Nonlinear receivers for DS-CDMA

Rudolf Tanner

A thesis submitted for the degree of Doctor of Philosophy.
The University of Edinburgh.
September 1998

Abstract

The growing demand for capacity in wireless communications is the driving force behind improving established networks and the deployment of a new worldwide mobile standard. Capacity calculations show that the direct sequence code division multiple access (DS-CDMA) technique has more capacity than the time division multiple access technique. Therefore, most 3rd generation mobile systems will incorporate some sort of DS-CDMA.

In this thesis DS-CDMA receiver structures are investigated from the view point of pattern recognition which leads to new DS-CDMA receiver structures. It is known that the optimum DS-CDMA receiver has a nonlinear structure with prohibitive complexity for practical implementation. It is also known that the currently implemented receiver in 2nd generation DS-CDMA mobile handsets has poor performance, because it suffers from multiuser interference. Consequently, this work focuses on sub-optimum nonlinear receivers for DS-CDMA in the downlink scenario.

First, the thesis reviews DS-CDMA, established equalisers, DS-CDMA receivers and pattern recognition techniques. Then the new receivers are proposed. It is shown that DS-CDMA can be considered as a pattern recognition problem and hence, pattern recognition techniques can be exploited in order to develop DS-CDMA receivers. Another approach is to apply known equaliser structures for DS-CDMA. One proposed receiver is based on the Volterra series expansion and processes the received signal at the chip rate. Another receiver is a symbol rate radial basis function network (RBFN) receiver with reduced complexity. Subsequently, a receiver is proposed based on linear programming (LP) which is especially tailored for nonlinearly separable scenarios. The LP based receiver performance is equivalent to the known decorrelating detector in linearly separable scenarios. Finally, a hybrid receiver is proposed which combines LP and RBFN and which exploits knowledge gained from pattern recognition. This structure has lower complexity than the full RBF and good performance, and has a large potential for further improvements.

Monte-Carlo simulations compare the proposed DS-CDMA receivers against established linear and nonlinear receivers. It is shown that all proposed receivers outperform the known linear receivers. The Volterra receiver's complexity is relatively high for the performance gain achieved and might not suit practical implementation. The other receiver's complexity was greatly reduced but it performs nearly as well as an optimum symbol by symbol detector.

This thesis shows that DS-CDMA is a pattern recognition problem and that pattern recognition techniques can simplify DS-CDMA receiver structures. Knowledge is gained from the DS-CDMA signal patterns which help to understand the problem of a DS-CDMA receiver. It should be noted that from the large number of known techniques, only a few pattern recognition techniques are considered in this work, and any further work should look at other techniques.

Pattern recognition techniques can reduce the complexity of existing DS-CDMA receivers while maintaining performance, leading to novel receiver structures.

Acknowledgements

The three years of my PhD reminded me of hillwalking and climbing in the Swiss Alps. I started the nonlinear CDMA receiver tour with a well prepared and filled rucksack thanks to my parents, Gitte and Georg, the Paul Josef Jenni Foundation in Basle and the Dept. of Education Baselland for the funding.

The first year trail began with a walk along the signal processing path and the digital communications meadows. Then I arrived at the foothills of the CDMA mountains. The first boulder was high and steep. Fortunately my supervisor had installed a rope which made the first few steps easier. Thanks to David Cruickshank for his guidance, patience, encouragement and support over the three years. Finally I reached the top, but it appeared to be only one of a chain of summits. Here, I made a map of the mountains based on pattern recognition. Next, I decided to climb up the Volterra rock. At the summit of the Volterra rock there was a hut where I wrote a conference paper. In order to keep myself fit, I started running round Arthur Seat and met Christine. Consequently we met regularly, for running. Because the Volterra rock is large, it enabled me to write another paper. The second year path was very demanding because it went up and down lots of little hills. I spoke to many group members about problems and would like to thank John Thompson, Bernie Mulgrew, Dave Laurenson, Gunther Auer, Richard Stirling-Gallacher and Steve McLaughlin. Then the currency exchange rate went up. In order to fill the gap left in my rucksack, I went on two microway trips. I enjoyed it very much and would like to thank Iain Lindsay. Then, the nonlinear trail passed by the MLP mud where I got stuck in a local minima. Fortunately, I found a ladder which brought me back onto the surface and I wrote a letter about it. Thanks to Cathy Hassell for the discussions on linear programming. Because the LP region was not very attractive, I went on and reached the RBF hills. This region is safe because there are no local minima. Then I decided to climb up the Mahalanobis rock. At the summit I wrote a conference paper which Dave is going to present in sunny South-Africa, while I stay in rainy Scotland. Meanwhile, more than two years were over and the third year trail began, and I started writing up. Thanks to Ian Band for reading the thesis and the many discussions we had on the subject. Also thanks to Iain Mann for reading the thesis, and the funny discussions on: speech vs. CDMA. What is left now are several unknown paths which go in different directions. I leave it to someone else to investigate them and to draw the map. Finally, I would like to thank two particular group members, both finishing at the same time as me. Sarat Patra for the many discussions about God and the earth, and George Taylor for discussions on HW and SW engineering issues. Surprisingly, Christine accompanied me during my tour and did not give up despite the fact I spent more time in the mountains than with her. Last July, I married her in Edinburgh and Sarat welcomed me into the club of *happy husbands* (HH), which I am indeed. A very special thanks to my wife Christine.

I will never forget the three years spent in the CDMA mountains with the signals and systems group in Edinburgh. Thanks to all group members for the pub sessions, the food festival and the time spent together while working towards a PhD.

Declaration of originality

I hereby declare that the research recorded in this thesis and the thesis itself was composed and originated entirely by myself in the Department of Electronics and Electrical Engineering at The University of Edinburgh. The computer software used to perform the simulations was written by myself with the following exceptions:

- The routines used for memory allocation, memory deallocation, matrix inversion and to generate Gaussian distributed noise were obtained from $Numerical\ recipes\ in\ C$ [1].
- The algorithm used to generate slabs used in sub section 3.1.3 was supplied by Catherine Z. W. Hassell-Sweatman.
- The routine used to do the linear programming in sub section 3.1.3 and chapter 7 was part of a commercially available package, details are given in page 122.

September 1998 Rudolf Tanner

Nomenclature

α	group with even powers
β	group with odd powers
δ	threshold
ω_c	carrier frequency
ω_i	ith class
μ	mean
μ_m	mth mean, or mth permutation
ϕ_u	carrier phase shift for user u
$\phi(\cdot)$	nonlinear function
$\mathbf{\Phi}$	matrix having $\phi(\cdot)$ as rows
$\psi(t)$	chip waveform
σ	standard deviation
σ_m	mth neuron's spread
σ^2	variance or noise power
τ_u	time delay of the uth user's signal
$\|\cdot\|$	Euclidean norm
$\|\mathbf{A}\|$	determinate of matrix $\mathbf{A}$
$*$	Hadamard vector product
$\mathbf{A}$	a $(i \times j)$ matrix with i rows and j columns
$a_{i,j}$	element of ith row, jth column in $\mathbf{A}$
$\mathbf{a}_i$	ith row in $\mathbf{A}$
$\mathbf{a}$	a vector
a_i	ith vector element
$\mathbf{A}^H$,$\mathbf{a}^H$	the conjugate transpose
$\mathbf{A}^T$,$\mathbf{a}^T$	the transpose
B	(narrowband) bandwidth
$\mathbf{B}$	$(M \times U)$ combination matrix containing all $M = 2^U$ binary combinations
C	(channel) capacity
$C(O, M)$	Volterra complexity for order O and memory span M
C_i	ith cost
$\mathbf{C}$	matrix containing all spreading codes
$c_{u,n}$	element of uth row, nth column in $\mathbf{C}$
$\mathbf{c}_u$	uth users's spreading code
$\mathbf{c}_m$	mth RBF centre
$D_d(k)$	kth data bit of the desired user d
$D_u(k)$	kth data bit of user u
$\hat{D}_u(k)$	estimated kth data bit of user u
$d(\cdot)$	desired response
$\mathbb{E}^d$	d-dimensional Euclidean space
$E[\cdot]$	expected value
E_b	energy per bit

$e(\cdot)$	error
$\exp(\cdot)$	natural exponential function
$\mathcal{F}(\cdot)$	approximation function
f	frequency
$f_i(\cdot)$	ith function
G_i	ith group
$g(t)$	white Gaussian noise
$\mathbf{g}(k)$	kth symbol's noise term
$\mathbf{H}$	channel matrix
H_i	ith hypothesis
$\mathcal{H}$	hyperplane
H_{ch}	channel impulse response
$H_m[\cdot]$	mth order Volterra operator
$h_m(\cdot)$	mth order Volterra kernel
h_l	lth channel tap coefficient
$\mathbf{I}$	identity matrix
J_{min}	minimum mean square error
K	average factor
L	number of channel taps
LR	likelihood ratio
L_1,L_2	distance measure
M	number of combinations (noise free signal combinations)
$M(N,O)$	length of the Oth order Volterra expansion with memory length N
M_{13}	length of a 1st and 3rd order Volterra expansion
$\mathcal{N}(\mu,\sigma^2)$	normal distribution with mean μ and variance σ^2
N	processing gain
N_J	jamming power
N_j	number of patterns in the jth class
N_0	thermal noise density
$O(\cdot)$	proportional algorithm complexity
$P(x)$	$a\ priori$ probability of x
$P(x\|y)$	$a\ posteriori$ probability of x given y
P	signal power, or a set
P_u	uth user's signal power
P_e	probability of error
$\mathbf{P}$	generating matrix with all possible signals combinations
$\mathbf{p}$	prototype pattern (point)
p_i	ith group
$p(x\|y)$	conditional probability density function of x given y
$Q(\cdot)$	Gaussian error function
$\mathbb{R}^d$	d-dimensional space
$\mathbf{R}$	generating matrix with all possible preprocessed signal combinations
$\mathbf{R}_{yy}$	autocorrelation matrix
$\mathbf{r}_{xy}$	crosscorrelation vector
$\mathbf{r}(k)$	kth preprocessed signal at symbol rate
$r_u(\cdot)$	received preprocessed signal of user u
$\mathbf{S}$	covariance matrix, or crosscorrelation matrix

$\mathbf{s}_d$	desired signal
$\mathbf{s}_m$	mth signal
S_i	ith slab
SNR	signal to noise ratio
s_d	dth element of signal $\mathbf{s}$
$s_u(t)$	spreading waveform of user u
sgn$(\cdot)$	sign operator
T	bit interval
T_c	chip duration
t	time
U	number of transmitting users
$\mathbf{v}(k)$	kth Volterra expanded signal
$v(t)$	continuous Volterra series
W	spread bandwidth
$\mathbf{w}$	filter weights
X	set containing all possible noise free signal states
$\mathbf{x}(k)$	kth received noise free signal at chip rate
$\mathbf{x}_m$	mth pattern
$x(t)$	received (noise free) signal
$x_u(t)$	received signal of the uth user
$\widehat{y}(k)$	estimate of $y(k)$
$y(t)$	received noise corrupted signal
$y(kN+n)$	nth chip of the kth symbol with spreading length N
$\mathbf{y}(k)$	kth received symbol at symbol rate

Abbreviations

ANN	annealed neural network
AWGN	additive white Gaussian noise
BER	bit error ratio
BP	back propagation (algorithm)
BS	base station
BSC	base station controller
bps	bits per second
CLB	chip level based (receiver)
CDMA	code division multiple access
cdf	cumulative density function
CIR	carrier to interference ratio
CRBF	chip level based radial basis function receiver
DS	direct sequence
DSP	digital signal processor
DECO	decorrelating detector
EGC	equal gain combining
EMC	electromagnetic compatibility
ERBF	radial basis function with Euclidean distance measure
ETSI	European Telecommunications Standards Institute
FDMA	frequency division multiple access
FECC	forward error correction coding
FH	frequency hopping
FIR	finite impulse response
flops	floating point operations
GSM	Global System for Mobile
HNN	Hopfield neural network
HOS	higher order statistics
IMT 2000	International Mobile Telecommunications 2000
ICI	inter chip interference
ISI	inter symbol interference
IS-95	interim standard-95
ITU	International Telecommunication Union
kbps	kilo bit per second
LMS	least mean square
LP	linear programming
LPI	low probability of interception
LSE	least square error
LTE	linear transversal equaliser
MAC	multiply and accumulate
MAI	multiple access interference
MAP	maximum a posteriori

Mchip/s	mega chips per second
MF	matched filter
ML	maximum likelihood
MLP	multilayer perceptron
MLSE	maximum likelihood sequence estimator
MLSD	maximum likelihood symbol detector
MMSE	minimum mean square error
MRBF	radial basis function with Mahalanobis distance measure
MRC	maximum ratio combining
MSC	mobile switching centre
MSD	multistage detector
MSE	mean square error
MUD	multiuser detector
NN	neural network
pdf	probability density function
PG	processing gain
PN	pseudo-noise or pseudo-random
PPB	preprocessing based (receiver)
RBF	radial basis function
RBFN	radial basis function network
RLS	least recursive square
RNN	recurrent neural network
SS	spread spectrum
SPP	simplified preprocessing
SRBF	radial basis function with super centres
TDMA	time division multiple access
TH	time hopping
UMTS	Universal Mobile Telecommunication Standard
VS	Volterra series
VSi	ith order Volterra receiver
WCDMA	wideband CDMA
ZF	zero forcing

Contents

List of figures

List of tables

Chapter 1
Introduction

Over the last few decades society has changed considerably. More people than ever are affluent and educated. Therefore many more people become involved with technology such as computers. Communications is often called the market of the future in newspapers. However, customer needs and wishes can not be satisfied if the requested technology is not available at a reasonable price. Fortunately, as society tends towards mobility, technology tends towards portability. Technology has progressed from making large and bulky products to small and smart products. In the past, network operators offered mainly telephony (voice) and occasionally pager services. But the customer of the future will request in addition email, fax, local area network, Internet access and video services. A short list of possible features which should entice users includes:

- high speed hardware available;
- cost effective hardware available;
- interactive user interfaces established;
- packet data available and established;
- powerful application rich user devices available.

Therefore, it should not be difficult to develop equipment and provide a network. However, this is a very cost intensive investment and the network operator needs to be sure of some return on their investment.

The network operator stands between the service providers and the customers and provides the infrastructure in order to connect them together. Since both sides grow very rapidly, it becomes difficult for the network provider to keep pace with upgrading the network's capacity. A few points emphasise this dilemma:

- growth in ownership of mobile phones;
- growth in ownership of (portable) computers;
- growth in use of Internet applications and services;
- accelerating Internet commerce $\rightsquigarrow$ demand for home business;
- success of Internet applications $\rightsquigarrow$ demand for high data rate services;

- success of computer games $\leadsto$ demand for amusement.

Therefore, network operators need equipment which enables them to satisfy both parties. Finally, a list illustrates some network requirements:

- enhanced information and multimedia capabilities;
- subscriber control, billing, fraud detection;
- capacity to meet more demanding environments;
- interoperability;
- ease of operations and maintenance;
- robust voice systems to operate in more dynamic environment;
- preservation of previous investment.

This last list shows also that this equipment must be flexible in its nature in order to follow new trends. One step in this direction is a global standard for mobile communications [2]. Work in this direction is in progress with the regulatory bodies such as the *European Telecommunications Standards Institute* (ETSI) and the *International Telecommunication Union* (ITU) where the latter will define the 3rd generation mobile standard *International Mobile Telecommunications* 2000 (IMT 2000) based on the submitted proposals. The European proposal is known as *Universal Mobile Telecommunication Standard* (UMTS) [3]. It combines two techniques to meet the expected capacity demand, see [4–6] and [7–10]. A flexible standard is also desired in order to run a wide range of services. Thus, a combined cordless and mobile telephony system has been suggested [11], and a combined satellite and terrestrial system [12]. It also has been suggested to incorporate existing technology (standards) [13], in order to preserve the investments made in 2nd generation systems. Some ideas seem to contradict each other. Moreover, since the system must be very flexible it may be doubtful if a single standard can provide all desired services.

Nevertheless, the basic constraints in engineering are of a physical nature [14]. The available frequency spectrum is limited, and hence it must be ensured that it is exploited efficiently [15]. In order to do so, mobile communications such as mobile telephony must use techniques based on multiuser communications.

1.1 Multiuser communications

The aim in mobile telephony is to have as many transmitting users as possible using a certain frequency band (channel). In the past, where analogue technology dominated circuitry, *fre-*

quency division multiple access (FDMA) was used. Several users share a common channel while all users are frequency separated by sub-channels. Every user is communicating over an individual channel over the whole period of time of the call, see Figure 1.1. Figure 1.1 shows U transmitting users in the time t, frequency f and power P domain. After finishing a call, the sub-channel can be reused and allocated to another user.

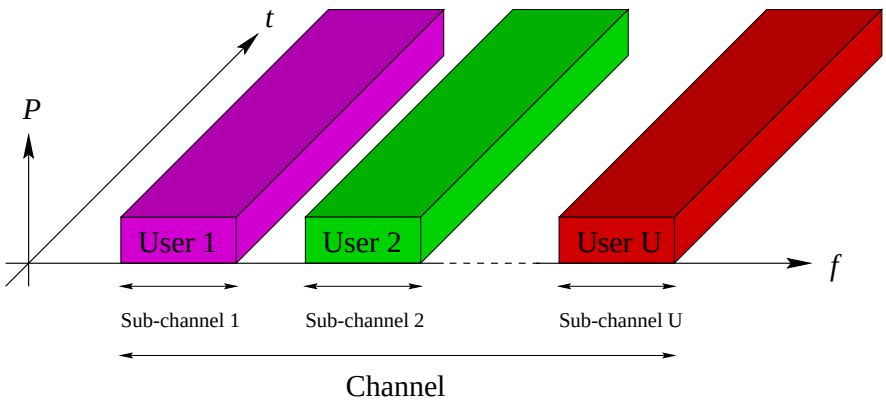


Figure 1.1: Frequency division multiple access (FDMA).

This is not an efficient way of exploiting the frequency spectrum. Because the sub-channel is occupied even if no information is transmitted, and the system requires guard bands. Because it was found that mobile conversations have a duty factor of $1/2$ (time used for conversation/time of call) [16], even more users can share a channel at the same time. The availability of cheap digital circuitry made it possible to deploy digital mobile network systems, the 2nd generation mobile networks. Most networks exploit a technique known as *time division multiple access* (TDMA), such as the popular *global system for mobile* (GSM) [17, 18]. Figure 1.2 shows how a TDMA system works. Several users share a common channel but they are separated by time. Each user transmits and receives for a short period of time (time slot) within a frame.

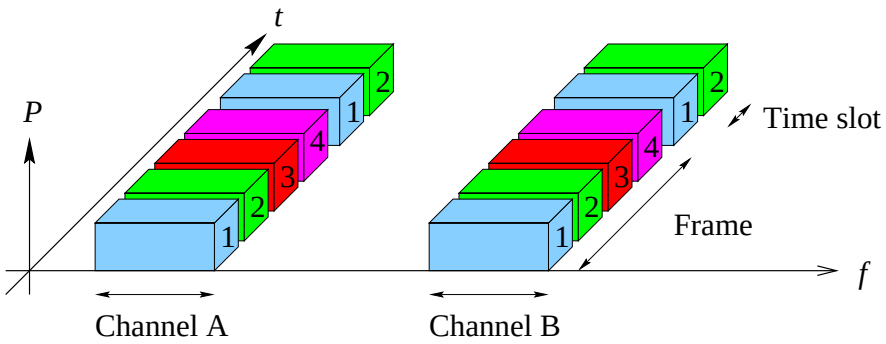


Figure 1.2: Time division multiple access (TDMA).

There is another technology which is used for 2nd generation mobile networks. The idea is based on *spread spectrum* (SS) [19, 20] which has been in use for a long time in military and space applications and is called *code division multiple access* (CDMA) [21]. In CDMA networks all users share a common channel in time and frequency. The separation is done using a code. Each user transmits with a unique code, the spreading sequence, and since the receiver knows the user's code it can demodulate and extract the information. Figure 1.3 shows how this technique works. Usually, within a network there are two channels, one for the uplink (mobile to base station) and one for the downlink (base station to mobile). All user share both channels at the same time. The number of users which can communicate simultaneously is dependent,

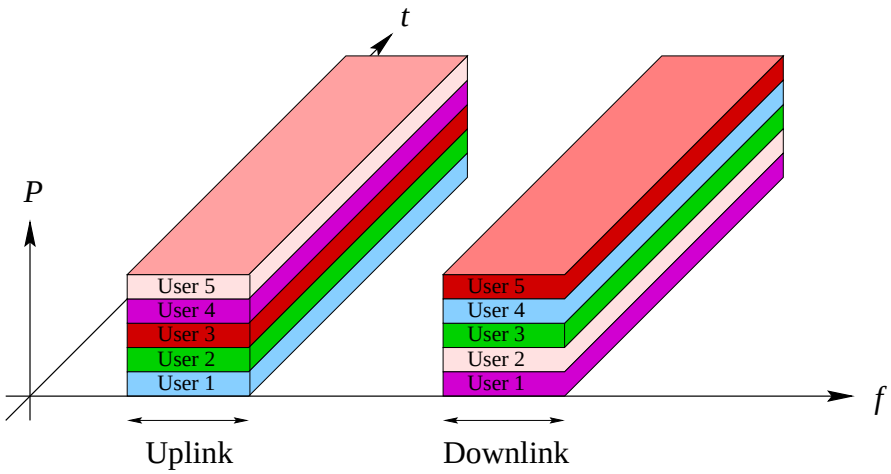


Figure 1.3: Code division multiple access (CDMA).

among other factors, on the length of the spreading sequence (code, a series of binary data). If this code is long and pseudo-random then the interference induced from other users can be interpreted as an increased background noise level. CDMA offers many advantages over the other two techniques [16, 22]. The capacity is soft limited, which means that as more users are active the higher the background noise becomes and performance (in terms of probability of errors P_e or *bit error ratio* (BER)) degrades. It is less susceptible to effects induced from a changing environment, which is important in mobile communications. Finally, it is generally believed that its capacity is much greater than that of the established (TDMA) systems [23, 24]. However, CDMA systems also have some additional constraints which must be considered if it is to be used for cellular mobile communications, e.g. requirement for power control [25].

1.2 Cellular communications

In this work, mobile communications is analogous to cellular communications and is thought of as a terrestrial network of cells, depicted as hexagons, and the investigations are carried out in a single cell system with U simultaneously transmitting users for the downlink scenario. Mobile satellite communications shall be omitted. Each cell covers a certain region and has in its centre a base station, to which all users (mobiles) within a cell are linked. The *base station* (BS) is connected to a *base station controller* (BSC) and the BSC is connected to the *mobile switching centre* (MSC), see Figure 1.4. Finally, the MSC is connected to the landline (public) network and the operator's data base for billing and network management.

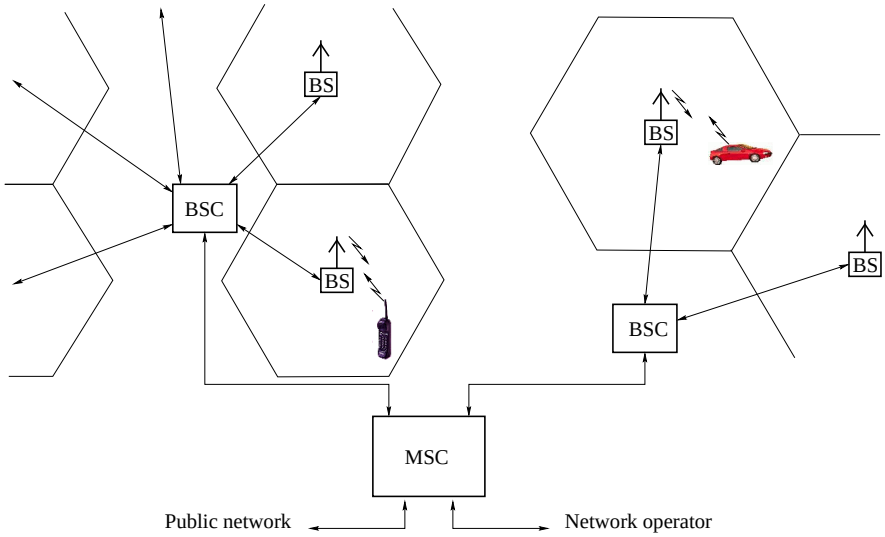


Figure 1.4: The structure of a cellular network.

While a mobile moves it may enter another cell, and in order to keep the link quality high, it should get linked to the nearest base station. This procedure is called handover, changing from one BS to another. Several handover techniques are in use. GSM uses a mobile assisted hard handover, where the BSC handles the handover between two BSs. CDMA systems use a soft handover [26, 27], where each user's signal strength is monitored and the MSC handles the handover. This technique uses the fact that in CDMA all users use the same carrier frequency, whereas in GSM different frequencies are used. Thus a GSM system over a larger area can be depicted as in Figure 1.5, where each colour represents a carrier frequency. The typical frequency reuse factor [28] for the system depicted in Figure 1.5 is $1/7$ [16]. Figure 1.5 would be a uniform colour for a CDMA network and the theoretical frequency reuse factor is one.

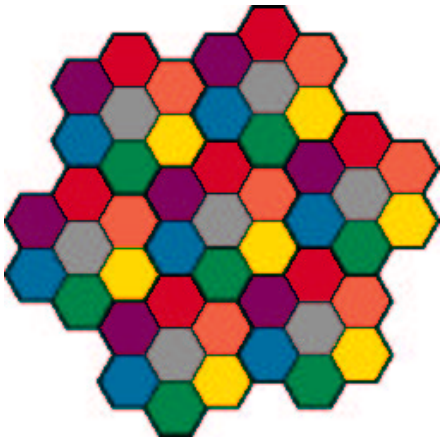


Figure 1.5: Frequency reuse pattern in established cellular networks.

1.3 Mobile environment

A possible mobile environment is depicted in Figure 1.6. It shows a car as a mobile which moves. The surroundings such as buildings, hills or lorries cause the radiated signal energy to be reflected. Hence, several delayed versions of a transmitted signal can be received. This effect is known as the multipath effect. Further, the frequency of the received signal varies due to the speed of the mobile due to the large number of incoming scatterers (Doppler spread).

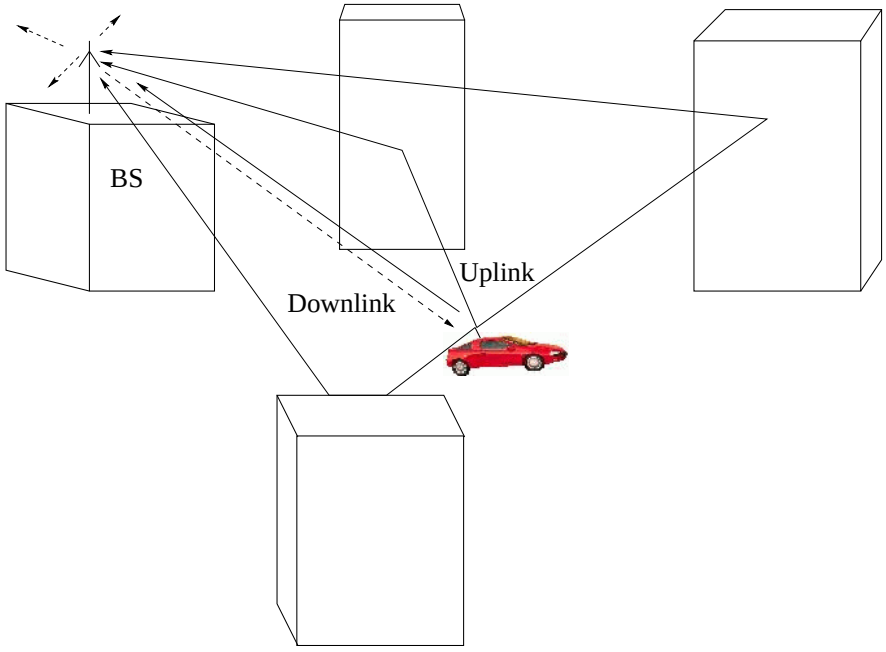


Figure 1.6: A possible mobile environment within a cell and a moving vehicle (mobile). The radiated signal will be reflected from the surrounding and the frequency is Doppler shifted due to the vehicle's motion.

A cellular network has to cope with the following effects:

- Doppler effect;
- fading;
- inter symbol interference;
- co-channel interference;
- multiple access interference;
- near-far problem.

Doppler spread is due to a moving mobile, it can be compensated within the receiver with a frequency follower. Multipath effects are harder to combat since they are also responsible for a change in the received signal strength [29, 30]. The common approach to overcome fading effects is to vary the transmitted signal power [31]. Multipath effects are also responsible for *inter symbol interference* (ISI), which can be combatted by equalising the received signal. Many equaliser structures are known with varying performance and complexity. Well designed equalisers can enhance the receiver performance at the mobile or at the base station considerably through diversity gain. Co-channel interference is interference from adjacent cells [32–34]. In addition, CDMA systems have to deal with a special kind of interference. Since all user transmit at the same carrier frequency, they interfere with each other. This effect is called *multiple access interference* (MAI). Related to this effect is the near-far problem [35]. This means that, if one user transmits at high power near the base station, its signal will cover up a weaker signal from a user which is far away from the BS. Hence CDMA systems need power control in order to ensure that all signals at the base station are received at equal strength [26, 36]. This brief introduction showed that mobile communications work in a most unfriendly environment. To get a mobile system running to satisfy the customers, a large amount of processing is involved [16, 37, 38].

The current trend in mobile communications is to apply CDMA, since CDMA has advantages over the other techniques. However, the current CDMA handset receiver idea (technology) used dates back to the fifties [16]. This receiver suffers from MAI. Hence, in order to cope with the expected demand in mobile communications, industry looks for new receiver designs. Thus, current research focuses on new receiver designs which exploit digital signal processing power [39, 40]. Recently, another way to improve performance is becoming popular called turbo coding [41]. Turbo coding can enhance system performance approaching the Shannon limit.

1.4 Thesis structure

The main objective of this thesis is to link pattern recognition with CDMA. Or in other words, CDMA is a pattern recognition problem, therefore, pattern recognition techniques can help to derive new receivers for CDMA.

The current chapter is a brief introduction into mobile telephony, and ends with the thesis structure outlined in Figure 1.7.

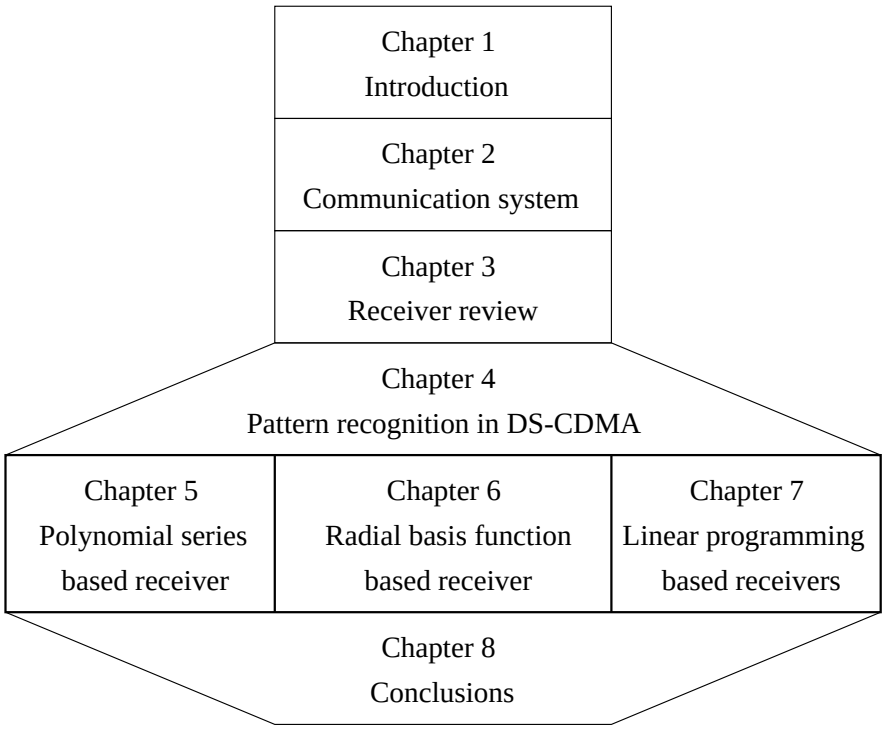


Figure 1.7: The structure of this thesis.

Chapter two discusses in more detail the CDMA system and its superiority over other multiple access schemes. Of special interest is the downlink scenario, since the handset receiver structure has to be limited in complexity for economic reasons such as handset price, battery life, processor and processor power. Currently, a simple structure is employed, which suffers from multiuser interference (all user share the same channel bandwidth) and therefore has limited performance. Next, a matrix and vector notation is introduced to describe the received signal. Chapter three reviews established receiver structures. First, equalisers are discussed followed by a survey of linear and nonlinear receivers for CDMA. It appears that only nonlinear receivers have the potential to considerably increase performance. Thus, the focus is on nonlinear receivers for CDMA handsets with reduced complexity.

Chapter 3 introduces pattern recognition. After a survey on known techniques in the field of pattern recognition, it will be shown that CDMA can be viewed as a pattern recognition problem. Examples are presented which illustrate that the CDMA signal has patterns which can be exploited in order to derive receiver structures. Further, decision boundaries are presented for two and three user scenarios which indicate that CDMA is a nonlinear separation problem.

Chapters five, six and seven present new CDMA receiver structures and contribute to the understanding of CDMA receivers together with chapter four. All proposed receiver structures are nonlinear because the optimum decision boundary is nonlinear. Simulation results show their performance which is compared against established receivers.

The last chapter is a summary and discussion on the work presented in this thesis where also further work is outlined.

Chapter 2
The communication system

This chapter focuses on mobile communication systems based on spread spectrum, particularly on direct sequence code division multiple access systems.

First, the benefit of SS systems is analysed generally. Then reasons are given for employing CDMA for cellular mobile applications followed by a descriptive analysis of cellular CDMA. The downlink scenario is of special interest and shall be investigated in more detail for two basic receiver designs. The mathematical formulation is followed by a brief description of an established cellular network. Finally, this chapter is summarised.

2.1 Spread spectrum communications

Spread spectrum communications have their origin in space and military applications. The motivation for SS was to hide the fact that one is transmitting a signal from an enemy. This feature is known as *low probability of interception* (LPI) of a SS signal, or privacy. If it is hard to determine whether a signal is present, then that signal cannot interfere substantially with other present signals, hence the system has good *electromagnetic compatibility* (EMC) [42].

Shannon [43] stated that the stationary Gaussian noise process which maximises capacity is the one that spreads its available power uniformly across the given bandwidth. Thus the capacity C for a given bandwidth B of a jamming channel is derived from the well known equation $C = B \times \log_2(1 + SNR)$, where SNR is the *signal to noise ratio* defined as:

$$SNR = \frac{P}{BN_0 + N_J}, \tag{2.1}$$

where P is the signal power, N_0 the thermal noise density and N_J the jamming power. The motivation is then to expand (spread) B in jamming situations until the total received noise power BN_0 dominates N_J [44]. The spreading leads to a reduction in required SNR which is

very advantageous for communications and can be represented as a *processing gain* (PG):

$$PG = \frac{W}{B}, \tag{2.2}$$

where W denotes the SS spreaded signal bandwidth and B the unspread signal bandwidth.

SS systems can be categorised by the techniques used to spread and despread the transmitted signal. There are four main techniques [20]:

Direct sequence (DS)	used for conventional and military applications.
Frequency hopping (FH)	popular in military applications.
Time hopping (TH)	less common, mainly used in hybrid systems.
Chirp	mainly used for radar applications.

These systems treat interference differently. DS SS averages the interference over a large period of time, whereas the other techniques combat interference by separating the desired signal in frequency or time (on average) from the majority of the interference. The spreading in DS SS is done by multiplying the user's bit with a signature sequence of bits (code), where these bits are called *chips*, which are generated by a *pseudo-noise* (PN) or random generator [20]. Such a transmitted sequence, which contains the information of the user bit is also referred to as a symbol.

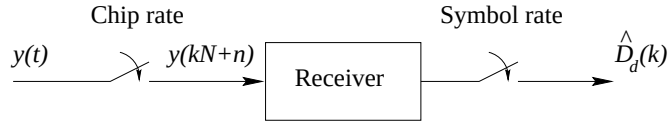
SS communication systems are also of interest for multiuser applications, since a LPI is related to a small inter user interference, giving the communication system good multiple access capability [42].

2.2 DS-CDMA

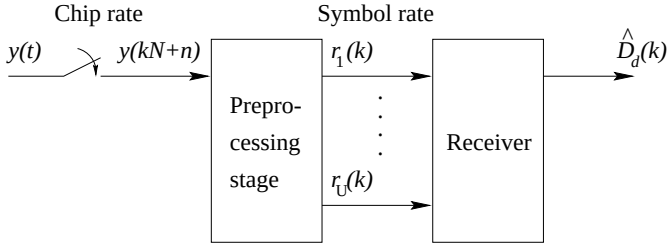
The performance gain obtained from a DS SS signal through the processing gain can be used to enable many DS SS signals to occupy the same channel bandwidth, provided that each signal has its own signature waveform. Thus it is possible to have several users transmit simultaneously over the same channel bandwidth. This type of communication is called CDMA, the combined DS SS CDMA system is simply called DS-CDMA which is often shortened to CDMA.

2.3 Cellular DS-CDMA

Consider a mobile receiver for CDMA with U simultaneous transmissions (U transmitting users) plus *additive white Gaussian noise* (AWGN). In order to describe the received signal $y(t)$ without looking at particular receiver designs (which are discussed in the next chapter), two general receiver concepts are introduced. One receiver design processes the received signal at the chip rate, whereas the other concept exploits preprocessed signals at the symbol rate, see Figure 2.1.



(a) *Chip rate based receiver.*

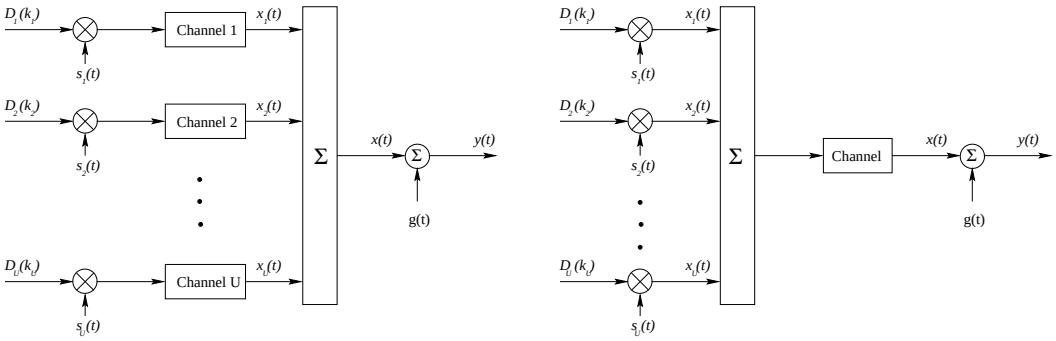


(b) *Symbol rate based receiver.*

Figure 2.1: The two CDMA receiver designs, which are discussed in the following subsections.

Figure 2.1(a) shows a conventional single user receiver, which is currently implemented in CDMA handsets. The receiver structure depicted in Figure 2.1(b) exploits preprocessed signals $r_u(k)$, which are synchronised to the symbol rate, where k stands for the kth symbol and u for the uth preprocessed output. The receiver's output is the reconstructed transmitted bit for the desired user d denoted as $\widehat{D}_d(k)$. The received signal $y(t)$ consists of a sum of user specific signals plus noise. This is illustrated in Figure 2.2, where the uplink and downlink scenario are shown for a single cell system.

According to Figure 2.2 and assuming a non-dispersive AWGN channel, the noise corrupted



(a) *The uplink scenario (mobiles to base station) for U mobiles.*

(b) *Downlink scenario (base station to mobile).*

Figure 2.2: The two possible links within a cellular DS-CDMA mobile system.

received signal $y(t)$ is defined as:

$$y(t) = x(t) + g(t) = \sum_{u=1}^{U} x_u(t) + g(t), \tag{2.3}$$

where $g(t)$ denotes the white Gaussian noise with double sided power spectral density $N_0/2$. The uth user's transmitted data bit for bit k is denoted as $D_u(k)$ and is either $+1$ or -1 with equal probability and all users are transmitting with equal power, normalised to one. Then, the received signal $x(t)$ due to the uth user is given by:

$$x_u(t) = \sqrt{2P_u} \sum_{k=-\infty}^{\infty} D_u(k) s_u(t - kT - \tau_u) \cos(\omega_c t + \phi_u) \qquad 1 \leq u \leq U, \tag{2.4}$$

where T is the bit interval, P_u, τ_u and ϕ_u are the power, delay and carrier phase shift of the uth user, and ω_c is the carrier frequency; $s_u(t)$ is the uth spreading sequence (signature) waveform given by:

$$s_u(t) = \sum_{n=0}^{N-1} c_{u,n}\, \psi(t - nT_c), \tag{2.5}$$

where $c_{u,n} \in \{1, -1\}$ is the nth element of the spreading sequence for user u, $\psi(t)$ is the chip waveform, N the processing gain (number of chips per user bit), and $T_c = T/N$ is the chip duration. It is assumed that $\psi(t)$ has unit energy and duration T_c. Each user in Figure 2.2(a) has its own delay τ_u which is taken into account in equation (2.4), while the channel delay is incorporated in the channel block. Thus the uplink (mobile to base station) is generally asynchronous.

If the delay τ_u is equal for all u, then the communication system is considered synchronous, which is typically the downlink scenario (base station to mobile), see Figure 2.2(b).

In order to simplify the notation, it is assumed that all carrier phases are equal to zero and base-band notation can be used [45]. In the downlink scenario all user signals share the same channel to a specific, whereas in the uplink scenario each user is transmitting through a unique channel with different delays. The channel is modelled by an L-tap *finite impulse response* (FIR) filter [46]. The downlink channel impulse response H_{ch} can be estimated from monitoring a pilot tone transmitted by the BS (e.g. IS-95). Since the signal processing task is done on sampled signals, it is more convenient to make use of vector and matrix notation. Therefore equation (2.3) can be rewritten for a fully synchronised downlink antipodal ($\{+1,-1\}$) DS-CDMA system with U independent users and a non-dispersive AWGN channel. It is assumed that the number of active users and their corresponding spreading sequences are known. Each uth user bit $D_u(k)$ is spread by a unique user specific spreading sequence $\mathbf{c}_u = [c_{u,1}\ c_{u,2} \dots\ c_{u,N}]$ of length N, with $n = 1, 2, \dots, N$ chips, where the chips are either $+1$ or -1[1]. Hence, equation (2.3) becomes:

$$y(kN+n) = \sum_{u=1}^{U} D_u(k) c_{u,n} + g(kN+n). \tag{2.6}$$

The received signal becomes $\mathbf{y}(k)$ in vector notation, where k denotes the kth user bit. Now, the received signal (2.6) shall be described from the view point of set theory. It can be assumed that for a certain period of time the number of users U is constant, hence also the number of possible transmitted noise free signal combinations, which is denoted by M. It is shown in the following subsections, that when U and all U spreading sequences are known, such a set of signals can be constructed. Therefore, the received signal (2.6) for the kth transmitted symbol may be rewritten as:

$$\mathbf{y}(k) = \mathbf{x}(k)^T + \mathbf{g}(k), \tag{2.7}$$

where $\mathbf{g}(k)$ is a vector containing the random noise and is the only non-deterministic component. Vector $\mathbf{x}(k)$ is drawn from a set X with equal probability, where the M elements $\mathbf{x}_m$ in X

[1] In this thesis, only real signals are considered, the extension to complex signals is straightforward.

are the noise free signal states:

$$X = \{\mathbf{x}_m : 1 \leq m \leq M\}.$$

Set X can be given as a matrix, where each row corresponds to an element of this set. The elements of X are the rows of a $(M \times N)$ matrix for a non-dispersive channel, or of a $(M \times (N + L - 1))$ matrix for multipath channels, for chip rate receivers. While the elements of X are the rows of a $(M \times U)$ matrix for symbol rate receivers. The matrices are derived in the next subsections and shall be referred to as the *generation matrices*. The variable M can have different values for different scenarios and is also defined in the following subsections. Further, different aspects and interpretations on the signal properties are discussed in the following chapters. Thus, only the basic construction of this set of possible signals shall be given here. Further, the signal description is given from a signal processing perspective, which enables the designer to understand how a receiver structure processes the received stream of samples.

2.3.1 Chip rate based receiver

The chip rate based receiver presented in Figure 2.1(a) is first analysed for simplicity for the less practical non-dispersive AWGN channel, or simply AWGN channel, and then extended to the multipath channel. The elements in X are the rows of a matrix, e.g. $\mathbf{P}$.

2.3.1.1 AWGN channel

The number of possible noise free vectors $\mathbf{x}(k)$ is given by 2^U for a U user DS-CDMA system [46], hence $M = 2^U$. Since there are N chips per bit, $\mathbf{x}(k)$ has N elements and $\mathbf{P}$ is a $(M \times N)$ matrix. Matrix $\mathbf{P}$ is constructed from a $(2^U \times U)$ combination matrix $\mathbf{B}$ containing the 2^U combinations of vectors with length U (columns) and elements with values $\{+1, -1, \}$, and a $(U \times N)$ matrix $\mathbf{C}$ with the N chip spreading code $\mathbf{c}_u$ in the uth row. For convenience, matrix $\mathbf{C}$ is denoted as a partitioned matrix [47] $\mathbf{C} = [\mathbf{c}_1^T \ldots \mathbf{c}_U^T]^T$, where $\mathbf{c}_u$ is the uth user's spreading sequence. Thus:

$$\mathbf{P} = \mathbf{BC}, \tag{2.8}$$

where the combination matrix $\mathbf{B}$ is described by[2]:

$$\mathbf{B} = \begin{bmatrix} 1 & 1 & \dots & 1 & 1 \\ 1 & 1 & \dots & 1 & -1 \\ 1 & 1 & \dots & -1 & 1 \\ \dots & \dots & \dots & \dots & \dots \\ -1 & -1 & \dots & 1 & -1 \\ -1 & -1 & \dots & -1 & 1 \\ -1 & -1 & \dots & -1 & -1 \end{bmatrix}$$

and matrix $\mathbf{C}$ is given by:

$$\mathbf{C} = \begin{bmatrix} \mathbf{c}_1^T \\ \mathbf{c}_2^T \\ \vdots \\ \mathbf{c}_U^T \end{bmatrix}.$$

Each row of $\mathbf{P}$ has length N and is the sum of U positive and/or negative spreading sequences. With the $(M \times U)$ combination matrix $\mathbf{B}$ with $M = 2^U$, as given in (2.8), this results in:

$$\mathbf{P} = \mathbf{BC} = \begin{bmatrix} \mathbf{c}_1^T + \mathbf{c}_2^T + \ldots + \mathbf{c}_{U-1}^T + \mathbf{c}_U^T \\ \mathbf{c}_1^T + \mathbf{c}_2^T + \ldots + \mathbf{c}_{U-1}^T - \mathbf{c}_U^T \\ \mathbf{c}_1^T + \mathbf{c}_2^T + \ldots - \mathbf{c}_{U-1}^T + \mathbf{c}_U^T \\ \dots \\ -\mathbf{c}_1^T - \mathbf{c}_2^T - \ldots + \mathbf{c}_{U-1}^T - \mathbf{c}_U^T \\ -\mathbf{c}_1^T - \mathbf{c}_2^T - \ldots - \mathbf{c}_{U-1}^T + \mathbf{c}_U^T \\ -\mathbf{c}_1^T - \mathbf{c}_2^T - \ldots - \mathbf{c}_{U-1}^T - \mathbf{c}_U^T \end{bmatrix}. \tag{2.9}$$

2.3.1.2 Multipath channel

The notation for the multipath channel is more difficult due to ISI. Here, more than one symbol must be taken into account, namely the previous, current and the next symbol in order to collect all of the current symbol's energy. The symbol of interest is the current symbol. It can be assumed that the spreading sequence length N is much longer than the length L of the

[2] Rows can be interchanged, since the order is not important. The columns correspond to each user's transmitted bit and the rows to the possible signal combinations between the users.

channel impulse response $H_{ch} = [h_1\ h_2\ \ldots\ h_L]$. Due to the three symbols needed $\mathbf{B}$ is of size $(M \times 3U)$ where $M = 2^{3U}$. This requires a different code matrix $\mathbf{C}$ which is an extended version of the previous one and has the form $[\mathbf{c}_1^T \ldots \mathbf{c}_U^T \mathbf{c}_1^T \ldots \mathbf{c}_U^T \mathbf{c}_1^T \ldots \mathbf{c}_U^T]^T$. However, in order to simplify the notation, combination matrix $\mathbf{B}$ is partitioned into three sub-matrices and no extended code matrix is necessary:

$$\mathbf{B} = [\mathbf{B1}\ \mathbf{B2}\ \mathbf{B3}],$$

where each sub-matrix $\mathbf{Bi}$ $(i \in \{1, 2, 3\})$ is of size $(M \times U)$. Matrix $\mathbf{C}$ is also partitioned as $\mathbf{C} = [\mathbf{C1}\ \mathbf{C2}\ \mathbf{C3}]$, where each $\mathbf{Ci}$ corresponds to the set with all spreading codes, given by $[\mathbf{c}_1^T \ldots \mathbf{c}_U^T]^T$. The channel impulse response H_{ch} is stored in a $((N + L - 1) \times 3N)$ matrix $\mathbf{H}$ which is Toeplitz [47, 48]. $\mathbf{H}$ does not have to be $(3N \times 3N)$ since only the current symbol and its ISI affected chips are of interest. Thus, a $((N + L - 1) \times 3N)$ matrix is sufficient where the first and last $(N - L + 1)$ columns of $\mathbf{H}$ only contain zeros. So for instance for a 3-tap channel $(L = 3)$ and a 5 chip spreading code $(N = 5)$, $\mathbf{H}$ becomes:

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 \end{bmatrix}.$$

Finally, in order to derive the $(M \times (N+L-1))$ matrix $\mathbf{P}$, which contains all possible received sequences, the Hadamard product $(*)$ [47] [3] is used:

$$\mathbf{P} = [\mathbf{B}^T * \mathbf{C}^T]^T \, \mathbf{H}^T \tag{2.10}$$

$$= \left[\begin{bmatrix} \mathbf{B1} \\ \mathbf{B2} \\ \mathbf{B3} \end{bmatrix} * \begin{bmatrix} \mathbf{C1} \\ \mathbf{C2} \\ \mathbf{C3} \end{bmatrix} \right]^T \mathbf{H}^T \tag{2.11}$$

$$= \begin{bmatrix} \mathbf{B1} \begin{bmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_U^T \end{bmatrix} & \mathbf{B2} \begin{bmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_U^T \end{bmatrix} & \mathbf{B3} \begin{bmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_U^T \end{bmatrix} \end{bmatrix} \mathbf{H}^T. \tag{2.12}$$

In the non-dispersive channel scenario $\mathbf{H}$ is an identity matrix $\mathbf{I}$ of size $(N \times N)$. Thus $\mathbf{H}$ has been omitted in (2.8).

This derivation is an extension of the basic formulation given by Kailath [49, 50], but on a chip by chip basis. However, most papers describe the signals on a symbol by symbol basis. This sometimes makes it difficult to understand how a receiver processes the signals, but there are articles which present a geometrical interpretation [51–53] in order to support a visual description for different receivers.

2.3.2 Symbol rate based receiver

The symbol rate based receiver presented in Figure 2.1(b) is first analysed for the non-dispersive channel and then extended to the multipath channel. Again, the elements in a set X are given by the rows of a matrix, here denoted as $\mathbf{R}$.

2.3.2.1 AWGN channel

Figure 2.1(b) shows that symbol rate based receivers process signals $r_u(k)$ for $u = 1, 2, \ldots, U$, where $\mathbf{r}(k)$ is the preprocessor output in vector notation. If the receiver is symbol synchronised, then vectors $\mathbf{y}(k)$ and $\mathbf{r}(k)$ are of interest. Again, all possible noise free signal (states), which can be fed into the receiver can be stored in a matrix. This $(M \times U)$ matrix $\mathbf{R}$ is derived from matrix $\mathbf{P}$ in (2.8). If the preprocessing stage consists of a bank of matched filters (each filter

[3] Dot product: $z = \mathbf{x}^T\mathbf{y}$; Hadamard: $\mathbf{z} = \mathbf{x} * \mathbf{y} \Rightarrow z_i = x_i y_i$ or $\mathbf{Z} = \mathbf{X} * \mathbf{Y} \Rightarrow z_{i,j} = x_{i,j} y_{i,j}$

matched to one of the U spreading sequences), then the preprocessed signal is said to have sufficient statistics [38, 54]. Thus, $\mathbf{R}$ is given by:

$$\begin{aligned} \mathbf{R} &= \mathbf{P}\mathbf{C}^T \\ &= \mathbf{B}\mathbf{C}\mathbf{C}^T = \mathbf{B}\mathbf{S}, \end{aligned} \tag{2.13}$$

where $\mathbf{B}$ is the combination matrix and $\mathbf{S}$ is the crosscorrelation matrix between the spreading codes, thus:

$$\mathbf{S} = \mathbf{C}\mathbf{C}^T = \begin{bmatrix} \mathbf{c}_1^T\mathbf{c}_1 & \mathbf{c}_1^T\mathbf{c}_2 & \dots & \mathbf{c}_1^T\mathbf{c}_{U-1} & \mathbf{c}_1^T\mathbf{c}_U \\ \mathbf{c}_2^T\mathbf{c}_1 & \mathbf{c}_2^T\mathbf{c}_2 & \dots & \mathbf{c}_2^T\mathbf{c}_{U-1} & \mathbf{c}_2^T\mathbf{c}_U \\ \mathbf{c}_3^T\mathbf{c}_1 & \mathbf{c}_3^T\mathbf{c}_2 & \dots & \mathbf{c}_3^T\mathbf{c}_{U-1} & \mathbf{c}_3^T\mathbf{c}_U \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{c}_{U-1}^T\mathbf{c}_1 & \mathbf{c}_{U-1}^T\mathbf{c}_2 & \dots & \mathbf{c}_{U-1}^T\mathbf{c}_{U-1} & \mathbf{c}_{U-1}^T\mathbf{c}_U \\ \mathbf{c}_U^T\mathbf{c}_1 & \mathbf{c}_U^T\mathbf{c}_2 & \dots & \mathbf{c}_U^T\mathbf{c}_{U-1} & \mathbf{c}_U^T\mathbf{c}_U \end{bmatrix}. \tag{2.14}$$

If the spreading codes are orthogonal, then the main diagonal of $\mathbf{S}$ contains ones and all the off diagonal elements in $\mathbf{S}$ are zero, yielding $\mathbf{R} = \mathbf{B}$. If $\mathbf{R} = \mathbf{B}$ then $\mathbf{R}$ consists (only) of vertices (corners) of a hypercube, and the signal constellation is convex [55]. A more elaborate analysis will be given in the following chapters.

The preprocessing stage (block) shown in Figure 2.1(b) for a multipath channel, can have different architectures, and each architecture has its unique properties. Since these methods are discussed in the forthcoming chapters, the discussion shall be omitted at this point for brevity.

2.4 Commercial system

Currently, several CDMA based mobile satellite systems are being introduced, e.g. Iridium, Globalstar [12, 56]. Moreover, proposals have been made for the global third generation mobile standards based on wideband CDMA [6, 57] and TDMA-CDMA [10, 58]. Thus it looks as if most mobile systems in the future will have an element of CDMA. In the US and some Asian countries a CDMA based cellular system is already in use, based on IS-95.

IS-95 is a digital cellular standard endorsed by the US Telecommunications Industry Association based on CDMA [16]. The IS-95 system allows a number of users within a cell to use

the same channel, and users in adjacent cells also use the same channel. It completely eliminates the need for frequency planning within a geographical area. Spectral allocations are in the 800-900 MHz and 1800 MHz region for two operators. Both operators have 12.5 MHz of spectrum in each direction, this is divided into 1.25 MHz channels, giving 20 full duplex carriers. The mobile station transmit frequency is always 45 MHz lower than the base station transmit frequency. Different cells use different phases of a short PN code with period $(2^{15} - 1)$. Each carrier is divided into a maximum of 63 information channels using orthogonal spreading codes (Walsh codes). The user data is convolutional coded and spread, which leads to a channel chip rate of 1.2288 Mchip/s, with a total spreading factor of 128 including *forward error correction coding* (FECC), where the maximum data rate is 9.6 kbps. In order to ensure privacy, user data is encrypted by a user specific long PN sequence with a period of $(2^{42} - 1)$. At both the base station and the mobile RAKE [46] receivers are used. A new feature of IS-95 is the ability to provide soft handover, connections of a mobile to two base stations at the same time. To solve the near-far problem in the uplink, open and closed loop power control systems are used.

2.4.1 Downlink

The downlink channel consists of a pilot, a synchronisation, up to 7 paging and up to 63 traffic channels. Of special interest is the pilot channel, which allows a mobile to acquire the timing of the channel, provides a phase reference for coherent demodulation, and provides each mobile with a means for signal strength comparisons between base stations to determine when to handoff. The pilot channel is also modulated by the base station specific short PN code. The traffic channel contains the data, which is grouped into 20 ms frames. User data is first convolutionally coded, interleaved and formatted to adjust the data rate, and then spread with a Walsh code and encrypted with a long PN sequence.

2.4.2 Uplink

The uplink channel consists of access and traffic channels. The access channel is used by the mobile to initiate communication with the base station and to respond to paging messages. User data is grouped into 20 ms frames. All data transmitted is convolutionally encoded, block interleaved, modulated by an orthogonal modulation, and spread prior to transmission. The user data may vary between 1200 bps and 9600 bps.

2.5 Summary

Established multiuser communication systems are based on FDMA or TDMA, and will reach their capacity limit soon. Military and efficiency considerations lead to the development of SS systems, which became possible with the development of suitable hardware such as digital circuitry and *digital signal processors* (DSPs). Since SS techniques are more robust to interference than common techniques, they are well suited to the difficult mobile environment [59].

To satisfy the growing demand on mobile communications, SS has been applied for mobile communications in the form of DS-CDMA , which enables service providers to offer higher capacities at lower costs. The third generation mobile standards and the satellite based mobile systems are highly likely to be based on DS-CDMA [3, 60].

This work is focusing on receivers for mobile handsets. This is partly due to the tough limitations on available computer power, hardware size and costs, and also due to the fact that the current receiver structure deployed is far from optimum. Generally, linear receivers do not perform well in DS-CDMA since the optimum processing task is nonlinear, thus, nonlinear signal processing techniques (e.g.[61]) must be used in order to improve receiver performance.

Chapter 3
Established receivers

This chapter reviews equalisers and receiver structures for digital communications and DS-CDMA.

After an introduction to optimum signal detection and a survey on equalisers, a review is given on optimum, linear and nonlinear receivers for DS-CDMA. Finally, the chapter will be summarised.

3.1 Equalisers

In the mobile environment, the radio channel can have a non-flat frequency response and nonlinear phase responses in the signal passband. Sending digital data at high speed through these channels often results in ISI, caused by signal pulse smearing in the dispersive medium. Thus, ISI is associated simply with the time delay between first and last significantly large echoes. One of the earliest paper concerned with ISI was due to Price [62], where he introduced the concept of the RAKE receiver. This structure is still popular [46], although other techniques have been developed. However, ISI is not the only obstacle in mobile CDMA communications. CDMA receivers must also take MAI into account and should be near-far resistant [35, 63, 64]. All of these factors must be taken into account, while maintaining reasonable complexity.

Equalisation techniques may be divided into two general types, linear and nonlinear equalisers, see Figure 3.1.

Equalisation in data modems combats ISI by filtering the incoming signals. When the filter weights become a channel inverse, then the filter can compensate for the irregularities in channel magnitude and phase response. A non-stationary environment requires equalisers, which can follow the changing channel impulse response [65]. Hence, the equaliser must be adaptive [65–70]. A discussion on adaptive algorithms such as the *least mean square* (LMS) or the *recursive least square* (RLS) can be found in Haykin [48].

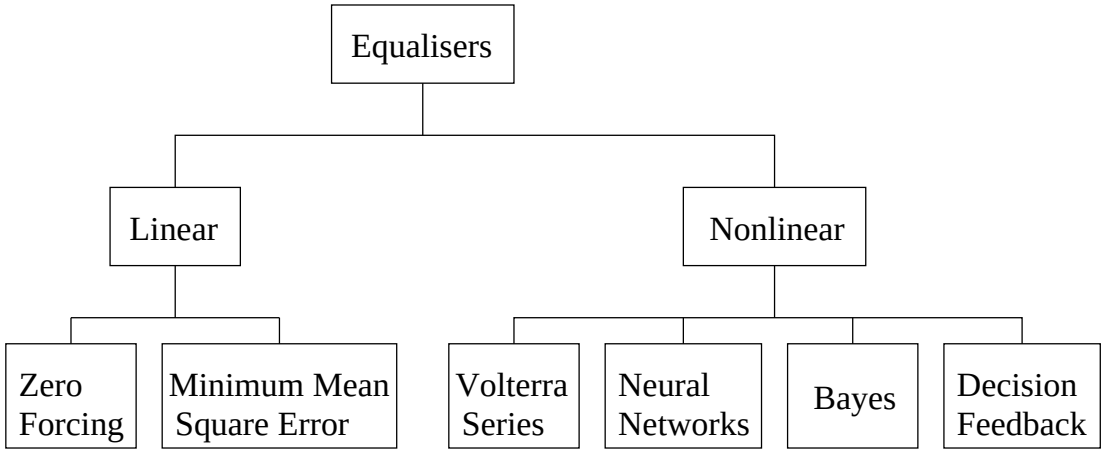


Figure 3.1: A possible grouping of equaliser structures.

3.1.1 Optimum detector

Optimum receivers can be compared according to their decision criterion [71]. The Bayesian [72] receiver's decision criterion, for a signal y and binary signalling, where no assumptions are made, is given by:

$$\frac{p(y|0)}{p(y|1)} \underset{0}{\overset{1}{\lessgtr}} \frac{C_0\ P(1)}{C_1\ P(0)}, \tag{3.1}$$

where C_i is the ith cost of deciding wrongly for i, $p(y|i)$ the conditional *probability density function* (pdf) and $P(i)$ the *a priori* probability. If the costs are unknown and the assumption is made such that $C_0 = C_1$, then the *maximum a posteriori* (MAP) decision criterion is given as:

$$\frac{p(y|0)}{p(y|1)} \underset{0}{\overset{1}{\lessgtr}} \frac{P(1)}{P(0)}. \tag{3.2}$$

If neither the $a\ priori$ probability nor the costs are known and the assumption $C_0\ P(1) = C_1\ P(0)$ is made, then equation (3.2) is called a *maximum likelihood* (ML) decision criterion and becomes:

$$\frac{p(y|0)}{p(y|1)} \underset{0}{\overset{1}{\lessgtr}} 1. \tag{3.3}$$

The ratio of the pdfs on the left hand side of (3.1) is the *likelihood ratio* LR, and the Bayesian decision rule compares the likelihood ratio to a threshold δ, which is one in (3.1).

The minimum probability of error is achieved when the receiver guesses the transmitted signal to be that signal which, given the received signal plus noise, was most likely to have been transmitted. Such a receiver is called a ML receiver. The optimum decision rules, in the sense of minimising the probability of false detection, based on an observation vector $\mathbf{y}$ is given according to [38, 54]. A signal transmitted over an AWGN channel can be demodulated by N matched filter type filters and produces a vector $\mathbf{y} = [y_1 \; y_2 \ldots y_N]^T$ with sufficient statistics. The joint conditional probability density function of the received data vector is:

$$p(y_1, y_2, \ldots, y_N|\mathbf{s}_m) = \frac{1}{\sqrt{(\pi \; N_0)^N}} \exp\left(-\frac{\sum_{n=1}^{N}(y_n - s_{m,n})^2}{N_0}\right), \tag{3.4}$$

where $\mathbf{s}_m$ is a signal from a set of $m = 1, 2, \ldots, M$ possible signals. Maximising the exponent is equivalent to minimising the sum of squares of the distance measure between the received data and those possible signal vectors $\mathbf{s}_m$, resulting in maximising the probability. The wish is to design a signal detector such that the probability of a correct decision is maximised. The decision rule is based on the computation of the $a\ posteriori$ probabilities, defined as:

$$P(\text{signal } \mathbf{s}_m \text{was transmitted}|\mathbf{y}),$$

which shall be abbreviated as $P(\mathbf{s}_m|\mathbf{y})$. The decision criterion is based on selecting the signal corresponding to the maximum of the set of *a posteriori* probabilities $\{P(\mathbf{s}_m|\mathbf{y}) : \; m = 1, 2, \ldots, M\}$, which is the MAP criterion. Applying Bayes' rule, the *a posteriori* probabilities can be stated as:

$$P(\mathbf{s}_m|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{s}_m)\,P(\mathbf{s}_m)}{p(\mathbf{y})}, \tag{3.5}$$

where $p(\mathbf{y}|\mathbf{s}_m)$ is the likelihood function that results when symbol $\mathbf{s}_m$ is transmitted, and $P(\mathbf{s}_m)$ is the prior probability of the mth signal being transmitted. Some simplifications can be made in the MAP criterion, when the M signals are equally probable prior ($P(\mathbf{s}_m) = 1/M$ for all m). Furthermore, if the denominator in (3.5) is independent of which signal is transmitted, then the decision rule can be stated as finding the signal that maximises $p(\mathbf{y}|\mathbf{s}_m)$, which is the ML

criterion. The likelihood function of $p(\mathbf{y}|\mathbf{s}_m)$ can be given in its natural logarithm form:

$$\ln(p(\mathbf{y}|\mathbf{s}_m)) = -\frac{N}{2}\ln(\pi N_0) - \frac{1}{N_0}\sum_{n=1}^{N}(y_n - s_{m,n})^2. \tag{3.6}$$

The first term in (3.6) is a constant and can be ignored, hence, the maximum of $\ln(p(\mathbf{y}|\mathbf{s}_m))$ over $\mathbf{s}_m$ is given when the Euclidean distance is minimum. This shows that the ML decision rule is simply to choose the signal closest to the received signal in terms of the Euclidean distance. The second term in (3.6) can further be split into three terms: a term independent of m (which can be ignored), the inner product of $\mathbf{y}$ and $\mathbf{s}_m$, and the energy of the signal. So an equivalent decision rule (for the AWGN channel) is selecting the signal that maximises:

$$2\,\mathbf{y}^T\mathbf{s}_m - |\mathbf{s}_m|^2, \tag{3.7}$$

where the first term is the correlation between the received vector and the mth signal. If all signals have equal energy, $|\mathbf{s}_m|^2$ may also be ignored. Hence, the optimum ML detector for equally probable signals computes a set of distances and selects the signal with the smallest distance.

A well known ML receiver is Forney's [73] *maximum likelihood sequence estimator* (MLSE), which comprises a sampled linear filter, called a whitened matched filter [74], and a recursive nonlinear processor, called the Viterbi algorithm [75] for the distance measure. This equaliser is often considered as the optimum receiver in literature, although the optimum receiver is a symbol by symbol detector which incorporates the MAP criterion, e.g.[76, 77]. Moreover, recently it has been shown that a MAP receiver outperforms the MLSE receiver, see [65, 78].

Optimum detectors are generally complex and computationally expensive. Thus the aim is to find a receiver structure with near optimum performance and reasonable complexity, so called *sub- optimum* or *near-optimum* receivers.

3.1.2 Linear equalisers

In the early work on linear equalisation, equalisers were generally designed to minimise the peak distortion. This is achieved, under certain conditions, by a technique known as *zero forcing* (ZF) [79]. Here, the equaliser forces the receiver output to be zero at all the sampling instances, except at the time instant that corresponds to the transmitted pulse. The shortcoming of the ZF

is that the linear filter removes ISI without regard to the effect on the noise. The result is that in eliminating the ISI (or a portion thereof) it necessarily enhances the noise [80]. Moreover, this technique cannot be used in its simple form for equalising severely time varying signal distortion [79].

Another form of optimising the filter design is by minimising a *cost function*. The most common criteria is the *mean square* value of the estimation *error* (MSE), because it leads to tractable mathematics. The *minimum mean square error* (MMSE) J_{min} is defined by:

$$J_{min} = E[|e(k)|^2], \tag{3.8}$$

where $e(k) = d(k) - \widehat{y}(k)$, and $\widehat{y}(k)$ denotes the estimated (filter output) for the kth symbol of the desired response $d(k)$[1], and $E[\cdot]$ denotes the expected value. The optimum, in the MSE sense, is given by the Wiener-Hopf equations [48]. They define the linear filter coefficients (or weights) in terms of two correlation functions: the autocorrelation function of the filter input, and the crosscorrelation function between the filter input and the desired response. The matrix formulation of the Wiener-Hopf equation is:

$$\mathbf{w} = \mathbf{R}_{yy}^{-1}\mathbf{r}_{xy}, \tag{3.9}$$

where $\mathbf{w}$ are the filter weights or coefficients. $\mathbf{R}_{yy}$ denotes the autocorrelation matrix of the received signal defined as $\mathbf{R}_{yy} = E[\mathbf{y}(k)\mathbf{y}^T(k)]$ where $\mathbf{y}$ is the $(N \times 1)$ input vector, and $\mathbf{r}_{xy} = E[d(k)\mathbf{y}(k)]$ is the crosscorrelation vector between the desired response and the filter input. This is the most popular equaliser, known as the feedforward *linear transversal equaliser* (LTE). An adaptive form of this equaliser is known as the Widrow-Hoff LMS, which converges to the MMSE solution [48, 79].

3.1.3 Nonlinear equalisers

A nonlinear structure known in nonlinear system modelling has been suggested, based on the Volterra[2] series [81], since its inverse can be used to compensate for nonlinear distortion. The Volterra series expansion can be seen as a Taylor series expansion with memory. The limitations of the Volterra series expansion are similar to those of the Taylor series expansion: they do

[1] $d(k)$ represents the known transmitted signal.
[2] Vito Volterra, Ital. mathematician (1860-1940).

not perform well, when there are discontinuities in the system description [82]. Taylor series converge and represent a function, e.g. $f(x)$, if and only if the remainder of the Taylor series becomes zero as the order of the Taylor series tends to infinity [83]. If the order of the system is very high or even infinite, then, generally, the contribution of each Volterra operator cannot be separated from the total system response. For this case, there is no exact method for the measurement of the system Volterra kernels, and approximation techniques must be used [81, 82].

The continuous time Volterra model is given by:

$$v(t) = H_0 + H_1[x(t)] + \cdots + H_m[x(t)] + \cdots + H_M[x(t)], \tag{3.10}$$

where H_0 is the DC term, $H_m[x(t)]$ is the mth-order Volterra operator given by (3.11), and $h_m(\tau_1, \tau_2, \ldots, \tau_m)$ is the mth-order Volterra kernel:

$$H_m[x(t)] = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} h_m(\tau_1, \tau_2, \ldots, \tau_m) x(t-\tau_1) \cdots x(t-\tau_m) d\tau_1 \cdots d\tau_m. \tag{3.11}$$

Volterra series have been applied to equalisation [84, 85], image processing [86], and also their adaptive behaviour has been investigated [82, 87, 88]. However, a Volterra structure can become computationally expensive, which is why most Volterra filter proposals use low order (up to 3). Sid and Figueiras [89] showed an equalisation example for a maximum phase channel, which is known to be difficult to equalise [79, 90] and may require a prohibitively large equaliser structure. In order to construct the optimum decision boundary, it required a Volterra structure of order 17. However, this scenario is not typical. Tsimbinos and Lever [91] investigated the Volterra FIR filter complexity $C(O, M)$ in terms of multiplications needed. Figure 3.2 shows the true complexity $C(O, M)$, where O denotes the highest filter-order and M the memory span, in terms of the total *floating point operations* (flops) needed. However, current DSPs need one clock cycle for a *multiply and accumulate* (MAC) operation, therefore it is reasonable to count the multiplications only. Note that the complexity scale is logarithmic.

Gibson *et al.* [90] and Georgoulakis and Theodoridis [92] considered the equalisation problem as a classification problem. The equaliser's task is to construct the boundary which minimises classification errors. Gibson *et al.* showed that the optimum decision boundary is not generally linear and hence nonlinear equaliser structures can outperform linear ones. Results were given in [90] for the LTE and a *multilayer perceptron* (MLP) *neural network* (NN) [93], which indic-

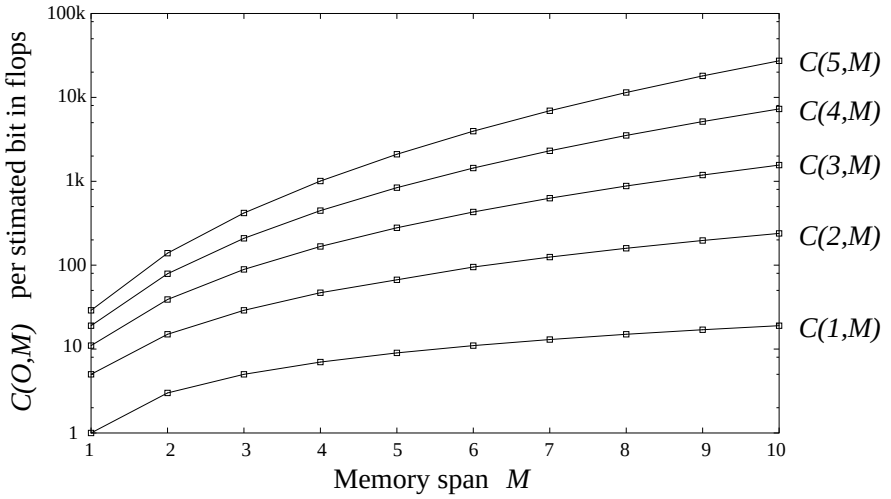


Figure 3.2: Complexity $C(O, M)$ per estimated user bit (receiver output) against the memory span M at the FIR filter; M corresponds to the Volterra filter length (taps) and parameter O is the Volterra-order. $C(1, M)$ is the complexity curve for a FIR filter with M-taps.

ate the superiority of the nonlinear NN structure over the LTE. However, MLP networks must be trained according to an algorithm (e.g. *back propagation* (BP) [94]). This training can take a very long time, hence the MLP is not suited to real time applications. In their paper, Gibson *et al.* also showed that under certain circumstances, when the channel is not of minimum phase nature, the equalisation problem becomes nonlinearly separable. This means, any linear equaliser will fail to perform well regardless of the SNR, and only nonlinear structures succeed in achieving good performance.

An approach is given in Hassell *et al.* [95] to tackle the nonlinear separation problem. They applied *linear programming* (LP) [96] in order to construct slabs[3]. Results presented indicate little performance loss against the optimum Bayesian equaliser. However, due to the complex nature of the problem a higher SNR was often needed to achieve good performance. Figure 3.3 shows the results reproduced according to [95] for two different channel models, where Channel 1 has impulse response $H_{ch}(z) = 0.5 + z^{-1}$ and Channel 2 $H_{ch}(z) = 0.333 + 0.666z^{-1} + z^{-2}$. Clearly, the LTE can no longer resolve the nonlinear separable scenario and fails to achieve good performance since its structure is linear. The proposed LP based equaliser (SLAB) can perform nearly as well as the optimum Bayesian equaliser.

[3]The vector space is sub-divided with hyperplanes into regions (bounded by hyperplanes) with a joint property, which look like slabs.

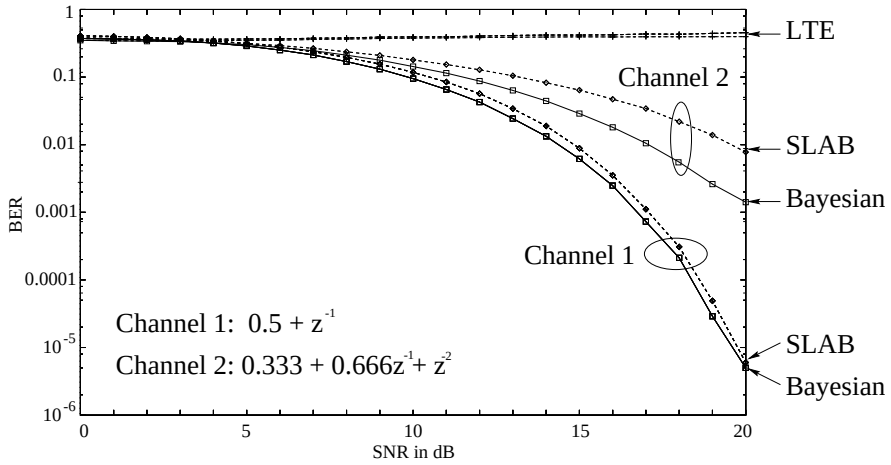


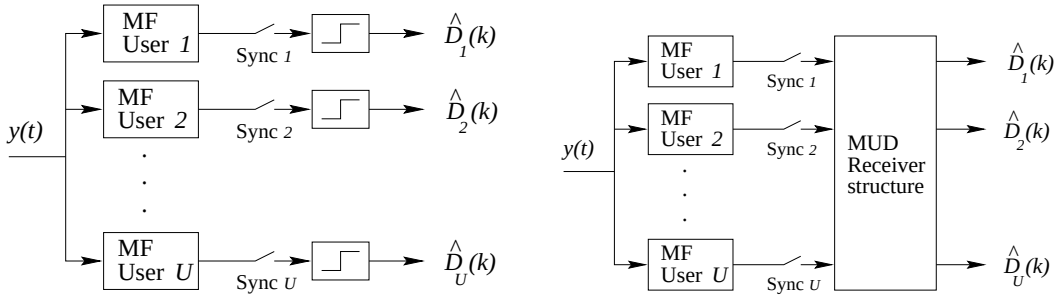
Figure 3.3: Hassell's proposed slab receiver compared against the linear LTE and the Bayesian receiver for two nonlinearly separable scenarios.

Another approach for NN based equalisers is based on approximation [97, 98]. Here, *radial basis function networks* (RBFN) were considered, which can reconstruct any hypersurface boundary [98]. Chen *et al.* [99] proposed the adaptive RBF equaliser and concluded that its performance depends on the selection of the RBF centres. In [100], Chen *et al.* presented a supervised clustering technique. Moreover, a derivation has been given which showed the structural equivalence between the optimal Bayesian solution and the RBFN. This explains why NN equalisers outperform linear equalisers. Theodoridis *et al.* [101] also applied the RBFN and reduced the number of centres. Due to this reduction, the cluster distribution is no longer spherical. Hence, the Euclidean distance measure is no longer the appropriate choice, and was replaced by the Mahalanobis distance measure [102]. Finally, Cha and Kassam [103] examined a number of different RBF NN structures as well as training algorithms for interference cancellation.

3.2 Optimum multiuser receiver for DS-CDMA

A CDMA receiver can either process the received signal at the chip rate or symbol rate (user bit rate), see Figure 3.4. Figure 3.4(a) shows chip rate receivers, which consists of a bank of *matched filters* (MFs) or RAKEs. A bank of MFs is for the non-dispersive AWGN channel, whereas RAKEs are considered for multipath channels. Current mobiles have a simple RAKE because of its simplicity, whereas base stations can have a bank of MFs (or RAKEs) as depicted in figures 3.4(a) and 3.4(b). However, structure Figure 3.4(a) suffers from MAI and

therefore has limited performance. Performance improvement can be gained, when $carrier$ to $interference\ ratio$ (CIR) information from the interferers is taken into account to combat MAI, as structure 3.4(b) suggests. This structure is known as the *multiuser detector* (MUD) and is usually suggested for the asynchronous uplink receiver [104]. It could also be used in a modified version as a single user detector in mobiles and might be implemented in the next generation of mobile systems.



(a) *Conventional bank of single user receivers with MFs or RAKEs.*

(b) *Verdu's proposed multiuser detector scheme with MFs for the AWGN channel.*

Figure 3.4: Two different MUD receiver structures.

A receiver structure which processes the received signal $y(t)$ at the chip rate, will be known as a *chip level based* (CLB) receiver. Receivers, such as 3.4(b), which process $y(t)$ at the symbol rate and consist of a front end bank of filters, will be called *preprocessing based* (PPB) receivers.

Verdu [105] showed, based on Forney's paper [73], that the U user maximum likelihood detector for CDMA in AWGN consists of a bank of U single user matched filters followed by a Viterbi algorithm with a time complexity of $O(2^U)$ per bit (binary signalling). This receiver structure shall be called Verdu's receiver (for DS-CDMA). The complexity of the Verdu receiver makes it unsuitable for practical applications. Verdu's receiver structure is often considered in the literature as the optimum receiver for CDMA, in terms of minimising the number of incorrectly detected symbols. However, there is also evidence that a MAP based receiver outperforms Verdu's receiver, as has been proven for equalisers. Moreover, Jung and Alexander [52] investigated MUD receivers and also state that the optimum CDMA receiver, defined as the receiver with the least detection errors, is a MAP detector such as that in [77]. Thus, the optimum detector is a *maximum likelihood symbol detector* (MLSD) since it minimises the symbol error probability and not the sequence error probability as Verdu's receiver does.

The optimum receiver is also too complex to implement since it compares the received signal against all possible signal states.

Because all optimum receivers are too complex for practical applications, the search for simpler and near optimum receivers became vital and goes on. Most proposals are based on the multiuser concept, which is preprocessing based (PPB) for several reasons. First, they relate to Verdu's MUD receiver, since they consider it optimum. Second, when long spreading sequences are used, as suggested for *wideband* CDMA (WCDMA) [3], then matrix and vector manipulations become very expensive, whereas preprocessing reduces the signal dimensionality to a reasonable size, from N to U, since generally the number of users is smaller than the number of chips. Third, preprocessing filters (RAKEs) are fast, easy to build and embed into hardware as a simple block, while the output signal has sufficient statistics.

3.3 Linear receivers for DS-CDMA

The general form of a linear receiver is given by $\widehat{D}_d = \text{sgn}(\mathbf{w}^T\mathbf{y})$, where the sgn$(\cdot)$ function returns the sign of the operand and where the filter weight vector $\mathbf{w}$ is chosen to minimise a cost function, while $\widehat{D}_d$ is the estimated transmitted bit of the desired user d and $\mathbf{y}$ is the received signal, see Figure 2.1.

The simplest CDMA receiver is the MF receiver, where $\mathbf{w}$ is replaced by $\mathbf{c}_d$, the spreading sequence vector of the desired user. In a multipath fading channel, $\mathbf{w}$ corresponds to the convolution between $\mathbf{c}_d$ and H_{ch}, implemented as a RAKE [106]. The theoretical performance P_e of a MF receiver for a single cell system with U users, long random codes, where N is the number of chips (processing gain) in AWGN is [23]:

$$P_e^{MF} = Q\left(\sqrt{\frac{N}{\sigma^2 + (U-1)}}\right), \tag{3.12}$$

where $Q(\cdot)$[4] may be written in terms of the complementary error function as:

$$Q(x) = 0.5\,\text{erfc}(x/\sqrt{2}),$$

[4]Defined as: $Q(x) = \frac{1}{\sqrt{2\,\pi}}\int_x^\infty \exp(-z^2/2)dz$.

and σ^2 denotes the noise power, derived from:

$$\frac{E_b}{N_0} = \frac{N}{2\sigma^2}, \tag{3.13}$$

where $\sigma^2 = N_0/2$ is the two sided noise power spectral density and E_b the bit energy.

Minimising the MSE results in the optimal Wiener filter (MMSE), where the optimal weights are given by equation (3.9). The autocorrelation matrix of the signal of the input of the receiver for CDMA has been shown by Cruickshank [107] to be equivalent to $\mathbf{R}_{yy} = \mathbf{C}^T\mathbf{P}\mathbf{C} + \sigma^2\mathbf{I}$. The $(U \times N)$ matrix $\mathbf{C}$ contains the spreading codes as vector rows. The matrix $\mathbf{P}$ has size $(U \times U)$ and has P_u at its diagonal and is zero elsewhere, P_u being the signal power at the receiver of user u. Term $\sigma^2\mathbf{I}$ is the noise term. The crosscorrelation vector $\mathbf{r}_{xy}$ is given by $\mathbf{r}_{xy} = E[D_d\ \mathbf{r}]$, where D_d is the desired response and $\mathbf{r}$ the preprocessed received vector. This results in the spreading code of the desired user with the assumptions made in [107]. The theoretical performance of the MMSE receiver is [108]:

$$P_e^{MMSE} = Q\left(\sqrt{\frac{\mathbf{w}^T\mathbf{r}_{xy}}{1 - \mathbf{w}^T\mathbf{r}_{xy}}}\right). \tag{3.14}$$

To obtain the average BER performance, equation (3.14) must be averaged over all users in the system to give $\overline{P_e^{MMSE}}$. If the chip level based MMSE is applied to a multipath scenario, the spreading codes $\mathbf{c}_u$ used in the matrix $\mathbf{C}$ are replaced by the convolution between $\mathbf{c}_u$ and H_{ch}. The multiuser MMSE receiver has also been investigated by Madhow and Honig [109] and has been shown to be near-far resistant. A different derivation of the filter weights is presented next.

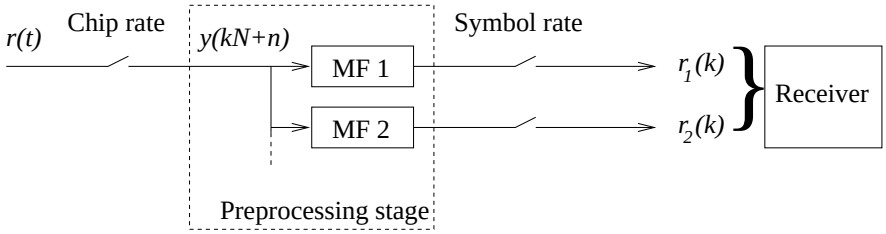


Figure 3.5: A bank of matched filters, the preprocessing stage, as for the optimum multiuser receiver suggested.

Figure 3.5 shows a synchronised two user system, from which the MMSE description is derived. Received signal (2.6) for bit k, here denoted as $y(kN + n)$, is preprocessed and fed as signal $\mathbf{r}(k) = [r_1(k)\ r_2(k)]^T$ into the receiver. For a three chip spreading code ($N = 3$) and $U = 2$,

vector $\mathbf{y}(k)$ has components (chips):

$$\begin{aligned} y_1 &= D_1 c_{1,1} + D_2 c_{2,1} + g(1), \\ y_2 &= D_1 c_{1,2} + D_2 c_{2,2} + g(2), \\ y_3 &= D_1 c_{1,3} + D_2 c_{2,3} + g(3), \end{aligned}$$

where D_u denotes the user bit, which is equiprobable +1 or -1 and $c_{u,n}$ is chip n of user (code) u which is either +1 or -1, and $g(\cdot)$ the added noise. Vector $\mathbf{r}(k)$ components are substituted with $a = r_1(k)$ and $b = r_2(k)$, and become after matched filtering:

$$\begin{aligned} \mathbf{r}(k) = [a \;\; b]^T & \\ a &= a_1 + a_2 + a_3 = y_1 c_{1,1} + y_2 c_{1,2} + y_3 c_{1,3} \\ b &= b_1 + b_2 + b_3 = y_1 c_{2,1} + y_2 c_{2,2} + y_3 c_{2,3}. \end{aligned}$$

The crosscorrelation vector $\mathbf{r}_{xr}$ is:

$$\begin{aligned} \mathbf{r}_{xr} &= E[D_d \mathbf{r}] \\ &= E[D_1 [a, b]^T] = [\{D_1 \times (a_1 + a_2 + a_3)\} \{D_1 \times (b_1 + b_2 + b_3)\}]^T, \end{aligned} \tag{3.15}$$

without loss of generality, it is assumed that user one is the desired user, hence $D_d = D_1$. Terms with factor D_u^i are one for even i (powers) and zero for odd i. Bearing this in mind, the first element in $\mathbf{r}_{xr}$ becomes

$$\begin{aligned} (\mathbf{r}_{xr})_1 &= E[D_1 \times (a_1 + a_2 + a_3)] \\ &= E[D_1 \times \{D_1 c_{1,1}^2 + D_2 c_{1,1} c_{2,1} + c_{1,1} g(1) + \\ &\quad D_1 c_{1,2}^2 + D_2 c_{1,2} c_{2,2} + c_{1,2} g(2) + \\ &\quad D_1 c_{1,3}^2 + D_2 c_{1,3} c_{2,3} + c_{1,3} g(3)\}] \\ &= c_{1,1}^2 + c_{1,2}^2 + c_{1,3}^2 \end{aligned}$$

and the second element is

$$(\mathbf{r}_{xr})_2 = E[D_1 \times (b_1 + b_2 + b_3)] = c_{1,1} c_{2,1} + c_{1,2} c_{2,2} + c_{1,3} c_{2,3}.$$

It turns out that the crosscorrelation vector of the MUD MMSE is the correlation between the

spreading sequence of the desired user and all transmitting users. The autocorrelation matrix $\mathbf{R}_{yy} = E[\mathbf{r}\mathbf{r}^T]$ is accordingly given by:

$$\mathbf{R}_{yy} = E\begin{bmatrix} a^2 & ab \\ ba & b^2 \end{bmatrix}, \tag{3.16}$$

with terms

$$\begin{aligned} a^2 &= (a_1 + a_2 + a_3)^2 = a_1^2 + a_2^2 + a_3^2 + 2a_1a_2 + 2a_1a_3 + 2a_2a_3 \\ b^2 &= b_1^2 + b_2^2 + b_3^2 + 2b_1b_2 + 2b_1b_3 + 2b_2b_3 \end{aligned}$$

and the two mixed terms, which can be assumed to be equal, are:

$$\begin{aligned} ab = ba &= (a_1 + a_2 + a_3)(b_1 + b_2 + b_3) \\ &= a_1b_1 + a_1b_2 + a_1b_3 + a_2b_1 + a_2b_2 + a_2b_3 + a_3b_1 + a_3b_2 + a_3b_3. \end{aligned}$$

Finally, with backsubstitution, the terms become:

$$\begin{aligned} a^2 = {} & c_{1,1}^4 + c_{1,2}^4 + c_{1,3}^4 + \\ & c_{2,1}^2c_{1,1}^2 + c_{2,2}^2c_{1,2}^2 + c_{2,3}^2c_{1,3}^2 + 2c_{1,1}^2c_{1,2}^2 + 2c_{1,1}^2c_{1,3}^2 + 2c_{1,2}^2c_{1,3}^2 + \\ & 2c_{1,1}c_{1,2}c_{2,1}c_{2,2} + 2c_{1,1}c_{1,3}c_{2,1}c_{2,3} + 2c_{1,2}c_{1,3}c_{2,2}c_{2,3} + \\ & c_{1,1}^2g(1)^2 + c_{1,2}^2g(2)^2 + c_{1,3}^2g(3)^2 \end{aligned}$$

$$\begin{aligned} b^2 = {} & c_{2,1}^4 + c_{2,2}^4 + c_{2,3}^4 + \\ & c_{2,1}^2c_{1,1}^2 + c_{2,2}^2c_{1,2}^2 + c_{2,3}^2c_{1,3}^2 + 2c_{1,1}^2c_{1,2}^2 + 2c_{1,1}^2c_{1,3}^2 + 2c_{1,2}^2c_{1,3}^2 + \\ & 2c_{1,1}c_{1,2}c_{2,1}c_{2,2} + 2c_{1,1}c_{1,3}c_{2,1}c_{2,3} + 2c_{1,2}c_{1,3}c_{2,2}c_{2,3} + \\ & c_{2,1}^2g(1)^2 + c_{2,2}^2g(2)^2 + c_{2,3}^2g(3)^2 \end{aligned}$$

$$\begin{aligned} ab = {} & c_{1,1}^3c_{2,1} + c_{2,1}^3c_{1,1} + c_{1,2}^3c_{2,2} + c_{2,2}^3c_{1,2} + c_{1,3}^3c_{2,3} + c_{2,3}^3c_{1,3} + \\ & c_{1,1}^2c_{1,2}c_{2,2} + c_{2,2}^2c_{1,1}c_{1,2} + c_{1,1}^2c_{1,3}c_{2,2} + c_{2,3}^2c_{1,1}c_{1,2} + c_{3,2}^2c_{1,1}c_{2,1} + \\ & c_{2,1}^2c_{1,2}c_{2,2} + c_{1,2}^2c_{1,3}c_{2,3} + c_{2,3}^2c_{1,2}c_{2,2} + c_{1,3}^2c_{1,1}c_{2,1} + c_{2,1}^2c_{1,3}c_{2,3} + \\ & c_{1,3}^2c_{1,2}c_{2,2} + c_{2,2}^2c_{1,3}c_{2,3} + c_{1,1}c_{2,1}g(1)^2 + c_{1,2}c_{2,2}g(2)^2 + c_{1,3}c_{2,3}g(3)^2. \end{aligned}$$

These results show that $\mathbf{R}_{yy}$ can be split into two matrices, the first covers all possible received noise free signals, and the second takes the correlated nature of the noise into account (due to preprocessing), hence:

$$\mathbf{R}_{yy} = \mathbf{P}^T\mathbf{P} + \sigma^2\mathbf{S}, \tag{3.17}$$

where $\mathbf{S}$ is the covariance matrix of the spreading codes, and $\mathbf{P}$ denotes a $(2^U \times U)$ matrix containing all possible received signal states as its rows. For the multipath case with RAKE preprocessing, $\mathbf{r}_{xr}$ is the correlation among the convolved sequences of the user codes $\mathbf{c}_u$ and H_{ch} and $\mathbf{S}$ the covariance matrix of the convolved sequences. Matrix $\mathbf{P}$ has size $(2^{3U} \times U)$ and contains all possible received signals of $\mathbf{r}$, taking the previous, current and next symbol into account, which is computationally prohibitive for a large U. A common approach is to apply MFs instead of RAKEs for preprocessing, which reduces the matrix size of $\mathbf{P}$ to $(2^{2U} \times U)$ at the expense of performance.

As the level of background noise tends to zero ($\sigma^2 \rightarrow 0$), the MMSE receiver converges to the *decorrelating detector* (DECO), introduced by Lupas and Verdu [110], which eliminates MAI at the expense of noise enhancement. This receiver structure can be considered as the ZF filter for CDMA. The decorrelating receiver is also near-far resistant [35].

Moshavi *et al.* [111] proposed a polynomial based structure, a *multistage* receiver. This structure converges to the Wiener solution as the number of stages tends towards infinity. The authors showed that a small number of stages is sufficient to outperform the MF receiver.

Simulation results for linear CDMA receivers are presented in figures 3.6 and 3.7. The simulation scenarios were set according to [112], with short randomly generated spreading sequences ($N = 7$) and noise power $E_b/N_0 = 7$dB. The simulation results show how the MF suffers from MAI and is outperformed by the decorrelating (DECO) and the MMSE receiver, see also [39].

3.4 Nonlinear receivers for DS-CDMA

A straightforward way to improve the performance of CDMA receivers is to cancel MAI. This class of receivers, known as *interference cancellers*, have drawn much attention. Most designs use combinations of linear receiver for pre and postfiltering since it keeps the complexity low [39, 107, 108, 113, 114]. However, in order to cancel MAI well, a good estimate of each

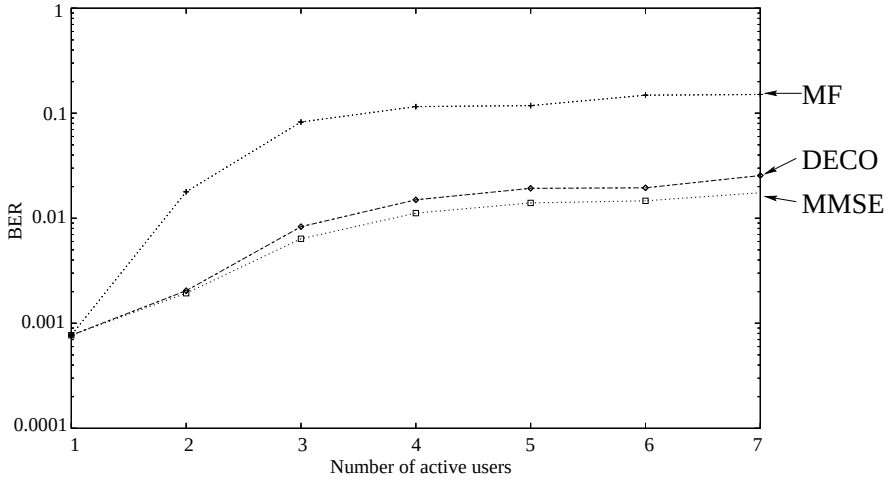


Figure 3.6: BER against the number of users in AWGN at $E_b/N_0 = 7$dB and with randomly generated spreading codes with 7 chips.

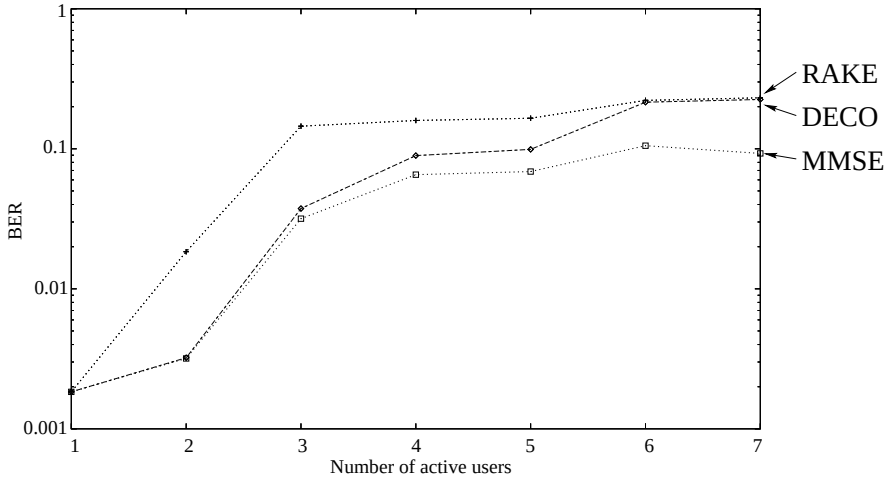


Figure 3.7: BER against the number of users in a stationary multipath, with $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$ at $E_b/N_0 = 7$dB and with randomly generated spreading codes with 7 chips.

users signal power is essential, otherwise interference is added and not subtracted.

The *multistage detector* (MSD) proposed by Varanasi and Aazhang [115] improves each stage's estimate by subtracting an MAI estimate obtained from the previous stage, a linear processing followed by an iterative nonlinear processing step. Kechriotis and Manolakos [116] showed that the MSD is a special case of a discrete time approximation of their proposed *Hopfield Neural Network* (HNN) [117]. Moreover, the HNN can correspond to an infinite number of stages of MSD sharing similar characteristics.

Miyajima and Yamanaka [118] proposed a *blind* adaptive receiver for DS-CDMA. Their receiver consists of a user specific MF and a bank of MFs which match the received signal against an orthonormal set of signals, derived from the users' spreading sequence. The filter bank output is fed into an adaptive block. The adaptive block generates a signal which is subtracted from the output of the user specific MF. The receiver, which works at the symbol rate in a Gaussian channel, was compared against Verdu's receiver and performed nearly as well for the used set of short Gold codes.

Aazhang *et al.* [119] investigated the MLP for high bandwidth efficiency[5]. Two training algorithms were considered. First the BP algorithm [94], where the supervisor has access to the desired user bit. This resulted in the incorrect decision boundary of the decorrelating receiver due to an improper choice of cost function, since it was based on minimising the *least square error* (LSE) which is not the same as minimising the number of misclassifications [120]. Thus they considered the assisted backpropagation algorithm, where the supervisor knows all user bits, which leads to a near optimum decision boundary. The open question of determining the number of neurons was solved by assigning a sufficient number of neurons by analysing the decision boundary first. Once again, the MLP NN showed excellent performance at the expense of long training time, computational complexity and network size uncertainty.

Hopfield networks are single layer recurrent neural networks with symmetric weight matrices in which the diagonal elements are ideally zero. *Hopfield neural networks* introduce the concept of an *energy function* into neural networks and are often viewed as a memory model, performing a recognition or retrieval task [93]. When viewed as a memory, the network's storage capacity (the maximum number of patterns that can be stored with acceptable error for retrieval) becomes vital. The capacity of a HNN is limited by a certain ratio between the number of

[5]Measured as the ratio of the number of users to spread factor.

patterns and the number of neurons; if this ratio is exceeded, then the HNN loses its memory. Nevertheless, the objective function of the multiuser problem [121] can be translated into an energy function, which makes HNN applicable for CDMA [116]. Results given in [122, 123] claim near optimum performance for many scenarios. A hybrid HNN with reduced complexity is presented in [123], and hardware aspects are investigated in [124], which suggest that the HNN might be an interesting alternative to linear structures. The HNN is also referred to as a *recurrent neural network* (RNN) [125]. Unlike MLP NN, RNN do not require iterative training since the connection weights can be computed in many cases but the network must be brought into a stable state. It appears that [116] and [125] are very similar. However, this RNN seems to perform as well as the optimum receiver since the network always converges to one vertex of the energy surface hypercube, which is close to, or, is the optimum solution. Teich and Seidl [126] extended this RNN for multipath scenarios and presented near optimum results. Their RNN complexity grows linearly with the number of users. They also state that the RNN requires only a few training iterations, dependent on ISI and MAI, to obtain the desired performance, but the performance degrades if too few iterations are used.

A derivative of the HNN is the *annealed neural network* (ANN) by analogy to the real annealing process in metallurgy. Such an annealing function can be used as a cost function, where the network is expected to find a configuration which minimises an energy function. Here, the HNN is viewed as solving an optimisation problem. Yoon and Rao [127] reported that the ANN performs better than the DECO.

Mitra and Poor [128] applied the RBF network for CDMA as an adaptive RBF since the MUD receiver may not have sufficient parameter knowledge. The RBFN consists of centres and weights, the centres were created (set up) by clustering techniques, and the weights were estimated with the adaptive LMS algorithm. A mathematical formulation is also given, which derives the weights and the relationship between RBF and ML detection. The received signal is $\mathbf{r} = \mathbf{s}_d + \mathbf{g}$, where $\mathbf{s}_d$ is the desired signal vector and $\mathbf{g}$ the noise component inclusive of MAI. The observation density conditioned on a hypothesis H_i, for $i = 0$ or 1 (binary signalling), is:

$$p(\mathbf{r}|i) = \frac{1}{2^{U-1}} \sum_{m=1}^{2^{U-1}} \frac{1}{\sqrt{(2\pi)^N \sigma^N}} \exp\left(-\frac{\|\mathbf{r} - \mathbf{s}_d(i) - \mu_m\|^2}{2\sigma^2} \right), \tag{3.18}$$

where U is the number of users, σ^2 is the Gaussian noise variance, $\mathbf{s}_d(i)$ is the desired signal

vector given H_i and μ_m is the mth permutation of the sum of the interferer's signal's[6]. The likelihood ratio becomes then:

$$LR = \frac{\sum_{m=1}^{2^{U-1}} \exp\left(-\frac{\|\mathbf{r}-\mathbf{s}_d(1)-\mu_m\|^2}{2\sigma^2}\right)}{\sum_{m=1}^{2^{U-1}} \exp\left(-\frac{\|\mathbf{r}-\mathbf{s}_d(0)-\mu_m\|^2}{2\sigma^2}\right)}, \tag{3.19}$$

from which the authors derived the Bayesian decision rule as:

$$\delta = \text{sgn}\left\{\sum_{m=1}^{2^{U-1}} \exp\left(-\frac{\|\mathbf{r}-\mathbf{s}_d(1)-\mu_m\|^2}{2\sigma^2}\right) - \sum_{m=1}^{2^{U-1}} \exp\left(-\frac{\|\mathbf{r}-\mathbf{s}_d(0)-\mu_m\|^2}{2\sigma^2}\right)\right\}. \tag{3.20}$$

The RBF network structure is defined as:

$$y(\mathbf{r}) = \sum_{m=1}^{M} w_m \, \phi\left(\frac{\|\mathbf{r}-\mathbf{c}_m\|^2}{\sigma_m}\right), \tag{3.21}$$

where $\phi(\cdot)$ is a continuous, nonlinear function from $\mathbb{R}^d \rightarrow \mathbb{R}$, $\mathbf{r}$ the input data vector, $\mathbf{c}_m$ is called a centre of the RBF neuron, and σ_m is the spread of the neuron and w_m are the linear weights. Each centre $\mathbf{c}_m$ corresponds to a row of a generating matrix. Mitra and Poor showed that the optimum Bayesian decision rule in terms of the MAP criterion, can be implemented with a RBF network, by substitution the terms in (3.21). However, it has been shown that the RBF is an optimum receiver for DS-CDMA. A non-adaptive RBF receiver for CDMA was proposed by Cruickshank [112]. Since all network coefficients were computed, this deterministic procedure does not require any training. However, the receiver must know the number of users, their spreading sequences and signal power, and the channel impulse response. Unfortunately, the mobile has only access to a limited number of parameters, whereas the base station does have knowledge of all of them. Moreover, in the multipath scenario, the receiver complexity grows with factor 2^{3U}. The simulations of [112] were repeated and presented in Figure 3.8 and 3.9, showing the superiority of the RBF receiver over the linear receiver structures (matched filter and MMSE receiver) in both cases, Gaussian and multipath scenario. The simulation has been stopped at six users for the multipath scenario due to the complexity of the RBFN.

[6]Means that the interfering signals can be constructed from the spreading codes allocated to each (interfering) active user.

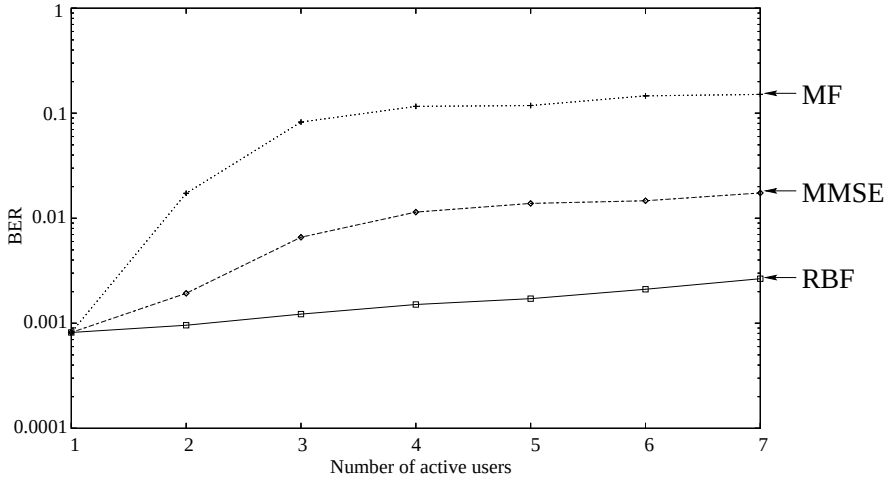


Figure 3.8: BER against the number of users in AWGN at $E_b/N_0 = 7$dB for randomly generated spreading codes with 7 chips.

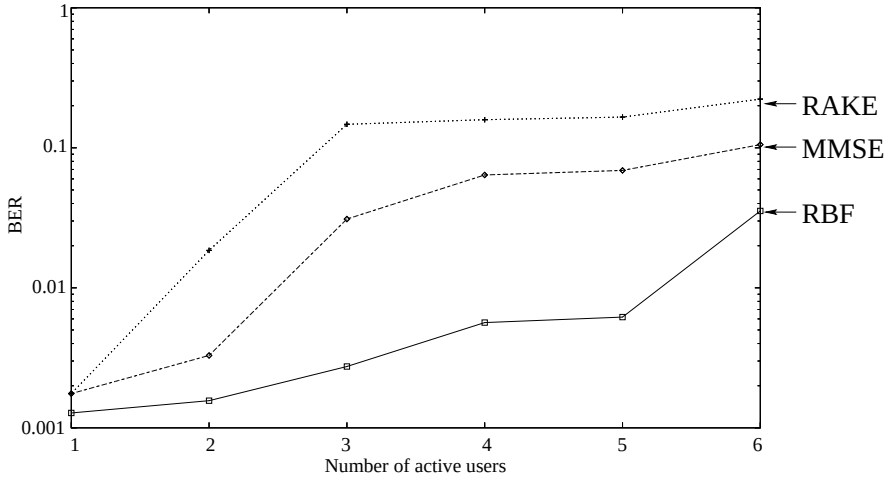


Figure 3.9: BER against the number of users in a stationary multipath, with $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$ at $E_b/N_0 = 7$dB and for randomly generated spreading codes with 7 chips.

3.5 Summary

Due to multipath effects the orthogonality among the spreading codes at the receiver is destroyed and linear filters are no longer optimum. The optimum receiver is nonlinear. It has been shown that nonlinear equaliser structures can be applied successfully to DS-CDMA. Optimum and sub-optimum nonlinear receivers appear to be complex, but the trend in increasing hardware power makes them interesting in future applications. A fact which has not been considered is that most receivers are required to be adaptive, due to the fact that the channel is not stationary and must be tracked. The non-stationary mobile environment needs an adaptive receiver structure which can track the fast changing channel characteristics. Linear (and feedback) receiver structures are in this respect advantageous, which explains their current dominance [129, 130].

Further work should consider investigating nonlinear network structures which have some sort of determinism incorporated such as multistage detectors, hybrid networks, Hopfield networks and radial basis function networks. HNNs appear very interesting since it has been pointed out that there is a relationship with the optimum detector and fast hardware is available. However, they require a loop too in order to converge to the optimum network state (training). However, the authors claim performances as being close to the MLSE's performance. This may be questioned because the HNN can be related to the MSD receiver, and the MSD was shown to perform like the MMSE receiver [131]. Also of interest are RBF networks since they perform a nonlinear and a linear mapping, can be made adaptive and can be constructed (no training).

Among the many equaliser designs proposed, Volterra and RBF networks are of interest since little work has been published concerned their application to DS-CDMA. Thus both structures appear worthwhile for further investigation.

Chapter 4
Pattern recognition in DS-CDMA

This chapter introduces the concept of pattern recognition and shows how it can be applied to digital communications such as DS-CDMA.

After a brief historical introduction, common techniques known in pattern recognition are reviewed. Then, pattern recognition is applied to DS-CDMA. Decision boundaries are presented, followed by an introduction to approximation theory. Finally, a discussion with conclusions summarises this chapter.

4.1 Introduction

Any signal, in particular digital signals, can be viewed as a pattern. A signal or pattern, given as a vector, can be interpreted as a point in the vector space. Points with similar properties can be assigned to sets, groups or classes. A machine's task is then to store, retrieve, compare, assign or recognise patterns. Work done in the field of neuro biology prepared the way to develop brain like computers: neural networks.

The very early models of neurons and neural networks, such as McCulloch-Pitts, were based on simple threshold devices to perform logic functions [93]. Later, Rosenblatt proposed *perceptrons* [132] and Widrow introduced *adalines* (adaptive linear elements) [133]. Since these models were of a linear nature, the most important requirement was linear separability [134] of the classes to apply linear networks [135–137]. Mangasarian [138] applied linear programming (LP), already well known in *operations research*, for testing linear separability and recently for cancer cell diagnosis [139]. However, the LP algorithm can be very time consuming when applied to large networks.

A network which shall perform well as a pattern recogniser, must be able to construct decision boundaries, or must have a memory. These are typical features of neural networks. Thus, common network solutions are *associative memory networks*, e.g. HNNs, MLPs, RNNs and

RBFNs. The increasing obtainable computer power will increase NN applications. The current trend in pattern recognition is applying nonlinear models, where popular approaches rest upon neural networks [140–143] and RBFs [144], encouraged by industrial applications [145]. Nevertheless, linear models (and machines) are still important, since they are very simple, fast and therefore cheap.

4.2 Pattern recognition

Pattern recognition means identifying, classifying and assigning a signal (e.g. represented by a point) to a known class. If the signal is antipodal then the network deals with two classes or groups, representing the desired user's bit sign.

Usually, points within a group form a cluster[1]. Clusters can be sharp bounded or disperse into other clusters. Assume there are M known patterns each denoted as vector $\mathbf{x}_m = [x_1\ x_2 \ldots x_d]^T$ ($m = 1, 2, \ldots, M$) in an Euclidean space $\mathbb{E}^d$ and each pattern belongs to a class denoted as ω_i. Further, there exists an infinite number of patterns when they are noise corrupted, denoted as $\mathbf{x}_k$, where the kth pattern originally has been drawn from the set with M patterns. The first class of pattern clusters is deterministic in nature and can be described by its (convex) hull [146, 147], whereas the second class is probabilistic and may be described by its statistics [144]. Signals in digital communications belong to the probabilistic class because of the added (Gaussian) noise. Obviously, if the convex hull is known, then the cluster can be described more easily which reduces the network complexity by a large amount if the set is large. For describing probabilistic classes, it makes sense to find the boundary between the classes, since this is the region where classification errors occur. Ideally, to find these decision boundaries, knowledge of the points and the noise statistics should be available to the detector, the pattern classifier.

[1] In pattern recognition is it more common to use the term point, instead of signal, since any signal in vector form can be interpreted as a point in the vector space.

4.2.1 Linear discriminant function

The simplest way to classify two classes of patterns is by applying a *linear discriminant function*, defined as:

$$f(\mathbf{x}_k) = \sum_{i=1}^{d} w_i x_{k,i} + w_0, \tag{4.1}$$

where w_i are the weights and w_0 denotes an offset. Figure 4.1 presents a linear machine with a classifier at the output which is not stated in equation (4.1) but introduced in the next subsection. A complete specification of any linear discriminant function is achieved by specifying the values of the weights.

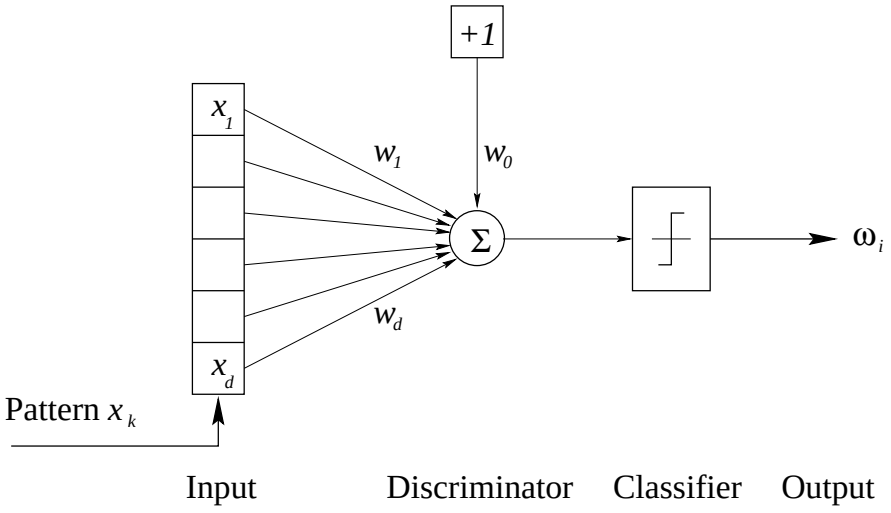


Figure 4.1: A model of a linear machine.

An important special case of a linear machine is a minimum distance classifier with respect to points in $\mathbb{E}^d$ [148–150].

4.2.2 Linear pattern classification

Equation (4.1) is better known from its modified version as perceptrons and adalines, where the offset w_0 is called the threshold and the sign of the sum is taken, so (4.1) becomes a *linear threshold element* [151], depicted in Figure 4.1 and defined as:

$$f(\mathbf{x}_m) = \operatorname{sgn}\left(\sum_{i=1}^{d} w_i x_{m,i} + w_0\right). \tag{4.2}$$

Equation (4.2) can distinguish in its simple form between two sets (classes), those which have a positive output, and those with a negative output, which corresponds to matched filtering.

Ideally, the separation of a two class problem is carried out with a matched filter, which generates a simple *plane* or *hyperplane* [152]. But if the problem is not of a simple linear nature (e.g. orthogonal points), then the optimum solution becomes more complex. Figure 4.2 presents several possible pattern configurations and a possible separation boundary, depicted as a dashed line.

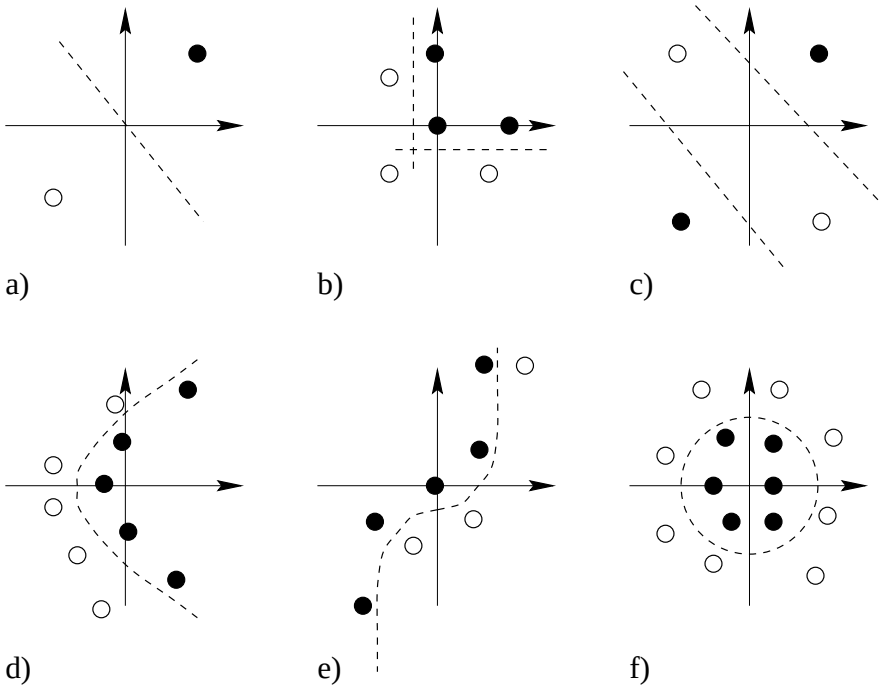
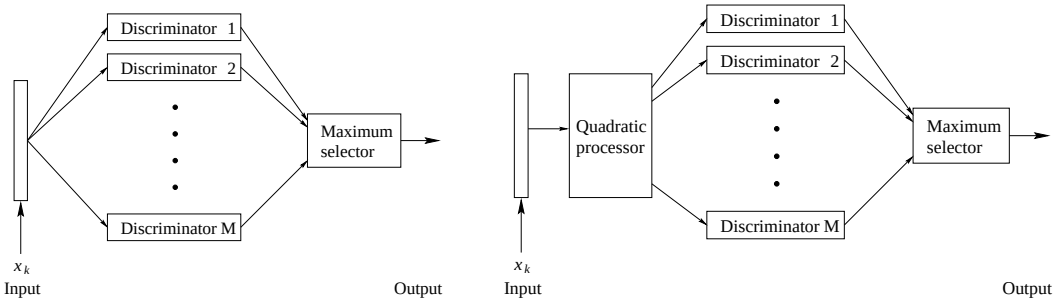


Figure 4.2: Different classes of pattern arrangements and a decision boundary, where b-f are nonlinearly separable. a) Linearly separable b) Piecewise linear c) Double bounded d) Quadratically e) Cubically f) Elliptically

Figure 4.2(a) is the simplest way to classify the two classes of dots and 4.2(f) the most complex way. All patterns could also be separated linearly by piecewise linear approximation, of course with increasing complexity. Such a piecewise linear machine is depicted in Figure 4.3(a). The machine generates M linear segments with the discriminator (4.2). Since the number of linear segments required, e.g. for approximating a circle, can generate a prohibitively large network, alternative functions which describe or approximate nonlinear functions, such as polynomial functions [153] and series should be considered, e.g. Figure 4.3(b).



(a) *Piecewise linear decision boundaries.* **(b)** *Quadratically nonlinear decision boundaries.*

Figure 4.3: Two models of nonlinear machines which generate decision boundaries.

4.2.3 Minimum distance classifier

The minimum distance classifier compares each pattern $\mathbf{x}_m$ against prototype patterns $\mathbf{p}$. This network becomes large as their number increases. Hence, minimum distance classifiers are appropriate in situations, where each class is represented by a single prototype pattern, e.g. circles, centroids, around which all other patterns in the class cluster [151]. This is already known since this technique corresponds to the ML technique, introduced in equation (3.3)[2].

4.2.4 Linear separability in pattern recognition

Starting from a pattern vector $\mathbf{x} = [x_1\ x_2 \ldots\ x_d]^T$ belonging to one of the M classes $\{\omega_1, \omega_2, \ldots, \omega_M\}$ which needs to be classified, there are M decision functions $f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})$ with the property that if a pattern $\mathbf{x}$ belongs to ω_i, then $f_i(\mathbf{x}) > f_j(\mathbf{x})$, $j = 1, 2, \ldots, M$ and $i \neq j$. The decision boundary for separating ω_i from ω_j is given by values of $\mathbf{x}$ for which $f_i(\mathbf{x}) = f_j(\mathbf{x})$, thus:

$$f_{ij}(\mathbf{x}) = f_i(\mathbf{x}) - f_j(\mathbf{x}) = 0, \tag{4.3}$$

which describes a hyperplane $\mathcal{H}$. Therefore, for patterns of class ω_i $f_{ij}(\mathbf{x}) > 0$ and for patterns of class ω_j $f_{ij}(\mathbf{x}) < 0$, see Figure 4.4.

The mentioned minimum distance classifier exploits the Euclidean distance $L_j(\mathbf{x}) = \|\mathbf{x} - \mu_j\|$, where $\mathbf{x}$ is assigned to the class ω_i if $L_i(\mathbf{x})$ is the smallest distance. Term μ_j is the mean of a

[2]Nilsson pointed out that the terms correlation detection and matched filtering are also common.

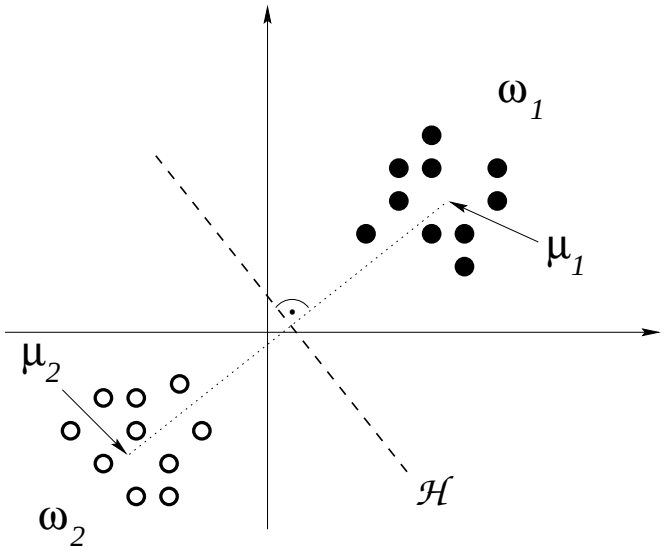


Figure 4.4: A possible 2-dimensional pattern classification problem, where two classes are separated by the linear boundary $\mathcal{H}$.

class given as:

$$\mu_j = \frac{1}{N_j} \sum_{\mathbf{x} \in \omega_j} \mathbf{x},$$

and N_j denotes the number of patterns within class ω_j. The decision boundary (4.3) between classes ω_i and ω_j becomes then (after several transformations):

$$f_{ij}(\mathbf{x}) = \mathbf{x}^T(\mu_i - \mu_j) - \frac{\mu_i^T \mu_i - \mu_j^T \mu_j}{2} = 0. \tag{4.4}$$

A Bayesian classifier for a (0,1) loss function can be seen as the implementation of decision functions of the form $f_j(\mathbf{x}) = p(\mathbf{x}|\omega_j)\,P(\omega_j)$ [154, 155]. For the 1-dimensional Gaussian case (*normal distribution*), and $M = 2$, $f_j(\mathbf{x})$ becomes:

$$f_j(\mathbf{x}) = p(\mathbf{x}|\omega_j)P(\omega_j) = \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left(-\frac{(\mathbf{x} - \mu_j)^2}{2\sigma_j^2}\right) P(\omega_j), \tag{4.5}$$

if all classes are equally likely, $P(\omega_j)$ can be neglected. For the d-dimensional case, the *multivariate normal distribution* [155], the density of $\mathbf{x}$ in the jth class is:

$$p(\mathbf{x}|\omega_j) = \frac{1}{\sqrt{(2\pi)^d}\sqrt{|\mathbf{S}_j|}} \exp\left(-\frac{(\mathbf{x} - \mu_j)^T\, \mathbf{S}_j^{-1}\, (\mathbf{x} - \mu_j)}{2}\right), \tag{4.6}$$

where $\mu_j = E_j[\mathbf{x}]$ is the mean, and $\mathbf{S}_j = E_j[(\mathbf{x} - \mu_j)(\mathbf{x} - \mu_j)^T]$ is the covariance matrix of class j. Both can be approximated , yielding:

$$\mu_j = \frac{1}{N_j} \sum_{\mathbf{x} \in \omega_j} \mathbf{x} \qquad \text{and} \qquad \mathbf{S}_j = \frac{1}{N_j} \sum_{\mathbf{x} \in \omega_j} \left(\mathbf{x}\mathbf{x}^T - \mu_j \mu_j^T \right),$$

where N_j is the number of patterns in class j.

The next step has a serious implication. The common approach taken to simplify the Bayesian mathematics is to take the logarithm, thus $f_j(\mathbf{x}) = p(\mathbf{x}|\omega_j)\, P(\omega_j)$ becomes:

$$\ln(f_j(\mathbf{x})) = \ln(p(\mathbf{x}|\omega_j)) + \ln(P(\omega_j)). \tag{4.7}$$

This results in the simpler form for (4.6):

$$\ln(f_j(\mathbf{x})) = -\frac{d \ln(2\pi)}{2} - \frac{\ln(|\mathbf{S}_j|)}{2} - \frac{(\mathbf{x} - \mu_j)^T\, \mathbf{S}_j^{-1}\, (\mathbf{x} - \mu_j)}{2}, \tag{4.8}$$

which is a function of *hyperquadratics*, because no term is higher than the second degree, which also explains Nilsson's [151] interest in the quadratic decision function. The first term in (4.8) can be eliminated since it is assumed to be equal for all classes, then (4.8) can be analysed. If $\mathbf{S}_j = \mathbf{S}$ for all j, then $f_j(\mathbf{y})$ are linear decision functions, and if $\mathbf{S}_j = \mathbf{I}$ then (4.8) represents the minimum distance classifier. Of particular interest is the first case, since it results in the Lupas and Verdu [110] DS-CDMA decorrelating detector solution. For further explanation, refer also to Figure 4.4. First, the centre of gravity (μ_j) is computed for each class ω_j. Then, in order to maximise the distance to each class, the distance (dotted line) between them is halved. At that point, a hyperplane $\mathcal{H}$ perpendicular to the dotted line is placed, which is the decision boundary $\mathcal{H}$. $\mathcal{H}$ is determined by its weight vector in the d-dimensional space, which is known from the distance vector between the means.

It is also worth mentioning that sometimes, this linear boundary is referred to in literature as an optimum Bayesian decision boundary. On one hand it is true since it has been derived from the Bayesian statistics, but only for orthogonalised data. On the other hand, the optimum boundary (in terms of minimising the classification errors) is in many cases nonlinear.

4.2.5 Nonlinear pattern classification

Nilsson [151] investigated the quadratic boundary in more detail. For brevity, the discussion is kept short and shall show the relationship between a quadratic function and the MLSE, which exploits the Euclidean distance measure, a quadratic function too. Decision surfaces of quadratic machines are sections of second degree, defined as:

$$\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{x}^T\mathbf{b} + c = 0. \tag{4.9}$$

If $\mathbf{A}$ is positive definite, the surface of (4.9) is called a $hyperellipsoid$. The axes of the hyperellipsoid are the directions of the eigenvectors of $\mathbf{A}$. When $\mathbf{A}$ is an identity (or any select identity) matrix, then the surface is called a $hypersphere$ [156]. A possible implementation is given in Figure 4.3(b), where a quadratic preprocessor is embedded. A quadratic preprocessor generates a new pattern from $\mathbf{x}$, which is a nonlinear mapping into a higher space. Of course, this idea can be extended to higher polynomials, although the complexity rises. This technique is similar to the Volterra expansion, e.g. equation (4.9) is related to a 2nd-order Volterra equation [151].

A reasonable decision rule for optimum CDMA multiuser detection is to select the set of symbols corresponding to that signal among the possible ones that resembles most closely the received waveform. Verdu [121] stated the multiuser problem as:

$$\arg \max_{\{\mathbf{r} \in X^U\}} (2\mathbf{r}^T\mathbf{y} - \mathbf{r}^T\mathbf{S}\mathbf{r}), \tag{4.10}$$

where $\mathbf{r}$ is a symbol vector, $\mathbf{y}$ is the received signal, $\mathbf{S}$ the crosscorrelation matrix and X^U the set with all possible signals for U users. Again, a quadratic function is involved because the decision rule is based upon the mean square measurement.

4.3 Decision boundary in DS-CDMA

It is known from the ML technique that a received signal can be matched against all possible signal combinations and this also applies to CDMA. Assuming that the number of active users and their spreading sequences are known, all possible combinations can be constructed. For antipodal signalling and a multipath free channel, there are $M = 2^U$ possible signal combinations among the codes, see equation (2.6). These signals are binned into two sets, accord-

ing to the sign of the desired user bit. It is assumed that the desired user is user one, hence $D_1^+ = \{\mathbf{p}_m | m = 1, 2, \ldots, 2^U/2\}$ and $D_1^- = \{\mathbf{p}_m | m = 2^U/2 + 1, 2^U/2 + 2, \ldots, 2^U\}$. So, for a two user system, the 4 points $\mathbf{p}_m$ are derived from equation (2.8):

$$\begin{aligned}
\mathbf{p}_1 &= +D_1\mathbf{c}_1^T + D_2\mathbf{c}_2^T = +\mathbf{c}_1^T + \mathbf{c}_2^T \\
\mathbf{p}_2 &= +D_1\mathbf{c}_1^T - D_2\mathbf{c}_2^T = +\mathbf{c}_1^T - \mathbf{c}_2^T \\
\mathbf{p}_3 &= -D_1\mathbf{c}_1^T + D_2\mathbf{c}_2^T = -\mathbf{c}_1^T + \mathbf{c}_2^T \\
\mathbf{p}_4 &= -D_1\mathbf{c}_1^T - D_2\mathbf{c}_2^T = -\mathbf{c}_1^T - \mathbf{c}_2^T,
\end{aligned}$$

where $\mathbf{c}_u$ is the spreading code of user u, and $\mathbf{p}_m$ id the mth row in matrix $\mathbf{P}$. Without loss of generality the preprocessing based scheme (PPB) can be used, which enables a graphic representation of the points for $N > 3$. The PPB construction of the two sets for a three user scenario ($U = 3$) with randomly generated spreading codes of size ($N = 7$) with chips $\mathbf{c}_1 = [-1-1+1-1-1-1+1]^T$, $\mathbf{c}_2 = [-1+1+1-1-1+1+1]^T$ and $\mathbf{c}_3 = [-1+1+1+1+1+1-1]^T$ based upon equation (2.14) returns:

$$\begin{aligned}
D_1^+ &= \{[+7, +11, +5], [+1, -3, +3], [+13, +9, -9], [+7, -5, -11]\} \\
D_1^- &= \{[-7, -11, -5], [-1, +3, -3], [-13, -9, +9], [-7, +5, +11]\},
\end{aligned} \tag{4.11}$$

where each set consists of vectors with length $U = 3$. A ML receiver's task would be to make use of all points for distance measure, which means, that the distance is measured between a received signal and all signals within a set. Another way to classify a received signal could be to assign the received signal to one of the two sets by working out on which side of a decision boundary it lies. This implies that the decision boundary is known a prior. The optimum decision boundary of the above example is derived from the shape of a Bayesian structure as Figure 4.5 illustrates for a two user scenario. Clearly, the two points belonging to the desired user bit sign $+D_1$ can be separated from the two points belonging to $-D_1$, giving the two classes.

In order to compare linear and nonlinear decision boundaries, another presentation is given. Figure 4.6(b) presents the boundary derived from Figure 4.5 and in Figure 4.6(a) linear boundaries are depicted. These boundaries are found by putting a grid over an area, whose range is found empirically. Then, each point in the grid is fed into a receiver structure. The points

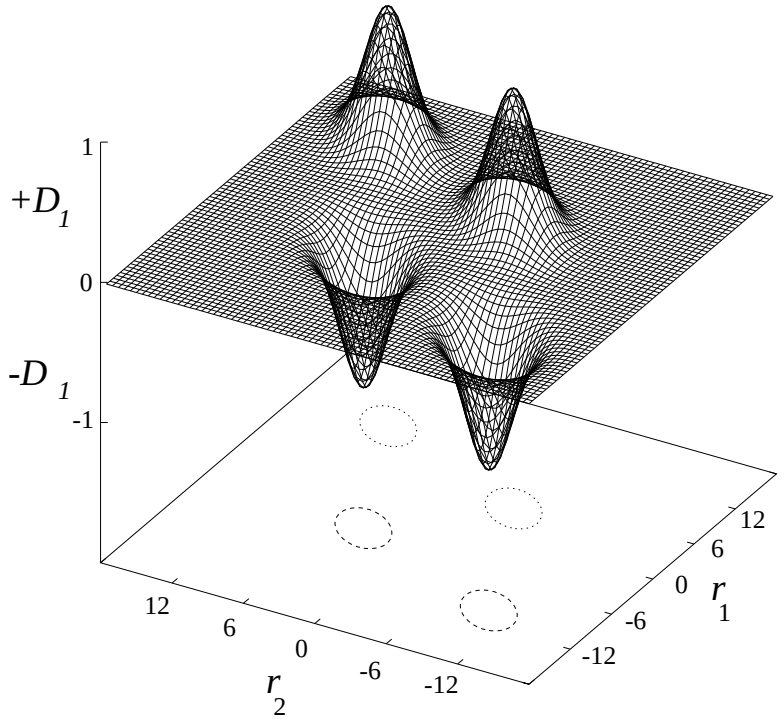
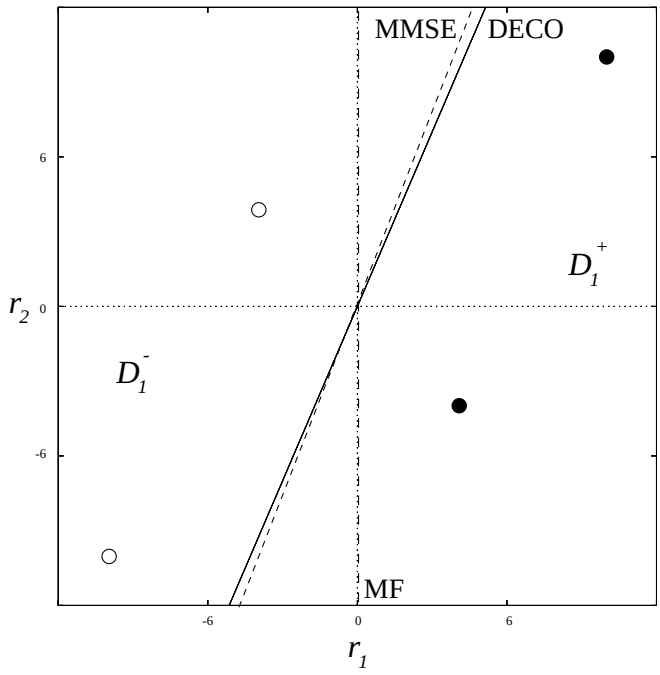


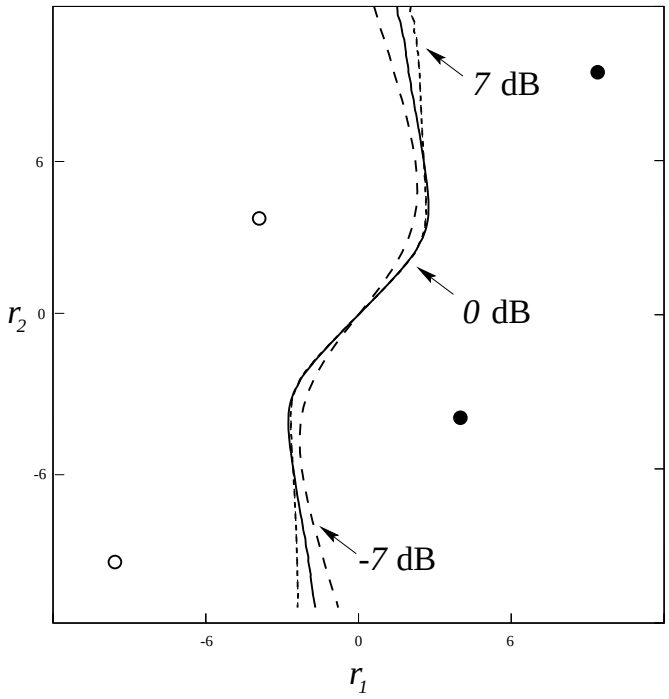
Figure 4.5: The decision surface (shape) of the Bayesian receiver structure for a two user CDMA scenario in AWGN and $E_b/N_0 = 7$dB. The axis r_1 and r_2 are the outputs of the preprocessor. Each hub represents the noise distribution around one of the four possible noise free signal states, see also Figure 3.5.

(receiver input) corresponding to the change of sign (receiver output) are stored in a file and used to draw the decision boundary.

Figure 4.6(a) shows linear decision boundaries for the two user scenario in AWGN with $N = 7$ and $E_b/N_0 = 7$dB, obtained from the system depicted in Figure 3.5. The MF's boundary is the r_2 axis. The DECO's boundary is optimum in the sense that the orthogonal distance from the hyperplane to all points is minimised. This can also be achieved by LP, since it minimises the L_1 distance criterion. The MMSE's boundary is optimised in terms of the MSE for a noise power of $E_b/N_0 = 7$dB, thus DECO and MMSE do not coincide. The optimum Bayesian decision boundary, in terms of classification error minimisation, is depicted in Figure 4.6(b) for three different noise powers. A comparison between Figure 4.6(a) and 4.6(b) gives an indication of the potential for receiver performance improvement when nonlinear decision boundaries are constructed. This becomes even more apparent for a three user scenario.



(a) *Linear decision boundaries of the decorrelating and the MMSE receivers at* $E_b/N_0 = 7$*dB.*

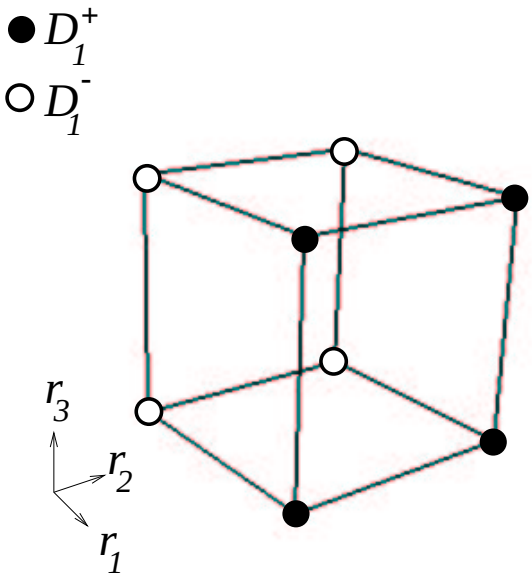


(b) *Optimum Bayesian decision boundaries for three different values of* E_b/N_0.

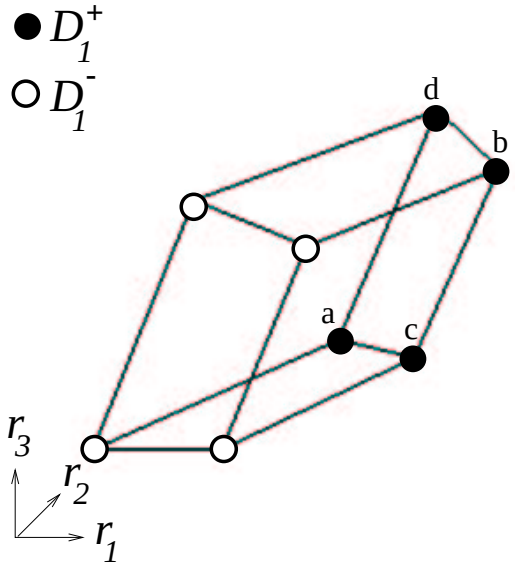
Figure 4.6: Decision boundaries derived from three different receiver structures for a two user scenario in a non-dispersive AWGN channel.

Figure 4.7(b) shows all 8 points given in (4.11). Figure 4.7(a) shows for comparison the points for orthogonal codes, where linear filters are optimum. Figures 4.7 reveals the fact that all points are vertices of a hypercube, if the spreading codes are orthogonal. This is described by matrix $\mathbf{B}$ in equation (2.8) and (2.13). The covariance matrix $\mathbf{S}$ describes the shape of the hypercube. If $\mathbf{S}$ is a diagonal matrix, then the hypercube's shape looks like Figure 4.7(a). If $\mathbf{S}$ contains crosscorrelation factors between the spreading codes, then the hypercube's shape looks like Figure 4.7(b).

The optimum decision boundary is given in Figure 4.8(a) and the MMSE decision boundary in Figure 4.8(b). Note that the coordinate system has been changed for better visualisation. The points $\{a, b, c, d\}$ are not directly shown but their location can easily be assumed by comparing Figure 4.8(a) and 4.8(b) with Figure 4.7(b).

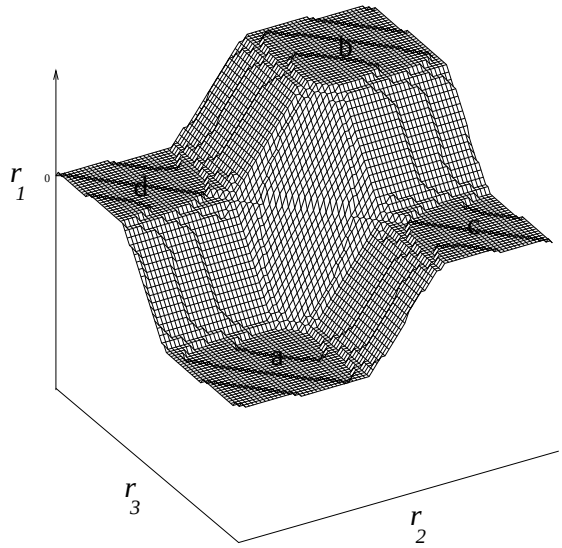


(a) *Orthogonal codes.*

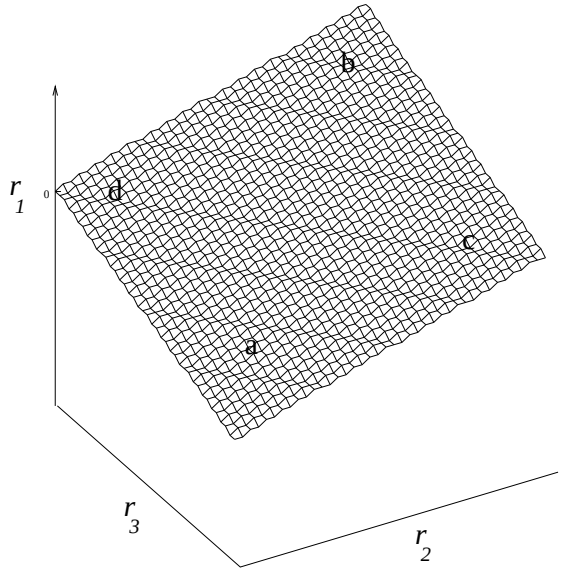


(b) *Nonorthogonal codes, the points are given in equation (4.11).*

Figure 4.7: The 8 possible points of a three user CDMA scenario in a noise free non-dispersive AWGN channel, which correspond to the 8 expected possible received signal points.



(a) *Optimum Bayesian decision boundary.*



(b) *MMSE decision boundary (plane).*

Figure 4.8: Decision boundaries for the scenario given in Figure 4.7(b). Due to the chosen stepsize to compute both graphs, which is a compromise between accuracy of the shape and the contrast, the surfaces appear not very smooth or plain.

Figure 4.9 shows all possible received signals from a three user multipath scenario which form eight clusters. There are in total $M = 2^{3\times U} = 2^{3\times 3}$ points which form $2^U = 2^3 = 8$ clusters, where each cluster has $2^{2\times U} = 2^{2\times 3} = 64$ points. The points within a cluster are not explicitly shown but their convex hull. The centre of each cluster does not correspond to the points shown in Figure 4.7(b) but can be determined. This is discussed in chapter 6 and 7. Important is to note the fact, that ISI causes a signal (or point) to spread and clusters can arise. For a better presentation, the convex hull [146, 147] of each cluster is drawn[3].

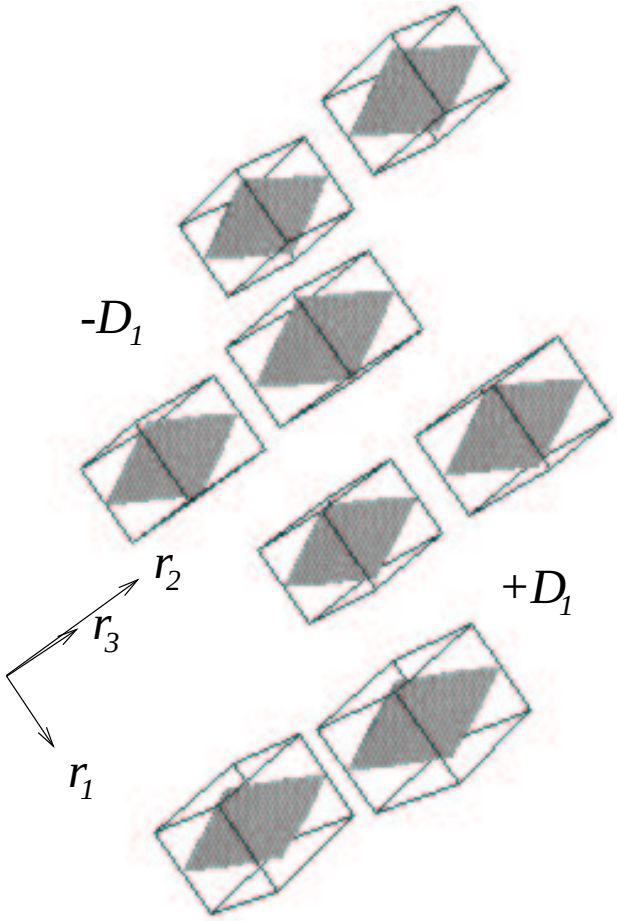


Figure 4.9: Eight convex hulls which contain in total all possible received signals.

[3]With Geomview [157], see: http://www.geom.umn.edu/software/download/geomview.html

4.4 Approximation theory

Constructing sub-optimum decision boundaries, for the sake of network complexity reduction, can also be considered as approximating the optimum Bayesian decision boundary with something simpler. This important technique known as *approximation theory* is reviewed in this section, based on Poggio and Girosi [98].

A multivariate function $f(\mathbf{x})$ shall be approximated by a function $\mathcal{F}(\mathbf{w}, \mathbf{x})$, having a fixed number of parameters $\mathbf{w}$ belonging to some set P. $\mathcal{F}(\mathbf{w}, \mathbf{x})$ corresponds to multilayer (neural) networks and performs a mapping from $\mathbb{R}^d \rightarrow \mathbb{R}$. The classical linear mapping is:

$$\mathcal{F}(\mathbf{w}, \mathbf{x}) = \mathbf{w}^T \mathbf{x}, \tag{4.12}$$

which corresponds to a (neural) network without a hidden layer. The classical approximation scheme is:

$$\mathcal{F}(\mathbf{w}, \mathbf{x}) = \sum_{m=1}^{M} w_m \, \phi_m(\mathbf{x}) \tag{4.13}$$

and corresponds to a single hidden layer (neural) network. $\mathcal{F}$ is a polynomial when the $\phi_m(\cdot)$ are products and powers of the input. The nested sigmoid scheme is:

$$\mathcal{F}(\mathbf{w}, \mathbf{x}) = \delta \left(\sum_i \mathbf{w}_i \delta \left(\sum_j \mathbf{w}_j \mathbf{x}_j \right) \right), \tag{4.14}$$

where δ is a linear threshold function or a sigmoid. This network corresponds to a multilayer network, such as a MLP, and has its origin in representing Boolean functions. Networks of this type with one hidden layer can approximate any continuous multivariate function. Poggio calls them *regularization networks* and shows that they are strictly related to the interpolation method of RBFs.

Poggio reviews the RBF method as a possible solution to the real multivariate interpolation problem. Given M different points $\{\mathbf{x}_i \in \mathbb{R}^d : i = 1, 2, \ldots, M\}$ and M real numbers $\{\mathbf{y}_i \in \mathbb{R}^1 : i = 1, 2, \ldots, M\}$ find a function $\mathcal{F}$ from $\mathbb{R}^d$ to $\mathbb{R}$ satisfying the interpolation:

$$\mathcal{F}(\mathbf{x}_i) = \mathbf{y}_i. \tag{4.15}$$

Then the RBF approach is choose a function $\mathcal{F}$, which has the form:

$$\mathcal{F}(\mathbf{x}) = \sum_{i=1}^{M} w_i \, \phi(\|\mathbf{x} - \mathbf{x}_i\|), \tag{4.16}$$

where $\phi(\cdot)$ is a continuous function from $\mathbb{R}^+$ to $\mathbb{R}$, called the *radial basis function*. $\|\cdot\|$ is the Euclidean norm on $\mathbb{R}^d$. Micchelli [158] showed that the coefficients $\mathbf{w}_i$ can be derived from the linear system:

$$\mathbf{y} = \mathbf{w}\Phi, \tag{4.17}$$

where $\mathcal{F}(\mathbf{x}_j) = \mathbf{y}_j$, and $\Phi_{ij} = \phi(\|\mathbf{x}_i - \mathbf{x}_j\|)$ $(i, j = 1, 2, \ldots, M)$. A list of possible interpolation functions are also given, such as:

$$\begin{aligned}
\phi(\mathbf{x}) &= \mathbf{x} && \text{(linear)} \\
\phi(\mathbf{x}) &= \exp(-\frac{\mathbf{x}^2}{\mathbf{c}^2}) && \text{(Gaussian)} \\
\phi(\mathbf{x}) &= \frac{1}{(\mathbf{c}^2 + \mathbf{x}^2)^\alpha} && \alpha > 0 \\
\phi(\mathbf{x}) &= (\mathbf{c}^2 + \mathbf{x}^2)^\beta && 0 < \beta < 1.
\end{aligned}$$

The linear and the Gaussian functions are the most popular. The linear case corresponds, in one dimension, to piecewise linear interpolation, the simplest case of what is called *spline* interpolation [1]. However, all of these functions depend on a parameter that will generally depend on the distribution of the data points. Since the quality of interpolation and approximation of a decision surface depends on parameters, an accurate parameter estimation appears to be vital for a good network performance.

4.5 Summary

This chapter reviewed pattern recognition techniques and showed how it can be applied to DS-CDMA. Linear techniques developed in the sixties are still desirable due to their simplicity, and are even more powerful in combination with current DSP technology. Since most classification applications have a nonlinear decision boundary, and fully nonlinear techniques are complex in nature, it appears to be reasonable to approximate them with piecewise linear decision boundaries.

It also has been shown that linear decision boundaries derived from Bayesian statistics do not necessarily lead to the optimum boundary. A reason for this is that it is assumed to be dealing with whitened data, which is orthogonal, in which case the linear MSE boundary becomes optimum. But this assumption calls for a whitening filter for data preprocessing, which adds additional complexity to the network. However, it is known that other distance metrics than the Euclidean metric take the correlated noise into account, for example the Mahalanobis distance measure [155]. This technique adds little additional computation to the network and it seems straightforward to apply. Also presented were several graphs which show the decision boundary for CDMA systems. Linear and nonlinear CDMA decision surfaces were shown and it has been demonstrated that the optimum boundary is nonlinear and consists at least of 2^U hyperplanes (in the ML sense). Moreover, reasons for the performance loss of linear boundaries became obvious. Finally, a brief overview on approximation theory was given which showed the relations between linear, polynomial and RBF networks. The theory states that the input data is mapped linearly or nonlinearly into another space (generally of higher dimension), then a linear mapping maps them back to the desired output space. It appeared that the Wiener-Hopf equation and the Volterra series can be derived from it as well. A useful feature seems to be the fact that nonlinear functions can easily be embedded, while a linear mapping remains which enables the use of linear techniques to find the coefficients for the final linear mapping.

This review showed that techniques known in pattern recognition can classify CDMA data. Moreover, approximation theory support this fact since it enables the engineer to apply nonlinear techniques for receiver designs which can be made adaptive. This can lead to a new series of nonlinear receivers (which may be adaptive) with superior performance compared with the established linear receiver designs.

Chapter 5
The Volterra approach

This chapter presents a CLB Volterra receiver structure based on the Wiener-Hopf equation.

Firstly, a general introduction is given followed by a detailed analysis of the Volterra expansion. Then the 1st-order Volterra receiver is shown to be the MMSE receiver. The investigation of the 2nd-order Volterra system shows that it is not applicable. Consequently, higher- order Volterra systems are investigated. A statistical analysis is given for the autocorrelation matrix of a 3rd-order Volterra system, and a technique is presented to circumvent the problems encountered. Simulation results compare the performance of the proposed receiver structure against established ones. Then decision boundaries obtained from preprocessed based Volterra structures are presented. Finally a discussion summarises this chapter. Results and investigations, which lead to the conclusions presented in this chapter, but which are too lengthy to present here, can be found in the appendices A and B.

5.1 Introduction

The general *Volterra series* (VS) is given as an infinite series expansion [159], which is not useful for practical applications. Thus, one must work with a truncated VS, such as the third-order VS given in (5.1), which consists of products up to 3rd-order.

$$\begin{aligned} v(kN+n) \;=\; & \sum_{a=0}^{N-1} h_1(a)\, y(kN+n-a) + \\ & \sum_{a=0}^{N-1}\sum_{b=0}^{N-1} h_2(a,b)\, y(kN+n-a)\, y(kN+n-b) + \\ & \sum_{a=0}^{N-1}\sum_{b=0}^{N-1}\sum_{c=0}^{N-1} h_3(a,b,c)\, y(kN+n-a)\, y(kN+n-b)\, y(kN+n-c), \end{aligned} \tag{5.1}$$

where $y(kN+n)$ denotes the filter input and $v(kN+n)$ the output for the kth symbol of length N with $n = 1, 2, \ldots, N$ chips. The term h_o in (5.1) denotes the oth-order Volterra kernel (coefficients, or weights w) of the system. Without loss of generality, it can be assumed that

the kernels are symmetric (e.g. $h_2(a,b) = h_2(b,a)$). The symmetric terms can be omitted since they do not contribute any additional information, which results in half the number of coefficients for h_o [81]. Thus the Volterra kernels h_o are fixed for any of the possible permutations. Hence, (5.1) can be rewritten for a symbol synchronised receiver:

$$\begin{aligned}\widehat{D}_d(y(k)) \;=\; \text{sgn}\,(\; & \sum_{a=0}^{N-1} h_1(a)\,y(kN-a) + \sum_{a=0}^{N-1}\sum_{b=a}^{N-1} h_2(a,b)\,y(kN-a)\,y(kN-b) + \\ & \sum_{a=0}^{N-1}\sum_{b=a}^{N-1}\sum_{c=b}^{N-1} h_3(a,b,c)\,y(kN-a)\,y(kN-b)\,y(kN-c)\;)\,, \end{aligned} \tag{5.2}$$

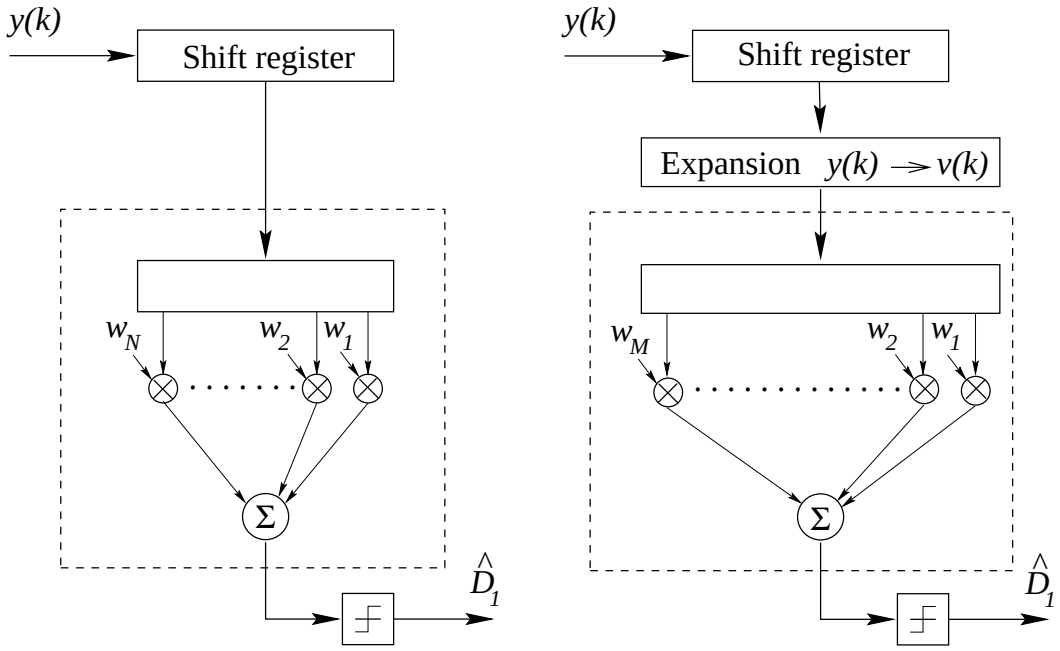
where $\widehat{D}_d(y(k))$ stands for the kth estimated transmitted bit of the desired user d. A possible filter structure is depicted in Figure 5.1(b). It becomes apparent from equation (5.2), that the term in sgn$(\cdot)$ is a sum of products between a received sequence $\mathbf{y}(k)$ and Volterra coefficients h_o.

5.2 The Volterra expansion

The Volterra expansion and the expansion sequence are analysed for a one user CLB CDMA system in AWGN. Due to the binomial growth in number of coefficients, the analysis is presented with a short spreading code of length $N = 3$. In order to apply the Volterra filter to the received signal $\mathbf{y}(k)$, it must first be expanded to a larger sequence, denoted by $\mathbf{v}(k)$. Figure 5.1 shows the difference between a simple FIR filter and a Volterra based FIR filter.

The expansion process is a mapping from the input space N to the Volterra space M, $\mathbf{y}(k) \Rightarrow \mathbf{v}(k) \equiv \mathbb{R}^N \Rightarrow \mathbb{R}^M$, where N is the number of chips and M the number of Volterra coefficients[1]. The M elements of $\mathbf{v}(k) = [v_1\ v_2 \ldots v_M]^T$ are computed corresponding to the desired order o of the VS (5.2). This process is depicted for a 1st and 3rd-order system in Figure 5.2. The second-order sequence can be omitted for equiprobable and antipodal signals, for reasons which will be explained in section 5.4. For the same reasons, the even terms are omitted in equation (5.3) and Figure 5.2.

[1]Note, this M is not related to the number of patters, which is also M.



(a) *Common N-tap FIR.* **(b)** *Volterra based approach with M-taps.*

Figure 5.1: Conventional FIR filtering and the Volterra approach.

The expansion sequence **v** of Figure 5.2 is given by (2.6) and (5.2) and is defined as:

$$
\begin{aligned}
\mathbf{v} &= [v_1\ v_2 \ldots v_M]^T \qquad (5.3)\\
v_1 &= \sum_{u=1}^{U} D_u c_{u,1} + g(1)\\
&\vdots\\
v_N &= \sum_{u=1}^{U} D_u c_{u,N} + g(N)\\
v_{N+1} &= \left\{\sum_{u=1}^{U} D_u c_{u,1} + g(1)\right\}^3\\
v_{N+2} &= \left\{\sum_{u=1}^{U} D_u c_{u,1} + g(1)\right\}^2 \times \left\{\sum_{u=1}^{U} D_u c_{u,2} + g(2)\right\}\\
&\vdots\\
v_M &= \left\{\sum_{u=1}^{U} D_u c_{u,N} + g(N)\right\}^3,
\end{aligned}
$$

where D_u is the transmitted bit of user u and $c_{u,n}$ the nth chip of the uth user's spreading

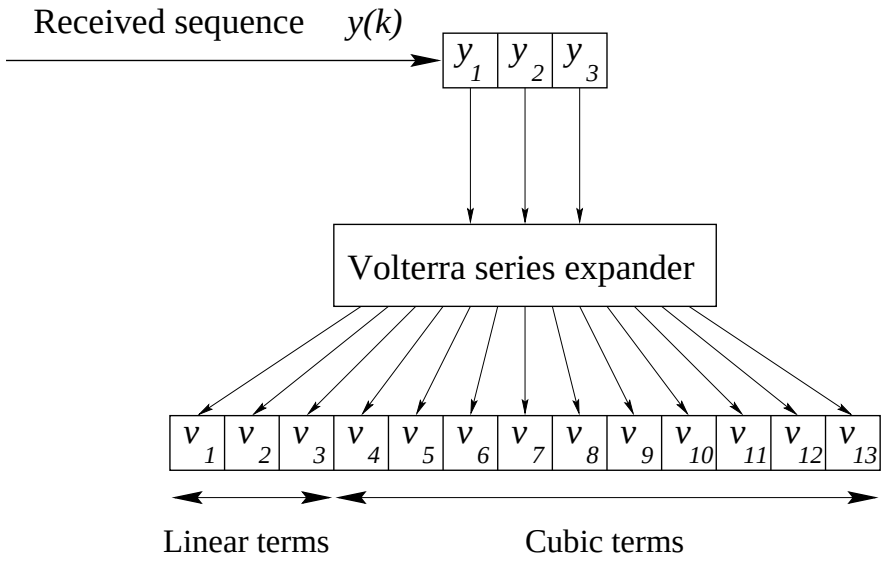


Figure 5.2: The Volterra expansion for a combined 1st and 3rd-order system.

sequence.

The vector length of $\mathbf{v}$ is M, where M is the number of filter weights or Volterra coefficients, and is determined by the binomial expression:

$$M(N,O) = \sum_{order=1\ :\ order\in\{1,3,5,\ldots\}}^{O} \left(\begin{array}{c} N+order-1 \\ order \end{array} \right), \tag{5.4}$$

where N is the length of the input sequence (memory) and O the highest Volterra order. $M(N,3)$ is the number of coefficients for a combined 1st and 3rd-order expansion. Thus, if $\mathbf{v}$ of equation (5.3) is of length $M(N,3)$ then it has first $M(N,1) = N$ linear terms and then $(M(N,3) - M(N,1))$ cubic terms, see Figure 5.2. Since N can be assumed to be constant for a system, a short form M_{13} of $M(N,3)$ is used instead. Tsimbinos and Lever [91] investigated the computational complexity $C(N,O)$ in terms of multiplications needed, given as:

$$C(N,O) = \sum_{o=1}^{O} \frac{(N+o-1)!}{(o-1)!\,(N-1)!}, \tag{5.5}$$

where $C(N,O)$ does not represent the number of flops, since it does not take the additions into account, see also Figure 3.2. The difference between (5.4) and (5.5) is a factor of o (order). Table 5.1 presents some results obtained by (5.4) and (5.5) of VS systems. It does not show the binomial sum but, the specific complexity for a specific value of N and an order. Table 5.1 shows the growing complexity as order and memory increases.

	M(N,O)	M(N,O)	M(N,O)	C(N,O)	C(N,O)	C(N,O)
order	N=3	N=7	N=9	N=3	N=7	N=9
1	3	7	9	3	7	9
2	6	28	45	15	63	99
3	10	84	165	45	315	594
4	15	210	495	105	1155	2574
5	21	462	1287	210	3465	9009

Table 5.1: Complexity in terms of operations required per estimated bit for different Volterra filters, where $M(N,O)$ is the expansion length and $C(N,O)$ the number of multiplications for a particular VS filter with order O and memory N, the tap length.

From Table 5.1 it can be seen that for $N = 7$ a combined 1st and 3rd-order VS FIR filter has $M_{13} = 91$ taps, and a combined 1st,3rd and 5th-order filter has $M_{135} = 553$ taps. To reduce the filter complexity, a VS filter can also be designed as a PPB receiver, which processes $\mathbf{r}(k)$ instead of $\mathbf{y}(k)$ [118]. However, in every case, a VS receiver system has to expand the received sequence in order to filter it. Another point worth mentioning is that the VS expansion changes the statistics of the sequence the Wiener FIR filter is applied to. Higher-order moments are induced and may be exploited by *higher-order statistics* (HOS) [160], which was not the subject of this work.

5.2.1 Statistical properties of the expansion sequence

A received vector $\mathbf{y}(k) = [y_1\ y_2\ y_3]^T$ is according to (2.6):

$$\begin{aligned} y_1 &= \sum_{u=1}^{U} D_u c_{u,1} + g(1) \\ y_2 &= \sum_{u=1}^{U} D_u c_{u,2} + g(2) \\ y_3 &= \sum_{u=1}^{U} D_u c_{u,3} + g(3), \end{aligned}$$

where $g(n)$ is the uncorrelated Gaussian noise with zero mean and given variance added onto the nth chip, D_u the data bit, which has equal probability of being $+1$ or -1 and $c_{u,n}$ is the nth chip of the uth user's spreading sequence. The 3rd-order Volterra expansion (5.3) induces products among the vector components $\{y_n\}$, which can be grouped according to their statist-

ical properties. For a 3rd-order system there are four groups which represent all permutations:

$$\begin{aligned} p_1 &= \{y_i\} \\ p_2 &= \{y_i^3\} \\ p_3 &= \{y_i^2 y_j\} \qquad i \neq j \\ p_4 &= \{y_i y_j y_k\} \qquad i \neq j \neq k, \end{aligned}$$

where $i, j, k \in \{1, 2, \ldots, N\}$. Obviously, each group p_t represents a term, which is a product of powers of $\{y_n\}$, where the terms have even or odd powers. This leads to another grouping representing the powers of y_i, $\alpha = \{2, 4, 6\}$ and $\beta = \{1, 3, 5\}$. Further, the noise is considered to be normally distributed with $\mathcal{N}(0, 1)$ for simplicity, since it is an white Gaussian noise channel. First the distribution for the α group is derived, then for the β group.

Define $X \sim \mathcal{N}(0, 1)$ and $Y = X^\alpha$. Then the cdf for the even function is [161, 162]

$$\begin{aligned} G(y) &= P\{Y \leq y\} = P\{X^\alpha \leq y\} \\ &= P\{-y^{1/\alpha} < X < y^{1/\alpha}\} \\ &= F(y^{1/\alpha}) - F(-y^{1/\alpha}) \end{aligned} \tag{5.6}$$

and the pdf is

$$\begin{aligned} g(y) &= \frac{d}{dy} G(y) \\ &= \frac{1}{\alpha}\, y^{\frac{1}{\alpha}-1} f(y^{1/\alpha}) + \frac{1}{\alpha}\, y^{\frac{1}{\alpha}-1} f(-y^{1/\alpha}). \end{aligned}$$

Substituting the Gaussian function

$$f(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) \tag{5.7}$$

into the pdf gives

$$g(y) = \frac{2}{\alpha}\, y^{\frac{1-\alpha}{\alpha}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{y^{2/\alpha}}{2}\right). \tag{5.8}$$

For the β group, $X \sim \mathcal{N}(0,1)$ and $Y = X^\beta$, the cdf for the odd function is

$$\begin{aligned} G(y) &= P\{Y \leq y\} \\ &= P\{X^\beta \leq y\} \\ &= P\{X \leq y^{1/\beta}\} \end{aligned} \tag{5.9}$$

and the pdf is

$$g(y) = \frac{1}{\beta}\, y^{\frac{1-\beta}{\beta}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{y^{2/\beta}}{2}\right). \tag{5.10}$$

The resulting distributions and their expected value are given in appendix A and can also be found in [163].

5.3 First-order systems

A 1st-order Volterra system is given in equation (5.11)

$$v(k) = \sum_{a=0}^{N-1} h_1(a) y(kN - a) \tag{5.11}$$

for a received sequence $\mathbf{y}(k)$. Equation (5.11) is a linear convolution, from which the Wiener solution (3.9) has been derived, based upon the Wiener-Hopf equations [48].

Equation (5.11) can also be seen as an approximating function, such as the classical approximation (4.13): $\mathcal{F}(\mathbf{w}, \mathbf{v}) = \sum_{i=1}^{M} \mathbf{w}_i \phi(\mathbf{v})$, where $\phi(\cdot)$ is a linear or nonlinear function. Again, if $\phi(\cdot)$ is a linear function, then it will result in the MMSE solution [48, 158], which has been introduced in subsection 3.1.2 and section 3.3.

5.4 Second-order systems

If the function $\phi(\cdot)$ is taken to be a polynomial function, such as the 2nd-order Volterra series, then the MSE approach still applies [158]. Hence, the Wiener-Hopf equation can be applied

and becomes:

$$\mathbf{w} = \mathbf{R}_{vv}^{-1}\mathbf{r}_{xv}, \tag{5.12}$$

which now deals with the Volterra sequence $\mathbf{v}$ instead of the received sequence $\mathbf{y}$. Thus the filtering can be done with a simple FIR filter.

In order to solve (5.12), the autocorrelation matrix $\mathbf{R}_{vv}$ and the crosscorrelation vector $\mathbf{r}_{xv}$ must be derived. $\mathbf{r}_{xv}$ is defined as the correlation between the desired user bit D_d and the input sequence $\mathbf{v}$. Without loss of generality, assume the desired user is user one. Then $\mathbf{r}_{xv}$ becomes for a 2nd-order Volterra system:

$$\begin{aligned}
\mathbf{r}_{xv} &= E[D_u \times \mathbf{v}] = E[D_1 \times \mathbf{v}] \qquad (5.13)\\
&= E[D_1[D_1^1 c_{1,1} \ldots D_1^1 c_{1,N}\ D_1^2 c_{1,1}^2 \ldots D_1^2 c_{1,N}^2]^T]\\
&= E[[D_1^2 c_{1,1} \ldots D_1^2 c_{1,N}\ D_1^3 c_{1,1}^2 \ldots D_1^3 c_{1,N}^2]^T]\\
\mathbf{r}_{xv} &= [c_{1,1}\ c_{1,2} \ldots c_{1,N}]^T. \qquad (5.14)
\end{aligned}$$

Since D_u is typically equally likely to be $+1$ or -1, $E[D_u^{even}]$ with even powers becomes one and $E[D_u^{odd}]$ with odd powers becomes zero. Thus, the crosscorrelation vector for a second-order Volterra system results in the first-order crosscorrelation vector, which is the spreading sequence of the desired user.

Hence, for the DS-CDMA system considered, a second-order Volterra filter collapses into a first-order Volterra filter [164].

5.5 Higher-order systems

The result found in section 5.4 implies that for equiprobable antipodal signals only odd-order Volterra DS-CDMA filters exist. This leads to an increased filter complexity. In order to keep the structure reasonable small, 3rd-order Volterra structures shall be investigated.

Again, the starting point is equation (5.12), where vector $\mathbf{v}$ has the structure given in (5.3). In order to derive the VS filter weights, the expected value of the autocorrelation matrix and the crosscorrelation vector has to be derived, $\mathbf{R}_{vv} = E[\mathbf{v}\mathbf{v}^T]$ and $\mathbf{r}_{xv} = E[D_1 \times \mathbf{v}]$ respectively. Due to the Volterra expansion process, the expected value is not easily derived. The received

signal consists of a sum of spreading codes and Gaussian noise. The expansion process induces cubic terms, which become of even higher-orders in the autocorrelation matrix.

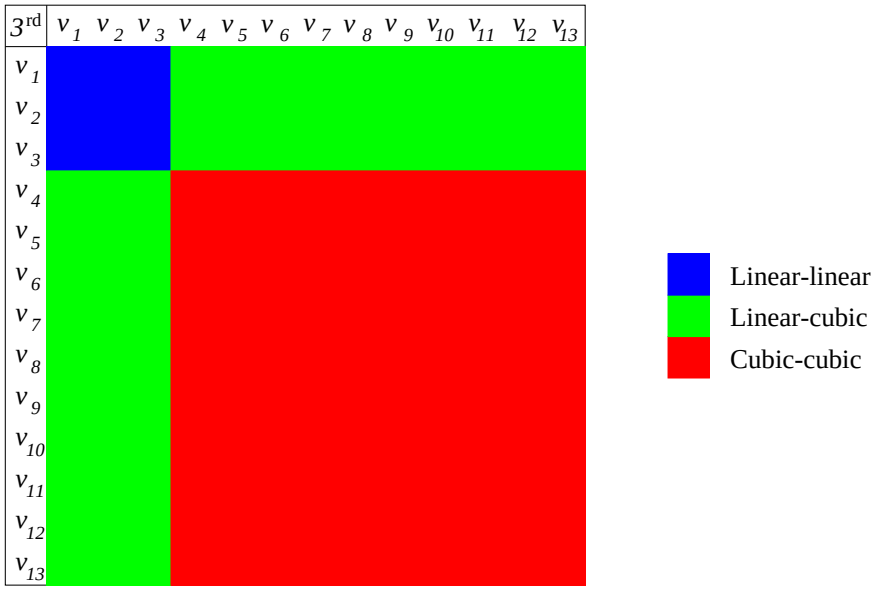


Figure 5.3: The autocorrelation matrix which consists of basically three groups of products, see also subsection 5.5.1 for more details.

Figure 5.3 shows $\mathbf{R}_{vv}$ partitioned into the three major product terms. The linear-linear terms are known from the MMSE solution [108]. The products among the linear-cubic and the cubic-cubic terms are not described in the literature and must be derived. An analysis for the autocorrelation matrix is given next, and from this the crosscorrelation vector also can be derived.

5.5.1 Statistical properties of the autocorrelation matrix

All possible occurring terms within $\mathbf{R}_{vv}$ can be grouped according to their expected value. The terms in the autocorrelation matrix follow the rules for the powers of $E[D_u]$, hence $E[(D_u)^{odd}] = 0$ and $E[(D_u)^{even}] = 1$. The noise is assumed to be uncorrelated with itself and the data bit, hence $E[g(i)g(j)] = 0$ for $i \neq j$ and $E[D_u g(j)] = 0$. This leads to 2 linear-linear, 4 linear-cubic and 7 cubic-cubic groups G_m, for $m = 1, 2, \ldots, M$, which have expected value $G_m = E[v_i v_j]$ with $i, j \in \{1, 2, \ldots, M\}$.

In (5.15), $E[v_i v_j]$ $i \neq j$ has been replaced by the chips, hence $E[y_i y_j y_k]$ with $i, j, k \in \{1, 2, 3\}$ and $i \neq j \neq k$. For convenience, $c_{u,i}$ is replaced in (5.15) by c_i since $u = 1$. The results

obtained for one user from that derivation are represented by the 13 groups:

$$
\begin{aligned}
G_1 &= E\{y_i y_j\} = c_i c_j \\
G_2 &= E\{y_i^2\} = c_i^2 + g(i)^2 \\
G_3 &= E\{y_i^4\} = c_i^4 + 6c_i^2 g(i)^2 + g(i)^4 \\
G_4 &= E\{y_i^3 y_j\} = c_i^3 c_j + 3c_i c_j g(i)^2 \\
G_5 &= E\{y_i^2 y_j^2\} = c_i^2 c_j^2 + c_i^2 g(j)^2 + c_j^2 g(i)^2 + g(i)^2 g(j)^2 \\
G_6 &= E\{y_i^2 y_j y_k\} = c_i^2 c_j c_k + c_j c_k g(i)^2 \\
G_7 &= E\{y_i^6\} = c_i^6 + 15c_i^4 g(i)^2 + 15c_i^2 g(i)^4 + g(i)^6 \\
G_8 &= E\{y_i^5 y_j\} = c_i^5 c_j + 10c_i^3 c_j g(i)^2 + 5c_i c_j g(i)^4 \\
G_9 &= E\{y_i^4 y_j^2\} = c_i^4 c_j^2 + 6c_i^2 c_j^2 g(i)^2 + c_j^2 g(i)^4 + c_i^4 g(j)^2 + 6c_i^2 g(i)^2 g(j)^2 + g(i)^4 g(j)^2 \\
G_{10} &= E\{y_i^4 y_j y_k\} = c_i^4 c_j c_k + 6c_i^2 c_j c_k g(i)^2 + c_j c_k g(i)^4 \\
G_{11} &= E\{y_i^3 y_j^3\} = c_i^3 c_j^3 + 3c_i c_j^3 g(i)^2 + 3c_i^3 c_j g(j)^2 + 9c_i c_j g(i)^2 g(j)^2 \\
G_{12} &= E\{y_i^3 y_j^2 y_k\} = c_i^3 c_j^2 c_k + c_i^3 c_k g(j)^2 + 3c_i c_j^2 c_k g(i)^2 + 3c_i c_k g(i)^2 g(j)^2 \\
G_{13} &= E\{y_i^2 y_j^2 y_k^2\} = c_i^2 c_j^2 c_k^2 + c_i^2 c_k^2 g(j)^2 + c_j^2 c_k^2 g(i)^2 + c_k^2 g(i)^2 g(j)^2 + \\
&\quad c_i^2 c_j^2 g(k)^2 + c_i^2 g(j)^2 g(k)^2 + c_j^2 g(i)^2 g(k)^2 + g(i)^2 g(j)^2 g(k)^2.
\end{aligned}
\tag{5.15}
$$

Linear-linear:$\{G_1, G_2\}$ Linear-cubic:$\{G_3 \ldots G_6\}$ Cubic-cubic:$\{G_7 \ldots G_{13}\}$

Obviously, the terms given in (5.15) are sums of expected values for different products such as signal-signal, signal-noise and noise-noise terms, where the noise-noise products have a different distribution function than the original signal.

The term G_1 has no added noise components since the noise added to each chip is uncorrelated with each other. The diagonal terms G_2 have the Gaussian noise component squared, which results in the Chi-square distribution [163]. Hence the added noise component (power) is described with the noise variance σ^2 [71]. Equations (5.15) have noise products up to power six, and the derivation presented in subsection 5.2.1 can also be used here. Results are also given in Papoulis [163] and Kay [165]. The important point is that the expected value of a Gaussian random variable x with odd-order power becomes zero [165]. Therefore only even powers

exist:

$$E[x^2] = \int_{-\infty}^{\infty} x^2 f(x)dx = \sigma^2 \tag{5.16}$$

$$E[x^4] = \int_{-\infty}^{\infty} x^4 f(x)dx = 3\ (E[x^2])^2 = 3(\sigma^2)^2 \tag{5.17}$$

$$E[x^6] = \int_{-\infty}^{\infty} x^6 f(x)dx = 15\ (E[x^2])^3 = 15(\sigma^2)^3 \tag{5.18}$$

hence, $g(i)^2 = \sigma^2$, $g(i)^4 = 3\sigma^4$ and $g(i)^6 = 15\sigma^6$.

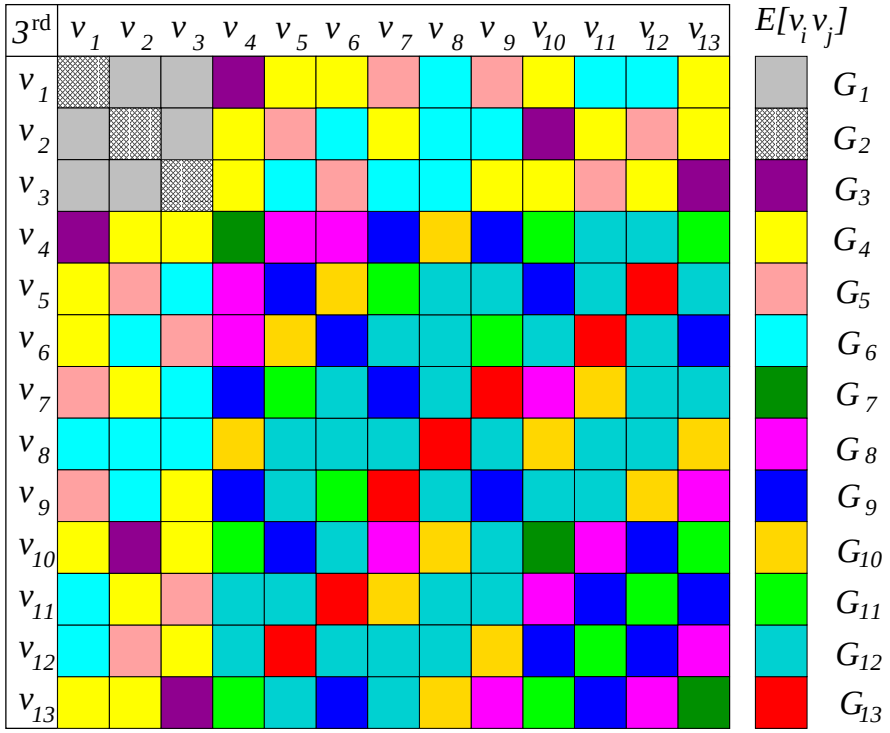


Figure 5.4: The autocorrelation matrix with all (13) possible received expected values.

Figure 5.4 shows a graphical representation of the results found in (5.15). Each group G_m is depicted by a unique colour. In order to describe the occurring distributions within the terms of the autocorrelation matrix and the crosscorrelation vector, it might be better making use of cumulants instead of working with the pdf, since the calculation with the moments obtained in appendix A are very intricate. However, all terms of the autocorrelation for a one user scenario with $N = 3$ have been determined. The derivation of the crosscorrelation vector follows the example of (5.15). Obviously, the calculations are tedious, especially for large U and N. Therefore, an algorithm which computes $\mathbf{R}_{vv}$ and $\mathbf{r}_{xv}$ is desirable. Attempts made to find such an algorithm failed due to the unique nature of the problem, see appendix B. The

author's literature search leads to the conclusion that such an algorithm, if it exists, has not been published. To circumvent the mathematical derivation for each term of $\mathbf{R}_{vv}$ and $\mathbf{r}_{xv}$, an estimation technique has been derived.

5.5.2 The estimation technique

The estimation technique exploits the property that the signal's autocorrelation matrix and crosscorrelation vector can be found by adding up all possible sub-autocorrelations and sub-crosscorrelations, derived from all possible signal combinations, and dividing the sum by the number of such combinations, which is M^2. This is a cyclic process, where during each cycle or loop one possible signal combination is processed. If $g(n)$ in equation (2.6) is zero, then it results in the signals (spreading codes) $E[\mathbf{R}_{vv}]$ and $E[\mathbf{r}_{xv}]$. However, due to the influence of the noise $g(n)$, the resulting matrix and vector are an estimate of the true matrix and vector respectively, denoted $\widehat{E}[\mathbf{R}_{vv}]$ and $\widehat{E}[\mathbf{r}_{xv}]$. In order to obtain a good estimate, the estimation must be done over K times, where the estimation improves as $K \gg M$. After $\mathbf{R}_{vv}$, $\mathbf{r}_{xv}$ and counters k and m are initialised, the estimation process is started. The iteration is performed over four steps, where M is the number of possible combinations among the spreading codes and K the total number of loops:

Step 1 Take the mth sequences $\mathbf{p}_m$ out of the set $\mathbf{P}$, which contains all M sequences, and add Gaussian noise onto the sequence and store it in $\mathbf{y}(k)$. Increment m.

Step 2 Expand $\mathbf{y}(k)$ corresponding to the desired Volterra order.

Step 3 Compute $\mathbf{R}_{vv}(k)$ and $\mathbf{r}_{xv}(k)$ and add the result onto $\mathbf{R}_{vv}(k-1)$ and $\mathbf{r}_{xv}(k-1)$.

Step 4 Increment k and goto step 1 until $k = K$

Step 5 Dividing $\mathbf{R}_{vv}(K)$ and $\mathbf{r}_{xv}(K)$ by K will result in $\widehat{E}[\mathbf{R}_{vv}]$ and $\widehat{E}[\mathbf{r}_{xv}]$.

This method has two constraints, which are:

- $K = i \times M$, where i is an integer
- $K \gg M$

[2]Note, here, this M corresponds to the number of signal combinations and is not related to the Volterra expansion length.

If these constraints are not satisfied, then $\widehat{E}[\mathbf{R}_{vv}]$ and $\widehat{E}[\mathbf{r}_{xv}]$ are biased, and the filter weights are incorrect. There is no rule of thumb, which suggests a number K, since the noise power and the chosen random sequence of the PN generator influence the result, thus K has been found empirically. For small K such as $K = M \leq 100$, this technique does not result in a good estimate of the added noise component. Moreover, the products between spreading code and noise are insufficiently estimated which is even more severe. This is due to the fact that the signal power is the dominant part when $U \gg 1$, and the estimate of the autocorrelation matrix and crosscorrelation vector is less affected by the contributions due noise. These effects are shown in Figure 5.5 for different K for the 1st order Volterra system in an AWGN scenario, since the exact MMSE solution in known and can serve as a reference. Generally, as the total signal power increases (as u increases), the higher signal power to noise power ratio results in a much better estimate and $\widehat{E}[\cdot] \approx E[\cdot]$ since the signal power dominates.

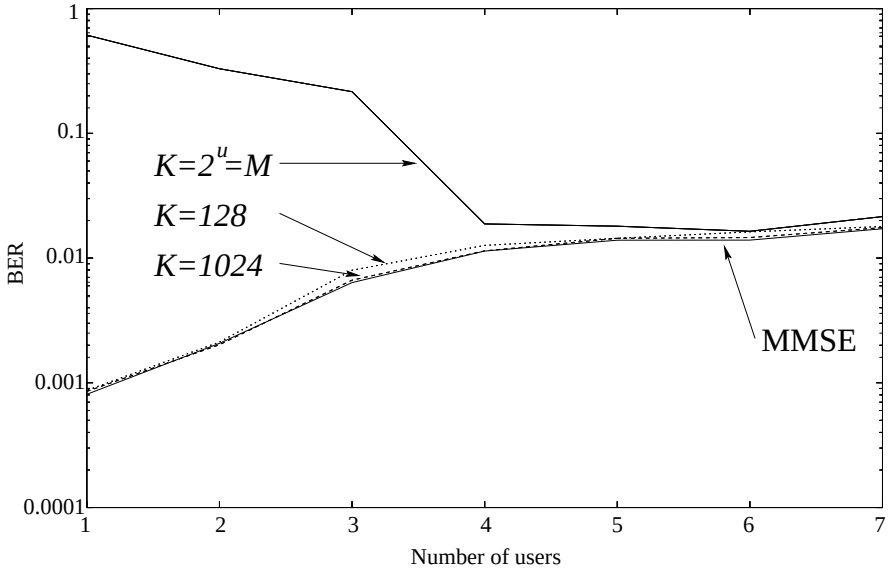


Figure 5.5: BER against the number of users u for different parameters of K for an estimated 1st-order Volterra receiver over an AWGN channel, $E_b/N_0 = 7$dB and 7 chip spreading codes.

Figure 5.5 shows that for $K \gg M \geq 1024$ and $M = 2^U$, then the filter performance converges to the MMSE performance. It also shows that $K = 1024$ loops are sufficient to achieve near MMSE performance, although $K = 128$ loops suggest a good estimate. However, the quality of Monte-Carlo simulations depend on the length of the simulation, since random generators may influence the outcome, which was found to be the case for one or two active users. Thus, if K is not chosen to be large enough, then these effects may result in insufficient estimated

values. Therefore, in order to reduce these effects, a large K is advantageous such as $1024 \leq K \leq 2^{L \times U}$, where $L = 1$ in a non-dispersive channel and $L = 3$ in a multipath channel.

5.6 Simulation Results

This section presents Monte-Carlo simulation results obtained from a CLB DS-CDMA system for the desired user, which is user one. Due to the binomial growth of the Volterra complexity, only short spreading sequences were considered with length $N = 7$. Two different spreading codes were used, randomly generated sequences and Gold codes [166]. Five receiver structures are compared against each other, a 3rd-order Volterra receiver (VS3), a 5th-order Volterra receiver (VS5), the established linear MF or RAKE [46], the linear MMSE [107] and a nonlinear RBF [112] receiver. The first two graphs show the performance for the AWGN scenario, the next two results for a stationary AWGN multipath channel. In both scenarios, the SNR was set to $E_b/N_0 = 7$dB. Both Volterra receiver coefficients were determined by the estimation technique where $K = 8192$ for $L = 1$, and $8192 \leq K \leq 2^{3 \times U}$ for $L = 3$ (the number of channel taps).

5.6.1 AWGN channel

Figure 5.6 shows the BER performance for randomly generated spreading sequences. The optimum performance is given by the RBF[3] and serves as a lower performance bound. An upper bound for this investigation is given by the MMSE receiver, since the MMSE is a subset of the group of Volterra receivers. The MF suffers from MAI and performs badly as the MAI increases. Both Volterra receivers outperform the linear MMSE. Between two and six users, VS5 has around half the error rate of the MMSE, and VS3 performs around 30% better than the MMSE receiver. Figure 5.7 shows results obtained for Gold codes. The performance appears to be flatter due to the lower correlation between the codes.

[3]This receiver is discussed in the next chapter.

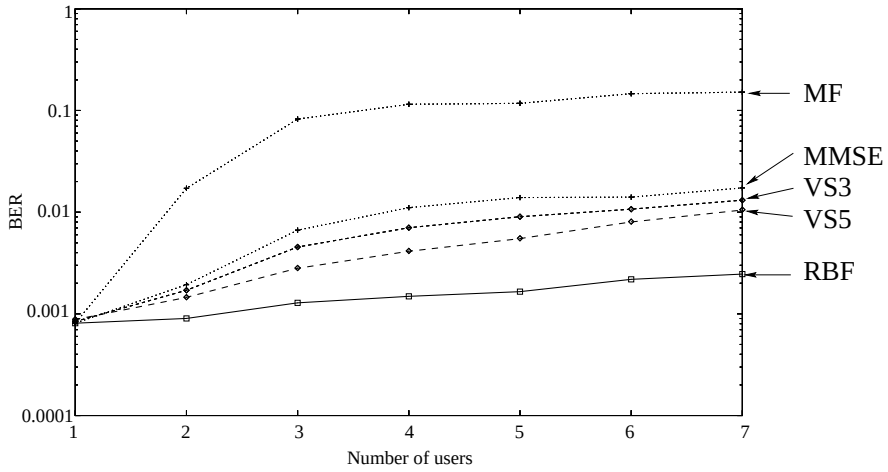


Figure 5.6: BER against the number of users for randomly generated seven chip spreading codes and $E_b/N_0 = 7$dB in AWGN.

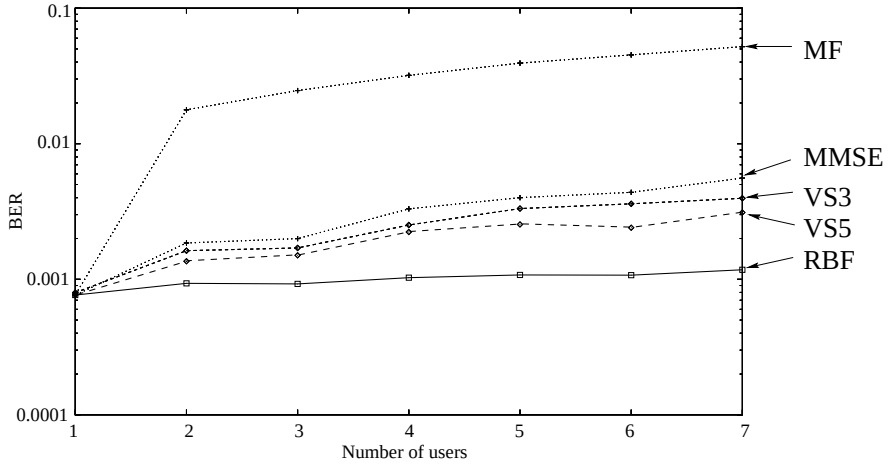


Figure 5.7: BER against the number of users for seven chip Gold codes and $E_b/N_0 = 7$dB in AWGN.

5.6.2 Multipath channel

Figure 5.8 and 5.9 show results obtained from a channel with impulse response $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$. All receiver structures exploit $N + (L - 1) = 9$ chips instead of seven, and take ISI induced from the previous and next symbol into account. The simulation has been stopped after six users due to the growing complexity of the Volterra and RBF receiver (2^{3U} possible sequences). The three finger RAKE receiver performs poorest followed by the MMSE. The RBF performs best, since it is a maximum likelihood symbol detector. Both Volterra structures show a significant improvement in performance compared with the results obtained for the AWGN scenario with respect to the MMSE. However, the price paid for this performance gain is the increased filter complexity, as discussed in [167].

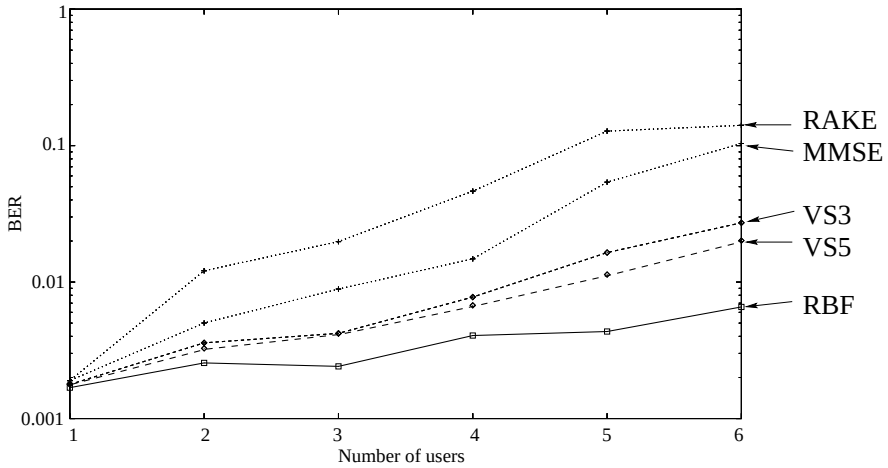


Figure 5.8: BER against the number of users for randomly generated seven chip spreading codes in multipath with $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$ and $E_b/N_0 =$ 7dB.

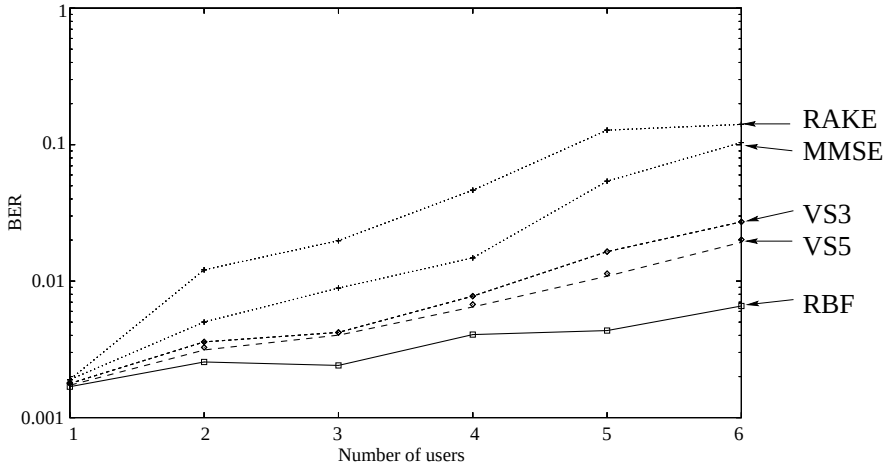
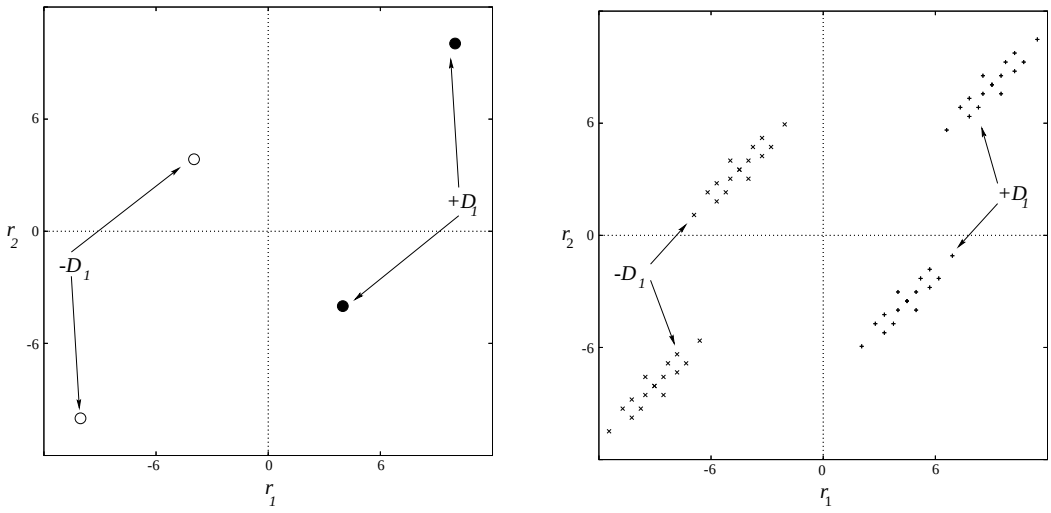


Figure 5.9: BER against the number of users for seven chip Gold codes in multipath with $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$ and $E_b/N_0 = 7$dB.

5.7 Decision boundaries

This section presents the decision boundaries obtained from PPB Volterra receivers. All CDMA scenarios use randomly generated spreading codes with length $N = 7$. First, the boundaries for two users are presented followed by three user scenarios, in both cases, for AWGN and a stationary multipath channel.

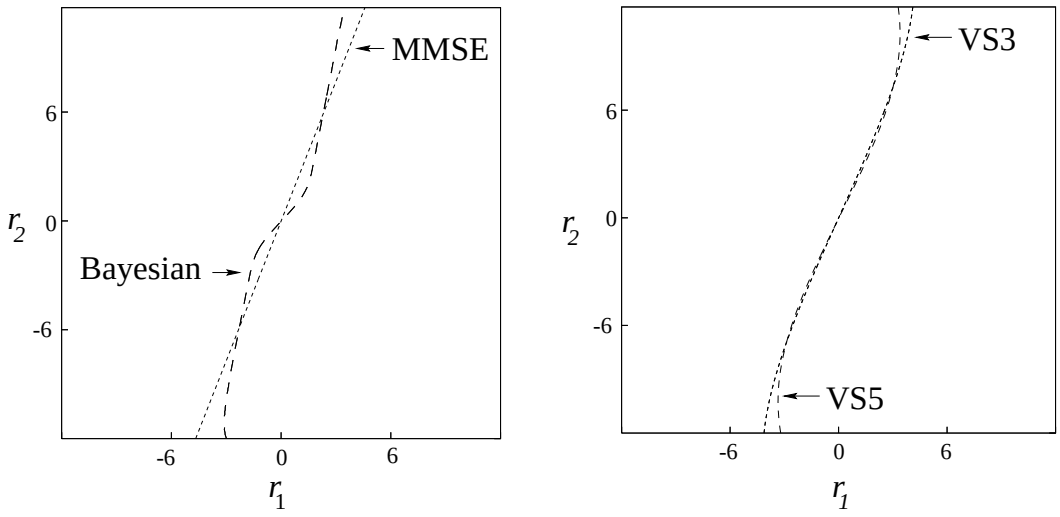


(a) *In AWGN.* **(b)** *In a stationary multipath.*

Figure 5.10: Signal constellations for two user PPB CDMA scenarios. The points belonging to $+D_1$ and $-D_1$ correspond to all possible transmitted signals.

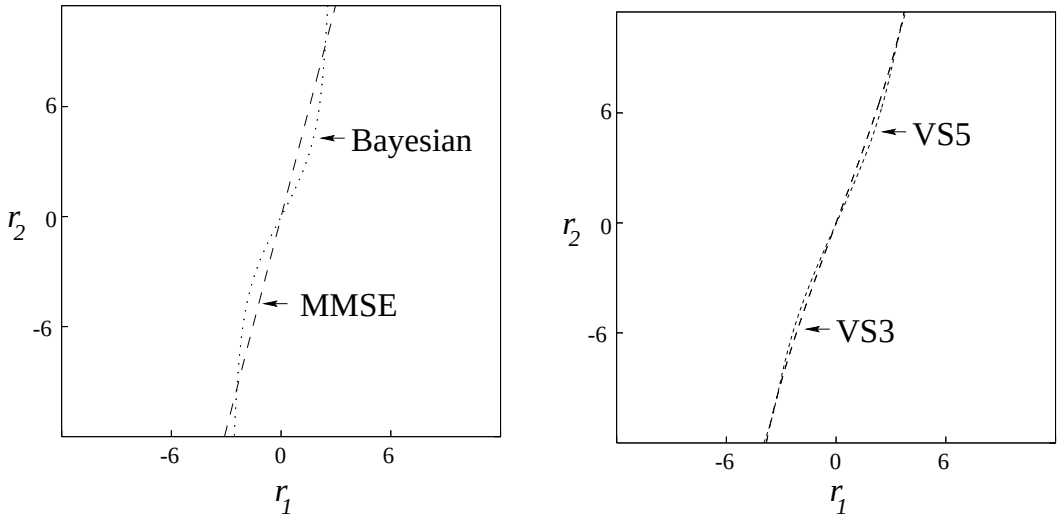
The expected signals for the two user scenario in AWGN are depicted in Figure 5.10(a) which are the rows of the generating matrix $\mathbf{R}^{AWGN}$. Figure 5.10(b) presents the points for a stationary multipath channel with $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$ which are the rows of the generating matrix $\mathbf{R}^{MP}$. The decision boundaries for the AWGN scenario depicted in Figure 5.10(a) is given in Figures 5.11(a) and 5.11(b).

The difference between a linear and the optimum boundary can be seen in Figure 5.11(a). It shows clearly that the MMSE lacks the ability to construct the optimum decision surface, especially near the origin. Both, the 3rd and 5th order Volterra based receiver lack this ability too, see Figure 5.12(b). This is mainly due to the low Volterra order chosen for practical reasons, and the limitation of the Volterra series in approximating. Even so, a performance gain is achieved in the regions far away from the origin of Figure 5.11(b).



(a) *Bayesian and MMSE receiver.* **(b)** *3rd and 5th-order Volterra receiver.*

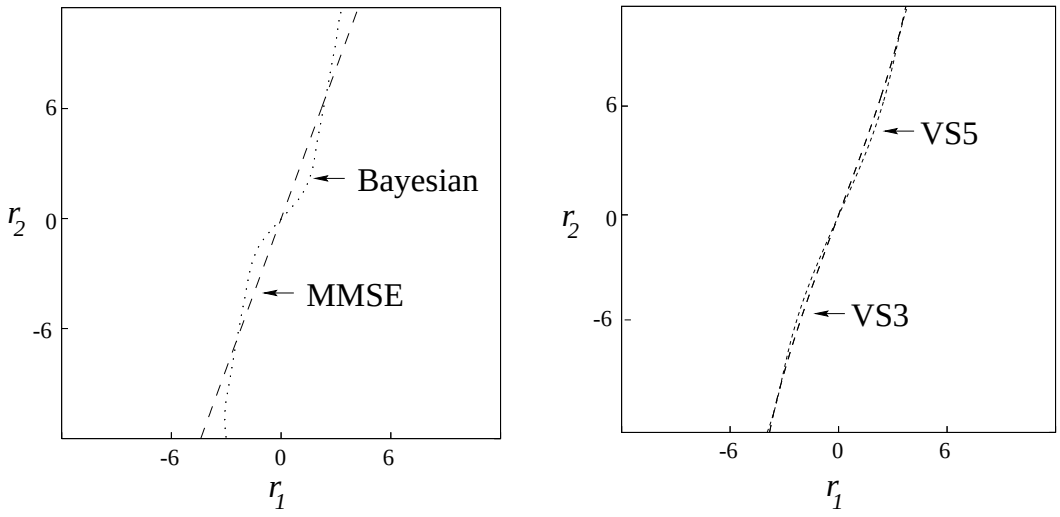
Figure 5.11: The decision boundaries in AWGN with $E_b/N_0 = 7$dB, derived from a two user PPB CDMA system.



(a) *For two receiver structures.* **(b)** *For two Volterra receivers.*

Figure 5.12: The decision boundaries in a stationary multipath channel with $E_b/N_0 = 0$dB, derived from a two user PPB CDMA system.

The decision boundaries for the stationary multipath scenario given in Figure 5.10(b) are presented for two different SNRs. Figures 5.12(a) and 5.12(b) show the boundaries for $E_b/N_0 = 0$dB, and Figure 5.13(a) and Figure 5.13(b) show the boundaries for $E_b/N_0 = 7$dB. Due to the high noise power (in the first two pictures), the difference of the curve near the origin is reduced and smoothed, which enables the Volterra series to approximate it well. However, since the decision boundary is relatively straight, the performance gain of a nonlinear structure over the MMSE is limited.



(a) *For two receiver structures.* **(b)** *For two Volterra receivers.*

Figure 5.13: The decision boundaries in a stationary multipath channel with $E_b/N_0 = 7$dB, derived from a two user PPB CDMA system.

Results for the three user case are presented next. Figures 5.14, 5.15 and 5.16 show the decision surface for the MMSE, 5th-order Volterra (VS5) and the Bayesian receiver in an AWGN scenario with $E_b/N_0 = 0$dB. The points (all possible received signals) are shown in Figures 4.7. There is little difference between the shape of the MMSE and VS5, whereas the Bayesian receiver clearly shows a nonlinear shape.

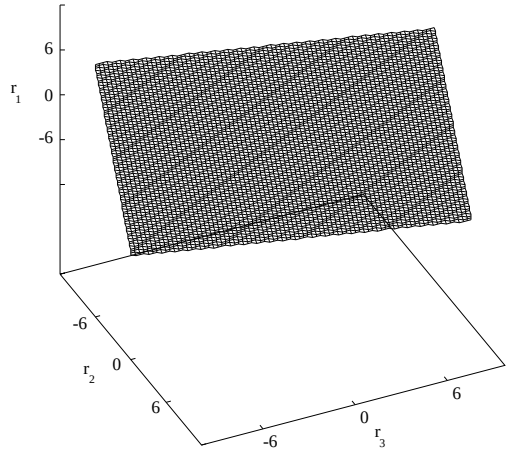


Figure 5.14: The decision surface for the MMSE receiver for three users in an non-dispersive AWGN channel and $E_b/N_0 = 0$dB.

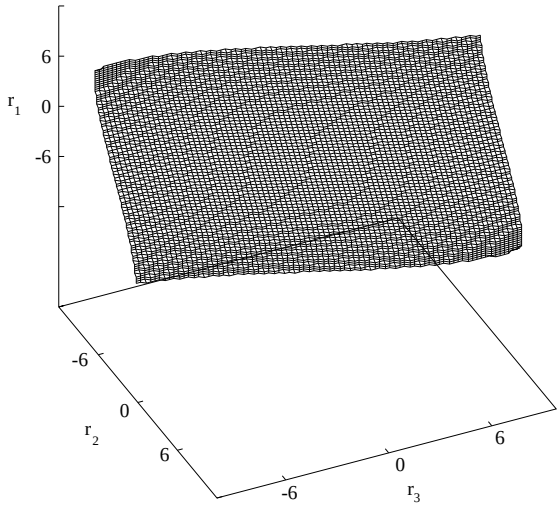


Figure 5.15: The decision surface for the 5th-order Volterra (VS5) receiver for three users in an AWGN channel and $E_b/N_0 = 0$dB.

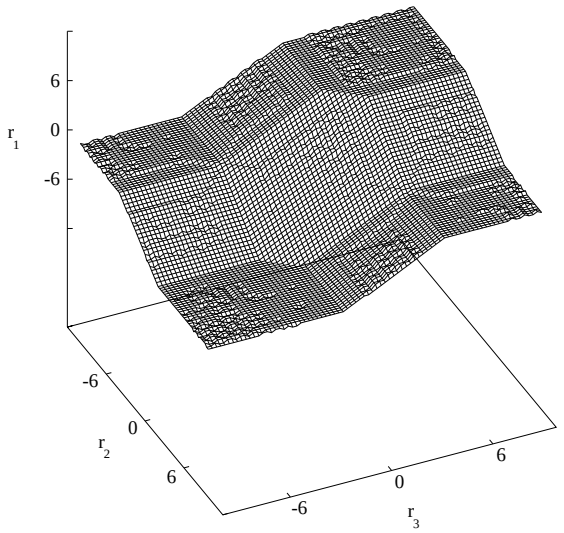


Figure 5.16: The decision surface for the Bayesian receiver for three users in an AWGN channel and $E_b/N_0 = 0$dB.

Figure 4.9 shows the points for a three user scenario in multipath, and the decision surfaces obtained for this stationary multipath are presented in Figure 5.17, 5.18 and 5.19 with $E_b/N_0 = 0$dB. It can be seen that the surface of the 5th-order Volterra receiver is nonlinear, although not as pronounced as the Bayesian structure.

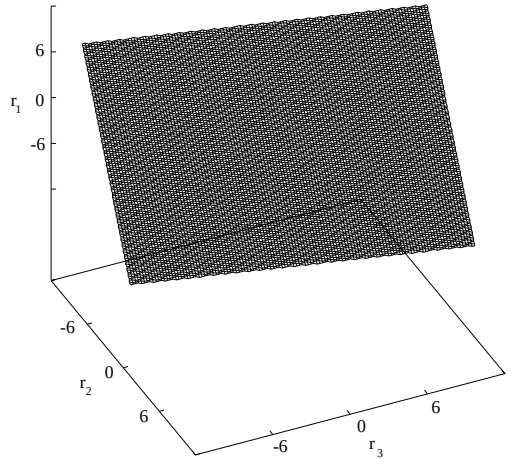


Figure 5.17: The decision surface for the MMSE receiver in multipath with $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$ for three users and $E_b/N_0 = 0$dB.

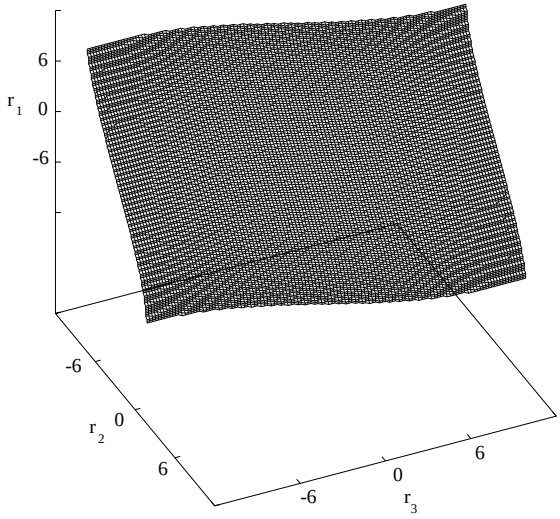


Figure 5.18: The decision surface for VS5 receiver in multipath for three users and $E_b/N_0 = 0$dB.

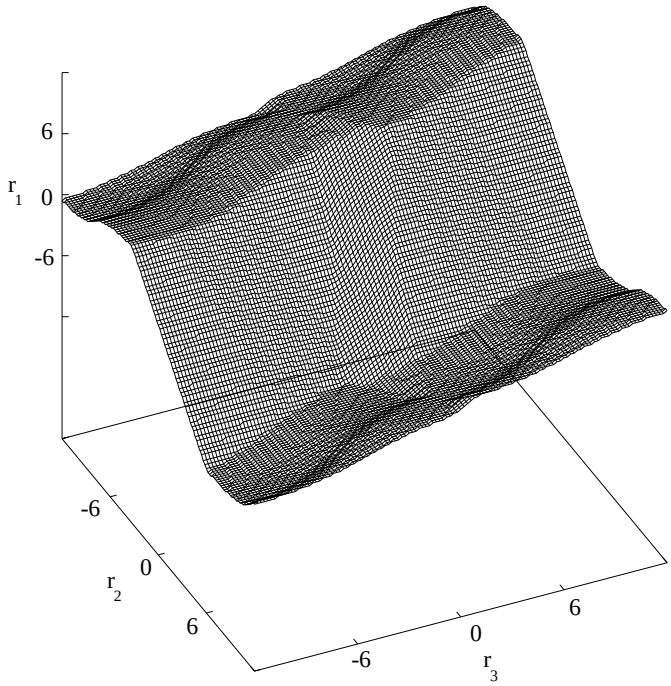


Figure 5.19: The decision surface for the Bayesian receiver in multipath for three users and $E_b/N_0 = 0$dB.

5.8 Discussion

This chapter presented a novel nonlinear CLB receiver structure based upon the Volterra series expansion. The receiver performs a nonlinear mapping first, the Volterra expansion, followed by a linear mapping, the FIR filtering. It has been shown that the VS expansion introduces higher-order components which are tedious to calculate for higher orders. This makes it complicated to derive the expected value of the terms within the expansion sequence if U or N are large.

It also has been shown that the 1st-order Volterra receiver is equivalent to the known CLB MMSE receiver. Investigations conducted on the 2nd-order Volterra receiver revealed that the 2nd-order crosscorrelation vector reduces to the 1st-order crosscorrelation vector if the signals are equiprobable and antipodal. Hence, there exists no even-order CLB Volterra receiver, and higher odd-order Volterra receivers must be applied. This implies that Volterra receivers become complex when higher-order systems are considered. In order to derive the filter coefficients for such complex systems, an algorithm which computes them has been searched for. A literature search was not fruitful. Attempts made to derive such an algorithm which computes the Volterra filter coefficients by computing the autocorrelation matrix and the crosscorrelation vector failed due to the unique nature of the problem. It has been shown that no straightforward algorithm can compute all matrix and vector terms for the user as a parameter. This is due to the VS expansion which induces products of higher-order, based upon the statistical assumptions made for noise and data bits. An alternative approach, which depends upon an estimation technique has been presented. It is also possible to make the Volterra structure adaptive [168], but this has not been considered since it was already the subject of previous publications. Moreover, due to the large eigenvalue spread, especially in DS-CDMA, the convergence is very slow for higher-order systems. Simulation results obtained from Monte-Carlo simulations compared two Volterra structures (3rd and 5th-order) against established receivers. It has been shown that both VS receivers outperform the MMSE. However, the cost of this performance improvement is its complexity. It may be argued that the performance gain is little compared to the filter complexity needed. However, explanations were given by comparison of the decision boundaries of the Volterra structures against the MMSE and the Bayesian ones. Since a low Volterra order has been chosen, the Volterra decision boundary is far from optimum, and hence its performance is likewise.

The proposed Volterra receiver showed superior performance compared with the MMSE receiver. In order to implement the proposed Volterra receiver, an algorithm should be derived

which enables the system designer to compute the Volterra filter coefficients exactly, instead of using the estimation technique. Despite its complexity, it may be an alternative compared with the ML receiver complexity for very slow changing or stationary channels or it may be worth considering as an adaptive receiver structure.

Chapter 6
Radial basis function receivers

This chapter is concerned with the RBF receiver, which is applied to DS-CDMA as a deterministic (neural) network.

The first section reviews the established CLB RBF receiver. The following section presents a PPB RBF receiver structure. There the main drawback appears to be its large network size in a multipath environment, which makes it too complex for practical implementations. Therefore a smaller network with equivalent performance is desirable. Thus the next section proposes a RBF network with reduced complexity, followed by results obtained from Monte-Carlo simulations. Finally, the chapter ends with a discussion.

6.1 CLB RBF reviewed

A class of network models, which possess universal approximation capabilities is the radial basis function network [97, 143]. The RBFN is also referred to as neural network in [128], where its adaptive characteristics have been investigated.

RBF methods perform an approximation of mappings from a set of data points in a multi dimensional space [98]. The approximation problem requires a mapping for every input vector onto a target vector. In CDMA the mapping is performed from the N-dimensional input space (processing gain) of $\mathbf{y}(k)$ for transmitted symbol k, to a 1-dimensional target (output) space, which is the bit of the desired user, D_d. The data set consists of all M, M may be 2^U, possible received signals $\mathbf{y}(m)$ for $m = 1, 2, \ldots, M$, together with corresponding targets $D_d(m)$, the desired user bit, which can be obtained from a generating matrix. For clarity, $\mathbf{y}(m)$ shall be denoted as $\mathbf{y}_m$ and corresponds to the mth row of a generating matrix, e.g. $\mathbf{P}$ in equation (2.8). Ideally, there exists a function $f(\mathbf{y})$ such that:

$$f(\mathbf{y}_m) = D_d(m) \qquad m = 1, 2, \ldots, M. \tag{6.1}$$

The RBF approach [98, 144] introduces a set of M basis functions, one for each data point.

They take the form $\phi(\|\mathbf{y}(k) - \mathbf{y}_m\|)$ where $\phi(\cdot)$ may be a nonlinear function. Thus each function depends on the Euclidean distance $\|\mathbf{y}(k) - \mathbf{y}_m\|$ between the received signal $\mathbf{y}(k)$ and the legitimate data point $\mathbf{y}_m$. Many functions for $\phi(\cdot)$ have been proposed, see section 4.4, however the Gaussian is the most common. The network output is a linear combination of M weighted basis functions:

$$f(\mathbf{y}(k)) = \sum_{m=1}^{M} w_m \, \phi(\|\mathbf{y}(k) - \mathbf{y}_m\|). \tag{6.2}$$

The received signal $\mathbf{y}(k)$ consists of the signal component and added Gaussian noise. Since the noise component $g(n)$ added onto each vector element $\{y_n(k)\}$ is neither correlated with the signal nor other noise components, $y(k)$ has a univariate normal density, defined as:

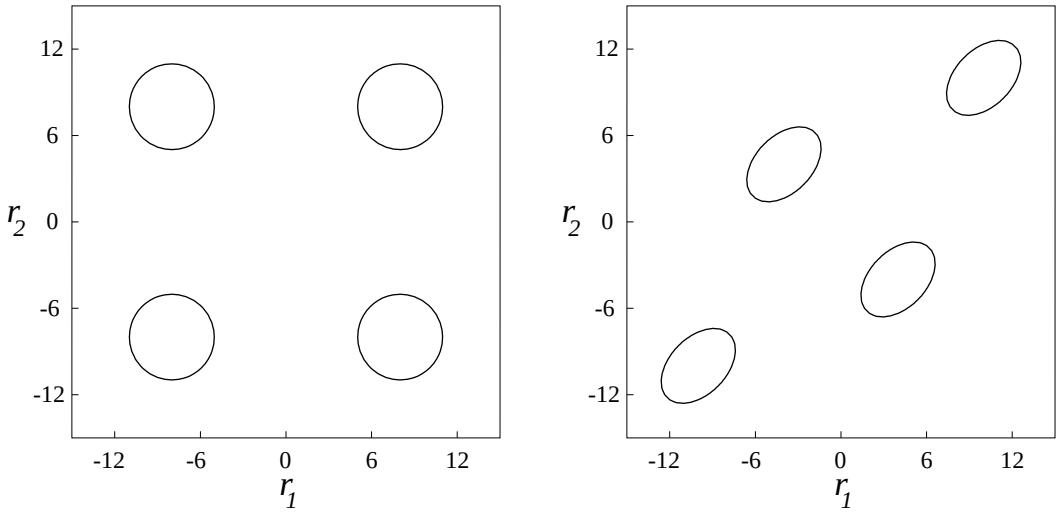
$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \, \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \tag{6.3}$$

for which $\mu = E[x]$ and $\sigma^2 = E[(x-\mu)^2]$. The univariate normal density is fully described by its mean μ and its variance σ^2. These normally distributed samples tend to cluster about the mean with a spread proportional to the standard deviation σ [102]. This fact results in circular distributed signals $\mathbf{y}(k)$ about its mean. Figure 6.1(a) illustrates this fact for a two user PPB CDMA system. There the preprocessed signal $\mathbf{r}$ forms circular clusters about the four possible mean (centres), where the diameter of each circle is proportional to σ^2, the noise power. The circles were derived from the contour lines of the Bayesian decision surface, the circles represent a slice of a Gaussian shaped bell, which represents the noise spread, see also Figure 4.5.

Therefore, the CLB RBF for CDMA has a Gaussian basis function and is defined by:

$$f(\mathbf{y}(k)) = \sum_{m=1}^{M} w_m \, \exp\left(-\frac{\|\mathbf{y}(k) - \mathbf{y}_m\|^2}{2\sigma^2}\right). \tag{6.4}$$

The coefficients w_m, known as weights, can be derived according to (4.17), which also enables the network to be adaptive, while exploiting a *gradient descent* algorithm [128]. The parameter σ^2 controls the smoothness or spread about a mean $\mathbf{y}_m$ of the approximating function and represents the noise power. The parameter $\mathbf{y}_m$ is known in the literature as the RBF *centre* and shall be denoted from now onwards as $\mathbf{c}_m$. If the data set of all possible received signals is known, $\mathbf{c}_m$ is known and the network is determined. Otherwise, $\mathbf{c}_m$ must be constructed



(a) *Noise is uncorrelated.* **(b)** *Noise is correlated.*

Figure 6.1: The noise corrupted signal clusters differently about the four possible mean in a two user PPB CDMA system.

according to an established clustering algorithm [128, 144]. Of course, the network is only fully determined, if w_m and σ^2 are also known. Assuming that the noise power is known, e.g. from measurements, then w_m and $\mathbf{c}_m$ are the remaining two unknowns. In CDMA both can be pre-calculated. Figure 6.2 shows a CLB RBFN.

The network consists of four parts. First, the input layer, which is at the left hand side, where the synchronised N chips of $\mathbf{y}(k)$ are fed into the network. The next layer is the hidden layer, which consist of M centres (also called neurons, units or nodes), and represent the points around which the received signals lie or cluster. Each centre is connected to all chips of the input layer. The hidden layer performs a mapping from the input space $\mathbb{R}^N$ to the hidden space $\mathbb{R}^M$. The output of the mth centre is determined by $\phi_m(\mathbf{y}(k)) = \phi(\|\mathbf{y}(k) - \mathbf{c}_m\|)$. Next, the output of each centre $\phi_m(\mathbf{y})$ is weighted by w_m and summed over all M values. This is a mapping from $\mathbb{R}^M$ to $\mathbb{R}$. Finally, this sum is sliced (for antipodal signalling by a step function with outputs $\{+1, -1\}$) and the estimate of the desired user bit is produced, hence: $\widehat{D}_d(k) = \text{sgn}(f(\mathbf{y}(k)))$.

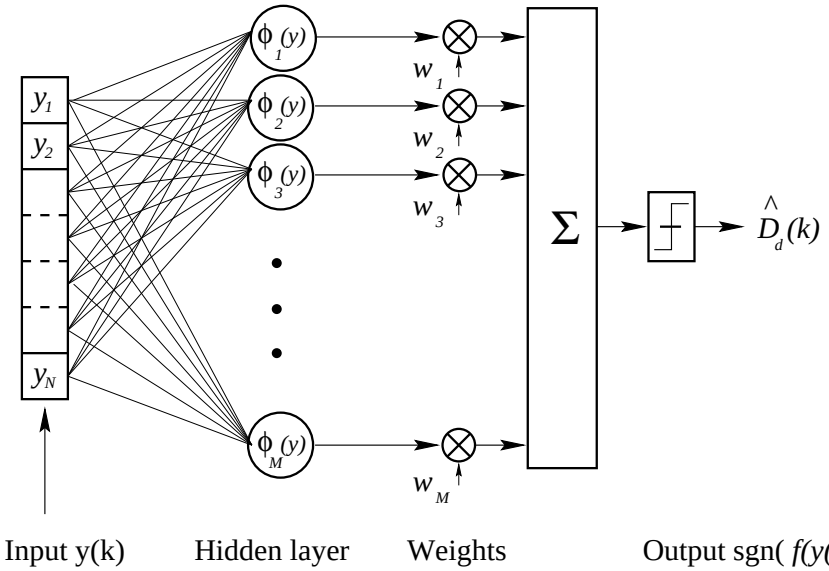


Figure 6.2: The structure of the CLB RBF network.

6.1.1 CLB RBF centre construction

Consider the construction of all possible received signals in CDMA for U users in a memoryless (AWGN) channel with their unique spreading sequences. These signals form a set with $M = 2^U$ points in $\mathbb{R}^N$ and are stored in a $(M \times N)$ matrix $\mathbf{P}^{AWGN}$, where each row represents a point, hence $[\mathbf{p}_1\ \mathbf{p}_2 \ldots \mathbf{p}_M]^T$. The spreading codes are stored in a $(U \times 1)$ partitioned matrix $\mathbf{C}$ with vector elements $[\mathbf{c}_1\ \mathbf{c}_2 \ldots \mathbf{c}_U]^T$. The notation $\{\mathbf{c}_u\}$ indicates the uth spreading sequence, whereas $\{\mathbf{c}_m\}$ shall refer to the mth RBF centre. Instead of using the term signal, the term point shall be used, since each signal vector can be seen as a point in the vector space. Bearing this in mind, all possible points are derived from:

$$\begin{aligned}\mathbf{P}^{AWGN} &= \mathbf{B}\,\mathbf{C} \qquad (6.5)\\ &= \begin{bmatrix} 1 & 1 & \ldots & 1 & 1 \\ 1 & 1 & \ldots & 1 & -1 \\ 1 & 1 & \ldots & -1 & 1 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ -1 & -1 & \ldots & 1 & -1 \\ -1 & -1 & \ldots & -1 & 1 \\ -1 & -1 & \ldots & -1 & -1 \end{bmatrix} \begin{bmatrix} \mathbf{c}_1^T \\ \mathbf{c}_2^T \\ \mathbf{c}_3^T \\ \ldots \\ \mathbf{c}_{U-2}^T \\ \mathbf{c}_{U-1}^T \\ \mathbf{c}_U^T \end{bmatrix} = \begin{bmatrix} \mathbf{c}_1^T + \mathbf{c}_2^T + \cdots + \mathbf{c}_{U-1}^T + \mathbf{c}_U^T \\ \mathbf{c}_1^T + \mathbf{c}_2^T + \cdots + \mathbf{c}_{U-1}^T - \mathbf{c}_U^T \\ \mathbf{c}_1^T + \mathbf{c}_2^T + \cdots - \mathbf{c}_{U-1}^T + \mathbf{c}_U^T \\ \ldots \\ -\mathbf{c}_1^T - \mathbf{c}_2^T - \cdots + \mathbf{c}_{U-1}^T - \mathbf{c}_U^T \\ -\mathbf{c}_1^T - \mathbf{c}_2^T - \cdots - \mathbf{c}_{U-1}^T + \mathbf{c}_U^T \\ -\mathbf{c}_1^T - \mathbf{c}_2^T - \cdots - \mathbf{c}_{U-1}^T - \mathbf{c}_U^T \end{bmatrix}, \end{aligned}$$

where $\mathbf{B}$ is a $(M \times U)$ combination matrix with all possible binary signal combinations as its rows, and code vector $\mathbf{c}$ having all user spreading sequences $\mathbf{c}_u$ as its elements. Column u of $\mathbf{B}$ becomes the RBF weight vector $\mathbf{w}$ if the RBF is thought of as a single user detector for the uth user [112]. Elements $[\mathbf{p}_1\ \mathbf{p}_2 \ldots \mathbf{p}_M]^T$ consist of the sum of spreading codes in equation (6.5) and become the M CLB RBF centres $[\mathbf{c}_1\ \mathbf{c}_2 \ldots \mathbf{c}_M]^T$. The $(M \times 1)$ vector $\mathbf{p}^{AWGN}$ with its vector elements can also be denoted as a $(M \times N)$ matrix $\mathbf{P}^{AWGN}$, since the spreading codes $\mathbf{c}_u$ are of length N. Now, the RBF is constructed for the memoryless (Gaussian) scenario with 2^U centres, where each centre has vector length N. Note the notation used for spreading codes $\mathbf{c}_u$, centres $\mathbf{c}_m$ and points $\mathbf{p}_m^{AWGN}$.

The RBF centre construction for channels with memory is computationally much more demanding, and can be carried out in different ways. Three different approaches shall be discussed. Further, it is assumed that a perfect estimate of the L-tap channel impulse response H_{ch} (chip spaced) is available at the receiver.

Full multipath Centres $\mathbf{p}^{FM}$ are derived from the previous, current and next symbol sequence, hence, 2^{3U} centres with length $(N + L - 1)$.

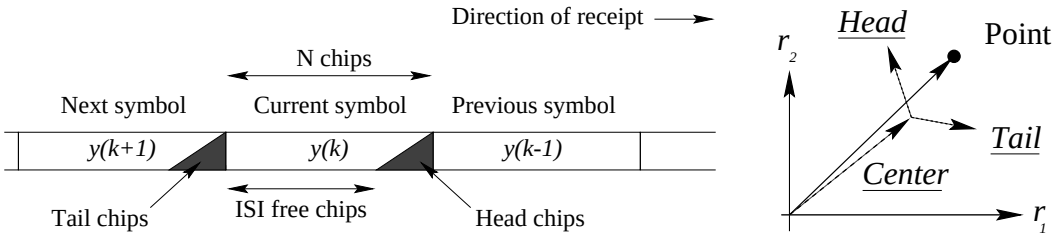
Reduced multipath Centres $\mathbf{p}^{RM}$ are derived from the previous and current symbol sequence, results in 2^{2U} centres with length $(N + L - 1)$.

Extended Gaussian Centres $\mathbf{p}^{EG}$ are derived from the memoryless centres by convolving each centre with H_{ch}, hence the number of centres is 2^U and the vector length of each centre is $(N + L - 1)$.

The different aspects of constructing RBF centre vectors shall be discussed in conjunction with Figure 6.3(a). Here a fully synchronised receiver is assumed. IS-95 has a three finger RAKE receiver implemented at the mobile, which combines the three strongest multipath components [16, 169]. This structure can be seen as an FIR filter, whose filter weights are equal to the convolution of the spreading sequence of the desired user with H_{ch}[1]. Thus, there are $(N+L-1)$ chips of $\mathbf{y}(k)$ taken into account to estimate the transmitted bit $\widehat{D}_d$. Symbol $\mathbf{y}(k)$ consists of two ISI affected chip sequences named head and tail, see Figure 6.3(a). The $(L-1)$ head chips are affected by the previous transmitted symbol sequence, whereas the tail chips are the head

[1]The channel coefficients used at the IS-95 RAKE receiver may not be adjacent (e.g. $z^0, z^{-1}, z^{-2}, \ldots$) as it is assumed in this work.

chips of the next symbol, into which the current symbol spreads. In order to take all the energy of the current symbol into account, the chip or vector span at the receiver must be $(N + L - 1)$. Due to the fact that a RAKE receiver can implement a *maximum ratio combining* (MRC) or an *equal gain combining* (EGC) receiver, the term RAKE shall be replaced by MRC or EGC in order to specify the structure used [16, 46]. Another approach is to take a MF receiver instead of a RAKE. This results in a much simpler receiver structure. Of course, this is at the cost of severe performance degradation since ISI and $inter\ chip\ interference$ (ICI) is not appropriately taken into account.



(a) *From a chip level perspective.*

(b) *From a vector perspective.*

Figure 6.3: The effect of ISI on the chips within a symbol causes interference. The head and tail chips are understood as rotating vectors, which depend on the transmitted user bits in the previous and next symbol, and cause the preprocessed signal (point) to lie differently (spread).

Figure 6.3(a) depicts the statements made so far. A pure MF receiver would suffer from the ISI affected head chips and the neglected ICI. Hence, it is not advisable to implement a simple MF as the receiver in a multipath environment. The first step to improve system performance is by taking H_{ch} into account and combating ICI. The extended Gaussian RBF centres fulfill this requirement. It provides the RBFN with a reasonable number of centres, where the matrix containing all RBF centres has a size of $(2^U \times (N + L - 1))$.

The next step is to take at least half of the ISI into account, induced from the past symbol, named reduced multipath. This increases the size of the matrix containing all centres to $(2^{2U} \times (N + L - 1))$.

Finally, the structure which corresponds best to the maximum likelihood symbol by symbol detection scheme is named the full multipath implementation, since all possible received symbols become centres. Of course, its complexity is prohibitively high for practical implementations

since the matrix $\mathbf{P}^{FM}$ containing all centres has size $(M \times (N + L - 1))$ with $M = 2^{3U}$. The centres of the full multipath implementation are given by the product between an extended code matrix $\mathbf{C}^{ECM}$ and a matrix $\mathbf{H}$ describing the channel characteristics. The extended code matrix shall be of size $(M \times 3N)$, where the first N columns represent the previous, the next N columns the current and the last N columns the next symbol. Again, the starting point is a combination matrix $\mathbf{B}$, but this time of size $(M \times 3U)$, thus:

$$\mathbf{B}^{ECM} = \begin{bmatrix} 1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 \\ 1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & -1 \\ 1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ -1 & \dots & -1 & -1 & \dots & -1 & -1 & \dots & -1 \\ -1 & \dots & -1 & -1 & \dots & -1 & -1 & \dots & 1 \\ -1 & \dots & -1 & -1 & \dots & -1 & -1 & \dots & -1 \end{bmatrix}. \tag{6.6}$$

In order to simplify the notation, $\mathbf{B}^{ECM}$ is partitioned into three sub-matrices, thus:

$$\mathbf{B}^{ECM} = [\mathbf{B1}\ \mathbf{B2}\ \mathbf{B3}], \tag{6.7}$$

where each sub-matrix has size $(M \times U)$. The next step is to compute the $(M \times 3N)$ matrix $\mathbf{C}^{ECM}$ with the Hadamard product. Therefore, each sub-matrix is multiplied by the spreading codes. This results in three $(M \times N)$ sub-matrices, but since they are adjacent (6.7) the rows are of length $3N$:

$$\mathbf{C}^{ECM} = \begin{bmatrix} \mathbf{B1} \begin{bmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_U^T \end{bmatrix} & \mathbf{B2} \begin{bmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_U^T \end{bmatrix} & \mathbf{B3} \begin{bmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_U^T \end{bmatrix} \end{bmatrix}. \tag{6.8}$$

Finally, the $(M \times (N+L-1))$ matrix $\mathbf{P}^{FM}$ containing all possible sequences for three adjacent symbols is constructed. In order to find $\mathbf{P}^{FM}$ the matrix $\mathbf{H}$ has to be derived. The channel impulse response H_{ch} with coefficients $\{h_1, h_2, \dots, h_L\}$ is stored in a $((N + L - 1) \times 3N)$ matrix $\mathbf{H}$ which is Toeplitz [47]. $\mathbf{H}$ does not have to be $(3N \times 3N)$ since only the current symbol together with its ISI affected head and tail chips is of interest. Thus, a $((N + L - 1) \times 3N)$ matrix is sufficient where the first $(N - L + 1)$ columns in $\mathbf{H}$ are zero. So for instance for

$L = 3$ and $N = 5$, it becomes:

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 \end{bmatrix}. \tag{6.9}$$

Then, all possible centres are found from the product between:

$$\mathbf{P}^{FM} = \mathbf{C}^{ECM}\,\mathbf{H}^T. \tag{6.10}$$

Matrix $\mathbf{H}$ for the extended Gaussian approach is given in (6.11). It is a truncated version of (6.9) with size $((N + L - 1) \times N)$ since no ISI has to be taken into account, hence:

$$\mathbf{H} = \begin{bmatrix} h_1 & 0 & 0 & \dots & 0 & 0 & 0 \\ h_2 & h_1 & 0 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 & h_L & h_{L-1} \\ 0 & 0 & 0 & \dots & 0 & 0 & h_L \end{bmatrix}. \tag{6.11}$$

The centres $\mathbf{P}^{EG}$ of the extended Gaussian RBF are given by the product between $\mathbf{P}^{AWGN}$ and $\mathbf{H}$. Thus, the M centres with length $(N + L - 1)$ are derived from the $((N + L - 1) \times N)$ channel matrix $\mathbf{H}$ and the M memoryless RBF centres in $\mathbf{P}^{AWGN}$:

$$\begin{aligned} \mathbf{P}^{EG} &= \mathbf{P}^{AWGN}\mathbf{H}^T = [\mathbf{p}_1\ \mathbf{p}_2 \cdots \mathbf{p}_M]^T\,\mathbf{H}^T \\ &= \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \dots \\ \mathbf{p}_{M-1} \\ \mathbf{p}_M \end{bmatrix} \begin{bmatrix} h_1 & 0 & 0 & \dots & 0 & 0 & 0 \\ h_2 & h_1 & 0 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 & h_L & h_{L-1} \\ 0 & 0 & 0 & \dots & 0 & 0 & h_L \end{bmatrix}^T. \end{aligned} \tag{6.12}$$

The construction of $\mathbf{P}^{RM}$ is straightforward from the previous two derivations and shall not be discussed in detail.

6.2 Preprocessed based RBF receiver

It has already been pointed out that multiuser receivers exploit preprocessed signals, see Figure 3.5. Preprocessing can be done with different pre-filter structures. For the memoryless channel, matched filtering is optimum [46]. In multipath, the simple MF is no longer optimum since it does not take ISI effects into account, but receiver structures based on Price's RAKE [62] do. The preprocessing approach taken in this work rests upon the MRC, where the MRC is derived from the RAKE [16]. Since real data has been used, the MRC weights the output of each RAKE finger with its corresponding lth channel coefficient, while $(L-1)$ fingers process a delayed version of $\mathbf{y}(k)$, see Figure C.2 in appendix C. If the weights are unity, the RAKE structure corresponds to the EGC [16]. This process takes advantage of the fact that matched filters enhance the SNR, while in addition the signal dimensionality is reduced. Thus, more sophisticated receiver algorithms can be exploited in order to enhance the performance by combating MAI and ISI more effectively. However, the general approach is to assume uncorrelated noise at the receiver input. This is achieved by embedding a noise whitening filter between the preprocessor and the actual receiver. Generally, no attempts are made to take correlated noise into account due to the assumption made that the noise is uncorrelated[2], which simplifies the analysis. A noise whitening filter as such shall not be considered in this work[3].

The correlated nature of the noise is easily pointed out. All MFs match the same sampled sequence $\mathbf{y}(k)$ against their (user) specific sequence $\mathbf{c}_u$. Thus the noise becomes correlated, since the orthogonality among the spreading codes can be destroyed by multipath or when nonorthogonal spreading codes are employed in the first place. Therefore, all possible received

[2]Either due to whitening filters (preprocessing), or due to a crosscorrelation matrix as suggested in Joint-Detection [170, 171].

[3]But as the reader will notice, a decorrelating matrix will be used as part of a new metric instead.

PPB signals $\mathbf{r}$ in a memoryless channel are given by:

$$\begin{aligned}\mathbf{R}^{AWGN} &= \mathbf{P}\mathbf{C}^T = \mathbf{B}\mathbf{C}\mathbf{C}^T \qquad (6.13)\\ &= \begin{bmatrix} 1 & 1 & \dots & 1 & 1 \\ 1 & 1 & \dots & 1 & -1 \\ 1 & 1 & \dots & -1 & 1 \\ \dots & \dots & \dots & \dots & \dots \\ -1 & -1 & \dots & 1 & -1 \\ -1 & -1 & \dots & -1 & 1 \\ -1 & -1 & \dots & -1 & -1 \end{bmatrix} \begin{bmatrix} \mathbf{c}_1^T \\ \mathbf{c}_2^T \\ \mathbf{c}_3^T \\ \dots \\ \mathbf{c}_{U-2}^T \\ \mathbf{c}_{U-1}^T \\ \mathbf{c}_U^T \end{bmatrix} \begin{bmatrix} \mathbf{c}_1^T \\ \mathbf{c}_2^T \\ \mathbf{c}_3^T \\ \dots \\ \mathbf{c}_{U-2}^T \\ \mathbf{c}_{U-1}^T \\ \mathbf{c}_U^T \end{bmatrix}^T\end{aligned}$$

resulting in

$$\begin{aligned}\mathbf{R}^{AWGN} &= \mathbf{B}\mathbf{S} \qquad (6.14)\\ &= \begin{bmatrix} 1 & 1 & \dots & 1 & 1 \\ 1 & 1 & \dots & 1 & -1 \\ \dots & \dots & \dots & \dots & \dots \\ -1 & -1 & \dots & -1 & 1 \\ -1 & -1 & \dots & -1 & -1 \end{bmatrix} \begin{bmatrix} \mathbf{c}_1^T\mathbf{c}_1 & \mathbf{c}_1^T\mathbf{c}_2 & \dots & \mathbf{c}_1^T\mathbf{c}_{U-1} & \mathbf{c}_1^T\mathbf{c}_U \\ \mathbf{c}_2^T\mathbf{c}_1 & \mathbf{c}_2^T\mathbf{c}_2 & \dots & \mathbf{c}_2^T\mathbf{c}_{U-1} & \mathbf{c}_2^T\mathbf{c}_U \\ \dots & \dots & \dots & \dots & \dots \\ \mathbf{c}_{U-1}^T\mathbf{c}_1 & \mathbf{c}_{U-1}^T\mathbf{c}_2 & \dots & \mathbf{c}_{U-1}^T\mathbf{c}_{U-1} & \mathbf{c}_{U-1}^T\mathbf{c}_U \\ \mathbf{c}_U^T\mathbf{c}_1 & \mathbf{c}_U^T\mathbf{c}_2 & \dots & \mathbf{c}_U^T\mathbf{c}_{U-1} & \mathbf{c}_U^T\mathbf{c}_U \end{bmatrix},\end{aligned}$$

which results in the product between the combination matrix $\mathbf{B}$ and the crosscorrelation matrix $\mathbf{S}$ of the user codes. In the case of a memoryless channel and orthogonal spreading codes being used, $\mathbf{S}$ contains zeros on its off diagonals. The received signal $\mathbf{r}$ has vector elements $[\mathbf{r}_1\ \mathbf{r}_2 \dots \mathbf{r}_M]^T$ (the output of each MF), where each $\mathbf{r}_m$ is equivalent to the mth row in $\mathbf{B}$. Hence, the properties of signal $\mathbf{r}_m^T$ depend on $\mathbf{B}$. In the case of orthogonal spreading, the received signals (seen as points) will form a hypercube[4] in $\mathbb{R}^U$ centered at the origin, because $\mathbf{B}$ contains all possible binary combinations, which are the 2^U vertices for the U-dimensional space, see Figure 4.7(a). If nonorthogonal codes are employed, the hypercube becomes skewed due to $\mathbf{S}$, see Figure 4.7(b). The matrix $\mathbf{S}$ affects the noise statistics at the preprocessor output. The noise is no longer univariate normal as in equation (6.3) but multivariate normal distributed [155]:

$$p(\mathbf{r}) = \frac{1}{\sqrt{(2\pi)^U}\sqrt{|\mathbf{S}|}} \exp\left(-\frac{(\mathbf{r}-\mu)^T\,\mathbf{S}^{-1}\,(\mathbf{r}-\mu)}{2}\right), \qquad (6.15)$$

[4] A square in 2-dimensions, a cube in 3-dimensions, a hypercube in higher dimensions.

where $\mathbf{r}$ is a U-dimensional vector, μ is the U- dimensional mean vector, $\mathbf{S}$ is the $(U \times U)$ covariance matrix. For simplicity, (6.15) is abbreviated as $p(\mathbf{r}) \sim \mathcal{N}(\mu, \mathbf{S})$, where

$$\mu = E[\mathbf{r}]$$

is the mean, and

$$\mathbf{S} = E[(\mathbf{r} - \mu)(\mathbf{r} - \mu)^T]$$

is the expected value of the matrix, found by taking the expected values of its components [102]. Thus, the covariance matrix is equivalent to the crosscorrelation matrix of the users' spreading codes and noise, since the signal's mean is zero [172]. The effect of correlated noise is shown in Figure 6.1(b) for a two user PPB CDMA system with nonorthogonal spreading codes. Compared with Figure 6.1(a), it shows an elliptical distributed signal, whereas orthogonal codes result in a circular distribution about each mean. The exponential term (within the brackets) in (6.15) is known in pattern recognition literature as the *Mahalanobis* distance measure [102, 144, 173]. Figure 6.4 illustrates the impact of the two distance measures presented. It shows the decision boundaries for a two user DS-CDMA scenario in AWGN with $E_b/N_0 = 7$dB and 7 chip Gold codes. Clearly, the receiver performance of a PPB RBF receiver with Euclidean distance (ERBF) or Mahalanobis distance (MRBF) measure will be different, because of their different boundaries.

The next step is to replace the Euclidean distance measure used in (6.4) by the Mahalanobis distance measure in (6.15). Therefore, the new radial basis function has the form:

$$f(\mathbf{r}(k)) = \sum_{m=1}^{M} w_m \, \exp\left(-\frac{(\mathbf{r}(k) - \mathbf{c}_m)^T \, \mathbf{S}^{-1} \, (\mathbf{r}(k) - \mathbf{c}_m)}{2} \right), \tag{6.16}$$

where the centres $\mathbf{c}_m$ are the preprocessed CLB RBF centres, which can also be constructed according to equation (6.15). Vector $\mathbf{r}(k)$ is the signal fed into the PPB RBFN, which has a correlated noise component. Since this RBF receiver structure takes the correlated noise into account, it must perform as well as the CLB RBF, which implements the Bayesian function and hence is optimum for the non-dispersive AWGN scenario.

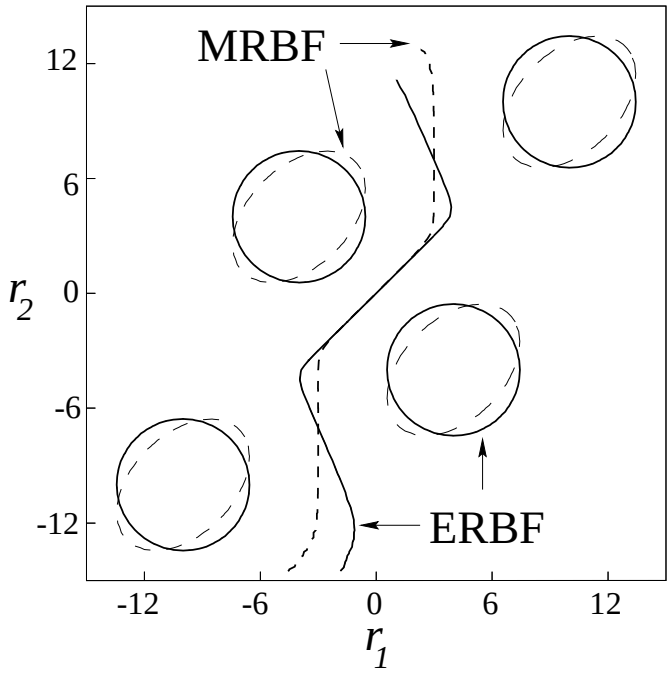


Figure 6.4: Two decision boundaries for two different PPB RBF structures for a two user CDMA system with 7 chip Gold codes and $E_b/N_0 = 7$dB. Shown are the decision boundaries obtained with Euclidean distance measure (ERBF) and Mahalanobis distance measure (MRBF).

6.3 Reduced PPB RBF receiver

The Mahalanobis based RBF also has 2^{3U} centres in a multipath scenario though with shorter centre vectors, which are of length U. However, this implies 2^{15} centres when five users are active. Clearly, this is a prohibitively large network for mobile telephony applications. In order to reduce the RBF complexity, the properties of the processed signals should be analysed. Then, it may be possible to find a technique to reduce the network size.

MRC preprocessing returns 2^{3U} centres with a lower dimensionality, since it can be assumed that $((N + L - 1) > U)$. An example for a two user multipath scenario is given next. Figure 6.5 shows all points (centres) for this scenario with $H_{ch} = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$, where seven chip Gold codes have been used. A three user scenario is depicted in Figure 4.9. More figures for different H_{ch} and spreading codes can be found in appendix C.

An empirical investigation of such patterns leads to the hypothesis the MRC preprocessing creates clusters of points, see also appendix C. It was found that there exists 2^U clusters, where 2^{2U} points form a cluster. Appendix C shows that different preprocessing filters form different

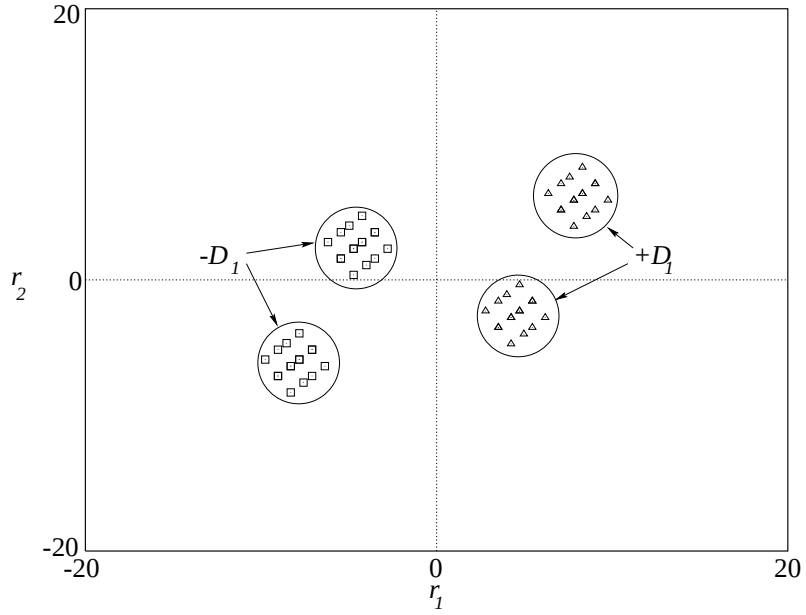


Figure 6.5: All possible points for a two user CDMA system with multipath and MRC preprocessing in their U-dimensional space. There are only 48 points instead of 64, because of the spreading sequence chosen and the channel impulse response (which is even), some points become redundant.

clusters. An explanation shall be given in conjunction with Figure 6.3(b). There, a vector representation is given for a single point based on the illustration presented in Figure 6.3(a). A vector $Centre$ is given by the chips unaffected by ISI. On top of this vector, the ISI affected vectors $Head$ and $Tail$ are added. Since vectors $Head$ and $Tail$ are dependent on the previous, current and next symbol, they are not fixed and can "rotate". Hence they can point in different directions. Each state results in a point, thus, four clusters appear in the given example of Figure 6.5. The shape of these clusters is dependent on the correlation among the spreading codes and the channel impulse response. Generally, if the codes are orthogonal the clusters tend to look circular, whereas highly correlated codes form very elliptical clusters.

6.3.1 Centre construction

In order to exploit the finding that 2^{2U} centres form one cluster and can be replaced by a single centre, its construction must be derived. For convenience, the centres of the reduced PPB RBF are named *super centres*, while centres for the normal PPB RBF are simply named centres. The super centres represent μ in equation (6.15). Thus the 2^U super centres can be derived from the 2^{3U} centres given in (6.10). Again, Figure 6.3(a) will help to explain the procedure. The 2^{3U}

centres are constructed from the 2^U previous, 2^U current and 2^U next symbol combinations. In order to find one super centre, all points must be averaged, which can be constructed from the different head and tail combinations (ISI causing sequences). In other words, take one $Centre$ (see Figure 6.3(b)) and find all associated $Head$ and $Tail$ sequences, add them up and divide the sum by the number of them. This is still a time consuming task, since it is based on the 2^{3U} centres. A much simpler derivation was found.

The previously stated procedure does the following. It averages the ISI component induced by the previous symbol and current symbol, respectively. Since this takes all possible combinations of them into account, while the current symbol is left unchanged, the ISI component will cancel itself. In other words, since the signal (spreading code) is assumed to be antipodal and the number of symbol combinations is even, the average of the signal (in "head" and "tail") is zero. Hence, what is left is the current symbol convolved with H_{ch}. Hence, the super centres can be found by preprocessing the CLB centres $\mathbf{p}^{AWGN}$ constructed according to the extended Gaussian method. The covariance matrix $\mathbf{S}$ is left unchanged, since the super centres do not change correlation properties between the spreading codes.

6.4 Simulation results

This section presents simulation results obtained from Monte-Carlo simulations. Different RBF structures are compared in terms of their BER performance against established receivers. Results for the memoryless channel shall prove the claim that the PPB RBF with Mahalanobis distance measure performs as well as the CLB RBF receiver. Then, simulation results conducted in a stationary multipath environment are presented. Due to the large complexity of some receiver structures only stationary channels are considered.

6.4.1 AWGN channel

A DS-CDMA system with U users and seven chip spreading codes is considered. The SNR was chosen to be $E_b/N_0 = 7$dB. Three RBF structures are compared against the known PPB MMSE receiver [174]. A CLB RBF receiver (CRBF) acts as the optimum performance bound since it is equivalent to a Bayesian structure. Two PPB RBF receivers are investigated. The PPB RBF with Euclidean distance measure (ERBF), and the PPB RBF with Mahalanobis distance measure (MRBF).

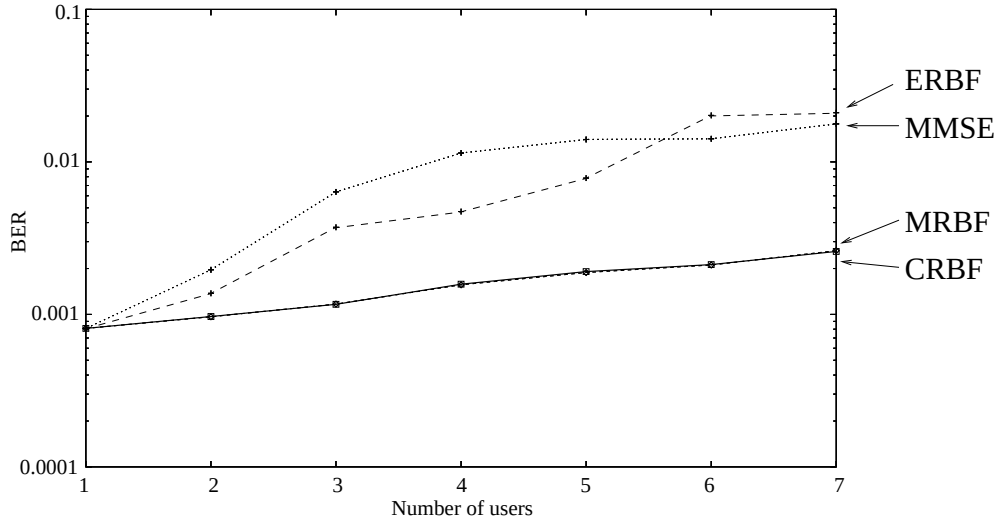


Figure 6.6: BER against the number of users for a CDMA scenario in AWGN with randomly generated 7 chip spreading codes and $E_b/N_0 = 7$dB in an AWGN channel.

Figure 6.6 shows results obtained from a set of randomly generated spreading sequences. It can be seen from Figure 6.6 that the MMSE is a long way from achieving optimum performance. Moreover, the ERBF structure suffers severely from not constructing the optimum decision boundary and performs even worse than the MMSE when the MAI is high. Further, the results show that the MRBF performs as well as the CRBF. Therefore, the Mahalanobis distance measure is the optimum distance measure for a PPB RBF receiver structure.

Figure 6.7 shows results obtained from a set of seven chip Gold spreading sequences. Again, the results lead to the same conclusion. MMSE and ERBF perform poorly, while MRBF and CRBF perform optimally.

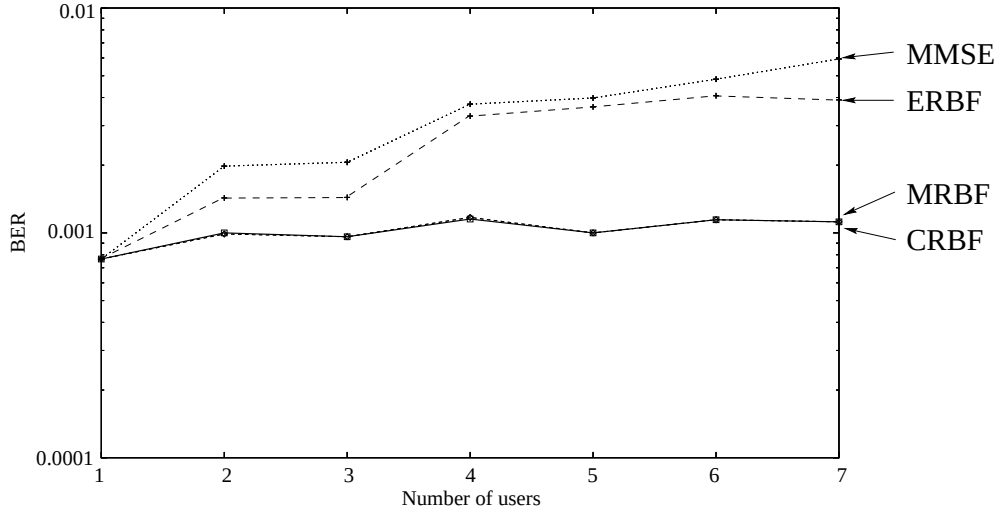


Figure 6.7: BER against the number of users for a CDMA scenario with 7 chip Gold spreading codes and $E_b/N_0 = 7$dB.

6.4.2 Multipath channel

A PPB DS-CDMA system with U users and short spreading codes is considered with preprocessing by a bank of MRC filters. Further, perfect knowledge of the channel impulse response is assumed. The number of active users is kept small since some receivers become too complex to simulate, e.g. if $U > 6$. Different RBF structures are compared against the known PPB MMSE receiver. The CLB RBF receiver (CRBF) with 2^{3U} centres of length $(N + L - 1)$ and the PPB RBF with Mahalanobis distance measure (MRBF) and 2^{3U} centres of length U are compared against the RBF with 2^U super centres (SRBF) and the RBF with Euclidean distance measure (ERBF) with 2^{3U} centres of length U.

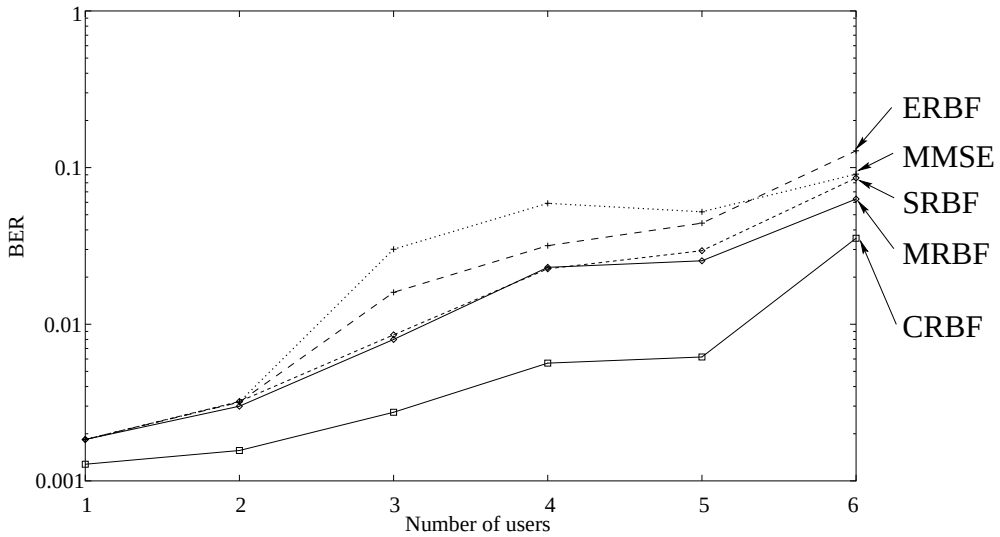


Figure 6.8: BER against the number of users for a CDMA scenario with randomly generated 7 chip spreading codes and $E_b/N_0 = 7$dB in a multipath channel with $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$.

Figure 6.8 presents the results obtained from a set of randomly generated spreading codes with seven chips and $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$. It shows that CRBF, MRBF and SRBF outperform the linear MMSE structure. Moreover, the MRBF and SRBF perform quite similarly over a large number of users, while the ERBF performs less well and is even outperformed by the MMSE at six users. The performance degradation at five users between the SRBF and the MRBF is due to the simplified decision boundary constructed by the SRBF. This becomes more severe as the dimension (U) is increased. However, it was found that the SRBF still outperforms the MMSE. Of interest is the superior performance of the CRBF receiver, which performs much better than the MRBF. Since the preprocessing is done by a bank of RAKEs (MRCs), the noise of the combined signal is no longer white because the output of

each finger is weighted by a channel coefficient. This induced additional noise distortion which is not perfectly described by the multivariate normal distribution (6.15). In order to circumvent this problem, preprocessing could be done by a bank of grouped MFs, where each group has L MFs [175]. However, this increases the signal's dimension used at the receiver structure to $(L \times U)$ and makes it more complicated to separate the received signals since all RAKE finger outputs are equally weighted, especially the weaker ones. Moreover, super centres must be computed differently, if they exist at all.

In Figure 6.9 results obtained from seven chip Gold codes and $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$ are presented. Figure 6.9 shows that all RBF structures (ERBF is not included) outperform the MMSE. The SRBF and MRBF receivers diverge at 5 users.

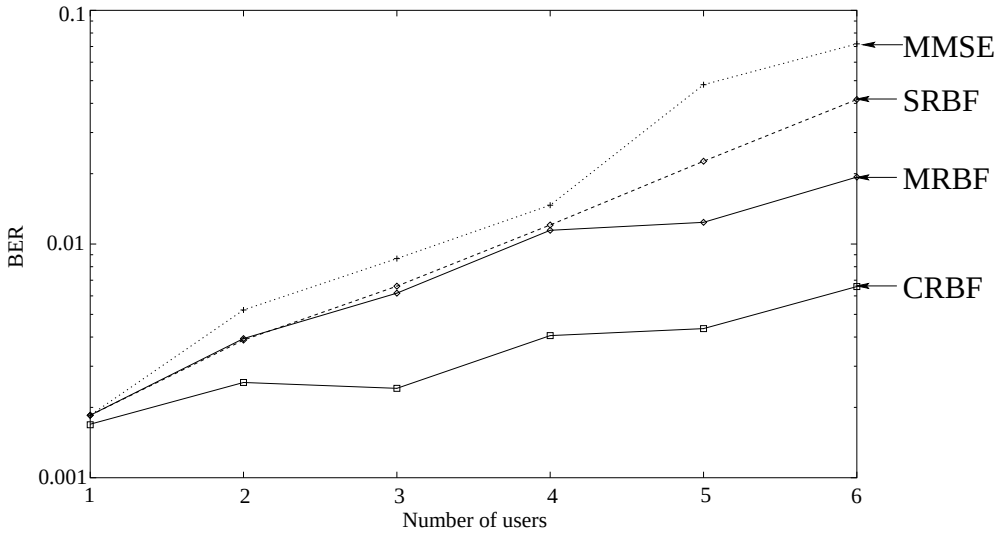


Figure 6.9: BER against the number of users for a CDMA scenario with 7 chip Gold codes and $E_b/N_0 = 7$dB in a multipath channel with $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$.

Figure 6.10 shows the results obtained for randomly generated spreading codes of length 16 with $E_b/N_0 = 7$dB and $H_{ch}(z) = 0.25 + 0.5z^{-1} + z^{-2}$. The results are similar to the ones presented in Figure 6.9 in the sense that the MMSE is not greatly outperformed by the RBF based receivers. For comparison the MRC performance is given, which becomes very poor as MAI increases. It is important to note that the SRBF is better than the MMSE with the spreading sequence of length 16.

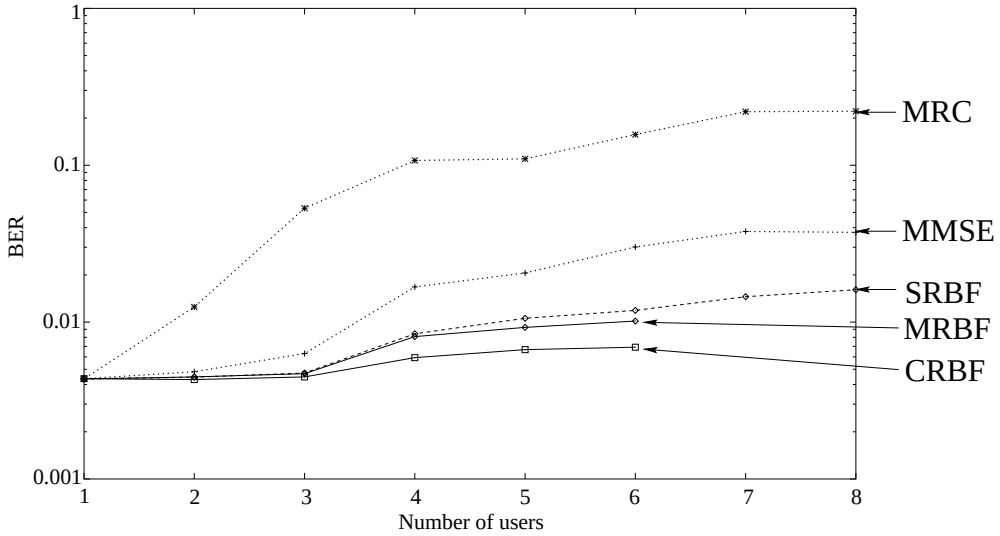
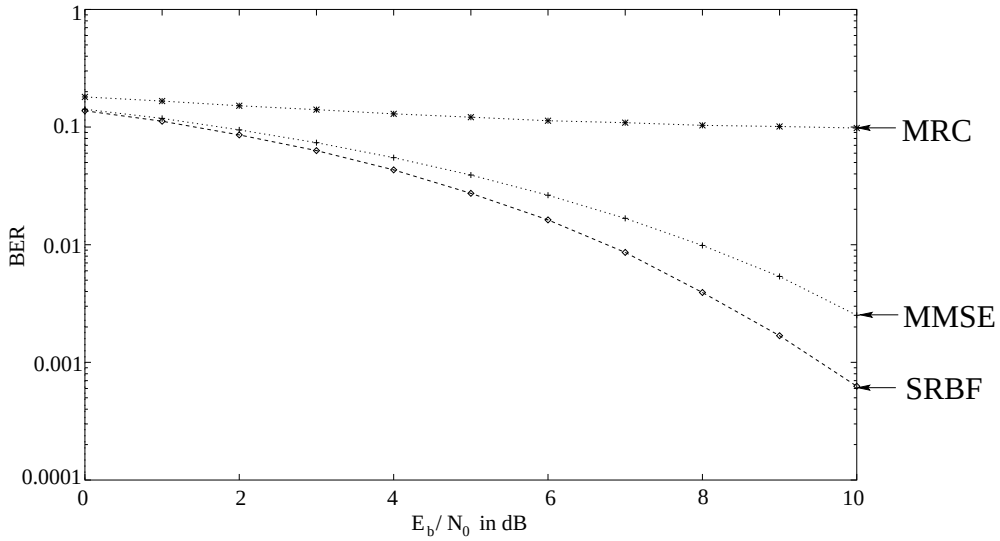
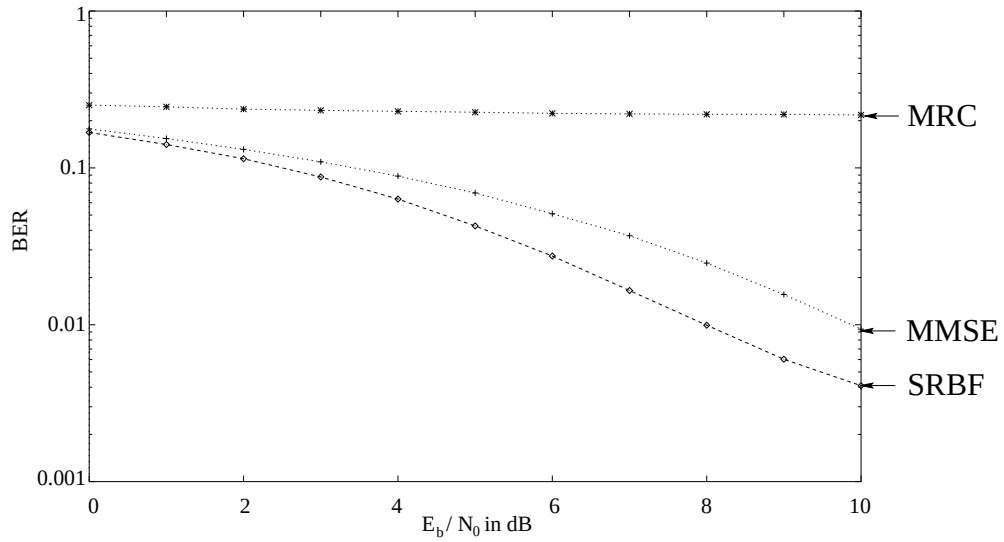


Figure 6.10: BER against the number of users for a CDMA scenario with randomly generated 16 chip spreading codes and $E_b/N_0 = 7$dB in a multipath channel with $H_{ch}(z) = 0.25 + 0.5z^{-1} + z^{-2}$.

Results for two different numbers of users over a SNR range between 0dB and 10dB are given in Figure 6.11. The performance for the MRBF is omitted due to its complexity. In both scenarios, 6.11(a) and 6.11(b), the MRC performs very poorly. Figure 6.11 shows that the performance gap between the SRBF and the MMSE increases, as the number of users increases. In the 4 user scenario, the SRBF has at $E_b/N_0 = 7$dB half the error ratio of the MMSE, and has a gain of around 1.5dB at a BER of 0.01 over the MMSE. For 8 active users, the SRBF has half the error ratio of the MMSE at 7dB and a gains 2dB at a BER of 0.01 over the MMSE.



(a) *For 4 users.*



(b) *For 8 users.*

Figure 6.11: BER against the SNR for a CDMA scenario with randomly generated 16 chip spreading codes and a multipath channel with $H_{ch}(z) = 0.25 + 0.5z^{-1} + z^{-2}$.

6.5 Discussion

This chapter presented the chip rate and symbol rate RBF receiver. The RBF construction for the CLB structure was introduced from which the PPB structure has been derived. Both structures have the same number of centres (2^U) in a memoryless channel which grows exponentially with the number of users. It has been shown that the Euclidean distance measure used for the CLB RBF is only under certain circumstances optimum for the PPB RBF. This is due to the correlated nature of the noise induced by the preprocessing stage. If nonorthogonal spreading sequences are used, then the Euclidean distance measure must be replaced by the Mahalanobis distance measure. Then, the PPB RBF and the CLB RBF perform the same in AWGN.

Applying the PPB RBF for multipath scenarios is of limited use since its computational complexity may exceeds the resources available at a mobile. A technique which introduces super centres has been proposed. This technique reduces the number of centres from 2^{3U} to 2^U. Since it rests upon the Mahalanobis distance measure, it also takes the correlation among the spreading codes into account. Although it needs a matrix inversion, its complexity is less since the matrix inversion has to be done only once for a set of spreading codes used. If the set of codes is known a prior then it might be possible to store the matrix inverse in a look up table. Monte-Carlo simulations showed little performance loss over a wide range of users for this RBF structure, if compared with the Mahalanobis based RBF which takes all 2^{3U} into account. However, the proposed technique of computing super centres is restricted to certain CDMA systems in order to be beneficial. These systems must use short spreading sequences, such as 16 chip codes, e.g. UMTS [3]. Because long codes, e.g. as used in IS-95, are less ISI affected (in the Head and Tail chips) hence the ISI free middle part (Centre) provides the receiver with sufficient information in order to detect the transmitted bit.

All multipath simulations exploit a bank of RAKEs based on the MRC, because MRC performs better than EGC [37]. Since each RAKE branch is weighted by a channel coefficient, the noise of the combined signal no longer has a white characteristic. The simulations showed that the CLB RBF performs much better than the PPB RBF. This is because the Mahalanobis distance measure rests upon the assumption that all signals have the same properties described by the multivariate normal distribution. In order to enhance performance, the output of each RAKE finger may be fed into the receiver structure. This increases the dimensionality of the preprocessed signal to (number of RAKE fingers $\times U$) which is still a moderate number since $N \gg U$.

Chapter 7
Alternative receivers

This chapter presents alternative receiver structures which are based on linear programming (LP).

An introduction presents a brief summary of preprocessing stages and looks at a simplified preprocessing technique. Ideas taken from pattern recognition and previously proposed equaliser schemes are applied to CDMA. Two different receiver structures based on LP are presented, both of which can be extended for use with other preprocessing schemes. Finally, this chapter ends with a discussion.

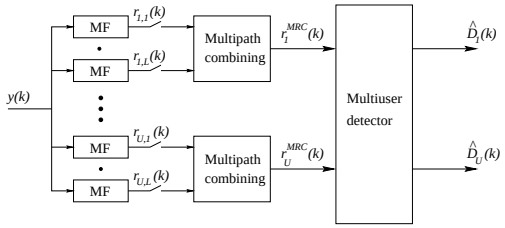
7.1 Introduction

It has been mentioned that the optimum (MUD) receiver has a prohibitive computational complexity and sub-optimum (MUD) receivers are the subject of current research. Most of these receivers can process either the MRC filter bank output (Figure 7.1(a)) or its MF version (Figure 7.1(b)), where the first structures do not eliminate the effect of MAI on channel estimation. Junatti and Glisic [175] concluded that MUD receivers, which process the MF bank output are often more desirable in practice. Reasons which back this statement were drawn from the results presented in section 6.4. There, the PPB RBF with MRC preprocessing did not perform as well as the CLB RBF.

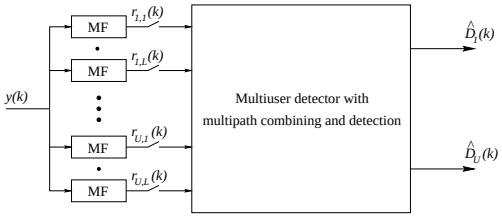
However, while investigating preprocessed sequences and their pattern structures, some interesting pattern features were found. From the patterns obtained from a bank of MFs (synchronised at the symbol rate) two CDMA receiver structures were developed. Both structures can be extended and applied to the preprocessing schemes in Figure 7.1.

7.1.1 Simplified preprocessing (SPP)

Although it has been stated that a simple bank of MFs will perform poorly, it shall be used in the first instance to create nonlinearly separable scenarios [173]. Because the preprocessing stage is



(a) *Channel is taken into account in the preprocessing stage.*



(b) *Channel is taken into account after the MUD.*

Figure 7.1: Two different preprocessing concepts for MUD receiver structures.

synchronised with the symbol rate, its output is strongest only if the first multipath component (h_1) is the biggest. The *simplified preprocessing* (SPP) stage is shown in Figure 7.2.

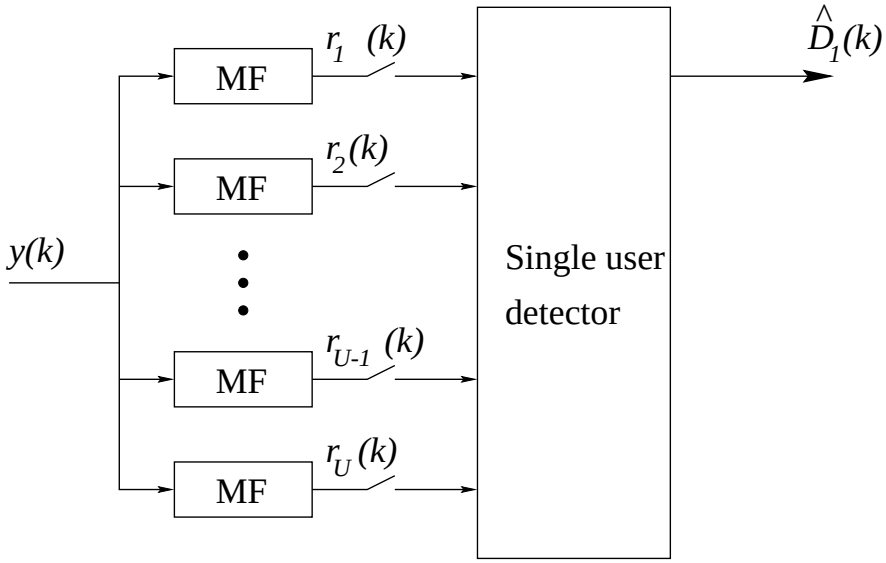


Figure 7.2: A simplified preprocessing technique where the preprocessing stage consists of MFs which do not take multipath delay into account.

It was also found that Walsh sequences have a strong tendency to form nonlinearly separable scenarios. This is basically due to the fact that Walsh spreading codes have highly coloured spectral characteristics [55, 176]. This problem is circumvented in communication systems, e.g. IS-95 [16], by employing randomisation. In nonlinearly separable scenarios, the performance of linear receiver structures is dominated by the number of points which lie on the wrong side of the hyperplane. Only nonlinear receiver structures can resolve such problems, resulting in zero classification errors for the noise free case.

7.1.2 System description

The output $\mathbf{r}(k)$ in Figure 7.2 is given from the input sequences $\mathbf{y}(k)$. This simplified preprocessing scheme takes only half of the ISI into account, the ISI induced from the previous symbol. Thus the number of possible received sequences (points) is $M = 2^{2U}$ for antipodal signalling, and the matrix containing all sequences for the simplified preprocessing scheme is defined according to 2.10 as:

$$\mathbf{P}^{SPP} = \mathbf{C}^{SPP}\mathbf{H}^T, \tag{7.1}$$

where the $(M \times 2N)$ code matrix $\mathbf{C}^{SPP}$ contains all combinations of the spreading codes over two symbols, and the $(N \times 2N)$ matrix $\mathbf{H}$ represents the channel characteristics. So for a $L = 3$ tap channel and $N = 7$, $\mathbf{H}$ is defined as:

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 \end{bmatrix}. \tag{7.2}$$

After all possible received sequences are defined (in $\mathbf{P}^{SPP}$), the signals are derived, which are fed into the receiver structure. All possible preprocessed signals are given in a $(M \times U)$ matrix $\mathbf{R}^{SPP}$, derived from:

$$\begin{aligned} \mathbf{R}^{SPP} &= \mathbf{P}^{SPP}\,\mathbf{C}^T \qquad (7.3) \\ &= \mathbf{C}^{SPP}\,\mathbf{H}^T\,\mathbf{C}^T = [(\mathbf{B}^{SPP})^T * \mathbf{C}^T]^T\,\mathbf{H}^T\,\mathbf{C}^T \\ &= \left[\begin{bmatrix}\mathbf{B1}\\ \mathbf{B2}\end{bmatrix} * \begin{bmatrix}\mathbf{C1}\\ \mathbf{C2}\end{bmatrix}\right]^T \mathbf{H}^T\mathbf{C}^T \\ &= \left[\mathbf{B1}\begin{bmatrix}\mathbf{c}_1^T\\ \vdots\\ \mathbf{c}_U^T\end{bmatrix}\ \mathbf{B2}\begin{bmatrix}\mathbf{c}_1^T\\ \vdots\\ \mathbf{c}_U^T\end{bmatrix}\right]\mathbf{H}^T\,\mathbf{C}^T, \qquad (7.4) \end{aligned}$$

where the $(M \times 2U)$ combination matrix $\mathbf{B}^{SPP}$ contains all combinations for two adjacent symbols, and $\mathbf{C}$ is a $(U \times N)$ partitioned code matrix.

7.1.3 Linearly separable scenario

Assume there is a $(M \times U)$ matrix $\mathbf{R}$ of real numbers and each row defines a single pattern. This pattern matrix consists of two sets of patterns, say subset $\{\mathbf{R}^+\}$ and $\{\mathbf{R}^-\}$, which contain the patterns belonging to $+D_d$ and $-D_d$ respectively. LP finds a plane in the U-dimensional Euclidean space $\mathbb{E}^U$, such that each subset lies on one side of a plane (hyperplane) if and only if $\{\mathbf{R}^+\}$ and $\{\mathbf{R}^-\}$ are linearly separable. The plane will then satisfy the equations:

$$\mathbf{R}^+\mathbf{w} - e\delta \geq 0 \tag{7.5}$$

$$\mathbf{R}^-\mathbf{w} - l\delta < 0, \tag{7.6}$$

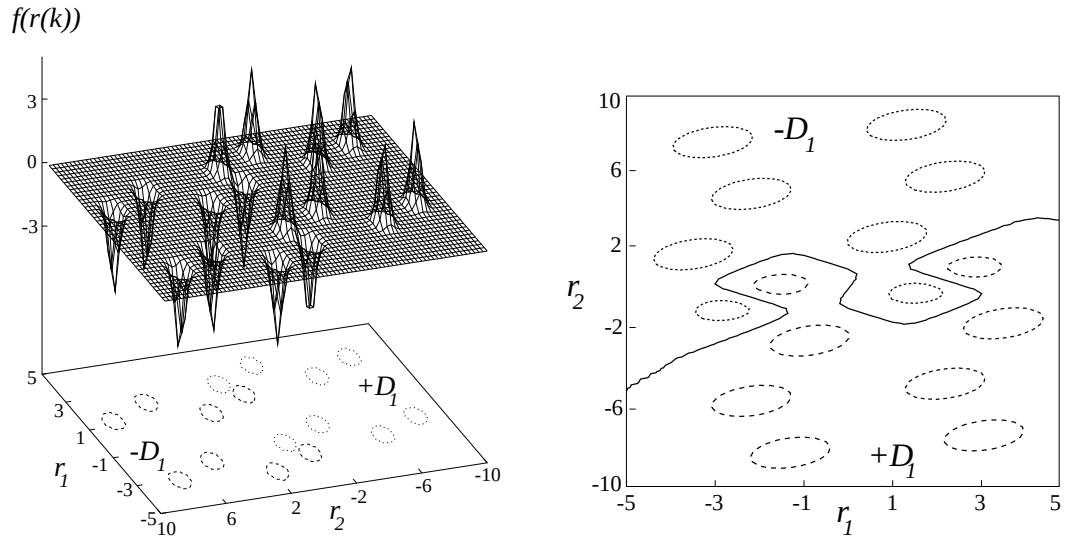
whereas the plane is determined by:

$$\mathbf{x}^T\mathbf{w} - \delta = 0, \tag{7.7}$$

where $\mathbf{x}$ is a U-dimensional vector representing a point in $\mathbb{E}^U$, $\mathbf{w}$ is a U-dimensional vector of constants and δ is a scalar constant (threshold) such that (7.5) and (7.6) hold. There, e and l are $M/2$-dimensional column vector of ones. A definition and proof for linear separability is given in [138] which is examined briefly. Two sets of patterns $\{\mathbf{R}^+\}$ and $\{\mathbf{R}^-\}$ are linearly separable, if and only if there exists a U-dimensional column vector of constants $\mathbf{w}$ and constant scalars a and b such that:

$$\begin{aligned} \mathbf{R}^+\mathbf{w} - ea &\geq 0 \\ -\mathbf{R}^-\mathbf{w} + lb &\geq 0 \\ a - b &> 0 \\ \mathbf{f} \geq \mathbf{w} &\geq -\mathbf{f}, \end{aligned}$$

where $\mathbf{f}$ is a vector of ones. Since LP finds a separating plane based on the L_1 distance measure criterion, it could be used to determine the filter weights of the decorrelating receiver. If the subsets are not linearly separable, then LP fails to provide a single plane and other techniques must be used.



(a) *The decision surface for high SNR.* **(b)** *The optimum nonlinear decision boundary.*

Figure 7.3: A two user CDMA nonlinearly separable scenario obtained from the simplified preprocessing scheme, with multipath channel $H_{ch}(z) = 0.25 + 0.5z^{-1} + z^{-2}$.

7.1.4 Nonlinearly separable scenario

As a consequence of the improper treatment of the ISI due to the simple preprocessing scheme used, nonlinearly separable scenarios are possible. However, it must be acknowledged that these scenarios are rare. Nevertheless, it has been shown that non-minimum phase channels can create such hard to resolve and therefore demanding equalisation scenarios [90, 99].

A $U = 2$ user DS-CDMA system is assumed. The channel impulse response is $H_{ch}(z) = 0.25 + 0.5z^{-1} + z^{-2}$ and the spreading codes were randomly generated with length $N = 7$. The scenario's points (patterns) are presented in Figure 7.3. This example shows that it is not possible to linearly separate the two groups of points. This also becomes evident from the optimum separation boundary depicted in Figure 7.3(b). Thus, simple linear filters must be replaced by more complex structures, such as ML based receivers. Another promising structure in order to classify patterns is based on LP.

7.2 A linear programming based technique

The LP algorithm rests upon Dantzig's simplex algorithm [177] and is a common technique used in game theory [178] and operations research for optimisation [96]. It also can be used to construct convex hulls from a set of points in a d-dimensional space [146, 147].

Basically LP does the following [1]. Find the column vector $\mathbf{x}$, which minimises (or maximises) a linear function

$$\mathbf{f} = \mathbf{c}^T\mathbf{x},$$

subject to some constraints, such as

$$\mathbf{A}\mathbf{x} = \mathbf{b} \qquad \text{and} \qquad \mathbf{x} \geq 0.$$

Figure 7.4 illustrates such an example for two independent variables in a 2-dimensional space. The region which contains all possible solutions is called the feasible region. The feasible region (grey) is defined by the constraints, depicted as dashed lines. Some feasible vectors (solutions) are depicted as bullets, and one of them is the optimum solution, the optimal feasible vector.

In order to find the optimum solution, the LP algorithm has only to look at the vertices of the feasible region. Thus, only the points which are part of the convex hull are of interest, since the optimum solution is a vertex [96]. The example given in Figure 7.4 shows several parallel lines across the feasible region, where each line corresponds to a constant value of the solution f, e.g. $f_1, f_2, \ldots, f_4$. Hence, LP starts at one vertex and computes the corresponding solution. Then LP takes one of the adjacent vertices and proceeds in the same manner, until the optimum solution is found, defined as a maximisation or minimisation problem.

Therefore, only the vertices, which are part of the convex hull are of interest to determine the optimum solution. However, LP is not always efficient, since the computational time grows exponentially with the number of vertices. Thus, other algorithms are known, which can construct the convex hull [146, 157] faster. Nevertheless, LP is still popular and has been applied over the last forty years to many applications, and integer and nonlinear programming techniques are also available [179]. LP also has been used for training and constructing NNs [180–182].

Even nonlinearly separable problems can be solved with the LP algorithm, if it is combined with

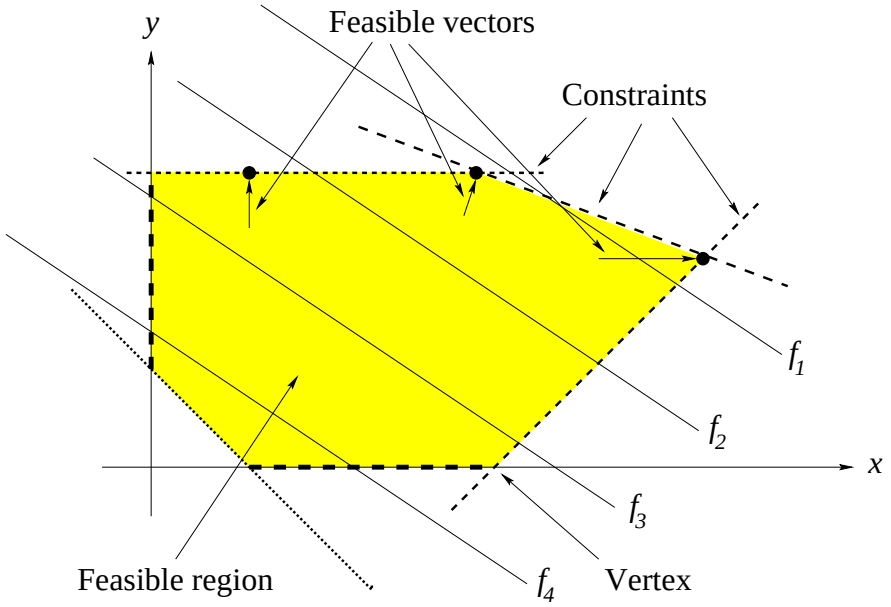


Figure 7.4: The basic concept of linear programming. A feasible region (shaded) containing the optimum solution at one vertex is bounded by the constraints, depicted as dashed lines. LP finds the optimum solution by computing $\mathbf{f}$ for a vertex and its adjacent vertices, hence it will converge to the optimum vertex.

the *multisurface method* [183, 184]. With the multisurface method it is possible to construct a piecewise linear separation boundary [185–187] or slabs [188]. The vertices in Figure 7.4 are given by the points given for $+D_1$ or $-D_1$, which form two overlapping hulls.

7.2.1 The SLAB algorithm

This algorithm constructs slabs in a multi dimensional space which can separate sets of points. Each slab consists of (at least) two planes or hyperplanes, which bound it. A slab with zero thickness is a single hyperplane. It is well known that such hyperplanes can be constructed with perceptrons, since a hyperplane is defined by a linear function [93, 132]. A (neural) network which can incorporate this slab structure is the common MLP network, where the neurons are McCulloch-Pitts units [93]. The unit's activation function is a step function defined as:

$$f(\mathbf{r}) = \begin{cases} +1 & \text{if } \mathbf{r}^T\mathbf{w} - w_0 \geq 0, \\ -1 & \text{if } \mathbf{r}^T\mathbf{w} - w_0 < 0, \end{cases} \tag{7.8}$$

where $\mathbf{r}$ is the received vector, $\mathbf{w}$ is the weight vector of a unit and w_0 is its threshold. A possible MLP structure with K units is presented in Figure 7.5.

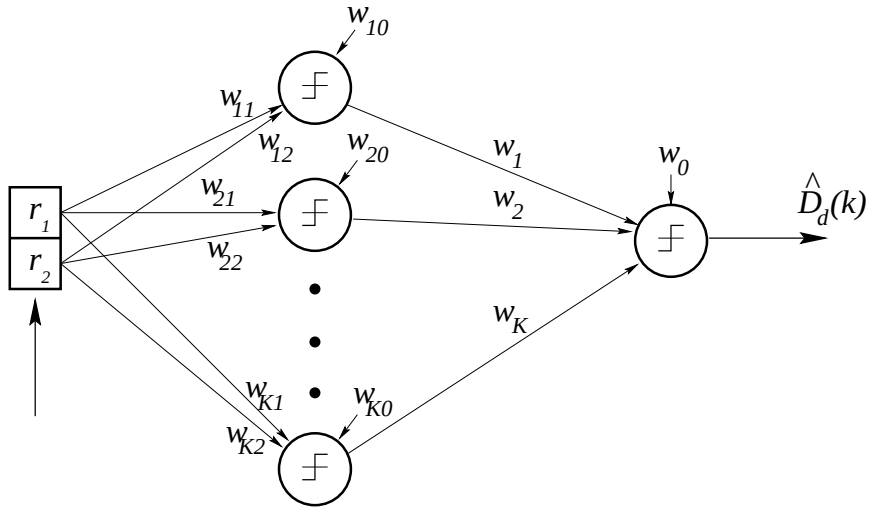


Figure 7.5: A MLP network, which can construct hyperplanes. Each unit in the hidden layer constructs one plane and the output unit combines their outputs and makes the final decision.

It is assumed that two sets of points in $\mathbb{R}^U$ are available. Both sets $\{\mathbf{R}^+\}$ and $\{\mathbf{R}^-\}$ correspond to the desired user bit sign $D_d = +1$ and $D_d = -1$ respectively. The SLAB algorithm should construct a classifier that separates points in $\{\mathbf{R}^+\}$ from those in $\{\mathbf{R}^-\}$ and can be described by:

Start with $\{\mathbf{R}_0^+\} = \{\mathbf{R}^+\}$ and $\{\mathbf{R}_0^-\} = \{\mathbf{R}^-\}$, and iterate the steps $i \geq 1$:

1. Construct a slab:

   $$S_i = \{\mathbf{r} \in \mathbb{R}^U : a_i \leq \mathbf{r}^T \mathbf{w}^i \leq b_i\}$$

   where $\{a_i, b_i\} \in \mathbb{R}$, $a_i \leq b_i$, $\mathbf{w}^i \in \mathbb{R}^U$, $\mathbf{w}^i \neq 0$, such that:

   (a) $\mathbf{R}_{i-1}^+ \subseteq \{\mathbf{r} \in \mathbb{R}^U : a_i \leq \mathbf{r}^T \mathbf{w}^i\}$, $\mathbf{R}_{i-1}^- \subseteq \{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T \mathbf{w}^i \leq b_i\}$ and

   (b) the width of the slab is minimal.

2. If

   $$\mathbf{R}_{i-1}^+ \cap S_i = \{\} \text{ and } \mathbf{R}_{i-1}^- \cap S_i = \{\}^1 \text{ then stop,}$$

   else

   $$\mathbf{R}_i^+ = \mathbf{R}_{i-1}^+ \cap S_i, \quad \mathbf{R}_i^- = \mathbf{R}_{i-1}^- \cap S_i \text{ and}$$

   goto 1) and increment i.

[1] This means the set is empty.

LP can find the slabs by minimising the width of a slab, which means:

- If $\{\mathbf{R}_{i-1}^{+}\}$ and $\{\mathbf{R}_{i-1}^{-}\}$ are linearly separable, then S_i must be a hyperplane such that $\{\mathbf{R}_{i-1}^{+} \cup \mathbf{R}_{i-1}^{-}\} \cap S_i = \{\}$,
- else minimise the number of elements in $\{\{\mathbf{R}_{i-1}^{+} \cup \mathbf{R}_{i-1}^{-}\} \cap S_i\}$.

For the case of a linearly separable scenario, the SLAB algorithm yields a single hyperplane, otherwise, there are $S_1, S_2, \ldots, S_{j+1}$ slabs ($j \geq 1$). If $1 \leq i \leq j$, each S_i specifies a pair of units in the hidden layer. Each pair has weights $[w_1^i \; w_2^i \ldots w_U^i]$ with threshold $w_0^i = a_i$, and $[-w_1^i \; -w_2^i \ldots -w_U^i]$ with threshold $w_0^i = -b_i$.

The output layer consists of a single unit, see Figure 7.5. The weights for the two hidden units corresponding to S_i are $1/2^i$ and $-1/2^i$. Both units combine to contribute $1/2^i$ to the output unit from the points in $\{\mathbf{R}_{i-1}^{+} \setminus S_i\}$, and $-1/2^i$ to the output unit from the points in $\{\mathbf{R}_{i-1}^{-} \setminus S_i\}$ and zero from the points in $\{\{\mathbf{R}_{i-1}^{+} \cup \mathbf{R}_{i-1}^{-}\} \cap S_i\}$, $1 \leq i \leq j$. The last unit in the hidden layer is a separating hyperplane (slab with zero thickness):

$$S_{j+1} = \{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T \mathbf{w}^{j+1} = a_{j+1} = b_{j+1}\}$$

where $\mathbf{R}_j^{+} \subseteq \{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T \mathbf{w}^{j+1} > b_{j+1}\}$ and $\mathbf{R}_j^{-} \subseteq \{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T \mathbf{w}^{j+1} < b_{j+1}\}$.
The weights of this unit are $[w_1^{j+1} \; w_2^{j+1} \ldots w_U^{j+1}]$ with threshold $w_o = b_{j+1}$, and the corresponding weight for the output unit is $1/2^{j+1}$. The threshold w_0 of the output unit is $1/2^{j+2}$. The weights assignment for the output layer is appropriate since the weights of the output unit are based on a geometric series. Therefore, since $j \geq 1$ and $1 \leq i \leq j$, points in $\{\mathbf{R}_{i-1}^{+} \setminus S_i\}$ and $\{\mathbf{R}_{i-1}^{-} \setminus S_i\}$ are classified by a pair of units corresponding to S_i and will not be misclassified by the addition of further hidden layer units.

The number of hidden units is finite because it can be ensured that at least one point in $\{\mathbf{R}_{i-1}^{+} \cup \mathbf{R}_{i-1}^{-}\}$ is excluded from the slab S_i at each iteration step. If the number of hidden units is very large, then multiple threshold elements can be used in order to reduce network complexity [189].

7.2.1.1 The SLAB algorithm in steps

First, the SLAB tries to separate $\{\mathbf{R}_{i-1}^{+}\}$ from $\{\mathbf{R}_{i-1}^{-}\}$ with a single hyperplane

$$S_i = \{\mathbf{r} \in \mathbb{R}^U : a_i = \mathbf{r}^T \mathbf{w}^i = b_i\}$$

such that

$$\mathbf{R}_{i-1}^{+} = \{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T \mathbf{w}^i > a_i\} \qquad \text{and} \qquad \mathbf{R}_{i-1}^{-} = \{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T \mathbf{w}^i < a_i\}$$

The LP constraints are linear inequalities, but not strict inequalities. A separating slab of non-zero thickness is given by

$$S_i' = \{\mathbf{r} \in \mathbb{R}^U : c_i \leq \mathbf{r}^T \mathbf{w}^i \leq d_i\}$$

where $\{c_i, d_i\} \in \mathbb{R}$, $(c_i + 1) = d_i$, $\mathbf{w}^i \in \mathbb{R}^U$ and $\mathbf{w}^i \neq 0$, such that:

$$\mathbf{R}_{i-1}^{+} \subseteq \{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T \mathbf{w}^i \geq d_i\} \qquad \text{and} \qquad \mathbf{R}_{i-1}^{-} \subseteq \{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T \mathbf{w}^i \leq c_i\}$$

The constraints are

- $\mathbf{r}^T \mathbf{w}^i \geq d_i$ if $\mathbf{r} \in \mathbf{R}_{i-1}^{+}$
- $\mathbf{r}^T \mathbf{w}^i \leq c_i$ if $\mathbf{r} \in \mathbf{R}_{i-1}^{-}$
- $0 \leq (d_i - c_i) \leq 1$

and the function to minimise is $(c_i - d_i)$. Start the algorithm by setting $\mathbf{w}^i = 0$ and $c_i = d_i = 0$, and if the two sets are linearly separable (by LP), then a separating slab S_i' can be found such that $(c_i + 1) = d_i$ and $\mathbf{w}^i \neq 0$. Then set

$$S_i = \{\mathbf{r} \in \mathbb{R}^U : (c_i + d_i)/2 = \mathbf{r}^T \mathbf{w}^i\}$$

and all points are classified. Alternatively, if the two sets are nonlinearly separable and the first trial fails, then find a slab

$$S_i' = \{\mathbf{r} \in \mathbb{R}^U : a_i' \leq \mathbf{r}^T \mathbf{w}^i \leq b_i'\}$$

where $\{a_i^{'}, b_i^{'}\} \in \mathbb{R}$, $(a_i^{'} + 1) = b_i^{'}$, $\mathbf{w}^i \in \mathbb{R}^U$ and $\mathbf{w}^i \neq 0$, and

$$\mathbf{R}_{i-1}^{+} \subseteq \{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T\mathbf{w}^i \geq a_i^{'}\} \qquad \text{and} \qquad \mathbf{R}_{i-1}^{-} \subseteq \{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T\mathbf{w}^i \leq b_i^{'}\}$$

containing as few elements as possible in set $\{\mathbf{R}_{i-1}^{+} \cup \mathbf{R}_{i-1}^{-}\}$. The constraints are

- $\mathbf{r}^T\mathbf{w}^i \geq a_i^{'}$ if $\mathbf{r} \in \mathbf{R}_{i-1}^{+}$
- $\mathbf{r}^T\mathbf{w}^i \leq b_i^{'}$ if $\mathbf{r} \in \mathbf{R}_{i-1}^{-}$
- $b_i^{'} - a_i^{'} = 1$

and the function to minimise is

$$-\sum_{\mathbf{r} \in \mathbf{R}_{i-1}^{+}} \mathbf{r}^T\mathbf{w}^i + (|\mathbf{R}_{i-1}^{+}| \times b_i^{'}) + \sum_{\mathbf{r} \in \mathbf{R}_{i-1}^{-}} \mathbf{r}^T\mathbf{w}^i - (|\mathbf{R}_{i-1}^{-}| \times a_i^{'})$$

where $|\mathbf{R}_{i-1}|$ represents the number of points in set $\{\mathbf{R}_{i-1}\}$. Start with $\mathbf{w} = 0$ and $a_i^{'} = b_i^{'} = 0$, which will yield a slab

$$S_i^{'} = \{\mathbf{r} \in \mathbb{R}^U : a_i^{'} \leq \mathbf{r}^T\mathbf{w}^i \leq b_i^{'}\}$$

with an upper bound (hyperplane) $\{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T\mathbf{w}^i = b_i^{'}\}$ containing elements in $\{\mathbf{R}_{i-1}^{-}\}$ and a lower bound $\{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T\mathbf{w}^i = a_i^{'}\}$ which contains elements $\{\mathbf{R}_{i-1}^{+}\}$. Since the data is noise corrupted, the hyperplanes (bounds) must be adjusted (to maximise the distance between the points close to the hyperplanes). Thus, if $\mathbf{R}_{i-1}^{+} \setminus S_i \neq \{\}$, then let

$$b_i^{''} = \min_{\mathbf{r} \in \{\mathbf{R}_{i-1}^{+} \setminus S_i\}} (\mathbf{r}^T\mathbf{w}^i)$$

and $b_i = (b_i^{'} + b_i^{''})/2$, else $b_i = (b_i^{'} + 1/2)$. If $\mathbf{R}_{i-1}^{-} \setminus S_i \neq \{\}$, then let

$$a_i^{''} = \max_{\mathbf{r} \in \{\mathbf{R}_{i-1}^{-} \setminus S_i\}} (\mathbf{r}^T\mathbf{w}^i)$$

and $a_i = (a_i^{'} + a_i^{''})/2$, else $a_i = (a_i^{'} - 1/2)$.
Finally, specify slab

$$S_i = \{\mathbf{r} \in \mathbb{R}^U : a_i \leq \mathbf{r}^T\mathbf{w}^i \leq b_i\}$$

Sometimes, one of a pair of hidden layer units may be eliminated. If $\{\mathbf{R}_{i-1}^{-} \subset S_i\}$, then replace S_i by a single hyperplane $\{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T\mathbf{w}^i = b_i\}$ which corresponds to a single hidden unit with weights $\mathbf{w}^i$ and threshold $w_0^i = b_i$. The corresponding weight for the output unit is $1/2^i$. If $\{\mathbf{R}_{i-1}^{+} \subset S_i\}$, then replace S_i by a single hyperplane $\{\mathbf{r} \in \mathbb{R}^U : \mathbf{r}^T\mathbf{w}^i = a_i\}$ corresponding to a hidden unit with weights $-\mathbf{w}^i$ and threshold $w_0^i = -a_i$, with a corresponding output unit weight of $-1/2^i$.

7.2.1.2 The SLAB algorithm by example

It shall be assumed that there are two sets in $\mathbb{R}^2$. Each set consists of points, depicted as filled and hollow circles in Figure 7.6. Figure 7.6(a) shows the scenario. The aim is to classify the sets with slabs or hyperplanes, where each slab has minimum width. LP starts with a slab of zero thickness at the origin which is actually a hyperplane, see Figure 7.6(b). Obviously, the two sets are not linearly separable. Thus, LP increases the width of the slab, and rotates the hyperplanes. The width of this first slab is increased until LP can separate the two sets (outside the slab) from each other, which is the case in Figure 7.6(c). The dashed lines in Figure 7.6(c) mark where LP stops. Then, it can be calculated which points are closest to either side of the hyperplane. This enables the algorithm to place the hyperplane perpendicularly between them, which is the optimum place in terms of classifying noisy data. Thus, two hyperplanes are constructed, which are depicted as solid lines $\mathcal{H}_{1,1}, \mathcal{H}_{1,2}$ in Figure 7.6(c). The next step is to remove all points which can be classified (depicted with a cross in Figure 7.6); there are two points in the given example.

Another iteration shall classify the remaining points. Again, LP starts as depicted in Figure 7.6(b). This time, LP stops after it exceeds the radius of an imaginary circle on which lie the four points in the centre. Once more, the distance between the closest points on each side of the hyperplane is measured and halved in order to place the hyperplanes $\mathcal{H}_{2,1}, \mathcal{H}_{2,2}$ optimally. The slope of the hyperplane is given by the LP algorithm, but the threshold (a_i, b_i) must be adjusted in order to place the hyperplane between the points which a slab S_i wants to separate. Then, the classified points are removed from the set of points. Again, starting at the origin (Figure 7.6(b)), LP immediately finds that the remaining four points are linearly separable by a hyperplane $\mathcal{H}_3$, through the origin. Thus, the threshold is zero since it is not a slab, see Figure 7.6(e). The final result consists of five hyperplanes, see Figure 7.6(f). Clearly, both sets are separated from each other. However, in order to show that the algorithm works with

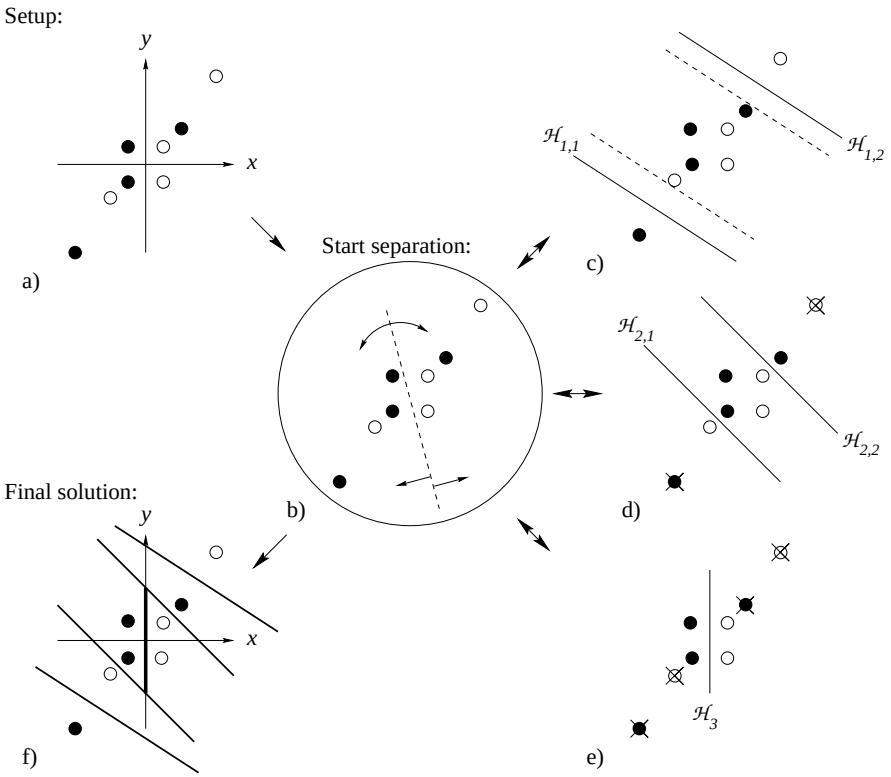


Figure 7.6: Steps made by the SLAB algorithm in order to separate a nonlinearly separable scenario with hyperplanes.

real data, an example is presented. This example is the one whose optimum nonlinear boundary was presented in Figure 7.3(b).

7.2.1.3 The SLAB algorithm in DS-CDMA

A $U = 2$ user scenario is given, where the points were obtained from the simplified preprocessing technique. Randomly generated spreading codes with length $N = 7$ were used, and the channel impulse response was $H_{ch}(z) = 0.25 + 0.5z^{-1} + z^{-2}$. The optimum classification is given by the boundary depicted in Figure 7.3(b). The SLAB algorithm constructs three hyperplanes and an MLP can classify the points correctly in the noise free case, see Figure 7.7.

As Figure 7.7 shows, the distance between hyperplane $\mathcal{H}_3$ and the points between $\mathcal{H}_1$ and $\mathcal{H}_2$ is small. Hence, the performance in a noisy environment will be degraded since the distance is short and so misclassifications will occur. Because the scenario is symmetric some misclassifications will cancel each other. Thus, instead of a residual BER corresponding to 4 points out

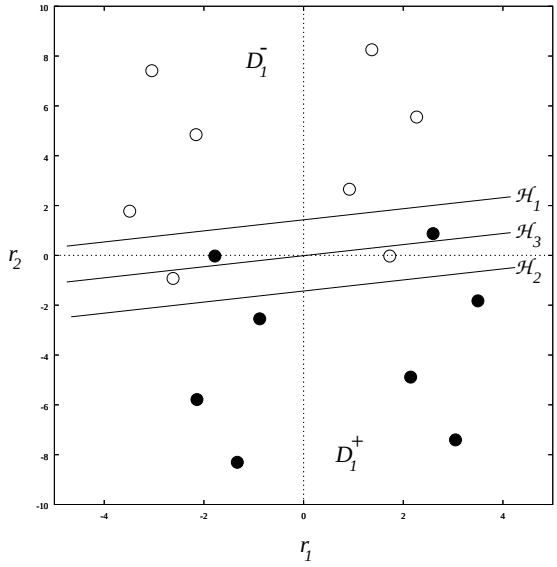


Figure 7.7: A nonlinearly separable scenario for two users and its three separating hyperplanes.

of 16 equals $4 \times 1/16 = 1/4$, the residual BER converges to $4 \times 0.5 \times 1/16 = 1/8$ as the SNR increases. However, this leads to another receiver structure which combines the simplicity of the SLAB based receiver and the excellent classification capabilities of the RBF network.

7.3 Hybrid receiver

In order to improve the receiver performance, a combined SLAB and RBF receiver is presented, giving a hybrid structure. The hybrid receiver exploits the knowledge given from the position of a slab, i.e. the region between two hyperplanes, which causes most classification errors. Therefore, it runs the SLAB algorithm first and finds all hyperplanes. Then, the hybrid receiver determines the two hyperplanes, between which most misclassifications may occur. The simplest criterion is the width of a slab, thus, all points within the narrowest slab are taken to become the RBF centres of the hybrid structure. Next, it constructs a RBF network, which has these points within this slab as centres. A major drawback with RBF networks is its complexity, when it has many centres. This might also be the case for this proposed hybrid structure. However, since it is difficult to generate nonlinearly separable scenarios in CDMA, no generally valid complexity figures are available. It may be assumed that the number of points will be between 25% and 50% of the total number of points, since the region where the two sets overlap may be considered to be small.

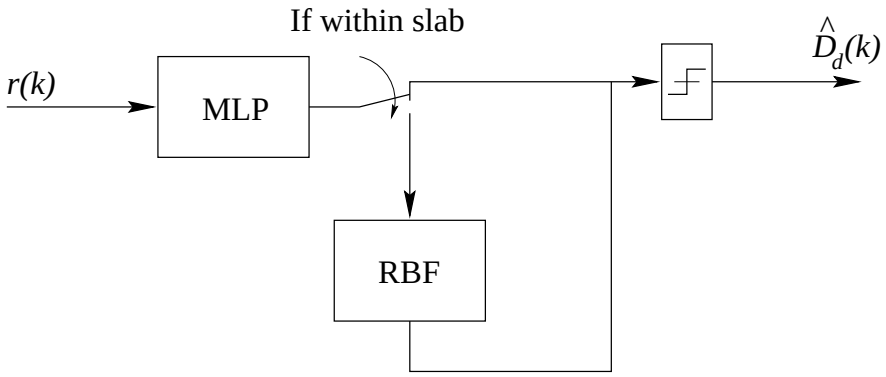


Figure 7.8: The hybrid receiver.

The hybrid receiver implementation is presented in Figure 7.8. The received signal $\mathbf{r}(k)$ is processed by the MLP. If $\mathbf{r}(k)$ lies between the hyperplanes, then $\mathbf{r}(k)$ is processed by the RBF, otherwise the MLP shall determine $\widehat{D}_d(k)$.

7.4 Simulation results

The performance has been obtained from Monte-Carlo simulations for different receiver structures. The channel impulse response is $H_{ch}(z) = 0.25+0.5z^{-1}+z^{-2}$, which is a non-minimum phase channel. Different scenarios are presented for two preprocessing schemes. Throughout this work, an LP algorithm has been used in the SLAB algorithm from a commercially available package. The `nage04.h` from `libnagc` is a standard NAG [2] library and the LP function embedded is `e04mfc`.

The first results were obtained with the SPP technique of subsection 7.1.1 and show that the LP based techniques perform well. Four receivers exploit the SPP preprocessed signals, whereas the fifth is the chip level based MRC (RAKE). The other four receivers are an MMSE (where the filter weights were obtained with the LMS algorithm [48]), a Mahalanobis based RBF receiver (MRBF), a slab based receiver (LP) and the hybrid receiver (Hybrid). There are no results given for a simple MF receiver since its performance is very poor. Then results are presented which were obtained with the PPB based technique, introduced in chapter 6. Results are given for the MRC (RAKE), the MMSE [109], the LP and Hybrid receiver, and the SRBF and MRBF.

[2]NAG: Numerical Algorithms Group Ltd, Oxford, UK. C-code for Sun (SPARC) Solaris, implementation code CLSOL04DA, Mark 4

7.4.1 Results for the SPP technique

The first scenario is a two user CDMA system with randomly generated spreading sequences of length $N = 7$. It is a nonlinearly separable scenario as depicted in Figure 7.7 and 7.3. Figure 7.9 shows the corresponding performance results for the five receivers. Figure 7.9 shows that the MRC outperforms all other receiver structures. This is due to the fact that the MRC takes the channel into account and MAI is low. As anticipated, the MMSE performance converges to the final value of 0.125 ($0.5 \times 4 \times 1/16$) which corresponds to 4 misclassified points out of 16. The MRBF performs best among the four PPB receivers due to the fact that it implements the optimum decision boundary for this 16 point scenario. The proposed slab based receiver (LP) clearly shows its superiority over the simpler linear MMSE receiver. The BER tends to zero as the SNR increases, since it can classify the points correctly. The Hybrid receiver consists of an RBF with four centres out of 16. The graph reveals that the Hybrid receiver outperforms the LP receiver, moreover, it converges to the MRBF's performance. This means that most errors occur within the slabs near the middle hyperplane ($\mathcal{H}_3$ in Figure 7.7) where the distance between the points and their bounding hyperplanes is short.

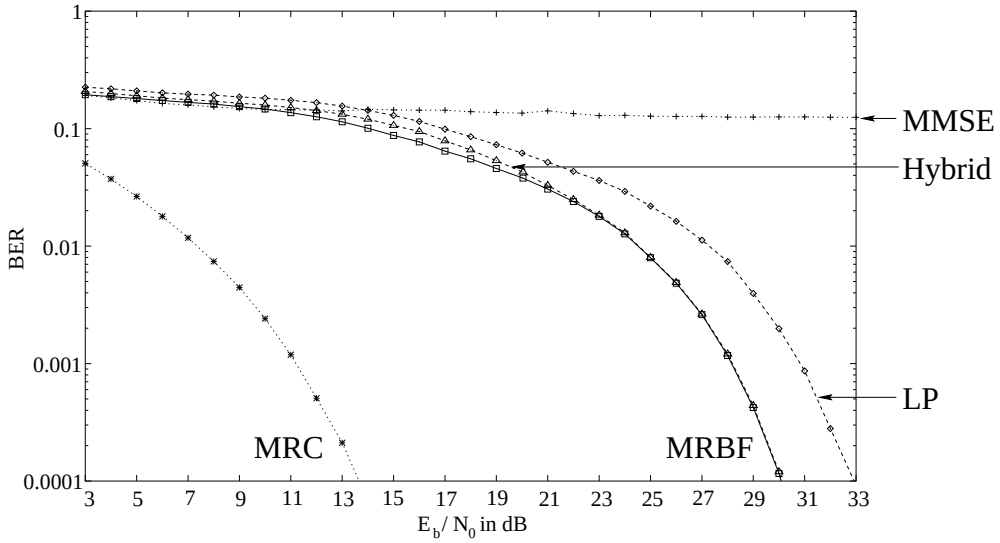


Figure 7.9: BER against SNR for a two user scenario with randomly generated spreading codes of length seven and channel $H_{ch}(z) = 0.25 + 0.5z^{-1} + z^{-2}$.

A similar scenario but for three users is given in Figure 7.10. The major difference is the fact that the MRC performs as badly as the MMSE. This is due to the fact that this scenario is non-linearly separable for both the preprocessed and the simple MRC case. The MRC performance converges to 0.125 ($0.5 \times 128 \times 1/512$) which corresponds to 128 misclassified points out of

512 points[3]. The MRBF performs best, followed by the Hybrid and the LP receiver. The Hybrid consists of 24 centres out of 64 possible points. The LP consists of five units and hence hyperplanes. The performance of both the LP and Hybrid is not much worse than the MRBF with 64 centres.

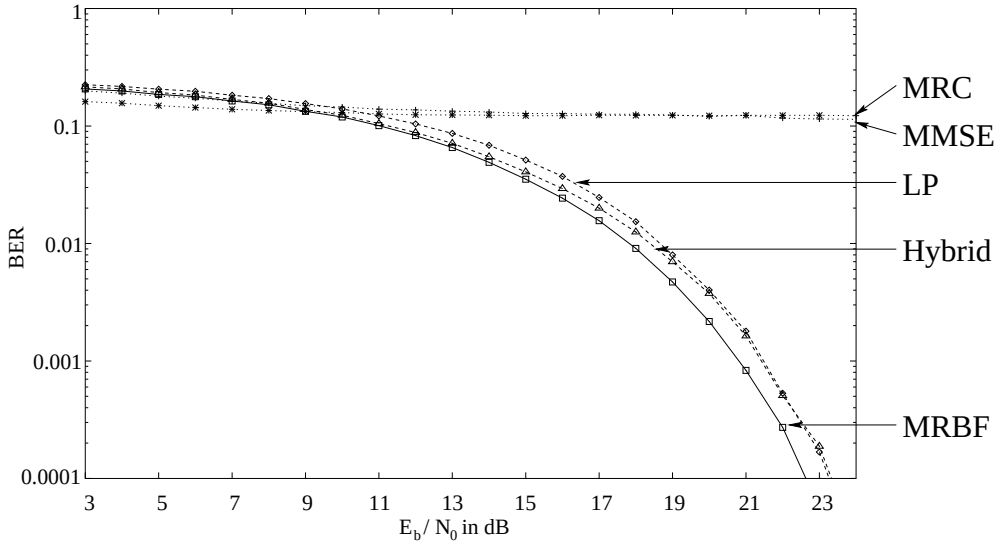


Figure 7.10: BER against SNR for a three user scenario with randomly generated spreading codes of length seven.

Another two simulations were made for CDMA scenarios with seven chip Gold codes. Figure 7.11 shows the performance for three users and Figure 7.12 for five users. The three user scenario has 3 hyperplanes and the Hybrid receiver has 16 centres out of 64 points. The MRC performs best since the scenario is linearly separable in the MRC domain, followed by the MRBF, the Hybrid and LP receiver. The MMSE performance converges to the value 0.125 ($0.5 \times 16 \times 1/64$) which corresponds to 16 misclassified points out of 64.

Figure 7.12 shows the results for a five user scenario. This scenario is nonlinearly separable in the SPP and MRC preprocessed domain. Figure 7.12 illustrates that the MRC fails to outperform the SPP based receivers. The MRC performance converges to 0.1, which corresponds to about 6554 misclassified points out of 32768. The LMS based SPP MMSE receiver converges to a BER of 0.137 which corresponds to about 280 misclassified points out of 1024. The LP receiver has five hyperplanes and the Hybrid receiver has 512 centres out of 1024 points. The LP outperforms the MRC and the MMSE at around 31dB, and converges to zero as the SNR further increases. The poor performance of the LP is due to the fact that many points lie

[3]For the MRC, the total number of points equals 2^{3U} and not 2^{2U}.

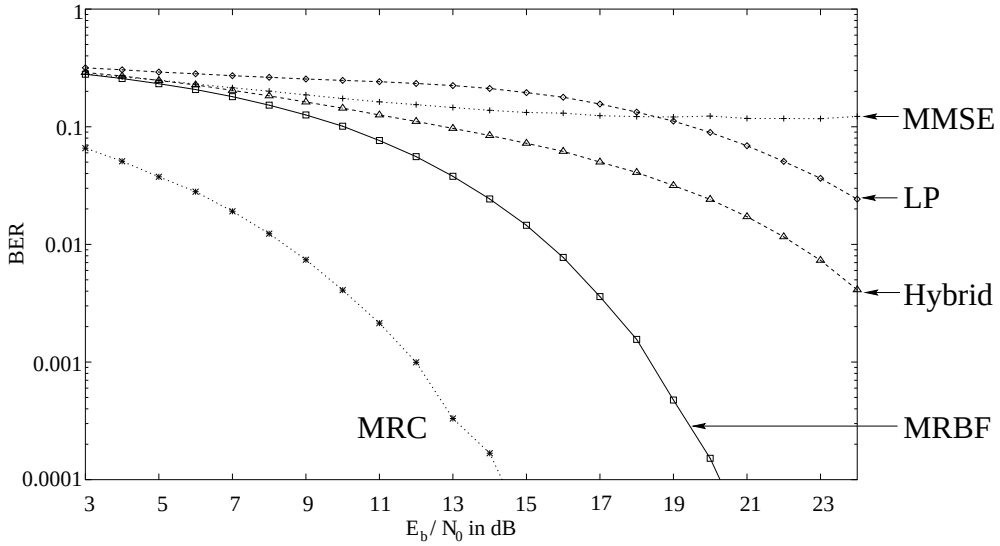


Figure 7.11: A three user scenario with 7 chip Gold codes.

near hyperplanes since one slab is thin which results in many misclassifications. The Hybrid performs nearly as well as the SPP MRBF mainly because of the large number of centres.

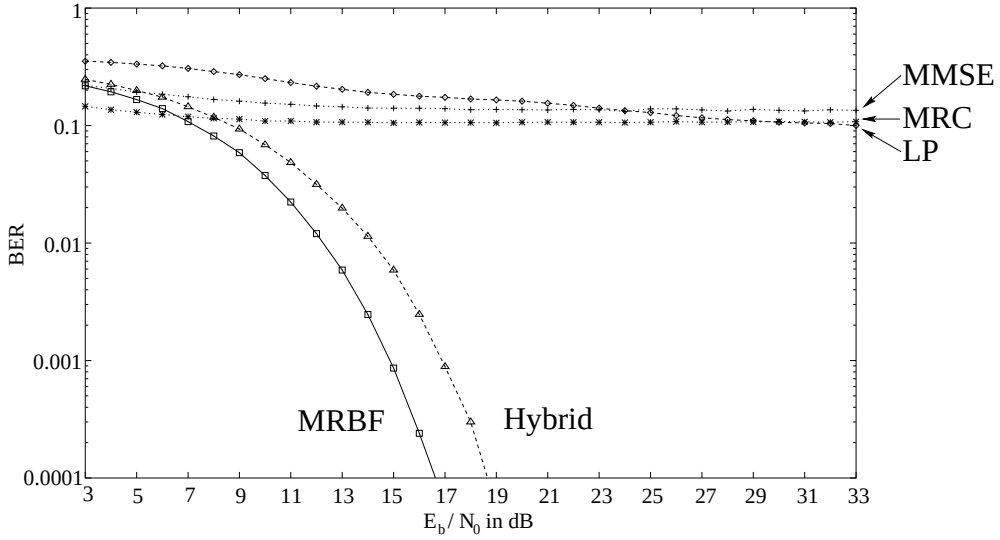


Figure 7.12: A five user scenario with 7 chip Gold codes.

7.4.2 Results for the PPB technique

A four user scenario with randomly generated 7 chip spreading codes was found to be non-linearly separable. Figure 7.13 shows the performance for six different receivers. The MRC converges to a BER of 0.133 which suggests that 544 points are misclassified, out of 4096. The

MMSE receiver which exploits the MRC preprocessed signals, converges to a BER of 0.0056 which is 24 misclassified points. The LP consists of 5 hyperplanes and can classify the points correctly, provided that the SNR is sufficiently high such that the distance between hyperplane and point is greater than the noise spread. Correct classification occurs at 48dB for the LP and Hybrid (not shown in Figure 7.13). The RBF of the hybrid receiver has 112 centres out of 4096. The other two receivers are the MRBF with 4096 centres and the SRBF with 16 centres. Surprisingly, Figure 7.13 shows that even the SRBF can cope with this nonlinearly separable scenario. This means that the clusters do not intersect but are ordered in space in such a manner, that no single hyperplane exists which can separate all points.

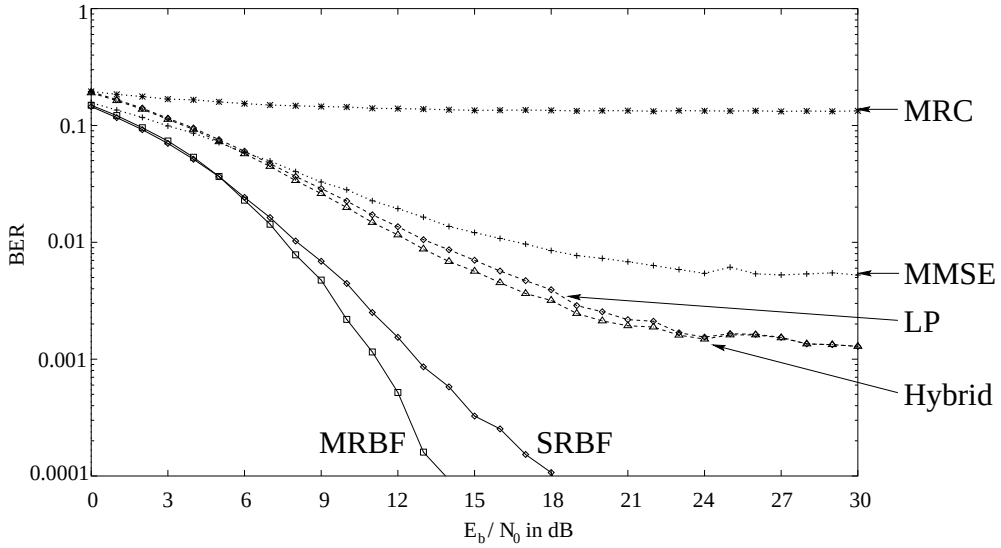


Figure 7.13: A four user scenario with randomly generated spreading codes of length seven.

Also the five user scenario was found to be nonlinearly separable which is presented in Figure 7.14. In this scenario, the points of the clusters intersect. The LP structure has three hyperplanes and the RBF of the Hybrid structure consists of 15872 centres out of 32768. Since this number is not a power of two the hyperplanes do not separate clusters, moreover, hyperplanes divide clusters. Therefore the distance between the points and the hyperplanes is short and many classification errors occur at low SNR, yielding to a high SNR requirement in order to classify the points perfectly. Figure 7.14 shows that the MRC converges to a BER of 0.145 and the MMSE to a BER of 0.0068. It also reveals that LP and Hybrid do not perform well at low SNR. A high SNR is required to outperform the MMSE. The MRBF performs best since it is the only structure which can perfectly classify the points. Because of the fact that the clusters intersect, the SRBF converges to a BER of 0.00026. This means that only a symbol by symbol

based receiver, which takes all points into consideration, can resolve this scenario.

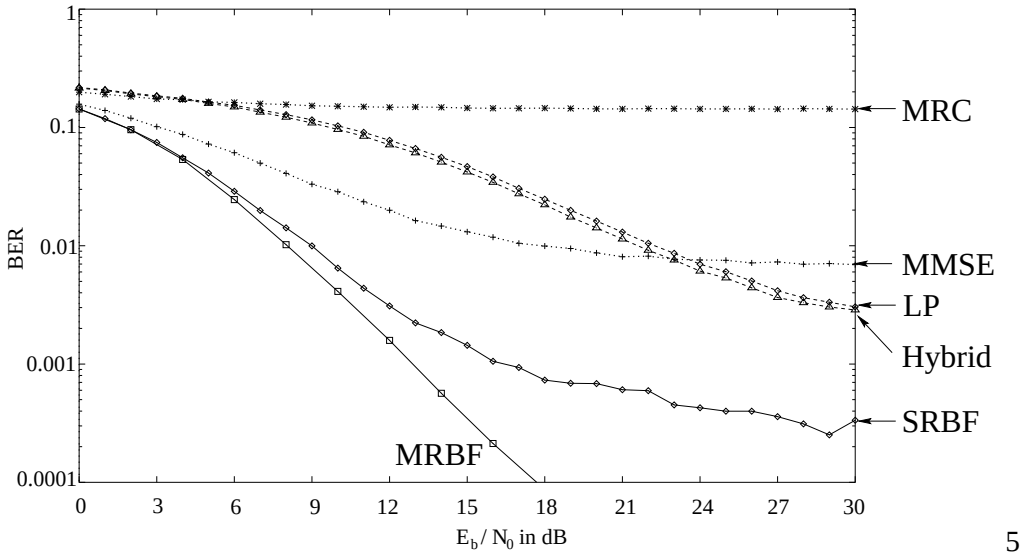


Figure 7.14: A five user scenario with randomly generated spreading codes of length seven.

The last graph, Figure 7.15, presents results for a five user scenario with 7 chip Gold codes. The LP has five hyperplanes and the RBF of the Hybrid structure consists of 256 centres out of 2^{15}. The SRBF has 32 centres where each super centre replaces 1024 centres, and the MRBF has 2^{15} centres. The number of centres in the Hybrid suggests that the LP does not separate clusters but individual points, otherwise the number would be a multiple of 1024. Moreover, if the clusters intersect then the SRBF would not be able to classify them, but it does. The SRBF classifies the points correctly as the noise reduces. Both LP and Hybrid suffer from the fact that LP separates points and not clusters. If a hyperplane is placed through a cluster, then the distance between a point and a hyperplane is short and at low SNR many misclassifications occur. The results show that MRBF and SRBF perform best followed by the Hybrid and LP structure. Both linear receivers, MRC and MMSE, converge to their BER floor of of 0.108 and 0.002 respectively. The LP and Hybrid outperform the MMSE at 12dB. Between a SNR of 30dB and 48dB (not shown), LP and Hybrid have a constant BER before they tend to a zero BER. This means that the hyperplanes lie close to the points and a high SNR is required in order to classify them correctly.

In a further step, the LP (SLAB) algorithm was fed with all super centres. The SLAB could separate the super centres with a single hyperplane. This means that the clusters, or in other words, the ISI induced spread, is responsible for nonlinear separability. It would be of interest to prove that the super centres are vertices of a hypersphere and linearly separable.

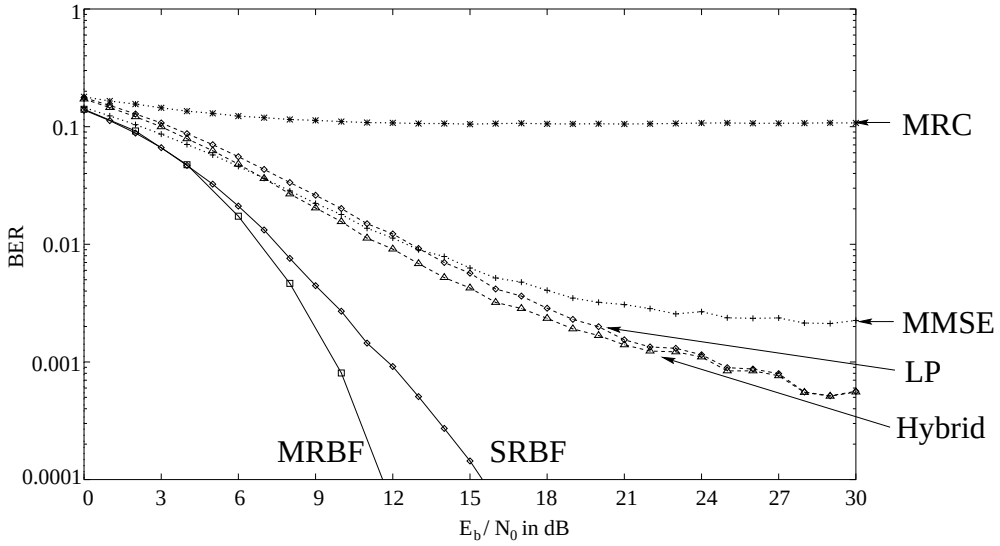


Figure 7.15: A five user scenario with Gold spreading codes of length seven.

7.5 Discussion

This chapter presented two novel DS-CDMA receiver structures tailored for nonlinearly separable scenarios. A brief review introduced linear programming and the definition of linear separability followed by the formulation of the preprocessing stage used. The object was not to focus on receivers with this preprocessing stage but on techniques which can resolve nonlinearly separable scenarios. Then these techniques may be used in a modified version to enhance the performance of the established linear receivers further, such as the decorrelating and MMSE receiver.

Before introducing the SLAB algorithm, the well known linear programming algorithm was reviewed. The SLAB algorithm computes hyperplanes which can classify the two sets of points correctly in the absence of noise. A very simple MLP based NN is used for implementation, where each unit constructs a hyperplane. The Hybrid receiver consists of a reduced MLP NN and an RBF with a small number of centres. The RBF is constructed in conjunction with the narrowest slab, and has as its centres all points which lie between the two hyperplanes. This shall ensure that the points which are most difficult to classify are processed by the optimum method for classification. First, the received signal is analysed to determine if it lies within this slab. If so, the RBF processes it, otherwise the MLP computes the receiver output. Usually, most points lie outside this slab thus the RBF structure is not always used. Thus, the RBF complexity may not be a significant drawback, because there is more time left for signal processing at the RBF. In the examples presented, the RBF processed between 25% and 50% of the received signals with a smaller network. Simulation results showed that both structures outperform the equivalent linear receivers. In the given examples, the Hybrid receiver is superior to LP and often performed as well as the MRBF. It can even happen that the LP and Hybrid receiver outperform the MRC which takes the channel into account but suffers from MAI. The simulation results also reveal some weaknesses of the two proposed receiver structures, LP and Hybrid. In general, a high SNR is essential in order to classify the received signals correctly. On one hand, this is due to the poor preprocessing scheme applied (SPP), on the other hand, the SLAB is designed to separates points and not clusters. This suggests a reformulation of the SLAB algorithm in order to separate clusters rather than points. This might result in a better overall performance since the distance between the clusters is larger than the distance between the points within a cluster. The short distance between points and hyperplanes affects the LP more than the Hybrid since the LP relies fully on the predetermined hyperplanes and can not

adjust them in order to optimise performance. The Hybrid takes all critical points as centres and thus enhances its performance. The cost of this performance improvement may be the number of centres selected based on a criterion. There might be some room for improvements, such as optimising the criterion for selecting the RBF centres of the Hybrid structure. Alternatively, all points could be selected which have a certain distance to a hyperplane which is less than a predefined threshold, dependent on the expected noise spread.

In order to improve established linear CDMA receivers, it may be worth considering combining them with RBF. Such a sub-optimum linear boundary or a part thereof, which may consists of many linear boundaries, may be first computed with the decorrelating detector (or MMSE), instead with LP due to its growing complexity. Further, this technique can also be used to construct a nonlinear receiver which consists of piecewise linear decision boundaries, or a hybrid structure with a simple linear decision boundary and a RBF which has all the critical points as its centres.

Chapter 8
Conclusions

This thesis began with an introduction to mobile communications and DS-CDMA, reviewed established receiver structures and introduced pattern recognition techniques. Then the main novelty of this thesis suboptimum nonlinear receivers for DS-CDMA, was introduced and analysed.

8.1 Summary

Chapter 1 surveyed current trends in mobile communications and the efforts to establish a global standard for mobile systems. The requirements for a cellular mobile system were discussed, with the focus on mobile handsets. Problems which arise in a cellular system were addressed and briefly discussed. Finally, the thesis structure was outlined.

Chapter 2 was concerned with spread spectrum communications and its application for cellular DS-CDMA systems. The downlink of a DS-CDMA system is considered because of the constraints associated with mobile handsets. A description of the downlink in a cellular DS-CDMA was presented from the viewpoint of set theory. Each possible noise free signal state is interpreted as a pattern, or point, in a d-dimensional space and stored in a generation matrix. This analysis was conducted for two receiver concepts, the chip rate based receiver and the symbol rate based receiver. A receiver's task is then to classify the received pattern and thus estimate the transmitted user bit sign. Then a brief description was given for the DS-CDMA mobile standard IS-95.

Chapter 3 reviewed equalisers and DS-CDMA receiver structures. First, linear and nonlinear equalisers were discussed. Simulation results revealed their performance and showed that nonlinear structures can outperform linear structures. Then DS-CDMA receivers were investigated. The autocorrelation matrix of the symbol rate MMSE receiver was derived because the result can be used in the succeeding chapters. Again, linear and nonlinear receivers were compared against each other and Monte-Carlo simulations showed that nonlinear receivers can be superior.

Chapter 4 introduced some pattern recognition techniques which may be unfamiliar to the communications engineer. First, linear and nonlinear pattern classification techniques were discussed. Examples showed different classification tasks and possible separation boundaries. Once more, nonlinear techniques were shown to be better than linear techniques in most cases. However, linear techniques are simpler and thus more attractive for implementation. Decision boundaries were presented for two and three user DS-CDMA scenarios for a linear and the Bayesian structure. They showed that the optimum decision boundary is nonlinear and gave reasons for the limited performance of linear receivers. Finally, a brief introduction to approximation theory was given.

In Chapter 5 a novel nonlinear receiver was presented for DS-CDMA. Its structure depends upon the Volterra series expansion, which is well known from channel equalisation applications. The proposed receiver is a chip rate based structure which consists of an expansion stage and a filter stage, implemented by a FIR filter. The filter weights are derived from the Wiener-Hopf equation, which has been extended to the nonlinear Volterra sequence. It was shown that only odd order Volterra structures can be applied to DS-CDMA receivers since the even order crosscorrelation terms are zero. This adds additional complexity to the receiver since the Volterra structure becomes computationally expensive due to the binomial growth of the number of Volterra coefficients and the requirement for a matrix inversion. The expansion process also introduces higher order terms, which has hampered attempts to derive either the theoretical performance or an algorithm which computes the required autocorrelation matrix. Alternatively, an estimation technique was presented which computes the autocorrelation matrix. Monte-Carlo simulations compared the proposed receiver against established linear and nonlinear receivers. The simulations showed that in AWGN the Volterra receiver outperforms the MMSE, because the MMSE corresponds to the first order Volterra structure, which is a subset of the proposed higher order Volterra receiver. The performance gain achieved in multipath over the MMSE is even bigger due to the additional nonlinearity of the scenario. Decision boundaries were presented for the symbol rate Volterra receiver. They showed that low order Volterra structures are unable to closely approximate the optimum decision boundary, which results in limited performance improvement over the MMSE for symbol rate systems.

In Chapter 6 the established chip rate based RBF receiver was introduced and extended to the symbol rate. Since the preprocessing is conducted with matched filters instead of whitening filters, the Euclidean distance measure is not optimum. Thus the Mahalanobis distance measure

was introduced. Simulation results confirmed that the Mahalanobis distance measure is optimum for symbol rate based RBF receivers. However, this RBFN is too complex when used in multipath. A RBFN with reduced complexity was proposed. It exploits the fact that ISI causes clustering of the patterns. By replacing each cluster of points with a single point, the number of RBF centres is reduced from 2^{3U} to 2^U and hence the RBFN complexity is also reduced. Monte-Carlo simulations compared this novel DS-CDMA receiver against established receivers and different RBF structures. The proposed RBFN which exploits super centres was shown to be an excellent compromise between performance and complexity, since an 8 user DS-CDMA system only requires 256 centres instead of 2^{24}. The SRBF outperformed the MMSE and can have little performance loss against the *full* RBF which consists of all possible centres. The results showed that the centre reduction may not significantly affect the performance, compared with an equivalent RBF which has 2^{3U} centres. The SRBF was also shown to be less susceptible to nonlinearly separable scenarios than the MMSE.

In Chapter 7 a novel hybrid receiver was presented. It is tailored for nonlinearly separable scenarios which can arise when short spreading sequences are used. It combines the good features of the RBFN and the simple structure of an MLP with linear nodes. The weights of the MLP are computed with linear programming and used to construct hyperplanes which enable the network to classify the received signals perfectly in the noise free transmission case. The RBFN is then constructed from these hyperplanes and takes as centres those points which cause the scenario to be nonlinearly separable. Thus, all sequences which are responsible for the failure of a linear receiver become RBF centres. Simulation results compared the receiver against established receivers and showed that it outperforms the MMSE. However, due to the fact that these points are close to each other in the d-dimensional space, a high SNR is required in order to achieve a reasonable performance such as would be required for DS-CDMA.

8.2 Conclusions

This thesis shows that a DS-CDMA system can be interpreted and described as a pattern recognition problem. Each received noise free state is a pattern made up of the spreading sequences of the active users. A single user receiver's task is to choose between two classes whereas a multiuser detector has more options, dependent on the defined classification scheme. The classification task can be accomplished either by constructing or approximating the optimum decision boundary, or by assigning the received signal to the class of points to which it most

likely belongs to. It is shown that both techniques work successfully in DS-CDMA. To the authors best knowledge, there is no other publication which connects DS-CDMA and pattern recognition, as has been done in this work.

Moreover, not many people working with DS-CDMA seem to be aware of the fact that DS-CDMA is a nonlinear separation problem. Thus it is worth emphasising this fact. However, a nonlinear problem requires a nonlinear solution. Nonlinear techniques are known to be computationally expensive, not as tractable as linear techniques, and difficult to understand. Nevertheless, increasing DSP power will enable the engineer to apply nonlinear techniques.

It was shown that only odd order Volterra structures can be applied to DS-CDMA. Simulations showed that the Volterra receiver outperforms the MMSE, and the performance gain achieved in multipath is even bigger.

It was shown that the Mahalanobis distance measure is the optimum distance measurement for the symbol rate RBF with MF preprocessing. The proposed SRBF was shown to be an excellent compromise between complexity and performance. The SRBF outperformed the MMSE and is also less susceptible to nonlinearly separable scenarios.

It was shown that the LP based hybrid receiver outperforms the MMSE.

To conclude, DS-CDMA is a pattern recognition problem thus pattern recognition techniques can be applied in order to enhance receiver performance.

8.3 Future work

The basic problem in developing a DS-CDMA receiver which has better performance than any linear receiver is to find an appropriate approximation technique. Such a technique has two constraints: it should have a simple receiver structure with little computational overhead and provide a good approximation. The receivers reported in this thesis also have scope for further work.

The Volterra receiver's filter weights were computed by an estimation technique which introduces an error. Thus there is room to search for an algorithm which computes the filter weights directly, and to derive the theoretical receiver performance. Although it does not suit practical considerations in the presented form, its adaptive version may be of practical interest.

The proposed SRBF has been investigated for stationary scenarios. Thus it is worth looking at its performance in a fading environment which requires a continuous centre update. Since a mobile receiver must be adaptive, the SRBF should be made adaptive.

The proposed LP and hybrid receiver performed well in terms of signal classification. However, further investigation is needed to prove that nonlinear separable scenarios are frequent in the real mobile environment. The author's empirical experience is that long spreading codes reduce the likelihood of nonlinearly separable scenarios (or maybe prohibit it). It was also shown that the required RBFN can be large in terms of the number of centres while in addition its performance can also be poor at low SNR. Thus the presented LP and hybrid receivers may not suit practical applications. However, in a further step, LP could be used to compute hyperplanes which approximate the optimum decision boundary, for instance with three hyperplanes for the depicted scenario in Figure 4.6(b).

Other attempts in this direction have been proposed in [190] and in [191] for DS-CDMA, both techniques being based upon the Voronoi diagram [192]. The problem is to find the points which are (mainly) responsible for the performance, which are the points near the optimum decision boundary. One attempt may be to start off with a linear boundary and search for the critical points close to it, which is a nearest neighbour search problem [146, 148–150, 193]. Another path is to look at NNs in order to construct their network [180–182, 194–196] instead of using some sort of training which is time consuming. Also of interest might be a polynomial structure (e.g. [153]), where the signal which is fed into the receiver is orthogonalised [197, 198] or processed with an orthogonalising preprocessor.

Appendix A
Distribution functions

The pdf $g(y)$ of a 3rd-order Volterra expansion sequence $\mathbf{v}$ for different values of α and β. The derivation is given in subsection 5.2.1.

For $\beta = 1$

$$g(y) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{y^2}{2}\right) \tag{A.1}$$

For $\alpha = 2$

$$g(y) = \frac{1}{\sqrt{y}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{y}{2}\right) \tag{A.2}$$

For $\beta = 3$

$$g(y) = \frac{1}{3} \frac{1}{\sqrt[3]{y^2}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\sqrt[3]{y^2}}{2}\right) \tag{A.3}$$

For $\alpha = 4$

$$g(y) = \frac{1}{2} \frac{1}{\sqrt[4]{y^3}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\sqrt{y}}{2}\right) \tag{A.4}$$

For $\beta = 5$

$$g(y) = \frac{1}{5} \frac{1}{\sqrt[5]{y^4}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\sqrt[5]{y^2}}{2}\right) \tag{A.5}$$

For $\alpha = 6$

$$g(y) = \frac{1}{3} \frac{1}{\sqrt[6]{y^5}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\sqrt[3]{y}}{2}\right) \tag{A.6}$$

If $x \sim \mathcal{N}(0, \sigma^2)$, then the moments are:

$$E[x^n] = \begin{cases} 0 & \text{if } n \text{ odd} \\ 1 \cdot 3 \cdots (n-1)\sigma^n & \text{if } n \text{ even} \end{cases} \tag{A.7}$$

The odd moments of x are zero because $f(-x) = f(x)$ [165].

Appendix B
Investigation of VS algorithm

The problem encountered with designing an algorithm which computes $\mathbf{R}_{vv}$ and $\mathbf{r}_{xv}$ for higher-order Volterra systems is considered. Since the spreading sequence length can be assumed to be constant, the VS expansion length is fixed, hence, the algorithm is only required to compute $\mathbf{R}_{vv}$ and $\mathbf{r}_{xv}$ for a varying number of users. First, some definitions are reviewed, then the technique used to develop an algorithm is presented. This technique is first applied to the 2nd order Volterra system, since it is less complex and then for the 3rd order system. However, it seems to be the case that there is no straightforward way to compute $\mathbf{R}_{vv}$ and $\mathbf{r}_{xv}$. It would be desirable to develop an algorithm which works for any number of users and for any term within $\mathbf{R}_{vv}$ and $\mathbf{r}_{xv}$.

B.1 Review

The received signal $\mathbf{y}(k)$ for bit k and spread code length $N = 3$ ($n = 1, 2, \ldots, N$) has elements $[y_1 \; y_2 \ldots y_N]^T$. Index k is not relevant for the derivation and is no longer stated. According to (2.6), vector $\mathbf{y}$ is for U users:

$$\mathbf{y} = [y_1 \; y_2 \; y_3]^T = \left[\sum_{u=1}^{U} D_u c_{u,1} + g(1) \; \sum_{u=1}^{U} D_u c_{u,2} + g(2) \; \sum_{u=1}^{U} D_u c_{u,3} + g(3) \right]^T,$$

where $g(n)$ is the uncorrelated noise with zero mean, D_u the data bit which is equally probable of being $+1$ or -1 and $c_{u,n}$ the nth chip of the uth users spreading sequence. Thus for a two user system ($U = 2$), the vector elements $\{y_n\}$ are:

$$\begin{aligned} y_1 &= D_1 c_{1,1} + D_2 c_{2,1} + g(1) \\ y_2 &= D_1 c_{1,2} + D_2 c_{2,2} + g(2) \\ y_3 &= D_1 c_{1,3} + D_2 c_{2,3} + g(3), \end{aligned}$$

The Volterra expansion performs a mapping from N space onto M space (see equation 5.4) where $\mathbf{v}$ is derived from $\mathbf{y}$.

B.2 Second-order Volterra system

For a 2nd-order Volterra system, elements $\{v_m\}$, for $m = 1, 2, \ldots, M$ with $M = M_{12} = 3 + 6 = 9$ are:

$$\begin{aligned}
v_1 &= D_1c_{1,1} + D_2c_{2,1} + g(1) \\
v_2 &= D_1c_{1,2} + D_2c_{2,2} + g(2) \\
v_3 &= D_1c_{1,3} + D_2c_{2,3} + g(3) \\
v_4 &= (D_1c_{1,1} + D_2c_{2,1} + g(1))^2 \\
v_5 &= (D_1c_{1,1} + D_2c_{2,1} + g(1))(D_1c_{1,2} + D_2c_{2,2} + g(2)) \\
v_6 &= (D_1c_{1,1} + D_2c_{2,1} + g(1))(D_1c_{1,3} + D_2c_{2,3} + g(3)) \\
v_7 &= (D_1c_{1,2} + D_2c_{2,2} + g(2))^2 \\
v_8 &= (D_1c_{1,2} + D_2c_{2,2} + g(2))(D_1c_{1,3} + D_2c_{2,3} + g(3)) \\
v_9 &= (D_1c_{1,3} + D_2c_{2,3} + g(3))^2,
\end{aligned} \tag{B.1}$$

The VS expansion also introduces terms with higher-orders. So for instance, there are three groups of them within the expansion vector (B.1) of a 2nd-order system:

$$p_1 = \{v_1, v_2, v_3\} \equiv E\{y_i\} \quad p_2 = \{v_4, v_7, v_9\} \equiv E\{y_i^2\} \quad p_3 = \{v_5, v_6, v_8\} \equiv E\{y_i y_j\}$$

with $i, j \in \{1, 2, \ldots, N\}$, and each group has its unique statistics. The expected value for each group p_t (t is an index with no special meaning) consists of two components, the signal and the noise. Group p_1 has the noise statistics of the received signal, which is Gaussian. The square law applies to group p_2, since the input noise is Gaussian, hence noise becomes Chi-square distributed [163]. Finally, there is no noise component in p_3, because the noise terms are uncorrelated to each other and therefore the expected value is zero [165]. Since $\mathbf{R}_{vv} = E[\mathbf{v}\mathbf{v}^T]$, products among p_t arise which consist of even higher-orders. Again, these terms can be grouped

according to their statistics as:

$$\begin{aligned} G_1 &= E\{y_i y_j\} & G_3 &= E\{y_i^3\} & G_6 &= E\{y_i^4\} \\ G_2 &= E\{y_i^2\} & G_4 &= E\{y_i^2 y_j\} & G_7 &= E\{y_i^3 y_j\} \\ & & G_5 &= E\{y_i y_j y_l\} & G_8 &= E\{y_i^2 y_j^2\} \\ & & & & G_9 &= E\{y_i^2 y_j y_l\} \end{aligned} \tag{B.2}$$

with $i, j, l \in \{1, 2, \ldots, N\}$. The resulting matrix $\mathbf{R}_{vv}$ which shows all possible expected values G_t is presented in Figure B.1.

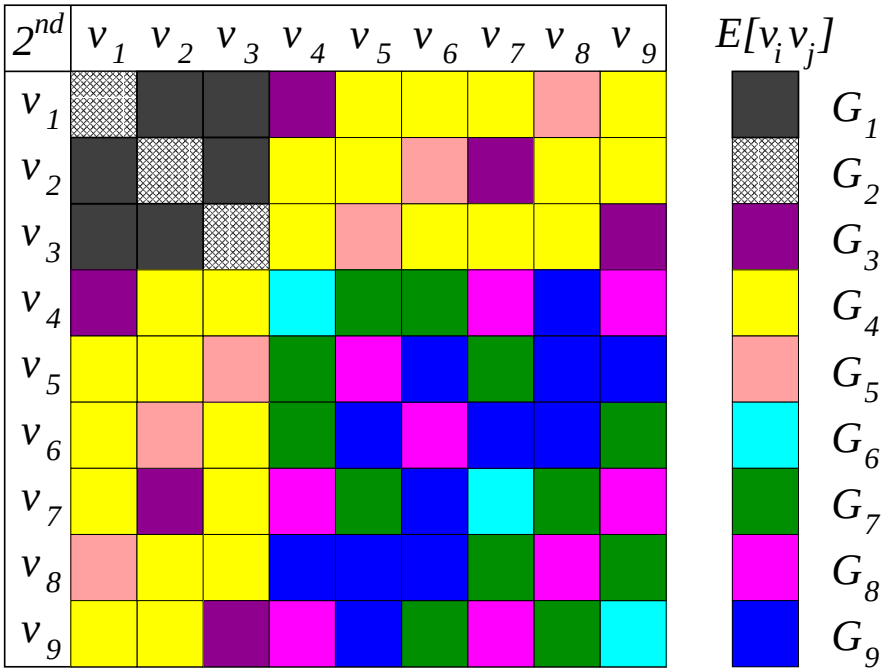


Figure B.1: $\mathbf{R}_{vv}$ of a 2nd-order Volterra system, each colour represents an expected value.

$$
\begin{aligned}
\mathbf{v} \;=\; & [v_1\, v_2 \ldots v_N\, v_{N+1} \ldots v_M]^T \;= \\
& [\,(D_1 + D_2 + g(i)) \\
& \;(D_1 + D_2 + g(j)) \\
& \;(D_1 + D_2 + g(l)) \\
& \;((D_1 + D_2 + g(i))^2) \\
& \;((D_1 + D_2 + g(i))(D_1 + D_2 + g(j))) \\
& \;((D_1 + D_2 + g(i))(D_1 + D_2 + g(l))) \\
& \;((D_1 + D_2 + g(j))^2) \\
& \;((D_1 + D_2 + g(j))(D_1 + D_2 + g(l))) \\
& \;((D_1 + D_2 + g(l))^2)\,]^T. \qquad \text{(B.3)}
\end{aligned}
$$

Next, the idea used for developing an algorithm to compute $\mathbf{R}_{vv}$ is presented. In order to simplify the notation used, the signal component $c_{u,n}$ are no longer stated, since the probabilistic variables D_u and $g(n)$ determine if a product between $D_u c_{u,n}$ and $g(n)$ is part of the expected value $E[G_t]$ or not. Thus, vector $\mathbf{v}$ is rewritten in a simpler form, where the chips $c_{u,n}$ are discarded. Further, $\{v_m\}$ shall list all possible product terms and it shall not be merged and simplified, as it is common practice for binomial series. The notation becomes very complex if U is large. Thus only the two and the three user scenarios are presented. $\mathbf{v}$ is restated without the signal components in (B.3) with $i, j, l \in \{1, 2, \ldots, N\}$. Terms of the three different main products evolve and are (see also Figure 5.3):

• Linear-linear products, which are known from the MMSE solution [108].

• Linear-quadratic products, products between the linear terms $\{v_1, v_2, \ldots, v_N\}$ and the quadratic terms $\{v_{N+1}, v_{N+2}, \ldots, v_M\}$ which result in the most general term:

$$
\begin{aligned}
\left[D_1^2 + D_1 D_2 + D_1 g(l) + D_1 D_2 + D_2^2 + D_2 g(l) + D_1 g(j) + D_2 g(j) + g(j) g(l)\right] \;\times \\
[D_1 + D_2 + g(i)] = 0 \qquad \text{(B.4)}
\end{aligned}
$$

Thus all linear-quadratic terms are zero and do not have to be computed.

• Quadratic-quadratic products, where it turns out that the terms of the right column in (B.2)

reduce to the more general form:

$$[D_1 + D_2 + g(i)]^2 \times [(D_1 + D_2 + g(j))(D_1 + D_2 + g(l))]. \tag{B.5}$$

Since this is a quadratic function, it makes sense to apply a graphical representation to find the expected value $\{G_6, G_7, G_8, G_9\}$ in (B.2). $E[G_t]$ is computed by adding up all products occurring in equation (B.5) which are not zero. Product terms whose result is not equal to zero are referred to as $true\ terms$. Again, according to the rule that $D_u^{odd} = 0$, $g(i)^{odd} = 0$ and that $g(i)g(j) = 0$ for $i \neq j$. Thus, only a few true terms remain in (B.5) which are denoted by crosses in Figure B.2 and B.3.

The expected values of a 2nd-order VS system with two users can be computed by adding up the products which are marked by a cross in the following figures, taking the associated signal term $c_{u,n}$ into account. Figures B.2 and B.3 show the pattern for the two user scenario, where each group has its own pattern of crosses. This means that the expected value for each G_t has to be computed separately. Figures B.5, B.6, B.7 and B.8 show the resulting products for the three user case. Again, no joint pattern appears. Moreover, the step from two users to three users shows that no regularity exists among the groups in terms of a pattern which could be exploited in order to construct an algorithm. In a further step (trial), matrix $\mathbf{R}_{vv}$ was split into a signal and a noise matrix, $\mathbf{R}_{vv} = \mathbf{S} + \mathbf{N}$.

Figure B.4 shows the steps used to compute the matrix $\mathbf{S}$ and $\mathbf{N}$ respectively. Figure B.9 shows the signal matrix $\mathbf{S}$, where the noise terms $g(t)$ are set to zero for the two, three and the four user scenario. It appears that within these rasters a joint pattern exists, which is independent of the group G_t membership and the number of users. By rearranging the order of the terms $(D_1, D_2, \ldots, D_U)$ at the top and left side, all crosses appear as a square of crosses, because the crosses form patterns due to the binomial nature of the signals D_1 and D_2. The algorithm found which computes the value which the crosses represent in this special scenario does the following. First, the chips of the spreading codes are multiplied with all possible chips. This is called VS expansion in Figure B.4 and the result ($N \times N$ terms) is stored in an expansion matrix V. Two rows of this expansion matrix are copied into sub-matrices A, B, see Figure B.4, and multiplied with each other in order to compute the value each cross represents. This process exploits the fact that each square in Figure B.9 can be divided into $U \times U$ sub-squares, and each sub-square has a pattern related to the number of users which can be described by an algorithm. Thus, the terms in $\mathbf{S}$ (or $\mathbf{R}_{vv}$ respectively) can be computed according the rule stated as:

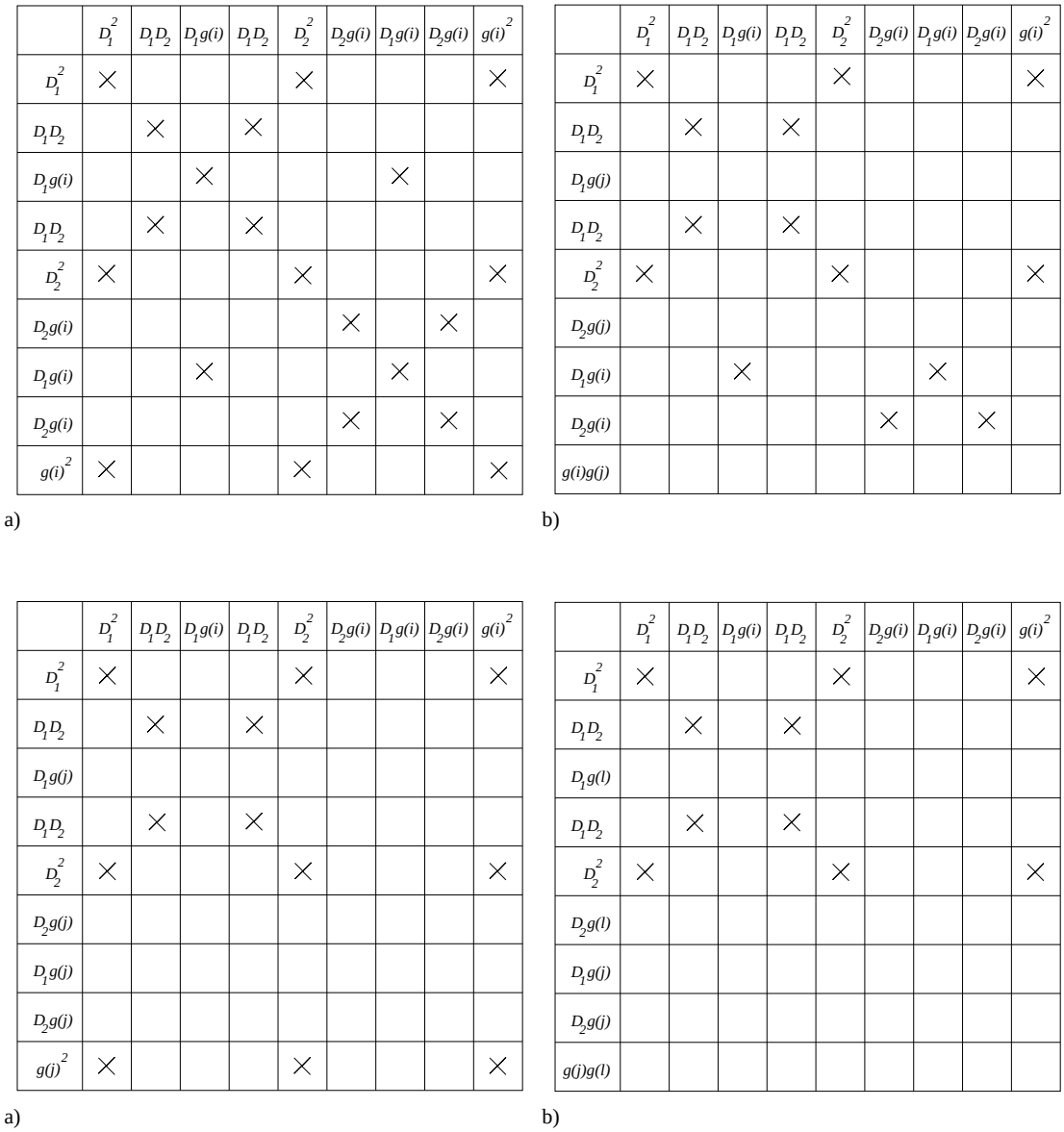


Figure B.3: The true terms for the groups a) G_8 and b) G_9.

1. Compute a VS expansion matrix `V[M][U*U]` derived from a spreading code matrix `C[U][N]`.
2. Copy sequences `V[m][U*U]` into `A[U][U]` and `V[n][U*U]` into `B[U][U]`
3. Compute $\mathbf{R}_{vv}$ from matrix `A[U][U]` and `B[U][U]`.
4. Goto 2 until `(n == M)` and `(m == M)`

A possible implementation in C is presented on page 145.

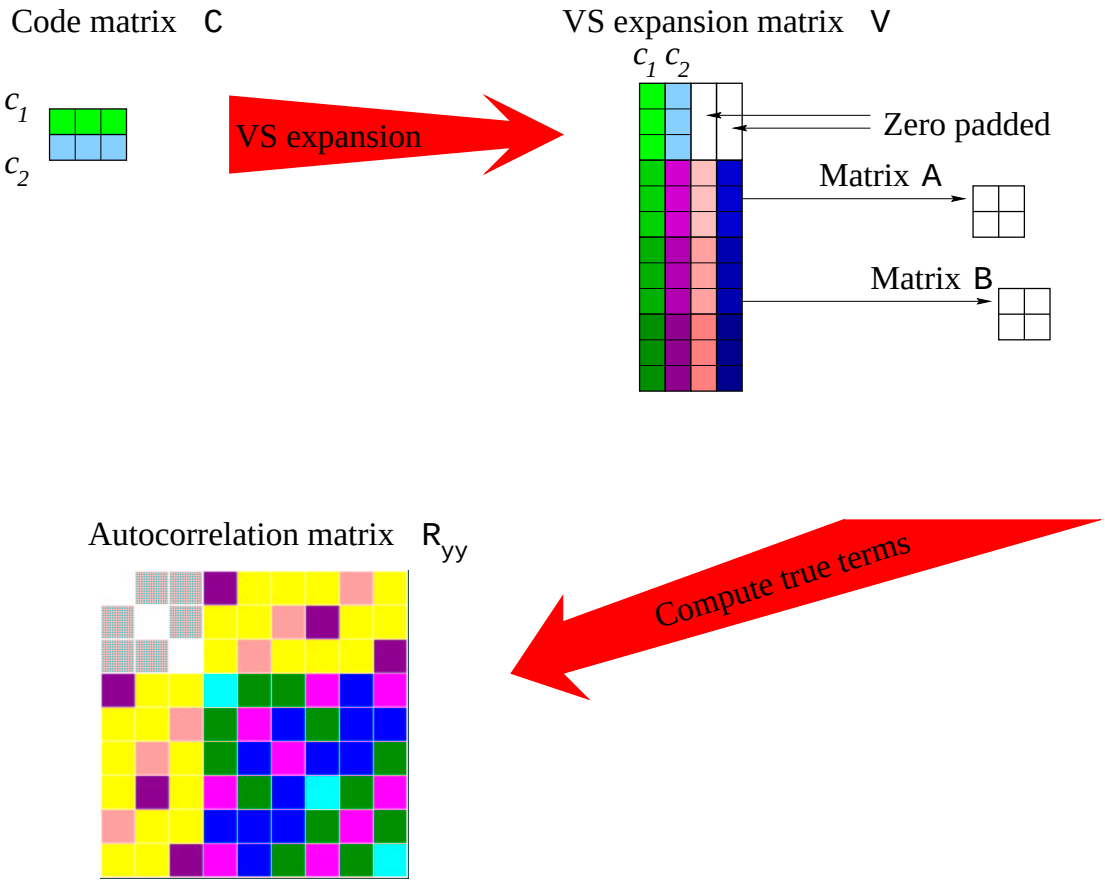


Figure B.4: The steps proposed for computing the autocorrelation matrix of a Volterra receiver.

The noise matrix $\mathbf{N}$, depicted in Figure B.10, deals only with the noise terms. It also bears a structure which could be exploited to construct an algorithm. However, the structure found is dependent on the noise indices. This makes the algorithm inflexible if applied to the groups G_t. This means that many more terms have to be computed separately.

It may be concluded that due to the uncorrelated nature of the noise, it is not straightforward to develop an algorithm to construct the terms $E[v_i v_j]$ for $i, j = \{1, 2, \ldots, M\}$ of $\mathbf{R}_{vv}$, for a 2nd-order Volterra autocorrelation matrix.

The algorithm in C:

```
/*
 * start Volterra expansion: C[U][N] -> V[M][U*U]
 */
 U = user*user;
 for(a=0; a < N ;a++)      /* linear sequence */
   for(b=0; b < U ;b++)
     if( b < user )
       V[a][b] = code[b*N + a];
     else
       V[a][b] = 0.0;
 m = 0;
 for(a=0; a < N ;a++)      /* quadratic sequence */
   for(b=a; b < N ;b++)
     {
      for(i=0; i < user ;i++)
        for(n=0; n < user ;n++)
          V[N + m][i*user + n] = code[i*N + a] * code[n*N + b];
      m++;
     }
/* compute R_yy , first the linear-linear products (Wiener-Hopf)
 */
 for(i=0; i < N ;i++)
   for(j=0; j < N ;j++)
     for(u=0; u < U ;u++)
       Ryy[i][j] += V[i][u]*V[j][u] + V[i][u]*V[j][U - 1 - u];

/* then the quadratic-quadratic products
 */
 for(m=N; m < M ;m++)
   {
    for(i=0; i < user ;i++)
      for(j=0; j < user ;j++)
        A[i][j] = V[m][i*user + j];
    for(n=N; n < M ;n++)
      {
       for(i=0; i < user ;i++)
         for(j=0; j < user ;j++)
           B[i][j] = V[n][i*user + j];
       for(i=0 ; i < user ;i++)
         for(j=0; j < user ;j++)
           Ryy[m][n] += A[i][i] * B[j][j];
       for(i=0 ; i < user ;i++)
         for(j=0; j < user ;j++)
           if( i != j )
             Ryy[m][n] += A[i][j]*B[i][j] + A[i][j]*B[j][i];
      }
   }
```

	D_1^2	D_1D_2	D_1D_3	$D_1g(i)$	D_1D_2	D_2^2	D_2D_3	$D_2g(i)$	D_1D_3	D_2D_3	D_3^2	$D_3g(i)$	$D_1g(i)$	$D_2g(i)$	$D_3g(i)$	$g(i)^2$
D_1^2	×					×					×					×
D_1D_2		×			×											
D_1D_3			×						×							
$D_1g(i)$				×									×			
D_1D_2		×			×											
D_2^2	×					×					×					×
D_2D_3							×			×						
$D_2g(i)$								×						×		
D_1D_3			×						×							
D_2D_3							×			×						
D_3^2	×					×					×					×
$D_3g(i)$												×			×	
$D_1g(i)$				×									×			
$D_2g(i)$								×						×		
$D_3g(i)$												×			×	
$g(i)^2$	×					×					×					×

Figure B.5: The true terms for group G_6 for a three user scenario.

	D_1^2	D_1D_2	D_1D_3	$D_1g(i)$	D_1D_2	D_2^2	D_2D_3	$D_2g(i)$	D_1D_3	D_2D_3	D_3^2	$D_3g(i)$	$D_1g(i)$	$D_2g(i)$	$D_3g(i)$	$g(i)^2$
D_1^2	×					×					×					×
D_1D_2		×			×											
D_1D_3			×						×							
$D_1g(j)$																
D_1D_2		×			×											
D_2^2	×					×					×					×
D_2D_3							×			×						
$D_2g(j)$																
D_1D_3			×						×							
D_2D_3							×			×						
D_3^2	×					×					×					×
$D_3g(j)$																
$D_1g(i)$				×									×			
$D_2g(i)$								×						×		
$D_3g(i)$													×		×	
$g(i)g(j)$																

Figure B.6: The true terms for group G_7 for a three user scenario.

	D_1^2	D_1D_2	D_1D_3	$D_1g(i)$	D_1D_2	D_2^2	D_2D_3	$D_2g(i)$	D_1D_3	D_2D_3	D_3^2	$D_3g(i)$	$D_1g(i)$	$D_2g(i)$	$D_3g(i)$	$g(i)^2$
D_1^2	×					×					×					×
D_1D_2		×			×											
D_1D_3			×						×							
$D_1g(j)$																
D_1D_2		×			×											
D_2^2	×					×					×					×
D_2D_3							×			×						
$D_2g(j)$																
D_1D_3			×						×							
D_2D_3							×			×						
D_3^2	×					×					×					×
$D_3g(j)$																
$D_1g(j)$																
$D_2g(j)$																
$D_3g(j)$																
$g(j)^2$	×					×					×					×

Figure B.7: The true terms for group G_8 for a three user scenario.

	D_1^2	D_1D_2	D_1D_3	$D_1g(i)$	D_1D_2	D_2^2	D_2D_3	$D_2g(i)$	D_1D_3	D_2D_3	D_3^2	$D_3g(i)$	$D_1g(i)$	$D_2g(i)$	$D_3g(i)$	$g(i)^2$
D_1^2	×					×					×					×
D_1D_2		×			×											
D_1D_3			×						×							
$D_1g(l)$																
D_1D_2		×			×											
D_2^2	×					×					×					×
D_2D_3							×			×						
$D_2g(l)$																
D_1D_3			×						×							
D_2D_3							×			×						
D_3^2	×					×					×					×
$D_3g(l)$																
$D_1g(j)$																
$D_2g(j)$																
$D_3g(j)$																
$g(j)g(l)$																

Figure B.8: The true terms for group G_9 for a three user scenario.

	D_1^2	D_1D_2	D_1D_2	D_2^2
D_1^2	×			×
D_1D_2		×	×	
D_1D_2		×	×	
D_2^2	×			×

a)

	D_1^2	D_1D_2	D_1D_3	D_1D_2	D_2^2	D_2D_3	D_1D_3	D_2D_3	D_3^2
D_1^2	×				×				×
D_1D_2		×		×					
D_1D_3			×				×		
D_1D_2		×		×					
D_2^2	×				×				×
D_2D_3						×		×	
D_1D_3			×				×		
D_2D_3						×		×	
D_3^2	×				×				×

b)

	×					×					×				×
		×			×										
			×						×						
				×								×			
		×			×										
	×					×					×				×
							×			×					
								×					×		
			×						×						
							×			×					
	×					×					×				×
												×		×	
				×									×		
								×						×	
												×			
	×					×					×				×

c)

Figure B.9: The graphical representation for the noise free scenario: a) for two users b) three users and c) four users. The labelling in (c) is omitted to save space but it is an extension of that in (a) and (b).

B.3 Third-order Volterra system

The investigation for the 3rd-order Volterra systems has been carried out in the same manner as for a 2nd-order system. Due to the large number of terms, the investigation is limited to one and two users, which is sufficient for an analysis.

• The crosscorrelation representation is presented in Figure B.11(a) and B.11(b) for one and two users respectively. The three different possible combinations among the noise terms are marked by a cross, circle and a square. It turns out that the crosscorrelation must be computed separately for each number of users due to the noise terms.

• The linear-cubic terms for all possible expected values are given in Figure B.11(c) for one user and in Figure B.11(d) for two users. Again, no straightforward algorithm was found which matches the pattern obtained from the theoretical calculations.

• The cubic-cubic pattern is shown in Figure B.12 for the single user scenario. For visualisation purposes, the possible products of equation (5.15) given the noise indices $i, j, l \in \{1, 2, \ldots, N\}$ were denoted by three symbols. In order to search for a useful pattern, the graphical investigation has also been done for the noise free case. Thus, Figure B.13(a) shows the result for the two user, and Figure B.13(b) for the three user scenario. However, it turned out that within each scenario, several patterns exist. Hence, to determine each term in $\mathbf{R}_{vv}$, different ways to compute the (quadratic) product must be used. This is a tedious task if U is large. Thus, it may be concluded that no straightforward algorithm exists to compute the autocorrelation matrix and crosscorrelation vector.

One solution could be suggested for stable systems ($U = const$) with look-up tables, where beforehand the computation for each terms has been specified and then stored in a memory as a rule. So for different noise scenarios, the expected value is determined by the rule given in the look-up table. However, this is an impractical solution for real DS-CDMA scenarios where U changes with time.

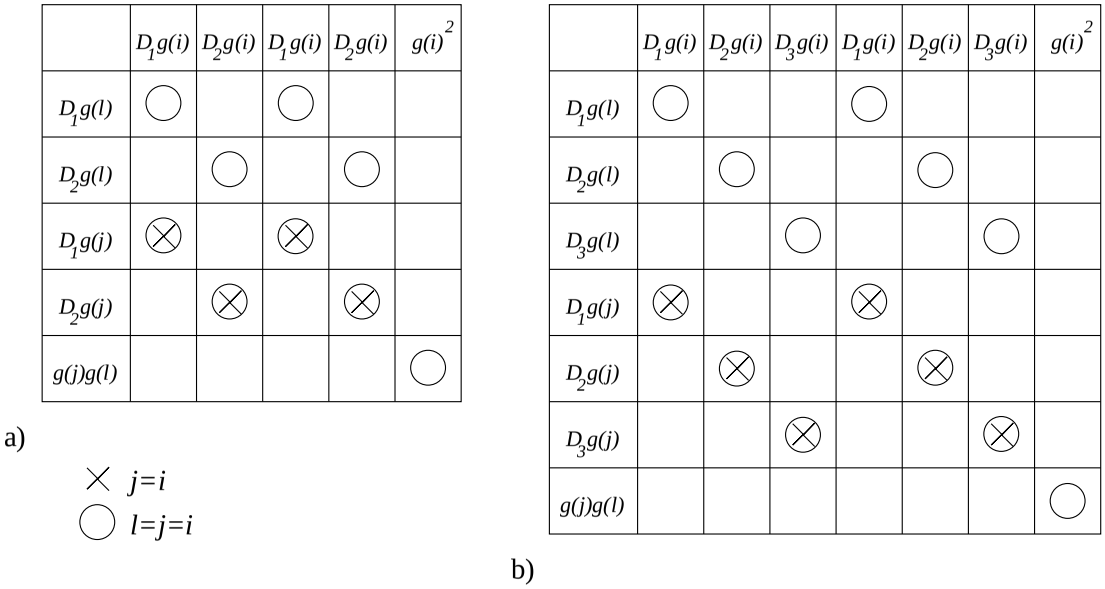


Figure B.10: The graphical representation for the noise terms for a) two users and b) three users.

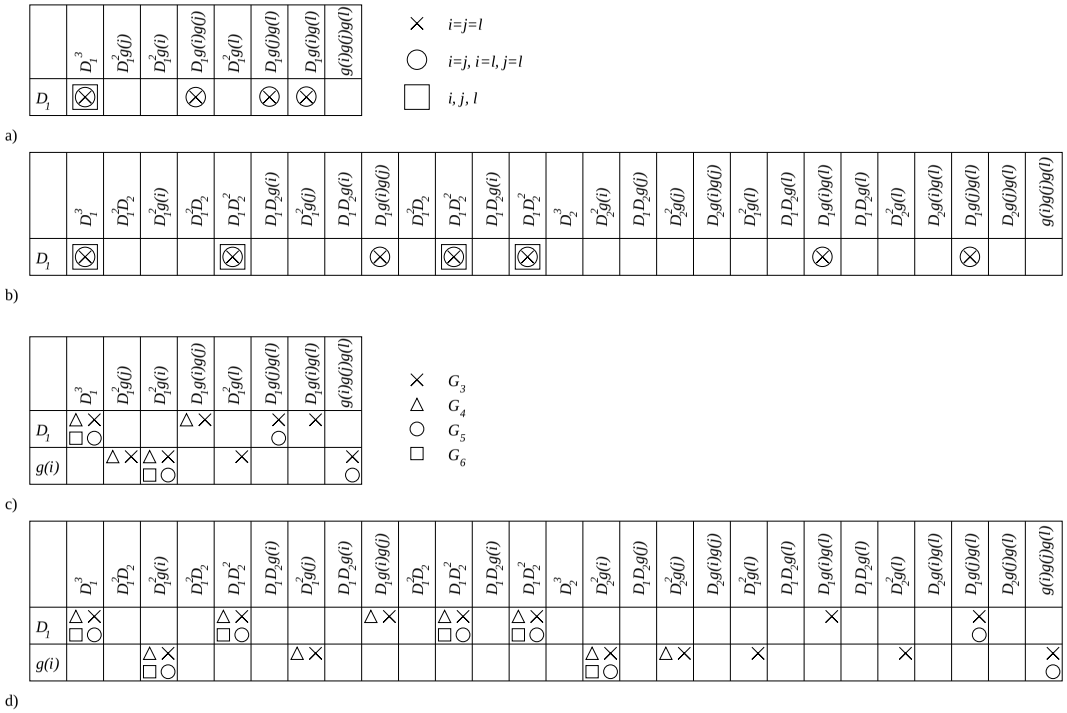


Figure B.11: a) The crosscorrelation for a) one user and b) two users. The terms of the linear-cubic products for c) one user and d) two users.

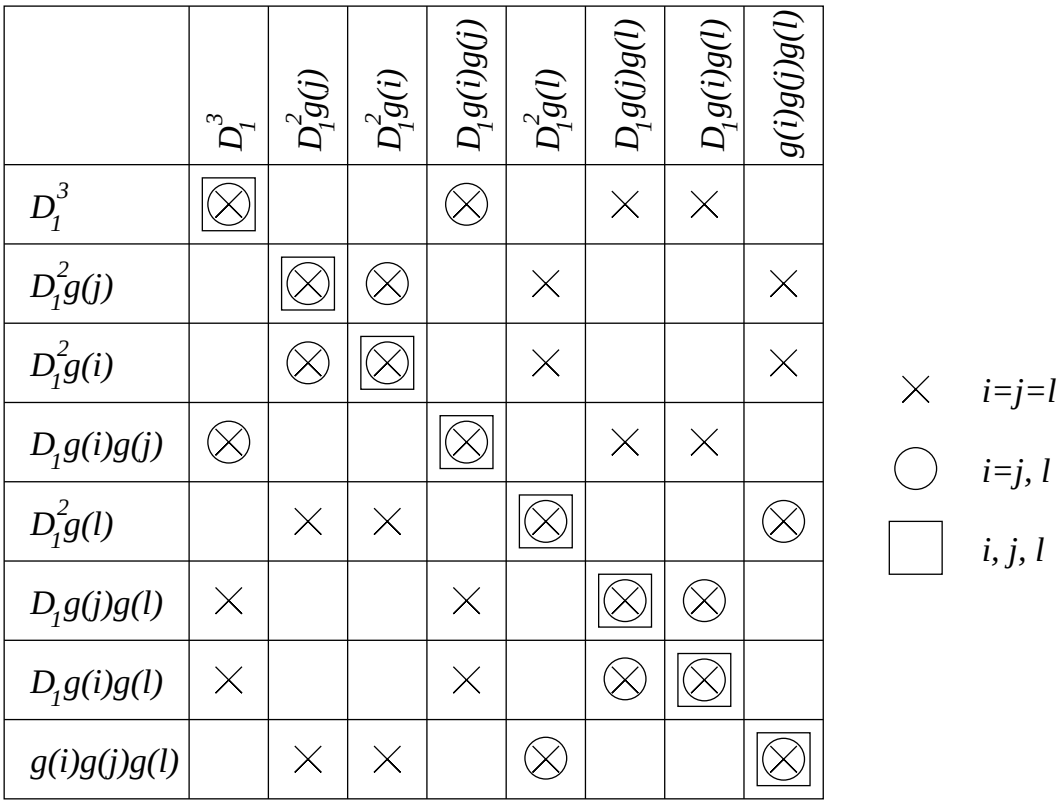
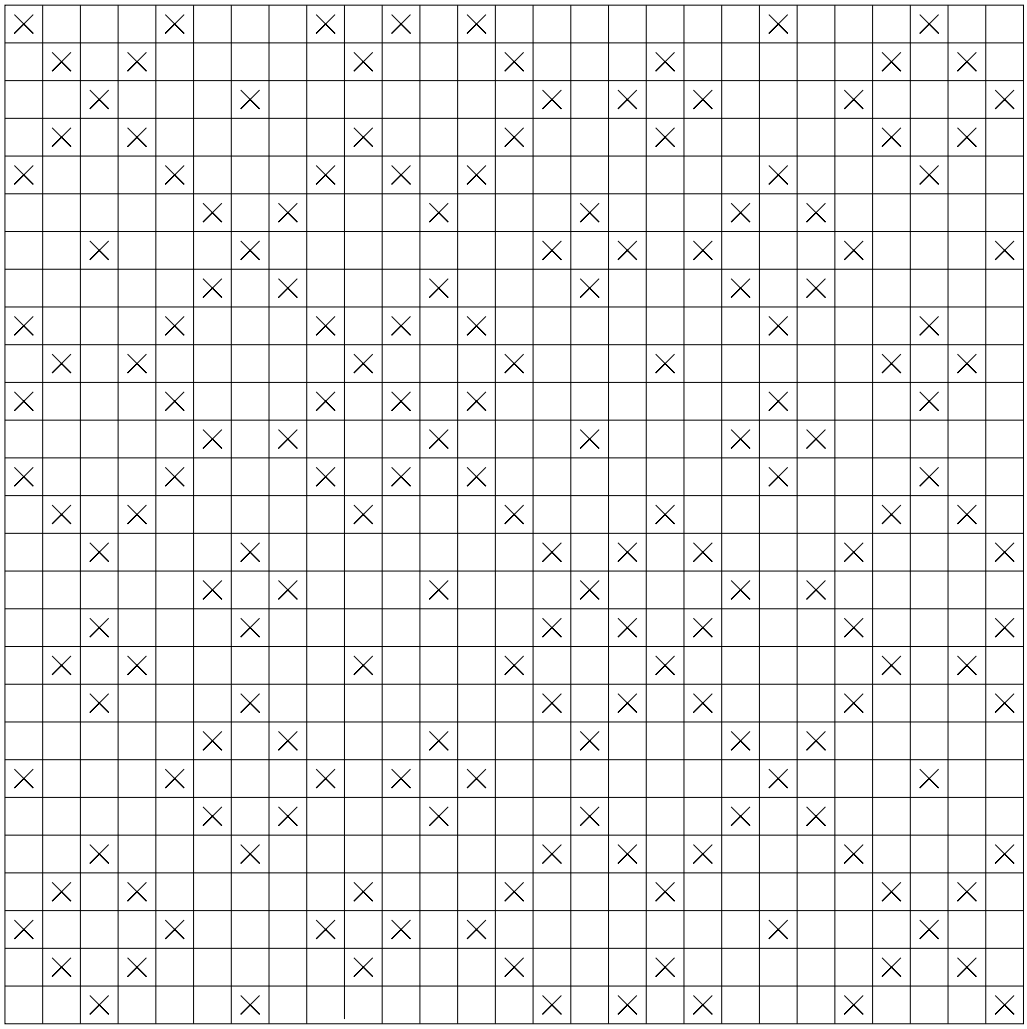


Figure B.12: The true terms for a 3rd-order Volterra system with one user.

	D_1^3	$D_1^2D_2$	$D_1^2D_2$	$D_1D_2^2$	$D_1^2D_2$	$D_1D_2^2$	$D_1D_2^2$	D_2^3
D_1^3	×			×		×	×	
$D_1^2D_2$		×	×		×			×
$D_1^2D_2$		×	×		×			×
$D_1D_2^2$	×			×		×	×	
$D_1^2D_2$		×	×		×			×
$D_1D_2^2$	×			×		×	×	
$D_1D_2^2$	×			×		×	×	
D_2^3		×	×		×			×

a)



b)

Figure B.13: The graphical representation for the noise free scenario for a) two users and b) three users. The labelling in (b) is omitted because of the space but is simple to derive from (a).

Appendix C
DS-CDMA scenarios and their points

This appendix presents the construction and the resulting points for different DS-CDMA scenarios. The expected signals (points) for a two user scenario is given for different channel models.

It is assumed that the matrix $\mathbf{P}$ is a $(M \times N)$ matrix, with $M = 2^U$ for the dispersive free AWGN channel ($L = 1$), where each point should be in the N-dimensional space, as defined in equation (2.10) is available. This matrix has M rows, each row represents one point in the U-dimensional space. In order to compute all M preprocessed points, each row in $\mathbf{P}$ is fed into a preprocessing stage and the output is stored in matrix $\mathbf{R}$ as defined in equation (2.13), where each row is a point in the U-dimensional space. The preprocessing for the non-dispersive (AWGN) channel is conducted by a bank of matched filters, illustrated in Figure C.1. The mth row of $\mathbf{P}$ is fed into the preprocessing stage and the output vector $\mathbf{r}(m)$ is stored in $\mathbf{R}$.

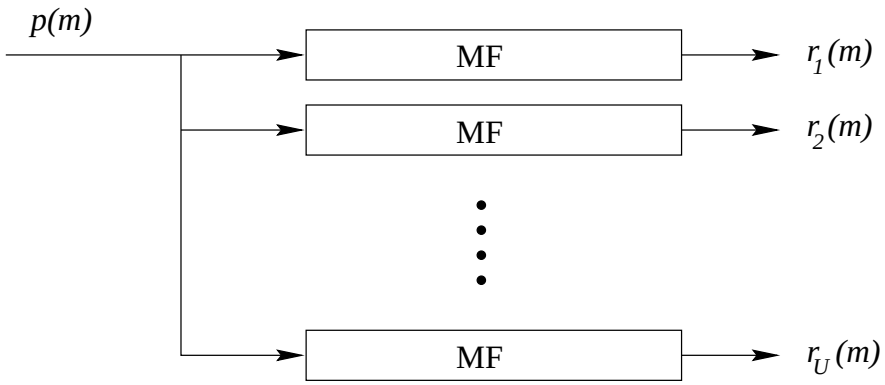


Figure C.1: The preprocessing stage used for dispersive free AWGN channels.

In multipath scenarios, the bank of MFs is replaced by a bank of RAKEs. Figure C.2 presents a single RAKE[1] structure used to resolve the multipath components. Since real data has been considered, the output of the lth RAKE finger is weighted by its corresponding lth channel

[1]Note that the RAKE presented here is not necessarily equivalent to the RAKE used in IS-95 systems. Here it is assumed that the channel coefficients are adjacent, whereas the IS-95 RAKE takes the three strongest components which do not have to be adjacent.

coefficient ($H_{ch} = [h_1\ h_2 \ldots h_L]$). This RAKE structure corresponds to the MRC receiver, and if all weights are equal then the RAKE corresponds to the EGC receiver.

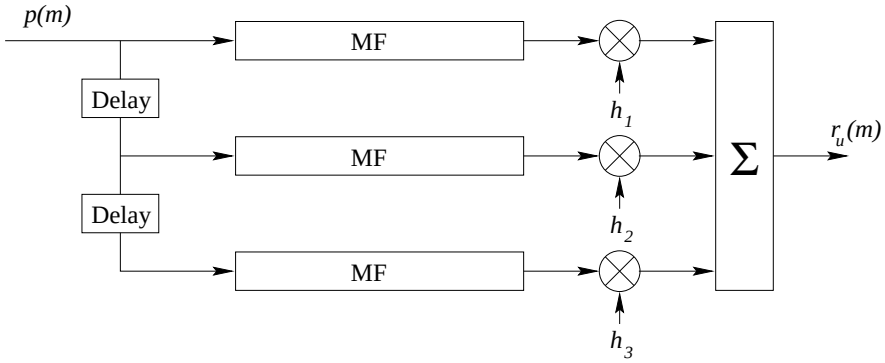


Figure C.2: A three finger RAKE. If all coefficients h_l are unity, the receiver structure corresponds to an EGC. If the coefficients correspond to the channel impulse response, the the RAKE is termed MRC.

The resulting points for three different preprocessing stages are presented. Each series of results presents the points for different channel coefficients and spreading codes. The left columns shows the resulting points for seven chip randomly generated spreading codes. The right column presents results obtained from seven chip Gold codes. The rows represent the chosen channel model (from the top to the bottom), where the normalised impulse response of these channels has been used.

Row 1 Minimum phase with $H_{ch}(z) = 1.0 + 0.666z^{-1} + 0.333z^{-2}$

Row 2 Mixed phase with $H_{ch}(z) = 0.407 + 0.815z^{-1} + 0.407z^{-2}$

Row 3 Mixed phase with $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$

Row 4 Maximum phase with $H_{ch}(z) = 0.3333 + 0.6667z^{-1} + z^{-2}$

The spreading codes are listed in appendix D.

Figure C.3 presents the points obtained with a three finger MRC preprocessing stage. In Figure C.4, a three finger EGC preprocessing stage has been applied, with equal filter coefficients, set to unity. Figure C.5 shows the points when a MF is taken to preprocess the received signals. The points form clusters where the right clusters belong to the class representing $+D_1$ and the left clusters represent $-D_1$. It can be seen that MRC and EGC preprocessing creates four clear clusters, whereas MF preprocessing creates clusters which are more in a line. Moreover, in some cases the clusters tend to overlap each other.

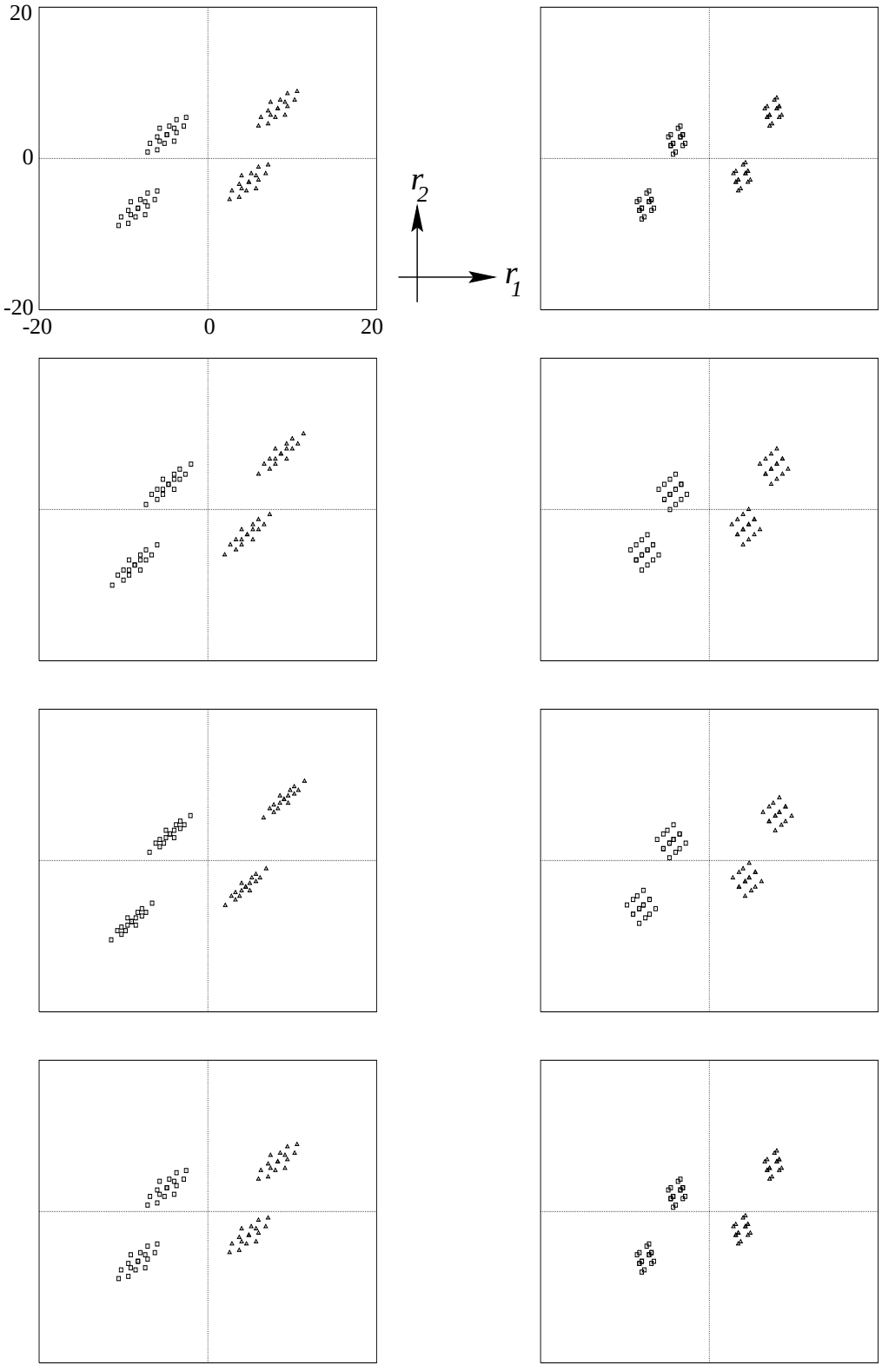


Figure C.3: The patterns of the received signal when the preprocessor consists of a bank of MRC receivers. The rows correspond to four different multipath channels, and the two columns correspond to two different spreading codes with $N = 7$.

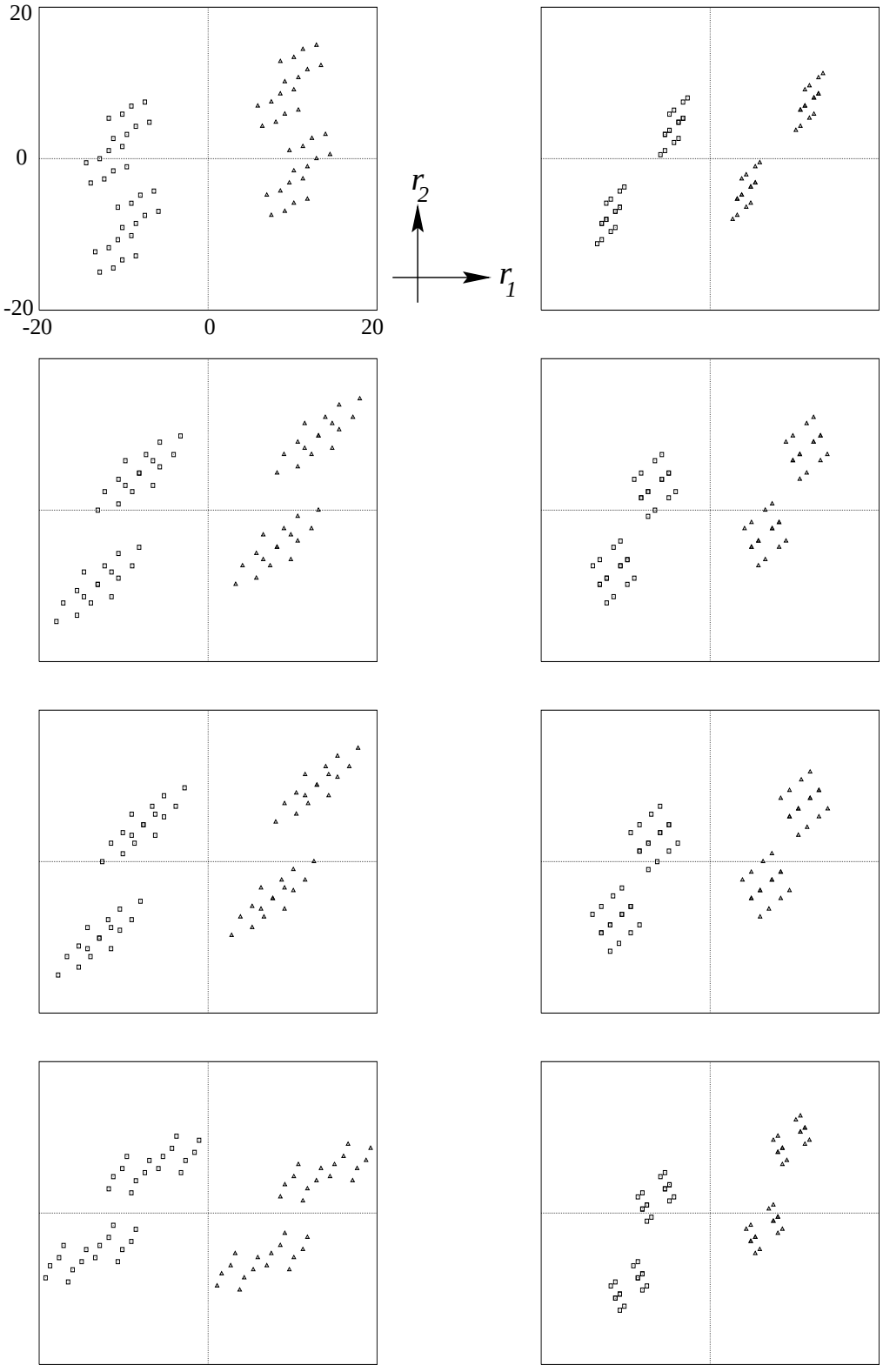


Figure C.4: The patterns of the received signal when the preprocessor consists of a bank of EGC receivers. The rows correspond to four different multipath channels, and the two columns correspond to two different spreading codes with $N = 7$.

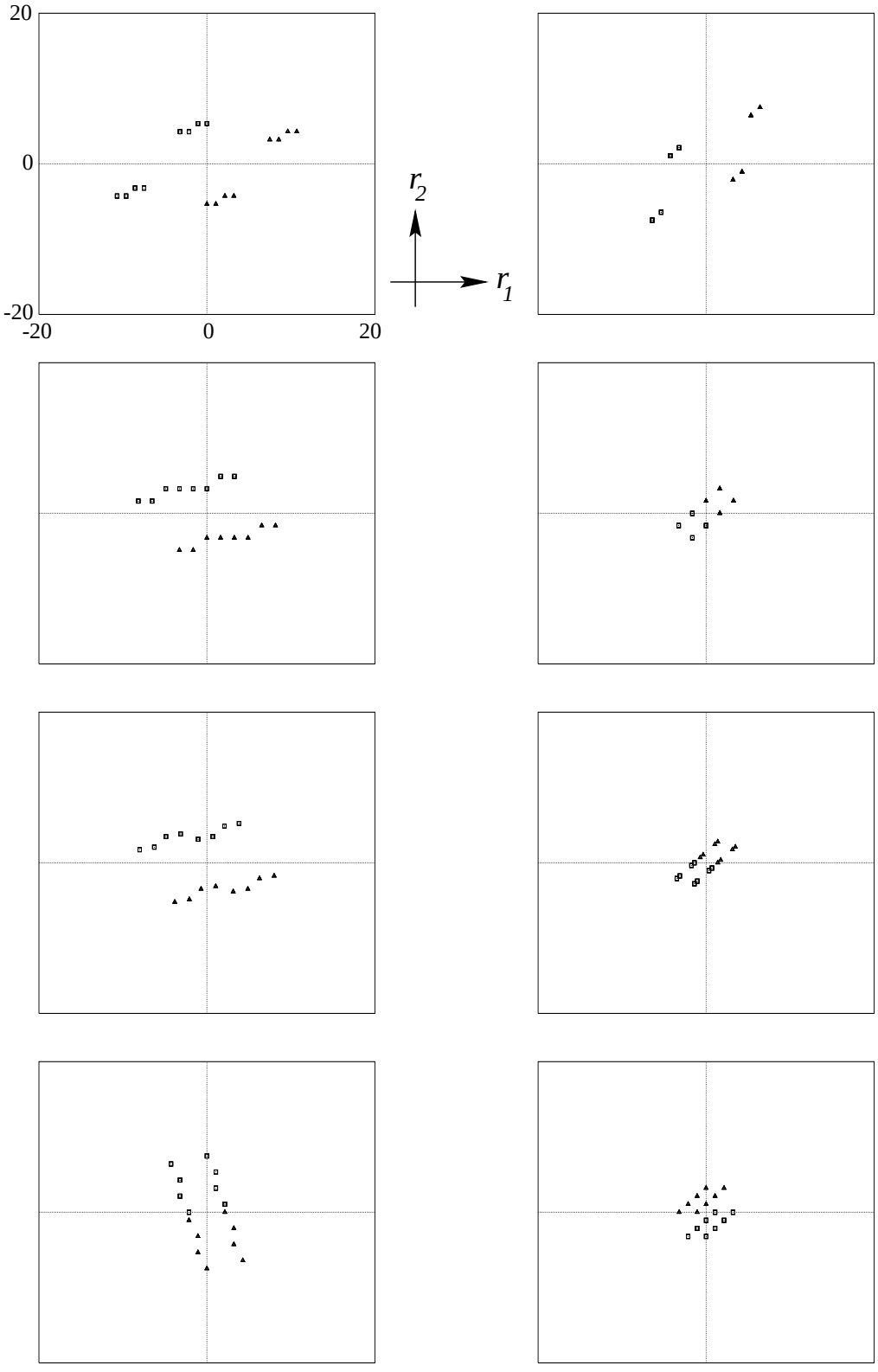


Figure C.5: The patterns of the received signal when the preprocessor consists of a bank of MF. The rows correspond to four different multipath channels, and the two columns correspond to two different spreading codes with $N = 7$.

Appendix D
Simulation parameters

This appendix lists the parameters used in the simulation results presented in all chapters.

The randomly generated seven chip spreading code $\mathbf{c}_u$ for user u:

$$\begin{aligned}
\mathbf{c}_1 &= [-1\ -1\ 1\ -1\ -1\ -1\ 1]^T \\
\mathbf{c}_2 &= [-1\ 1\ 1\ -1\ -1\ 1\ 1]^T \\
\mathbf{c}_3 &= [-1\ 1\ 1\ 1\ 1\ 1\ -1]^T \\
\mathbf{c}_4 &= [-1\ -1\ 1\ 1\ -1\ 1\ 1]^T \\
\mathbf{c}_5 &= [-1\ -1\ -1\ 1\ 1\ -1\ 1]^T \\
\mathbf{c}_6 &= [1\ 1\ 1\ -1\ 1\ 1\ -1]^T \\
\mathbf{c}_7 &= [1\ -1\ 1\ 1\ 1\ -1\ 1]^T
\end{aligned}$$

The seven chip Gold spreading code $\mathbf{c}_u$ for user u:

$$\begin{aligned}
\mathbf{c}_1 &= [-1\ 1\ 1\ -1\ -1\ -1\ 1]^T \\
\mathbf{c}_2 &= [-1\ 1\ 1\ -1\ 1\ -1\ -1]^T \\
\mathbf{c}_3 &= [1\ 1\ 1\ 1\ -1\ 1\ -1]^T \\
\mathbf{c}_4 &= [-1\ 1\ -1\ -1\ 1\ 1\ -1]^T \\
\mathbf{c}_5 &= [-1\ -1\ 1\ 1\ 1\ 1\ 1]^T \\
\mathbf{c}_6 &= [1\ 1\ -1\ 1\ 1\ -1\ 1]^T \\
\mathbf{c}_7 &= [-1\ -1\ -1\ 1\ -1\ -1\ -1]^T
\end{aligned}$$

The randomly generated 16 chip spreading code $\mathbf{c}_u$ for user u:

$$
\begin{aligned}
\mathbf{c}_1 &= [-1\ 1\ -1\ 1\ 1\ -1\ -1\ -1\ 1\ 1\ -1\ 1\ -1\ -1\ -1\ 1\ 1\ -1]^T \\
\mathbf{c}_2 &= [-1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ -1\ -1\ 1\ 1\ 1\ -1\ -1]^T \\
\mathbf{c}_3 &= [-1\ -1\ -1\ -1\ -1\ -1\ -1\ -1\ -1\ -1\ -1\ 1\ 1\ -1\ -1\ 1\ 1\ 1]^T \\
\mathbf{c}_4 &= [1\ 1\ -1\ 1\ -1\ 1\ 1\ -1\ -1\ -1\ 1\ 1\ 1\ -1\ -1\ -1\ 1\ 1\ 1]^T \\
\mathbf{c}_5 &= [1\ 1\ 1\ 1\ -1\ -1\ 1\ 1\ -1\ 1\ 1\ -1\ -1\ -1\ 1\ 1\ -1\ 1\ 1]^T \\
\mathbf{c}_6 &= [-1\ -1\ 1\ 1\ 1\ -1\ -1\ 1\ 1\ -1\ -1\ -1\ -1\ -1\ 1\ 1\ 1\ -1]^T \\
\mathbf{c}_7 &= [-1\ -1\ -1\ -1\ 1\ 1\ 1\ 1\ 1\ 1\ -1\ 1\ 1\ 1\ 1\ 1\ 1\ -1]^T \\
\mathbf{c}_8 &= [1\ -1\ 1\ 1\ 1\ 1\ 1\ -1\ 1\ 1\ -1\ 1\ 1\ -1\ -1\ 1\ 1\ 1]^T
\end{aligned}
$$

Different channel impulse responses for the stationary multipath channel:

$$
\begin{aligned}
\text{model2} &\equiv H_{ch}(z) = 0.333 + 0.666z^{-1} + z^{-2} \\
\text{model3} &\equiv H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2} \\
\text{model10} &\equiv H_{ch}(z) = 0.407 + 0.815z^{-1} + 0.407z^{-2} \\
\text{model12} &\equiv H_{ch}(z) = 1.0 + 0.666z^{-1} + 0.333z^{-2} \\
\text{modelz} &\equiv H_{ch}(z) = 0.25 + 0.5z^{-1} + z^{-2}
\end{aligned}
$$

Note, during the simulation, the H_{ch} was normalised to unity, e.g. $h_1^2 + h_2^2 + h_3^2 = 1$.

Appendix E
Publications

List of publications:

- R. Tanner, D. G. M. Cruickshank;
  “Nonlinear Volterra Filter Receiver for DS-CDMA”,
  Proceedings 4th International Conference on Mathematics in Signal Processing, IMA, University of Warwick, UK, IEE, December 1996.

- R. Tanner, D. G. M. Cruickshank;
  ”Volterra Based Receivers for DS-CDMA”,
  Proceedings International Symposium on Personal, Indoor and Mobile Radio Comminications, Helsinki, Finland, IEEE, September 1997.

- R. Tanner, D. G. M. Cruickshank, C. Z. W. Hassell Sweatman, B. Mulgrew;
  ”Receivers for nonlinearly separable scenarios in DS-CDMA”,
  IEE Electronics Letters, 33, pp2103-2105, December 1997.

- R. Tanner, D. G. M. Cruickshank;
  ”RBF Based Receivers for DS-CDMA with Reduced Complexity”,
  Accepted for International Symposium on Spread Spectrum Techniques and Applications, Sun City, South Africa, IEEE, 1998.

Volterra Filter Structure for DS-CDMA

R. Tanner D. G. M. Cruickshank

Department of Electrical Engineering, Signals & Systems Group,
The University of Edinburgh, Edinburgh, UK

Abstract

This paper presents a receiver filter, based on *Volterra* functions, for a direct-sequence code-division-multiple-access (DS-CDMA) system. The envisaged application would be cellular telephony, where the receiver will be required to be adaptive due to changes to the channel characteristics. The results obtained from the *Volterra* filter are compared against the linear minimum-mean-square-error (MMSE) filter and the nonlinear radial-basis-function (RBF) network for an additive-white-*Gaussian*-noise (AWGN) channel.

1 Introduction

The growing demand for mobile communications will reach the capacity limits of existing systems soon. Hence, only the introduction of new concepts and systems will increase data transmission capacities since the frequency spectrum is limited. With the introduction of spread-spectrum (SS) technology and the availability of cheap microprocessor technology (especially digital-signal-processors (DSP's)) we can exploit the SS technique with DS-CDMA to increase the communications system capacity.

To overcome multiuser interference (MAI) in CDMA, many receivers have been proposed in literature. Most detectors for multiuser purpose are linear structures. Traditional approaches are based on a bank of matched filters, with a RAKE structure and/or cancellation. Approaches using the nonlinear radial-basis-function network [7, 6] show excellent performance. Since it has similar structure to the maximum-likelihood (ML) receiver, its complexity also grows exponentially and does not suit mobile terminals such as phones. However, there may be a nonlinear filter which is a good compromise between complexity and good performance.

This paper is structured as follows, first a brief introduction to the DS-CDMA communication system is described, followed by a survey of receiver structures (filters). Next we present a description of the *Volterra* system and its filter structure. This is followed by simulation results and finally, we end the paper with conclusions.

2 The DS-CDMA System

The use of DS-CDMA for cellular purposes is mainly for capacity reasons. The basic idea of a DS-CDMA multiuser system is that all users are transmitting at the same time and share the same frequency bandwidth.

In Figure 1 we can see a downlink (base station to mobile station) multiuser DS-CDMA system for U users. To separate the information the desired user is transmitting, we allocate for every user a unique spread-code, and use this code like a key at the receiver to extract the desired information from the received signal. So in this system, each user has a unique spread-code

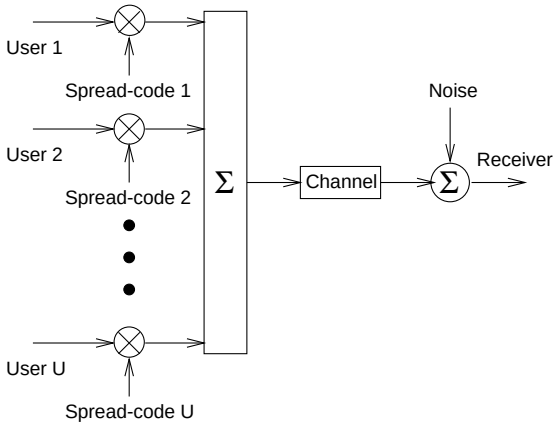


Figure 1: *A U-user downlink DS-CDMA system.*

sequence of length N, which is randomly generated. The spread-code symbols (chips) are either $+1$ or -1. The transmitted user data bits D_u are randomly generated with zero mean and equal probability of being either $+1$ or -1.
We now consider our DS-CDMA system as follows.
There are $1, 2, \ldots, U$ users, and $0, 1, \ldots, (N-1)$ chips per spread-code, the spread-code of user u is $[C_{u,0}, C_{u,1}, \ldots, C_{u,(N-1)}]$, and finally, the user data bits are $D_1, D_2, \ldots, D_U$. Following this definition, we can derive the received sequence for user bit k of a DS-CDMA system, where $\underline{y}$ is the received signal of length N.

$$y(kN+n) = \sum_{u=1}^{U} D_u C_{u,n} + g(kN+n) \qquad n = 0, 1, \ldots, (N-1) \tag{1}$$

$g(n)$ denotes the thermal noise component. Assuming an additive-white-*Gaussian*-noise (AWGN) channel, we add to every chip a noise component $g(n)$ only.

The mobile environment, which we take into account by using a channel model, is certainly different from the AWGN model. The major distortions are:

Multipath effects due to the different paths. This destroys orthogonality between spread-codes.

Multiuser interference (MAI) is when the receiver also receives transmission signals which do not belong to the desired user. This interference is structured and can be suppressed.

The mobility of the user and the time varying channel makes these interference components non-stationary, hence the filter has to be adaptive.

3 Other Filter Structures

The simplest receiver is the Matched-Filter (MF) or correlation receiver, but, this filter has the disadvantage of poor performance in multiuser interference.

Another linear filter is a first-order *Volterra* system [1] representing the minimum-mean-square-error (MMSE) filter, or better known as the *Wiener* filter [2].
The MMSE filter is also an N-tap FIR filter and the optimal set of filter coefficients $\underline{c}_{opt}$ is given (in vector form) by the $Wiener - Hopf$ equation:

$$\underline{c} = \Phi_{yy}^{-1} \underline{\phi}_{xy} \tag{2}$$

Where Φ_{yy} is the autocorrelation matrix of size $(N \times N)$ of the input signal and for a stationary input is:

$$\Phi_{yy} = E\left[\begin{array}{cccc} y^2(n) & y(n)y(n-1) & .. & y(n)y(n-N+1) \\ y(n)y(n-1) & y^2(n) & .. & y(n)y(n-N+2) \\ y(n)y(n-2) & y(n)y(n-1) & .. & y(n)y(n-N+3) \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & .. & \vdots \\ y(n)y(n-N+1) & y(n)y(n-N+2) & .. & y^2(n) \end{array}\right] \tag{3}$$

$E[y]$ denotes the expected value of y. $\underline{\phi}_{xy}$ is the crosscorrelation vector between the desired user data bit D_u and the received sequence $\underline{y}$. In this case, $\underline{\phi}_{xy}$ is the spread-code for the desired user. Assuming that the data streams among the different users are independent, i.e. $E[D_a D_b] = 0$ for $a \neq b$, we can say:

$$\Phi_{yy} = \phi P \phi^T + \sigma^2 I \tag{4}$$

ϕ is the $(N \times U)$ code matrix for all users. P is a $(U \times U)$ matrix and has the power P_u at the receiver along its leading diagonal of user u. σ^2 is the noise power and I the identity matrix. The minimum-mean-squared-error is then $\varepsilon_{min} = E[x^2(i)] - \underline{c}_{opt}^T \underline{\phi}_{xy}$.

The MMSE filter also has its limitations when we introduce multiuser interference [5, 6]. To overcome these bounds, we suggest using a nonlinear filter.

The RBF network is a nonlinear filter, also referred to as neural network (NN) [7], because it has a NN structure with centres and weights (coefficients). The RBF nonlinearity is given by the *Gaussian*-shape function
$f(x) = exp(-x^2/(2\sigma^2))$, because of the AWGN channel model. Since we know the number of users and the spread-code of every user, we are able to calculate the optimum centres and weights. First, we compute the number of centres L, which for an AWGN channel is $L = 2^U$. Then we compute all centres, which are given by all possible combinations of the noise free transmitted waveforms among the spread-codes. Our complete RBF function is then:

$$r(y) = \sum_{i=0}^{L-1} w_i \exp\left(-\frac{(\|\underline{y} - \underline{c}_i\|^2)}{2\sigma^2}\right) \tag{5}$$

w_i are the linear weights and $(\|\cdot\|)$ denotes the *Euclidean* norm where $\underline{y}$ is the input data vector and $\underline{c}_i$ are the centre vectors. Again, σ^2 is the noise power.

4 The *Volterra* Filter

The motivation for using a *Volterra* function based filter is the ability to cope with MAI better than the MMSE does, but also to have less computation than with the RBF network. The *Volterra* representation for nonlinear systems [1, 8] is based on the *Volterra* series functional representation. Our interest is in having a fast adaptive filter system with little computation that can cope with MAI.

As a compromise we use a truncated *Volterra* series of order three. It is explained later why the second-order system has not been taken into account. Equation (6) presents the truncated and modified discrete *Volterra* expansion form from which we derived the DS-CDMA applied form, h_i are the *Volterra* coefficients for order i.

$$y(n) = \sum_{a=0}^{N-1} h_1(a)x(n-a) + \sum_{a=0}^{N-1}\sum_{b=a}^{N-1}\sum_{c=b}^{N-1} h_3(a,b,c)x(n-a)x(n-b)x(n-c) \tag{6}$$

The first sum of (6) represents the linear term or first-order term, followed by the third-order term. The *Volterra* expansion is a vector $\underline{v} = V(\underline{y}) = [v_0, v_1, \ldots, v_{N-1}, v_N, \ldots, v_{M-1}]^T$, where M is the total length of the *Volterra* expansion, the expansion size. This expansion is shown in Figure 2 for a third-order system. The received vector $\underline{y}$ of size N is expanded after (6) into a

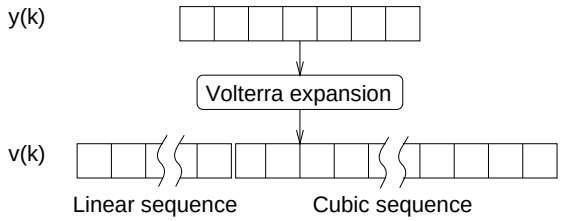


Figure 2: *Example of a Volterra expansion for a third-order system.*

vector $\underline{v}$ (expansion sequence) of size $M = M_1 + M_3$, $m = 0, 1, \ldots, (M-1)$ so, $\underline{v}$ defined in (7) is a vector which has first the linear part, then the cubic part.

$$\underline{v} = [\left\{\sum_{u=1}^{U} D_u C_{u,0} + g(0)\right\}, \ldots, \left\{\sum_{u=1}^{U} D_u C_{u,(N-1)} + g(N-1)\right\}, \left\{\sum_{u=1}^{U} D_u C_{u,0} + g(0)\right\}^3, \left\{\sum_{u=1}^{U} D_u C_{u,0} + g(0)\right\}^2 \times \left\{\sum_{u=1}^{U} D_u C_{u,1} + g(1)\right\}, \ldots, \left\{\sum_{u=1}^{U} D_u C_{u,N-1} + g(N-1)\right\}^3]^T \quad (7)$$

The expansion size $M(N, O)$ can be determined by the binomial coefficient for system-order O and spread-code length N as: $M_{order} = \binom{N+order-1}{order}$.

	Order		
N	1	3	5
7	7	84	462
9	9	165	1287

Table 1. *M(N,O) for two selected spread-code lengths.*

The calculation of the coefficients $\underline{c}$ is derived from the $Wiener - Hopf$ form $\underline{c} = \Phi_{vv}^{-1} \underline{\phi}_{xv}$, where $\underline{c}$ is the coefficient vector, Φ_{vv} is the autocorrelation matrix of the expanded sequence and $\underline{\phi}_{xv}$ is the crosscorrelation vector between the desired user bit D_u and the expanded sequence $\underline{v}$.

4.1 Second-order System

We anticipated that a second-order *Volterra* does not work. The reason for this can be seen from the derivation of the crosscorrelation vector $\underline{\phi}_{xv}$. We assume that the user bit D_u has equal probability of being ± 1, hence the expected value of products with odd power of D_u are zero and with even power are one. Further, we assume without loss of generality, since noise and user bits

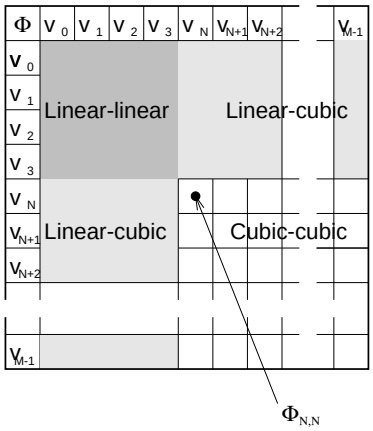


Figure 3: *The autocorrelation matrix and a term of interest.*

are uncorrelated to each other, there is no noise ($g(n) = 0$) and one user is active ($U = 1$).

$$\begin{aligned} \underline{\phi}_v &= E[D_u \times \underline{v}] \\ &= E[D_1[D_1C_{1,0}, \ldots, D_1C_{1,N-1}, D_1^2C_{1,N}^2, \ldots, D_1^2C_{1,(N-1)}^2]^T] \\ &= E[[D_1^2C_{1,0}, \ldots, D_1^2C_{1,(N-1)}]^T] \\ &= [C_{1,0}, C_{1,1}, \ldots, C_{1,(N-1)}]^T \end{aligned} \tag{8}$$

It can be seen, that the solution is the linear solution, since all odd power products are zero. The introduction of the noise term $g(n)$ does not change the outcome. So we can conclude, a second-order *Volterra* system produces the first-order (MMSE) solution.

4.2 Third-order System

The autocorrelation matrix for a *Volterrra* system can be calculated from vector (7). One of the N linear terms is given in (9) and the first cubic term is (10).

$$v_n = y_n = \sum_{u=1}^{U} D_u C_{u,n} + g(n) = D_1C_{1,n} + D_2C_{2,n} + \ldots + D_UC_{U,n} + g(n) \tag{9}$$

$$v_N = {y_0}^3 = \left\{ \sum_{u=1}^{U} D_u C_{u,0} + g(0) \right\}^3 = \{D_1C_{1,0} + D_2C_{2,0} + \ldots + D_UC_{U,0} + g(0)\}^3 \tag{10}$$

It can be seen, that v_m is spread-code, user bit and noise dependent. In the autocorrelation matrix, we have products of linear-linear, linear-cubic and cubic-cubic terms, as shown in Figure 3. The linear-linear part is known from the MMSE. For the other two parts, we derive them, based on the properties of D_u and $g(n)$. So for instance, the first cubic-cubic autocorrelation term, see Figure 3, is:

$$\Phi_{N,N} = v_N v_N = E[(\phi_{N,N})^2] = E[\left\{ \sum_{u=1}^{U} D_u C_{u,0} + g(0) \right\}^6] \tag{11}$$

Taking the properties of D_u and $g(n)$ into account, we can only have products with even power for D_u's and $g(n)$'s, we call them here true terms. For a single user system with the noise component, the true term of $\Phi_{N,N}$ is:

$$\Phi_{N,N} = D_1^6 + 15C_0^4 D_1^4 g(0)^2 + 15C_0^2 D_1^2 g(0)^4 + g(0)^6 \tag{12}$$

Obviously, the difficult task is in finding these true terms and to do this, we need to sum to obtain the autocorrelation term. It also can be seen that the number of terms in (11) grows exponentially with the number of users and finding these true terms becomes tedious to calculate. Alternative methods are described in the next section.

The derivation of the noise-power is based on the moments theory of statistics [3, 4]. The noise characteristics are *Gaussian*, with zero mean and variance $\sigma^2 = 1$. The true terms we get have noise $g(n)$ with exponent 2, 4 and 6. The expected value for the *Gaussian* distributed random variable x and its higher even moments are [4]:

$$E[x^2] = \int_{-\infty}^{\infty} x^2 f(x)dx = \sigma^2 \tag{13}$$

$$E[x^4] = \int_{-\infty}^{\infty} x^4 f(x)dx = 3\{E[x^2]\}^2 \tag{14}$$

$$E[x^6] = \int_{-\infty}^{\infty} x^6 f(x)dx = 15\{E[x^2]\}^3 \tag{15}$$

Due to this, equation (12) becomes:

$$\Phi_{N,N} = 1 + 15C_0\sigma^2 + 45C_0(\sigma^2)^2 + 15(\sigma^2)^3 \tag{16}$$

The autocorrelation matrix could therefore be determined, as above, if we calculate all true terms. This is also valid for the crosscorrelation vector, the expected value between the desired data bit and the desired *Volterra* expanded sequence $\underline{v}$.

5 Simulation Results

To find the filter coefficients without calculating the autocorrelation matrix explicitly, we can use either an estimation technique or an adaptive algorithm [2]. Our adaptive system is based on the LMS algorithm. The other technique makes an estimate of the autocorrelation matrix and the crosscorrelation vector. To get a good estimate, we have to average them over many times. To find the noise free auto- and crosscorrelation, we must average over all possible combinations of data bits. But we also have to take the noise power into account. To get the noise power we add the noise onto the chips and proceed in the same way as before, averaging the auto- and crosscorrelation over many trials. For good estimations, we have to average over a long time, which makes this technique unsuitable for practical applications, but enables us to verify our LMS results.

The described receiver structures were simulated using Monte-Carlo simulations. The graphs show the bit-error-ratio (BER) or probability of error for desired user one. Figure 4 and figure 5 show the performance of these receivers for a noise power of $E_b/N_0 = 9$dB and spread-code length $N = 7$, E_b is the normalised bit energy to N_0 the noise power spectral density.

In Figure 4 the coefficients for the *Volterra* filter (VS_LMS) are estimated with the LMS approach. The performance is compared against the other filters. The VS_LMS performance is better than the MMSE performance. The performance for matched filtering becomes poor as the MAI increases. The RBF receiver is the maximum-likelihood receiver in this AWGN channel and has the best performance. Since the separation problem, extracting the information out of the

received signal, is nonlinear, nonlinear filters outperform linear filters. Our proposed filter has a nonlinear component and should have a performance between the linear MMSE and the nonlinear RBF filter, as shown in the graphs.
The other simulation, Figure 5, shows the performance of the *Volterra* filter (VS_EST) when the coefficients are computed using the estimation technique. This approach leads to better results because the coefficients are more accurate, since the training length for the LMS approach was limited for time reasons.

However, both simulations indicate, that the proposed *Volterra* filter has better performance than the linear filters.

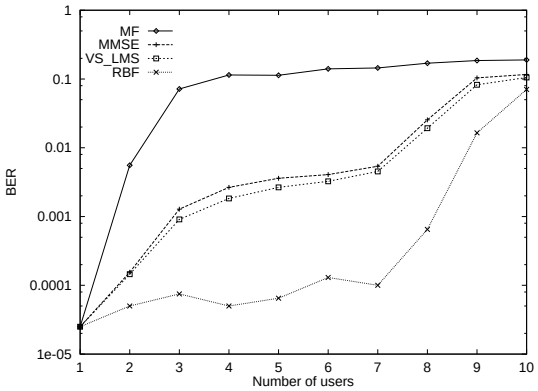


Figure 4: *BER against the number of users in AWGN channel, Volterra filter coefficients are estimated with the LMS algorithm.*

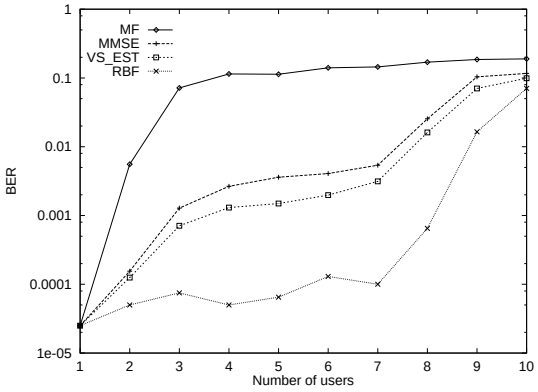


Figure 5: *BER against the number of users in AWGN channel, Volterra filter coefficients were calculated with the estimation technique.*

6 Conclusions

We derived a *Volterra* structure based filter and compared its BER performance on an AWGN channel against other receivers. The results we obtained from Monte-Carlo simulation show for a fixed number of users, that the BER for the new DS-CDMA filter can be 30% better than for the

MMSE. Since the truncated polynomial of the *Volterra* structure is nonlinear, the filter might perform even better in a multipath environment.
So, future work will investigate this receiver filter for multipath channels. Furthermore, we want to be able to calculate the filter coefficients without using the estimation technique or an adaptive algorithm when we know (or can estimate) the channel impulse response.

References

[1] M. Schetzen. (1980). *The Volterra and Wiener theories of nonlinear systems* (1st edn). John Wiley and Sons Inc, New York.

[2] S. Haykin. (1996) *Adaptive Filter Theory* (3rd edn). Prentice Hall International, Englewood Cliffs, New Jersey, USA.

[3] A. Papoulis. (1991) *Probability, Random Variables, and Stochastic Processes* (3rd edn). McGraw Hill, Singapore.

[4] S. M. Kay. (1993) *Fundamentals of Statistical Signal Processing: Estimation Theory* (1st edn). Prentice Hall International, Englewood Cliffs, New Jersey, USA.

[5] D. G. M. Cruickshank. (1994). Optimal and Adaptive FIR Filter Receivers for DS-CDMA. *International Symposium on Personal Indoor and Mobile Communications PIMRC*, 1339–43.

[6] D. G. M. Cruickshank. (1996). Radial basis function receivers for DS-CDMA. *IEE Electronics Letters*, *32*, 188–90.

[7] U. Mitra and H. V. Poor. (1994). Neural Network Techniques for Adaptive Multiuser Demodulation. *IEEE Transactions on Selected Areas in Communications*, *12*, 1460–70.

[8] M. Schetzen. (1981). Nonlinear System Modeling Based on the Wiener Theory. *Proceedings of the IEEE*, *69*, 1557—73.

Volterra Based Receivers for DS-CDMA

R. Tanner D. G. M. Cruickshank

The University of Edinburgh,
Department of Electrical Engineering,
Signals & Systems Group
The King's Buildings, Mayfield Road
Edinburgh, EH9 3JL, UK
e-mail: rt@ee.ed.ac.uk

ABSTRACT

This paper presents a *Volterra* function based filter as the receiver for a direct-sequence-code-division-multiplex-access (DS-CDMA) system. The envisaged application would be cellular telephony, where the receiver will be required to be adaptive due to channel effects. The results obtained from the *Volterra* filter are compared against the linear minimum-mean-square-error (MMSE) filter and the nonlinear radial-basis-function (RBF) network in a stationary multipath channel. Further, the computational complexity will be investigated and compared against the MMSE.

INTRODUCTION

The demand for spread-spectrum (SS) communications and the introduction of new mobile systems, space and ground based, supports research in multi-user communications, particularly in DS-CDMA.

To overcome multiuser interference (MAI), many receivers have been proposed, see [1, 2]. Most detectors for multiuser purpose are linear structures. Traditional solutions are based on matched filtering, with a RAKE structure and/or cancellation. Another approach is a polynomial linear detector, based on a polynomial expansion, proposed in [3].

Approaches using the nonlinear radial-basis-function network [4, 5] show excellent performance. Since it has similar structure to the maximum-likelihood (ML) receiver, its complexity also grows exponentially and does not suit mobile terminals such as phones. However, there may be a nonlinear filter which is a good compromise between complexity and good performance.

This paper is structured as follows, first we introduce the DS-CDMA communication system, followed by a survey of other receivers (filters). Then, we present the *Volterra* filter. This is followed by simulation results and a conclusion.

SYSTEM CONSIDERATION

We examine a downlink DS-CDMA system with U independent active users, $1, 2, \ldots, U$. We allocate for every user a unique spreading code of length N with $0, 1, \ldots, N-1$ chips. Each user is transmitting a DS-CDMA sequence which is chip and bit synchronous, and all users are transmitting with equal power. The data bit transmitted by user u during time k will be denoted as $D(k)_u$ and is either $+1$ or -1 with equal probability. The spreading code set used is seven chips long and randomly generated, denoted as $C_{u,n}$. Following this definition, we can derive the received sequence for a DS-CDMA system and an additive-white-*Gaussian*-noise (AWGN) channel, where $\underline{y}$ is the received signal (in vector form) of length N, for chip $n = 0, 1, \ldots, N-1$.

$$y(kN+n) = \sum_{u=1}^{U} D(k)_u C_{u,n} + g(kN+n) \tag{1}$$

$g(kN+n)$ denotes the noise component we add to every chip. Now we consider a stationary multipath channel. The channel impulse response is assumed to be known, due to pilot channel estimation, as for example in IS-95, and is $H(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$. All signals pass through the same channel. So the received signal at the time of data bit k and chip n becomes:

$$\begin{aligned} y(kN+n) \;=\; & 0.3482 \times \sum_{u=1}^{U} D_u(k) C_{u,n} \\ & + 0.8704 \times \sum_{u=1}^{U} D_u(k) C_{u,n-1} \\ & + 0.3482 \times \sum_{u=1}^{U} D_u(k) C_{u,n-2} + g(kN+n) \end{aligned} \tag{2}$$

If $n-x$ is negative then the term $D_u(k)C_{u,n-x}$ is replaced by $D_u(k-1)C_{u,N+n-x}$ and if n or $n-x$ is $\geq N$ then $D_u(k)C_{u,n-x}$ is replaced by $D_u(k+1)C_{u,n-N-x}$. It shows clearly the presence of channel induced ISI.

RECEIVER STRUCTURES

First, we describe filter structures for the AWGN channel since the structures used for the multipath channel are basically the same, adapted for the channel impulse response. We assumed that the spreading codes and the signal power are known at the receiver.

The simplest receiver is the Matched-Filter (MF) or correlation receiver. It is a N-tap FIR filter whose coefficients c_i are the spreading code for the desired user, $c_i^{MF} = C_{0,i}$.

Another N-tap FIR filter is a first-order *Volterra* system representing the minimum-mean-square-error (MMSE) filter, or better known as the *Wiener* filter [6]. The optimal set of filter coefficients derived from the Wiener-Hopf equation is given (in vector form) by:

$$\underline{c}^{MMSE} = \Phi_{yy}^{-1}\underline{\phi}_{xy} \tag{3}$$

Where Φ_{yy} is the autocorrelation matrix (of size $N \times N$) of the input signal. $\underline{\phi}_{xy}$ is the crosscorrelation vector between the desired user data bit D_u and the received sequence $\underline{y}$. In this case, $\underline{\phi}_{xy}$ is the desired spread-code. Assuming that the data streams among the different users are independent, i.e. $E[D_a D_b] = 0$ for $a \neq b$, we have [7]:

$$\Phi_{yy} = \phi P \phi^T + \sigma^2 I \tag{4}$$

where ϕ is the $(N \times U)$ code-matrix for all users. P is a $(U \times U)$ matrix and has the power P_u at the receiver along its leading diagonal of user u. σ^2 is the noise power and I the identity matrix. The minimum-mean-squared-error is then $\varepsilon_{min} = E[x^2(i)] - \underline{c}_{opt}^T \underline{\phi}_{xy}$.

The third filter, an RBF network, is a nonlinear filter, also referred to as a neural-network (NN) [4], because it has a NN structure with centres and weights (coefficients). The RBF nonlinearity is given by the *Gaussian* shape function $f(x) = exp(-x^2/(2\sigma^2))$, due to the white noise. Since we know the number of users and the spread-code of every user, we are able to calculate the optimum centres and weights. First, we compute the number of centres L, which for an AWGN channel is $L = 2^U$. Then we compute the centres from all possible combinations of the noise free transmitted waveforms. Our complete RBF function is then:

$$r(y) = \sum_{i=0}^{L-1} w_i \exp\left(-\frac{(\|\underline{y} - \underline{c}_i\|^2)}{2\sigma^2}\right) \tag{5}$$

w_i are the linear weights and $(\|\cdot\|)$ denotes the *Euclidean* norm where $\underline{y}$ is the input data vector and $\underline{c}_i$ are the centre vectors. Again, σ^2 is the noise power.

These filters can also be used in the multipath channel. The receivers are based on a new set of input samples $y(kN_n)$ where the range of n is extended to $0 \leq n < (N+2)$ to capture all the signal energy that originated from a single data bit. The MF receiver becomes a FIR filter with $(N+2)$-taps and the coefficients are given by the convolution between the desired spread- code and the channel impulse response H, $\underline{c}^{RAKE} = C_0 * H$. The MMSE is given by the same equation as before. The autocorrelation matrix Φ now has $((N+2) \times (N+2))$ elements and the crosscorrelation vector $\underline{\phi}$ is of length $(N+2)$. The elements must be calculated taking into account the multipath. The RBF receiver is also essentially unchanged apart from the fact that the number of centres is now 2^{3U} since we have to take the previous, current and next data bit into account to get all possible combinations, and the length of the centre vectors is again increased to $(N+2)$.

VOLTERRA RECEIVER

The *Volterra* representation for nonlinear systems [8] is based on the *Volterra* series functional representation. Our proposed *Volterra* filter (VS) can be described as three stages.

First, the received spread sequence of length N is expanded by a *Volterra* function (6) into a longer sequence of length M. The expanded sequence is processed with an M-tap FIR filter. Finally, a decision is made and the received user bit D_u estimated. The expansion is shown in Figure 1, the received sequence $\underline{y}(k)$ is expanded into $\underline{v}(k)$ of size M, $\underline{v}(k)$ consists of the first (linear) and the third-order *Volterra* terms for bit k. It has been shown in [9] that even-order

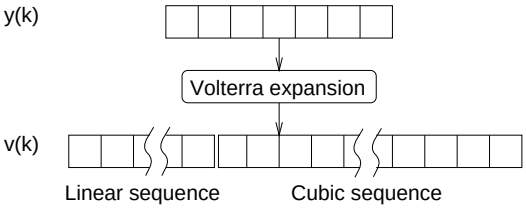


Figure 1: *The Volterra expansion for 7 chips.*

systems do not work due to the user data bit property of having equal probability of being either $+1$ or -1, resulting in zero coefficients. Thus the second-order system collapses to the first-order system. Hence, we focus on odd-order *Volterra* systems only.

Equation (6) presents the modified discrete *Volterra* expansion form for a third-order system, from which we derive the DS-CDMA applied form. h_i are the *Volterra* coefficients for order i.

$$y(n) = \sum_{a=0}^{N-1} h_1(a)x(n-a) + \sum_{a=0}^{N-1}\sum_{b=a}^{N-1}\sum_{c=b}^{N-1} h_3(a,b,c)x(n-a)x(n-b)x(n-c) \tag{6}$$

The received vector $\underline{y}$ of bit k is expanded using (6) into a vector $\underline{v}$ of size M, so $\underline{v}$ is a vector which has first the linear part, then the cubic part.

$$\underline{v} = [\sum_{u=1}^{U} D_u C_{u,0} + g(0), \ldots,$$
$$\sum_{u=1}^{U} D_u C_{u,(N-1)} + g(N-1), \left\{\sum_{u=1}^{U} D_u C_{u,0} + g(0)\right\}^3,$$
$$\left\{\sum_{u=1}^{U} D_u C_{u,0} + g(0)\right\}^2 \times \left\{\sum_{u=1}^{U} D_u C_{u,1} + g(1)\right\}, \ldots,$$
$$\left\{\sum_{u=1}^{U} D_u C_{u,(N-1)} + g(N-1)\right\}^3]^T \quad (7)$$

The filter coefficients are computed following the Wiener-Hopf equation (3). The autocorrelation for a *Volterra* system can be calculated from vector $\underline{v}$ as we do for the MMSE. One of the N linear terms in (7) is given in (8) and the first cubic term of (7) is (9).

$$v_n = y_n = \sum_{u=1}^{U} D_u C_{u,n} + g(n) =$$
$$D_1 C_{1,n} + D_2 C_{2,n} + \ldots + D_U C_{U,n} + g(n) \quad (8)$$

$$v_N = {y_0}^3 = \left\{\sum_{u=1}^{U} D_u C_{u,0} + g(0)\right\}^3 =$$
$$\{D_1 C_{1,0} + D_2 C_{2,0} + \ldots + D_U C_{U,0} + g(0)\}^3 \quad (9)$$

It can be seen that $v(k)$ is spread-code, user bit and noise dependent. Taking the properties of D_u and $g(n)$ into account, only those products with even power for D_u's and $g(n)$'s are non-zero, which we call here true terms. Obviously, the difficult task is in finding these true terms and to do this, we need to calculate every term's power. Due to the autocorrelation, we get terms up to a power of six. This becomes especially important for evaluating the noise terms. The noise characteristics are *Gaussian*, with zero mean and variance $\sigma^2 = 1$. The true terms we get have noise $g(n)$ with power 2, 4 and 6. The expected value for a *Gaussian* distributed random variable and its higher even moments can be found in Kay [10]. The autocorrelation matrix could therefore be determined if we can work out all the true terms and sum them. This is also valid for the crosscorrelation vector, the expected value between the desired data bit and the desired *Volterra* expanded sequence. Since no algorithm has yet been found that calculates all autocorrelation terms in a straightforward way, we use two different approaches to estimate the filter coefficients, to investigate the filter performance. Both are described in the next section.

Another aspect to be explored is the computational expense, since nonlinear functions generally have a higher complexity.

The *Volterra* expansion size M can be determined by the binomial coefficient (10), where N denotes the memory length, which here is the spreading size, and O the highest order. Note that we have odd orders only.

$$M(N,O) = \sum_{order=1}^{O} \binom{N+order-1}{order} \quad (10)$$

Results obtained from (10) are presented in Table 1. For $N = 7$ we have reasonable size, even for a 5th-order series. But if we introduce multipath or increase the processing gain, then M becomes very large and the autocorrelation matrix takes longer to compute, despite the fact that it is symmetric.

	M(N,O)	M(N,O)	C(N,O)	C(N,O)
O	N=7	N=9	N=7	N=9
1	7	9	7	9
3	84	165	252	495
5	462	1287	2310	6435

Table 1. *Coefficients M and Complexity C.*

The complexity of a *Volterra* compensator has been investigated by Tsimbinos [11]. He used as a measure of complexity the total number of multiplications for a discrete time Oth-order *Volterra* model, which in our case this is applied to (6). The complexity is then:

$$C(N,O) = \sum_{o=1}^{O} \frac{(N-1+o)!}{(o-1)!(N-1)!} \quad (11)$$

This complexity reflects the computational expense for the *Volterra* expansion, before we can process the signal with the FIR filter.

This complexity definition does not completely cover our system consideration. We propose an M-tap FIR filter, whose filter coefficients have been found by exploiting the Wiener-Hopf form. Perhaps more of interest for us is a ratio of complexity between the MMSE and the VS. We do not count the computation it needs for the expansion, we focus on the Wiener-Hopf equation only. So the expansion and the longer FIR filter size are additional computations. Now we derive a ratio for the needed floating point operations (flops) to find the filter coefficients. The difference between MMSE and VS is in the size of the received vector since we deal with the expanded sequence. So we can use the same form (4) and replace N by M. We assume that the matrices ϕ, P and $\sigma^2 I$ are known.

If we split (4) to work out the number of flops, we get first $(2NU^2 - NU)$ flops, followed by $(2N^2U - N^2)$ flops. If we do not add the diagonals of the identity matrix only, additional N^2 otherwise N, so in total $(2NU^2 + 2N^2U - NU)$ flops. The matrix inverse is

done by LU-Decomposition [12]. The total count, including the forward and backsubstitution, is N^3, for a $(N \times N)$ matrix. Finally, we compute the coefficients in (3) with the crosscorrelation, which require another $(2N^2-N)$ flops. So, the total number of flops for the MMSE is $(N^3+N^2(2U+2)+N(2U^2-U-1))$, so the *Volterra* filter requires $(M^3 + M^2(2U + 2) + M(2U^2 - U - 1))$. Now we rewrite (10) so we can substitute.

$$M = \sum_{o=1}^{O} \frac{[\prod_{i=1}^{o}(N+i-1)](N-1)!}{o!(N-1)!} \tag{12}$$

The ratio between VS and MMSE in terms of flops becomes:

$$R(N,U) = \frac{M^3 + M^2(2U+2) + M(2U^2-U-1)}{N^3 + N^2(2U+2) + N(2U^2-U-1)} \tag{13}$$

For $N = 7$, a third-order *Volterra* filter with $M = 91$ and 10 users, $R(N,U)$ is about 347, and for $N = 9$, $M = 174$ and 10 users, $R(N,U)$ is about 1417. The ratio becomes worse for a low number of active users. This due to the 3rd term in (13), which is linear, and is zero with $U = 1$. So we can conclude, the proposed *Volterra* filter needs many more operations compared to the MMSE and this becomes difficult in real-time applications.

RESULTS

Estimating the filter coefficients, without calculating the autocorrelation matrix explicitly, can be done using either an estimation technique or an adaptive algorithm [6]. Our adaptive system is based on the LMS algorithm. The other technique makes an estimate of the auto- and crosscorrelation. To get a good estimate, we have to average them over many ensembles. To find the noise free auto- and crosscorrelation, we must average over all possible combinations of data bits. We also have to take the noise power into account. To get the noise power we add the noise onto the chips and proceed in the same way as before, average the auto- and crosscorrelation over many times. For good estimations, we have to average over a long time, which makes the technique unsuitable for practical applications but enables us to verify its approximate performance.
The described receiver structures were simulated using Monte-Carlo simulations. The graphs show the probability of error (Pe) for the desired user one, and every active user transmitted 150k data bits D_u. All figures show the performance of these receivers for $E_b/N_0 = 9$dB and $N = 7$, where E_b is the normalised bit energy and N_0 the noise power-spectral density. The Pe for the *Volterra* filter with a small number of users is negligibly worse than the MMSE due to the error in the estimation technique, this is also true for the 5th-order LMS simulation.

First, we present in Figure 2 the results for the AWGN channel and compare a 3rd-order (VS3) and a 5th-order (VS5) *Volterra* filter against the other receivers. For both receivers, the coefficients are computed with the estimation technique. The performance for matched filtering (MF) becomes poor as multiple access interference increases. The RBF receiver is the maximum likelihood receiver in this AWGN channel and thus has the best performance. Since multiuser interference is nonlinear, nonlinear filters outperform linear filters, and the optimum receiver is a nonlinear filter. Due to the fact that a higher-order *Volterra* filter has a nonlinear component, its performance is between the performance of the MMSE and the RBF filter. Figure 3 shows the

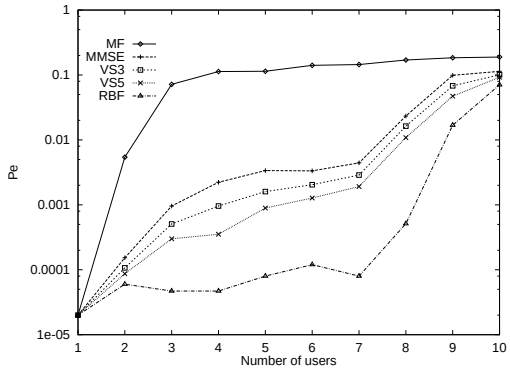


Figure 2: *Pe against the number of users in an AWGN channel.*

performance for VS3_EST against the other filters in stationary multipath. The simulation is done for less users since the number of centres for the RBF receiver becomes very large. Further, the estimation technique for VS3_EST used, suffers from the same problem, because we have to average over all possible combinations. However, the results show a significant improvement of VS3_EST in performance against the MMSE. We also investigated a 5th-order *Volterra* filter. Because of its size, we used the LMS approach to estimate the coefficients. The results are presented in Figure 3 and indicate, that VS5_LMS performs better than VS3_EST, but the gain is at the expense of computational costs.
Both simulations show, compared to the MMSE, that the *Volterra* filter performs even better in the multipath channel than in the AWGN channel, due to benefits of nonlinear equalisation. The last investigation, Figure 4, shows the Pe for a range of noise power in an AWGN channel. The number of active users was five, and every user transmitted 150k user data bits, except for 12dB, where 400k have been transmitted. The theoretical Pe (MMSE theory) for the MMSE filter, for desired user one, is also included [7]. The RBF did not give any errors for 12dB.

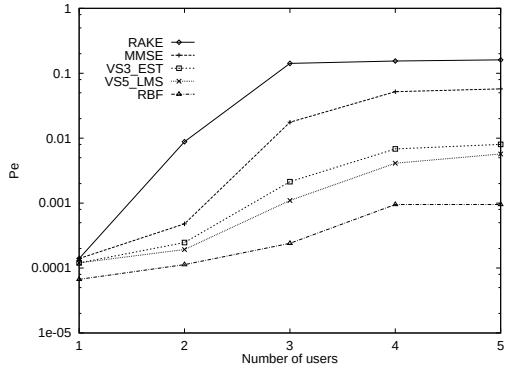


Figure 3: *Pe against the number of users in a multipath channel.*

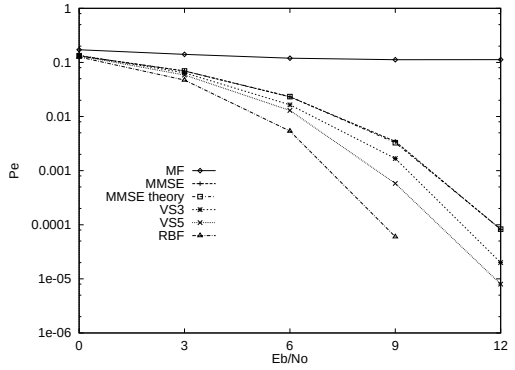


Figure 4: *Pe against a range of E_b/N_0 in dB for a system with 5 active users in an AWGN channel.*

CONCLUSIONS

We have shown that the proposed receiver based on the *Volterra* series performs better than the MMSE. The results we obtained from Monte-Carlo simulations show a significant improvement in performance for a higher number of active users, compared with the MMSE. Since the proposed receiver is nonlinear, performance was even better in the stationary multipath channel used.

A drawback is its complexity, especially for systems higher than 3rd-order. We showed that the *Volterra* filter is computationally more expensive than the MMSE, but gains in performance. However, it might be still too complex to implement in real-time applications, since in a mobile environment the receiver has to be adaptive.

Future work will investigate the theoretical performance and the computation of the coefficients without using an estimation technique, and seek other nonlinear algorithms with lower complexity.

ACKNOWLEDGMENTS

We would like to thank the PJJ Foundation Basel, the Dept. of Education Baselland and the Engineering and Physical Science Research Council (UK) for their support.

REFERENCES

[1] S. Moshavi, "Multi-User Detection for DS-CDMA Communications," *IEEE Communications Magazine*, vol. 34, pp. 124–136, October 1996.

[2] T. S. Rappaport, *Wireless communications*. Upper Saddle River, New Jersey, USA: Prentice Hall, 1st ed., 1996.

[3] S. Moshavi, E. G. Kanterakis, and D. L. Schilling, "Multistage Linear Receivers for DS-CDMA Systems," *Int. Journal of Wireless Information Networks*, vol. 3, pp. 1–17, January 1996.

[4] U. Mitra and H. V. Poor, "Neural Network Techniques for Adaptive Multiuser Demodulation," *IEEE Transactions on Selected Areas in Communications*, vol. 12, pp. 1460–1470, December 1994.

[5] D. G. M. Cruickshank, "Radial basis function receivers for DS-CDMA," *IEE Electronics Letters*, vol. 32, pp. 188–190, February 1996.

[6] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey, USA.: Prentice Hall International, 3rd ed., 1996.

[7] D. G. M. Cruickshank, "Optimal and Adaptive FIR Filter Receivers for DS-CDMA," in *Proceedings International Symposium on Personal Indoor and Mobile Communications*, pp. 1339–1343, IEEE, September 1994.

[8] M. Schetzen, *The Volterra and Wiener theories of nonlinear systems*. New York, USA: John Wiley and Sons Inc., 1st ed., 1980.

[9] R. Tanner and D. G. M. Cruickshank, "Nonlinear Volterra Filter Receiver for DS-CDMA," in *Proceedings 4th International Conference on Mathematics in Signal Processing, IMA, University of Warwick, UK*, IEE, December 1996.

[10] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, USA: Prentice-Hall, 1st ed., 1993.

[11] J. Tsimbinos and K. V. Lever, "The Computational Complexity of Nonlinear Compensators Based on the Volterra Inverse," in *Proceedings Signal Processing Workshop on Statistical Signal and Array Processing, Corfu, Greece*, vol. 1, pp. 387–390, IEEE, June 1996.

[12] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*. Cambridge, USA: Cambridge University Press, 2nd ed., 1994.

Receivers for Nonlinearly Separable Scenarios in DS-CDMA

R. Tanner*, D.G.M. Cruickshank, C.Z.W. Hassell Sweatman, B. Mulgrew

July 5, 1998

Abstract

Receiver structures are presented which overcome the nonlinear separation problem in a DS-CDMA system, based on linear programming and the radial basis function. When a bank of matched filters is applied for preprocessing, the received DS-CDMA sequence is, under certain circumstances, no longer linearly separable.

Indexing terms

DS-CDMA, non linear separation, radial basis function (RBF), linear programming (LP).

1 Introduction

An approach for solving non linearly separable scenarios [1] for equalisers has been presented in [2]. The equaliser is based on a neural network design where the weights have been computed by linear programming (LP).
Since the optimum separation boundary is nonlinear [3], linear receivers are outperformed by nonlinear receivers [4] and if the scenario is not linearly separable, linear receivers fail completely. The drawback with nonlinear receivers is they have either long training times such as neural networks (NN) or are computationally expensive such as maximum likelihood receivers (ML).
After we consider the DS-CDMA system used, receiver structures are proposed, followed by simulation results and a conclusion.

2 System consideration

We consider a DS-CDMA downlink system for U independent users as described in [3]. We allocate for each user a randomly generated spreading code of length N. The user data bits are either $+1$ or -1 with equal probability. Further, we assume a communication system that is bit and chip synchronous and all users are transmitting with equal power. Our investigation is focussed on stationary multipath channels whose channel impulse response H_{ch} is assumed to be known. The received sequence is preprocessed by a bank of matched filters (MF). This fast and simple preprocessing stage is only dependent on the number of active users and no channel coefficients have to be updated, as in a maximum ratio combining RAKE filter. The bit synchronised output of each MF is fed into several receiver structures to compare the performance. Due to the fact that H_{ch} is not taken into account in the preprocessing stage and intersymbol interference is high due to the short spreading codes used, there is a loss of information yielding a generally poor system performance. However, all receiver structures are fed with the same information which ensures that the comparison is fair.
We compare several new receiver structures against a MF, a linear minimum mean square error filter (U-tap FIR), which we train with the LMS algorithm [5] and therefore name LMS, and a RBF filter [3][5].

*The authors are with the Signal and Systems Group, Electrical Engineering, University of Edinburgh, Edinburgh EH9 3JL, Scotland, UK. E-mail: rt@ee.ed.ac.uk., Phone: 0131 - 650 - 5655, Fax: 0131 - 650 - 6554

3 Proposed receivers

Based on [2], we apply the idea of using LP to solve the nonlinearly separable problem. Figure 1 presents a nonlinearly separable scenario, there are two groups of points depicted, $(+1)$ and (-1), dependent on the desired user bit sign. The points represent all possible combinations of the preprocessing output which we also refer to as centres. It can be seen that the two groups overlap and therefore a linear receiver (LMS) can not separate them with a single hyperplane, see the LMS decision boundary in Figure 1. The linear receiver therefore gives poor performance.
To construct an RBF receiver, we have to compute its noise free centres and weights. In a multipath scenario, there are $2^{3\times U}$ centres and weights [4]. Due to preprocessing with the users spreading code, and not the convolution between spreading code and H_{ch}, some centres are duplicated and can be dismissed. This truncated set of centres represents all possible combinations with which the receivers have to deal. However, this set can become big and makes the RBF computationally expensive, hence we apply LP for reducing the RBF complexity. Our LP approach exploits the position of the centres, by grouping the centres, according to the desired user bit. The LP algorithm, set up according to [2], separates the two groups by hyperplanes and two hyperplanes form a slab. The function we minimise with LP results in a

change of the width of a slab [2].

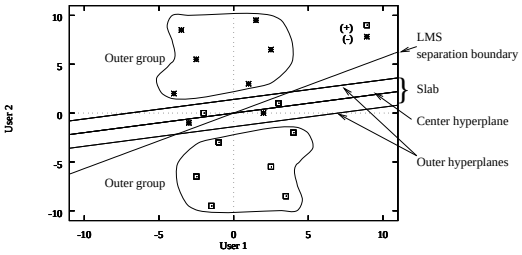


Figure 1: The preprocessing output grouped in (+1) and (-1), with the separation boundaries for the LMS and SLAB filter.

First, LP tries to separate the two groups with a single hyperplane, but this is not possible since the scenario is not linearly separable. Then LP increases the width of the hyperplane, resulting in two hyperplanes, which bound a slab, as depicted in Figure 1. After the width reaches a size where the two groups can be separated by two hyperplanes, by neglecting the centres between the hyperplanes, LP stops. This is the case in our example when the four centres closest to the origin are neglected. Now, there are two inner groups and two outer groups of centres. Since we know from each centre its position in the U-dimensional space, we can find the coefficients for each hyperplane which separates two groups. After all centres are dismissed from the two outer groups, the centres of the inner groups are left for separation. Again, LP tries to separate them with a single hyperplane and succeeds. Thus, LP with two slabs (3 hyperplanes) separates the groups from each other, as shown in Figure 1. The RBF has 16 centres in the scenario depicted in Figure 1. Our aim is to reduce receiver complexity and this can be done in many ways. The first receiver proposed, based purely on LP, is denoted in the simulation results as SLAB. The idea is to classify the received sequence dependent on its position in the U-dimensional space and to determine the received user bit. For the scenario presented in Figure 1, this would be, to determine in which slab the received sequence (point) lies and the group's sign. This classification task is fast and very simple. The next two proposed receiver structures we shall call hybrid receivers, since we combine LP with a truncated RBF. Again referring to Figure 1, the four inner centres are close to the dividing hyperplane. Due to the short distances, these points will cause most of the errors. Hence, we deploy for these four points an RBF with 4 centres. So first, we classify the received sequence and if it lies between the two outer hyperplanes, we make use of an RBF. Since all centres occur with equal probability, the RBF will be needed only 1/3 of the time, moreover, the RBF has 4 centres instead of 16, thus gaining in computational time and RBF complexity. This filter we shall name Hybrid4. The other hybrid based filter we name Hybrid8. Here, we set up a truncated RBF which also includes the centres which are close to the outer hyperplanes since they are most responsible for the performance loss, this gives 8 centres for the scenario depicted in Figure 1. Again, we gain from a truncated set of centres (8 instead of 16) and make use of the RBF only when the received sequence lies between the two outer hyperplanes which is 1/3 of the time.

4 Results

We consider a DS-CDMA scenario with two active users, a spreading code with size $N = 7$ and a 3-tap stationary multipath channel with impulse response $H_{Ch}(z) = 0.25 + 0.5z^{-1} + 1.0z^{-2}$. There are 16 different preprocessing output combinations the receivers have to deal with, see the points in Figures 1. In Figure 2 we present the bit error ratio (BER) against E_b/N_0 for this scenario.

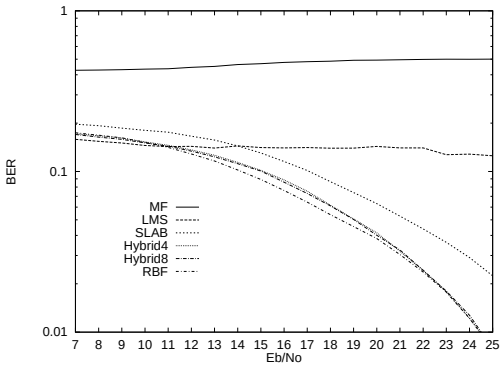


Figure 2: Simulation results, bit error ratio versus E_b/N_0.

The MF performs poorly due to multiuser and intersymbol interference. The LMS also performs poorly due to its inability to separate the four points between the two outer hyperplanes. The RBF receiver performs best due to its ML structure. The SLAB receiver is limited in performance by the short distance of the points close to the center hyperplane, this becomes severe for high SNR. Hybrid4 and Hybrid8 perform much better than the SLAB, and the simulations suggest that there is no significant loss in performance when we use four centres instead of eight in the hybrid structure. Moreover, it seems that giving a hybrid structure more than the four inner points (centres) does not enhance its performance. It can also be seen, that at high SNR the hybrid structures converge to the RBF's performance.

5 Conclusion

A scenario has been presented which is of a nonlinearly separable nature in DS-CDMA. Three receiver structures have been proposed.
We proposed a LP based receiver which uses slabs. To improve the performance of this SLAB receiver, we propose two hybrid receiver architectures, where we combine LP and the RBF. All three receivers outperform the best linear receiver, moreover, the hybrid structures converges to the RBF's performance. The SLAB receiver is simple to implement and very fast. Both hybrid structures are faster and have a subset of centres only while they perform nearly as well as an RBF to which we assign all centres.

References

[1] G. J. Gibson, S. Siu, and C. F. N. Cowan, "The Application of Nonlinear Structures to the Reconstruction of Binary Signals," *IEEE Transactions on Signal Processing*, vol. 39, pp. 1877–1884, August 1991.

[2] C. Z. W. Hassell, G. J. Gibson, and B. Mulgrew, "Constructive Neural Network Design for the Solution of Two State Classification Problems with Applications to Channel Equalization," in *Proceedings Workshop on Neural Networks for Signal Processing, Boston, USA*, IEEE, September 1995.

[3] B. Mulgrew, "Applying Radial Basis Functions," *IEEE Signal Processing Magazine*, vol. 13, pp. 50–65, March 1996.

[4] D. G. M. Cruickshank, "Radial basis function receivers for DS-CDMA," *IEE Electronics Letters*, vol. 32, pp. 188–190, February 1996.

[5] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey, USA.: Prentice Hall International, 3rd ed., 1996.

ISSSTA 1998, Sun City, South Africa, September 2 – September 4

RBF Based Receivers for DS-CDMA with Reduced Complexity

R. Tanner and D. G. M. Cruickshank
The University of Edinburgh,
Department of Electronics and Electrical Engineering,
Signals & Systems Group
The King's Buildings, Mayfield Road
Edinburgh, EH9 3JL, UK
e-mail: rt@ee.ed.ac.uk

Abstract — **This paper presents a Mahalanobis based RBF receiver structure with reduced complexity. It is also illustrated, how the RBF receiver could be employed as a multiuser detector. The RBF structure has superior performance over linear receiver structures and is equivalent to MLSD in an AWGN channel. However, a drawback is its complexity in multipath channels. Ideas from pattern recognition are exploited to reduce its complexity for multipath scenarios with little performance loss.**

I. INTRODUCTION

DS-CDMA is a candidate for the next generation mobile system [1]. Research focuses on multiuser detector (MUD) schemes which process signals derived from a bank of matched filters (MF) [2, 3, 4]. These preprocessed signals are then fed into the receiver structure (MUD). The multipath combining process can be conducted either before or as part of the MUD process [4].

It has been shown that nonlinear receivers outperform linear receivers since the optimal decision boundary in DS-CDMA is nonlinear [5]. Proposed nonlinear receiver structures based on neural networks or polynomial series can approximate the decision boundary well and therefore achieve superior performance [6, 7]. However, all of them are either highly complex or need long training times to converge and are therefore not of interest for practical applications.

A promising structure is the radial basis function network (RBFN), which is also referred to as a neural network due to its structure. It is fully determined when the spreading codes of all users and the channel impulse response are known. Hence, it needs no training. Moreover, it is a maximum likelihood sequence detector (MLSD) for the Gaussian channel. However, its complexity in terms of centres (neurons,nodes) grows exponentially with the number of users, and becomes too complex when applied to multipath channels. Thus we look for a near optimum performing network with less complexity in terms of centres.

This paper is organised as follows. First, we consider the DS-CDMA system used. Then we introduce the RBFN and propose a receiver structure with less centres. Next, we present results obtained from Monte-Carlo simulations and finally draw the conclusions.

II. THE DS-CDMA SYSTEM

There are two ways to process the received DS-CDMA signal to reconstruct the transmitted user bit. One way is to look at the chip level, where the receiver processes a signal (vector) whose length is equal to the length of the spreading codes used [6]. Another way is to feed a preprocessed signal (vector) into the receiver, derived from a bank of MFs [4], whose length is equal to the number of transmitting users. In this section, we consider the received signal and then we describe the preprocessing stage used.

The investigation is conducted on a fully synchronised downlink DS-CDMA system with U independent users, $u = 1, 2, ..., U$, and known channel impulse response H_{ch}. The user's transmitted data bit (symbol) for bit k is denoted as $D_u(k)$ and is either $+1$ or -1 with equal probability and all user are transmitting with equal power, normalised to one. Without loss of generality, the desired user is assumed to be user one. The user's unique spreading sequence of length N will be denoted as $C_{u,n}$, with $n = 1, 2, ..., N$ chips, where the chips are either $+1$ or -1. The received signal $y(n)$ at the chip rate for symbol k and an additive-white-Gaussian-noise (AWGN) channel can be expressed as:

$$y(kN+n) = \sum_{u=1}^{U} D_u(k) C_{u,n} + g(kN+n) \qquad (1)$$

where $g(kN+n)$ denotes the added noise component. Next, the definition for a multipath channel is given, the channel is modelled by a FIR structure with L-taps [8]. The channel impulse response is assumed to be known, e.g. due to pilot channel estimations. Due to the complexity of some receivers presented, we use a stationary channel with $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$. All signals pass through the same channel, so the received signal at the time of data bit k and chip n becomes:

$$\begin{aligned} y(kN+n) &= 0.3482 \times \sum_{u=1}^{U} D_u(k) C_{u,n} \\ &+ 0.8704 \times \sum_{u=1}^{U} D_u(k) C_{u,n-1} \qquad (2) \\ &+ 0.3482 \times \sum_{u=1}^{U} D_u(k) C_{u,n-2} + g(kN+n) \end{aligned}$$

If $(n-x) < 1$ then the term $D_u(k)C_{u,n-x}$ is replaced by $D_u(k-1)C_{u,N+n-x}$ and if n or $n-x$ is $> N$ then $D_u(k)C_{u,n-x}$ is replaced by $D_u(k+1)C_{u,n-N-x}$. It shows clearly the presence of channel induced intersymbol interference (ISI).

Due to bit synchronisation, the received signal $y(n)$ at chip level can be expressed as signal $\mathbf{y}(k)$ (in vector notation). In AWGN, the signal $\mathbf{y}(k)$ has length N and in multipath $N+(L-1)$, to cover all the bit's energy. If a receiver processes $\mathbf{y}(k)$, then we refer to this receiver structure as a chip level based receiver (CLB).

The preprocessing stage consists of a bank of U MFs for practical reasons and not whitening filters, which leads to correlated noise at the preprocessor output. Figure 1 shows the attached preprocessing stage at the front end of the RBF receiver. For the multipath scenario, we replaced the MFs with maximum ratio combining (MRC) RAKEs [8]. For the AWGN channel,

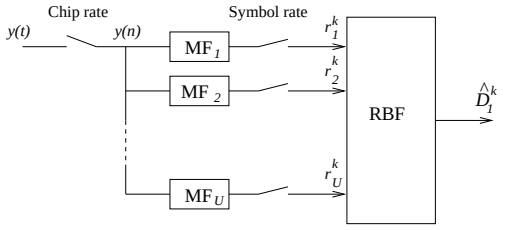


Figure 1: *The RBF receiver with the preprocessing stage.*

signal $\mathbf{y}(k)$ has length N will be mapped into signal $\mathbf{r}^k = [r_1^k, \ldots, r_U^k]^T$ (in vector notation) of length U, due to preprocessing. With multipath, the preprocessing stage will map signal $\mathbf{y}(k)$ with length $N+(L-1)$ into signal $\mathbf{r}^k$ with length U. This reduces the dimensionality of a DS-CDMA receiver's input, especially when $U << (N+(L-1))$, and can cause some performance loss. The preprocessing based (PPB) RBF receiver will process signal $\mathbf{r}^k$ and outputs the estimated bit k of the desired user one, $\widehat{D}_1^k$.

III. RADIAL BASIS FUNCTION

A DS-CDMA receiver has to deal with a nonlinear decision boundary [5], hence, nonlinear receivers will outperform any linear receiver. Thus, the optimum receiver approximates this nonlinear boundary perfectly, however, this can only be achieved at the expense of computational complexity. RBFN can approximate nonlinear boundaries well [9] and outperform established linear receiver structures [6]. Moreover, in AWGN, the RBFN is a MLSD. However, in a multipath scenario, ISI has to be taken into account and its structure grows exponentially with the number of users which makes it too complex for implementation. Therefore techniques to reduce its network complexity are desirable. So, after we introduced the CLB RBF receiver (see Figure 2), and the PPB RBF receiver, Figure 1, we present a way to reduce the network complexity and propose an PPB RBF receiver with less complexity.

The RBFN has a neural network like structure, see Figure 2, it shows a single user CLB RBF receiver. The

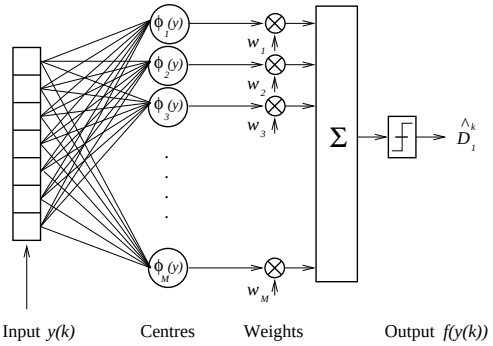


Figure 2: *The chip level based RBF receiver, the weights w_m determine to which user the estimated bit $\widehat{D}_u^k$ belongs. By recomputing the weights and preprocessing the right part of the network, the transmitted bits for each user can be estimated.*

RBF processes signal $\mathbf{y}(k)$, which is equal to the number of nodes in the input layer. It has a hidden layer (neurons or nodes are named centres), linear weights and the output layer. The output layer consists of a summation network and a slicer, which performs the bit sign decision (Heaviside). Before the RBF can be used for filtering, the network has to be initialised by computing centre vectors and the weights.

The RBF approach introduces a set of M basis functions (centres), one for each possible received signal, the most common basis function is the Gaussian (3), where $\phi(\cdot)$ is the nonlinear function.

$$\phi_m(\mathbf{y}(k)) = \exp\left(-\frac{||\mathbf{y}(k)-\mathbf{c}_m||^2}{2\sigma^2}\right) \qquad (3)$$

The mth such function depends on the Euclidean distance $||\mathbf{y}(k)-\mathbf{c}_m||$ and σ^2. The variance σ^2 determines the radius of influence for the Gaussian shaped bell and is the noise power. The centre vectors $\mathbf{c}_m$, for $m = 1, 2, ..., M$ centres, represent all possible received noise free signals of $\mathbf{y}(k)$, derived from all possible transmitted bits D_u. Hence, in AWGN there are $M = 2^U$ vectors $\mathbf{c}_m$ of length N. So, if one user is active and $\mathbf{c}_1$ is the spreading code, then $D_1 = 1$ and $\mathbf{c}_2$ is the negated spreading code, since D_1 can also be -1. According to this rule, all M centre vectors $\mathbf{c}_m$ can be constructed. The centre vectors are set up for the multipath scenario correspondingly, with $M = 2^{3U}$ centres however, since the previous, current and next bit are taken into account due to ISI, each centre vector has length $N+(L-1)$. Obviously, the RBFN becomes very

complex in multipath scenarios.
The weights w_m, $m = 1, 2, ..., M$, can be derived from the construction of the centre vectors, since each bit sign D_u is known from the combinatorial construction of the M centre vectors, hence, w_m is either $+1$ or -1.
Thus the complete RBF is described as:

$$f(\mathbf{y}(k)) = \text{sign}\left(\sum_{m=1}^{M} w_m \phi_m(\mathbf{y}(k))\right) \quad (4)$$

From this definition, we can easily derive a MUD receiver when all user share the same channel, by updating the weights to the value, corresponding to the bit sign of the desired user bit. The advantage is, that we do not have to compute the centre vectors again and only process the output layer, which is faster. Of course, this does not work for the asynchronous uplink with the assumptions made.

Next, we consider the PPB RBF see Figure 1. The centre vectors $\mathbf{c}_m$ are derived from the preprocessed signals, each centre vector $\mathbf{c}_m$ of the chip level based RBF returns after preprocessing a vector $\mathbf{r}$ of length U, which is a centre vector of the RBFN with preprocessing (PRBF). This PRBF has centre vectors with less dimensions. The weights are not influenced by the preprocessing and are constructed in the same way as without preprocessing. Preprocessing has three effects, first, we can depict scenarios for $N > 3$, which is more realistic. Second, preprocessing can create duplicates of centre vectors, dependent on the chip sign of the spreading codes which are involved in ISI and H_{ch}. Third, preprocessing induces additional correlation since each filter of the preprocessor unit processes the same noise, which is added to $\mathbf{y}(k)$ (1). Hence, there exists a correlated nature of the noise components. These effects are illustrated in Figure 3. Figure 3 presents all possible preprocessed combinations for a two user scenario,

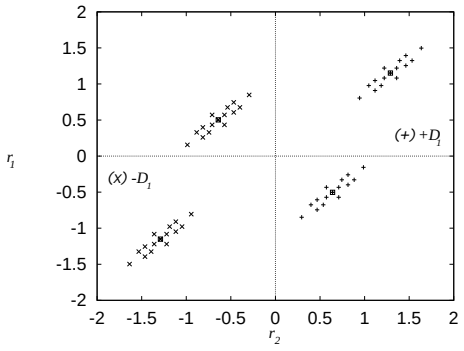


Figure 3: *All possible signal combinations (centres) for a two user multipath DS-CDMA. The centres are spread around a single centre and form a cluster due to ISI.*

with randomly generated spreading sequences of length $N = 7$ and $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$. This scenario has 4 clusters, each cluster has 15 points (equivalent to centres), and not 16 due to effect number two, and a PRBF could be constructed from these 60 centres according to (4). The centre of each cluster is determined by the mean vector of the cluster, and the shape of the cluster is determined by its covariance matrix [10]. The Euclidean distance measure (3) is optimum for the CLB RBF, since the signals have a univariate normal density. MF and RAKE preprocessed signals have a multivariate normal density (hyperellipsoid) [10], hence the basis function (3) has to be replaced by another function. Note, when (3) is used for a PRBF, then $\mathbf{y}(k)$ is replaced by $\mathbf{r}(k)$.
This new basis function (5) is based on the Mahalanobis distance measure [10, 9]:

$$\phi_m(\mathbf{r}(k)) = \exp\left(-\frac{(\mathbf{r}(k) - \mathbf{c}_m)^T \mathbf{S}^{-1}(\mathbf{r}(k) - \mathbf{c}_m)}{2}\right) \quad (5)$$

Knowledge of the covariance matrix $\mathbf{S}$ allows us to calculate the dispersion of the data in any direction [10]. The covariance matrix is derived from the expected value of the signals $\mathbf{r}$ and is $\mathbf{S} = E[(\mathbf{r}(k) - \mathbf{c}_m)(\mathbf{r}(k) - \mathbf{c}_m)^T]$, where $\mathbf{c}_m$ is a noise free state of $\mathbf{r}(k)$ [5]. The eigenvectors of $\mathbf{S}$ are the principal axes of these hyperellipsoids and the eigenvalues determine their length. If all the off diagonal elements of $\mathbf{S}$ are zero, (5) would be reduced to the product of the univariate normal densities and (3) could be used, however, this is not the case, even with orthogonal spreading codes, due to multipath effects.

Now we present an idea on PRNF centre reduction for multipath scenarios based on the Mahalanobis approach. Figure 4 shows the convex hulls [11][1] of eight clusters for a three user scenario from an unusual perspective (top). This scenario has the same parameters as the centres depicted in Figure 3. The location of the clusters are given by the spreading codes (covariance matrix) and the shape is due to ISI. Each cluster consists of a maximum 2^{2U} centres, which we replace by a single centre. We shall refer to the replacement centre as a super centre. This super centre is derived from the mean of the centres within a cluster and represents the ISI free centre. Hence, instead of computing 2^{3U} centres and deriving the super centres, we simply compute the 2^U centres, as for the AWGN scenario. Convolution with H_{ch} and preprocessing will lead to the super centres. This is a centre reduction from 2^{3U} to 2^U.

IV. RESULTS

The bit error ratio (BER) performance for different receiver structures is obtained by Monte-Carlo simulations and compared against each other. The PPB RBF receiver with Mahalanobis based radial function is named

[1] http://www.geom.umn.edu/software/download/geomview.html

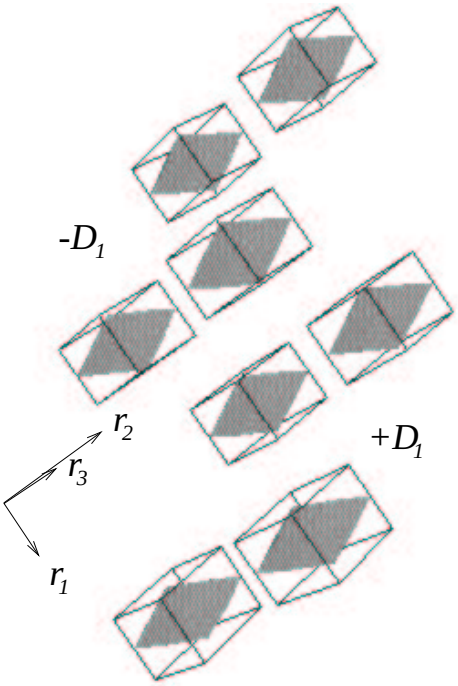


Figure 4: *A 3D-graph which shows the convex hulls of eight clusters for a three user multipath DS-CDMA scenario. Each cluster contains 64 centres and is replaced by a super centre to reduce the number of RBFN centres.*

MRBF and the PPB RBF receiver with Mahalanobis based radial function and super centres we shall name SRBF. These two Mahalanobis based RBF receivers are compared against the CLB RBF (CRBF), an CLB RBF with an Euclidean basis function (ERBF), and the well known MMSE receiver [12].

First, we present results obtained in AWGN for randomly generated spreading codes of length $N = 7$ and $E_b/N_0 = 7$dB, see Figure 5. Here in addition, we also present the results for a simple MF receiver. The MF and CRBF are both chip level based receivers, moreover, the CRBF is in this scenario a MLSD. All RBF receivers have 2^U centres, and the centre vectors for the ERBF and MRBF have size U and for the CRBF size N. The MF and MMSE receivers are both linear receiver structures and are outperformed by the nonlinear RBF receivers. The MF suffers from multiuser interference, and the MMSE lacks in its ability to perform a nonlinear decision boundary. Since the shape of the multidimensional clusters are not spherical, more missclassifications occur, hence, the ERBF's performance tends towards the MMSE receiver's performance. Figure 5 also shows that the MRBF outperforms the ERBF

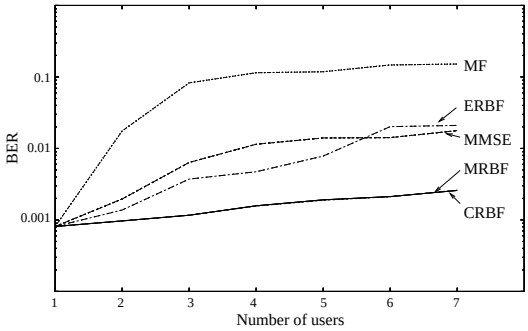


Figure 5: *BER against the number of users in AWGN for $E_b/N_0 = 7dB$ and $N = 7$. The proposed receiver MRBF performs as well as the maximum-likelihood-sequence-detector CRBF.*

receiver since the correlated noise is taken into account. Thus, this receiver has also optimum performance as the results show.

Figure 6 presents results obtained from a stationary multipath scenario, with $H_{ch}(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$. Again, the same receivers are compared against each other. Due to the large number of centres, the simulation has been stopped at six users. In addition, the performance for another CLB receiver is given, named CSRBF, where the centres of this receiver are the 2^U ISI free centres. Thus the CSRBF centres are derived from the AWGN CRBF centres, and then convolved with H_{ch}, hence of length $N+(L-1)$. Again, all

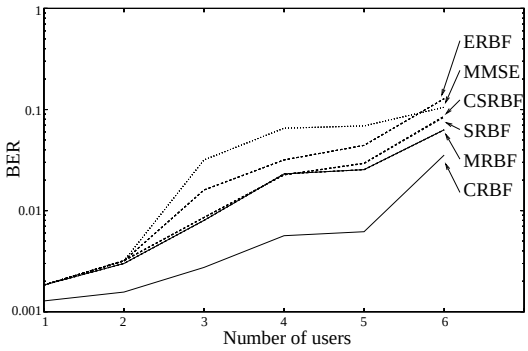


Figure 6: *BER against the number of users for a stationary multipath channel with spreading codes of length $N = 7$ and $E_b/N_0 = 7dB$.*

RBF receivers outperform the linear MMSE receiver. The results show the best performance for the CRBF since it processes a higher dimensional signal, as long as $U < N$. Receiver ERBF is again outperformed by both Mahalanobis based receivers, even by the SRBF struc-

tures. The best result among the receivers which have a preprocessing stage attached is achieved by MRBF, since it matches the preprocessed signal $\mathbf{r}(k)$ against all possible received combinations of $\mathbf{r}$ best. The highly simplified SRBF performs nearly as well as the MRBF, but with only 2^U centres instead of 2^{3U}. The performance loss is mainly due to the approximated shape of each cluster, whereas the MRBF deals with the true shape. Receiver CSRBF performs equal to SRBF, from this we conclude, that the cluster spread is induced by ISI.

V. CONCLUSIONS

A new RBF receiver structure for DS-CDMA systems with preprocessing has been presented which does not need whitening filters. The new structure exploits the Mahalanobis distance measure instead of the Euclidean distance measure, and takes the correlated nature of the noise component into account, which is induced when a preprocessing stage is used. Results obtained from Monte-Carlo simulations showed its performance compared against other RBF receivers and the linear MMSE receiver.

It has been shown that in AWGN, this new structure achieves optimum performance while its complexity is reduced, especially for long spreading sequences.

For multipath scenarios, we reduced the number of centres from the size of 2^{3U} to 2^U, by introducing super centres, with little loss in performance compared with the RBFN which consists of all possible centres.

Since the new structure has a reasonable complexity, it may be of interest to investigate its performance in a nonstationary channel. Also of interest is the investigation of the clustering behaviour, where we presume to gain some additional insight to reduce its complexity even more.

Acknowledgements

We would like to thank the PJJ Foundation Basel, the Dept. of Education Baselland and the Engineering and Physical Science Research Council (UK) for their support.

References

[1] T. Ojanperä, A. Klein, and P. O. Anderson, "FRAMES Multiple Access for UMTS," in *Proceedings Colloquium on CDMA and Applications for third generation mobile systems, ref no.1997/129*, pp. 7/1–7/8, IEEE, May 1997.

[2] R. M. Buehrer, N. S. Correal, and B. D. Woerner, "A Comparison of Multiuser Receivers for Cellular CDMA," in *Proceedings International Conference Globecom'96, London, UK*, pp. 1571–1577, IEEE, November 1996.

[3] T. Ojanperä, K. Rikkinen, H. Häkkinen, K. Pehkonen, A. Hottinen, and J. Lilleberg, "Design of a 3rd Generation Multirate CDMA System with Multiuser Detection, MUD-CDMA," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Mainz, Germany*, vol. 1, pp. 334–338, IEEE, 1996.

[4] M. Junatti and S. Glisic, "Advanced CDMA for wireless communications," in *Wireless Communications: TDMA vs. CDMA* (S. Glisic and P. Leppänen, eds.), pp. 447–490, Dortrecht, NL: Kluwer, 1997.

[5] B. Mulgrew, "Applying Radial Basis Functions," *IEEE Signal Processing Magazine*, vol. 13, pp. 50–65, March 1996.

[6] D. G. M. Cruickshank, "Radial basis function receivers for DS-CDMA," *IEE Electronics Letters*, vol. 32, pp. 188–190, February 1996.

[7] R. Tanner and D. G. M. Cruickshank, "Volterra Based Receivers for DS-CDMA," in *Proceedings International Symposium on Personal, Indoor and Mobile Radio Communications, Helsinki, Finland*, vol. 3, pp. 1166–1170, IEEE, September 1997.

[8] J. G. Proakis, *Digital Communications*. New York, USA: McGraw-Hill, 3rd ed., 1983.

[9] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford, UK: Oxford University Press, 1st ed., 1995.

[10] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York, USA: J. Wiley and Sons, 1st ed., 1973.

[11] C. B. Barber, D. O. Dobkin, and H. T. Huhdaanpaa, "The Quickhall algorithm for convex hulls," *ACM Transcations on Mathematical Software*, vol. 22, pp. 469–483, December 1996.

[12] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey, USA.: Prentice Hall International, 3rd ed., 1996.

References

[1] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*. Cambridge, USA: Cambridge University Press, 2nd ed., 1992.

[2] A. Baier, U. Fiebig, W. Granzow, W. Koch, P. Teder, and J. Thielecke, "Design Study for a CDMA-Based Third-Generation Mobile Radio System," *IEEE Transactions on Selected Areas in Communications*, vol. 12, pp. 733–743, May 1994.

[3] S. Pike, "The radio interface for UMTS," in *Proceedings Colloquium on Personal Communications in the 21st Century part I and II, ref no.1998/214 and 1998/242*, pp. 7/1–7/11, IEE, February 1998.

[4] K. Ohno, M. Sawahashi, and F. Adachi, "Wideband coherent DS-CDMA," in *Proceedings International Conference on Vehicular Technology, Chicago, USA*, vol. 2, pp. 779–783, 1995.

[5] F. Adachi, M.Sawahashi, T. Dohi, and K. Ohno, "Coherent DS-CDMA: Promising Multiple Access for Wireless Multimedia Mobile Communications," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Mainz, Germany*, vol. 1, pp. 351–358, IEEE, 1996.

[6] A. Fukasawa, T. Sato, Y. Takizawa, R. E. Fisher, T. Kato, and M. Kawabe, "Wideband CDMA System," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Mainz, Germany*, vol. 1, pp. 244–248, IEEE, 1996.

[7] T. Ojanperä and *et al.*, "FRAMES: - Hybrid Multiple Access Technology," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Mainz, Germany*, vol. 1, pp. 320–324, IEEE, 1996.

[8] T. Ojanperä, K. Rikkinen, H. Häkkinen, K. Pehkonen, A. Hottinen, and J. Lilleberg, "Design of a 3rd Generation Multirate CDMA System with Multiuser Detection, MUD-CDMA," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Mainz, Germany*, vol. 1, pp. 334–338, IEEE, 1996.

[9] T. Ojanperä, A. Klein, and P. O. Anderson, "FRAMES Multiple Access for UMTS," in *Proceedings Colloquium on CDMA and Applications for third generation mobile systems, ref no.1997/129*, pp. 7/1–7/8, IEEE, May 1997.

[10] T. Ojanperä, P. A. Ranta, S. Hämäläinen, and A. Lappeteläinen, "Analysis of CDMA and TDMA for 3rd Generation Mobile Radio Systems," in *Proceedings International Conference on Vehicular Technology, Phoenix, AZ, USA*, vol. 2, pp. 840–844, IEEE, May 1997.

[11] S. Ghaheri-Niri and R. Tafazolli, "Cordless-cellular integration for 3rd generation mobile communications systems," in *Proceedings Colloquium on Personal Communications in the 21st Century part I and II, ref no.1998/214 and 1998/242*, pp. 3/1–3/8, IEE, February 1998.

[12] P. Doany, "Global mobile personal communications by satellite (gmpcs) and other satellite services into the 21st century," in *Proceedings Colloquium on Personal Communications in the 21st Century part I and II, ref no.1998/214 and 1998/242*, pp. 2/1–2/6, IEE, February 1998.

[13] J. E. Padgett, C. G. Günther, and T. Hattori, "Overview of Wireless Personal Communications," *IEEE Communications Magazine*, vol. 33, pp. 28–41, January 1995.

[14] T. Edwards, "Technology trends for personal communications terminals," in *Proceedings Colloquium on Personal Communications in the 21st Century part I and II, ref no.1998/214 and 1998/242*, pp. 5/1–5/8, IEE, February 1998.

[15] E. Biglieri, G. Caire, and G. Taricco, "Coding and Modulation Under Power Constraints," *IEEE Personal Communications Magazine*, vol. 5, pp. 32–39, June 1998.

[16] T. S. Rappaport, *Wireless communications*. Upper Saddle River, New Jersey, USA: Prentice Hall, 1st ed., 1996.

[17] K. Raith and J. Uddenfeldt, "Capacity of Digital Cellular TDMA Systems," *IEEE Transactions on Vehicular Technology*, vol. 40, pp. 323–332, May 1991.

[18] L. Hanzo and R. Steele, "The Pan-European Mobile Radio System," *European Transactions on Communications*, vol. 5, pp. 245–276, March 1994.

[19] R. L. Pickholtz, D. L. Schilling, and L. B. Milstein, "Theory of Spread-Spectrum Communications-A Tutorial," *IEEE Transactions on Communications*, vol. 30, pp. 855–884, May 1982.

[20] R. C. Dixon, *Spread Spectrum Systems with Commercial Applications*. New York, USA: John Wiley and Sons, Inc., 3rd ed., 1994.

[21] A. J. Viterbi, *CDMA*. Reading, Massachusetts, USA.: Addison Weslay Publishing Company, 1st ed., 1995.

[22] P. Jung, P. W. Baier, and A. Steil, "Advantages of CDMA and Spread Spectrum Techniques over FDMA and TDMA in Cellular Mobile Radio Applications," *IEEE Transactions on Vehicular Technology*, vol. 42, pp. 357–364, August 1993.

[23] K. S. Gilhousen, I. M. Jacobs, R. Padovani, A. J. Viterbi, L. A. Weaver, and C. E. Wheatley, "On the Capacity of a Cellular CDMA System," *IEEE Transactions on Vehicular Technology*, vol. 40, pp. 303–312, May 1991.

[24] A. M. Viterbi and A. J. Viterbi, "Erlang Capacity of a Power Controlled CDMA System," *IEEE Transactions on Selected Areas in Communications*, vol. 11, pp. 892–899, August 1992.

[25] M. A. Mokhtar and S. C. Gupta, "Power Control Considerations for DS/CDMA Personal Communication Systems," *IEEE Transactions on Vehicular Technology*, vol. 41, pp. 479–487, November 1992.

[26] W. C. Y. Lee, "Overview of Cellular CDMA," *IEEE Transactions on Vehicular Technology*, vol. 40, pp. 291–302, May 1991.

[27] A. J. Viterbi, A. M. Viterbi, and E. Zehav, "Other-Cell Interference in Cellular Power-Controlled CDMA," *IEEE Transactions on Communications*, vol. 42, pp. 1501–1504, February 1994.

[28] R. Steele, *Mobile Radio Communications*. London, UK: Pentech Press Ltd., 1st ed., 1992.

[29] G. L. Turin, "Introduction to Spread-Spectrum Antimultipath Techniques and Their Application to Urban Digital Radio," *Proceedings of the IEEE*, vol. 68, pp. 328–353, March 1980.

[30] V. M. DaSilva, E. S. Sousa, and V. Jovanovic, "Effect of Multipath Propagation on the Forward Link of a CDMA Cellular System," *Wireless Personal Communications Journal*, vol. 1, pp. 33–41, 1994.

[31] G. L. Turin, "The Effects of Multipath and Fading on the Performance of Direct-Sequence CDMA Systems," *IEEE Transactions on Selected Areas in Communications*, vol. 2, pp. 597–603, July 1984.

[32] R. Kohno, "Spatial and Temporal Filtering for Co-Channel Interference in CDMA," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Oulu, Finland*, vol. 1, pp. 51–60, IEEE, 1994.

[33] Y. C. Yoon, R. Kohno, and H. Imai, "A Spread-Spectrum Multiaccess System with Cochannel Inteference Cancellation for Multipath Fading Channels," *IEEE Transactions on Selected Areas in Communications*, vol. 11, pp. 1067–1075, September 1993.

[34] L. K. Hansen and G. Xu, "A Hyperplane-Based Algorithm for the Digital Co-Channel Communicayions Problem," *IEEE Transactions on Information Theory*, vol. 43, pp. 1536–1548, September 1997.

[35] R. Lupas and S. Verdu, "Near-Far Resistance of Multiuser Detectors in Asynchronous Channels," *IEEE Transactions on Communications*, vol. 38, pp. 496–508, April 1990.

[36] A. J. Viterbi, A. M. Viterbi, and E. Zehavi, "Performance of Power-Controlled Wideband Terrestrial Digital Communication," *IEEE Transactions on Communications*, vol. 41, pp. 559–569, April 1993.

[37] W. C. Y. Lee, *Mobile Communications Engineering*. New York, USA: McGraw Hill, 2nd ed., 1997.

[38] J. G. Proakis and M. Salehi, *Communications Systems Engineering*. Englewood Cliffs, New Jersey, USA.: Prentice Hall International, 1st ed., 1994.

[39] R. M. Buehrer, N. S. Correal, and B. D. Woerner, "A Comparison of Multiuser Receivers for Cellular CDMA," in *Proceedings International Conference Globecom'96, London, UK*, pp. 1571–1577, IEEE, November 1996.

[40] R. Wichman and A. Hottinen, "Multiuser Detection for Downlink CDMA Communications in Multipath Fading Channels," in *Proceedings International Conference on Vehicular Technology, Phoenix, AZ, USA*, vol. 2, pp. 572–576, IEEE, May 1997.

[41] B. Sklar, "A primer on turbo code concepts," *IEEE Communications Magazine*, vol. 35, pp. 94–102, December 1997.

[42] J. L. Massey, "Information Theory Aspects of Spread-Spectrum Communications," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Oulu, Finland*, pp. 16–21, IEEE, 1994.

[43] C. E. Shannon, "Communications in the Presence of Noise," *Proceedings of the IRE*, vol. 37, pp. 10–21, January 1949.

[44] R. A. Scholtz, "The Evolution of Spread-Spectrum Multiple-Access Communications," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Oulu, Finland*, pp. 4–13, IEEE, 1994.

[45] S. Moshavi, "Multi-User Detection for DS-CDMA Communications," *IEEE Communications Magazine*, vol. 34, pp. 124–136, October 1996.

[46] J. G. Proakis, *Digital Communications*. New York, USA: McGraw-Hill, 3rd ed., 1993.

[47] G. H. Golub and C. F. V. Loan, *Matrix Computations*. Baltimore, MD, USA: John Hopkins University Press, 3rd ed., 1996.

[48] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey, USA.: Prentice Hall International, 3rd ed., 1996.

[49] T. Kailath, "Correlation Detection of Signals Perturbed by a Random Channel," *IEEE Transactions on Information Theory*, vol. 6, pp. 361–366, June 1960.

[50] T. Kailath, "On Multilink and Multidimensional Channels," *IEEE Transactions on Information Theory*, vol. 8, pp. 260–262, April 1962.

[51] L. Barbosa, "Maximum Likelihood Sequence Estimators: A Geometric View," *IEEE Transactions on Information Theory*, vol. 35, pp. 419–427, March 1989.

[52] P. Jung and P. D. Alexander, "A Unified Approach to Multiuser Detectors for CDMA and Their Geometrical Interpretations," *IEEE Transactions on Selected Areas in Communications*, vol. 14, pp. 1595–1601, October 1996.

[53] L. K. Rasmussen, T. J. Lim, and P. D. Alexander, "A New Geometrical Interpretation of the Decorrelator in Multiuser CDMA," in *Proceedings International Conference Globecom'96, London, UK*, vol. 3, pp. 1561–1565, IEEE, November 1996.

[54] S. Haykin, *Digital Communications*. New York, USA: John Wiley and Sons., 1st ed., 1988.

[55] J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering*. New York, USA: John Wiley and Sons Inc., 1st ed., 1965.

[56] B. Lyons, B. Mazur, J. Lodge, M. Mohrer, S. Crozier, and L. Erup, "A High Capacity Third-Generation Mobile Satellite System Design," *European Transactions on Communications*, vol. 9, pp. 337–352, August 1998.

[57] T. Dohi, Y. Okumura, and F. Adachi, "Further Results on Field Experiments of Coherent Wideband DS-CDMA Mobile Radio," *IEICE Transactions : Communication*, vol. E81-B, pp. 1239–1246, June 1998.

[58] A. Toskala, J. P. Castro, E. Dahlman, M. Latva, and T. Ojanperrä, "FRAMES FMA2 Wideband-CDMA for UMTS," *European Transactions on Communications*, vol. 9, pp. 325–335, August 1998.

[59] R. L. Pickholtz, L. B. Milstein, and D. L. Schilling, "Spread Spectrum for Mobile Communications," *IEEE Transactions on Vehicular Technology*, vol. 40, pp. 313–322, May 1991.

[60] B. Miller, "Satellites free the mobile phone," *IEEE Spectrum*, vol. 35, pp. 26–35, March 1998.

[61] L. M. Garth and H. V. Poor, "Detection of Non-Gaussian Signals: A Paradigm for Modern Statistical Signal Processing," *Proceedings of the IEEE*, vol. 82, pp. 1061–1095, July 1994.

[62] R. Price, "A Communication Technique for Multipath Channels," *Proceedings of the IRE*, vol. 48, pp. 555–570, March 1958.

[63] S. Verdu, "Optimum Multiuser Asymtotic Efficiency," *IEEE Transactions on Communications*, vol. 34, pp. 890–897, September 1986.

[64] U. Mitra and H. Poor, "Adaptive Receiver Algorithms for Near-Far Resistant CDMA," *IEEE Transactions on Communications*, vol. 43, pp. 1713–1724, March 1995.

[65] D. P. Taylor, G. M. Vitetta, B. D. Hart, and A. Mämmelä, "Wireless Channel Equalisation," *European Transactions on Communications*, vol. 9, pp. 117–134, March 1998.

[66] S. U. H. Qureshi, "Adaptive Equalization," *Proceedings of the IEEE*, vol. 73, pp. 1349–1387, September 1985.

[67] B. Widrow and R. Winter, "Neural Nets for Adaptive Filtering and Adaptive Patterns Recognition," *IEEE Computers*, vol. 21, pp. 25–39, March 1988.

[68] J. G. Proakis, "Adaptive Equalization for TDMA Digital Mobile Radio," *IEEE Transactions on Vehicular Technology*, vol. 40, pp. 333–341, May 1991.

[69] S. Verdu, "Adaptive Multiuser Detection," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Oulu, Finland*, vol. 1, pp. 43–45, IEEE, 1994.

[70] U. Mitra and H. V. Poor, "Analysis of an Adaptive Decorrelating Detector for Synchronous CDMA Channels," *IEEE Transactions on Communications*, vol. 44, pp. 257–268, February 1996.

[71] I. A. Glover and P. M. Grant, *Digital Communications*. Hemel Hempstead, UK: Prentice Hall Europe, 1st ed., 1997.

[72] J. M. Bernardo and A. F. M. Smith, *Bayesian Theory*. Chichester, UK: J. Wiley and Sons Inc., 1st ed., 1994.

[73] G. D. Forney, "Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference," *IEEE Transactions on Information Theory*, vol. IT-18, pp. 363–378, May 1972.

[74] I. N. Andersen, "Sample-Whitened Matched Filters," *IEEE Transactions on Information Theory*, vol. 19, pp. 653–660, September 1973.

[75] G. D. Forney, "The Viterbi Algorithm," *Proceedings of the IEEE*, vol. 61, pp. 268–278, March 1973.

[76] K. Abend and B. D. Fritchman, "Statistical Detection for Communication Channels with Intersymbol Interference," *Proceedings of the IEEE*, vol. 58, pp. 779–785, May 1970.

[77] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate," *IEEE Transactions on Information Theory*, vol. 20, pp. 284–287, March 1974.

[78] B. Mulgrew, "Nonlinear Signal Processing for Adaptive Equalisation and Multi-User Detection," in *IX European Signal Processing Conference EUSIPCO'98, Rhodes, Greece*, EURASIP, September 1998.

[79] A. P. Clark, *Equalisers for Digital Modems*. London, UK: Pentech Press, 1st ed., 1985.

[80] D. G. Messerschmitt, "A Geometric Theory of Intersymbol Interference," *The Bell System Technical Journal*, vol. 52, pp. 1483–1539, 1973.

[81] M. Schetzen, "Nonlinear System Modeling Based on the Wiener Theory," *Proceedings of the IEEE*, vol. 69, pp. 1557–1573, December 1981.

[82] V. J. Mathews, "Adaptive Polynomial Filters," *IEEE Signal Processing Magazine*, pp. 10–26, July 1991.

[83] E. Kreyszig, *Advanced Engineering Mathematics*. Singapore: John Wiley & Sons, 7th ed., 1997.

[84] S. Benedetto and E. Biglieri, "Nonlinear Equalization of Digital Satellite Channels," *IEEE Transactions on Selected Areas in Communications*, vol. 1, pp. 57–62, January 1983.

[85] R. D. Nowak and B. D. V. Veen, "Volterra Filter Equalization: A Fixed Point Approach," *IEEE Transactions on Signal Processing*, vol. 45, pp. 377–388, February 1997.

[86] I. Pitas and A. N. Venetsanopoulos, *Nonlinear Digital Filters*. Norwell, Ma, USA: Kluwer Academic Publishers, 1st ed., 1990.

[87] K. D. Rao and D. C. Reddy, "New method of designing adaptive nonlinear filters," *Proceedings-F of the IEE : Radar and Signal Processing*, vol. 138, pp. 513–519, October 1991.

[88] S. W. Nam and E. J. Powers, "Application of Higher Order Spectral Analysis to Cubically Nonlinear System Identification," *IEEE Transactions on Signal Processing*, vol. 42, pp. 1746–1765, July 1994.

[89] J. Sid-Sueiro and A. R. Figueiras-Vidal, "Channel equalization with neural networks," in *Digital Signal Processing in Telecommunications* (A. R. Figueiras-Vidal, ed.), Berlin: Springer, 1996.

[90] G. J. Gibson, S. Siu, and C. F. N. Cowan, "The Application of Nonlinear Structures to the Reconstruction of Binary Signals," *IEEE Transactions on Signal Processing*, vol. 39, pp. 1877–1884, August 1991.

[91] J. Tsimbinos and K. V. Lever, "The Computational Complexity of Nonlinear Compensators Based on the Volterra Inverse," in *Proceedings Signal Processing Workshop on Statistical Signal and Array Processing, Corfu, Greece*, vol. 1, pp. 387–390, IEEE, June 1996.

[92] K. Georgoulakis and S. Theodoridis, "Efficient clustering techniques for channel equalization in hostile environments," *Signal Processing (Eurasip)*, vol. 58, pp. 153–164, 1997.

[93] J. Hertz, R. G. Palmer, and A. S. Krogh, *Introduction of the Theory of Neural Computing*. Redwood City, Cal., USA: Addison-Weslay, 1st ed., 1991.

[94] D. E. Rumelhart, J. L. McClellend, and the PDP Research Group, *Parallel distributed processing, Volume 1: Foundations*. Cambridge, Massachusetts, USA.: MIT Press, 1st ed., 1986.

[95] C. Z. W. Hassell, G. J. Gibson, and B. Mulgrew, "A Constructive Algorithm for Neural Network Design with Application to Channel Equalization," in *Proceeedings IMA Conference on Applications of Combinatorial Mathematics, Oxford, UK*, IEE, December 1994.

[96] S. I. Gass, *Linear Programming*. New York, USA: McGraw-Hill, 1st ed., 1958.

[97] D. S. Broomhead and D. Lowe, "Multivariable Functional Interpolation and Adaptive Networks," *Complex Systems*, vol. 2, pp. 321–355, 1988.

[98] T. Poggio and F. Girosi, "Networks for Approximation and Learning," *Proceedings of the IEEE*, vol. 78, pp. 1481–1497, September 1990.

[99] S. Chen, G. Gibson, C. Cowan, and P.M.Grant, "Reconstruction of binary signals using an adaptive radial-basis-function equalizer," *Signal Processing (Eurasip)*, vol. 22, pp. 77–93, January 1991.

[100] S. Chen, B. Mulgrew, and P. M. Grant, "A Clustering Technique for Digital Communications Channel Equalization Using Radial Basis Function Networks," *IEEE Transactions on Neural Networks*, vol. 4, pp. 570–579, July 1993.

[101] S. Theodoridis, C. F. N. Cowan, C. P. Callender, and C. M. S. See, "Schemes for equalisation of communication channels with nonlinear impairments," *IEE Proceedings - Communication*, vol. 142, pp. 165–171, June 1995.

[102] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York, USA: J. Wiley and Sons, 1st ed., 1973.

[103] I. Cha and S. A. Kassam, "Interference cancellation using radial basis function networks," *Signal Processing (Eurasip)*, vol. 47, pp. 247–268, 1995.

[104] S. Verdu, "Demodulation in the Presence of Multiuser Interference: Progress and Misconceptions," in *Intelligent Methods in Signal Processing and Communications*, pp. 15–44, Birkhauser, Boston, USA, 1997.

[105] S. Verdu, "Minimum Probability of Error for Asynchronous Gaussian Multiple-Access Channels," *IEEE Transactions on Information Theory*, vol. IT-32, pp. 85–96, January 1986.

[106] P. M. Grant, G. J. R. Povey, and R. D. Pringle, "Performance of a Spread Spectrum Rake Receiver Design," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications*, pp. 71–74, IEEE, November 1992.

[107] D. G. M. Cruickshank, "Optimal and Adaptive FIR Filter Receivers for DS-CDMA," in *Proceedings International Symposium on Personal Indoor and Mobile Communications*, pp. 1339–1343, IEEE, September 1994.

[108] D. G. M. Cruickshank, "Suppression of multiple access interference in a DS-CDMA system using Wiener filtering and parallel cancellation," *IEE Proceedings - Communication*, vol. 143, pp. 226–230, August 1996.

[109] U. Madhow and M. L. Honig, "MMSE Interference Suppression for Direct Sequence Spread-Spectrum CDMA," *IEEE Transactions on Communications*, vol. 42, pp. 3178–3188, December 1994.

[110] R. Lupas and S. Verdu, "Linear Multiuser Detectors for Synchronous Code-Division Multiple-Access Channels," *IEEE Transactions on Information Theory*, vol. 35, pp. 123–136, January 1989.

[111] S. Moshavi, E. G. Kanterakis, and D. L. Schilling, "Multistage Linear Receivers for DS-CDMA Systems," *Int. Journal of Wireless Information Networks*, vol. 3, pp. 1–17, January 1996.

[112] D. G. M. Cruickshank, "Radial basis function receivers for DS-CDMA," *IEE Electronics Letters*, vol. 32, pp. 188–190, February 1996.

[113] J. M. Holtzman, "DS/CDMA Successive Interference Cancellation," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Oulu, Finland*, vol. 1, pp. 69–78, IEEE, 1994.

[114] D. Divsalar, M. Simon, and D. Raphaeli, "A New Approach to Parallel Interference Cancellation for CDMA," in *Proceedings International Conference Globecom'96, London, UK*, vol. 3, pp. 1452–1457, IEEE, November 1996.

[115] M. K. Varanasi and B. Aazhang, "Multistage Detection in Asynchronous Code-Division Multiple-Access Communications," *IEEE Transactions on Communications*, vol. 38, pp. 509–519, April 1990.

[116] G. I. Kechriotis and E. S. Manolakos, "Hopfield Neural Network Implementation of the Optimal CDMA Multiuser Detector," *IEEE Transactions on Neural Networks*, vol. 7, pp. 131–141, January 1996.

[117] J. J. Hopfield and D. W. Tank, "Computing with Neural Circuits: A Model," *Science*, vol. 233, pp. 625–633, August 1986.

[118] T. Miyajima and K. Yamanaka, "A Nonlinear Blind Adapative Receiver for DS/CDMA Systems," *IEICE Transactions : Communication*, vol. E79-A, pp. 2081–2083, December 1996.

[119] B. Aazhang, B. Paris, and G. Orsak, "Neural Networks for Multiuser Detection in Code-Division Multiple-Access Communications," *IEEE Transactions on Communications*, vol. 40, pp. 1212–1222, July 1992.

[120] M. L. Brady, R. Raghavan, and J. Slawny, "Back Propagation Fails to Separate Where Perceptrons Succeed," *IEEE Transactions on Circuits and Systems*, vol. 36, pp. 665–674, May 1989.

[121] S. Verdu, "Computational Complexity of Optimum Multiuser Detection," *Algorithmica*, vol. 4, pp. 303–312, 1989.

[122] T. Nagaosa, T. Miyjima, and T. Hasegama, "Multiuser Detection Using a Hopfield Network in Asynchronous M-ary SSMA Communications," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Mainz, Germany*, vol. 2, pp. 837–841, IEEE, September 1996.

[123] G. I. Kechriotis and E. S. Manolakos, "A Hybrid Digital Signal Processing-Neural Network CDMA Multiuser Detection Scheme," *IEEE Transactions on Circuits and Systems*, vol. 43, pp. 96–104, February 1996.

[124] S. H. Bang and B. J. Sheu, "A Neural Network for Detection of Signals in Communications," *IEEE Transactions on Circuits and Systems*, vol. 43, pp. 644–655, August 1996.

[125] T. Miyajima, T. Hasegawa, and M. Haneishi, "On the Multiuser Detection Using a Neural Network in Code-Division Multiple-Access Communications," *IEICE Transactions : Communication*, vol. E76-B, pp. 961–968, August 1993.

[126] W. G. Teich and M. Seidl, "Code Division Multiple Access Commuications: Multiuser Detection based on a Recurrent Neural Network Structure," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Mainz, Germany*, vol. 2, pp. 979–984, IEEE, September 1996.

[127] S. H. Yoon and S. S. Rao, "High Performance Neural Network Based Multiuser Detector in CDMA," in *Proceedings International Conference on Acoustics, Speech and Signal Processing, Munich, Germany*, vol. 4, pp. 3357–3360, IEEE, April 1997.

[128] U. Mitra and H. V. Poor, "Neural Network Techniques for Adaptive Multiuser Demodulation," *IEEE Transactions on Selected Areas in Communications*, vol. 12, pp. 1460–1470, December 1994.

[129] R. Kohno, P. Rapajcic, and B.S.Vucetic, "An Overview of Adaptive Techniques for Interference Minimization in CDMA Systems," *Wireless Personal Communications Journal*, vol. 1, pp. 3–21, 1994.

[130] P. B. Rapajcic and B. S. Vucetic, "Adaptive Receiver Structures for Asynchronous CDMA Systems," *IEEE Transactions on Selected Areas in Communications*, vol. 12, pp. 685–697, May 1994.

[131] D. Guo, L. K. Rasmussen, S. Sun, T. J. Lim, and C. Cheah, "MMSE-Based Linear Parallel Interference Cancellation in CDMA," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Sun City, South Africa*, vol. 3, pp. 917–921, IEEE, September 1998.

[132] M. Minsky and S. Papert, *Perceptrons : an introduction to computational geometry*. Cambridge, Mass., USA: MIT Press, 1st ed., 1969.

[133] B. Widrow and M. A. Lehr, "30 Years of Adaptive Neural Networks: Perceptron, Madaline, and Backpropagation," *Proceedings of the IEEE*, vol. 78, pp. 1411–1442, September 1990.

[134] R. C. Singleton, "A Test for Linear Separability as Applied to Self-Organising Machines," in *Conference on Self-Organising Systems* (G. D. G. M. C. Youvits, G. T. Jacobi, ed.), pp. 503–524, Wash. DC: Spartan Books, 1962.

[135] W. H. Highleyman, "Linear Decision Functions, with Application to Pattern Recognition," *Proceedings of the IRE*, pp. 1501–1514, June 1962.

[136] H. J. Greenberg and A. G. Konheim, "Linear and Nonlinear Methods in Pattern Classification," *IBM Journal Research and Development*, vol. 8, pp. 299–307, July 1964.

[137] T. M. Cover, "Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition," *IEEE Transactions on Electronic Computers*, vol. 14, pp. 326–334, June 1965.

[138] O. L. Mangasarian, "Linear and Nonlinear Separation of Patterns by Linear Programming," *Operations Research*, vol. 13, pp. 444–452, 1965.

[139] O. L. Mangasarian, W. N. Street, and W. H. Wolberg, "Breast Cancer Diagnosis and Prognosis via Linear Programming," *Tech. Report 94-10, University of Wisconsin, Dept. of Computer Science, Madison, WI, USA*, December 1994.

[140] R. P. Lippmann, "An Introduction to Computing with Neural Nets," *IEEE ASSP Magazine*, vol. 4, pp. 4–22, April 1987.

[141] S. Amari, "Mathematical Foundation of Neurocomputing," *Proceedings of the IEEE*, vol. 78, pp. 1443–1463, September 1990.

[142] D. R. Hush and B. G. Horne, "Progress in Supervised Neural Networks," *IEEE Signal Processing Magazine*, vol. 10, pp. 8–39, January 1993.

[143] S. Haykin, *Neural Networks*. Upper Sadle River, New Jersey, USA.: Prentice Hall International, 1st ed., 1994.

[144] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford, UK: Oxford University Press, 1st ed., 1995.

[145] B. Widrow, D. E. Rumelhart, and M. A. Lehr, "Neural Networks: Applications in Industry, Business and Science," *Communications of the ACM*, vol. 37, pp. 93–105, March 1994.

[146] T. H. Cormen, C. E. Leiseson, and R. L. Rivest, *Introduction to Algorithms*. New York: MIT Press, McGraw-Hill, 1st ed., 1996.

[147] F. P. Preparata and M. I. Shamos, *Computational Geometry: An Introduction*. New York, USA: Springer, 1st ed., 1985.

[148] T. M. Cover and P. E. Hart, "Nearest Neighbor Pattern Classification," *IEEE Transactions on Information Theory*, vol. 13, pp. 21–27, January 1967.

[149] R. D. Short and K. Fukunaga, "The Optimal Distance Measure for Nearest Neighbor Classification," *IEEE Transactions on Information Theory*, vol. 27, pp. 622–627, September 1981.

[150] J. H. Winters and C. Rose, "Minimum Distance Automata in Parallel Networks for Optimum Classification," *Neural Networks*, vol. 2, pp. 127–132, 1989.

[151] N. J. Nilsson, *Learning machines : foundations of trainable pattern-classifying systems*. New York, USA: McGraw-Hill, 1st ed., 1965.

[152] P. W. Cooper, "The hyperplane in pattern recognition," *Cybernetica*, vol. 5, pp. 215–238, 1962.

[153] D. F. Specht, "Generation of Polynomial Discriminant Functions for Pattern Recognition," *IEEE Transactions on Electronic Computers*, vol. 16, pp. 308–319, June 1967.

[154] R. C. Gonzales and R. E. Woods, *Digital Image Processing*. Reading, MA, USA: Addison-Wesley, 1st ed., 1992.

[155] T. W. Anderson, *An Introduction to Multivariate Statistical Analysis*. New York, USA: J. Wiley and Sons, 2nd ed., 1984.

[156] P. W. Cooper, "The Hypershere in Pattern Recognition," *Information and Control*, vol. 5, pp. 324–346, 1962.

[157] C. B. Barber, D. O. Dobkin, and H. T. Huhdaanpaa, "The Quickhall algorithm for convex hulls," *ACM Transcations on Mathematical Software*, vol. 22, pp. 469–483, December 1996.

[158] C. A. Micchelli, "Interpolation of Scattered Data: Distance Matrices and Conditionally Definite Functions," *Constructive Approximation*, vol. 2, pp. 11–22, 1986.

[159] M. Schetzen, *The Volterra and Wiener theories of nonlinear systems*. New York, USA: John Wiley and Sons Inc., 1st ed., 1980.

[160] C. L. Nikias and J. M. Mendel, "Signal Processing with Higher-Order Spectra," *IEEE Signal Processing Magazine*, vol. 10, pp. 10–37, July 1993.

[161] Personal communication with B. Worton from STATLAB at University of Edinburgh.

[162] S. M. Ross, *Introduction to probability and statistics for engineers and scientists*. New York Chichester: Wiley, 1st ed., 1987.

[163] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. Singapore: McGraw Hill, 3rd ed., 1991.

[164] R. Tanner and D. G. M. Cruickshank, "Nonlinear Volterra Filter Receiver for DS-CDMA," in *Proceedings 4th International Conference on Mathematics in Signal Processing, IMA, University of Warwick, UK*, IEE, December 1996.

[165] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, USA: Prentice-Hall, 1st ed., 1993.

[166] R. Gold, "Optimal Binary Sequences for Spread Spectrum Multiplexing," *IEEE Transactions on Information Theory*, vol. 13, pp. 619–621, October 1967.

[167] R. Tanner and D. G. M. Cruickshank, "Volterra Based Receivers for DS-CDMA," in *Proceedings International Symposium on Personal, Indoor and Mobile Radio Comminications, Helsinki, Finland*, vol. 3, pp. 1166–1170, IEEE, September 1997.

[168] J. Tsimbinos, *Identification and compensation of nonlinear distortion*. PhD thesis, University of South Australia, School of Electronic Engineering, Inst. of Telecommunications Research, The Levels, Sout Australia 5095, 1995.

[169] W. Mohr and M.Kottkamp, "Downlink Performance of IS-95 DS-CDMA under Multipath Propagation Conditions," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Mainz, Germany*, vol. 3, pp. 1063–1067, IEEE, September 1996.

[170] J. Blanz, A. Klein, M. Nasshan, and A. Steil, "Performance of a Cellular Hybrid C/TDMA Mobile Radio System Applying Joint Detection and Coherent Receiver Antenna Diversity," *IEEE Transactions on Selected Areas in Communications*, vol. 12, pp. 568–579, May 1994.

[171] P. Jung and J. Blanz, "Joint Detection with Coherent Receiver Antenna Diversity in CDMA Mobile Radio Systems," *IEEE Transactions on Vehicular Technology*, vol. 44, pp. 76–88, February 1995.

[172] S. J. Orfanidis, *Optimum Signale Processing*. New York, USA.: McGraw-Hill, 2nd ed., 1988.

[173] B. Mulgrew, "Applying Radial Basis Functions," *IEEE Signal Processing Magazine*, vol. 13, pp. 50–65, March 1996.

[174] H. V. Poor and S. Verdu, "Probability of Error in MMSE Multiuser Detection," *IEEE Transactions on Information Theory*, vol. 43, pp. 858–871, May 1997.

[175] M. Junatti and S. Glisic, "Advanced CDMA for wireless communications," in *Wireless Communications: TDMA vs. CDMA* (S. Glisic and P. Leppänen, eds.), pp. 447–490, Dortrecht, NL: Kluwer, 1997.

[176] G. E. Bottomley, "Signature Sequence Selection in a CDMA System with Orthogonal Coding," *IEEE Transactions on Vehicular Technology*, vol. 42, pp. 62–68, February 1993.

[177] G. B. Dantzig, *Linear Programming and extensions*. Princeton, USA: Princeton University Press, 2nd ed., 1965.

[178] D. Gale, "The basic theorems of real linear equations, inequalities, linear programming, and game theory," *Naval Research Logistics Quarterly, Wash. NY*, vol. 3, pp. 193–200, 1956.

[179] O. L. Mangasarian, *Nonlinear Programming*. Philadelphia, PA, USA: SIAM, preprint ed., 1994.

[180] F. W. Smith, "Pattern Classifier Design by Linear Programming," *IEEE Transactions on Computers*, vol. 17, pp. 367–372, April 1968.

[181] T. Ibaraki and S. Muroga, "Adaptive Linear Classifier by Linear Programming," *IEEE Transactions on systems Science and Cybernetics*, vol. 6, pp. 53–62, January 1970.

[182] J. S. Shawe-Taylor and D. A. Cohen, "Linear Programming Algorithm for Neural Networks," *Neural Networks*, vol. 3, pp. 575–582, 1990.

[183] O. L. Mangasarian, "Multisurface Method of Pattern Separation," *IEEE Transactions on Information Theory*, vol. 14, pp. 801–807, November 1968.

[184] R. Takiyama, "A LEARNING PROCEDURE FOR MULTISURFACE METHOD OF PATTERN SEPARATION," *Pattern Recognition*, vol. 12, pp. 75–82, 1980.

[185] K. P. Bennett and O. L. Mangasarian, "Neural Network Training via Linear Programming," *Tech. Report 948, University of Wisconsin, Dept. of Computer Science, Madison, WI, USA*, 1990.

[186] K. P. Bennett and O. L. Mangasarian, "Robust Linear Programming Discrimination of two Linearly Inseparable Sets," *Tech. Report 1054a, University of Wisconsin, Dept. of Computer Science, Madison, WI, USA*, 1991.

[187] K. P. Bennett and O. L. Mangasarian, "Multicategory Discrimination via Linear Programming," *Tech. Report 1127, University of Wisconsin, Dept. of Computer Science, Madison, WI, USA*, 1992.

[188] C. Z. W. Hassell, G. J. Gibson, and B. Mulgrew, "Constructive Neural Network Design for the Solution of Two State Classification Problems with Applications to Channel Equalization," in *Proceedings Workshop on Neural Networks for Signal Processing, Boston, USA*, IEEE, September 1995.

[189] R. Takiyama, "MULTIPLE THRESHOLD PERCEPTRON," *Pattern Recognition*, vol. 10, pp. 27–30, 1978.

[190] J. Monn and T. Jeon, "Sequence Detection for Binary ISI Channels Using Signal-Space Partitioning," *IEEE Transactions on Communications*, vol. 46, pp. 891–901, July 1998.

[191] I. Band and D. G. M. Cruickshank, "The Nearest Neighbour Receiver for the Downlink of a DS-CDMA System," in *Proceedings International Symposium on Spread Spectrum Techniques and Applications, Sun City, South Africa*, IEEE, September 1998.

[192] F. Aurenhammer, "Voronoi Diagrams - A Survey of a Fundamental Geometric Data Structure," *ACM Computing Surveys*, vol. 23, pp. 345–405, September 1991.

[193] L. Wei, L. K. Rasmussen, and R. Wyrwas, "Near Optimum Tree-Search Detection Schemes for Bit-Synchronous Multiuser CDMA Systems over Gaussian and Two-Path Rayleight-Fading Channels," *IEEE Transactions on Communications*, vol. 45, pp. 691–700, June 1997.

[194] A. Roy, L. S. Kim, and S. Mukhopadhyay, "A Polynomial Time Algorithm for the Construction and Training of a Class of Multilayer Perceptrons," *Neural Networks*, vol. 6, pp. 535–545, 1993.

[195] A. Roy, S. Govil, and R. Miranda, "An Algorithm to Generate Radial Basis Function (RBF)-like Nets for Classification Problems," *Neural Networks*, vol. 8, pp. 179–201, 1995.

[196] J. Bruck and M. Blaum, "Neural Networks, Error-Correcting Codes, and Polynomials over the Binary n-Cube," *IEEE Transactions on Information Theory*, vol. 35, pp. 976–987, September 1989.

[197] A. O. Steinhardt, "Housholder Transformations in Signal Processing," *IEEE Signal Processing Magazine*, pp. 4–12, July 1988.

[198] S. J. Orfanidis, "Gram-Schmidt Neural Nets," *Neural Computation*, vol. 2, pp. 116–126, 1990.