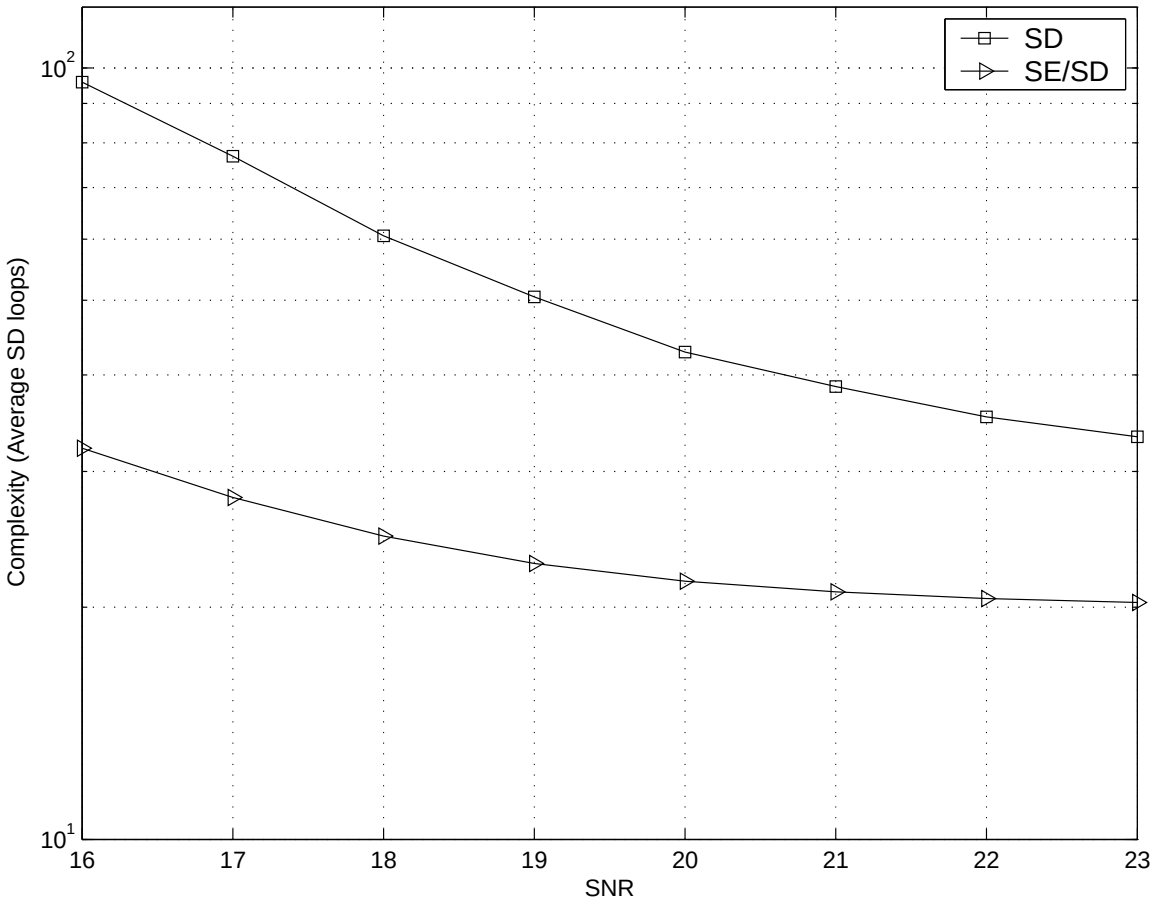

Fig. 2.6 Complexity comparison between SD and SE/SD: uncoded 16-QAM $N_{TX} = 5$ and $N_{RX} = 7$

with increasing SNR. This reducing complexity advantage occurs because at higher SNR fewer symbol vectors are found in the spheroid and therefore the order in which they are searched becomes less important.

2.6 LD Code for Multiple Antennas

The LD code used in the computer simulations was discovered in the original LD paper by Hassibi and Hochwald [10]. The code is designed for $N_{TX} = 2$, $N_{RX} = 2$, and a block length, L, of 2 channel-uses. A QPSK symbol constellation is used in conjunction with this code.

The simulations will compare the LD coded transmission with uncoded transmission at an equal rate. In uncoded transmission four symbols are transmitted from two transmit antennas in two channel uses which, assuming QPSK modulation, results in a rate, R_u, of 4. To set the LD coded transmission rate, R_c, equal to the uncoded rate, R_u, we choose $Q = 4$, and also transmit 4 symbols during each block length.

The LD code can be completely described by two sets of matrices: $\mathbf{A_q}$, $\mathbf{B_q} \in \mathcal{C}^{L \times N_{TX}}$, $q \in [1, Q]$ as shown in equation (2.11). For the code used in the simulations these matrices are:

$$\mathbf{A_1} = \mathbf{B_1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{2.67}$$

$$\mathbf{A_2} = \mathbf{B_2} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \tag{2.68}$$

$$\mathbf{A_3} = \mathbf{B_3} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \tag{2.69}$$

and

$$\mathbf{A_4} = \mathbf{B_4} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{2.70}$$

2.7 Computer Simulations

The C++ source code for the programs that generated the computer simulations included as part of this thesis can be found in an attached CD. Appendix A is a guide explaining how to use these C++ programs as well as their capabilities and limitations.

Of primary importance, however, is an understanding of the derived graphical results. Three types of results are presented: Bit Error Rate (BER) performance, Symbol Error Rate (SER) performance, and decoder complexity. To ensure accurate results a minimum of 100 errors was required for each decoding method at each SNR. The simulations also ensure a minimum of 100 000 channel realizations.

The decoder complexity from the simulations is defined as the SD complexity that was described in Section 2.5.1. The SD implemented in these simulations is based on the flowchart from [11]. The SE/SD implemented in these simulations is based on the flowchart from [14].

Chapter 3

Statistical Partial Coverage (SPC) Front-End

A detector combining the SPC front-end with the SD (SPC-SD) reduces computational complexity compared to the standalone SD. The SPC-SD's lower computational complexity as compared to the SD is a result of individually pre-detecting a percentage of the symbols in each symbol vector so that fewer symbols are left to be jointly detected by the SD. The performance of the SPC-SD detector can approach the performance of the SD detector because the SPC front-end is designed to only pre-detect symbols which have a very high probability of being correct. Correct pre-detection ensures that cancelation of these pre-detected symbols does not cause further distortion (from error propagation) and allows for joint detection of all the transmitted symbols.

The steps in the SPC front-end are:

1. Obtain the output of the zero-forcing detector ($\hat{\mathbf{u}}$ from Section 2.4).

2. Obtain the output of a decision feedback step.

3. For each symbol in the symbol vector:

(a) Calculate the Generalized Likelihood Ratio (GLR), L_i, as to whether $u_i = \hat{u}_i$.

(b) Compute a GLR threshold (κ_i) that ensures a small probability that $u_i \neq \hat{u}_i$ if $L_i < \kappa_i$.

(c) Pre-detect the i^{th} symbol in the symbol vector. If $L_i < \kappa_i$ we decide that $u_i = \hat{u}_i$.

4. Cancel the pre-detected symbols.

Section 3.2.3 describes how the error propagation associated with SPC front-end can be made as unlikely as desired by proper choice of the parameter U.

3.1 Decision Feedback Step

The SPC front-end begins by calculating $\hat{\mathbf{u}}$, the output of the zero-forcing detector, and then by calculating $\mathbf{w}$, the output of a decision feedback step.

$$\begin{aligned} \mathbf{w} &= \mathbf{r} - \sqrt{\frac{\rho}{N_{TX}}}\mathbf{H}\hat{\mathbf{u}} \\ &= \sqrt{\frac{\rho}{N_{TX}}}\mathbf{H}(\mathbf{u} - \hat{\mathbf{u}}) + \mathbf{n} \qquad (3.1) \\ &= \mathbf{He} + \mathbf{n}. \qquad (3.2) \end{aligned}$$

$$\mathbf{e} = \sqrt{\tfrac{\rho}{N_{TX}}}(\mathbf{u} - \hat{\mathbf{u}}). \qquad (3.3)$$

Each element of $\mathbf{e}$ is either zero or an integer multiple of $\sqrt{\frac{\rho}{N_{TX}}}d$. d is the minimum distance between symbols in the given constellation. If $\hat{\mathbf{u}} = \mathbf{u}$ then $\mathbf{w} = \mathbf{n}$ and the vector $\mathbf{w}$ will only be made up of noise. The purpose of the decision feedback step is to obtain $\mathbf{w}$, a vector that either contains only noise or a signal plus noise (if $\hat{\mathbf{u}} \neq \mathbf{u}$), in order to use SMF [21].

3.2 Individual Pre-Detection

In order to individually pre-detect the i^{th} symbol in the symbol vector we splice the channel matrix, $\mathbf{H}$, into $\mathbf{S_i}$, defined as $\mathbf{H}$ without the i^{th} column, and $\mathbf{h_i}$. Similarly, we slice $\mathbf{e}$ into its i^{th} element, e_i, and $\phi_\mathbf{i}$, defined as $\mathbf{e}$ without the i^{th} element. The values of e_i and $\phi_\mathbf{i}$ are treated as unknown parameters.

$$\mathbf{w} = \mathbf{h_i} e_i + \mathbf{S_i}\phi_\mathbf{i} + \mathbf{n}. \tag{3.4}$$

The pre-detection problem is to test between the following two probability distributions of $\mathbf{w}$ [21].

$$\begin{aligned} H_0 : \hat{u}_i = u_i \rightarrow e_i = 0 \rightarrow \mathbf{w} : \mathcal{N}\left[\mathbf{S_i}\phi_\mathbf{i}, \mathbf{I}\right] \qquad vs \\ H_1 : \hat{u}_i \neq u_i \rightarrow e_i \neq 0 \rightarrow \mathbf{w} : \mathcal{N}\left[\mathbf{h_i} e_i + \mathbf{S_i}\phi_\mathbf{i}, \mathbf{I}\right]. \end{aligned} \tag{3.5}$$

If, for the i^{th} symbol, the SPC front-end chooses the hypothesis H_0 then $\hat{u}_i$ becomes the pre-detected symbol otherwise the i^{th} symbol is not pre-detected and is left to be detected by the SD. The hypothesis test is done using a Generalized Likelihood Ratio (GLR), employing maximum likelihood estimates of the unknown parameters.

3.2.1 Maximum Likelihood Estimates (MLE)

The GLR requires the MLE of the unknown parameters in the likelihood functions. We use the convention where the MLE of a variable will have the same variable name with the addition of a hat. The particular hypothesis being assumed to be true will be indicated by a superscript.

The MLE of $\phi_\mathbf{i}$ under H_1, $\hat{\phi}_\mathbf{i}^\mathbf{1}$, can be obtained using the oblique projection matrix

$\mathbf{E_{S_i h_i}}$ [21].

$$\begin{aligned} \mathbf{S_i}\hat{\phi}_\mathbf{i}^\mathbf{1} &= \mathbf{E_{S_i h_i}}\mathbf{w} \\ \hat{\phi}_\mathbf{i}^\mathbf{1} &= (\mathbf{S_i^T P_{h_i}^{\perp} S_i})^{-1}\mathbf{S_i^T P_{h_i}^{\perp}}\mathbf{w} \\ &= \phi_\mathbf{i} + (\mathbf{S_i^T P_{h_i}^{\perp} S_i})^{-1}\mathbf{S_i^T P_{h_i}^{\perp}}\mathbf{n}. \end{aligned} \tag{3.6}$$

An orthogonal projection matrix, $\mathbf{P_{S_i}} = \mathbf{S_i}(\mathbf{S_i^T S_i})^{-1}\mathbf{S_i^T}$, is used to obtain $\hat{\phi}_\mathbf{i}^\mathbf{0}$.

$$\begin{aligned} \mathbf{S_i}\hat{\phi}_\mathbf{i}^\mathbf{0} &= \mathbf{P_{S_i}}\mathbf{w} \\ \hat{\phi}_\mathbf{i}^\mathbf{0} &= (\mathbf{S_i^T S_i})^{-1}\mathbf{S_i^T}\mathbf{w} \\ &= \phi_\mathbf{i} + (\mathbf{S_i^T S_i})^{-1}\mathbf{S_i^T}\mathbf{n}. \end{aligned} \tag{3.7}$$

The MLE of e_i under H_1, $\hat{e}_i^1$, can be obtained using the oblique projection matrix $\mathbf{E_{h_i S_i}}$ [21].

$$\begin{aligned} \mathbf{h_i}\hat{e}_i^1 &= \mathbf{E_{h_i S_i}}\mathbf{w} \\ \hat{e}_i^1 &= (\mathbf{h_i^T P_{S_i}^{\perp} h_i})^{-1}\mathbf{h_i^T P_{S_i}^{\perp}}\mathbf{w} \\ &= e_i + (\mathbf{h_i^T P_{S_i}^{\perp} h_i})^{-1}\mathbf{h_i^T P_{S_i}^{\perp}}\mathbf{n}. \end{aligned} \tag{3.8}$$

The MLE of e_i under H_0, $\hat{e}_i^0 = 0$.

3.2.2 Generalized Likelihood Ratio (GLR)

The probability density function of the multivariate normal random vector $\mathbf{w}$ is:

$$f(\mathbf{w}; e_i, \phi_\mathbf{i}) = (2\pi)^{-N_{TX}} \exp\{-\tfrac{1}{2}\|\mathbf{w} - \mathbf{h_i}e_i - \mathbf{S_i}\phi_\mathbf{i}\|_2^2\}. \tag{3.9}$$

The likelihood function for this distribution is a function of $(e_i, \phi_{\mathbf{i}})$ with parameter $\mathbf{w}$ [21].

$$l(e_i, \phi_{\mathbf{i}}; \mathbf{w}) = (2\pi)^{-N_{TX}} \exp\{-\tfrac{1}{2}\|\mathbf{w} - \mathbf{h_i}e_i - \mathbf{S_i}\phi_{\mathbf{i}}\|_2^2\}. \tag{3.10}$$

The likelihood ratio, Λ_{i10}, for any two values $(e_{i1}, \phi_{\mathbf{i1}})$ and $(e_{i0}, \phi_{\mathbf{i0}})$ is now:

$$\Lambda_{i10}(\mathbf{w}) = \tfrac{l(e_i^1, \phi_{\mathbf{i}}^{\mathbf{1}}; \mathbf{w})}{l(e_i^0, \phi_{\mathbf{i}}^{\mathbf{0}}; \mathbf{w})}. \tag{3.11}$$

Replacing the values $\phi_{\mathbf{i}}$ and e_i by the their MLE under H_1 and H_0 we obtain the GLR.

$$\begin{aligned}
\Lambda_{i10}(\mathbf{w}) &= \frac{l(\hat{e}_i^1, \hat{\phi}_{\mathbf{i}}^{\mathbf{1}}; \mathbf{w})}{l(\hat{e}_i^0, \hat{\phi}_{\mathbf{i}}^{\mathbf{0}}; \mathbf{w})} \\
&= \frac{(2\pi)^{-N_{TX}} \exp\{-\frac{1}{2}\|\mathbf{w} - \mathbf{h_i}\hat{e}_i^1 - \mathbf{S_i}\hat{\phi}_{\mathbf{i}}^{\mathbf{1}}\|_2^2\}}{(2\pi)^{-N_{TX}} \exp\{-\frac{1}{2}\|\mathbf{w} - \mathbf{h_i}\hat{e}_i^0 - \mathbf{S_i}\hat{\phi}_{\mathbf{i}}^{\mathbf{0}}\|_2^2\}} \\
&= \exp\{-\frac{1}{2}\|\mathbf{w} - \mathbf{h_i}\hat{e}_i^1 - \mathbf{S_i}\hat{\phi}_{\mathbf{i}}^{\mathbf{1}}\|_2^2 + \frac{1}{2}\|\mathbf{w} - \mathbf{h_i}\hat{e}_i^0 - \mathbf{S_i}\hat{\phi}_{\mathbf{i}}^{\mathbf{0}}\|_2^2\} \\
&= \exp\{\frac{1}{2}\|\mathbf{w} - \mathbf{P_{S_i}}\mathbf{w}\|_2^2 - \frac{1}{2}\|\mathbf{w} - \mathbf{E_{h_iS_i}}\mathbf{w} - \mathbf{E_{S_ih_i}}\mathbf{w}\|_2^2\} \\
&= \exp\{\frac{1}{2}\|\left[\mathbf{I} - \mathbf{P_{S_i}}\right]\mathbf{w}\|_2^2 - \frac{1}{2}\|\left[\mathbf{I} - (\mathbf{E_{h_iS_i}} + \mathbf{E_{S_ih_i}})\right]\mathbf{w}\|_2^2\}.
\end{aligned}$$

Combining $\mathbf{E_{h_iS_i}}$ and $\mathbf{E_{S_ih_i}}$ as described by equation (2.27) in section 2.3.1.

$$\begin{aligned}
\Lambda_{i10}(\mathbf{w}) &= \exp\{\frac{1}{2}\|\mathbf{P}_{\mathbf{S_i}}^{\perp}\mathbf{w}\|_2^2 - \frac{1}{2}\|\left[\mathbf{I} - \mathbf{P_{h_iS_i}}\right]\mathbf{w}\|_2^2\} \\
&= \exp\{\frac{1}{2}\|\mathbf{P}_{\mathbf{S_i}}^{\perp}\mathbf{w}\|_2^2 - \frac{1}{2}\|\mathbf{P}_{\mathbf{h_iS_i}}^{\perp}\mathbf{w}\|_2^2\}.
\end{aligned} \tag{3.12}$$

The GLR can be simplified further by replacing it by a scaled logarithmic GLR [21].

$$\begin{aligned} L_i(\mathbf{w}) &= 2\ln\Lambda_{i10}(\mathbf{w}) \\ &= \|\mathbf{P}^{\perp}_{\mathbf{S_i}}\mathbf{w}\|_2^2 - \|\mathbf{P}^{\perp}_{\mathbf{h_i S_i}}\mathbf{w}\|_2^2 \\ &= \mathbf{w^T}\mathbf{P}^{\perp}_{\mathbf{S_i}}\mathbf{w} - \mathbf{w^T}\mathbf{P}^{\perp}_{\mathbf{S_i h_i}}\mathbf{w} \\ &= \mathbf{w^T}(\mathbf{P}^{\perp}_{\mathbf{S_i}} - \mathbf{P}^{\perp}_{\mathbf{S_i h_i}})\mathbf{w} \\ &= \mathbf{w^T}(\mathbf{I} - \mathbf{P}_{\mathbf{S_i}} - (\mathbf{I} - \mathbf{P}_{\mathbf{S_i h_i}}))\mathbf{w} \\ &= \mathbf{w^T}(-\mathbf{P}_{\mathbf{S_i}} + \mathbf{P}_{\mathbf{S_i h_i}})\mathbf{w}. \end{aligned} \tag{3.13}$$

Using the projection decomposition described by equation (2.24) to decompose $\mathbf{P_{S_i h_i}}$:

$$\begin{aligned} L_i(\mathbf{w}) &= \mathbf{w^T}(-\mathbf{P}_{\mathbf{S_i}} + \mathbf{P}_{\mathbf{S_i}} + \mathbf{P}_{\mathbf{P}^{\perp}_{\mathbf{S_i}}\mathbf{h_i}})\mathbf{w} \\ &= \mathbf{w^T}\mathbf{P}_{\mathbf{P}^{\perp}_{\mathbf{S_i}}\mathbf{h_i}}\mathbf{w} \\ &= \mathbf{w^T}\mathbf{P}_{\mathbf{v_i}}\mathbf{w} \\ &= \|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2^2. \end{aligned} \tag{3.14}$$

The vector $\mathbf{v_i} = \mathbf{P}^{\perp}_{\mathbf{S_i}}\mathbf{h_i}$ represents the subspace of $\mathbf{h_i}$ orthogonal to the $\mathbf{S_i}$ subspace. This result makes intuitive sense because by projecting $\mathbf{w}$ onto the $\mathbf{v_i}$ subspace the scaled GLR is not influenced by ZF detection errors in any non-i^{th} symbols. Also, as was shown in section 2.4.2 ZF detection errors are strongly and uniquely influenced by the realization of $\mathbf{v_i}$.

The distribution of $L_i(\mathbf{w})$ is chi-squared with nonconcentrality parameter λ_i^2 and one degree of freedom [21].

$$L_i(\mathbf{w}) : \chi_1^2(\lambda_i^2). \tag{3.15}$$

$$\lambda_i^2 = e_i^2\mathbf{h_i^T}\mathbf{P}^{\perp}_{\mathbf{S_i}}\mathbf{h_i} = e_i^2\|\mathbf{v_i}\|_2^2. \tag{3.16}$$

3.2.3 SPC Front-End Individual Symbol Pre-Detection

In this case, instead of using $L_i(\mathbf{w}) = \|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2^2$ the SPC front-end uses $\|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2$ in the hypothesis test. $\|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2$ contains the same information as $\|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2^2$ but avoids one squaring operation. The decision threshold for the i^{th} symbol, κ_i, is calculated for each symbol in the symbol vector.

$$\|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2 \underset{H_1}{\overset{H_0}{\lessgtr}} \kappa_i. \tag{3.17}$$

If the above hypothesis test chooses H_0 then SPC pre-detection occurs and $\hat{u}_i$ is chosen as the i^{th} symbol output. The probability of false alarm (P_F) for the SPC front-end is defined as:

$$\begin{aligned} P_F &= P[H_1|H_0] \\ &= P[\|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2 \geq \kappa_i|H_0] \\ &= P[\|\mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \geq \kappa_i] \\ &= 1 - P[\chi_1(0) < \kappa_i]. \end{aligned} \tag{3.18}$$

If $P_F = 1$ then every time $e_i = 0$ the SPC front-end chooses H_1. However, the SPC front-end only reduces complexity when $e_i = 0$ and the hypothesis test chooses H_0. Choosing H_0 reduces complexity because this means that the i^{th} symbol is pre-detected and the SD therefore has one less symbol to jointly detect. Therefore, a smaller P_F results in a larger the SPC front-end complexity reduction. The probability

of detection (P_D) for the SPC front-end is defined as:

$$\begin{aligned} P_D &= P[H_1|H_1] \\ &= P[\|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2 \geq \kappa_i|H_1] \\ &= P[\chi_1(\lambda_i^2) \geq \kappa_i] \\ &= 1 - P[\chi_1(\lambda_i^2) < \kappa_i]. \end{aligned} \tag{3.19}$$

If $P_D = 1$ then every time $e_i \neq 0$ the SPC front-end chooses H_1. $1 - P_D$ is therefore the probability that the SPC front-end makes a pre-detection error. It is these pre-detection errors which cause the SPC-SD detector to have worse that ML performance.

The key to ensuring that the SPC-SD detector has close to ML performance is to find the largest κ_i than maintains a P_D close to one. A larger κ_i reduces P_F and therefore reduces complexity. We choose the hypothesis test threshold, κ_i, based on the channel realization, SNR, symbol constellation and a probabilistic measure of the noise projected onto the $\mathbf{v_i}$ subspace.

$$\kappa_i = \max(\|\mathbf{v_i}\|_2\sqrt{\tfrac{\rho}{N_{TX}}}d - U, 0). \tag{3.20}$$

The logic behind this choice of κ_i can be seen by considering the Figure 3.1 below. This figure shows the relationship between the noiseless value of $\|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2$ when $e_i = 0$ and when $e_i = \pm d\sqrt{\frac{\rho}{N_{TX}}}$ in the $\mathbf{v_i}$ subspace. These points are equidistant with this distance being $d\sqrt{\frac{\rho}{N_{TX}}}\|\mathbf{v_i}\|_2$. Considering noise in the $\mathbf{v_i}$ subspace, it can be seen that if $\|\mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \leq U$ and $\|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2 < \kappa_i$ then $e_i = 0$. Although it is not always the case that $\|\mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \leq U$ choosing a large U can ensure that this inequality is nearly always

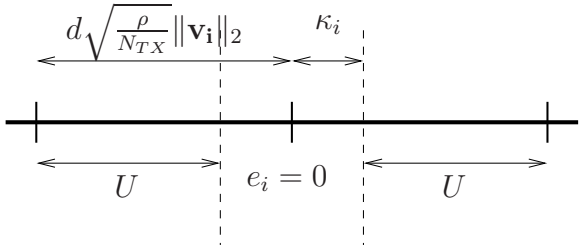

Fig. 3.1 Noiseless values of $\|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2$ when $e_i = 0$ and $e_i = \pm d\sqrt{\frac{\rho}{N_{TX}}}$ in the $\mathbf{v_i}$ subspace

satisfied.

$$\begin{aligned} \alpha &= P[\|\mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \leq U] \\ &= P[\chi_1(0) \leq U]. \end{aligned} \tag{3.21}$$

Clearly α can be made arbitrarily close to one by proper selection of U.

If, for the i^{th} symbol, $\kappa_i = 0$ then $P_F = 1$ and $P_D = 1$ since the hypothesis test, equation (3.17), always selects H_1. When $\kappa_i = 0$ the SPC front-end never pre-detects the i^{th} symbol.

In the following analysis we assume that $\|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d > U$, and so $\kappa_i > 0$, the case where the SPC front-end may pre-detect the i^{th} symbol. We wish to show that the value of $\kappa_i = \|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d - U$ assures a P_D close to 1. Letting $\tilde{\mathbf{n}}$ be the realization of $\mathbf{n}$ we can define the event E.

$$\begin{aligned} E &: \|\mathbf{P}_{\mathbf{v_i}}\tilde{\mathbf{n}}\| \leq U. \\ \overline{E} &: \|\mathbf{P}_{\mathbf{v_i}}\tilde{\mathbf{n}}\| > U. \end{aligned} \tag{3.22}$$

From our definition of U we have the following two probabilities:

$$\begin{aligned} P[E] &= \alpha \\ P[\overline{E}] &= 1-\alpha. \end{aligned} \tag{3.23}$$

The orthogonal projection $\mathbf{P}_{\mathbf{v_i}}\mathbf{w}$ is a random vector because it contains the random noise vector, $\mathbf{n}$, in the following way:

$$\begin{aligned} \mathbf{P}_{\mathbf{v_i}}\mathbf{w} &= \mathbf{P}_{\mathbf{v_i}}\left(\mathbf{h_i}e_i + \mathbf{S_i}\phi_\mathbf{i} + \mathbf{n}\right) \\ &= \mathbf{v_i}e_i + \mathbf{P}_{\mathbf{v_i}}\mathbf{n}. \end{aligned} \tag{3.24}$$

$$\|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2 = \|\mathbf{v_i}e_i + \mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2. \tag{3.25}$$

In the next subsection we will prove that $P[\|\mathbf{v_i}e_i + \mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \geq \kappa_i | H_1, E] = 1$ and that $P[\|\mathbf{v_i}e_i + \mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \geq \kappa | H_1, \overline{E}] \geq 1/2$ so that, employing the chain rule:

$$\begin{aligned} P_D &= P[\|\mathbf{P}_{\mathbf{v_i}}\mathbf{w}\|_2 \geq \kappa_i | H_1] \\ &= P[\|\mathbf{v_i}e_i + \mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \geq \kappa_i | H_1, E]P[E] + P[\|\mathbf{v_i}e_i + \mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \geq \kappa_i | H_1, \overline{E}]P[\overline{E}] \\ &= P[\|\mathbf{v_i}e_i + \mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \geq \kappa_i | H_1, E]\alpha + P[\|\mathbf{v_i}e_i + \mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \geq \kappa_i | H_1, \overline{E}](1-\alpha) \\ &\geq [1]\alpha + [1/2](1-\alpha) \\ &\geq \frac{1}{2} + \frac{1}{2}\alpha \\ &\geq \frac{1}{2} + \frac{1}{2}P[\chi_1(0) \leq U]. \end{aligned} \tag{3.26}$$

This shows that the lower bound on P_D can be made arbitrarily close to 1 by proper selection of U, the SPC front-end system parameter. A larger U guarantees a tighter lower bound on P_D and thus guarantees that the SPC-SD detector will have closer to ML performance. However, a larger U also results in less of a complexity advantage,

as expressed in terms of P_F in equation (3.18). The SPC-SD simulations will show this U dependent tradeoff between performance and complexity.

3.2.4 Proofs for the Lower Bound on P_D

For both of the proofs used to derive a lower bound on P_D figure 3.1 in the preceding section will be important.

We begin with the proof of

$$P_1 = P[\|\mathbf{v_i}e_i + \mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \geq \kappa_i | H_1, E] = 1. \tag{3.27}$$

Substituting for κ_i in (3.27):

$$P_1 = P[\|\mathbf{v_i}e_i + \mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \geq \|\mathbf{v_i}\|_2\sqrt{\tfrac{\rho}{N_{TX}}}d - U | H_1, E]. \tag{3.28}$$

Given H_1, the smallest value that e_i can have is: $e_i = \sqrt{\frac{\rho}{N_{TX}}}d$, where d is the minimum distance between constellation points.

$$P_1 \geq P[\|\mathbf{v_i}\sqrt{\tfrac{\rho}{N_{TX}}}d + \mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \geq \|\mathbf{v_i}\|_2\sqrt{\tfrac{\rho}{N_{TX}}}d - U | E]. \tag{3.29}$$

We are given the event $E: \|\mathbf{P}_{\mathbf{v_i}}\tilde{\mathbf{n}}\| \leq U$. The worst case is that $\mathbf{v_i}$ and $\mathbf{P}_{\mathbf{v_i}}\tilde{\mathbf{n}}$ are in opposite directions and $\|\mathbf{P}_{\mathbf{v_i}}\tilde{\mathbf{n}}\| = U$.

$$\begin{aligned} P_1 &\geq P[\|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d - \|\mathbf{P}_{\mathbf{v_i}}\mathbf{n}\|_2 \geq \|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d - U | E] \\ &\geq P[\|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d - U \geq \|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d - U] = 1. \end{aligned} \tag{3.30}$$

qed.

Proceeding to the proof of

$$P_2 = P[\|\mathbf{v_i}e_i + \mathbf{P_{v_i}}\mathbf{n}\|_2 \geq \kappa_i | H_1, \overline{E}] \geq 1/2. \tag{3.31}$$

Substituting for κ_i in (3.31):

$$P_2 = P[\|\mathbf{v_i}e_i + \mathbf{P_{v_i}}\mathbf{n}\|_2 \geq \|\mathbf{v_i}\|_2\sqrt{\tfrac{\rho}{N_{TX}}}d - U | H_1, \overline{E}]. \tag{3.32}$$

Given H_1, the smallest value that e_i can have is: $e_i = \sqrt{\frac{\rho}{N_{TX}}}d$, where d is the minimum distance between constellation points.

$$P_2 \geq P[\|\mathbf{v_i}\sqrt{\tfrac{\rho}{N_{TX}}}d + \mathbf{P_{v_i}}\mathbf{n}\|_2 \geq \|\mathbf{v_i}\|_2\sqrt{\tfrac{\rho}{N_{TX}}}d - U | \overline{E}] = P_3. \tag{3.33}$$

The event that the random vector $\mathbf{P_{v_i}}\mathbf{n}$ is in the same direction as the known vector $\mathbf{v_i}$ is called R.

$$P[R] = 1/2. \tag{3.34}$$

Because $\mathbf{v_i}$ is an arbitrary vector and $\mathbf{n}$ is a white gaussian random vector with a spherical pdf geometry there will be a $1/2$ probability that $\mathbf{n}$ is in the same direction as the half sphere centered by $\mathbf{v_i}$.

Employing the chain rule:

$$\begin{aligned} P_3 &= P[\|\mathbf{v_i}\sqrt{\frac{\rho}{N_{TX}}}d + \mathbf{P_{v_i}n}\|_2 \geq \|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d - U|\overline{E}, R]P[R] + \\ &\quad P[\|\mathbf{v_i}\sqrt{\frac{\rho}{N_{TX}}}d + \mathbf{P_{v_i}n}\|_2 \geq \|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d - U|\overline{E}, \overline{R}]P[\overline{R}] \\ &= P[\|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d + \|\mathbf{P_{v_i}n}\|_2 \geq \|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d - U|\overline{E}]1/2 + \\ &\quad P[\|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d - \|\mathbf{P_{v_i}n}\|_2 \geq \|\mathbf{v_i}\|_2\sqrt{\frac{\rho}{N_{TX}}}d - U|\overline{E}]1/2 \\ &= [1]1/2 + [0]1/2 = 1/2. \end{aligned} \tag{3.35}$$

qed.

3.3 Canceling the Pre-Detected Symbols

The final step in the SPC front-end is to cancel from the received vector, $\mathbf{r}$, those symbols that have been pre-detected. By canceling the pre-detected symbols we obtain, if there are no pre-detection errors, ML joint detection performance.

Let $\mathcal{D}$ be the set of indexes of the transmitted real symbol vector, $\mathbf{u}$, where hypothesis H_0 is satisfied. The size of $\mathcal{D}$, called g, depends on the system configuration and channel realization.

$$\mathcal{D} = \{i \in [1, 2N_{TX}] \mid H_0\} \tag{3.36}$$

$$= \{i \in [1, 2N_{TX}] \mid \|\mathbf{P_{v_i}w}\|_2 < \kappa_i\}. \tag{3.37}$$

The columns of $\mathbf{H} \in \mathcal{R}^{2N_{RX} \times 2N_{TX}}$ are divided between $\mathbf{H_d} \in \mathcal{R}^{2N_{RX} \times g}$ and $\mathbf{H_d} \in \mathcal{R}^{2N_{RX} \times (2N_{TX} - g)}$ according to set $\mathcal{D}$. The columns of $\mathbf{H}$ with indexes $\in \mathcal{D}$ are put in $\mathbf{H_d}$ and the columns of $\mathbf{H}$ with indexes $\notin \mathcal{D}$ are put in $\mathbf{H_n}$. Similarly, $\hat{\mathbf{u}}_\mathbf{d} \in \mathcal{R}^g$, are

the elements of $\hat{u}$ with indexes $\in \mathcal{D}$. The result of the cancelation is called $\tilde{\mathbf{r}}$.

$$\tilde{\mathbf{r}} = \mathbf{r} - \sqrt{\frac{\rho}{N_{TX}}}\mathbf{H_d}\hat{\mathbf{u}}_\mathbf{d}. \tag{3.38}$$

Then, $\tilde{\mathbf{r}}$ is pre-multiplied by a ZF matrix created from $\mathbf{H_n}$.

$$\tilde{\mathbf{u}} = (\mathbf{H_n^T H_n})^{-1}\mathbf{H_n^T}\tilde{\mathbf{r}}. \tag{3.39}$$

The values of $\tilde{\mathbf{u}}_\mathbf{n}$ and $\mathbf{H_n}$ can then be used as inputs to the SD that follows the SPC front-end.

3.4 Simulation of the SPC Front-End

Computer simulations of the SPC front-end are done using the software on the attached CD. These simulations are intended to confirm the analytical results from section 3.2 and to show the tradeoff between performance and complexity for three values of the SPC front-end parameter U. SPC-SD denotes a decoder combining the SPC front-end with the SD. SPC-SD simulation is done for both the uncoded and LD coded frameworks. For each framework, both BER and SER performance curves are presented as well as a measure of complexity. The table bellow summarizes the SPC front-end simulations.

	Uncoded Framework	LD Coded Framework
SER Performance	figure 3.2	figure 3.5
BER Performance	figure 3.3	figure 3.6
Complexity	figure 3.4	figure 3.7

Table 3.1 Summary of the SPC front-end simulations

3.4.1 Uncoded Framework

The uncoded framework SPC front-end simulations consider a system with $N_{TX} = 5$, $N_{RX} = 7$ with symbols drawn from a 16-QAM constellation. As explained in appendix A, these results are obtained by running the executable obtained by compiling the uncodedMIMO.cpp file. The system parameters as well as the range of SNRs can be adjusted in this program file. The SPC front-end parameter, U, influences the performance-complexity tradeoff of the system and so simulations are done for 4 informative values: $U = \{2.8, 3.0, 3.2, 3.6\}$.

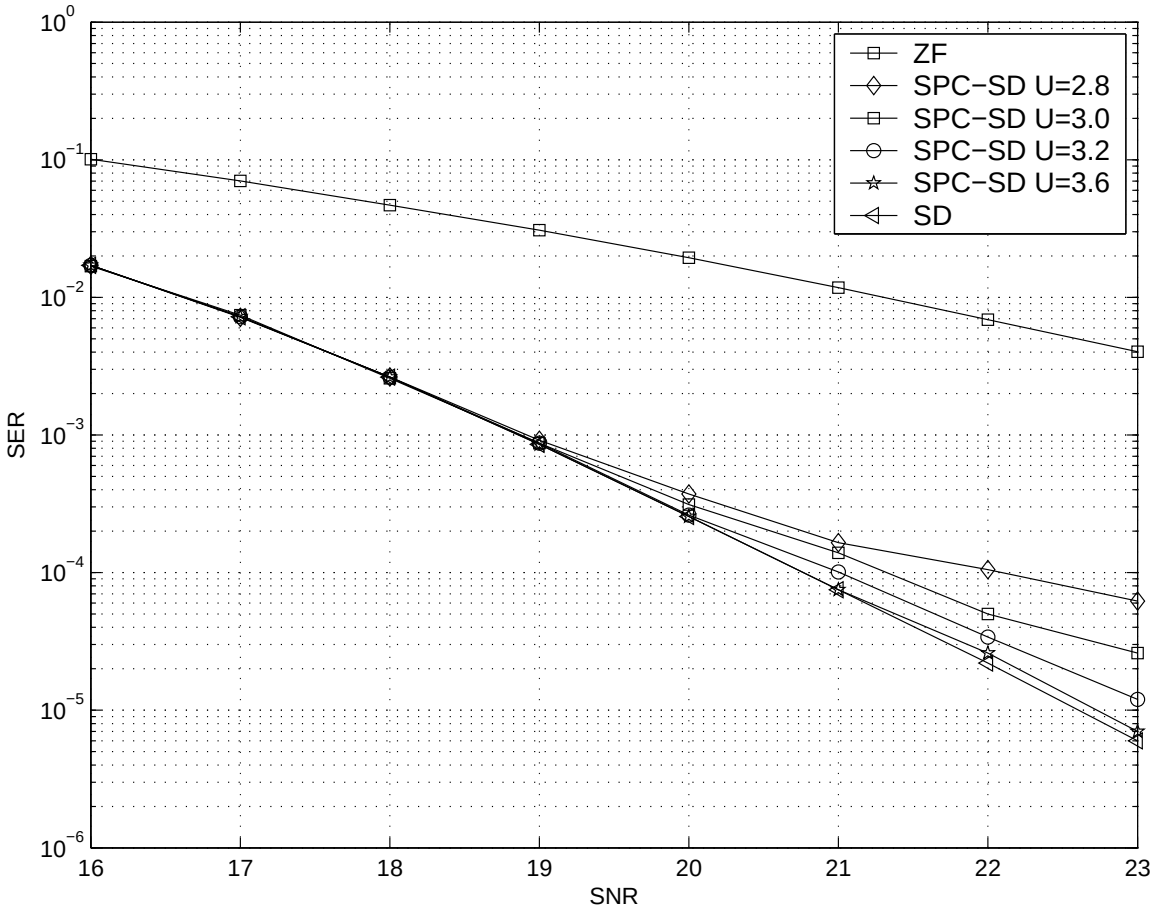

Fig. 3.2 Uncoded SPC-SD detector SER performance: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

In figure 3.2 it is seen that the detector achieving the best SER performance is the standalone SD followed by SPC-SD with parameter $U = 3.6$, SPC-SD $U = 3.2$,SPC-SD $U = 3.0$, SPC-SD $U = 2.8$, and the ZF detector. Since, among the three SPC-SD detectors, the SPC front-end with $U = 3.2$ has the largest P_D lower bound it has the best performance. For those curves with small U, SPC-SD has worse diversity than SD since not all transmitted symbols are jointly detected. The performance reduction of SPC-SD for small U is due to error propagation from the SPC front-end.

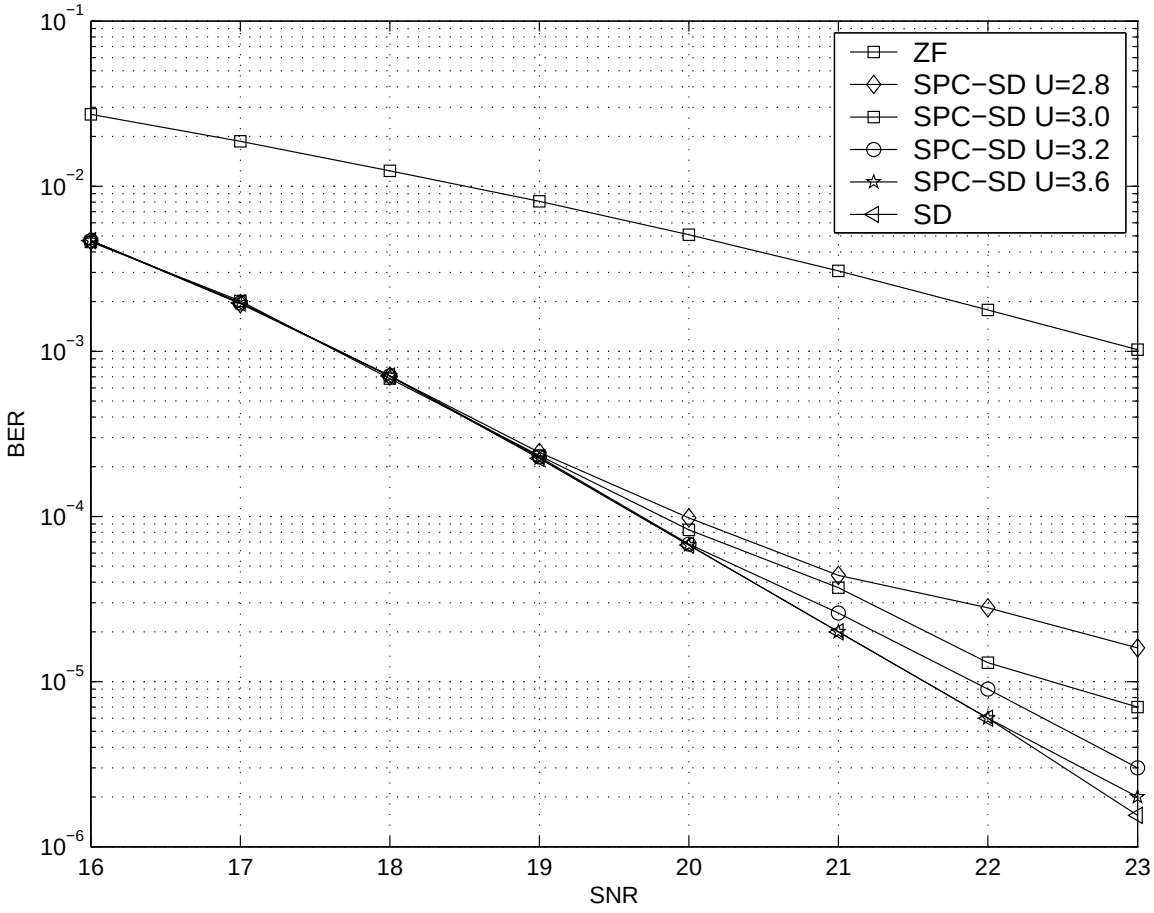

Fig. 3.3 Uncoded SPC-SD detector BER performance: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

The BER performance graph, figure 3.3, shows the same trends as the SER perfor-

mance graph. The BER performance of SD and SPC-SD have near identical performance below an SNR of 19dB for the given system. At lower SNR, SD errors dominate additional errors introduced by the SPC front-end. At higher SNR the performance of the SPC-SD detector is dominated by errors introduced by the SPC front-end and the performance becomes noticeably suboptimal.

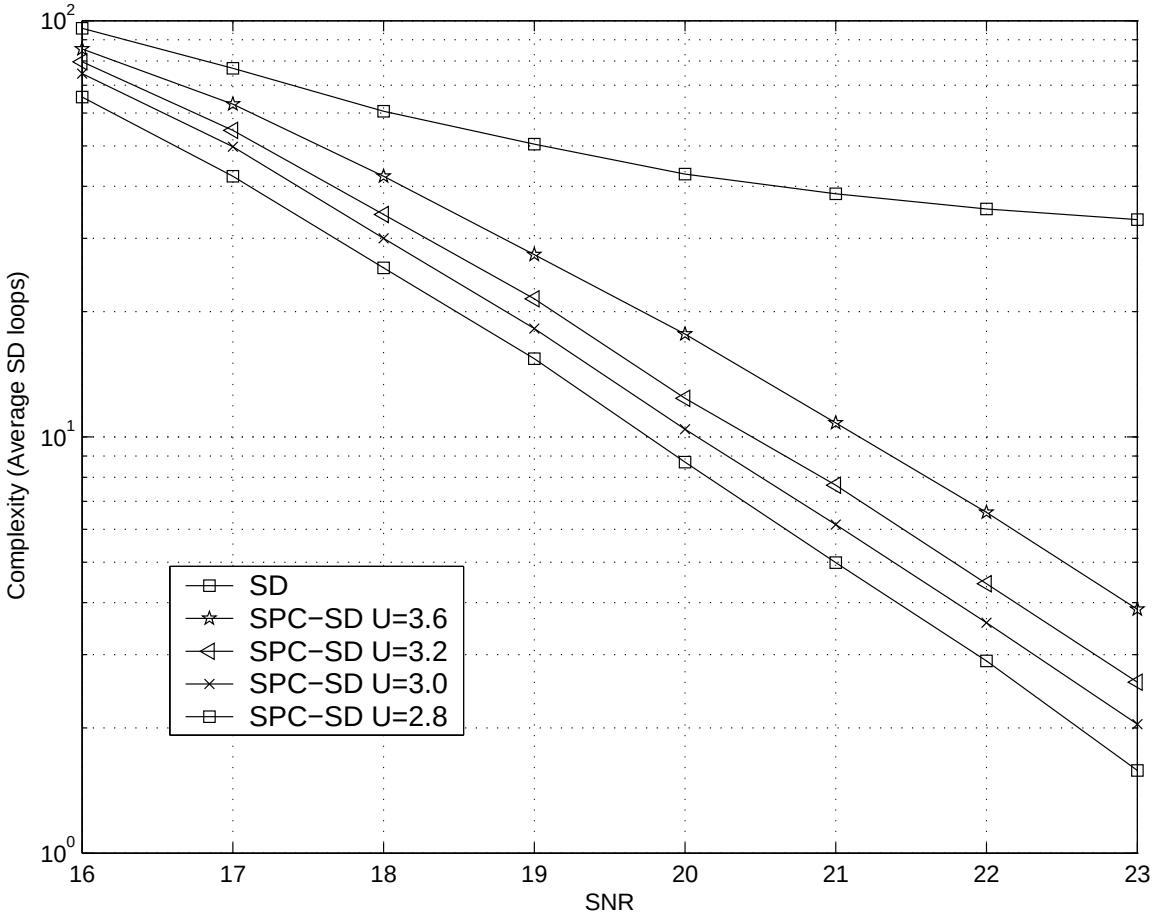

Fig. 3.4 Uncoded SPC-SD detector complexity: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

In figure 3.4 the SPC-SD detector is seen to have much less computational complexity than the standalone SD detector for all simulated values of U. The complexity advantage of SPC-SD over SD becomes greater the larger the SNR. An SPC-SD

detector with a smaller U has a smaller P_F and therefore a smaller computational complexity.

3.4.2 LD Coded Framework

The SPC front-end simulations for the LD coded framework consider a system with $N_{TX} = 2$, $N_{RX} = 2$ with symbols drawn from a QPSK constellation. The information symbols are encoded by the LD code described in section 2.6. As explained in appendix A, the coded results are obtained by running the executable obtained by compiling the LDcodedMIMO.cpp file. The system parameters as well as the range of SNRs can be adjusted in this program file. The SPC front-end parameter, U, influences the performance-complexity tradeoff of the system and so simulations are done for 3 informative values: $U = \{2.4, 2.6, 2.8\}$.

The performance and complexity trends of the decoder SPC-SD for the LD coded framework are similar to those for the uncoded framework. In figure 3.5, the SER performance of SPC-SD is shown. As analytically predicted by equation (3.26), an SPC-SD decoder with a larger U has better performance. As a comparison, figure 3.5 also shows the SER performance of uncoded transmission with the same system parameters ($N_{TX} = 2$,$N_{RX} = 2$,QPSK) at the same rate (R_c=R_u=4). The uncoded transmission detection is done using the SD to achieve ML performance. It is seen that the performance of the LD coded system, decoded with the SD, is better than the performance of the uncoded system with the plots matching the results published by Hassibi and Hochwald [10].

Figure 3.6 shows the BER performance of the same 3 SPC-SD decoders. The SPC-SD decoder with the largest U has the best performance but all three SPC-SD decoders are seen to have only marginally worse performance than the SD decoder.

When decoding LD coded symbols the simulation results, presented in figure 3.7,

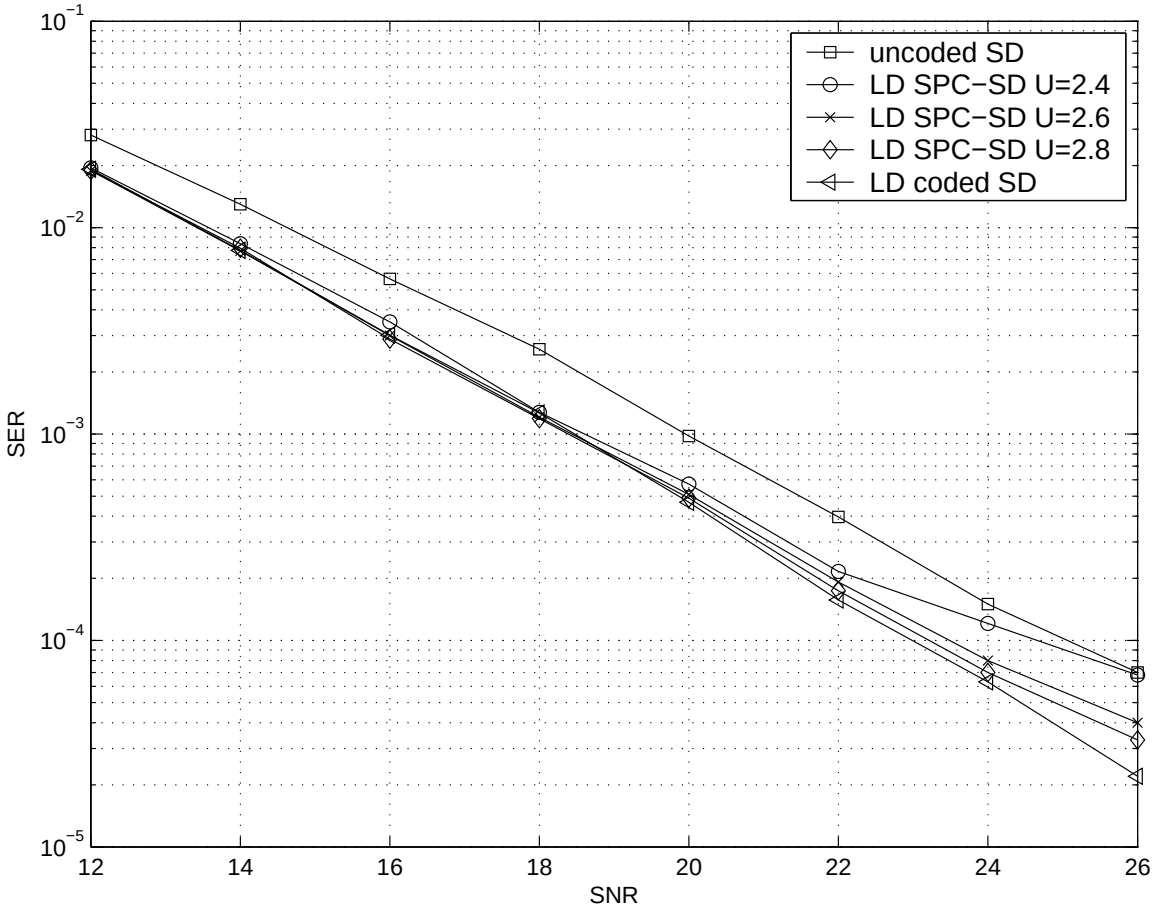

Fig. 3.5 LD coded SPC-SD decoder SER performance: QPSK, $N_{TX} = 2$, $N_{RX} = 2$

show that SPC-SD has lower computational complexity than SD. The complexity advantage of the two-stage decoder increases exponentially with the SNR. The relative difference in complexity between the various SPC-SD decoders is constant for all simulated SNRs. Furthermore, the difference in complexities for the various SPC-SD decoders is small compared to their difference from the complexity of the SD decoder.

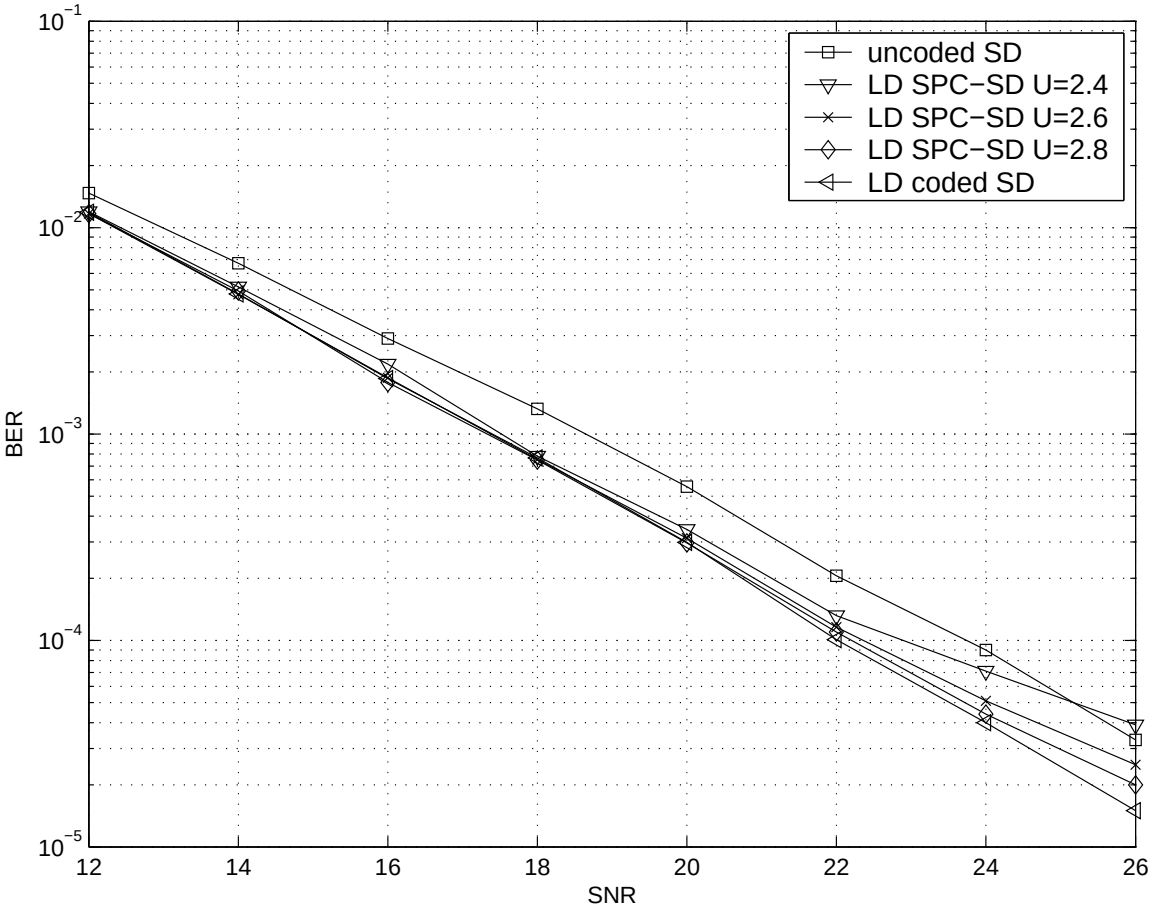

Fig. 3.6 LD coded SPC-SD decoder BER performance: QPSK, $N_{TX} = 2$, $N_{RX} = 2$

3.5 Comparison of the SPC Front-End to a Previously Published MIMO Detection Scheme

The SPC front-end is similar to a MIMO detection scheme recently proposed by Choi in [30]. Choi's basic scheme is called projection based maximum likelihood (PR-ML) and, unlike SPC, it isn't a front-end but rather a full detector with equal performance, in the case of uncoded transmission, as ZF detection [30].

The PR-ML detector, like the SPC front-end, operates on a symbol-by-symbol basis. Each symbol in the constellation has the following associated metric, where $\mathbf{S_i}$,

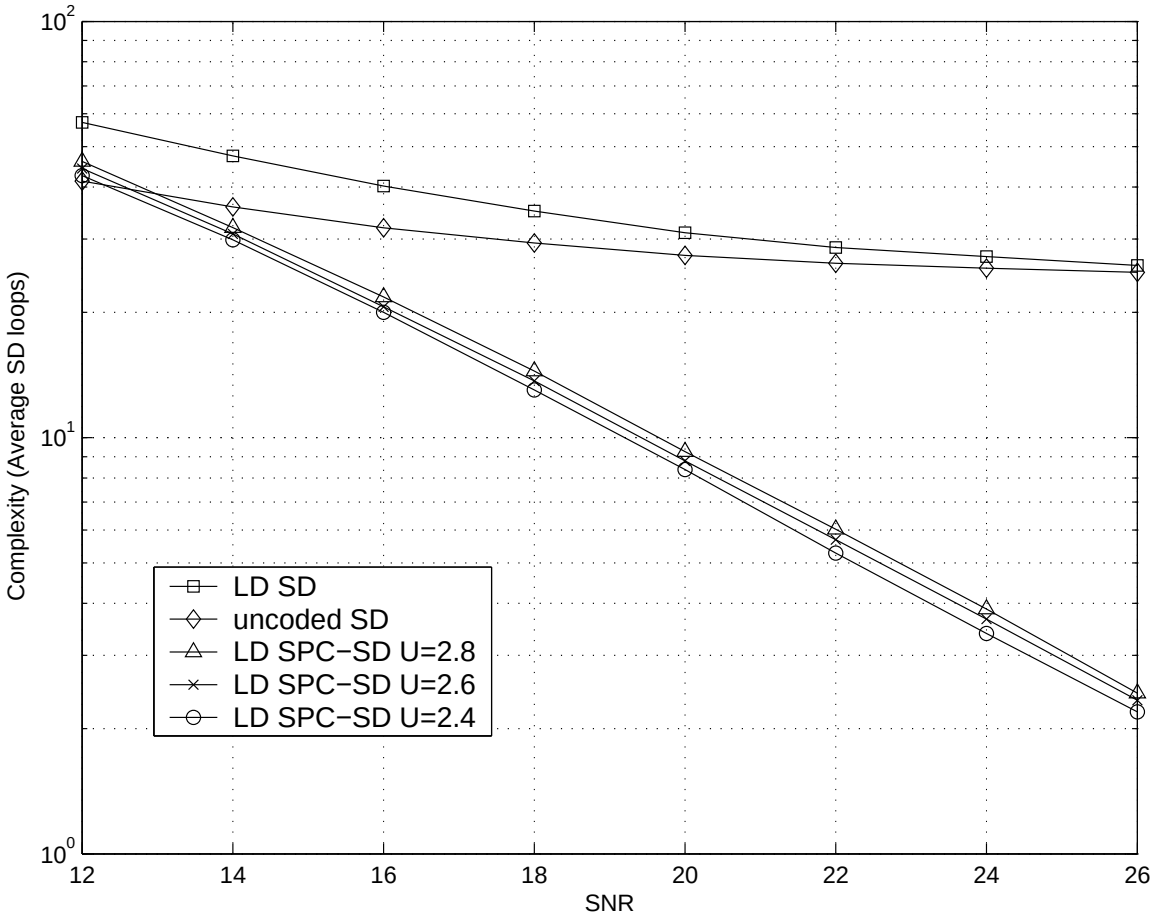

Fig. 3.7 LD coded SPC-SD decoder complexity: QPSK, $N_{TX} = 2$, $N_{RX} = 2$

$\mathbf{v_i}$, and $\mathbf{h_i}$ are defined as before, and $\mathbf{r}$ is the received vector.

$$\mathcal{M}_{PR}(\overline{u_i}) = \|P^{\perp}_{\mathbf{S_i}}(\mathbf{r} - \sqrt{\frac{\rho}{N_{TX}}}\mathbf{h_i}\overline{u_i})\|_2 \tag{3.40}$$

$$= \|P^{\perp}_{\mathbf{S_i}}(\sqrt{\frac{\rho}{N_{TX}}}\mathbf{Hu} + \mathbf{n} - \sqrt{\frac{\rho}{N_{TX}}}\mathbf{h_i}\overline{u_i})\|_2 \tag{3.41}$$

$$= \|\sqrt{\frac{\rho}{N_{TX}}}\mathbf{v_i}(u_i - \overline{u_i}) + P^{\perp}_{\mathbf{S_i}}\mathbf{n}\|_2. \tag{3.42}$$

$\mathcal{M}_{PR}(\overline{u_i})$ is computed for each $\bar{u}_i \in \mathcal{U}$, where $\mathcal{U}$ is the set of all symbols in the

constellation, so that the detected i^{th} symbol $u_{i\,PR}$ is:

$$u_{i\,PR} = \arg\min_{\overline{u_i} \in S} \mathcal{M}_{PR}(\overline{u_i}). \tag{3.43}$$

It has been proved, and it can be seen from the analysis in section 2.4.2, that $u_{i\,PR}$ is always equal to the i^{th} element of the output of the ZF detector, $\hat{u}_i$ [26].

Central to SPC and PR-ML respectively are the following projections.

$$\|P_{\mathbf{v_i}}(\mathbf{r} - \sqrt{\tfrac{\rho}{N_{TX}}}\mathbf{H}\hat{\mathbf{u}})\|_2 = \|\sqrt{\tfrac{\rho}{N_{TX}}}\mathbf{v_i}(u_i - \hat{u}_i) + P_{\mathbf{v_i}}\mathbf{n}\|_2. \tag{3.44}$$

$$\|P^{\perp}_{\mathbf{S_i}}(\mathbf{r} - \sqrt{\tfrac{\rho}{N_{TX}}}\mathbf{h_i}\overline{u_i})\|_2 = \|\sqrt{\tfrac{\rho}{N_{TX}}}\mathbf{v_i}(u_i - \overline{u_i}) + P^{\perp}_{\mathbf{S_i}}\mathbf{n}\|_2. \tag{3.45}$$

One difference between these two projections is the subspace onto which the noise is projected. The two subspaces, $\langle \mathbf{v_i} \rangle$ and $\langle \mathbf{S_i} \rangle^{\perp}$, are related by $\mathbf{v_i} = P^{\perp}_{\mathbf{S_i}}\mathbf{h_i}$. Besides including the noise in the subspace of $\mathbf{v_i}$, the PR-ML metric also includes the noise outside of the subspace of $\mathbf{H}$. The SPC front-end also allows for the pre-calculation of $\mathbf{w} = \mathbf{r} - \sqrt{\frac{\rho}{N_{TX}}}\mathbf{H}\hat{\mathbf{u}}$ which then only needs to be projected onto $\mathbf{v_i}$ for each dimension i.

Another difference between SPC and PR-ML is the way in which they are used. The SPC scheme is a complexity reducing front-end to the SD, however, the PR-ML scheme can be extended to the projection-based conditional ML (PBC-ML) scheme to improve on zero-forcing performance. The PBC-ML scheme is a recursive detection scheme where at each recursion a certain number of hard-decisions are made based upon a reliability measure. The reliability measure assumes that the information symbols were modulated using BPSK and therefore the PBC-ML can only detect BPSK symbols. In contrast, the SPC front-end can handle any squared QAM symbol constellation and is thus better adapted to higher rate applications. The metric used

for PBC-ML is the same as the one used for PR-ML except that those symbols deemed to have been previously detected are always subtracted out of the received vector. Letting $\mathbf{H_d}$ be a subset of the channel matrix $\mathbf{H}$ whose columns correspond to the decided symbols, $\mathbf{u_d}$, then the PBC-ML metric for the symbol $\overline{u_i}$ is:

$$\mathcal{M}_{PBC}(\overline{u_i}) = \|P^{\perp}_{\mathbf{C_i}}(\mathbf{r} - \sqrt{\tfrac{\rho}{N_{TX}}}\mathbf{H_d u_d} - \sqrt{\tfrac{\rho}{N_{TX}}}\mathbf{h_i}\overline{u_i})\|_2. \tag{3.46}$$

In this case the $\mathbf{C_i}$ subspace is the matrix of all columns of $\mathbf{H}$ except $\mathbf{h_i}$ and all of the columns of $\mathbf{H_d}$. The metric $\mathcal{M}_{PBC}(\overline{u_i})$ is minimized in the same way as was done for the PR-ML scheme in equation (3.43) and the performance improves the more correct, decided, symbols there are.

3.6 Channel Estimation Errors

The assumption that the receiver knows $\mathbf{H}$ perfectly is instrumental in the derivation of the ZF detector, the SD detector, and the SPC front-end. In practice, however, the receiver is required to estimate $\mathbf{H}$ and the estimation error will be non-zero. Figure 3.8 shows the performance impact of channel estimation errors on SPC-SD and SD.

In the simulations the detector uses an estimated channel, $\hat{\mathbf{H}}$, rather than the actual channel, $\mathbf{H}$.

$$\hat{\mathbf{H}} = \mathbf{H} + \mathbf{\Delta}. \tag{3.47}$$

Each element of the channel estimation error matrix, $\mathbf{\Delta}$, is modeled as independent and identically distributed real Gaussian random variables:

$$\Delta_{ij} \sim N[0, N_0/2], i \in [1, 2N_{RX}], j \in [1, 2N_{TX}]. \tag{3.48}$$

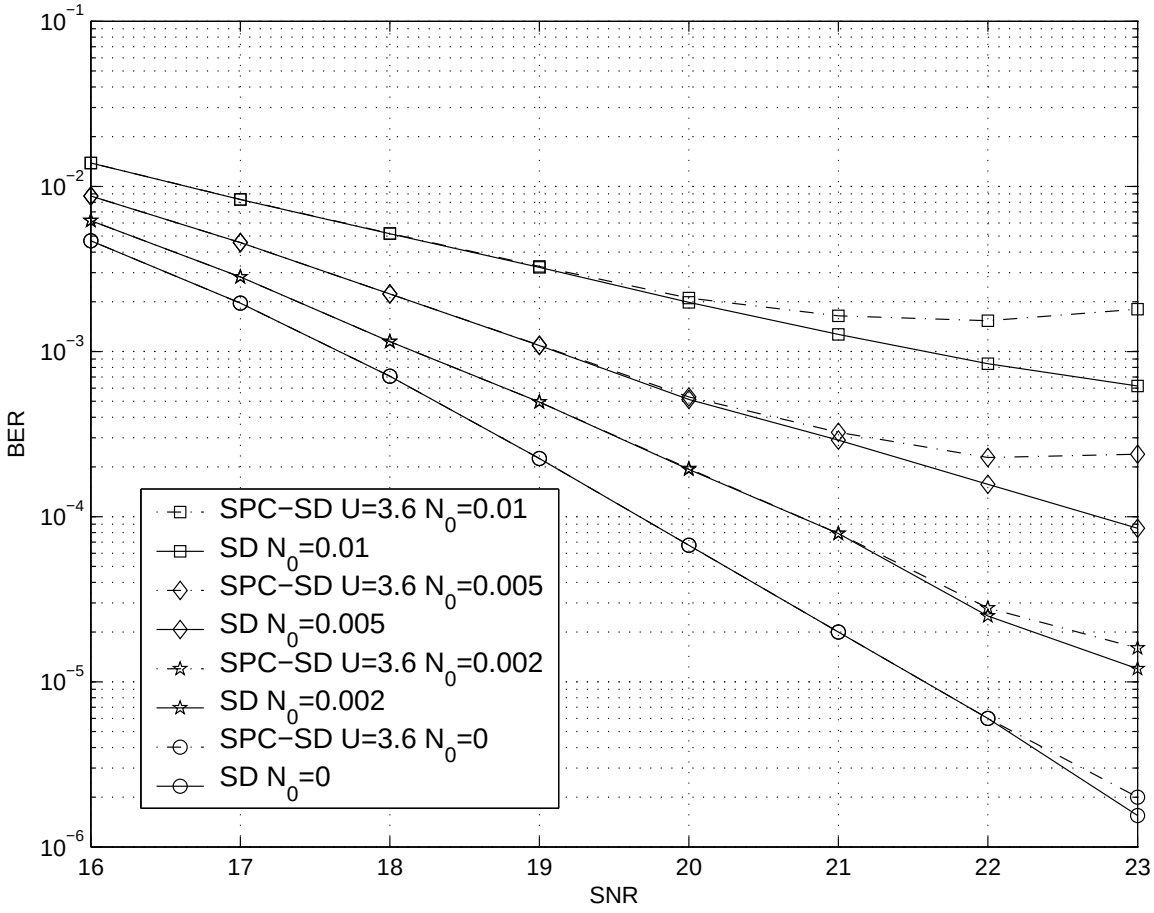

Fig. 3.8 SPC-SD detector BER performance comparison in the presence of channel estimation uncertainty: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

Figure 3.8 shows the impact of channel estimation errors for three values of the estimation error variance and when there are no errors. It is seen that the ML performance, obtained with the SD, is very dependent on an accurate estimation of $\mathbf{H}$. In fact, the introduction of a variance of 0.002 in the estimation of the channel matrix elements causes SD performance reduction of over 1 dB.

Figure 3.8 also shows that the SPC front-end performance is not especially dependent on the estimation uncertainty. The two-stage SPC-SD detector performs nearly as well as the SD detector even as channel estimation uncertainty is introduced.

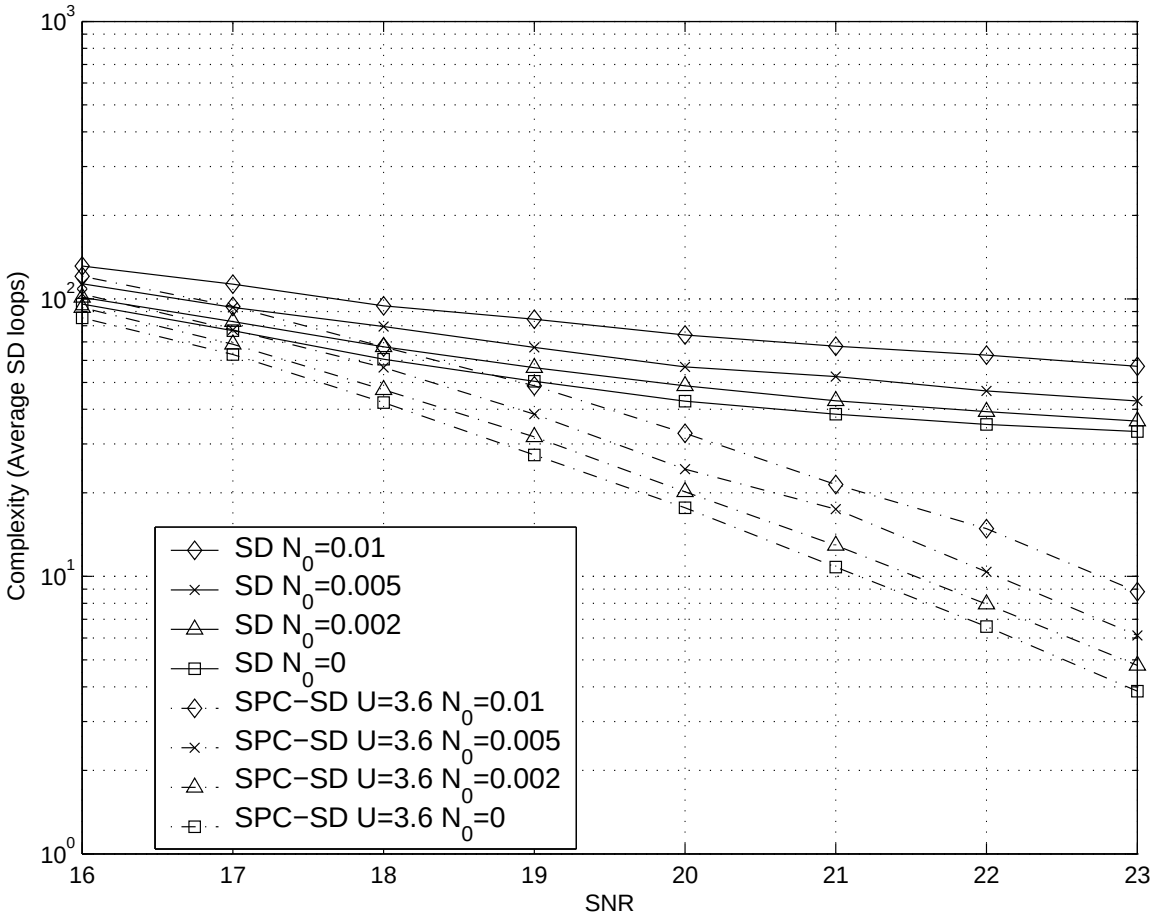

Fig. 3.9 SPC-SD detector complexity comparison in the presence of channel estimation uncertainty: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

Figure 3.9 shows the complexity of SPC-SD and SD for the same channel estimation error variances. It is seen that channel estimation uncertainty does not influence the complexity reducing potential of SPC-SD. Increasing estimation errors causes increased complexity SD and SPC-SD but does not change the relation between the two.

Chapter 4

Statistical Full Coverage (SFC) Front-End

A detector combining the SFC front-end with the SD (SFC-SD) reduces computational complexity compared to the standalone SD detector. The lower computational complexity as compared to the SD comes from only needing to detect a percentage of the symbol vectors with the SD. For the remaining symbol vectors the output of the ZF detector is used. The performance of the SFC-SD detector is kept close to the performance of the SD by ensuring that only symbol vectors very likely to have been correctly detected by the ZF detector are chosen. The steps in the SFC front-end are:

1. Obtain the output of the ZF detector ($\hat{\mathbf{u}}$ from Section 2.4).

2. Obtain the output of a decision feedback step. This step is identical to the SPC front-end's decision feedback step (Section 3.1) and calculates $\mathbf{w}$.

3. Perform a hypothesis test to see whether or not to detect the entire transmitted symbol vector, $\mathbf{u}$, sent in one channel-use from the N_{TX} transmit antennas, with the SD.

4.1 Hypothesis Test

The SFC hypothesis test can be formulated as follows:

$$\begin{aligned} H_0 : \hat{\mathbf{u}} = \mathbf{u} \rightarrow \mathbf{e} = \mathbf{0} \rightarrow \mathbf{w} : \mathcal{N}[0, \mathbf{I}] \quad vs \\ H_1 : \hat{\mathbf{u}} \neq \mathbf{u} \rightarrow \mathbf{e} \neq \mathbf{0} \rightarrow \mathbf{w} : \mathcal{N}[\mathbf{He}, \mathbf{I}]. \end{aligned} \tag{4.1}$$

This is a composite hypothesis test since $\mathbf{e}$ can assume multiple values under H_1. Deciding on H_0 results in $\hat{\mathbf{u}}$ being chosen as the output of the SFC-SD detector. Otherwise, the SD is used to detect this entire vector of transmitted symbols.

Maximum Likelihood Estimates (MLE)

We use the notational convention that the MLE of a particular variable will have the same variable name with the addition of a hat. The particular hypothesis being assumed will be indicated by a superscript.

The MLE of $\mathbf{e}$ under H_1, $\hat{\mathbf{e}}^{\mathbf{1}}$, can be obtained using the orthogonal projection matrix $\mathbf{P_H}$.

$$\begin{aligned} \mathbf{H}\hat{\mathbf{e}}^{\mathbf{1}} &= \mathbf{P_H w} \\ \hat{\mathbf{e}}^{\mathbf{1}} &= (\mathbf{H^T H})^{-1}\mathbf{H^T w}. \end{aligned} \tag{4.2}$$

The MLE of $\mathbf{e}$ under H_0 is $\hat{\mathbf{e}}^{\mathbf{0}} = \mathbf{0}$.

4.1.1 Generalized Likelihood Ratio (GLR)

Following the same reasoning as for the SPC front-end in Section 3.2.2 the GLR, $\hat{\Lambda}_{10}(\mathbf{w})$, can be obtained.

$$\begin{aligned}\hat{\Lambda}_{10}(\mathbf{w}) &= \frac{(2\pi)^{-N_{TX}}\exp\{-\frac{1}{2}\|\mathbf{w}-\mathbf{H}\hat{e}^1\|_2^2\}}{(2\pi)^{-N_{TX}}\exp\{-\frac{1}{2}\|\mathbf{w}-\mathbf{H}\hat{e}^0\|_2^2\}} \\ &= \exp\{\frac{1}{2}\|\mathbf{w}\|_2^2-\frac{1}{2}\|\mathbf{w}-\mathbf{P_H}\mathbf{w}\|_2^2\} \\ &= \exp\{\frac{1}{2}\|\mathbf{w}\|_2^2-\frac{1}{2}\|[\mathbf{I}-\mathbf{P_H}]\mathbf{w}\|_2^2\} \\ &= \exp\{\frac{1}{2}\|\mathbf{w}\|_2^2-\frac{1}{2}\|\mathbf{P_H^\perp}\mathbf{w}\|_2^2\}. \end{aligned} \tag{4.3}$$

The GLR can be simplified further by replacing it by a scaled logarithmic GLR [21].

$$\begin{aligned}L(\mathbf{w}) &= 2\ln\Lambda_{10}(\mathbf{w}) \\ &= \|\mathbf{w}\|_2^2-\|\mathbf{P_H^\perp}\mathbf{w}\|_2^2 \\ &= \mathbf{w^T}\mathbf{w}-\mathbf{w^T}\mathbf{P_H^\perp}\mathbf{w} \\ &= \mathbf{w^T}\mathbf{I}\mathbf{w}-\mathbf{w^T}\mathbf{P_H^\perp}\mathbf{w} \\ &= \mathbf{w^T}(\mathbf{I}-\mathbf{P_H^\perp})\mathbf{w} \\ &= \mathbf{w^T}\mathbf{P_H}\mathbf{w} \\ &= \|\mathbf{P_H}\mathbf{w}\|_2^2. \end{aligned} \tag{4.4}$$

The SFC front-end does its hypothesis test using the scaled logarithmic GLR, $L(\mathbf{w})$.

$$L(\mathbf{w}) \begin{array}{c} H_0 \\ < \\ \geq \\ H_1 \end{array} T^2. \tag{4.5}$$

The distribution of $\mathbf{w}$ is $\mathcal{N}[\mathbf{He}, \mathbf{I}]$ and so the distribution of $L(\mathbf{w})$ is chi-squared with nonconcentrality parameter λ^2 and N_{TX} degrees of freedom.

$$L(\mathbf{w}) : \chi^2_{N_{TX}}(\lambda^2). \tag{4.6}$$

$$\begin{aligned} \lambda^2 &= (\mathbf{He})^T(\mathbf{He}) \\ &= \mathbf{e^T H^T He}. \end{aligned} \tag{4.7}$$

For the SFC front-end, we define the false alarm (P_F) and detection (P_D) probabilities as follows.

$$P_F = P(H_1|H_0) = P(L(\mathbf{w}) \geq T^2|H_0) = P[\chi^2_{N_{TX}}(0) \geq T^2]. \tag{4.8}$$

$$P_D = P(H_1|H_1) = P(L(\mathbf{w}) \geq T^2|H_1) = P[\chi^2_{N_{TX}}(\lambda^2) \geq T^2]. \tag{4.9}$$

The P_D probability is an indication of the SFC-SD's reduction from ML performance. If $P_D = 1$ then the SFC-SD detector has ML performance. If $P_D < 1$ then the SFC-SD detector will have worse then ML performance. The P_F probability is an indication complexity reduction of the SFC-SD detector compared to the SD detector. If $P_F = 1$ then the SFC-SD has the same complexity as the SD since every transmitted symbol vector is detected by the SD. A low P_F results in a large complexity reduction since a larger percentage of symbol vectors that were correctly detected by the ZF detector are not also detected the SD.

4.1.2 Selection of the GLR Threshold, T

The decision threshold, T, can be selected using either a performance or complexity target. If a certain performance with respect to ML performance is desired than the P_D equation is set equal to a value close to one and and this equation is solved for T. If a certain complexity reduction target is desired then the P_F equation can be set equal to a value lower than one and then solved for T. Either of these equations can be solved with the use of a chi-squared distribution lookup table.

4.2 Simulation of the SFC Front-End

4.2.1 Uncoded Framework

The simulations of the two-stage SFC-SD detector for an uncoded framework consider a 5 transmit antenna and 7 receive antenna system. The symbols transmitted are chosen from a 16-QAM symbol constellation.

The SER performance of the combined SFC-SD detector for four distinct values of T is shown in figure 4.1. The SD detector and the ZF detector performance curves are also included for comparison. The linear ZF detector has the worst performance and the SD detector with ML performance has the best performance. Of the two-stage decoders, the best performance is achieved when the threshold, T, is low, and most of the frames are detected by the SD. For a low T the performance of the SFC-SD detector is nearly identical with the SD detector. As T is increased the performance of the SFC-SD detector becomes marginally worse than the SD detector. The reduction in performance of the SFC-SD detector is independent of SNR since the GLR threshold is independent of SNR. The BER performance graph, figure 4.2, exhibits the same trends as the SER performance graph. The performance slope of the SFC-SD detector

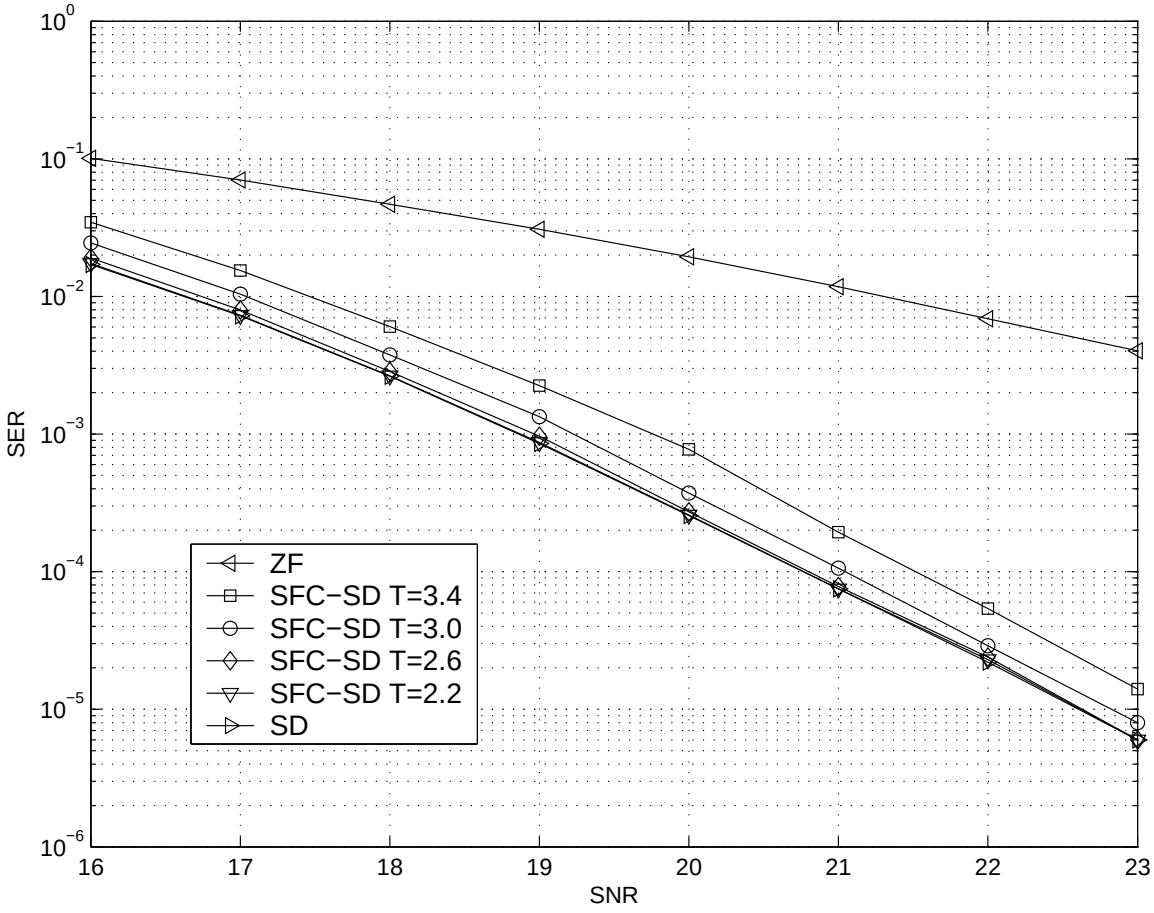

Fig. 4.1 Uncoded SFC-SD detector SER performance: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

is seen to be nearly identical to that of SD detector whereas the performance slope of the ZF detector is less steep.

In figure 4.3, the complexity of these detectors is plotted in terms of the average number of nodes visited in the SD search tree. The single-stage SD detector has the highest complexity followed by the two-stage SFC-SD detector with the lowest value of T. The lowest complexity is achieved by the SFC-SD detector with the largest value of T. A larger T ensures a greater complexity reduction since, in this case, fewer frames are required to be detected by the SD.

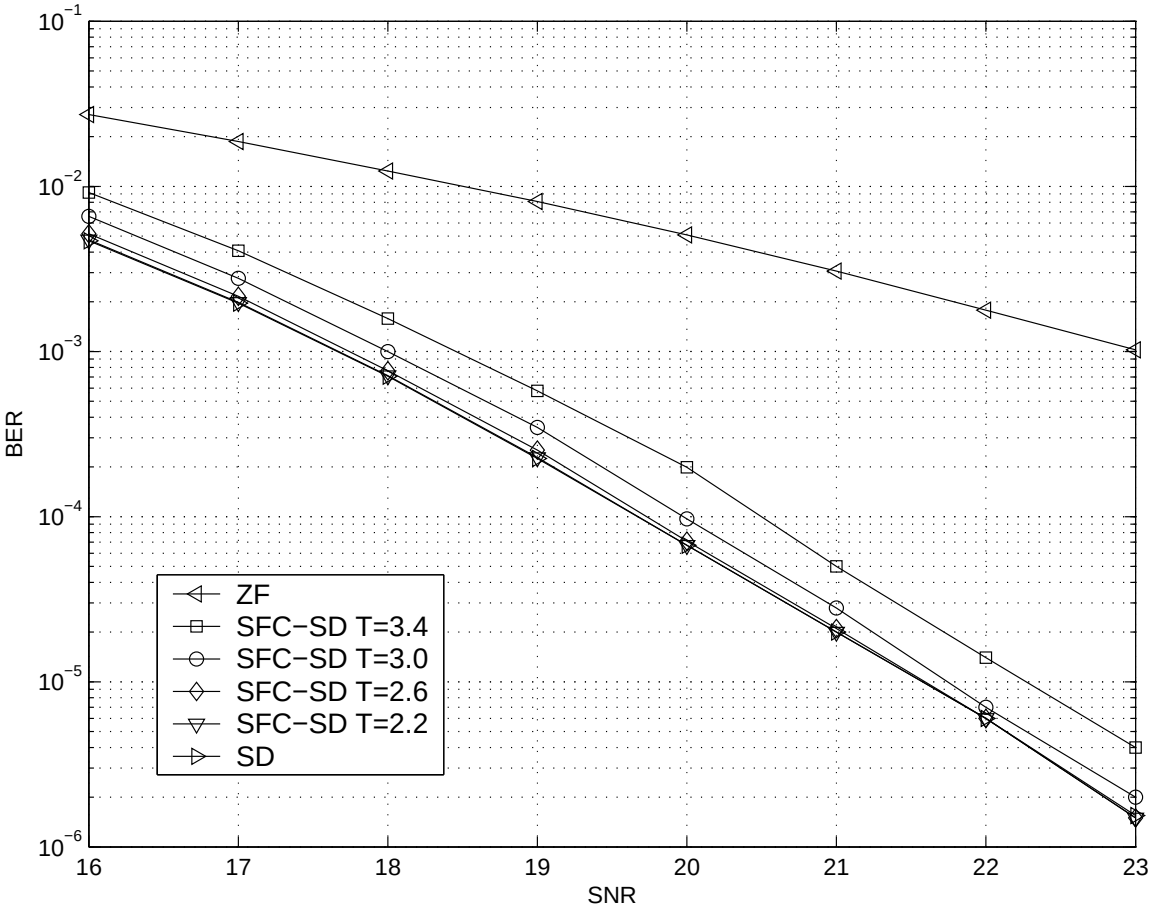

Fig. 4.2 Uncoded SFC-SD detector BER performance: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

The SFC-SD detector results, taken as a whole, show that by adjusting the threshold, T, we can adjust the fundamental tradeoff that exists between performance and complexity. For the uncoded framework it is possible, using the two-stage decoder, to achieve a meaningful complexity reduction with only a marginal reduction in performance.

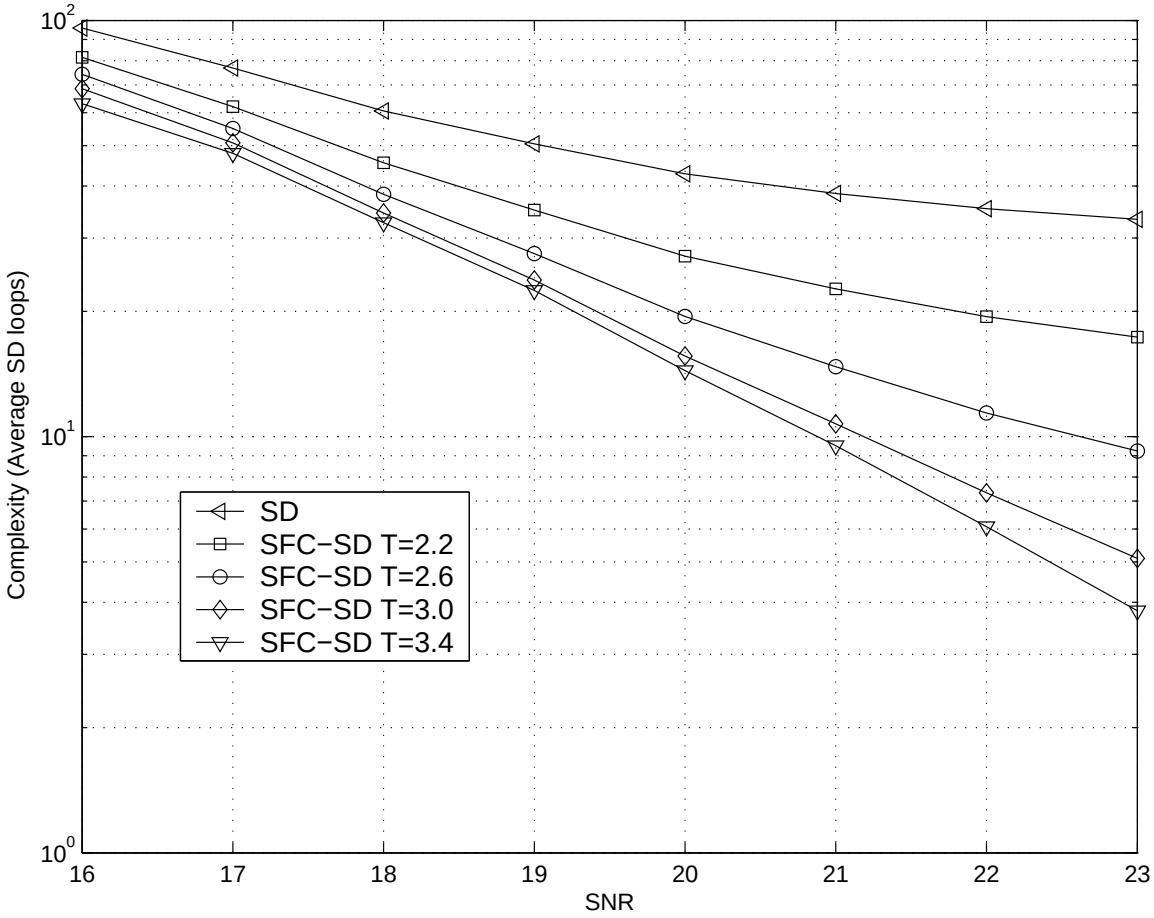

Fig. 4.3 Uncoded SFC-SD detector complexity: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

4.2.2 LD Coded Framework

Computer simulations of the LD coded framework and the two-stage SFC-SD decoder show all of the same trends as were described for the uncoded framework. Our computer simulations are of a system with 2 transmit antennas and 2 receive antennas where the symbols are taken from a QPSK constellation. The information symbols are encoded by the LD code described in section 2.6. The results of the simulations of this coded system, in terms of both performance and complexity, show the same trends as the uncoded system.

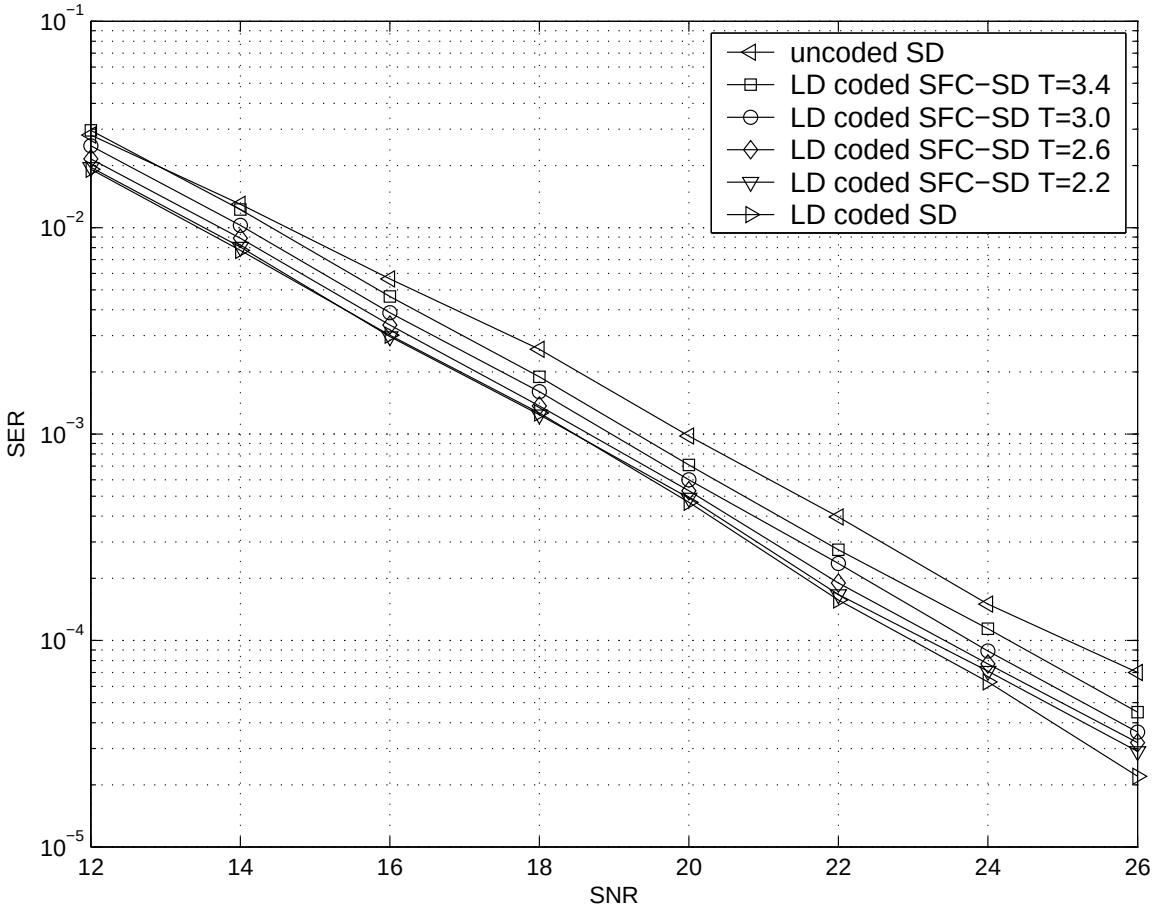

Fig. 4.4 LD coded SFC-SD decoder SER performance: QPSK, $N_{TX} = 2$, $N_{RX} = 2$

The SER performance of SFC-SD for four values of T is shown in figure 4.4. For comparison, the SER performance of an uncoded system at the same rate is also included. The rate of the coded system is found using equation (2.5) and, for this particular system, $R_c = 4$. To achieve $R_u = 4$ the uncoded system also transmits QPSK symbols. It can be seen that the LD coded system has better performance than the uncoded system but has only a slightly steeper slope. The BER performance graph, figure 4.5, exhibits the same trends as the SER performance graph.

In figure 4.6, the complexity of the two-stage SFC-SD decoder is seen to be less

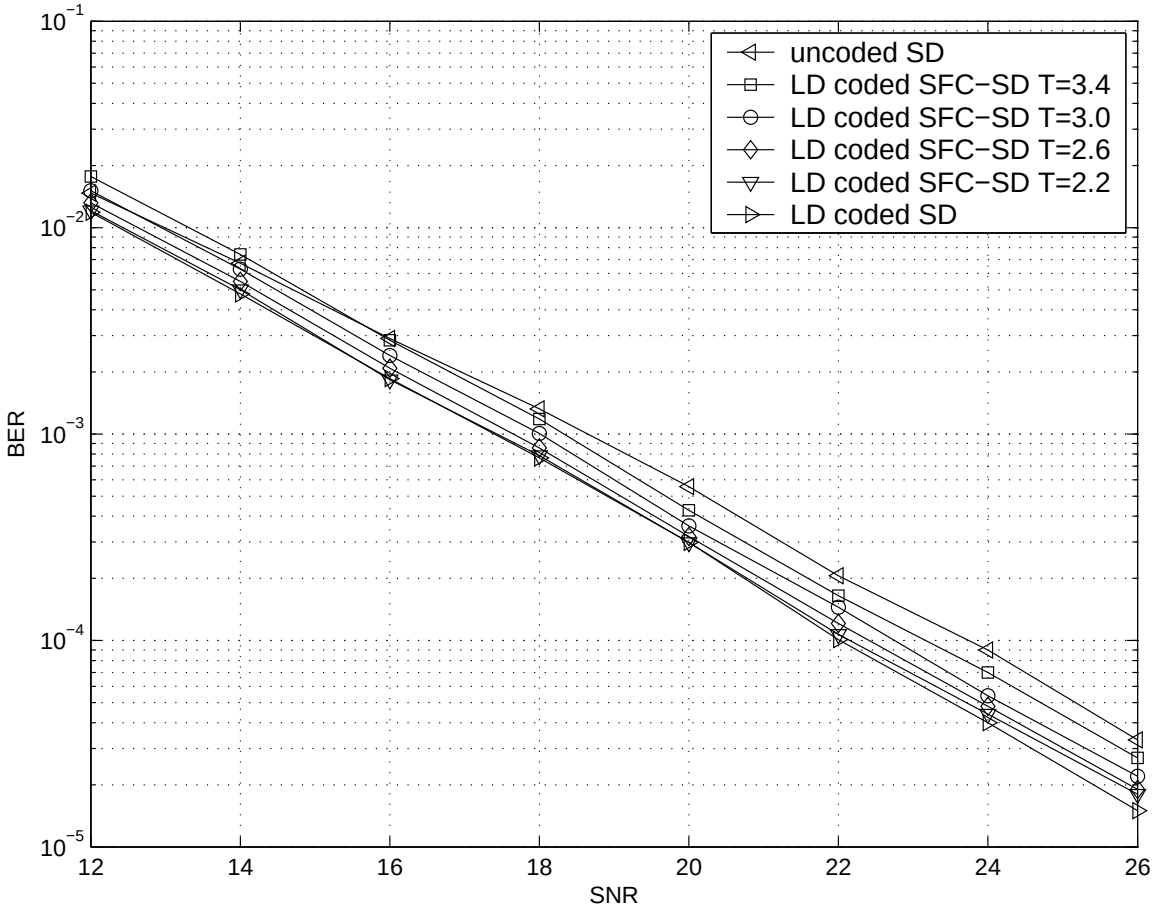

Fig. 4.5 LD coded SFC-SD decoder BER performance: QPSK, $N_{TX} = 2$, $N_{RX} = 2$

than the complexity of the single-stage decoder. The complexity is lowest when the parameter, T, is largest, and SD is done the least. The reduction in complexity achieved by increasing T becomes less however as T becomes larger. For a small T where the performance of the associated two-stage decoder is very close to ML a significant complexity reduction is still achieved.

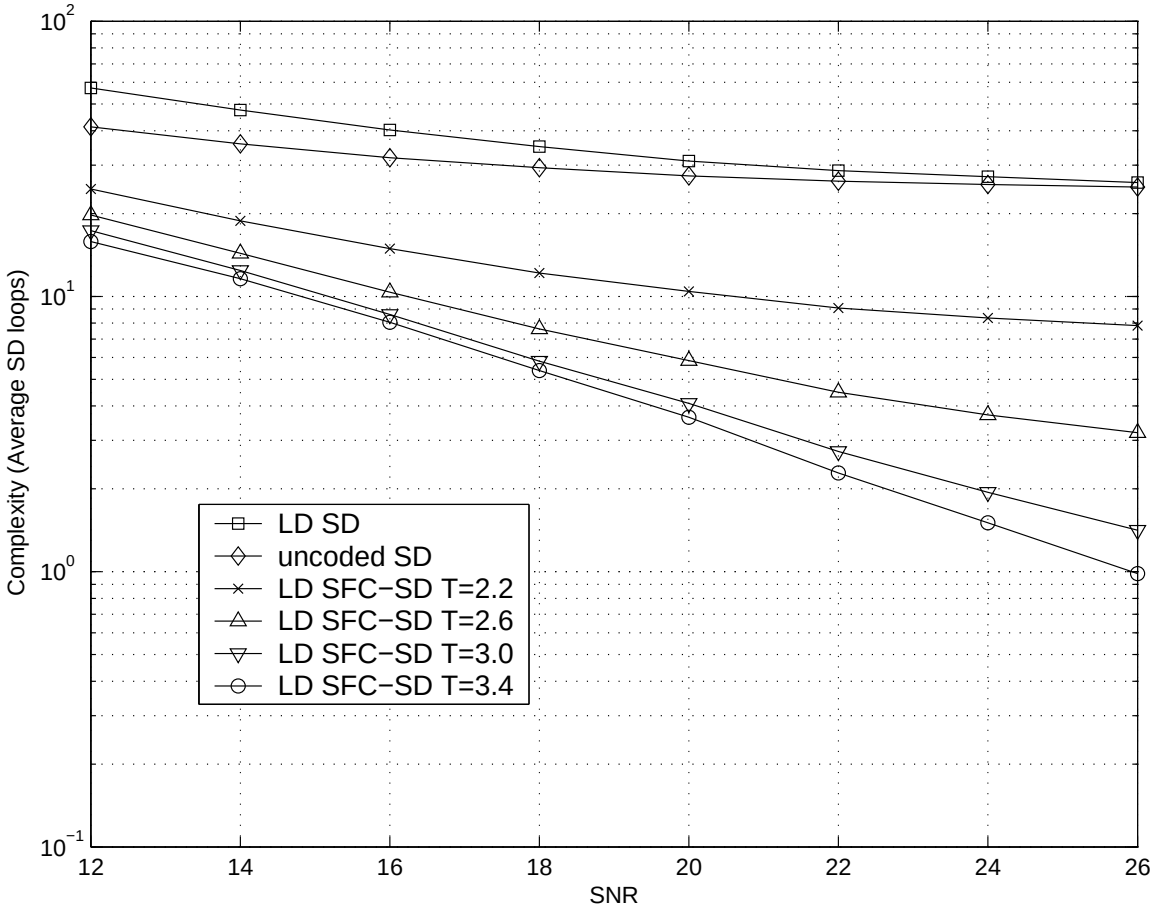

Fig. 4.6 LD coded SFC-SD decoder complexity: QPSK, $N_{TX} = 2$, $N_{RX} = 2$

4.3 Channel Estimation Errors

The assumption that the receiver knows **H** perfectly is instrumental in the derivation of the ZF detector, the SD detector, and the SFC front-end. In practice, however, the receiver is required to estimate **H** and the estimation error will be non-zero. The channel estimation errors are modeled as additive real Gaussian random variables and is further described in Chapter 3 using Equation (3.47) and Equation (3.48).

Figure 4.7 shows the impact of channel estimation errors for three values of the estimation error variance and when there are no errors. Figure 4.8 shows the complex-

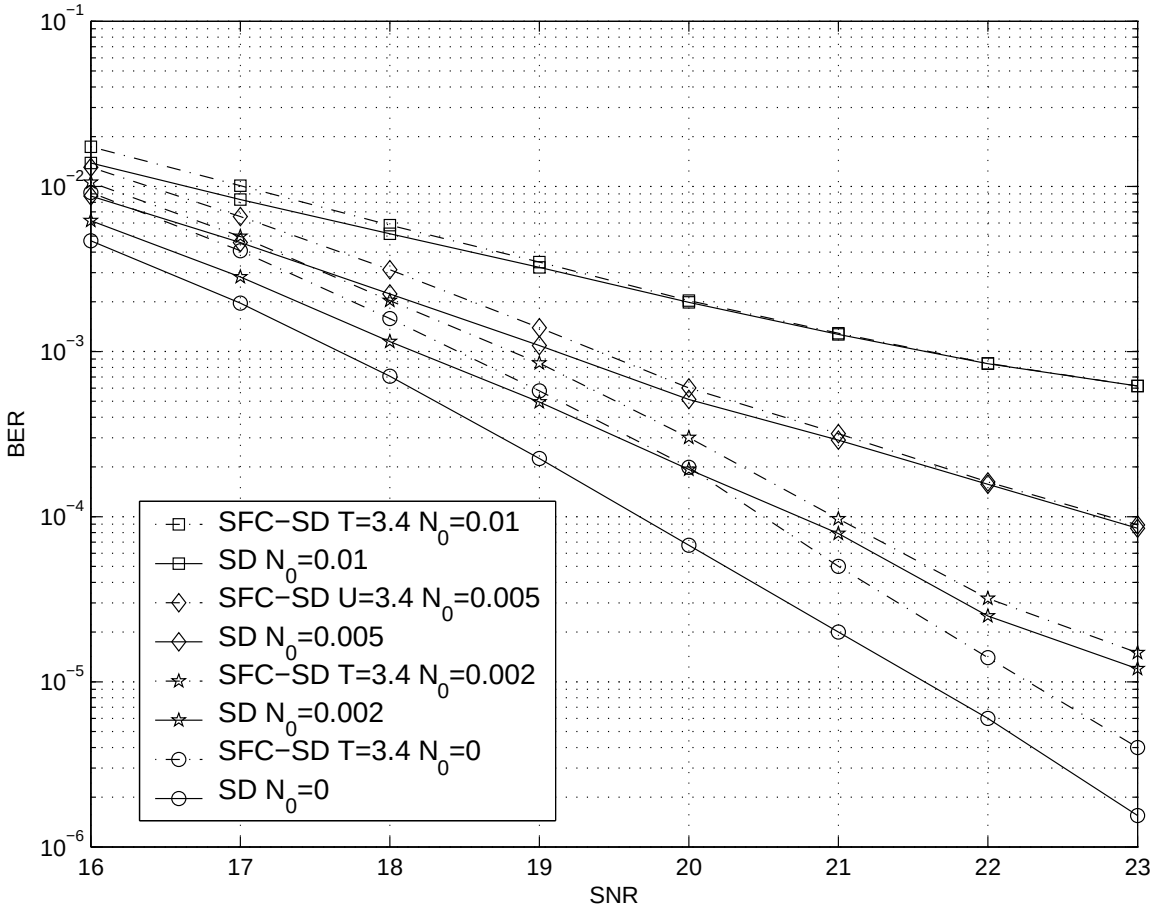

Fig. 4.7 SFC-SD detector BER performance comparison in the presence of channel estimation uncertainty: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

ity of SD and SFC-SD for the same estimation errors. Both of these plots are done keeping the SFC parameter $T = 3.4$ constant so as to emphasize the performance-complexity tradeoff. In Figure 4.7 we see that as the estimation variance increases the performance of both SD and SFC-SD decreases. However, the difference in performance between SD and SFC-SD also decreases as the estimation variance increases. Figure 4.7 shows that the SFC front-end performance can still be good in the presence of estimation uncertainty.

Figure 4.8 shows the complexity of SFC-SD and SD for the same channel estima-

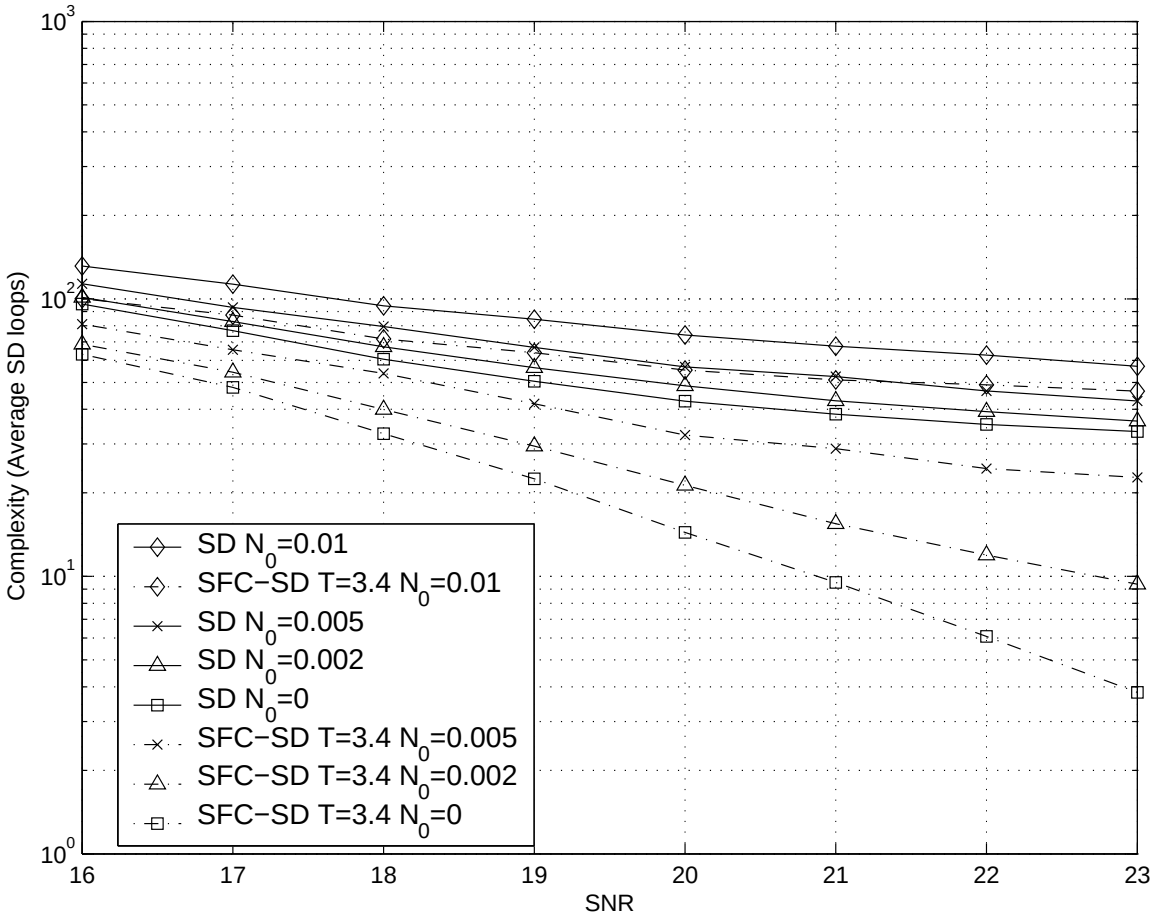

Fig. 4.8 SFC-SD detector complexity comparison in the presence of channel estimation uncertainty: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

tion errors. By looking at Figure 4.7 and Figure 4.8 we see that, even with channel estimation errors, the complexity-performance tradeoff remains. For example, when $T = 3.4$, with $N_0 = 0.01$ the performance of SFC-SD and SD are nearly identical but the complexity reduction is very small; however, with $N_0 = 0.002$ the performance of SFC-SD is worse than SD but the complexity reduction of SFC-SD is more significant.

Chapter 5

Detector Combinations and Comparisons

In this chapter we compare all of the MIMO detection schemes previously discussed and some promising combinations thereof. These comparisons are done using computer simulations for both the uncoded and LD coded frameworks. Both error rate performance and computational complexity are considered to emphasize how a tradeoff can be achieved.

It is shown that SE/SD can be combined with either the SFC front-end or the SPC front-end. Indeed, it is seen that any two, or even all three, of these complexity reducing techniques can be combined to even further reduce computational complexity with near ML performance.

5.1 Complexity Reducing Detector Combinations

In the last two chapters SPC and SFC front-ends were derived, analyzed, and simulated. It was seen that when either of these front-ends is appended to the SD the

combined detector is able to reduce computational complexity as compared to the standalone SD at very little cost to performance. It should not be surprising then that the same is true when either of these front-ends are appended to SE/SD. The SE/SD has the same inputs and outputs as the SD. Most importantly, thought, it has the exact same performance as the SD but lower complexity. Therefore, in order to achieve the lowest complexity detector possible, it makes sense to always append either the SPC front-end or the SFC front-end to the SE/SD rather than to the SD.

It is also possible to combine the SFC front-end, the SPC front-end, and the SE/SD to form what will be called the SFC-SPC-SE/SD detector. The SFC-SPC-SE/SD detector is a serial concatenation of first the SFC front-end, which attempts to pre-detect the entire symbol vector, then the SPC front-end, which pre-detects as many individual symbols as possible, and finally the SE/SD, which jointly detects the remaining symbols in the symbol vector. The block diagram, figure 5.1 below, shows how these three detection techniques are serially concatenated.

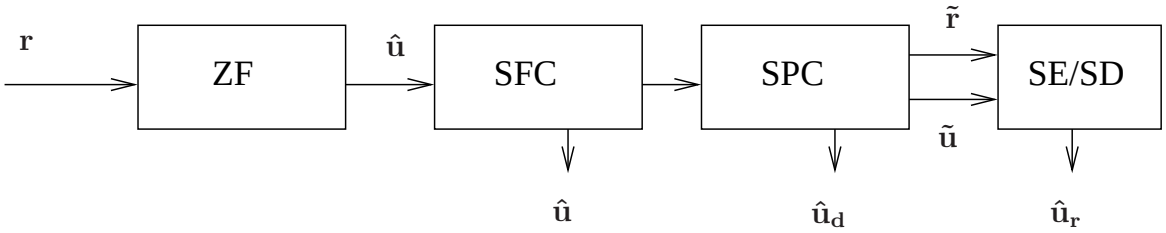

Fig. 5.1 Block diagram of SFC-SPC-SE/SD

The output of the SFC-SPC-SE/SD detector is either $\hat{\mathbf{u}}$ or the combination of $\hat{\mathbf{u}}_\mathbf{d}$ and $\hat{\mathbf{u}}_\mathbf{r}$. The SFC and SPC front-ends function in exactly the same way as they were described in the previous two chapters. The SFC-SPC-SE/SD detector has two parameters which affect its performance and computational complexity: T (for the SFC front-end) and U (for the SPC front-end).

5.2 Uncoded Detector Comparison: 16-QAM, $N_{TX} = 5$ and $N_{RX} = 7$

The detector comparison for the uncoded framework considers the same $N_{TX} = 5$ $N_{RX} = 7$ and 16-QAM system as was considered in the SFC and SPC front-end simulations. We present the results for certain parameters (T for SFC and U for SPC) that achieve near-ML performance.

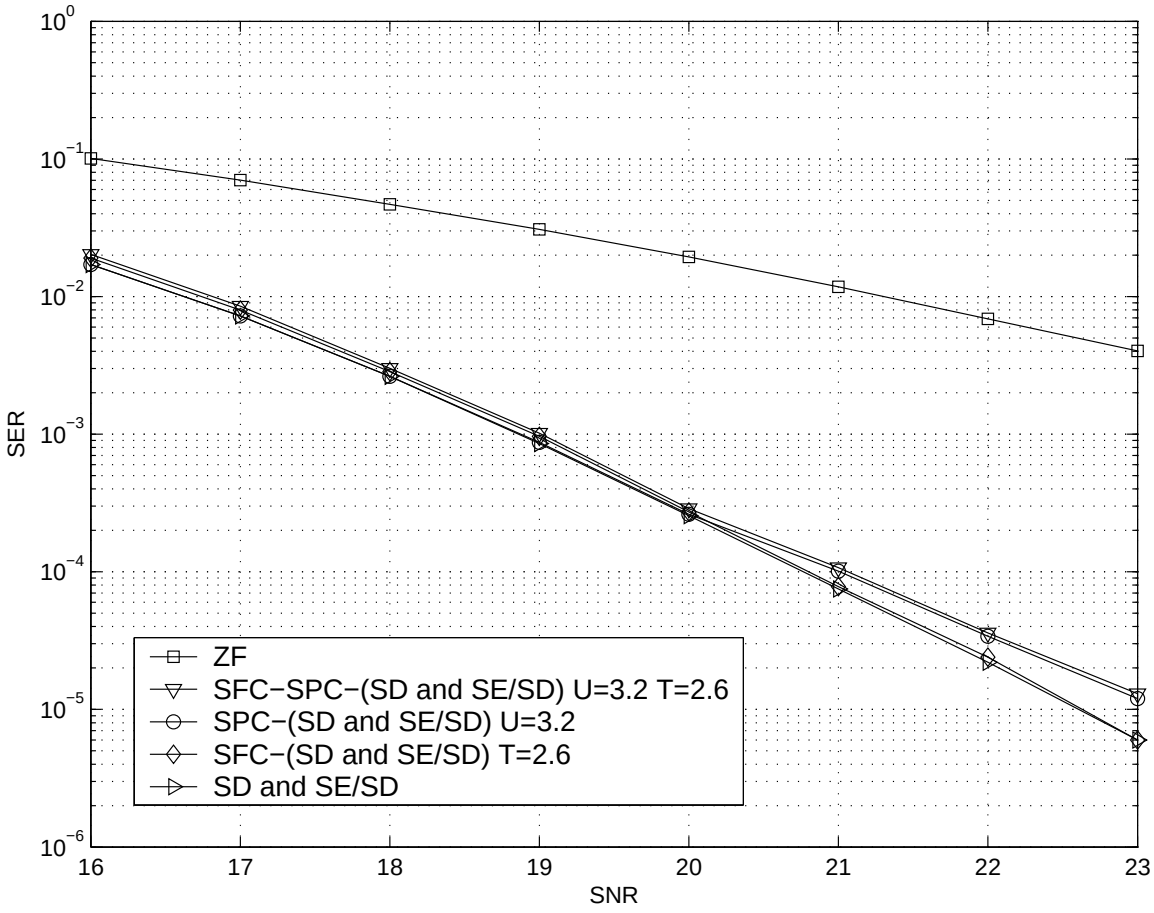

Fig. 5.2 Uncoded detector SER performance comparison: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

In terms of SER performance figure 5.2 shows that both two-stage detectors, SPC-SD and SFC-SD, have near-ML performance. In the range of simulated SNRs both

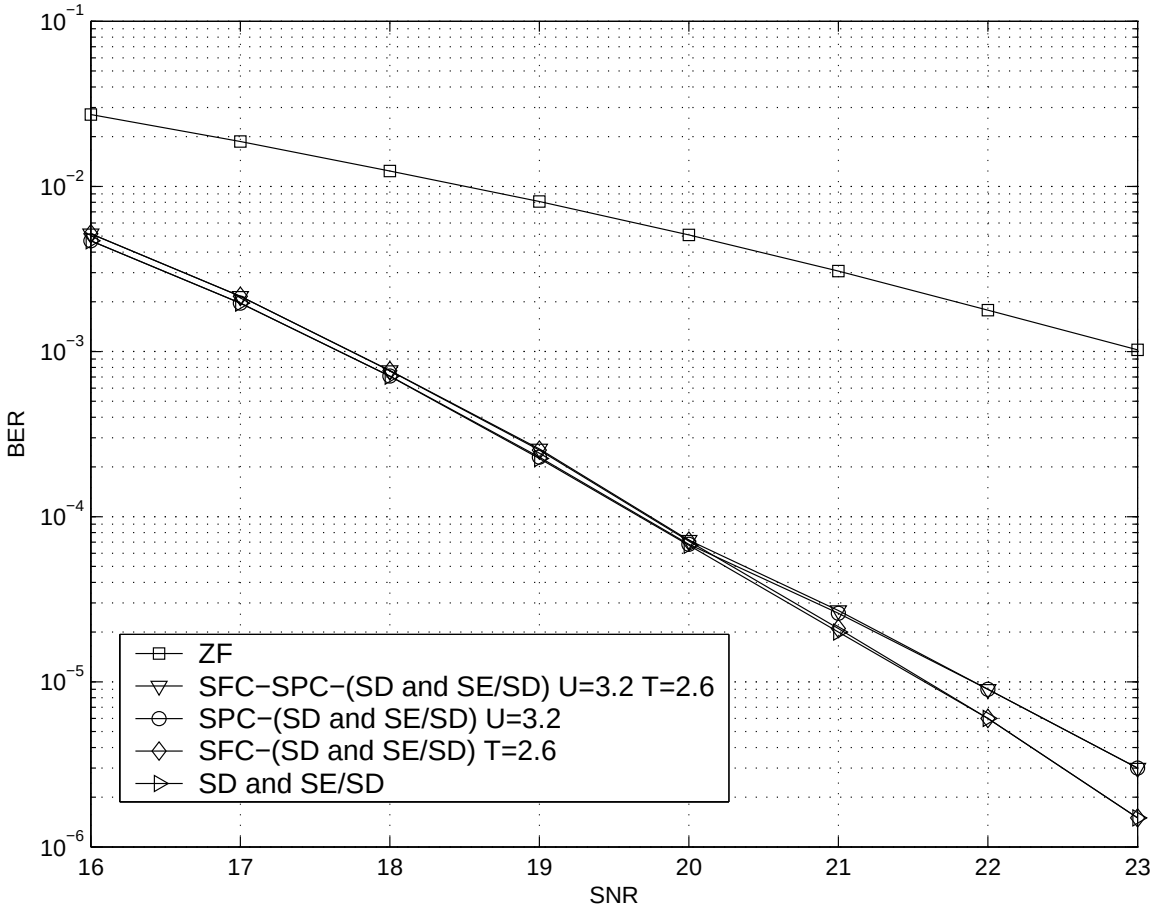

Fig. 5.3 Uncoded detector BER performance comparison: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

of these two-stage detectors have a significant performance advantage over the linear ZF detector. The SFC-SD detector performance, for a given value of T, has the same slope as the SD detector performance. The SPC-SD detector performance, for a given value of U, has nearly identical performance as the SD detector, below a certain SNR. Above that SNR the SPC-SD detector performance has a less steep slope than the SD detector performance.

It is therefore seen that the SPC-SD and SFC-SD detectors have different performance characteristics as a function of SNR. Either two-stage detector can have

performance as close to ML as desired at any SNR by proper parameter selection.

The three-stage SFC-SPC-SD detector performance characteristics are a combination of the performance characteristics of the SFC-SD and SPC-SD detectors. Below the SNR where the SPC-SD performance slope changes SFC-SPC-SD has identical performance to SFC-SD. Above this SNR SFC-SPC-SD has an equal performance slope as SPC-SD.

The BER performance curves for these same detectors are shown in figure 5.3 and they have the same trends as the SER performance curves.

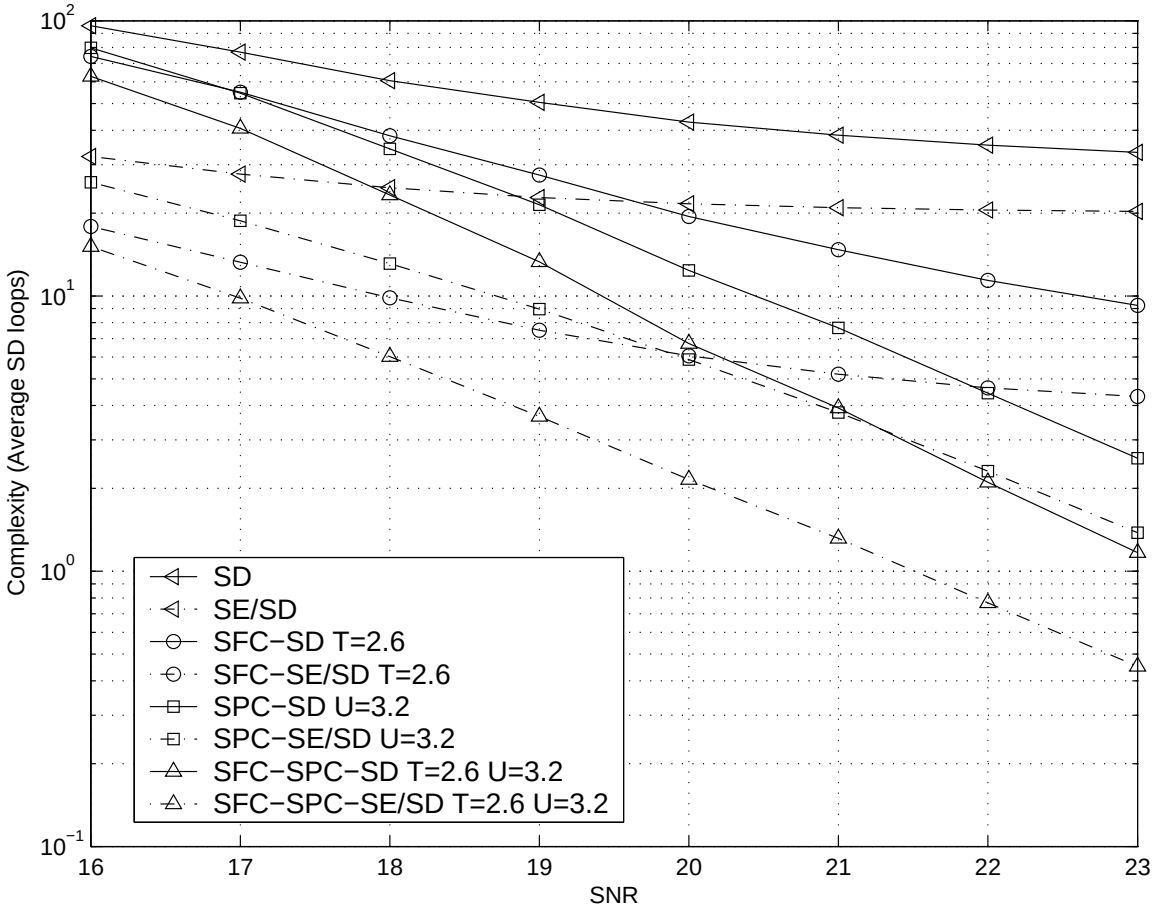

Fig. 5.4 Uncoded detector complexity comparison: 16-QAM, $N_{TX} = 5$, $N_{RX} = 7$

The complexity of the various detectors is shown in figure 5.4. When the SD is replaced by the SE/SD in any of the multi-stage decoders the performance is unchanged but the complexity is decreased. The complexity reduction from the substitution of SE/SD is seen to decrease as the SNR is increased.

Besides the substitution of the SD by the SE/SD there is seen to be always a cost of increased complexity associated with improved performance. The SPC front-end and the SFC front-end allow for different performance-complexity tradeoffs. It is never the case where either SPC-SD or SFC-SD achieves better performance *and* lower complexity. The lowest complexity detector presented is the SFC-SPC-SE/SD detector but this detector is not the best performing multi-stage detector.

5.3 LD Coded Decoder Comparison: QPSK $N_{TX} = 2$ and $N_{RX} = 2$

The detector comparison for the coded framework considers the same LD coded $N_{TX} = 2$ $N_{RX} = 2$ QPSK system as was considered in the SFC and SPC front-end simulations. The LD code used was described in section 2.6. Like the uncoded comparison we present the results for certain parameters (T for SFC and U for SPC) that achieve near-ML performance. However, in our decoder comparison, we consider the results for other parameter values presented in the previous two chapters.

In figure 5.5 it is seen that both two-stage detectors, the SPC-SD and the SFC-SD, have near-ML SER performance. Figure 5.6 presents the BER performance results for the same set of detectors. For the presented parameters, LD coded transmission decoded by either two-stage detector has better performance then uncoded transmission decoded by the SD but worse performance than coded transmission decoded by the SD. Both two stage decoders have the same performance trends as described in the

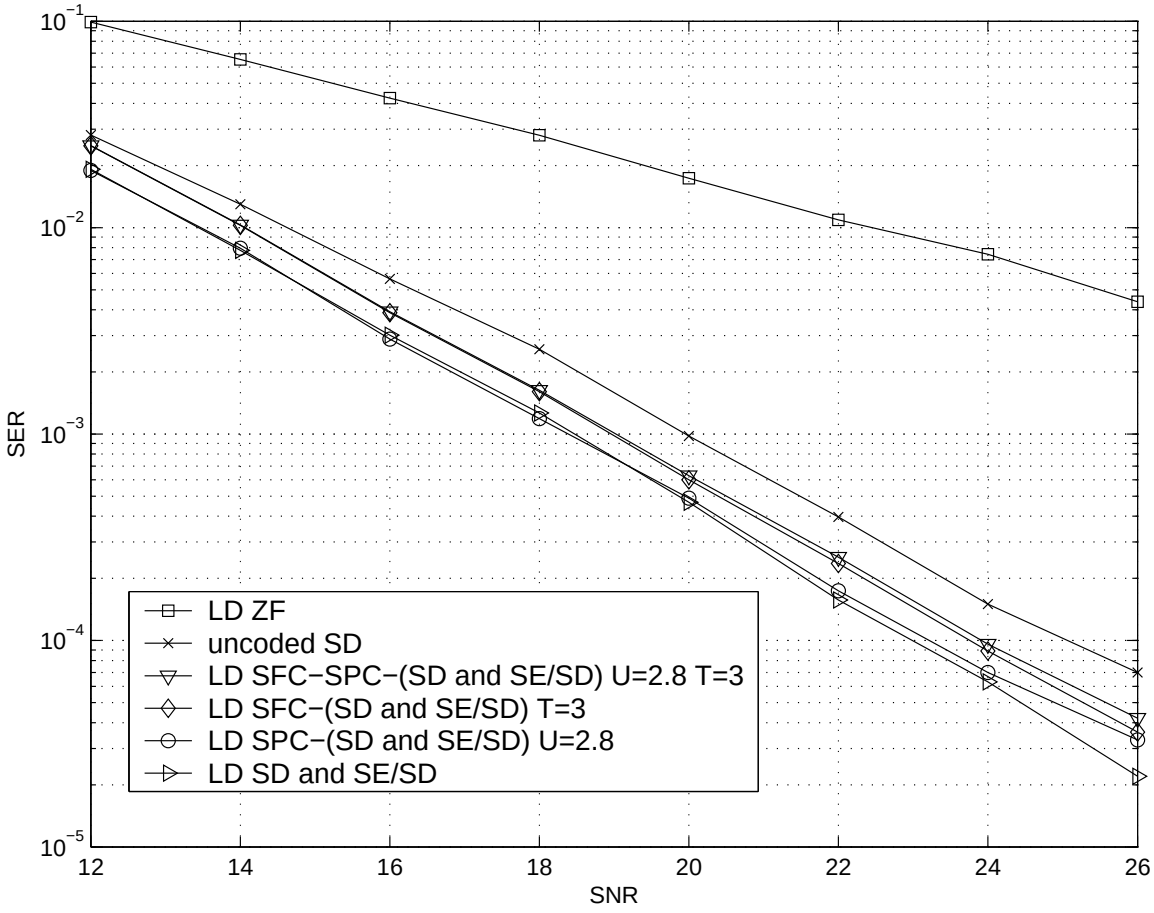

Fig. 5.5 LD decoder SER performance comparison: QPSK, $N_{TX} = 2$, $N_{RX} = 2$

uncoded case.

To further emphasize that neither front-end has intrinsically better performance the presented LD coded results use parameter values that result in SPC-SD having better performance (but higher complexity) than SFC-SD. In the uncoded detector comparison the chosen parameters resulted in SFC-SD having better performance (but higher complexity) than SPC-SD.

For the coded system, the SFC-SPC-SE/SD decoder is seen in figure 5.7 to have the lowest complexity. It is still the case that replacing the SD by the SE/SD in the

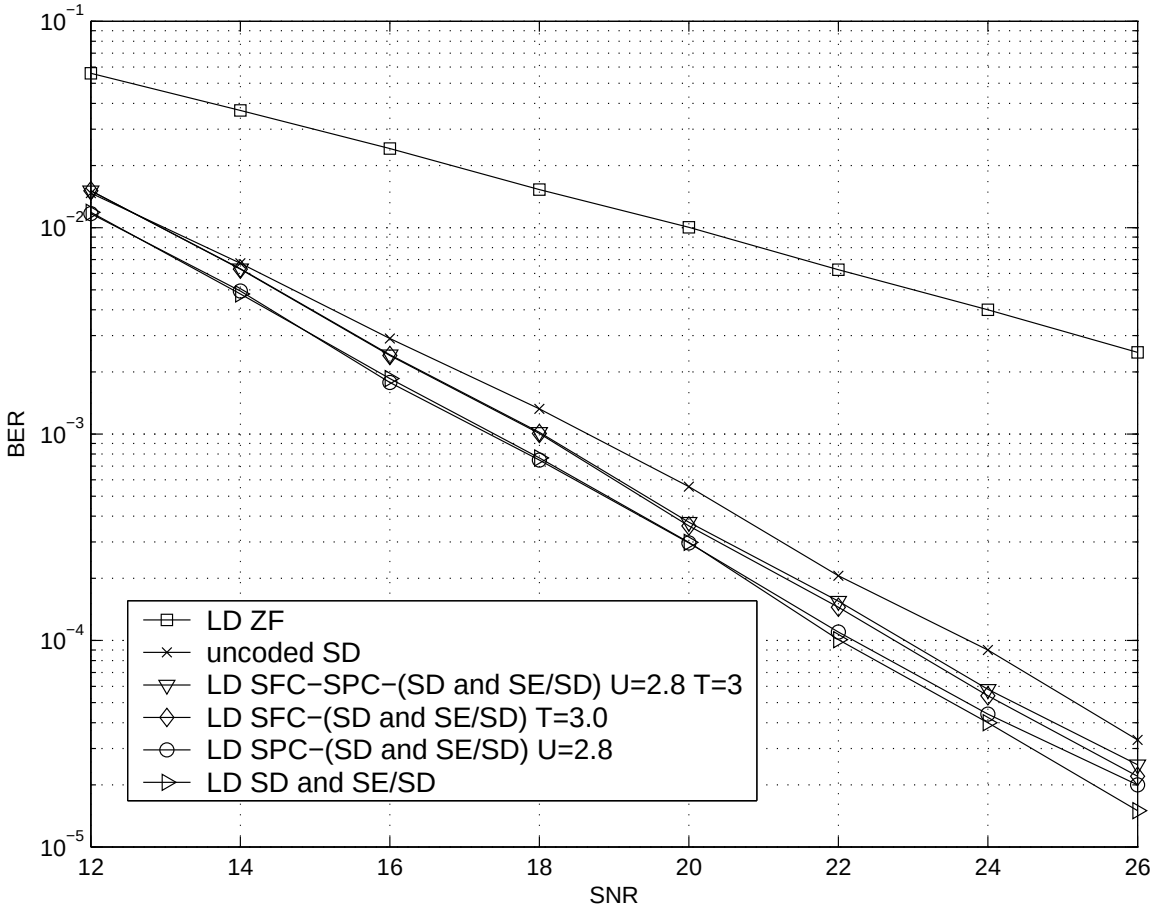

Fig. 5.6 LD decoder BER performance comparison: QPSK, $N_{TX} = 2$, $N_{RX} = 2$

multi-stage decoders reduces complexity with no reduction in performance.

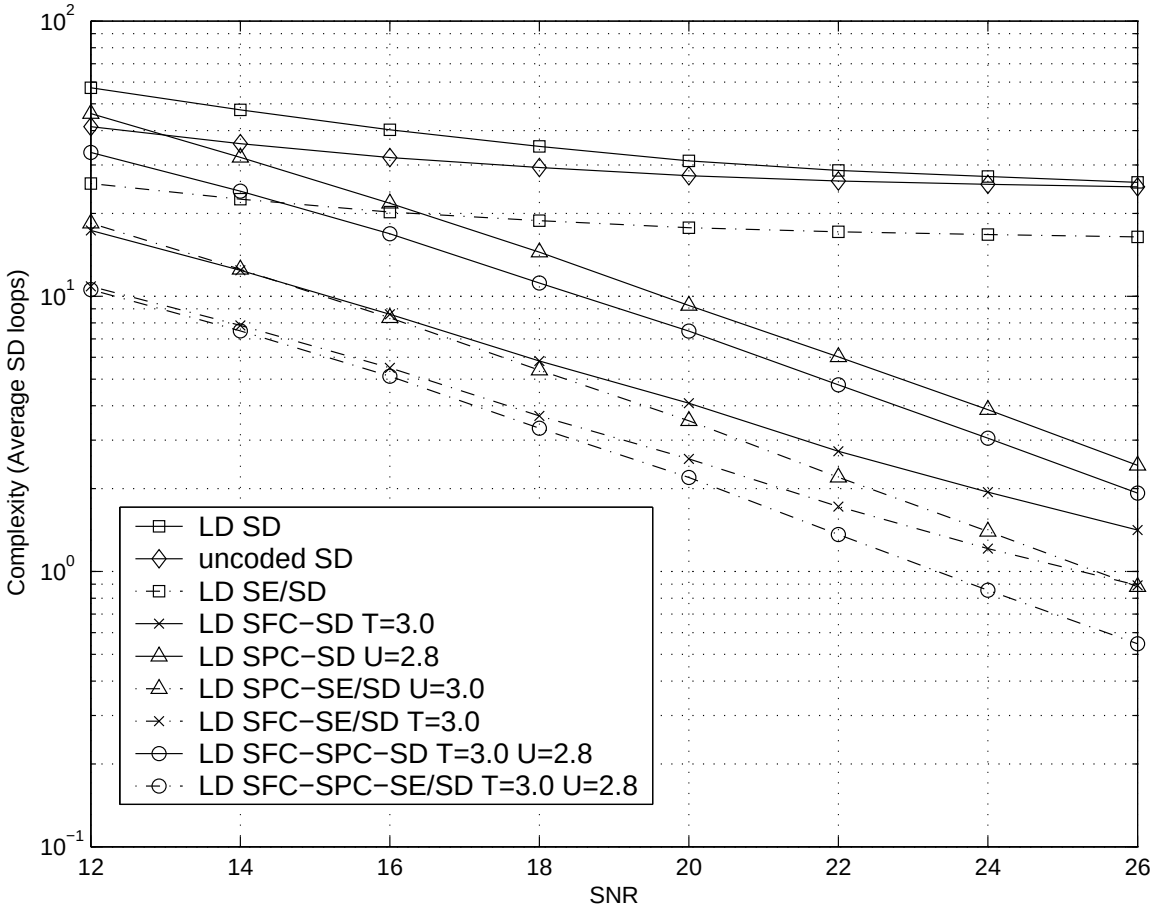

Fig. 5.7 LD decoder complexity comparison: QPSK, $N_{TX} = 2$, $N_{RX} = 2$

Chapter 6

Conclusions

There is a fundamental tradeoff between decoding complexity and error-rate performance. The two SD front-ends introduced in this work, SPC and SFC, allow both the LD decoder and the uncoded detector to exploit this tradeoff. Simulations of both SPC-SD and SFC-SD decoders demonstrate this tradeoff. Changing the system parameter, U for SPC and T for SFC, is the means by which the complexity-performance tradeoff is adjusted. Moreover, these simulations show how this tradeoff is affected by SNR. SPC-SD achieves ML performance up to a certain threshold. SFC-SD has a performance slope parallel to ML performance. By combining both front-ends into a SFC-SPC front-end further tradeoffs between complexity and performance can be achieved. It is possible to choose the parameters so that the performance of the two-stage decoder has only slightly worse performance but much lower complexity than the SD. Both SPC and SFC have their own advantages and disadvantages and each achieves a unique complexity-performance tradeoff.

A key feature of both the SPC front-end and the SFC front-end is their ability to precede any variant of the SD. These techniques can therefore leverage existing and future enhancements of the SD. By substituting SD by SE/SD the complexity

of the two-stage decoder is reduced with no reduction in performance. Although computer simulations show that, in the presence of channel estimation uncertainty, the performance of SD decreases it is also seen that SPC and SFC remain a viable means of obtaining a complexity-performance tradeoff.

Compared to a benchmark uncoded MIMO system detected with the SD, a LD coded MIMO system, transmitting at the same data rate, decoded with a sub-ML decoder, either the SPC-SD or the SFC-SD, can achieve better performance with lower complexity.

Appendix A

Computer Simulation Overview and Guide

This section explains how to use the C++ programs that were used to generate the simulations in this thesis. The capabilities and limitations of these programs are also discussed. The following table provides an overview of each source code file used in the simulations. All of these files can be found on the attached CD.

Table A.1 C++ source files

Source file name	Description
uncodedMIMO.cpp	Uncoded simulation entry point and primary flow.
LDcodedMIMO.cpp	LD coded simulation entry point and primary flow.
sphere1.cpp	Implements SD based on the flowchart of [11].
sphere2.cpp	Implements SE/SD based on the flowchart of [14].
QR.cpp	QR factorization on an arbitrary matrix.
utilities.cpp	Miscellaneous procedures.
matrix.cpp	Various matrix operations.

To run a simulation put all of the above files in the same directory. These programs have been successfully compiled and run using the Linux standard g++ version 3.2.3-42 compiler and the Microsoft Visual C++ version 6.0 compiler. To run a simulation

of the uncoded framework compile uncodedMIMO.cpp and run the executable and redirect the standard output to a text file. For example, you can simulate the uncoded framework on the Linux operating system by running the following commands:

Linux> g++ uncodedMIMO.cpp -o execute_test

Linux> execute_test > output_test_file.txt &

When the execute_test executable finishes, the text file output_test_file.txt stores the results. To run a simulation of the LD coded framework compile the LDcoded-MIMO.cpp file. For example, you can simulate the coded framework on the Linux operating system by running the following commands:

Linux> g++ LDcodedMIMO.cpp -o execute_coded

Linux> execute_coded > output_coded_file.txt &

The adjustable constants for an uncoded simulation and their allowable values are presented in table A.2. The range of SNR values to be simulated can also be set in both

Table A.2 Simulation adjustable constants

Description	Allowable values
The number of transmit antennas.	$1 \leq$ ACTUAL_NUM_TX ≤ 25
The number of receive antennas.	$1 \leq$ ACTUAL_NUM_RX ≤ 25
The size of the symbol constellation.	PAM $= \{2, 4, 8, 16\}$
The minimum number of channel ralizations per SNR.	$1 \leq$ TRIALS
The minimum number of symbol errors per SNR.	$0 \leq$ MIN_ERROR
The SPC front-end U parameter.	$0 \leq$ SPC_NOISE
The SFC front-end T parameter.	$0 \leq$ pure_Ph
Channel matrix elements error variance	$0 \leq$ VAR_UNCERT

the uncoded simulation and in the LD coded simulation. The adjustable variables are the same in the LD coded simulation except that the number of transmit and receive antennas are not adjustable.

References

[1] Y. Neuvo, "Future Wireless Technologies," *IEEE 6th CAS Symposium on Emerging Technologies: Mobile and Wireless Comm.*, Shanghai, China, May 31 - June 2 2004.

[2] N. Davies, K. Cheverst, A. Friday, and K. Mitchell, " Future Wireless Applications for a Networked City: Services for Visitors and Residents," *IEEE Wireless Communications Magazine*, Feb 2002.

[3] E. Biglieri, J. Proakis, and S. Shamai, " Fading Channels: Information-Theoretic and Communications Aspects," *IEEE Transactions on Information Theory*, Vol. 44, No. 6, Oct 1998.

[4] J. Proakis, *Digital Communications*, 4th. ed. McGraw-Hill, 2001.

[5] D. Gesbert, H. Bolcskei, D. A. Gore, and A. J. Paulraj, "Outdoor MIMO Wireless Channels: Models and Performance Prediction," *IEEE Transactions on Communications*, Vol. 50, No. 12, Dec 2002.

[6] S. M. Alamouti, "A simple transmitter diversity scheme for wireless communications," *IEEE Journal on Selected Areas in Commununications*, Vol. 16, Issue 8, pp. 1451-1458, Oct. 1998.

[7] P. W. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, "V-BLAST: An Architecture for Realizing Very High Data Rates Over The Rich-Scattering

Wireless Channel," *International Symposium on Signals, Systems, and Electronics, 1998, ISSSE 98. 1998 URSI*, 29 Sep.- 2 Oct. 1998.

[8] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," *Eur. Transactions Telecommunications*, Vol. 10, pp. 585-595, Nov. 1999.

[9] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Trans. Inform. Theory*, Vol. 45, pp. 1456-1467, July 1999.

[10] B. Hassibi, and B. M. Hochwald, "High-rate codes that are linear in space and time," *www.mars.bell-labs.com*

[11] E. Viterbo, and J. Boutros, "A Universal Lattice Code Decoder for Fading Channels," *IEEE Transactions on Information Theory*, Vol.45, No.5, pp. 1639-1642, July 1999.

[12] B. Hassibi, and H. Vikalo, "On the expected complexity of the sphere decoding," *2001 Conference of Signals, Systems, and Computers*, Vol. 2, pp. 1051 - 1055, 4 - 7 Nov. 2001.

[13] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Transactions on Information Theory*, Vol.48, No.8, pp. 2201-2214, August 2002.

[14] A. M. Chan, and I. Lee, "A new reduced-complexity sphere decoder for multiple antenna systems," *IEEE International Conference on ICC 2002*, Vol. 1, pp. 460-464, 28 April - 2 May 2002.

[15] C. P. Schorr and M. Euchner, " Lattice basis reduction: improved practical algorithms and solving subset sum problems," *Mathematical Programming*, Vol. 66, pp. 181-191, 1994.

[16] R. Gowaikar, and B. Hassibi, "Efficient Statistical Pruning for Maximum Likelihood decoding," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 5, pp. 49-52, 6 - 10 April 2003.

[17] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Technical Journal* Autumn 1996.

[18] A. K. Lenstra, H. W. Lenstra, and L. Lovasz, "Factoring polynomials with rational coefficients," *Math Ann.*, Vol. 21 pp. 515-534, 1982.

[19] H. Yao, and G. W. Wornell, "Lattice-reduction-aided detectors for MIMO communication systems," *Proceedings of IEEE Globecom 2002*, Taipei, Taiwan, November 2002.

[20] C. Windpassinger, R. F. H. Fischer, "Low-Complexity Near-Maximum-Likelihood Detection and Precoding for MIMO Systems using Lattice Reduction," *ITW 2003*, Paris, France, March 31 - April 4 2003.

[21] L. L. Scharf and B. Friedlander, "Matched Subspace Detectors," *IEEE Transactions on Signal Processing*, Vol. 42, No. 8, pp. 2146-2157, August 1994.

[22] L. L. Scharf, *Statistical Signal Processing*, Addison-Wesley, 1991.

[23] C. Schlegel, S. Roy, P. D. Alexander, Z-J. Xiang, " Multiuser Projection Receivers," *IEEE Journal on Selected Areas in Communications*, Vol. 14, Issue 8, pp. 1610-1618, Oct. 1996.

[24] S. Haykin, *Digital Communications*, Wiley, 1988.

[25] D. Seethaler, H. Artes, and F. Hlawatsch, "Efficient Near-ML Detection for MIMO Channels: The Sphere-Projection Algorithm," *Global Telecommunications Conference, 2003*, Vol. 4, pp. 2089-2093, Dec. 2003.

[26] P. D. Alexander, L. K. Rasmussen, and C. B. Schlegel, "A linear receiver for coded multiuser CDMA," *IEEE Trans. Comm.*, Vol. 45 pp. 605-610, May 1997.

[27] M. Pohst, "On the computation of lattice vectors of minimal length, successive minima and reduced basis with applications," *ACM SIGSAM Bull.*, Vol. 15 pp. 37-44, 1981.

[28] T. Kailath, A. H. Sayed, B. Hassibi, *Linear Estimation*, Prentice Hall, 1999.

[29] M. O. Damen, H. E. Gamal, G. Caire, "On Maximum-Likelihood Detection and the Search for the Closest Lattice Point," *IEEE Transactions on Information Theory*, Vol. 49, Issue 10, pp. 2389-2402, Oct. 2003.

[30] J. Choi, "An iterative detection for MIMO wireless systems using projection and hard-decision with a reliability measure," *ICICS-PCM 2003* Singapore, 15-18 Dec. 2003.