Développement de récepteurs avancés pour les systèmes de communication mobile de type WCDMA et HSDPA

Ahmet Ba

Advanced Receivers for
High Speed Downlink Packet Access in UMTS

by

Ahmet BAŞTUĞ

BSc, Middle East Technical University, Ankara TURKEY, 1999
MSc, Boğaziçi University, Istanbul TURKEY, 2002

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Electronics and Communications

in the

Graduate Division
of the
Ecole Nationale Supérieure des Télécommunications in Paris

Committee in charge:
Yves Grenier, Chair
Marie-Laure Boucheret (INP-ENSEEIHT, France, Reporter)
Marc Moonen (K.U. Leuven, Belgium, Reporter)
Yves Grenier (E.N.S.T. Paris, France, Examiner)
Emre Aktaş (Hacettepe University, Turkey, Examiner)
Giuseppe Montalbano (Philips Semiconductors, France, Examiner)
Dirk Slock (Eurecom, France, Thesis Director)
Fabrizio Tomatis (Philips Semiconductors, France, Invited)

Spring May 2006

Advanced Receivers for
High Speed Downlink Packet Access in UMTS

Acknowledgments

The first sign of the oncoming adventure popped up when I received an e-mail from my MSc advisor Professor Bülent Sankur from Boğaziçi University in Istanbul, saying that I should give consideration to doing Phd, preferably abroad. Having given no thought to such a thing before, having a well-paid job in industry and the last but not the least being married with a one-year old kid, at first instant I considered this as a crazy idea. A few days later, just by coincidence, I discovered that one of my good friends from the high school, Serkan Yeşildağ, was following a Phd program at Eurecom Institute in France. Discovering that Eurecom has a good mix of industrial and academic perspectives fitting my profile, though not being very serious, I decided to "fish" by a rifle shot applying to Professor Dirk Slock. I noticed what kind of trouble I put myself and my family in, only when I got a rapid acceptance from him the next day. At least for the sake of being consistent with my earlier actions, I had to continue... Many thanks to Serkan and Professor Bülent Sankur for the kick-off.

The first year passed in a large office shared with Maxime Guillaud, Albert Guillen Fabregas and Mari Kobayashi whom I would like to thank for their warm friendship. Then noticing that I was bored with sole research, Dirk routed me back to industry to be affiliated with Philips Semiconductors in Sophia Antipolis as an engineer where in addition to technical collaborations I also built intimate friendships with Fabrizio Tomatis, Andrea Ancora, Eric Alliot, Pierre Demaj, Calogero Bona, Stefania Sesia, Giuseppe Montalbano, Marc Soler, Rida Chaggara, Lili and many others. Thank you all simply for everything we shared. I would like to also thank Matthew Baker and Tim Moulsley from Philips Research at RedHill and Frank Heinle, Stefan Mueller and Udo Wachsmann from Philips Semiconductors at Nuremberg for our professional collaborations during my visits to UK and Germany.

The life in France was, of course, not very easy for the family due to the language difficulties and the famous French bureaucracy. Our *angel* Melek ran to our help whenever and wherever we needed her. Cédric was another intimate relative, helping us and bringing *French* blood to the family. I would like to also thank both of them and Rida Chaggara for taking care of the French translations in the thesis.

I would like to thank Yves Grenier, Marie-Laure Boucheret, Marc Moonen, Emre Aktaş, Giuseppe Montalbano and Fabrizio Tomatis for accepting to be the jury members,

some coming from very long distances.

Coming to Professor Slock, it is needless to mention too much about his research enthusiasm and strong supervision, which is quite well known. What I would like to in fact stress more is his nice humanitarian side, which many other professors do not have at a comparable level. I believe I benefited a lot from the *hours-lasting* chats I made with him on totally non-technical issues. In addition to expressing my deep gratitude, I would like him to also know that I want our friendship to last for many more years.

Finally I would like to say that I am grateful to my wife Elif for putting up with me during the last three years and my son Hüseyin who apparently was not giving any meaning to why his father was often reading those boring papers, writing some more himself and sitting in front of the laptop instead of playing with him more often.

Preface

Technology and society has always had profound impact on each other. We will not perhaps be committing a serious crime if we say that telecommunications has had the lion's share in this interaction. The desire to hear the speeches of our beloveds or at least to receive written words informing that they are safe is an incomparable *human* need.

The departures of telegraphy services by William Coke and Charles Wheatstone in UK in 1839 and by Samuel Morse in USA in 1844 were the important milestones that kicked off the ever-lasting telecommunications adventure. The invention of the telephone followed in 1876 simultaneously by Alexander Graham Bell and Elisha Gray. These two breakthrough communications means both suffered from attenuation and dispersion problems. Localized to the *man-made* cable transmission medium, it became possible to overcome these challenges in the following decades by the inventions of inductive loadings and electronic amplifiers [1].

At the wireless side the inventions from genius minds triggered others like a chain reaction. Faraday's discovery of electromagnetic induction in 1831 was followed by Maxwell's wave equations in 1864 and Hertz's 1886 lab demonstration of the wave character of the electrical transmission through space. The competition in the following years to turn these scientific findings into a commercial success was realized by Marconi in 1896 in the form of a *digital* wireless telegraphy equipment. Edwin Howard Armstrong perhaps deserves even more credit due to his inventions of frequency modulation (FM) in 1933 and the super-heterodyne receiver, which together enabled high quality radio broadcasting.

The two major problems associated with wireless transmission are fading and interference. Unlike the cable medium the engineers have no luxury to change the characteristics of the wireless channel to suppress the influence of these degradations. Sophisticated *signal processing* techniques should be used to *live with them*.

It is essential for our motivation to mention three important profiles from history

who built the foundations of modern communications signal processing: Norbert Wiener who invented the Wiener (LMMSE) filter, Bernard Widrow who derived the adaptive LMS algorithm from the Wiener filter and finally Robert Lucky who invented the decision directed adaptive equalization benefiting from the LMS algorithm [2, 3]. Their principles have always been and still today are very useful recipes for solving a majority of challenges in wired and wireless communications. Consider the Internet for example the foundations of which were put by Paul Baran in 1964 with the invention of packet switching. The major bottleneck for its initial acceptance to the *wired* medium was the high bit error rates over a few kbps of transmission rate. This was overcome by the adaptive modem from Robert Lucky [3].

Coming to today we are facing similar challenges to carry the Internet services to the *wireless* domain. The central theme in HSDPA compliant 3G mobile terminal development is the design of an LMMSE chip equalizer for which there have recently been many sophisticated proposals. Since the physical constraints such as channel fading and multi access interference complicate the situation to many orders of magnitude larger extents compared to the era of Widrow and Lucky, the simple LMS algorithm seems to be too simple to be a proper choice. The author of this thesis is nevertheless a strong believer that adaptive equalizers from LMS families are still the best choices when one takes into account all the other design constraints such as the limited battery and the processing capability of mobile terminals, the short design periods and the usage of specific target implementation platforms. Because of this belief, part of the thesis work was shaped towards equalizer design around the LMS algorithm.

Equalizers were originally developed in single user transmission contexts. They are very effective to suppress the channel dispersion effects. However in a multiuser transmission context such as CDMA they are ignorant about the user subspace structures. More effective approach is hence multiuser detection, which explicitly explores the subspaces in order to subtract out the undesired signal components from the received signal. The research activities in this subject started in 1986 with the Maximum Likelihood receiver by Verdú [4]. Since then there have been several works which attacked the problem in some suboptimal but less complex ways. However most of these approaches focused on the uplink since multiuser detectors in general require several side information such as the codes and the symbol amplitudes of all the users. There have been techniques focusing also on the downlink, but almost all of them again assumed the knowledge of all the active user symbol powers and channelization codes, which is totally unrealistic. In this thesis we are also proposing a

multiuser detector architecture for HSDPA compliant mobile terminals which is different in the practical sense that assumptions on unknown quantities are kept to minimum.

Apart from equalization and multiuser detection we also cover the topic of pilot-aided channel estimation which is a major supporting functionality for parametric equalization and multiuser detection techniques. We aimed exploiting all the training sequences provided by the system.

The thesis work was done in an industrial development context at Philips Semiconductors premises in Sophia Antipolis, France. Therefore the main concerns for the chosen topics were applicability in the short term and suitability to proprietary vector processor development platforms [5, 6].

Abstract

Advanced Receivers for
High Speed Downlink Packet Access in UMTS

by

Ahmet BAŞTUĞ
Doctor of Philosophy in Electronics and Communications

Ecole Nationale Supérieure des Télécommunications in Paris

Yves Grenier

The thesis studies advanced signal processing techniques at UMTS user equipment receivers, which are particularly suited for the so-called 3.5G high speed downlink packet access services (HSDPA).

In the first part, we consider families of user dedicated downlink channel estimation methods, which are especially beneficial when there is dedicated channel transmit beamforming. The methods do not assume any a priori knowledge of the path delays and the beamforming parameters and they exploit all the transmitted pilot sequences as well as the structured dynamics of the channel. We start with building least squares (LS) estimates of the channels associated with dedicated and common pilots in each slot. Then we improve the dedicated channel estimate quality by either jointly Kalman filtering the two LS estimates or by a suboptimum cascade of weighted LS combining and one-state Kalman filtering. In the latter case, the order of Kalman filtering and weighted LS combining results in differing performance and complexity in different conditions. In order to obtain the estimates of the missing model parameters we incorporate the expectation maximization (EM) algorithm to the Kalman mechanism expanded with one-lag smoothing which also improves the performance compared to filtering alone.

In the second part we first consider a *chip level* HSDPA-specific decision-directed normalized least mean squares (HDD-NLMS) equalization scheme which uses the previous base station chip sequence estimates as the desired responses for equalizer adaptation. Then we generalize this concept by incorporating the despreading operation. Group despreading,

i.e. despreading with a partial code of PCPICH pilot sequence is a means to increase the adaptation rate and hence to increase the channel tracking capability w.r.t. the standard PCPICH symbol level LMS algorithm which does full PCPICH despreading. However, it has the risk of interference amplification which again can be compensated by incorporation of a decision directed mechanism, which estimates the ingredients of the pseudo-symbol output of the partial despreading operation using hard decisions and LMMSE weightings. Finally we propose *HSDPA symbol level* N-Griffiths and HDD-NLMS equalizers which enable adapting 16 times more frequently than the PCPICH-symbol rate adaptation. All the proposed HSDPA-specific algorithms have reasonable complexity and close to Max-SINR performance in realistic working regimes. We compare the performance of the HDD-NLMS vis-à-vis the N-Griffiths equalizer at both chip level and HSDPA symbol level. Since the latter requires channel parameters, in order to make a fair comparison, we perturb the correct channel parameters by some amounts compliant with the performances of various channel estimation methods considered in the first part.

In the last part we assess the benefits of using chip equalizers w.r.t. the usage of the conventional Rake receiver and using hard decision or hyperbolic tangent symbol nonlinearities w.r.t. the usage of linear feedback operations in the context of an iterative parallel interference cancellation receiver that we derive from the polynomial expansion of the symbol level covariance matrix inverse after the first stage equalization. Since the equalizers at different stages of the considered interference canceller are to be different as well, in order to estimate the essential equalizer parameters of a particular stage we use the estimates from the preceding stage.

Yves Grenier
Dissertation Committee Chair

Résumé

La présente thèse est consacrée à l'étude des techniques avancées de traitement du signal pour le Terminal Utilisateur dans le contexte du standard UMTS. Ces techniques concernent particulièrement l'évolution dite 3.5G du standard et connue sous le nom de HSDPA (High Speed Downlink Packet Access).

Dans la première partie, on considère une famille de méthodes d'estimation des canaux utilisateurs dédiés en liaison descendante. Ces méthodes sont particulièrement avantageuses dans un contexte de transmission à formation de faisceaux. Elles ne supposent aucune connaissance a priori des retards des signaux reus ainsi que les paramètres de formation des faisceaux; exploitent la totalité des séquences pilotes transmises ainsi que la dynamique structurée du canal. On commence par la construction d'une solution d'estimation du canal basée sur le moindre carré (LS) et associée aux pilotes dédiés et communs dans chaque slot. Ensuite, on améliore l'estimation du canal dédié soit grce à un filtrage conjoint des deux solutions moindre carré, ou bien par une concaténation sous optimale des combinaisons moindre carré pondérées et un filtre de Kalman à un seul état. Dans ce cas, l'ordre du filtre de Kalman ainsi que la combinaison moindre carré pondérée affectent les performances ainsi que la complexité en fonction des circonstances. Afin d'estimer les paramètres manquant au model on incorpore l'algorithme EM (Expectation - Maximization) à un processus de Kalman étendu avec un lissage de premier ordre ce qui va aussi améliorer les performances par rapport à la seule opération de filtrage.

Dans la seconde partie, on commence par considérer un schéma d'égalisation au niveau chip et dite HDD-NLMS (HSDPA specific Decision Directed Normalized Least Mean Squares). Ce schéma utilise l'estimation de la précédente séquence chip comme réponse cible pour l'adaptation de l'égaliseur. Dans la suite on étend ce concept en incorporant le processus de désétalement. Le désétalement groupe, ç-a-d le désétalement avec un code partiel de la séquence pilot PCIPCH, est un moyen d'accrotre la vitesse d'adaptation aux

conditions de propagation. Il augmente ainsi la capacité de poursuite du canal par rapport à l'algorithme standard qui utilise un désétalement total du PCPICH. Toutefois, cette solution risque d'amplifier les interférences qui peuvent, de nouveau, tre compensées par l'incorporation d'un mécanisme decision directed qui estime les ingrédients de la sortie pseudo symbole après le désétalement. L'estimation utilise une décision hard ainsi qu'une pondération LMMSE. Enfin on propose les égalisateurs niveau symbole HSDPA N-Griffith et HDD-NLMS qui permettent une vitesse d'adaptation 16 fois plus élevée que dans le cas de l'utilisation des symboles HSDPA. Les solutions spécifiques HSDPA ont une complexité raisonnable et offrent des performances proches aux Max-SINR dans un environnement réel. On compare les performances du HDD-NLMS avec celles de l'égalisateur à la fois au niveau chip et niveau symbole HSDPA. Etant donnée que la deuxième requière la connaissance des paramètres du canal, et dans le but d'effectuer une comparaison équitable, on introduit une perturbation compatible avec les performances des différentes méthodes d'estimation du canal considérées dans la première partie.

Dans la dernière partie, nous évaluons l'avantage des égalisateurs au niveau chip par rapport au récepteur Rake conventionnel ainsi que l'avantage d'une décision hard (ou les non-linéarités symbole en tangente hyperbolique) par rapport à l'utilisation d'une contre réaction linéaire dans le contexte d'un récepteur avec élimination itérative parallèle des interférences. Cette contre réaction sera déduite d'un développement polynomial de l'inverse de la matrice de covariance en amont du premier étage d'égalisation. Etant donné que les égalisateurs à chaque étage sont différents, nous utilisons les résultats d'analyse des étages précédents afin de pouvoir estimer les paramètres de l'égalisateur à un étage donné.

Thesis Contributions

The author is the premier contributor to the following publications and patents submitted during his Phd work.

Publications

- A. Baştuğ, A. Ancora and D.T.M. Slock,"Adaptive Equalization at HSDPA Symbol Level," *Proc. of the IEEE International Symposium on Personal, Indoor and Mobile Communications*, Helsinki-Finland, September 2006.

- A. Baştuğ and D.T.M. Slock,"Optimization of Combined Chip and Symbol Level Equalization for Downlink WCDMA Reception," *Proc. of the IEEE International Workshop on Signal Processing Advances in Wireless Communications*, Cannes-France, July 2006.

- A. Baştuğ, S. Sesia and D.T.M. Slock,"Adaptive Chip Level Equalization for HSDPA," *Proc. of the IEEE International Conference on Communications*, Istanbul-Turkey, June 2006.

- A. Baştuğ, G. Montalbano and D.T.M. Slock,"Common and Dedicated Pilot-Based Channel Estimates Combining and Kalman Filtering for WCDMA Terminals," *Proc. of the Asilomar Conference on Signals, Systems and Computers*, Pacific Grove-USA, November 2005.

- A. Baştuğ, G. Montalbano and D.T.M. Slock,"Generalized Pilot-Assisted Channel Estimation for WCDMA," *Proc. of the EUSIPCO Conference*, Antalya-Turkey, September 2005.

- A. Baştuğ and D.T.M. Slock,"A Macroanalysis of HSDPA Receiver Models," *Proc. of the IEEE Wireless Communications and Networking Conference*, New Orleans-USA, March 2005.

- A. Baştuğ and D.T.M. Slock,"Optimization Issues in Combined Chip and Symbol Level Equalization for Downlink WCDMA Receivers," *Proc. of the Asilomar Conference on Signals, Systems and Computers*, Pacific Grove-USA, November 2004.

- A. Baştuğ and D.T.M. Slock,"Downlink WCDMA Receivers Based on Combined Chip and Symbol Level Equalization," *European Transactions on Telecommunications*, Volume 16, Issue 1, January/February 2005.

- A. Baştuğ and D.T.M. Slock,"Downlink WCDMA Receivers Based on Combined Chip and Symbol Level Equalization," *Proc. of the European Wireless Conference*, Barcelona-Spain, February 2004. (expanded to the above journal paper with the same title)

- A. Baştuğ and D.T.M. Slock,"Interference Canceling Receivers With Global MMSE-ZF structure and Local MMSE Operations," *Proc. of the Asilomar Conference on Signals, Systems and Computers*, Pacific Grove-USA, November 2003.

- A. Baştuğ and D.T.M. Slock,"Structured Interference Cancellation at a WCDMA Terminal," *Proc. of the International Symposium on Signal Processing and Its Applications*, Paris-France, July 2003.

- A. Baştuğ and D.T.M. Slock,"Multistage Estimation of Time-Varying Wireless Channel Parameters," *Proc. of the IEEE Signal Processing and Communications Applications Conference*, Kuşadası-Turkey, April 2004. (in Turkish)

- A. Baştuğ and D.T.M. Slock,"Chip Rate and Symbol Rate MMSE Functionalities for CDMA Receivers," *Proc. of the IEEE Signal Processing and Communications Applications Conference*, Kuşadası-Turkey, April 2004. (in Turkish)

- A. Baştuğ and D.T.M. Slock,"Polynomial Expansion WCDMA Receivers Refined by Local MMSE Functionalities," *Proc. of the IEEE Signal Processing and Communications Applications Conference*, Kuşadası-Turkey, April 2004. (in Turkish)

Patents as Part of the Thesis

- A. Baştuğ and E. Alliot,"Noise Variance, SINR and QPSK Amplitude Estimation in CDMA Systems with OVSF Codes," (filed)
- A. Baştuğ and S. Sesia,"Method and Receiver Apparatus for Chip Level Equalization," (filed)
- A. Baştuğ and G. Montalbano,"Enhanced Channel Estimation for User-Dedicated Channels," (filed)
- A. Baştuğ,"Decision-Directed NLMS Equalizer by Group Despreading," (filed)
- A. Baştuğ and S. Sesia,"16-QAM Amplitude Estimation for HSDPA," (filed)
- A. Baştuğ and A. Ancora,"Griffiths Equalization for Multicode CDMA and HSDPA," (filed)

Patents as not Part of the Thesis

- A. Baştuğ and F. Tomatis,"Determination of Active Spreading Codes and Their Powers in CDMA Systems", (filed)
- A. Baştuğ and P. Demaj,"A Pilot-Aided Adaptation Method of Equalizer Coefficients, Memory, Equalizer and Receiver", (filed)
- A. Baştuğ and P. Demaj," A Pilot Symbol Level Adaptation Method, Memory, Equalizer and Receiver for Implementing the Method", (filed)
- A. Baştuğ and A. Ancora,"Filter Weight Estimation Device with Update at HSDSCH Symbol Rate for a Symbol Level Equalizer," (filed)
- A. Baştuğ and P. Demaj,"STTD Detection in CDMA Downlink," (pending)
- A. Ancora and A. Baştuğ,"Decision-Directed Super-Imposed Pilots Channel-Estimation For Downlink Reference Signaling in E-UTRA," (filed)
- A. Ancora, A. Baştuğ and F. Tomatis,"Hybrid Channel estimation For Super-Imposed Pilot 3GPP LTE 2nd Reference Signaling," (filed)

Contents

List of Figures

List of Tables

List of Abbreviations

i.e.	namely
w.r.t.	with respect to
s.t.	such that
3GPP	The 3rd Generation Partnership Project
ACU	Address Computation Unit
ALU	Arithmetic Logic Unit
AR(n)	Autoregressive Process of Order n
AWGN	Additive White Gaussian Noise
BS	Base Station
CDF	Cumulative Distribution Function
CDMA	Code Division Multiple Access
CGU	Code Generation Unit
CMF	Channel Matched Filter
CQI	Channel Quality Indicator
CRC	Cyclic Redundancy Check
DCH	Dedicated Channel
DPCH	Dedicated Physical Channel
DPCCH	Dedicated Physical Control Channel
DPDCH	Dedicated Physical Data Channel
EDGE	Enhanced Data rates for GSM Evolution
EM	Expectation Maximization
EVP	Embedded Vector Processor
FDD	Frequency Division Duplexing
FWHT	Fast Walsh Hadamard Transform
FIR	Finite Impulse Response

GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HARQ	Hybrid Automatic Repeat Request
HSDPA	High Speed Downlink Packet Access
HSDCCH	High Speed Dedicated Control Channel
HSDSCH	High Speed Downlink Shared Channel
HSPDSCH	High Speed Physcial Downlink Shared Channel
HSSCCH	High Speed Shared Control Channel
IC	Interference Canceller
ICI	Interchip Interference
ISI	Intersymbol Interference
IVU	Intra Vector Unit
LMMSE	Linear Minimum Mean Square Error
LMS	Least Mean Squares
LOS	Line of Sight
LS	Least Squares
LSU	Load and Store Unit
MAC	Multiply and Accumulate Unit
MIMO	Multiple Input Multiple Output
MIPS	Mega Instructions Per Second
ML	Maximum Likelihood
MMSE	Minimum Mean Square Error
MMSE-ZF	Minimum Mean Square Error Zero Forcing
MSE	Mean Square Error
MUD	Multiuser Detector
MUI	Multiuser Interference
NLMS	Normalized Least Mean Squares
Node-B	Base Station Terminal in UMTS
NMSE	Normalized Mean Square Error
NP-hard	Non-deterministic Polynomial-time Hard
OCNS	Orthogonal Channel Noise Simulator
OF	Orthogonality Factor

OVSF	Orthogonal Variable Spreading Factor
pdf	Probability Density Function
PCCPCH	Primary Common Control Physical Channel
PCPICH	Primary Common Pilot Channel
PE	Polynomial Expansion
PIC	Parallel Interference Cancellation
PSCH	Primary Synchronization Channel
RNC	Radio Network Controller
rrc	root raised cosine
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
QAM	Quadrature Amplitude Modulation
SER	Symbol Error Rate
SCPICH	Secondary Common Pilot Channel
SF	Spreading Factor
SHU	Shuffle Unit
SIMD	Single Instruction Multiple Data
SIMO	Single Input Multiple Output
SINR	Signal-to-interference-plus-noise Ratio
SIR	Signal-to-interference Ratio
SIC	Serial Interference Cancellation
SISO	Single Input Single Output
SNR	Signal-to-Noise Ratio
SSCH	Secondary Synchronization Channel
TDD	Time Division Duplexing
TDMA	Time Division Multiple Access
TTI	Transmission Time Interval
UE	User Equipment
U-FWHT	Unitary Fast Walsh Hadamard Transformation
ULMMSE	Unbiased LMMSE
UMTS	Universal Mobile Telecommunications Systems
VLIW	Very Long Instruction Word

WCDMA	Wideband CDMA
WHT	Walsh Hadamard Transformation
WSS-US	Wide Sense Stationary Uncorrelated Scattering
ZF	Zero Forcing

Notation and Commonly Used Symbols

bold symbols	Vectors or matrices
$(\cdot)^T$	Transpose operator
$(\cdot)^H$	Hermitian operator
$\otimes$	Kronecker product
$\|\| \cdot \|\|$	Frobenius Norm
$\| \cdot \|$	Absolute value operator
$\dagger$	Paraconjugate operator (matched filtering in time domain)
$*$	Convolution operator
$\mathbb{C}$	set of complex numbers
$\Re(.)$	Real part
$\Im(.)$	Imaginary part
$\rho(.)$	Spectral radius operation
$\lceil . \rceil$	Ceiling function to the closest higher integer value
$\lfloor . \rfloor$	Flooring function to the closest lower integer value
$[n]$	Index for symbol sequences
$[l]$	Index for chip sequences
$argmin(.)$	Argument minimizing the in-paranthesis expression
$argmax(.)$	Argument maximizing the in-paranthesis expression
E_b/N_o	Energy per bit to noise power spectral density ratio
E_c/N_o	Energy per chip to noise power spectral density ratio
$\boldsymbol{I}_{K\times K}$	$K \times K$ identity matrix
I_{oc}	Total received power from the surrounding cells
I_{or}	Total transmitted BS power
$\hat{I}_{or}$	Total received BS power

$min(.)$	Minimum
N	FIR channel length in chips
m	Samples per chip
$mod(x, y)$	Modulus of x in basis y
M	FIR channel length in symbols
$N_x(y, z)$	Normal distribution over x with mean and variance (y, z)
$q^{-1}(.)$	Delay operator
$Q(.)$	Gaussian tail function
T	Symbol period
T_c	Chip period
$\boldsymbol{X}(z)$	z-transform of $\boldsymbol{x}[n]$
$\boldsymbol{X}(z)^\dagger$	Paraconjugate of $\boldsymbol{X}(z) = \boldsymbol{X}^*(z^{-1})$

Chapter 1

Introduction

First generation analog mobile systems from 1960s were so inefficient that they were only privileges to provide speech connection to a limited number of selected people like city mayors and police forces. Second generation (2G) digital cellular systems such as GSM have been big commercial successes extending the speech service to the *grand public*. Today the number of worldwide GSM registered users is well over 1 billion and is increasing at a steady pace. The speech data rate support of GSM is around 9kbps per user. The 2.5G data network extensions over GSM known as GPRS and EDGE increased the typical user data rate support to 40kbps and 120kbps respectively. Initial deployment plans for the 3G UMTS systems which are based on the WCDMA air access technology promise an average data rate of 384kbps. Although seems to be an attractive rate at first sight, 3G UMTS system deeply feels a threat from alternative systems like WiMax which uses the OFDM air access technology which in general is believed to be superior to WCDMA in terms of spectral efficiency. This situation forced the 3GPP community to rapidly standardize the 3.5G HSDPA technology which aims to provide downlink throughput rates up to 14.4 Mbps with a nominal rate of 2Mbs. Higher spectrum efficiency requirements of the HSDPA services oblige the CDMA mobile terminals to substitute their conventional Rake receivers with more advanced ones like *chip equalizers* at first stage and *multiuser detectors* which are also known as *interference cancellers* in the long term. Most of these advanced receiver modules necessitate at the same time advanced *channel estimators*. This thesis deals with all these three aspects.

1.1 Organization of the Thesis

In this chapter we give a brief description of the UMTS downlink system, HSDPA features, downlink channel properties and the state of the art mobile receivers. The coverage is only sufficient to an extent to support the main contributions of the thesis. For details the reader is directed to the relevant literature.

Chapter 2 is a brute force attempt to give an answer to the question of whether there is a need to consider advanced receivers for HSDPA. Instead of considering a selected subset of parameter values that affect the receiver performance metrics, we model these parameters in a sophisticated way at several point locations which are uniformly and densely distributed to the cell surface. In this way, we obtain a cellular CDF of the maximum reachable throughput computed from Shannon's capacity equation for Gaussian channels.

In Chapter 3 we propose a group of channel estimation methods which exploit the channel dynamics and all the pilot training sequences provided by the system in a close-to-optimal manner.

In Chapter 4 first we propose a decision directed chip level NLMS equalizer which is applicable when the mobile receives HSDPA service. Then we explain some novel methods that estimate the symbol error variances and the symbol amplitudes which are necessary not only for the adaptation of the equalizer but also for other system functionalities. Finally we compare the performance of the proposed equalizer with the performance of another chip level equalizer derived from the Griffiths algorithm. Since the latter requires channel parameters, in order to make a fair comparison, we perturb the correct channel parameters by amounts compliant with the performances of different channel estimation methods considered in Chapter 3.

In Chapter 5 we generalize the decision directed equalization concept by incorporating the despreading operation in order to benefit from the spreading gain which is an inherent property of CDMA systems. This receiver can also be considered as a backup solution to the chip level decision directed equalizer when the mobile does not receive HSDPA service but is obliged to continue adapting the filter weights in order not to lose the capability of tracking time-varying channels.

In Chapter 6 we propose HSDPA symbol level adaptation schemes via Griffiths and decision directed mechanisms which have the advantage of including a subset of HSDPA code domain in order to make a compromise between complexity and performance.

In Chapter 7 we assess the benefits of using chip level equalizers instead of the Rake receivers and hard decision or hyperbolic tangent symbol nonlinearities instead of the linear feedback operations in the context of iterative parallel interference cancellation receivers which are modeled from the polynomial expansion of the symbol level covariance matrix inverse after the first stage equalization. Since the equalizers at different stages of the covered interference cancellers are to be different as well, in order to estimate the essential equalizer parameters of a particular stage we use the the analysis results from the preceding stages.

Chapter 8 is a wrap up, first comparing the complexities of the chip level and symbol level equalizers and then a listing of the future research possibilities.

Figure 1.1 shows the connection between the Chapters of the thesis.

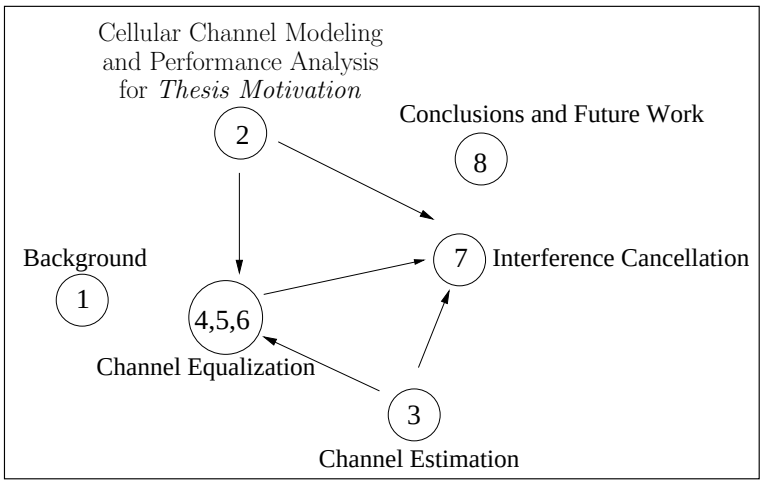


Figure 1.1: Thesis dependency chart

1.2 Multi-access in the UMTS FDD Downlink

This thesis is concerned with the downlink communications of the Frequency Division Duplexed (FDD) mode of UMTS. Transmissions are done in frequency bands of 5Mhz around 2.1 GHz. Each UMTS operator has in general 2 or 3 bands for the downlink transmission. The base station which is known as Node B in the UMTS radio access network (RAN) context is the source of transmissions for its *logical cell*. If a sectorized cell planning is deployed, then Node B is responsible for more than one logical cell as shown in Figure 1.2.

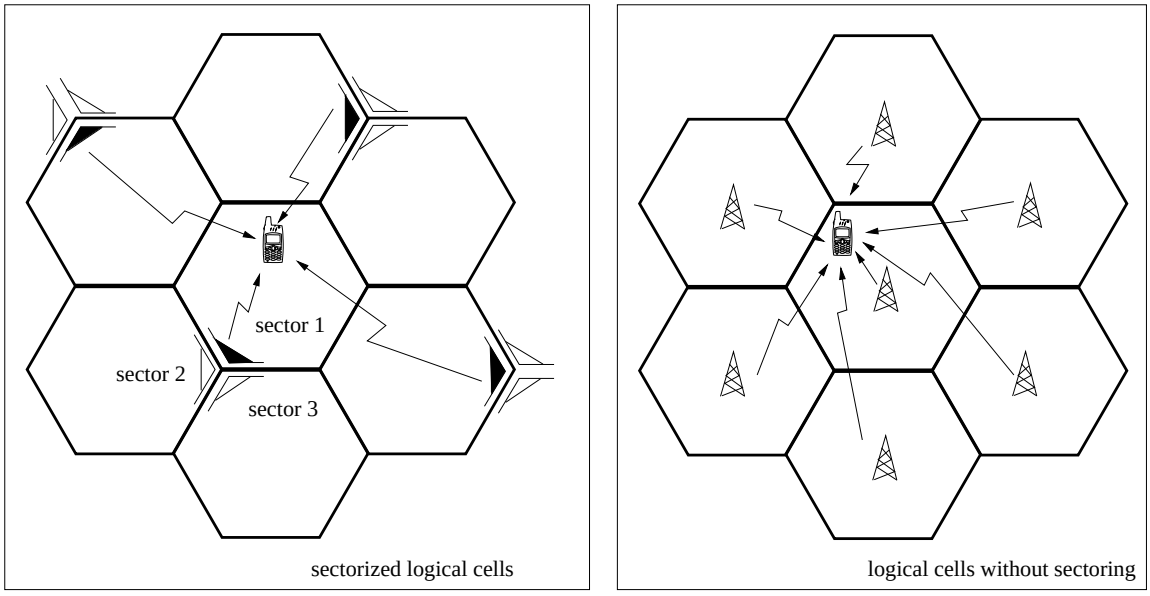


Figure 1.2: Logical cells differentiated by scrambling codes

The signals transmitted from different logical cells are differentiated from each other by the assignment of different *pseudo-random scrambling codes* which are repeated every UMTS *frame* of 38400 chips, hence are known as *long* overlay codes.

Multi-access of the users in the same logical cell is realized by a CDMA scheme which uses *short orthogonal channelization codes* from various levels of the OVSF code tree shown in Figure 1.3 each level of which contains codes corresponding to the columns of the Walsh-Hadamard transformation (WHT) with relevant size. A channelization code assigned to a user is periodically used for the transmission of each symbol. Any particular code in the i-th position at the SF level t is related to its two closest child codes at SF level $t+1$ with the transformation

$$\begin{bmatrix} \boldsymbol{c}_{2^{t+1},2^i} & \boldsymbol{c}_{2^{t+1},2^i+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \boldsymbol{c}_{2^t,i} \tag{1.1}$$

The valid code lengths are from the set $\{2^t,\ t \in 2,3,\ldots,9\}$. The largest code length, i.e. the spreading factor 512, is very rarely used. When a particular code is assigned to a user, then all its parent or child codes are blocked for usage in order to preserve orthogonality among used codes. These properties make UMTS FDD downlink a *code-limited* system. To illustrate, in case only a single spreading level is used from the OVSF tree, then the number of available codes for that particular scenario is the associated spreading factor.

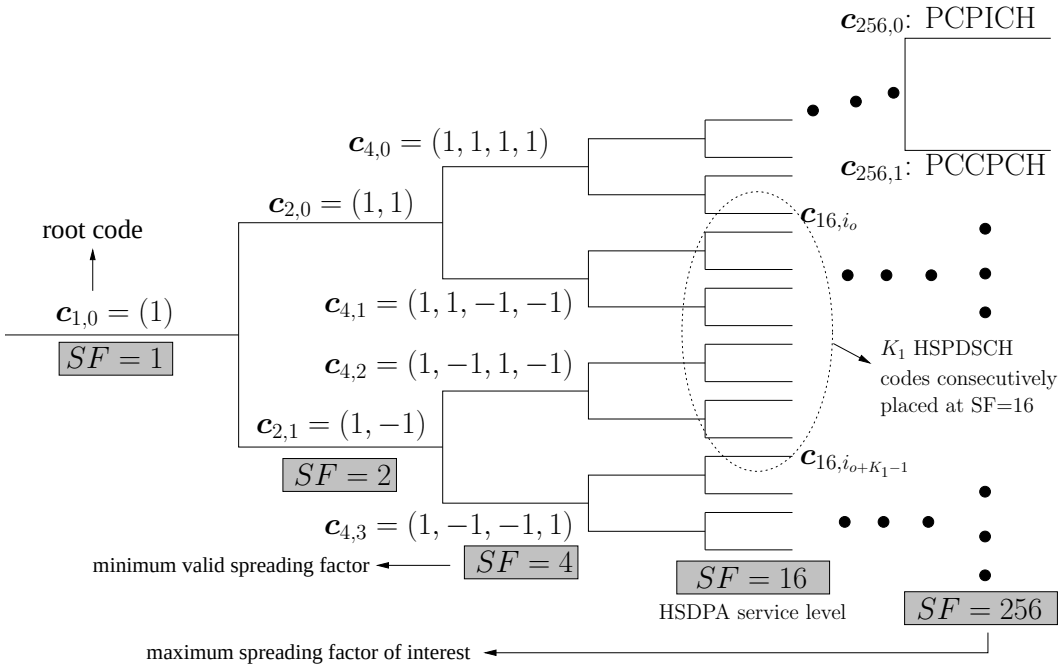


Figure 1.3: Partial schematic of the OVSF code tree

1.3 UMTS Services

The flexibility of using different length codes makes UMTS a *multi-rate* system, enabling services with different QoS.

Just like any other cellular multi-access system, UMTS has also some requirements to satisfy the most important of which are providing enough capacity, coverage and a variety of services each with different QoS and rate requirements. The first two requirements are conflicting in the sense that increasing the coverage area of cells, i.e. decreasing the number of deployed base stations, decreases the total system capacity. From this relation it is easy to judge that any advanced receiver or transmission diversity technique that improves the spectral efficiency of the system can instead or, by a compromise, at the same time be exploited to increase the coverage.

At the launch of a new WCDMA network, operators' first priority is covering a large area, providing mostly speech and low-rate data services. However, once full coverage is achieved, capacity becomes an immediate concern triggered mostly by high rate data-service requirements. To satisfy these requirements, UMTS standard defines four QoS classes with differing delay and ordering needs [7]:

Conversational: low delay, strict ordering, e.g: voice

Streaming: modest delay, strict ordering, e.g: video

Interactive: modest delay, modest ordering, e.g: web browsing

Background: no delay guarantee, no ordering, e.g: bulk data transfer

1.4 HSDPA Features

Background and *Interactive* UMTS service classes have a burst nature. This characteristics triggered the idea of time sharing of some of the system resources among the users, most importantly the orthogonal codes in the downlink, along with applying the following supporting techniques, extensions, changes, removals on the shared codes, leading to the standardization of HSDPA in the UMTS Standard Release-5 [8].

Allocation of multiple access codes for HSDPA service: Motivated by the burst nature of the data, as shown in Figure 1.3, $K_1 \in \{1, 2, \ldots, 15\}$ of the 16 channelization code resources at SF=16 are allocated as High Speed Physical Downlink Shared Channels (HSPDSCHs) and dynamically time multiplexed among demanding users in order to achieve a higher spectral efficiency and a larger link adaptation dynamic range. The variable $i_o \in \{1, 2, \ldots, 15\}$ denotes the position of the first HSPDSCH code. The single *transport* channel counterpart spanning the HSPDSCHs of a user is called HSDSCH.

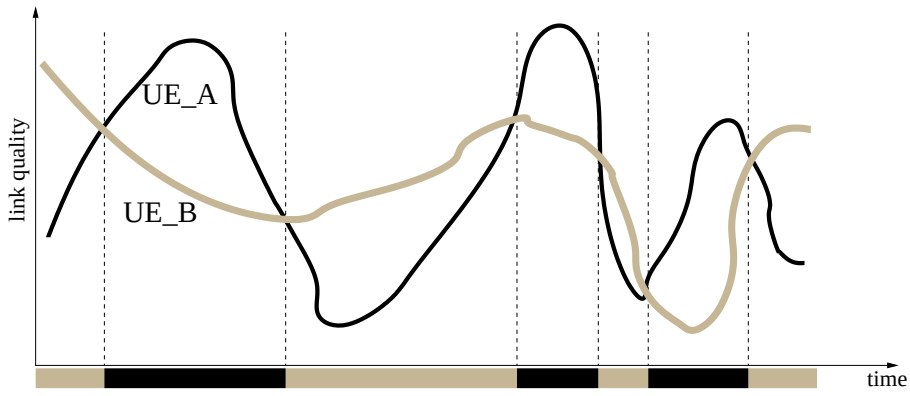


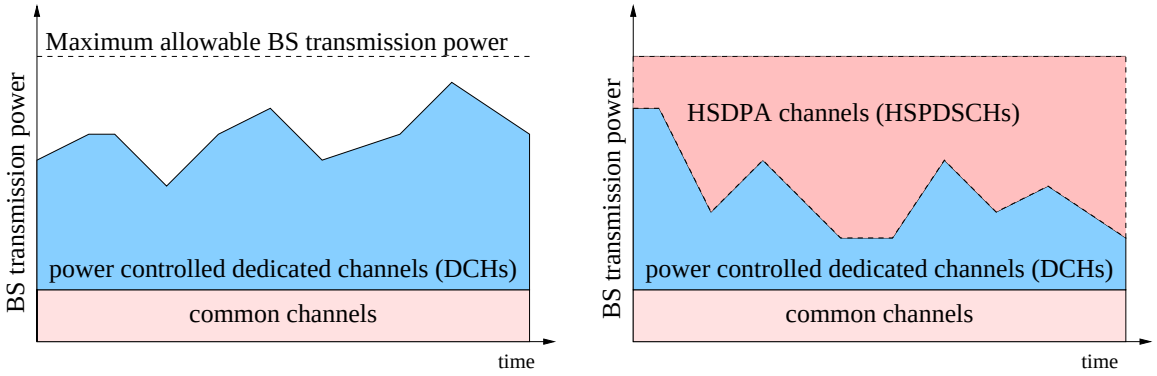
Figure 1.4: Principle of multiuser diversity

Fast scheduling of allocated codes: The goal is to exploit *multiuser diversity*, i.e. the temporal channel quality variance among the users, in order to increase the *sum capacity*, i.e. the total delivered payload by the BS. By one extreme approach, as demonstrated in Figure 1.4 in a simple 2-user system context, one can preferably assign all the codes to a single user with the instantaneously best channel conditions, maximizing the throughput. At the other extreme, users might be served in a *fair* round-robin fashion. In this respect operators are free to choose any set of schedulers compromising throughput and fairness by basing their decisions on the predicted channel quality, the cell load and the traffic priority class. In order to reduce the delay in signaling and to better track the channel variations, scheduling is performed at Node-B which is closer to the air interface compared to the Radio Network Controller (RNC) which was responsible with such tasks in the earlier standard releases. Moreover scheduling period is decreased to 2ms sub-frame duration, i.e. 1 TTI, from the 10ms frame duration[1]. Soft handover is also replaced by fast *best-signaling-cell* selection which can be considered as a kind of *spatial* scheduling complementing the *temporal* scheduling.

Link adaptation: As schematically demonstrated in Figure 1.5, perhaps one of the most important changes which influences almost all the new applied HSDPA techniques is that practically there is *no fast power control* on HSPDSCHs and all the instantaneously remaining *allowed* BS power is assigned to HSPDSCHs which creates a high amount of *temporal power variance* on the HSPDSCH codes [9]. In this case, as can be interpreted from the figure, the system is also capable of utilizing the available BS power more efficiently than the power controlled case. Furthermore, as shown in Figure 1.10, different user distances from the BS and different user mobility levels create a high amount of *inter-user link quality variance*. These two properties make an impact on the variance of the user link quality at scheduled instants and Node-B has a very high degree of freedom in deciding for the number of allocated HSPDSCH codes, coding rate, puncturing rate and the modulation scheme, 16 QAM a possibility besides QPSK at LOS or high received power conditions, to maximize the throughput of the instantaneously scheduled user. For this purpose Node-B might use either the explicit CQI measurement reports from the UE based on the SINR of PCPICH or

[1]Low-end and medium-end HSDPA UEs which do not have enough buffering capability are obliged to wait at least two or three TTI periods respectively between two consecutive TTI data scheduling [8].

the known transmit power of the power-controlled downlink DPCH associated with the HSPDSCHs. Another new aspect related to link adaptation and which reduces the overhead is that only one CRC block is added for one transmission time interval (TTI) of 2ms. This is motivated by the assumption that there is no benefit in knowing which of the several parallel transport blocks are erroneous for when one transport block is erroneous, most probably the others are as well.



UMTS FDD downlink *R99* power control mechanism

UMTS FDD downlink *R5* power control mechanism

Figure 1.5: Power control with and without HSDPA

Hybrid automatic repeat request (HARQ): When transmission entities are identified to be erroneous by a standard protocol such as *selective-repeat* or *stop-and-wait*, fast retransmit request is done from Node-B and combinations of soft information from the original transmission and previous retransmissions are utilized to increase the probability of correct reception [10, 11]. These operations fine-tune the effective code rate, in a way compensating for errors in the channel quality estimates used for link adaptation. The two well known such methods are *chase combining* where weighting of identical retransmissions is done and the *incremental redundancy* where additional parity bits are sent each time.

To support the listed functionalities, two new channel types are introduced. In the downlink, one or more shared control channels (HSSCCHs) broadcast the scheduled UE identity, the transport format and the HARQ process identifier. The UE monitors up to 4 different HSSCCHs and tries to find out if it is going to be scheduled or not. In the uplink, the High Speed Dedicated Physical Control Channel (HSDPCCH) carries the status reports for HARQ and the Channel Quality Indicators (CQIs). Figure 1.6 briefly demonstrates the

order of events in the HSDPA transmission protocol. More detailed timing information and the slot structures are given in Figure 1.7 together with other UMTS channels relevant to the topics of the thesis.

HSSCCH channel is frame aligned with the PCPICH channel which is in general used as a reference by several other UMTS channels and synchronization procedures as well.

HSPDSCHs are offset by 2 time slots w.r.t. HSSCCH which gives the UE some time to decode the time critical control and supervision information carried by the first slot of HSSCCH before receiving the HSDSCH payload data. Learning the scheduling of UE two slots beforehand is at the same time very useful for the *adaptive* equalizers that we will discuss in the following chapters. In order to do power savings, it is in general preferable to freeze the adaptation mechanism of an equalizer when the UE does not receive any HSDPA data. On the other hand it is beneficial to start the adaptation process some time earlier than the start of the useful data to force the equalizer to converge earlier.

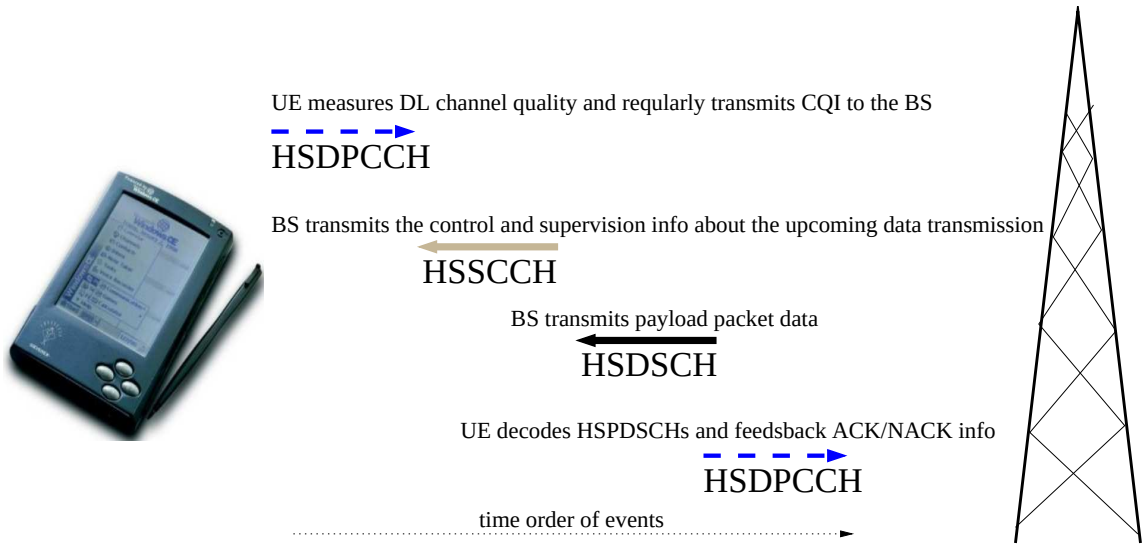


Figure 1.6: HSDPA transmission protocol

For a more detailed coverage of HSDPA, see [12, 13, 14, 15].

1.5 Downlink Transmission Model

The baseband downlink transmission model of the UMTS-FDD mode system with HSDPA support is given in Figure 1.8.

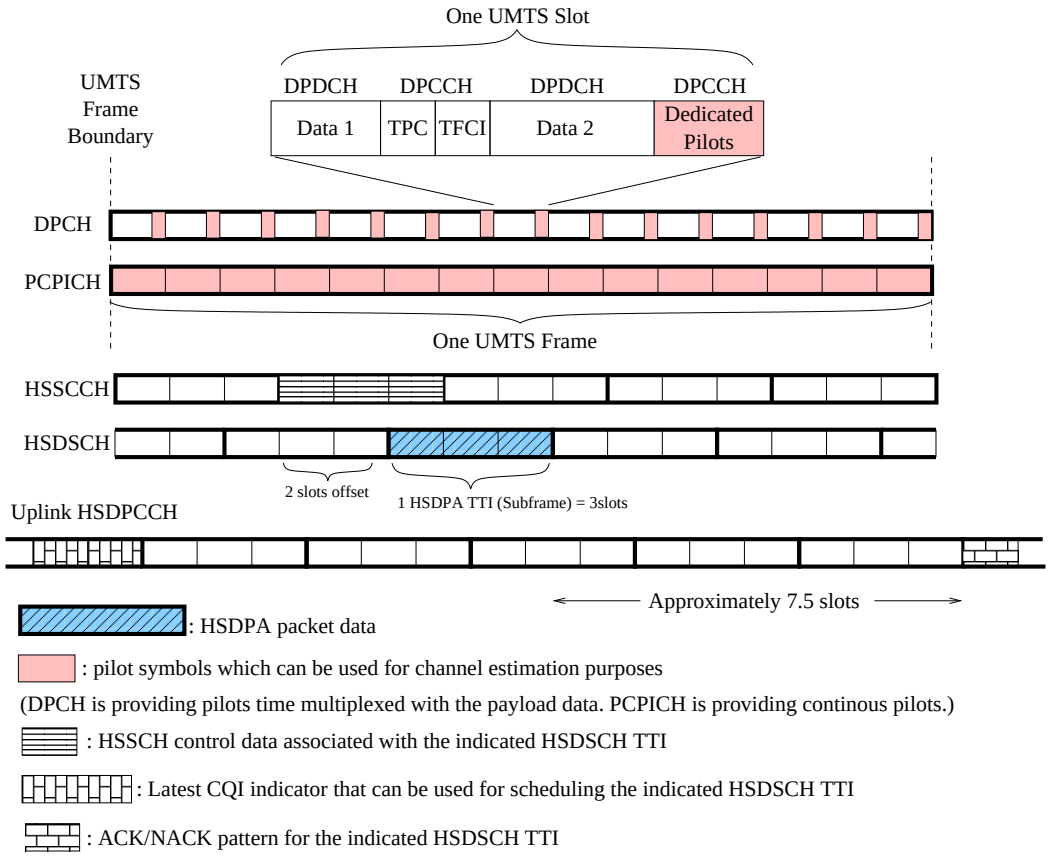


Figure 1.7: Slot structures and timings of UMTS channels of interest

At the transmitter, the first group of K_1 i.i.d QPSK or 16-QAM modulated symbol sequences $\{a_1[n], a_2[n], \ldots, a_{K_1}[n]\}$ which belong to the HSDPA transmission are first upsampled by a factor of 16 and then convolved with their respective unit-amplitude channelization codes $\{\boldsymbol{c}_{16,i_o}, \boldsymbol{c}_{16,i_o+1}, \ldots, \boldsymbol{c}_{16,i_o+K_1-1}\}$ shown in Figure 1.3. An important property that we will often exploit is that all the HSPDSCH symbols have the same power and the same modulation scheme.

The second group of multi-rate transmissions $\{\tilde{a}_1[n_1], \tilde{a}_2[n_2], \ldots, \tilde{a}_{K_2}[n_{K_2}]\}$[2] representing the dedicated physical channels (DPCHs), HSSCCHs and other control channels are similarly upsampled and convolved with their respective channelization codes $\left\{\boldsymbol{c}_{L_1,i_1}, \boldsymbol{c}_{L_2,i_2}, \ldots, \boldsymbol{c}_{L_{K_2},i_{K_2}}\right\}$.

The third group of chip sequences associated with PCCPCH, PCPICH, PSCH and

[2]different symbol indices such as $n, n_1, n_2, \ldots, n_{K_2}, n_p$ are used in the text and on Figure 1.8 to stress the *multi-rate* property of the transmission scheme

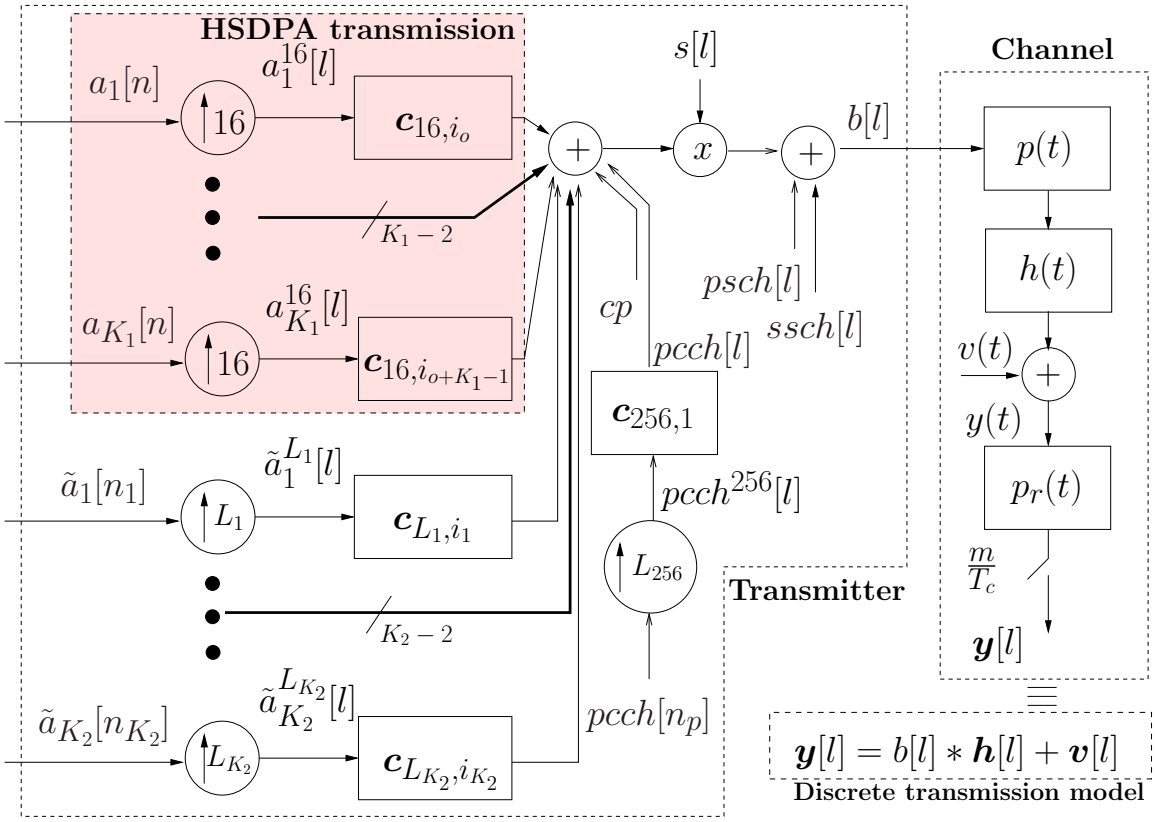


Figure 1.8: Baseband UMTS downlink transmission model

SSCH channels which are subjects to discussions in the following chapters are explicitly demonstrated as $pcch[l]$, cp, $psch[l]$ and $ssch[l]$ respectively. It is particularly important for further discussion to state that cp is a positively scaled form of the unit vector $\frac{1+j}{\sqrt{2}}$ which has 45 degrees phase.

The sum sequence of all the generated chip sequences is multiplied with the unit-energy BS-specific aperiodic scrambling sequence $s[l]$. PSCH and SSCH are the exceptions, multiplexed after the scrambler, since as a first-step task in the receiver they are utilized for determining, i.e. searching, which scrambling sequence is assigned to the BS. The resultant effective BS chip sequence $b[l]$ is transmitted to the channel.

1.6 Downlink Channel Model

UMTS downlink channel has three cascade components in the order of a *root-raised-cosine* (rrc) pulse shape $p(t)$ with a roll-off factor of 0.22 shown in Figure 1.9, the *time-varying multipath propagation* channel $h(t)$ and a receiver front-end filter $p_r(t)$ which is in general chosen to be again an rrc pulse shape with a roll-off factor of 0.22 due to the fact that the *raised cosine* (rc) result of the rrc-rrc cascade is a *Nyquist pulse* whose T_c-spaced disrete time counterpart is a single unit pulse at time instant 0. In this case the only ICI source[3] is $h(t)$. Alternatively a low pass antialiasing filter with a cutoff frequency between $\frac{1.22}{T_c}$ and $\frac{2}{T_c}$ might be considered as $p_r(t)$ in the case of twice chip rate sampling. The latter case is a reasonable choice for fractionally spaced equalizers [16, 17]. The *effective* continuous time channel is hence given as

$$h_{eff}(t) = p(t) * h(t) * p_r(t) \tag{1.2}$$

When there is no beamforming, the propagation channel and the effective overall channel are unique for all the transmitted data from the same BS.

1.6.1 Time-varying Multipath Propagation Channel

Modeling of the propagation channel $h(t)$ is a very subtle and sophisticated field in itself [18, 19]. Therefore we restrict the discussion in this section to intuitive explanations of some aspects which are essential for the main chapters of the thesis.

The most observable effect of the propagation channel on the received signal quality is the time varying signal amplitude attenuation which is more often known as *fading* and is a combined consequence of different scale effects in space which however manifests itself in again different scales in time.

The environment-dependent *large-scale* statistics of the UE received power at a distance d in kilometers is modeled as[4]

$$P_r(\text{dBm}) = P_t(\text{dBm}) - G(\text{dB}) - 10nlog_{10}d(\text{dB}) + 10log_{10}x(dB) \tag{1.3}$$

where P_t is the transmitted power[5], G is the amount of path loss at a reference distance of 1km, n is the path loss exponent and x is the log-normal shadowing term with *geomet-*

[3] there is no notion of ISI in UMTS downlink since scrambling distorts the symbol level cyclostationarity
[4] dBm is a relative measure w.r.t. 1mW power level
[5] In UMTS terminology $I_{or} = P_t$ and $\hat{I}_{or} = P_r$

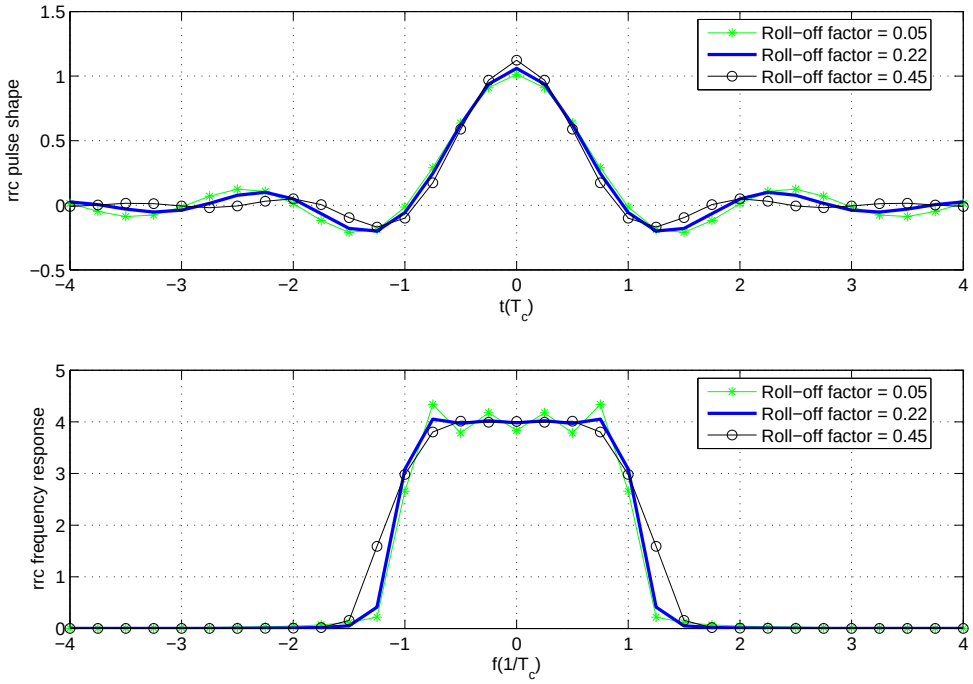


Figure 1.9: Root-raised cosine pulse shapes with different roll-off factors. Higher factors induce less ICI since the tails decay faster but they consume more bandwidth.

ric mean 0 and *geometric* standard deviation σ_x. Shadowing is a consequence of signal absorption by the obstacles in the terrain between the BS and UE such as hills, trees, buildings, cars and it causes a *variance* around the distance-dependent *mean* path loss. It is a *slow-fading* parameter which only varies when the UE changes its place by a distance proportional to the length of an obstructing object.

The most important propagation channel characteristics is the *multi-path* effect. Infinitely many replicas of the transmitted signal which are reflected from several objects reach the UE with different delays and different complex attenuation factors. Specular replicas are clustered together to generate the effective multi-paths shown in Figure 1.10. The *sparse channel model* which takes into account only the most dominant P paths can be formulated as

$$h(t) = \sum_{i=1}^{P} h_i(t)\delta(t - \tau_i(t)) \tag{1.4}$$

The difference between the largest and smallest delay elements $\Delta_\tau = \tau_P - \tau_1$ is the *delay*

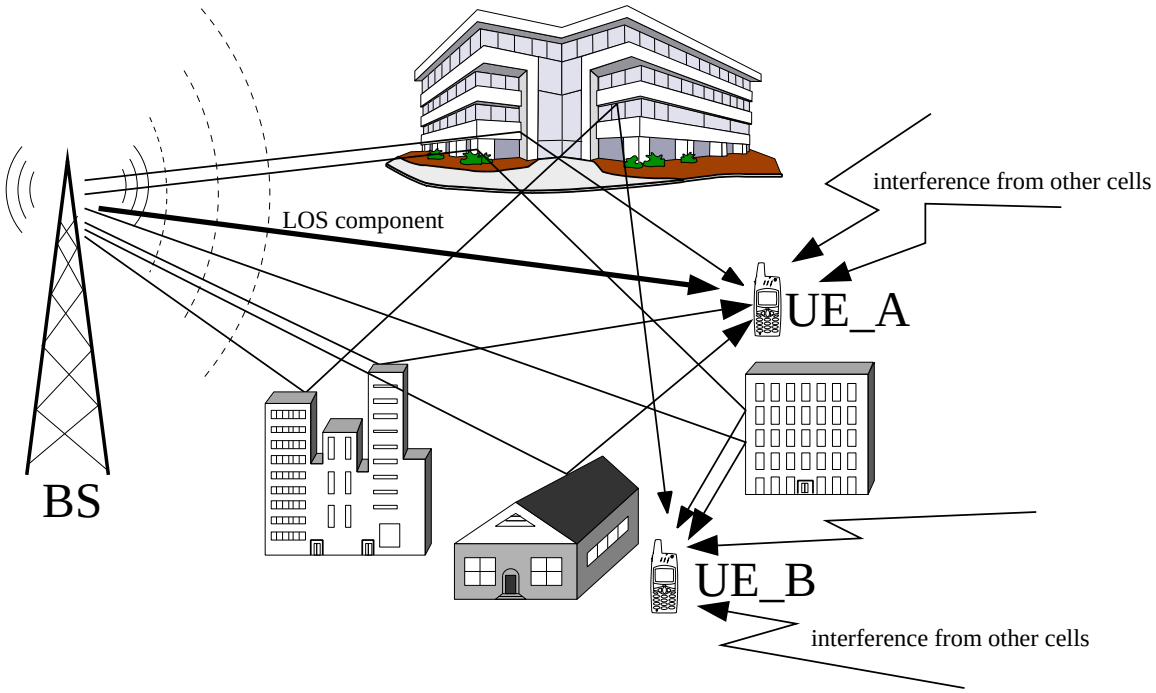


Figure 1.10: Multipath effect

spread of the channel. If $\Delta_\tau \geq T_c$, then the channel is *frequency selective*. This notion comes from the inverse of the delay spread known as *coherence bandwidth* $B_o \approx \frac{1}{\Delta_\tau}$. The physical meaning of B_o is that when two different sinusoidal components with frequencies f_1 and f_2 are transmitted they are impacted differently by the channel if $\Delta f = |f_1 - f_2| \geq B_o$. In other words if the signal BW is larger than B_o, which is very often the case for UMTS *wideband* CDMA downlink, signal spectrum is non-uniformly affected by the channel. On the one hand, if no channel equalization is applied this is a very dispersive situation driving the communication unreliable. On the other hand it is an opportunity to exploit the inherent *frequency diversity* coming from different sub-bands of the spectrum which are considered to be *independently fading*. In the time domain this property manifests itself in a different shape as the *resolution of the paths* which are separated from each other by at least a distance of T_c. Conventional Rake receiver exploits this fact by collecting energy via multiple correlations at time instants corresponding to the path delays. Although exact resolution is lower bounded by the chip period, some more diversity is expected from decreasing this constraint to slightly lower values such as $\frac{3T_c}{4}$ [20]. If an opposite situation occurs, i.e. if the signal BW is smaller than B_o, then the channel is *flat fading* meaning that there is no

ICI. At first sight this seems to be a good situation not requiring a complicated equalization procedure. However when there are deep and long lasting channel fades as is the case in slow fading channels, other means such as transmit diversity or receive diversity are necessary to recover the UE from the *outage* state. These techniques, on the other hand, complicate the BS and the UE.

When all the specular components that generate one dominant path are modeled as i.i.d complex random variables, by central limit theorem, channel parameters turn out to be circularly symmetric Gaussian random variables with zero mean and $2\sigma_i^2$ variance. Consequently, their complex envelope amplitudes are Rayleigh distributed.

$$p(|h_i(t)|) = \frac{|h_i(t)|}{\sigma_i^2} e^{-\frac{|h_i(t)|^2}{2\sigma_i^2}} \text{ for } |h_i(t)| \geq 0, \;\; 0 \text{ otherwise} \tag{1.5}$$

When there is one very dominant line of sight (LOS) path as is the case for UE_A in Figure 1.10, its distribution is *Rician* which is more desirable since in that case there are less frequent and less deep fades. In this thesis we are not considering LOS situations.

Sparse multipath channel parameters are modeled to be wide sense stationary and uncorrelated with each other (WSS-US model) [18]. Therefore each one of them experiences an independent *small-scale* fading due to the movement of the UE, the movements of the objects which have impacts on that particular path and even the microscopic changes in the air particles. Previously mentioned shadowing is a *large-scale* complement manifesting itself as birth or death of a path.

The *time-variance* of sparse channel parameters is a metric associated with the amount of signal spectral broadening caused by a Doppler shift which in return is proportional to the *effective* UE velocity in the direction of the coming path ray. The dual relation of a broadening in the frequency domain transfer function is a narrowing of the non-zero channel autocorrelation window in the time domain from infinity to a finite quantity known as channel *coherence time* T_o. The physical meaning of T_o is that when a sinusoid is transmitted twice at times t_1 and t_2, the two are influenced differently by the channel if $\Delta t = |t_1 - t_2| \geq T_o$. The channel is very often considered as *fast-fading* when $T_o < T_c$ since in this case different parts of a chip are influenced by different-valued channel parameters. With this reasoning, CDMA channels always fall into the contrary *slow fading* category. A better criterion to judge whether a channel is fast or slow fading is to compare T_o with the delay requirements of the considered application or receiver block. If we consider a UE chip equalizer, for example, which recomputes its weights periodically from scratch by using the

channel parameter estimates, then coherence time should be more than the chosen update period[6].

One might think that slow fading is always a desired situation, however as explained before, it causes the deep fades to last very long. In some catastrophic cases none of the diversity measures might help. Recently some work is going on to remedy such situations by artificially generating fast fading channel conditions via a transmission scheme called *opportunistic beamforming* [21].

Figure 1.11 gives a brief summary of the common channel impairments and the principal techniques for mitigating them.

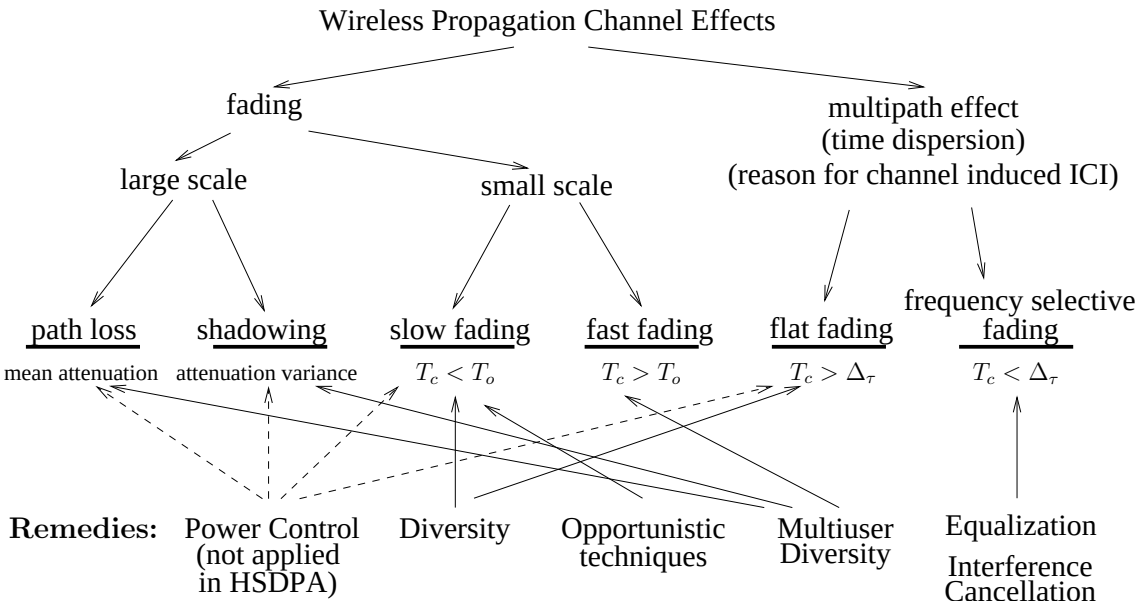


Figure 1.11: Summary of channel impacts and *most relevant* procedures against them.

1.6.2 Sources of Multichannels

The discrete time counterpart of $h_{eff}(t)$ after the sampling operation becomes an FIR *multi-channel* $\boldsymbol{h}[l]$ or equivalently a *poly-phase* channel $\boldsymbol{h}_p[l]$ in the presence of multiple antennas and/or integer factor oversampling w.r.t. the chip rate as is shown in Figure 1.12. In such cases the received *vector stationary*[7] signal can be modeled as the output of a $m \times 1$

[6] A typical requirement for the computation period of nonadaptive HSDPA equalizer weights is 512 chips

[7] meaning each phase is stationary, see Appendix A for a more detailed explanation.

single input multi output (SIMO) system[8] with a past memory of $N-1$ input elements and with the I-O relations

$$\boldsymbol{y}[l] = \boldsymbol{h}[l] * b[l] + \boldsymbol{v}[l] \tag{1.6}$$

$$\boldsymbol{y}[l] = \begin{bmatrix} y_1[l] \\ \vdots \\ y_m[l] \end{bmatrix}, \boldsymbol{h}[l] = \begin{bmatrix} h_1[l] \\ \vdots \\ h_m[l] \end{bmatrix}, \boldsymbol{v}[l] = \begin{bmatrix} v_1[l] \\ \vdots \\ v_m[l] \end{bmatrix} \tag{1.7}$$

where $\boldsymbol{v}[l]$ denotes the additive noise which represents the sum of the thermal noise and the intercell interference filtered by $p_r(t)$ and m denotes the product of the number of antennas and the oversampling factor. The multi-channel $\boldsymbol{h}$ spanning N chips with $m \times 1$ chip rate elements, its poly-phase equivalent[9] $\boldsymbol{h}_p$, the up-down flipped form $\overline{\boldsymbol{h}}_p$ and the poly-phase matched filter $\boldsymbol{h}_p^\dagger$ in row vector format can be written as

$$\boldsymbol{h} = [\boldsymbol{h}[0], \boldsymbol{h}[1], \ldots, \boldsymbol{h}[N-1]] \tag{1.8}$$

$$\boldsymbol{h}_p = \begin{bmatrix} \boldsymbol{h}[0] \\ \boldsymbol{h}[1] \\ \vdots \\ \boldsymbol{h}[N-1] \end{bmatrix}, \quad \overline{\boldsymbol{h}}_p = \begin{bmatrix} \boldsymbol{h}[N-1] \\ \boldsymbol{h}[N-2] \\ \vdots \\ \boldsymbol{h}[0] \end{bmatrix}, \quad \boldsymbol{h}_p^\dagger = \overline{\boldsymbol{h}}_p^H \tag{1.9}$$

1.7 Rake Receiver and LMMSE Chip Equalizer

As shown in Figure 1.13, all the *linear* UE receivers can be mathematically represented in the form a common chip level filter followed by correlators[10].

In order to motivate the discussion of this section, we consider the detection process of a *single* HSPDSCH user symbol $a_1[0]$ transmitted over the $L \times 1$ channelization code $\boldsymbol{c}_1 = \boldsymbol{c}_{16,0}$.

We consider a 2-phase linear filter which has a length of N chips which, as is detailed in Appendix B, is the minimum length to deconvolve, i.e. to zero force, a 2-phase channel with a length of N chips.

[8] although stationarity holds only for time-invariant channels we assume it also for the wireless channels considered in this text which vary slowly

[9] represented in a column format for compatibility with later formulations

[10] the order of the correlator and filtering can change like in the conventional Rake receiver

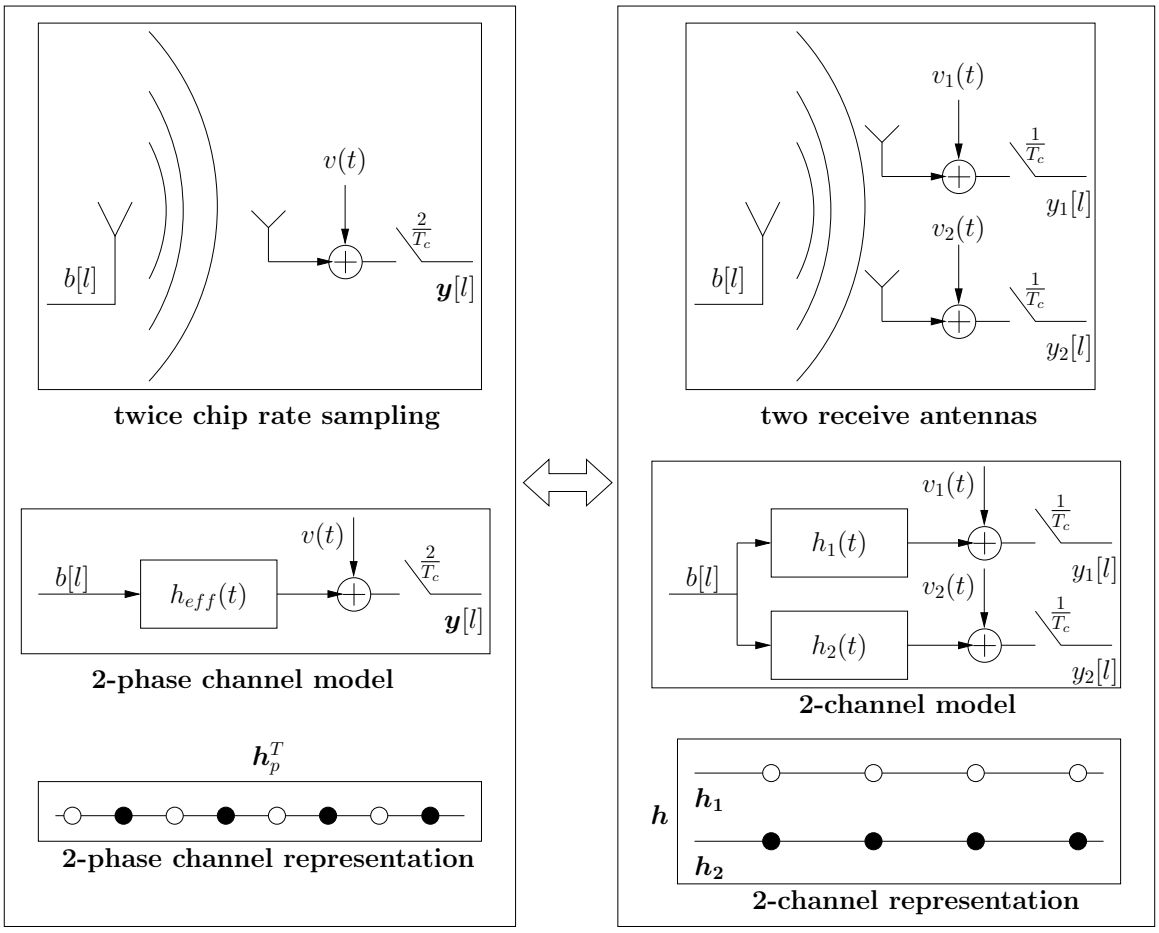


Figure 1.12: The equivalence of the poly-phase and the multi-channel models with a 2-phase example. It is possible to pass from one form to the other by P/S and S/P operations

We denote a block of the received signal as $\boldsymbol{Y}$ and denote a block of the *total* transmitted chip sequence as $\boldsymbol{B}$ whose elements are relevant to the estimation of the latter's subset $\boldsymbol{B}_0 = [b[L-1], \ldots b[0]]^T$ which overlaps with the period of the $a_1[0]$ symbol. $\boldsymbol{Y}$ and $\boldsymbol{B}$ are related by the $2(L+N-1) \times (L+2N-2)$ channel convolution matrix $\boldsymbol{\mathcal{T}}(\boldsymbol{h})$ with the perturbation term $\boldsymbol{V}$ which is modeled as an additive white Gaussian noise (AWGN) representing the sum of the thermal noise and the intercell interference.

$$\boldsymbol{Y} = \boldsymbol{\mathcal{T}}(\boldsymbol{h})\boldsymbol{B} + \boldsymbol{V} \tag{1.10}$$

$$= \begin{bmatrix} \boldsymbol{y}[L+N-2] \\ \vdots \\ \boldsymbol{y}[0] \end{bmatrix} \tag{1.11}$$

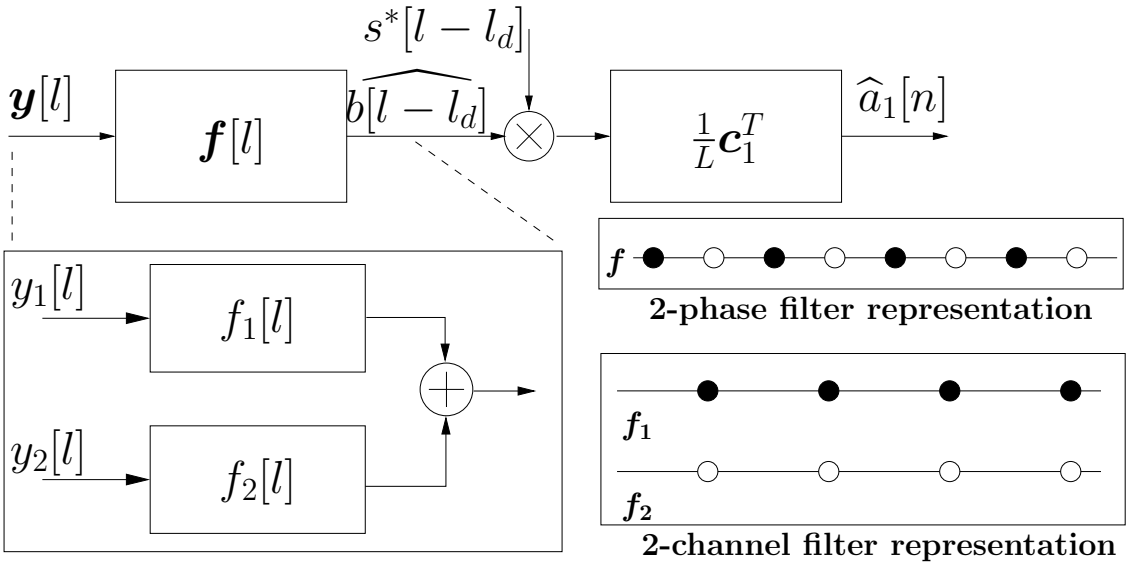


Figure 1.13: Receivers with linear *chip level filter - correlator* cascade. The order of the phases is reversed w.r.t. the channel phases order

$$\boldsymbol{\mathcal{T}}(\boldsymbol{h}) = \begin{bmatrix} \boldsymbol{h}[0] & \dots & \boldsymbol{h}[N-1] & \mathbf{0} & 0 \\ \mathbf{0} & \ddots & \ddots & \ddots & \mathbf{0} \\ 0 & \mathbf{0} & \boldsymbol{h}[0] & \dots & \boldsymbol{h}[N-1] \end{bmatrix}, \quad \boldsymbol{V} = \begin{bmatrix} \boldsymbol{v}[L+N-2] \\ \vdots \\ \boldsymbol{v}[0] \end{bmatrix} \tag{1.12}$$

$$\boldsymbol{B} = \sum_{k=1}^{K} \Big[b_k[L+N-2] \dots b_k[L-1] \dots b_k[0] \dots b_k[-N+1] \Big]^T \tag{1.13}$$

The linear filter $\boldsymbol{f}[l]$ is a 1×2 multi input single output (MISO) system which turns the overall channel to a single input single output (SISO) system $\tilde{g}[l]$.

$$\boldsymbol{f} = [\boldsymbol{f}[0], \boldsymbol{f}[1], \dots, \boldsymbol{f}[N-1]], \quad \boldsymbol{f}[l] = [f[0], f[1]], \quad \tilde{g}[l] = \sum_{i=0}^{l} \boldsymbol{f}[i]\boldsymbol{h}_p[l-i] \tag{1.14}$$

The estimated BS chip sequence $\boldsymbol{B}_0$ can be formulated by the equations

$$\widehat{\boldsymbol{B}}_0 = \boldsymbol{\mathcal{T}}(\boldsymbol{f})\boldsymbol{Y} = \boldsymbol{\mathcal{T}}(\boldsymbol{f})\boldsymbol{\mathcal{T}}(\boldsymbol{h})\boldsymbol{B} + \boldsymbol{\mathcal{T}}(\boldsymbol{f})\boldsymbol{V} = \boldsymbol{\mathcal{T}}(\boldsymbol{g})\boldsymbol{B} + \widetilde{\boldsymbol{V}} \tag{1.15}$$

$$\boldsymbol{\mathcal{T}}(\boldsymbol{f}) = \begin{bmatrix} \boldsymbol{f}[0] & \dots & \boldsymbol{f}[N-1] & \mathbf{0} & 0 \\ \mathbf{0} & \ddots & \ddots & \ddots & \mathbf{0} \\ 0 & \mathbf{0} & \boldsymbol{f}[0] & \dots & \boldsymbol{f}[N-1] \end{bmatrix}, \quad \widetilde{\boldsymbol{V}} = \begin{bmatrix} \widetilde{v}[L-1] \\ \vdots \\ \widetilde{v}[0] \end{bmatrix} \tag{1.16}$$

$$\boldsymbol{\mathcal{T}}(\boldsymbol{g}) = \begin{bmatrix} g[-N+1] & g[-N+2] & \dots & g[0] & \dots & g[N-1] & \mathbf{0} & 0 \\ \mathbf{0} & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \mathbf{0} \\ 0 & \mathbf{0} & g[-N+1] & g[-N+2] & \dots & g[0] & \dots & g[N-1] \end{bmatrix} \tag{1.17}$$

where $\boldsymbol{\mathcal{T}}(\boldsymbol{f})$ denotes the $L \times 2(L+N-1)$ filter convolution matrix, $\boldsymbol{\mathcal{T}}(\boldsymbol{g})$ denotes the $L \times (L+2N-2)$ overall channel convolution matrix and $g[l] = \tilde{g}[l+N-1]$ reflects a variable change in order to better represent the precursor and postcursor parts of the overall channel the central tap of which corresponds to $g[0] = \boldsymbol{f}\overline{\boldsymbol{h}}_p$. The channel matched filter (CMF) $\boldsymbol{f} = \overline{\boldsymbol{h}}_p^H$ which is an equivalent of the filtering part of the conventional Rake receiver maximizes the SNR collecting all the channel energy to the central tap as $g[0] = \left\|\overline{\boldsymbol{h}}_p\right\|^2 = \|\boldsymbol{h}_p\|^2$. The unbiased form of CMF can be written as $\boldsymbol{f} = \left(\boldsymbol{h}_p^H \boldsymbol{h}_p\right)^{-1} \overline{\boldsymbol{h}}_p^H$.

Expanding Equation 1.15, we reach to all the ingredients of $\widehat{\boldsymbol{B}}_0$ containing every user's chip sequences at different windows each scaled by the associated tap of the overall channel

$$\widehat{\boldsymbol{B}}_0 = \sum_{i=-N+1}^{N-1} g[i] \underbrace{\sum_{k=1}^{K} \boldsymbol{B}_{k,i}}_{\boldsymbol{B}_i} + \widetilde{\boldsymbol{V}} \tag{1.18}$$

where $\boldsymbol{B}_{k,i} = [b_k[i+L-1], \dots, b_k[i]]^T$ and $b_k[i] = a_k \left\lfloor \frac{i}{L_k} \right\rfloor c_k[mod(i, L_k)]s[i]$.

The second stage correlation part of the receiver can be written as

$$\widehat{a_1[0]} = \frac{1}{L} \boldsymbol{c}_1^T \boldsymbol{S}_0^* \widehat{\boldsymbol{B}}_0 \tag{1.19}$$

where $\boldsymbol{S}_i$ denotes a diagonal matrix with the scrambling elements $[s[L-1+i], \dots, s[i]]$ and the normalization by L is done to make up for the spreading-despreading gain.

Theorem 1.7.1 *The average SINR of the symbol estimate $\widehat{a_1[0]}$ by the usage of a general chip level linear filter $\boldsymbol{f}$ is equal to*[11] *[17]*

$$\Gamma_1 = \frac{L \left|g(0)\right|^2 \sigma_{b_1}^2}{\left(\|\boldsymbol{g}\|^2 - \left|g(0)\right|^2\right) \sigma_b^2 + \|\boldsymbol{f}\|^2 \sigma_v^2} \tag{1.20}$$

where L is the spreading factor of the first user, $\boldsymbol{g}$ is the impulse response of the channel filter cascade, $\sigma_{b_k}^2$ is the variance of the chips for user k and $\sigma_b^2 = \sum_{k=1}^{K} \sigma_{b_k}^2$.

[11] the redundant multiplicative term L at both the numerator and the denominator parts are kept for the motivation of a later discussion in Chapter 4

Proof. First we give a useful relation

$$\frac{1}{L}\mathbb{E}\{\boldsymbol{c}_1^T\boldsymbol{S}_0^*\boldsymbol{S}_i\boldsymbol{c}_k^T\} = \begin{cases} 1 & i=0, k=1 \\ 0 & i=0, k\neq 1 \\ \frac{1}{L} & i\neq 0 \end{cases} \tag{1.21}$$

The symbol estimate can be partitioned into four groups as

$$\widehat{a_1[0]} = \underbrace{g(0)a_1[0]}_{\text{signal}} + \underbrace{\frac{1}{L}\boldsymbol{c}_1^T\boldsymbol{S}_0^* g[0]\sum_{k=2}^{K}\boldsymbol{B}_{k,0}}_{0} + \underbrace{\frac{1}{L}\boldsymbol{c}_1^T\boldsymbol{S}_0^*\left(\sum_{\substack{i=-N+1\\ i\neq 0}}^{N-1} g[i]\sum_{k=1}^{K}\boldsymbol{B}_{k,i}\right)}_{\text{intracell interference}} + \underbrace{\frac{1}{L}\boldsymbol{c}_1^T\boldsymbol{S}_0^*\widetilde{\boldsymbol{V}}}_{\text{noise}} \tag{1.22}$$

The first component which represents the useful signal part is the symbol of interest scaled by the central channel tap $g[0]$. The second component is zero since at the central tap instant $i=0$, the scrambling and the descrambling blocks are aligned matching each-other, $\boldsymbol{S}_0^*\boldsymbol{S}_0 = \boldsymbol{I}_L$ and preserving the orthogonality among users

$$\boldsymbol{c}_1^T\boldsymbol{S}_0^*\boldsymbol{B}_{k,0} = \boldsymbol{c}_1^T\boldsymbol{S}_0^*\boldsymbol{S}_0\boldsymbol{c}_k a_k[0] = \boldsymbol{c}_1^T\boldsymbol{c}_k a_k[0] = 0 \quad \forall k\neq 1 \tag{1.23}$$

The third *intracell* interference component represents the sum of ICI and MUI from the subcomponents with indices $k=1$ and $k\neq 1$ respectively. The fourth component represents the noise contribution.

Taking the expected value of the symbol estimate power we obtain

$$\mathbb{E}|\widehat{a}_1[0]|^2 = |g(0)|^2\sigma_{a_1}^2 + \frac{1}{L}\sum_{\substack{i=-N+1\\ i\neq 0}}^{N-1}|g[i]|^2\sum_{k=1}^{K}\sigma_{a_k}^2 + \frac{1}{L}\|\boldsymbol{f}\|^2\sigma_v^2 \tag{1.24}$$

where $\sigma_{a_k}^2$ represents the symbol variances, the noise power is amplified by the filter energy as is observed in the third component and the cross terms in the second component disappear by the expectation relation $\mathbb{E}\{a_k[0]a_l^*[0]\} = 0, k\neq l$.

Using the equalities $\|\boldsymbol{f}\|^2\sigma_v^2 = \boldsymbol{f}\boldsymbol{R}_{vv}\boldsymbol{f}^H$ and $\|\boldsymbol{g}\|^2 = \boldsymbol{f}\boldsymbol{\mathcal{T}}(\boldsymbol{h})\boldsymbol{\mathcal{T}}(\boldsymbol{h})^H\boldsymbol{f}^H$, $\sigma_{b_k}^2 = \sigma_{a_k}^2$ due to the fact the channelization codes are not normalized, we obtain

$$\begin{aligned}\mathbb{E}|\widehat{a}_1[0]|^2 &= |g(0)|^2\sigma_{a_1}^2 + \frac{1}{L}\left(\|\boldsymbol{g}\|^2 - |g(0)|^2\right)\sigma_b^2 + \frac{1}{L}\|\boldsymbol{f}\|^2\sigma_v^2 \\ &= |g(0)|^2\sigma_{a_1}^2 + \frac{1}{L}\boldsymbol{f}\left(\underbrace{\sigma_b^2\boldsymbol{\mathcal{T}}(\boldsymbol{h})\boldsymbol{\mathcal{T}}(\boldsymbol{h})^H + \boldsymbol{R}_{vv}}_{\boldsymbol{R}_{yy}}\right)\boldsymbol{f}^H - \frac{1}{L}|g(0)|^2\sigma_b^2\end{aligned}$$

Accordingly we reach to the SINR expression

$$\Gamma_1 = \frac{|g(0)|^2 \sigma_{a_1}^2}{\frac{1}{L}\left(\|\boldsymbol{g}\|^2 - |g(0)|^2\right)\sigma_b^2 + \frac{1}{L}\|\boldsymbol{f}\|^2 \sigma_v^2} \tag{1.25}$$

$$= \frac{|g(0)|^2 \sigma_{b_1}^2}{\frac{1}{L}\left(\|\boldsymbol{g}\|^2 - |g(0)|^2\right)\sigma_b^2 + \frac{1}{L}\|\boldsymbol{f}\|^2 \sigma_v^2} \tag{1.26}$$

$$= \frac{L|g(0)|^2 \sigma_{b_1}^2}{\boldsymbol{f}\boldsymbol{R}_{yy}\boldsymbol{f}^H - |g(0)|^2 \sigma_b^2} \tag{1.27}$$

□

Although the SINR expression is given as a metric for the estimation of symbols, in reality the linear filter $\boldsymbol{f}$ estimates the BS chip sequence $b[l]$. Therefore the modified SINR expression for the estimation of the BS chip sequence can be written as

$$\Gamma_c = \frac{|g(0)|^2 \sigma_b^2}{\boldsymbol{f}\boldsymbol{R}_{yy}\boldsymbol{f}^H - |g(0)|^2 \sigma_b^2} \tag{1.28}$$

where at the numerator, i.e. the useful energy part, there is no spreading gain and $\sigma_{b_1}^2$ is replaced by σ_b^2.

The SINR metrics in Equation 1.25 and in Equation 1.28 are based on the estimation of $\boldsymbol{R_{yy}}$ statistics by taking expectation over the scrambler which is modeled as a random sequence and by using the orthogonality property of the codes. The receiver that maximizes these SINR metrics is the Max-SINR receiver which is more often known as chip level LMMSE receiver [16, 17].

Theorem 1.7.2 *The unbiased linear filter which achieves the maximum performance in terms of the SINR metric without exploiting the code and the power knowledge of the active users but by modeling the scrambling sequence as a random sequence and by taking expectations over it to approximate the received signal covariance matrix* $\boldsymbol{R_{yy}}$ *is equal to [17]*

$$\boldsymbol{f}_o = \left(\overline{\boldsymbol{h}}_p^H \boldsymbol{R}_{\boldsymbol{yy}}^{-1} \overline{\boldsymbol{h}}_p\right)^{-1} \overline{\boldsymbol{h}}_p^H \boldsymbol{R}_{\boldsymbol{yy}}^{-1} \tag{1.29}$$

Proof. We first define the unbiasedness constraint as $g[0] = \boldsymbol{f}_o \overline{\boldsymbol{h}}_p = 1$. Then the optimization problem can be formulated as

$$\boldsymbol{f}_o = \arg_{\boldsymbol{f}} \max_{\boldsymbol{f}_o \overline{\boldsymbol{h}}_p = 1} \Gamma_1 = \arg_{\boldsymbol{f}} \min_{\boldsymbol{f}_o \overline{\boldsymbol{h}}_p = 1} \boldsymbol{f}\boldsymbol{R}_{\boldsymbol{yy}}\boldsymbol{f}^{\boldsymbol{H}} \tag{1.30}$$

The solution can be obtained by the standard Lagrange multiplier technique as follows:

$$\begin{aligned}
\Omega(\boldsymbol{f}^H, \boldsymbol{f}) &= \boldsymbol{f}\boldsymbol{R}_{\boldsymbol{yy}}\boldsymbol{f}^{\boldsymbol{H}} + 2\Re\left[\lambda\left(\boldsymbol{f}\overline{\boldsymbol{h}}_p - 1\right)\right] \\
\nabla_{\boldsymbol{f}}\Omega(\boldsymbol{f}^H, \boldsymbol{f}) &= \boldsymbol{f}\boldsymbol{R}_{\boldsymbol{yy}} + \lambda\overline{\boldsymbol{h}}_p^H \\
&\Rightarrow \quad \boldsymbol{f}_o = -\lambda\overline{\boldsymbol{h}}_p^H\boldsymbol{R}_{\boldsymbol{yy}}^{-1} \\
\boldsymbol{f}_o\overline{\boldsymbol{h}}_p &= 1 \Rightarrow -\lambda\overline{\boldsymbol{h}}_p^H\boldsymbol{R}_{\boldsymbol{yy}}^{-1}\overline{\boldsymbol{h}}_p = 1 \\
&\Rightarrow \quad \lambda = \frac{-1}{\overline{\boldsymbol{h}}_p^H\boldsymbol{R}_{\boldsymbol{yy}}^{-1}\overline{\boldsymbol{h}}_p} \\
&\Rightarrow \quad \boldsymbol{f}_o = \frac{\overline{\boldsymbol{h}}_p^H\boldsymbol{R}_{\boldsymbol{yy}}^{-1}}{\overline{\boldsymbol{h}}_p^H\boldsymbol{R}_{\boldsymbol{yy}}^{-1}\overline{\boldsymbol{h}}_p} = \frac{\boldsymbol{h}_p^\dagger\boldsymbol{R}_{\boldsymbol{yy}}^{-1}}{\boldsymbol{h}_p^\dagger\boldsymbol{R}_{\boldsymbol{yy}}^{-1}\overline{\boldsymbol{h}}_p}
\end{aligned}$$

□

By taking an approximation of Equation 1.29, we can obtain a slightly biased but simpler chip level LMMSE filter

$$\tilde{\boldsymbol{f}}_o = \sigma_b^2\boldsymbol{h}_p^\dagger\boldsymbol{R}_{\boldsymbol{yy}}^{-1} = \boldsymbol{R}_{b\boldsymbol{y}}\boldsymbol{R}_{\boldsymbol{yy}}^{-1} \tag{1.31}$$

which fits to the Wiener filtering format.

1.8 State of the Art Multiuser Receivers

1.8.1 Optimal Receiver

The optimal multiuser detector in terms of minimum symbol error rate (SER) is the Maximum Likelihood Sequence Estimation (MLSE) which is an exhaustive search procedure over the symbol alphabets of all the possible transmitted sequences of all the users with the minimization criterion [22]

$$\hat{\boldsymbol{d}}_{ML} = \arg\min_{\boldsymbol{d}\in\mathcal{X}^{MK}} \left\| \boldsymbol{Y} - \overbrace{\boldsymbol{\mathcal{T}}(\boldsymbol{h})\boldsymbol{S}\boldsymbol{C}}^{\tilde{\boldsymbol{G}}} \underbrace{\boldsymbol{\mathcal{A}}\boldsymbol{d}}_{\boldsymbol{A}} \right\|^2 \tag{1.32}$$

where $\mathcal{X}$, K, $\boldsymbol{Y}$, $\tilde{\boldsymbol{G}}$, $\boldsymbol{\mathcal{T}}(\boldsymbol{h})$, $\boldsymbol{S}$, $\boldsymbol{C}$, $\boldsymbol{d}$, $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{A}$ respectively denote the symbol alphabet[12], the number of users, the received sample block spanning the channel output of M symbol periods transmission, symbol level channel transfer function, channel convolution matrix,

[12] representing the simple case of same constellation for all users

diagonal scrambling matrix, block-diagonal channelization codes matrix, unit-amplitude MK desired symbols vector, diagonal symbol amplitudes matrix and diagonal amplitude-scaled symbols vector. Since this criterion is finite-alphabet constrained, it is NP-hard[13] and perhaps the best that can be done is to use the Viterbi algorithm which is also exponentially complex with the factor MK [23]. Due to these reasons, ML receiver is mostly considered as not implementable and its performance serves only as an upper-bound. Note that (E.1) is at the same time the nonlinear LS estimator[14].

1.8.2 Decorrelation Receiver

One of the suboptimal but simpler approaches is to relax the finite alphabet constraint by fake mapping of $\boldsymbol{d}$ from the finite set $\mathcal{X}^{MK}$ into $\mathbb{C}^{MK}$ which turns the nonlinear LS problem in (E.1) into a linear LS problem

$$\hat{\boldsymbol{A}}_{LS} = \arg \min_{\boldsymbol{A} \in \mathbb{C}^{MK}} \left\| \boldsymbol{Y} - \tilde{\boldsymbol{G}} \boldsymbol{A} \right\|^2 \tag{1.33}$$

whose solution is

$$\hat{\boldsymbol{A}}_{LS} = \tilde{\boldsymbol{F}}_{Dec} \boldsymbol{Y} = \left(\overbrace{\tilde{\boldsymbol{G}}^H \tilde{\boldsymbol{G}}}^{\boldsymbol{R}} \right)^{-1} \tilde{\boldsymbol{G}}^H \boldsymbol{Y} = \boldsymbol{R}^{-1} \boldsymbol{X} \tag{1.34}$$

where $\boldsymbol{X}$ and $\boldsymbol{R}$ respectively denote the single user matched filter (SUMF) bank output symbol estimates and the symbol cross-correlation matrix.

An equivalent model in terms of linear systems of equations can be written as

$$\boldsymbol{R} \hat{\boldsymbol{A}}_{LS} = \boldsymbol{X} \tag{1.35}$$

Note that LS estimator treats the elements of $\boldsymbol{A}$ vector as deterministic unknown parameters having diffuse prior pdfs.

LS estimation, i.e. decorrelation, is the unique least MSE member of the ZF MUDs set with the general members expressed as

$$\hat{\boldsymbol{A}}_{ZF} = \left(\boldsymbol{T} \tilde{\boldsymbol{G}} \right)^{-1} \boldsymbol{T} \boldsymbol{Y}$$

with any proper $\boldsymbol{T}$ matrix.

[13] a decision problem which is at least as hard as any problem whose solution can be *verified* by polynomial complexity

[14] in statistical terms, LS is *disguised* ML when the measurement noise sequence $\boldsymbol{V}$ is zero-mean, i.i.d and Gaussian

1.8.3 LMMSE Receiver

Although completely deconvolves $\tilde{\boldsymbol{G}}$, the decorrelator $\tilde{\boldsymbol{F}}_{Dec}$ amplifies the noise term $\boldsymbol{V}$. A better approach is the LMMSE estimator which models $\boldsymbol{A}$ as a random Gaussian vector and solves the cost criterion

$$\tilde{\boldsymbol{F}}_{LMMSE} = \arg_{\tilde{\boldsymbol{F}}} \min_{\boldsymbol{A}\in\mathbb{C}^{MK}} \mathbb{E}\left(\tilde{\boldsymbol{F}}\boldsymbol{Y} - \boldsymbol{A}\right)\left(\tilde{\boldsymbol{F}}\boldsymbol{Y} - \boldsymbol{A}\right)^{H} \tag{1.36}$$

with the solution

$$\tilde{\boldsymbol{F}}_{LMMSE} = \left(\tilde{\boldsymbol{G}}^{H}\tilde{\boldsymbol{G}} + \sigma_v^2\boldsymbol{\mathcal{A}}^{-2}\right)^{-1}\tilde{\boldsymbol{G}}^{H} \tag{1.37}$$

which different from the decorrelator requires also the symbol amplitudes $\boldsymbol{\mathcal{A}}$. Note that for vanishing noise $\tilde{\boldsymbol{F}}_{LMMSE}$ becomes equivalent to the decorrelator. For high noise, on the other hand, it is identical to the SUMF.

The equivalent model in terms of linear systems of equations can be written as

$$\underbrace{\left(\boldsymbol{R} + \sigma_v^2\boldsymbol{\mathcal{A}}^{-2}\right)}_{\boldsymbol{M}}\hat{\boldsymbol{A}}_{LMMSE} = \boldsymbol{X} \tag{1.38}$$

where $\boldsymbol{M}$ denotes the SUMF bank output covariance matrix. Both decorrelator and LMMSE receiver are very complex due to the fact they require matrix-inversion operations with $O(M^3K^3)$ complexity. Therefore reduced rank approximations of the matrix inversion operation have been highly investigated in literature with iterative techniques. We will elaborate on only the so-called Parallel Interference Cancellation (PIC) family which is the counterpart of Jacobi iterations for the iterative solutions of linear systems of equations since it works particularly well when user symbols have close power levels which is the case for HSDPA. For other state of the art iterative techniques which are not the topics of this thesis such as Successive Interference Cancellation (SIC) which is the counterpart of Gauss-Seidel iterations in Matrix Algebra or the Decorrelating Decision Feedback Equalizer, see respectively [24] and [25].

1.8.4 Linear Parallel Interference Cancellation Receiver

Conventional LPIC corresponds to using Jacobi iterations for the solutions of linear system of equations [26]. Splitting the $\boldsymbol{R}$ expression in (1.35) into the two parts as $\boldsymbol{I}$ and $(\boldsymbol{R} - \boldsymbol{I})$ one can reach to the iterative decorrelation solution as[15]

$$\hat{\boldsymbol{A}}_{LS}^{(i)} = (\boldsymbol{I} - \boldsymbol{R})\hat{\boldsymbol{A}}_{LS}^{(i-1)} + \boldsymbol{X} \tag{1.39}$$

[15] similarly splitting $\boldsymbol{M}$ in (1.38) for the LMMSE receiver

The iterations converge provided that the spectral radius $\rho(\boldsymbol{I}-\boldsymbol{R})$ is less than 1, which is not guaranteed[16].

A better approach is to attack the problem from Cayley-Hamilton theorem which states that every square matrix satisfies its characteristic equation. This principle can be used to find the inverse of an $n \times n$ square matrix by a polynomial expansion as [27]

$$\begin{aligned} det(\boldsymbol{R}-\lambda\boldsymbol{I}) &= 0 \\ &\Rightarrow 1 - c_1\lambda - \ldots - c_{n-1}\lambda^{n-1} - c_n\lambda^n = 0 \\ &\Rightarrow \boldsymbol{I} - c_1\boldsymbol{R} \ldots - c_{n-1}\boldsymbol{R}^{n-1} - c_n\boldsymbol{R}^n = \boldsymbol{0} \\ &\Rightarrow \boldsymbol{I} = c_1\boldsymbol{R} + \ldots + c_{n-1}\boldsymbol{R}^{n-1} + c_n\boldsymbol{R}^n \\ &\Rightarrow \boldsymbol{R}^{-1} = c_1\boldsymbol{I} \ldots + c_{n-1}\boldsymbol{R}^{n-2} + c_n\boldsymbol{R}^{n-1} \end{aligned}$$

With polynomial expansion it is possible to obtain the decorrelator solution or the LMMSE solution in n iterations. Suboptimal solutions are obtained by stopping at a few iterations in which case the optimal weights change as well. Although looks as an attractive solution at first sight, the complexity depends on the weight adaptation. See [28] and [29] for two adaptation schemes one from the direct derivation from the MMSE expression for a particular number of iterations and one from large system analysis respectively. In the thesis we are not concerned with weight adaptation but instead with filter adaptation.

[16] $\rho(\boldsymbol{X}) = \max\{|\lambda|, \lambda \in \Lambda(\boldsymbol{X})\}$ where $\Lambda(\boldsymbol{X})$ is the eigenvalue matrix of $\boldsymbol{X}$

Chapter 2

Performance Analysis of Hsdpa Receiver Models

In this chapter we obtain the maximum achievable SINR and throughput performance metrics for various HSDPA service deployment scenarios by the usage of CMF and LMMSE equalizer type UE receiver components under particular residual intracell interference levels, which represent the situations after the possible usage of front end intracell interference cancellers. The distributions of the radio channel parameters and the received powers from the own and surrounding base stations are modeled under correlated shadowing w.r.t. the mobile position, the cell radius and the type of environment. From such modeling, more realistic performance figures might be obtained as compared to fixing them to a selected set of values.

2.1 Introduction

The goal of this chapter is to make broad computations of the achievable performance gains by the possible usage of various different types of advanced receivers instead of the CMF which is an FIR counterpart of the filtering part of the conventional Rake receiver which has been the preferred receiver for the initial stages of the UMTS development due to its simplicity. Although it is well known that Rake receiver is far from being optimal in multipath-rich environments, mobile vendors and chip manufacturers have been reluctant to switch to a more advanced and hence a more costly solution for the sole benefit of the base station side. In HSDPA, however, such a solution is meaningful due to the fact that the mobile terminal directly benefits from it by not only obtaining more data rate on HSPDSCHs once a connection is established during the scheduling process but also by increasing the probability of getting a connection if fairness is partially sacrificed for throughput in user

scheduling.

2.2 Hypothetical Receiver Models

We consider that possibly an IC structure is used in the first stage to cancel out one or both of the intracell interference contributions of the PCPICH and HSPDSCHs since we know their channelization codes and symbol constellations, even the symbol values for the PCPICH.

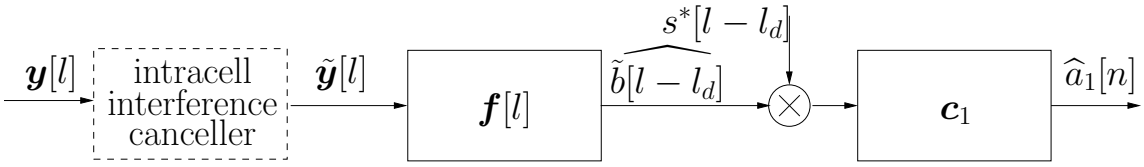


Figure 2.1: Hypothetical receiver model

We assume that the residual BS signal $\tilde{b}[l]$ contained in the remaining sequence $\tilde{\boldsymbol{y}}[l]$ after the IC block is still block stationary and the second order intercell interference plus noise statistics σ_ν^2 is the same as before.

The modified SINR expression at the unbiased linear filter output, i.e. when $g[0] = 1$, for the symbol estimates of a single HSPDSCH channel of a UE situated at a particular position of the cell can be written as:

$$\Gamma_1 = \frac{16\rho P_0}{(\frac{1}{\alpha}-1)\chi P_0 + \sum_{i=1}^{6} P_i \|\boldsymbol{g}_i\|^2 + \|\boldsymbol{f}\|^2 \sigma_n^2} \tag{2.1}$$

where 16 is the HSPDSCH spreading gain, P_0 is the received power of own BS, ρ is the BS signal power portion of one HSPDSCH channel, χ is the remaining BS signal power portion after the IC block, P_i are the received powers from the six first-tier interfering unsectored cells[1] shown in Figure 1.2, $\boldsymbol{g}_i$ is the convolution of the linear filter $\boldsymbol{f}$ and the channel $\boldsymbol{h}_i$ originating from this surrounding cell as $\boldsymbol{g}_i = \boldsymbol{f} * \boldsymbol{h}_i$ and σ_n^2 is the AWGN variance. AWGN term and intercell interference are treated separately for performance analysis purposes whereas they are treated together for filter adaptation due to the fact that it is impractical to incorporate the channel estimates and the signal variance estimates of the surrounding cells. As is exemplified by the demonstration of channel and CMF impulse responses in

[1] the analysis done in this chapter is valid for also the sectored case

Figure 2.2 the term $\alpha = \frac{1}{\|\boldsymbol{g}\|^2}$ represents the ratio of the useful effective channel energy to the total effective channel energy and is known as the *orthogonality factor* which has been treated in the literature only for the Rake receiver variants [30, 31].

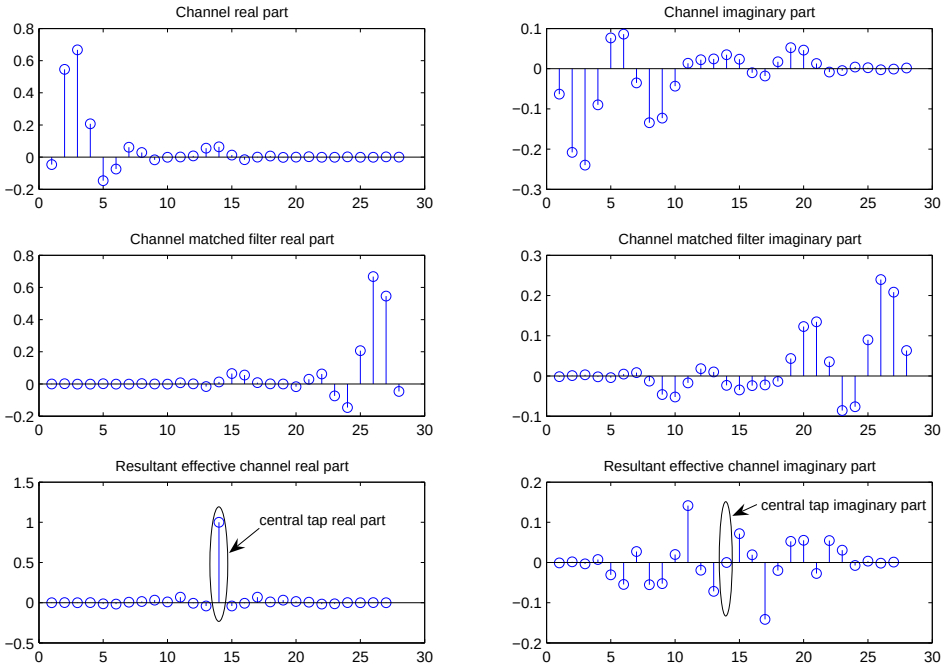


Figure 2.2: Orthogonality factor representation over an unbiased CMF. The central tap of the effective channel collects all the channel energy which is 1 due to unbiasedness. The cumulative energy of all the taps is $\|\boldsymbol{g}\|^2$.

2.3 Parameter Modeling

In this section, we model all the parameters which take part in the SINR expression and which implicitly or explicitly depend on one or more of the location of the UE in the cell, the radius (r) of the cell and the type of the environment.

2.3.1 Modeling the Received Powers

Received powers are calculated by the path loss and shadowing computations covered in Section 1.6.1. As a single difference, for shadowing we first randomly generate a vector of seven independent shadowing values $\tilde{\boldsymbol{x}}$ of the own and first-tier six cells and turn

it into a cross-correlated vector $\boldsymbol{x}$ by left multiplying with the lower triangular Cholesky factorization output matrix $\boldsymbol{L_x}$ of the symmetric shadowing correlation matrix $\boldsymbol{R_{xx}}$ whose elements $\rho_{x_i x_j}$ given in Table 2.1 are obtained from the distance ratio dr_{ij} and the angle values θ_{ij} between the corresponding couples among the seven BSs and the UE as shown in Figure 2.3 [32].

$$\boldsymbol{R_{xx}} = \begin{bmatrix} \rho_{x_0,x_0} & \rho_{x_0,x_1} & \cdots & \rho_{x_0,x_6} \\ \rho_{x_1,x_0} & \rho_{x_1,x_1} & \cdots & \rho_{x_1,x_6} \\ \vdots & \vdots & \vdots & \vdots \\ \rho_{x_6,x_0} & \rho_{x_6,x_1} & \cdots & \rho_{x_6,x_6} \end{bmatrix} = \boldsymbol{L_x} \boldsymbol{L_x}^T, \quad \boldsymbol{x} = \boldsymbol{L_x} \tilde{\boldsymbol{x}} \tag{2.2}$$

Table 2.1: Shadowing correlation matrix elements

	$0 < \theta_{ij} < 30°$	$30° \leq \theta_{ij} < 60°$	$60° \leq \theta_{ij} < 90°$	$90° \leq \theta_{ij}$
$dr_{ij} \in [0, 2]$	$\rho_{x_i x_j} = 0.8$	$\rho_{x_i x_j} = 0.5$	$\rho_{x_i x_j} = 0.4$	$\rho_{x_i x_j} = 0.2$
$dr_{ij} \in [2, 4]$	$\rho_{x_i x_j} = 0.6$	$\rho_{x_i x_j} = 0.4$	$\rho_{x_i x_j} = 0.4$	$\rho_{x_i x_j} = 0.2$
$dr_{ij} \geq 4$	$\rho_{x_i x_j} = 0.4$	$\rho_{x_i x_j} = 0.2$	$\rho_{x_i x_j} = 0.2$	$\rho_{x_i x_j} = 0.2$

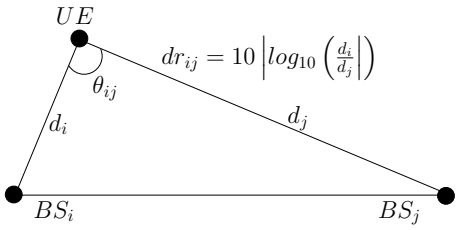


Figure 2.3: Distance and angle relations between two BSs and a UE

2.3.2 Modeling the Channel Parameters

The linear filter $\boldsymbol{f}$, the orthogonality factor α and the $\boldsymbol{g}_i$ terms depend on the channel parameters for which we refer to Greenstein's channel model derived from the rms delay spread σ_{τ_i} and the power delay profile $P(\tau_i)$ [33]. Delay spread is equal to $\sigma_{\tau_i} = T_1 d^{\epsilon} y_i$ where T_1 is the reference delay spread at 1km distance from the BS, ϵ is the model parameter which is around 0.5 for almost all types of environments except very irregular mountainous terrains and y_i is a coefficient which is log-normally distributed with geometric mean 0 and geometric standard deviation σ_{y_i}. From field tests $log(y_i)$ is also observed to be correlated

with $log(x_i)$ by a factor $\rho_{x_i y_i} = -0.75$, [33]. So we obtain the value of y_i from the correlation with the obtained shadowing value in the previous section. From the obtained delay spread we generate the power delay profile as $P(\tau_i) \propto e^{-\tau_i/\sigma_{\tau_i}}$ where $\propto$ is the proportionality sign and τ_i values are the sampling instants. Since this is an infinite length sequence, we truncate it at the position where the final significant tap has $15dB$ less power than the first tap. Then we pass the discrete power delay profile through *Rayleigh* fading to generate the *propagation* channel. The *transmission* channels are obtained by convolving the obtained propagation channels with the pulse shape and normalizing the result to unit energy.

2.3.3 Modeling χ

Among the common downlink channels, the pilot tone PCPICH has the highest interference with 10% BS power portion and it can be cancelled with high accuracy [34]. However it might be even more meaningful to consider cancelling the interference of HSPDSCH codes since by a highly probable deployment scenario, they will carry the majority (if not all) of the data traffic. We contemplate this because it would be easier to manage for an operator to dedicate one of its two or three carriers of 5Mhz to the HSDPA service instead of distributing it over two or three available carriers. Furthermore there is no justified advantage of carrying high-rate data on a DCH with a very low spreading factor instead of on multiple HSPDSCHs. So, in the reception chain for a single HSPDSCH, we define five *perfect* first stage interference cancellation (IC) scenarios:

1. No interference canceller exists: $\chi = 1$
2. Pilot tone cancelled: $\chi = 0.9$
3. All the other HSPDSCHs cancelled: $\chi = 1 - (K_1 - 1)\rho$
4. Pilot+HSPDSCHs cancelled: $\chi = 0.9 - (K_1 - 1)\rho$
5. All intracell interference cancelled: $\chi = \rho$

2.4 Simulations and Conclusions

Five different environments are considered, the relevant parameters of which are shown in Table 2.2 which are adopted from COST231 propagation models [35] and from [33]. We fix transmitted BS power and AWGN power to $P_t = 43dBm$ and $\sigma_n^2 = -102dBm$. Three different low-end to high-end HSDPA service scenarios are considered with $\{K_1, \rho\}$ sets as $\{1, 0.1\}$,$\{5, 0.06\}$ and $\{10, 0.06\}$.

We uniformly position 10^4 UEs in the own cell and approximately that many more in an expanded region penetrating into other cells considering the effect of shadowing handing off some UEs to not-closest BSs. We also exclude a few closest points to the BS since they would otherwise overwhelm all the other UEs. For each node receiving the highest power from the BS of interest, we determine the relevant second order statistics and make 10 Rayleigh fading channel realizations. At each realization we obtain the SINR and spectral efficiency bound $\mathcal{T} = log_2(1 + \Gamma_1)$ results for the CMF and LMMSE equalizer-correlator type receivers under five mentioned interference cancellation scenarios. It was shown that the interference at the output of multiuser detectors can be approximated by Gaussian distribution [36, 37]. Hence $\mathcal{T}$ is an approximate Shannon capacity and it is a more meaningful measure than SINR since it defines the overall performance bound that can be achieved by the usage of efficient transmission diversity, modulation and channel coding schemes. From such around 10^5 spatio-temporal results, we can obtain the distribution of $\mathcal{T}$. Cumulative distributions of $\mathcal{T}$ for 10 HSPDSCH codes deployment in the five reference environments are shown in Fig. 2.4 to Fig. 2.8. The calculated median values of $\mathcal{T}$ for all the defined settings are tabulated in Table 2.3.

On the figures and the table, C represents CMF; E represents LMMSE equalizer-correlator receiver; suffixes to C and E ($\{1, \ldots, 5\}$) represent in the same order the IC scenarios defined in section 2.3.3; {ind, umi, uma, sub} represent {indoor, urban microcell, urban macrocell, suburban macrocell} environments; the suffixes $\{1, 5, 10\}$ to these environments represent K_1.

As observed in the figures, an increasing gap occurs between matched filtering results and equalization results when we go to user locations closer to the own BS which correspond to higher SINR regions. This is especially the valid case for indoor cells, urban microcells and urban macrocells where the eye is open for all user locations since the white noise (the thermal noise and the partial intercell interference) suppressing CMF is much more effected by the intracell interference most of which however is suppressed and the need for an IC decreases when an equalizer is used. In other words, in such environments orthogonality factors at the output of LMMSE equalizers are much higher than those of CMFs.

In the suburban macrocell sizes, for the most distant %30 cellular positions, there is no difference in the performance of receivers. When we further go to the extreme rural cell sizes, there is almost no difference except at a small number of very close UE positions.

These figures clearly show the dominancy of multiuser interference in small cells where using interference suppressing equalizers becomes meaningful and the dominancy of AWGN in the large cells where CMF or Rake receivers are sufficient.

According to UMTS deployment scenarios, more than %80 of UMTS cells will be pico or micro cells and hence it will certainly pay off if a UE considers the LMMSE equalizer in order to be scheduled for a high SINR demanding HSDPA service. In these settings, the achievable *maximum* $\mathcal{T}$ by using equalizers is approximately twice that of CMFs. So, in an ideal condition, CMF has less chance of providing a very high rate demanding application.

In Table 2.3, we notice that when an equalizer is used, the median capacity for a UE increases when we move from indoor to urban environments which is mostly because of the trend of path loss exponent, for when it is low, intercell interference will be high. However as we further increase the size of the cell, AWGN starts to dominate and median capacity decreases. We also see that w.r.t. CMF, equalizers alone improve the median capacity of pico and micro cells between %60 and %115. When complete interference cancellation is achieved these figures increase to %98 and %199.

Cancelling the pilot tone alone brings very little gain. Moshavi et.al however claim that it is possible to obtain %11 capacity gain by cancelling the %10 power pilot tone since this much cancelled power can be exploited by the BS to accept a proportional number of new users [38]. This can be only valid if all the UE receivers at the same time cancel the pilot tone which is not dictated for the moment by the standard.

Note that the obtained results are valid when there is no LOS and surrounding cells have identical properties. In reality, we expect higher capacity from picocellular regions since they will be some isolated hot zones like airports and there will be a higher probability of LOS. Furthermore, note that capacity we are here concerned with is the single cell capacity. Of course, global capacity from the adoption of picocells will be much higher than others since there will be more cells and hence more users will be served.

Table 2.2: Cellular deployment scenarios

PARAMETERS	G_1	n	r	T_1	σ_x	σ_y
Indoor	138	2.6	0.2	0.4	12	2
Urbanmicro	131	3	0.5	0.4	10	3
Urbanmacro	139.5	3.5	1	0.7	8	4
Suburbanmacro	136.5	3.5	2	0.3	8	5
Rural	136.5	3.85	8	0.1	6	6

Table 2.3: Throughput bound median results

$\mathcal{T}$	C1	C2	C3	C4	C5	E1	E2	E3	E4	E5
ind1	2.46	2.54	2.46	2.54	4.54	3.94	3.99	3.94	3.99	4.87
ind5	1.86	1.94	2.09	2.21	4.13	3.22	3.27	3.33	3.42	4.34
ind10	1.86	1.96	2.53	2.73	4.11	3.31	3.35	3.63	3.73	4.30
umi1	2.18	2.29	2.18	2.29	4.54	4.16	4.20	4.16	4.20	5.30
umi5	1.65	1.74	1.89	2.02	4.34	3.56	3.59	3.71	3.77	4.94
umi10	1.70	1.79	2.41	2.63	4.32	3.59	3.63	4.00	4.16	4.95
uma1	1.78	1.88	1.78	1.88	4.13	3.80	3.86	3.80	3.86	5.16
uma5	1.30	1.37	1.50	1.62	3.97	3.09	3.14	3.26	3.35	4.72
uma10	1.30	1.38	1.95	2.15	3.91	3.10	3.17	3.58	3.74	4.50
sub1	1.11	1.17	1.11	1.17	1.60	1.40	1.42	1.40	1.42	1.61
sub5	0.79	0.82	0.87	0.91	1.19	1.02	1.03	1.05	1.06	1.19
sub10	0.77	0.81	0.97	1.02	1.18	0.98	1.00	1.07	1.09	1.18

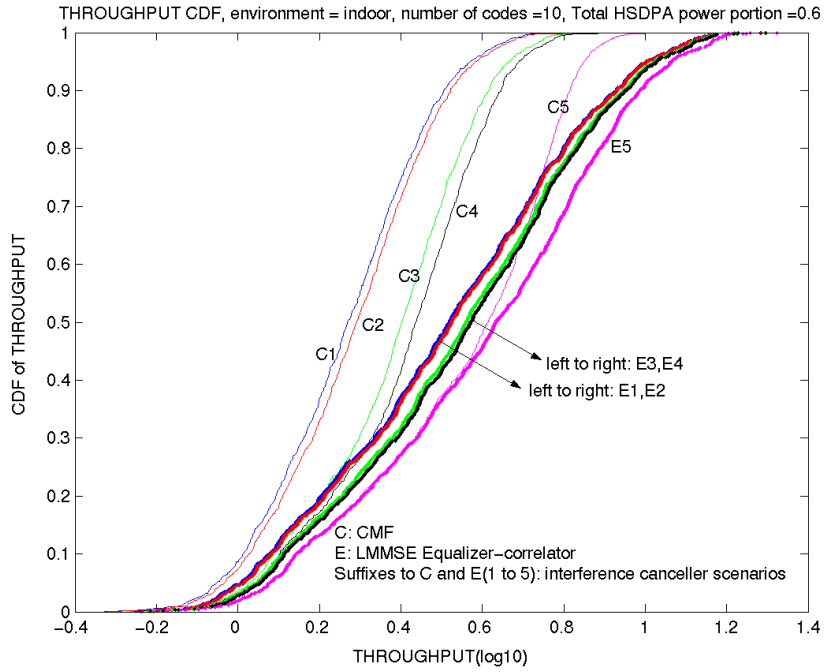


Figure 2.4: Throughput bound CDF of indoor microcell

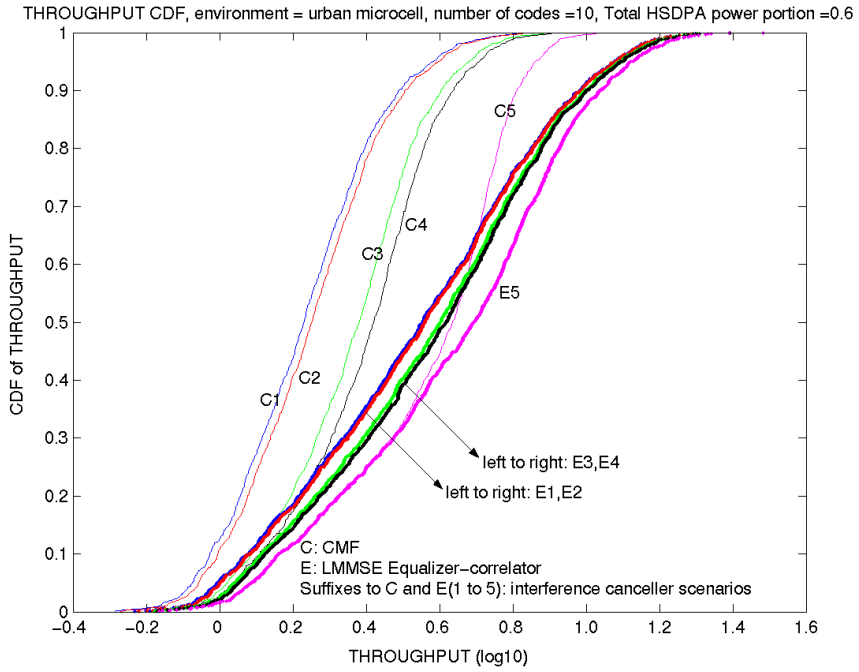


Figure 2.5: Throughput bound CDF of urban microcell

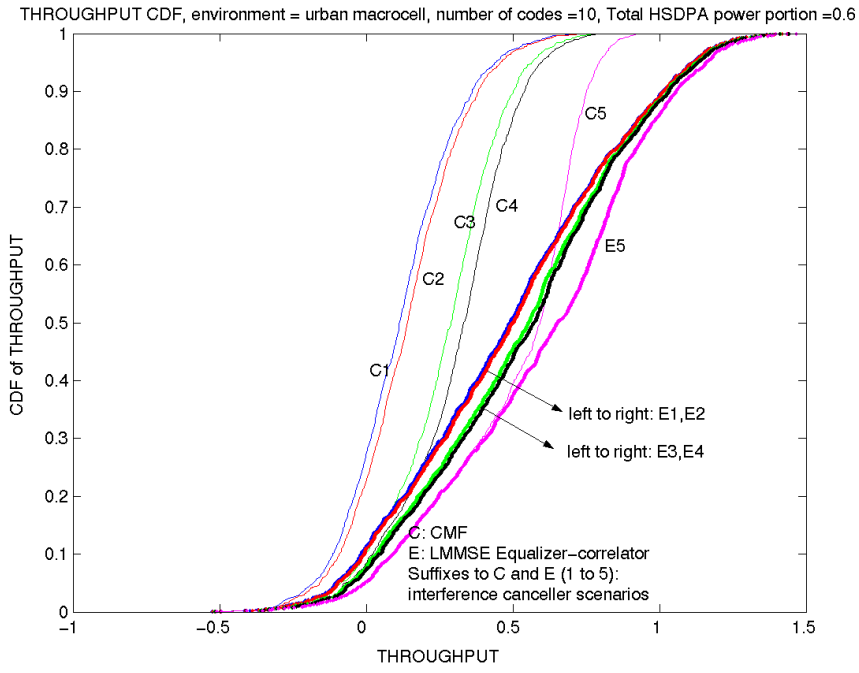


Figure 2.6: Throughput bound CDF of urban macrocell

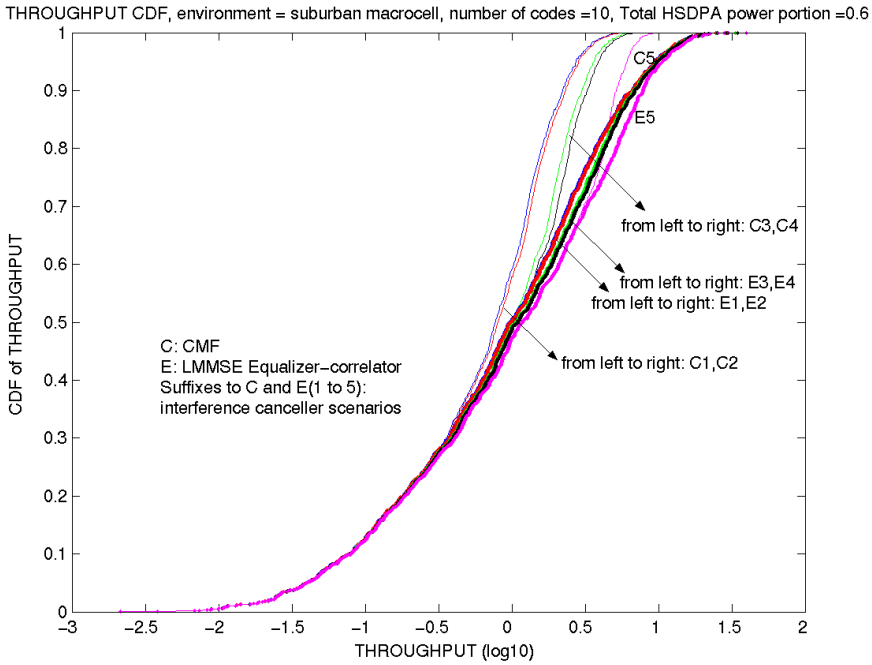


Figure 2.7: Throughput bound CDF of suburban macrocell

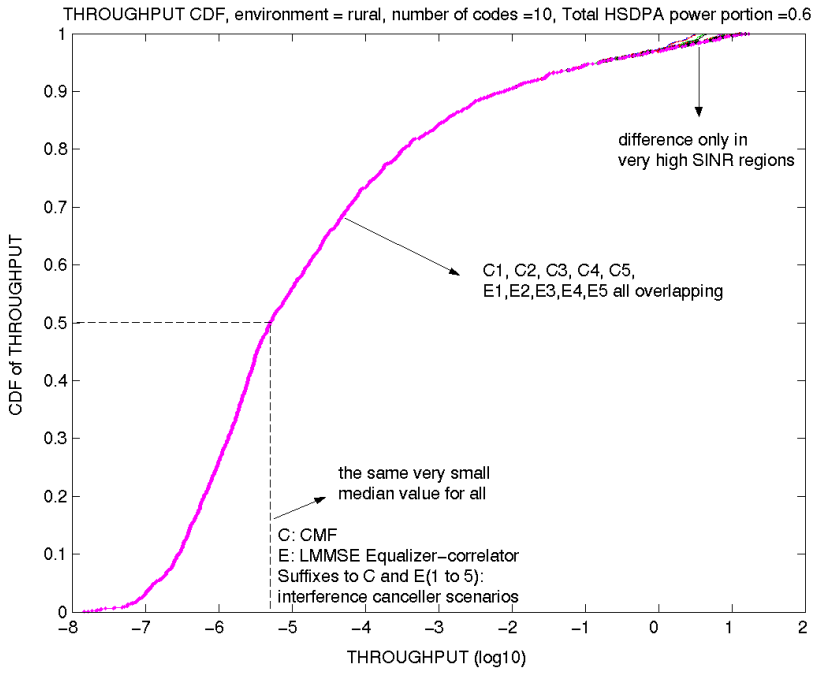


Figure 2.8: Throughput bound CDF of rural cell

Part I

Channel Estimation

Chapter 3

Pilot-Aided Channel Estimation

We consider a family of user dedicated downlink channel estimation methods in WCDMA receivers which are particularly suited for the presence of dedicated channel transmit beamforming and which assume no a priori knowledge of the path delays and the beamforming parameters. They exploit all the transmitted pilot sequences as well as the structured dynamics of the channel. First we build slotwise least squares (LS) estimates of the channels associated with dedicated and common pilots. Then we optimally improve the dedicated channel estimate quality by jointly Kalman filtering the two LS estimates or alternatively (suboptimally) Kalman filtering them separately and combining via weighted LS. In the suboptimal case, the order of Kalman filtering and weighted LS combining results in differing performance and complexity in different conditions. In order to estimate the model parameters we use the expectation maximization (EM) algorithm in the context of the one-lag Kalman smoothing algorithm.

3.1 Introduction

The channel estimation techniques explained in this chapter cover also the *transmit beamforming* situations. Although the channel models are different from the rest of the thesis, the absence of beamforming can be considered as a particular case with identical weights on the beamforming array elements.

Beamforming is a *spatial filtering* technique of applying different complex weights to the signal samples of a number of *correlated* antenna elements with the goal of generating a narrow beam to direct the transmission to an intended user or a groups of users when applied at the transmitter side or suppressing interference originating from other logical

cells or sectors when applied at the receiver side. The upper-bound for the beamforming gain factor in both cases is one less than the number of array elements.

As explained in Section 1.2, each UMTS logical cell has a unique primary scrambling sequence. Therefore, in the downlink, continuously provisioned pilots carried by differently scrambled PCPICH codes enable estimating and differentiating the channels in different logical cells. Possible downlink transmit beamforming techniques in a logical cell can be classified in two groups as *fixed* and *user dedicated* beamforming techniques. Fixed beamforming is simply sub-sectorizing the cell by the allocation of separate secondary scrambling sequences and secondary common pilot channels (SCPICH) which enable estimating and differentiating the *subsector* channels. However this results in a waste of power and code resources. User dedicated beamforming is a means to generate unique beans for each user without any redundancy. However, in such a case, UE cannot utilize the PCPICH as a training sequence for channel estimation purposes since the channels experienced by DPCH and PCPICH are different. Instead, as is shown in Figure 1.7 the user dedicated downlink physical channel (DPCH) in the UMTS FDD downlink consists of a dedicated physical control channel (DPCCH) which carries user dedicated pilots which can be used for channel estimation and which is time multiplexed with the dedicated physical data channel (DPDCH) which carries dedicated data[39]. In this chapter we will be concerned with only the user dedicated beamforming scenarios. However all the explained techniques are extendable or replaceable to also cover the fixed beamforming or nonbeamforming situations.

Most channel estimation techniques proposed for WCDMA receivers are based on either the DPCCH, see e.g. [40, 41] and references therein, or on the PCPICH, see e.g [42]). However, on the one hand, the accuracy of the channel estimation approaches relying only on the DPCCH is limited by the reduced number of dedicated pilots per slot and by the lack of pilots during the DPDCH period that prevents effective tracking of fast fading channels. On the other hand, classical channel estimation approaches based on the PCPICH can better adapt to fast fading conditions, but they are not suited for dedicated channel estimation in the presence of dedicated transmit beamforming. Both approaches remain suboptimal though, due to the fact that they neglect the shared structure by the common and the dedicated propagation channels.

There already exist some works for path-wise dedicated channel estimation which make use of both dedicated and common pilots [43], [44], under the assumption of perfect

a priori knowledge of the path delays. Moreover they implicitly assume the channel associated with the DPCH to be identical to the one associated with the PCPICH. However, as envisaged in the Release 5 of the UMTS standard, this assumption does not hold in the case when beamforming is employed for DPCH transmission. Indeed user dedicated transmit beamforming affects only the DPCH transmission while the PCPICH is evenly broadcasted to all users in the cell. Hence, when dedicated beamforming is present one would be tempted to conclude that PCPICH can no longer be used for dedicated channel estimation, while the dedicated pilots can still be exploited yet with all the previously described limitations. Actually in order to exploit the common pilots as well, the knowledge of the transmit beamforming parameters, i.e. the beamforming weight vector, antenna array responses corresponding to the excited angles and their related statistics should be known at the receiver. Furthermore, even in the absence of transmit beamforming, the offset between the transmit powers assigned to the DPCCH and PCPICH needs to be estimated in order to properly form a combined estimate of the actual dedicated channel.

In general, even in the presence of dedicated beamforming the DPCH and PCPICH associated propagation channels are correlated to a certain extent, as it has been shown by field test measurements. A general dedicated channel estimation technique which optimally combines the channel estimates from common and dedicated pilots via a generic PCPICH-DPCH channel correlation model was introduced in [45]. However, in addition to the correlation between dedicated and common channels, there is also the channel temporal correlation governed by the Doppler spread, which can also be exploited to improve the channel estimation accuracy. To this end, by fitting the channel dynamics to an autoregressive model of sufficient order, Wiener filtering or Kalman filtering can be applied to refine the previously block-wise obtained estimates.

In this section we elaborate on one optimal and two suboptimal spatio-temporal Kalman filtering and Kalman smoothing methods that benefit from all the known sources of information, i.e. the temporal and cross-correlations of common and dedicated pilots. Their performances are quantified via simulations in terms of the dedicated channel estimate normalized mean square error (NMSE).

3.2 Channel Models

We assume the time-varying continuous time channels associated with dedicated and common pilots, $h_d(t,\tau)$ and $h_c(t,\tau)$ respectively, to obey the wide sense stationary uncorrelated scattering (WSS-US) model [18]

$$\begin{aligned} h_d(t,\tau) &= \sum_{i=0}^{\mathcal{P}-1} c_{d,i}(t)p(\tau-\tau_i) \\ h_c(t,\tau) &= \sum_{i=0}^{\mathcal{P}-1} c_{c,i}(t)p(\tau-\tau_i) \end{aligned} \tag{3.1}$$

where $p(\tau)$ represents the pulse-shape filter, $\mathcal{P}$ denotes the number of significant paths, τ_i represents the i-th path delay, $c_{d,i}(t)$ and $c_{c,i}(t)$ are time-varying complex channel coefficients associated with the i-th path of the dedicated and common channel respectively. In many practical circumstances, the two coefficients $c_{d,i}(t)$ and $c_{c,i}(t)$ result to be fairly highly correlated even in the presence of dedicated downlink beamforming. Notice that in (3.1) the coefficients $c_{d,i}(t)$ for $i = 0, ...\mathcal{P}-1$ account for the complete cascade of the beamforming weight vector, the antenna array response on the excited angles, as well as for the actual propagation channel between the transmitter and the receiver. The receiver is assumed to sample m times per chip period the low-pass filtered received baseband signal. Stacking the m samples per chip period in vectors, the discrete time finite impulse response (FIR) representation of both common and dedicated channels at chip rate takes the form $\boldsymbol{h}_l = [h_{1,l} \dots h_{m,l}]^T$, which represents the vector of the samples of the overall channel, including the pulse shape, the propagation channel, the anti-aliasing receiver filter and, when applicable, the beamforming weighting. The superscript $(\cdot)^T$ denotes the transpose operator. Assuming the overall channel to have a delay spread of N chip periods, the dedicated and common channel impulse responses take the form $\boldsymbol{h}(n) = \boldsymbol{\Psi}\boldsymbol{c}(n)$ where $\boldsymbol{h} = [\boldsymbol{h}_1^T, \dots, \boldsymbol{h}_N^T]^T \in \mathcal{C}^{mN\times 1}$, $\boldsymbol{c}(n) = [c_1(n) \dots c_{\mathcal{P}}(n)]^T \in \mathcal{C}^{\mathcal{P}\times 1}$ are the complex path amplitudes and the temporal index n relates to the time instant at which the time-varying channel is observed. The assumption of fixed delays τ_i's over the observation window, yields to a constant pulse-shape convolution matrix $\boldsymbol{\Psi} \in \mathcal{R}^{mN\times\mathcal{P}}$ given by

$$\boldsymbol{\Psi} = \boldsymbol{\Psi}\left(\tau_1, \cdots, \tau_{\mathcal{P}}\right) = [\boldsymbol{p}(\tau_1), \dots, \boldsymbol{p}(\tau_{\mathcal{P}})]$$

where $\boldsymbol{p}(\tau_i)$ represents the sampled version of the pulse shape filter impulse response delayed by τ_i. The complex path amplitudes variations are modeled as an autoregressive (AR)

processes of order sufficiently high to characterize the Doppler spectrum. Matching only the channel bandwidth with the Doppler spread leads to a first-order AR(1) model of the form

$$\boldsymbol{c}(n) = \rho\,\boldsymbol{c}(n-1) + \sqrt{1-|\rho|^2}\Delta\boldsymbol{c}(n) = \frac{\sqrt{1-|\rho|^2}}{1-\rho q^{-1}}\Delta\boldsymbol{c}(n)$$

so that, $\boldsymbol{\Psi}$ being constant over the observation time interval, we obtain

$$\boldsymbol{h}(n) = \rho\boldsymbol{h}(n-1) + \sqrt{1-|\rho|^2}\Delta\boldsymbol{h}(n) = \frac{\sqrt{1-|\rho|^2}}{1-\rho q^{-1}}\Delta\boldsymbol{h}(n) \tag{3.2}$$

where q^{-1} denotes the delay operator such that $q^{-1}y(n) = y(n-1)$ and ρ represents the AR process temporal coherence correlation coefficient. Since the Doppler spread is assumed to be the same for both channels (3.1), the model (3.2) applies to both $\boldsymbol{h}_d(n)$ and $\boldsymbol{h}_c(n)$. The variance of k-th component $h_{c,k}(n)$ of $\boldsymbol{h}_c(n)$ is $\sigma^2_{h_{c,k}} = \sigma^2_{\Delta h_{c,k}} = \boldsymbol{p}_k\boldsymbol{D}_c\boldsymbol{p}_k^H$ where $\boldsymbol{p}_k$ denotes the k-th line of $\boldsymbol{\Psi}$ and $\boldsymbol{D}_c = \text{diag}\,(\sigma^2_{\Delta c_1}, \ldots, \sigma^2_{\Delta c_{c,\mathcal{P}}})$. Notice that $\sigma^2_{c_{c,i}} = \sigma^2_{\Delta c_{c,i}}$. Similarly the variance of k-th component $h_{d,k}(n)$ of $\boldsymbol{h}_d(n)$, is $\sigma^2_{h_{d,k}} = \sigma^2_{\Delta h_{d,k}} = \boldsymbol{p}_k\boldsymbol{D}_d\boldsymbol{p}_k^H$ where $\boldsymbol{D}_d = \text{diag}\,(\sigma^2_{\Delta c_{d,1}}, \ldots, \sigma^2_{\Delta c_{d,\mathcal{P}}})$.

3.3 LS Estimations of Common and Dedicated Channels

All the three proposed approaches start with block-wise dedicated and common channel least squares (LS) estimates $\hat{\boldsymbol{h}}_c(n)$ and $\hat{\boldsymbol{h}}_d(n)$ which are computed based on the a priori knowledge of the common and dedicated pilot chips. For the sake of simplicity, without loss of generality, in this section, we assume that block-wise corresponds to slot-wise estimates. We assume that dedicated pilot chips are sent in every slot. Let $\boldsymbol{S}_d(n) = \mathsf{S}_d(n) \otimes \mathbf{I}_m$, where $\otimes$ denotes the Kronecker product, represent the block Hankel matrix comprising the dedicated pilot chip sequence intended for the user of interest in slot n. Similarly we refer to $\boldsymbol{S}_c(n) = \mathsf{S}_c(n) \otimes \mathbf{I}_m$ as the block Hankel matrix containing the common pilot chip sequence in slot n. Let $\boldsymbol{Y}(n)$ be the received signal samples vector corresponding to slot n. The LS unstructured FIR common and dedicated channel estimates FIR are given by

$$\begin{aligned}\hat{\boldsymbol{h}}_d(n) &= \arg\min_{\boldsymbol{h}_d} \|\boldsymbol{Y}(n) - \boldsymbol{S}_d(n)\boldsymbol{h}_d(n)\|^2 \\ \hat{\boldsymbol{h}}_c(n) &= \arg\min_{\boldsymbol{h}_c} \|\boldsymbol{Y}(n) - \boldsymbol{S}_c(n)\boldsymbol{h}_c(n)\|^2\end{aligned} \tag{3.3}$$

The exact LS solutions of problems (3.3) are readily given by

$$\begin{aligned}\hat{\boldsymbol{h}}_d(n) &= (\boldsymbol{S}_d^H(n)\boldsymbol{S}_d(n))^{-1}\boldsymbol{S}_d^H(n)\boldsymbol{Y}(n)\\ \hat{\boldsymbol{h}}_c(n) &= (\boldsymbol{S}_c^H(n)\boldsymbol{S}_c(n))^{-1}\boldsymbol{S}_c^H(n)\boldsymbol{Y}(n)\end{aligned} \tag{3.4}$$

where $(\cdot)^H$ denotes Hermitian transpose. Note that the equations (3.4) reduce to

$$\hat{\boldsymbol{h}}_d(n) \approx \beta_d^{-1}\boldsymbol{S}_d^H(n)\boldsymbol{Y}(n); \quad \hat{\boldsymbol{h}}_c(n) \approx \beta_c^{-1}\boldsymbol{S}_c^H(n)\boldsymbol{Y}(n)$$

if the pilot chips can be modeled as i.i.d. random variables, where β_d and β_c represent the dedicated and common pilot chip sequences total energies respectively. We can estimate $\sigma^2_{e_{d,k}}$ and $\sigma^2_{e_{c,k}}$ from $\hat{h}_{d,k}$ and $\hat{h}_{c,k}$ at delays k where we expect the channel not to carry any energy. That can be achieved by, e.g., overestimating the channel delay spread, and using the tails of the channel estimates to obtain unbiased estimates $\sigma^2_{e_{d,k}}$ and $\sigma^2_{e_{c,k}}$.

3.4 Optimal Recursive Approach: Joint Kalman Filtering and Smoothing

Channel (State Vector) Dynamics

$\boldsymbol{h}(n) = \begin{bmatrix} h_d(n) \\ h_c(n) \end{bmatrix}$: Present State Vector
$\boldsymbol{h}(n+1) = \rho\boldsymbol{h}(n) + \boldsymbol{B}\boldsymbol{u}(n)$: State Transition Process
$\boldsymbol{B} = \sqrt{1-|\rho^2|}\begin{bmatrix} 1 & 0 \\ \alpha & \sqrt{1-\frac{\sigma^2_{h_d}}{\sigma^2_{h_c}}|\alpha^2|} \end{bmatrix}$: Input Gain
$\boldsymbol{u}(n) = \begin{bmatrix} \Delta h_d(n) \\ \Delta h_c(n) \end{bmatrix}$: Input Vector
α: PCPICH-DPCH correlation coefficient
$\boldsymbol{R_{uu}} = \begin{bmatrix} \sigma^2_{\Delta h_d} & 0 \\ 0 & \sigma^2_{\Delta h_c} \end{bmatrix}$: Input Covariance
$\boldsymbol{B}\boldsymbol{u}(n)$: Process Noise
$\boldsymbol{Q} = \boldsymbol{B}\boldsymbol{R_{uu}}\boldsymbol{B}^H$: Process Noise Covariance

First Step LS Estimation (State Measurement)

$\hat{\boldsymbol{h}}(n) = \begin{bmatrix} \hat{h}_d(n) \\ \hat{h}_c(n) \end{bmatrix} = \boldsymbol{h}(n) + \boldsymbol{w}(n)$: Measurement (*LS estimates*)
$\boldsymbol{w}(n) = \begin{bmatrix} e_d(n) \\ e_c(n) \end{bmatrix}$: Measurement Noise
$\boldsymbol{R_{ww}} = \begin{bmatrix} \sigma^2_{e_d} & 0 \\ 0 & \sigma^2_{e_c} \end{bmatrix}$: Measurement Noise Covariance

Algorithm Initialization

$\gamma(0) = 0$: Moving Averaging Weight
$\lambda = 0.95$: Forgetting Factor
$\hat{\rho}(0) = 0.999$: Temporal Correlation Coefficient Estimate
$\hat{\hat{\boldsymbol{h}}}(0 \mid 0) = \hat{\boldsymbol{h}}(0 \mid 0)$: Initial State Estimate
$\boldsymbol{S}(0 \mid 0) = \boldsymbol{R_{ww}}$: Initial State Error Covariance
$\boldsymbol{S}(1 \mid 0) =\mid \hat{\rho}(0) \mid^2 \boldsymbol{R_{ww}}$: Initial Prediction Error Covariance
$\hat{\boldsymbol{Q}}(0) = \mathbf{0}_{2\times 2}$: Process Noise Covariance Estimate
$\boldsymbol{\Xi}_1 = \boldsymbol{\Xi}_2 = \boldsymbol{\Xi}_{12} = \mathbf{0}_{2\times 2}$: Supporting Adaptation Parameters

Kalman Filtering and Smoothing (E-Step)

$\hat{\hat{\boldsymbol{h}}}(n+1 \mid n) = \hat{\rho}\hat{\hat{\boldsymbol{h}}}(n \mid n)$: Time Update
$\boldsymbol{G}(n+1) = \boldsymbol{S}(n+1 \mid n)\left[\boldsymbol{S}(n+1 \mid n) + \boldsymbol{R_{ww}}\right]^{-1}$: Filter Gain
$\hat{\hat{\boldsymbol{h}}}(n+1 \mid n+1) = \left[\boldsymbol{I}_{2\times 2} - \boldsymbol{G}(n+1)\right]\hat{\hat{\boldsymbol{h}}}(n+1 \mid n) + \boldsymbol{G}(n+1)\hat{\boldsymbol{h}}(n)$:
Measurement Update
$\boldsymbol{S}(n+1 \mid n+1) = \left[\boldsymbol{I}_{2\times 2} - \boldsymbol{G}(n+1)\right]\boldsymbol{S}(n+1 \mid n)$:
Filtered State Covariance
$\boldsymbol{A}(n) = \hat{\rho}^H \boldsymbol{S}(n \mid n)\boldsymbol{S}(n+1 \mid n)^{-1}$: Smoothing Gain
$\hat{\hat{\boldsymbol{h}}}(n \mid n+1) = \hat{\hat{\boldsymbol{h}}}(n \mid n) + \boldsymbol{A}(n)\left(\hat{\hat{\boldsymbol{h}}}(n+1 \mid n+1) - \hat{\hat{\boldsymbol{h}}}(n+1 \mid n)\right)$:
Smoothing Update
$\boldsymbol{S}(n \mid n+1) = \boldsymbol{S}(n \mid n) + \boldsymbol{A}(n)\left(\boldsymbol{S}(n+1 \mid n+1) - \boldsymbol{S}(n+1 \mid n)\right)\boldsymbol{A}(n)^H$:
Smoothed State Error Covariance

Adaptive Estimation of Model Parameters (M-Step)

$\boldsymbol{\Xi}_1 = \lambda\boldsymbol{\Xi}_1 + \hat{\hat{\boldsymbol{h}}}(n+1 \mid n+1)\hat{\hat{\boldsymbol{h}}}(n+1 \mid n+1)^H + \boldsymbol{S}(n+1 \mid n+1)$
$\boldsymbol{\Xi}_2 = \lambda\boldsymbol{\Xi}_2 + \hat{\hat{\boldsymbol{h}}}(n \mid n+1)\hat{\hat{\boldsymbol{h}}}(n \mid n+1)^H + \boldsymbol{S}(n \mid n+1)$
$\boldsymbol{\Xi}_{12} = \lambda\boldsymbol{\Xi}_{12} + \hat{\hat{\boldsymbol{h}}}(n+1 \mid n+1)\hat{\hat{\boldsymbol{h}}}(n \mid n+1)^H + \boldsymbol{S}(n+1 \mid n+1)^H\boldsymbol{A}(n)^H$
$\gamma(n+1) = \lambda\gamma(n) + 1$
$\hat{\rho}(n+1) = Trace\{\boldsymbol{\Xi}_{12}\boldsymbol{\Xi}_2^{-1}\}/2$
$\hat{\boldsymbol{Q}}(n+1) = \frac{1}{\gamma(n+1)}\left(\boldsymbol{\Xi}_1 - \boldsymbol{\Xi}_{12}\boldsymbol{\Xi}_2^{-1}\boldsymbol{\Xi}_{12}^H\right)$
$\boldsymbol{S}(n+2 \mid n+1) = \hat{\rho}(n+1)\boldsymbol{S}(n+1 \mid n+1)\hat{\rho}(n+1)^H + \hat{\boldsymbol{Q}}(n+1)$:
Prediction Error Covariance

Steady State Performance

$\boldsymbol{S}(\infty \mid \infty) = \boldsymbol{R}_{w,w}\left[\mid \hat{\rho}(\infty) \mid^2 \boldsymbol{S}(\infty \mid \infty) + \hat{\boldsymbol{Q}}(\boldsymbol{\infty}) + \boldsymbol{R}_{w,w}\right]^{-1} \times$
$\left[\mid \hat{\rho}(\infty) \mid^2 \boldsymbol{S}(\infty \mid \infty) + \hat{\boldsymbol{Q}}(\boldsymbol{\infty})\right]$: Steady State Error Variance
$\boldsymbol{S}(\infty + 1 \mid \infty) =\mid \hat{\rho}(\infty) \mid^2 \boldsymbol{S}(\infty \mid \infty) + \hat{\boldsymbol{Q}}(\boldsymbol{\infty})$:
Steady State Prediction Error Variance
$\boldsymbol{S}(\infty \mid \infty + 1) = \boldsymbol{S}(\infty \mid \infty) + \mid \hat{\rho}(\infty) \mid^2 \boldsymbol{S}(\infty \mid \infty)\boldsymbol{S}(\infty + 1 \mid \infty)^{-1}\left[\boldsymbol{S}(\infty \mid \infty) - \boldsymbol{S}(\infty + 1 \mid \infty)\right]\boldsymbol{S}(\infty + 1 \mid \infty)^{-H}\boldsymbol{S}(\infty \mid \infty)^H$:
Smoothed Steady State Error Variance

Above is given the optimal EM-Kalman filtering and smoothing algorithm corresponding to the scheme shown in Figure 3.1.b1 which we apply *independently* for each channel tap, the tap indices dropped for simplicity. It has been used in several other different contexts in order to estimate the states and the unknown model parameters of dynamic systems [46, 47, 48, 49, 50]. On its core lies the Expectation Maximization (EM) algorithm which is the most referred method when the problem in hand is suffering from incomplete data and the solution requires both completing this data and estimating some parameters [51]. The algorithm iterates between the E-phase which is the expected log-likelihood computation of the missing (imputed) data by using both the observed data and the present parameter estimates and the M-phase which computes the maximum likelihood (ML) value of the parameters by conditioning on the imputed data as if it were the correct data. For this section $\boldsymbol{h}(n) = [h_d(n)\ h_c(n)]^T$ channel parameters are the missing data and $\{\hat{\rho}, \hat{\boldsymbol{Q}}\}$ are the only needed parameter estimates. Fitting the EM mechanism to Kalman filtering context requires also smoothing in the E-phase. In all the mentioned papers fixed-interval smoothing mechanism is used which is very complex and large buffer sizes are required, except for [50] where single delay fixed-lag smoothing is considered. In this section we follow the latter strategy due to its suitability for implementation. We slightly modify the M-phase by taking the average of the two $\hat{\rho}$ estimates (diagonal components of $\boldsymbol{\Xi}_{12}\boldsymbol{\Xi}_2^{-1}$) via the *Trace* operation, considering the fact that the temporal correlation coefficients of dedicated and common taps are equal.

3.5 Suboptimal Scheme 1: EM-Kalman Procedure After UL-MMSE Combining

This scheme corresponds to Figure 3.1.b2, it runs independently for each channel tap and it has two phases as explained in the sequel.

3.5.1 Unbiased LMMSE Combining of LS Estimates

Let $\hat{\boldsymbol{h}}_k(n) = [\hat{h}_{d,k}(n)\ \hat{h}_{c,k}(n)]^T$ denote the vector of the LS estimates of the k-th elements of the dedicated and common pilot channel FIR responses at slot n, i.e.,

$$\hat{\boldsymbol{h}}_k(n) = \begin{bmatrix} \hat{h}_{d,k}(n) \\ \hat{h}_{c,k}(n) \end{bmatrix} = \begin{bmatrix} h_{d,k}(n) \\ h_{c,k}(n) \end{bmatrix} + \begin{bmatrix} e_{d,k}(n) \\ e_{c,k}(n) \end{bmatrix}. \tag{3.5}$$

In order for our derivation to be fully general, we introduce the following dedicated and common channel correlation model

$$h_{c,k}(n) = \alpha_k h_{d,k}(n) + x_{c,k}(n) \tag{3.6}$$

where $\alpha_k h_{d,k}(n)$ represents the short-term ULMMSE estimate of $h_{c,k}(n)$ on the basis of $h_{d,k}(n)$, and $x_{c,k}(n)$ represents the associated estimation error. Then, a refined estimate can be obtained as $\overline{h}_{d,k}(n) = \boldsymbol{f}_k \hat{\boldsymbol{h}}_k(n)$ by optimal combining of common and dedicated LS channel estimates. In order not to introduce bias for the processing in the next estimation step, we shall determine $\boldsymbol{f}$ as the ULMMSE filter, i.e. by solving for all k's the optimization problem

$$\min_{\boldsymbol{f}_k} \mathrm{E}|h_{d,k}(n) - \boldsymbol{f}_k \hat{\boldsymbol{h}}_k(n)|^2 \quad \text{s.t. } \boldsymbol{f}_k [1\ \alpha_k]^T = 1$$

The optimal ULMMSE filter $\boldsymbol{f}_k$ is obtained as

$$\begin{aligned} \boldsymbol{f}_{k,\mathrm{ULMMSE}} &= ([1\ \alpha_k^*] \boldsymbol{R}^{-1}_{\hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k} [1\ \alpha_k]^T)^{-1} [1\ \alpha_k^*] \boldsymbol{R}^{-1}_{\hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k} \\ &= ([1\ \alpha_k^*] \boldsymbol{R}^{-1} [1\ \alpha_k]^T)^{-1} [1\ \alpha_k^*] \boldsymbol{R}^{-1} \end{aligned}$$

where $\boldsymbol{R}_{\hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k} = \mathrm{E}\hat{\boldsymbol{h}}_k(n) \hat{\boldsymbol{h}}_k^H(n)$, $\boldsymbol{R} = \mathrm{diag}\,(\sigma^2_{e_{d,k}},\ (\sigma^2_{e_{c,k}} + \sigma^2_{x_{c,k}}))$, with $\sigma^2_{x_{c,k}} = \mathrm{E}|\hat{x}_{c,k}(n)|^2$. Notice that the covariance matrix $\boldsymbol{R}_{\hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k}$ is equal to

$$\boldsymbol{R}_{\hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k} = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} = \sigma^2_{h_{d,k}} \begin{bmatrix} 1 \\ \alpha_k \end{bmatrix} \begin{bmatrix} 1 \\ \alpha_k \end{bmatrix}^H + \begin{bmatrix} \sigma^2_{e_{d,k}} & 0 \\ 0 & \sigma^2_{e_{c,k}} + \sigma^2_{x_{c,k}} \end{bmatrix}$$

Having an estimate of the matrix $\boldsymbol{R}_{\hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k}$, e.g. by temporal averaging, we can apply the *covariance matching* criterion so that $\sigma^2_{h_{d,k}} = r_{11} - \sigma^2_{e_{d,k}}$, $\alpha_k = r_{21}/(r_{11} - \sigma^2_{e_{d,k}})$, (i.e. α_k has the same phase as r_{21}), where the following bound $|\alpha_k| \leq \sigma_{h_{c,k}}/\sigma_{h_{d,k}} = \sqrt{(r_{22} - \sigma^2_{e_{c,k}})/(r_{11} - \sigma^2_{e_{d,k}})}$ can be used in actual estimation. Furthermore, since $\sigma^2_{x_{c,k}} = r_{22} - \sigma^2_{e_{c,k}} - |r_{21}|^2/(r_{11} - \sigma^2_{e_{d,k}})$.

Finally, the variance of the estimation error $\hat{\hat{e}}_{d,k}(n)$ after ULMMSE combining is obtained as

$$\overline{\sigma}^2_{e_{d,k}} = \frac{\sigma^2_{e_{d,k}} (\sigma^2_{e_{c,k}} + \sigma^2_{x_{c,k}})}{\sigma^2_{e_{d,k}} |\alpha_k|^2 + \sigma^2_{e_{c,k}} + \sigma^2_{x_{c,k}}} \tag{3.7}$$

The dedicated channel estimate after ULMMSE combining, $\overline{h}_{d,k}(n) = h_{d,k}(n) + \overline{\sigma}^2_{e_{d,k}}$, is such that the post-combining estimation error $\overline{\sigma}^2_{e_{d,k}}$ is mutually uncorrelated with $h_{d,k}(n)$, $\overline{\sigma}^2_{e_{d,k}}$

and $\overline{\sigma}^2_{e_{d,j}}$ are mutually uncorrelated for any $k \neq j$, and the variance $\overline{\sigma}^2_{e_{d,k}}$ is independent of k while it depends on the Doppler spread, on the channel power and on the SINR.

3.5.2 Kalman Filtering of ULMMSE Combined Estimates

Once the ULMMSE combined dedicated channel estimates are obtained, we apply the causal Kalman filtering and smoothing to obtain the final estimates $\overline{\overline{h}}_{d,k}(n \mid n)$ and $\overline{\overline{h}}_{d,k}(n \mid n+1)$. This algorithm is similar to the one in Section 3.4 with the single difference that the state vector has now only one element, which is the *complexity advantage* w.r.t. the optimal scheme that has a state vector of two elements. This scheme is identical to the optimal one when the normalized correlation factor $\mid \zeta_k \mid = |\alpha_k|\sigma_{h_{d,k}}/\sigma_{h_{c,k}} \leq 1$ is unity, i.e. $\zeta_k = 1,\ \forall k$.

3.6 Suboptimal Scheme 2: ULMMSE Combining After Two Separate EM-Kalman Procedures

In this case we change the order of EM-Kalman procedure and ULMMSE combining as shown in Figure 3.1.b3. The EM-Kalman filtering outputs dedicated and common channel parameter estimates $\tilde{h}_{d,k}(n \mid n)$ and $\tilde{h}_{c,k}(n \mid n)$ and their associated error variances $\tilde{\sigma}^2_{e_{d,k}}(n \mid n)$ and $\tilde{\sigma}^2_{e_{c,k}}(n \mid n)$ are fed to the following ULMMSE block to obtain the final estimate $\tilde{\tilde{h}}_{d,k}(n \mid n)$. Similar procedure is applied to obtain the refined, smoothed estimate $\tilde{\tilde{h}}_{d,k}(n \mid n+1)$ from the EM-Kalman smoothing output parameters $\tilde{h}_{d,k}(n \mid n+1)$, $\tilde{h}_{c,k}(n \mid n+1)$, $\tilde{\sigma}^2_{e_{d,k}}(n \mid n+1)$ and $\tilde{\sigma}^2_{e_{c,k}}(n \mid n+1)$. Other necessary parameters are estimated similar to what is done in Section 3.5.1. This two stage procedure is suboptimal (due to the coloring of noises at EM-Kalman outputs) unless $\rho = 1$ and it is not as attractive for implementation as the first suboptimal scheme since the Kalman state vector has two elements as in the optimal case.

3.7 Simulations and Conclusions

The performances of the presented channel estimation methods in the presence of dedicated transmit beamforming are presented in Figure 3.2 to Figure 3.9 in terms of the channel estimate NMSE. We assume the DPCCH to occupy 20% of the UMTS slot,

and the DPCH spreading factor to be equal to 128. We define the normalized correlation factor $\mid \zeta_k \mid = |\alpha_k|\sigma_{h_{d,k}}/\sigma_{h_{c,k}} \leq 1$. Being interested in the impact of dedicated and common channel correlation we set, for the sake of simplicity, $|\alpha_k| = |\alpha_0|$ constant $\forall k$. We initially assume the DPCCH and the PCPICH to be respectively assigned 5% and 10 % of the base station transmitted power. We also assume an additional DPCH beamforming gain of 6 dB, yielding to a power offset between DPCCH and PCPICH equal to $\sigma^2_{h_{c,k}}/\sigma^2_{h_{d,k}} = 0.5$ for all k's, so that $\zeta = \zeta_k = \sqrt{2}|\alpha_0|$. Channels are randomly generated from the power delay profile of the *UMTS Vehicular A* channel [39]. Temporal correlation coeffcients $\rho = 0.99$ and $\rho = 0.9$ correspond in Jakes model to vehicle speeds 29km/h and 92km/h for a transmission at 1.8GHz. The legends {Dedicated LS, Kalman Filtering of Dedicated LS, Kalman Smoothing of Dedicated LS, ULMMSE Combining of Dedicated LS, Kalman Filtering After ULMMSE, Kalman Smoothing After ULMMSE, ULMMSE After Kalman Filtering, ULMMSE After Kalman Smoothing, Optimal Kalman Filtering, Optimal Kalman Smoothing} on the figures corresponds in the same order to the NMSE performances of the channel estimates $\{\hat{\boldsymbol{h}}_d(n), \tilde{\boldsymbol{h}}_d(n \mid n), \tilde{\boldsymbol{h}}_d(n \mid n+1), \overline{\boldsymbol{h}}_d(n), \overline{\overline{\boldsymbol{h}}}_d(n \mid n), \overline{\overline{\boldsymbol{h}}}_d(n \mid n+1), \tilde{\tilde{\boldsymbol{h}}}_d(n \mid n), \tilde{\tilde{\boldsymbol{h}}}_d(n \mid n+1), \hat{\hat{\boldsymbol{h}}}_d(n \mid n), \hat{\hat{\boldsymbol{h}}}_d(n \mid n+1)\}$ at the steady states of the EM-Kalman procedures.

Some interpretations of figures are as follows:

- $\overline{\boldsymbol{h}}_d(n)$ brings moderate improvement w.r.t. $\hat{\boldsymbol{h}}_d(n)$ at reasonably high cross correlations.
- $\tilde{\boldsymbol{h}}_d(n \mid n)$ performs much better than $\overline{\boldsymbol{h}}_d(n)$.
- $\hat{\hat{\boldsymbol{h}}}_d(n \mid n)$ is the best (optimal causal filter in MMSE sense), performs also as close as 0.2dB w.r.t. knowing the ρ and $\boldsymbol{Q}$ parameters (latter case not shown on the plots), but it is at the same time the most complex.
- $\overline{\overline{\boldsymbol{h}}}_d(n \mid n)$ is equivalent to $\hat{\hat{\boldsymbol{h}}}_d(n \mid n)$ when the channels associated with DPCH and PCPICH are fully correlated. Their performance difference is non-negligible only when the Doppler spread and DPCH-PCPICH correlations are both low. It is attractive also for the non-beamforming case, especially in order to increase the coverage since in that case DPCH power can become comparable to or even exceed the PCPICH power at cell edges due to power control.
- $\tilde{\tilde{\boldsymbol{h}}}_d(n \mid n)$ performs better than the $\overline{\overline{\boldsymbol{h}}}_d(n \mid n)$ when DPCH and PCPICH are not very much correlated.

- smoothing (backward pass) improves the performance w.r.t. filtering (only forward pass) in all the cases.

All the three methods are feasible for implementation since complexity is proportional to the number of channel taps and Kalman state vectors for each tap have at most two elements.

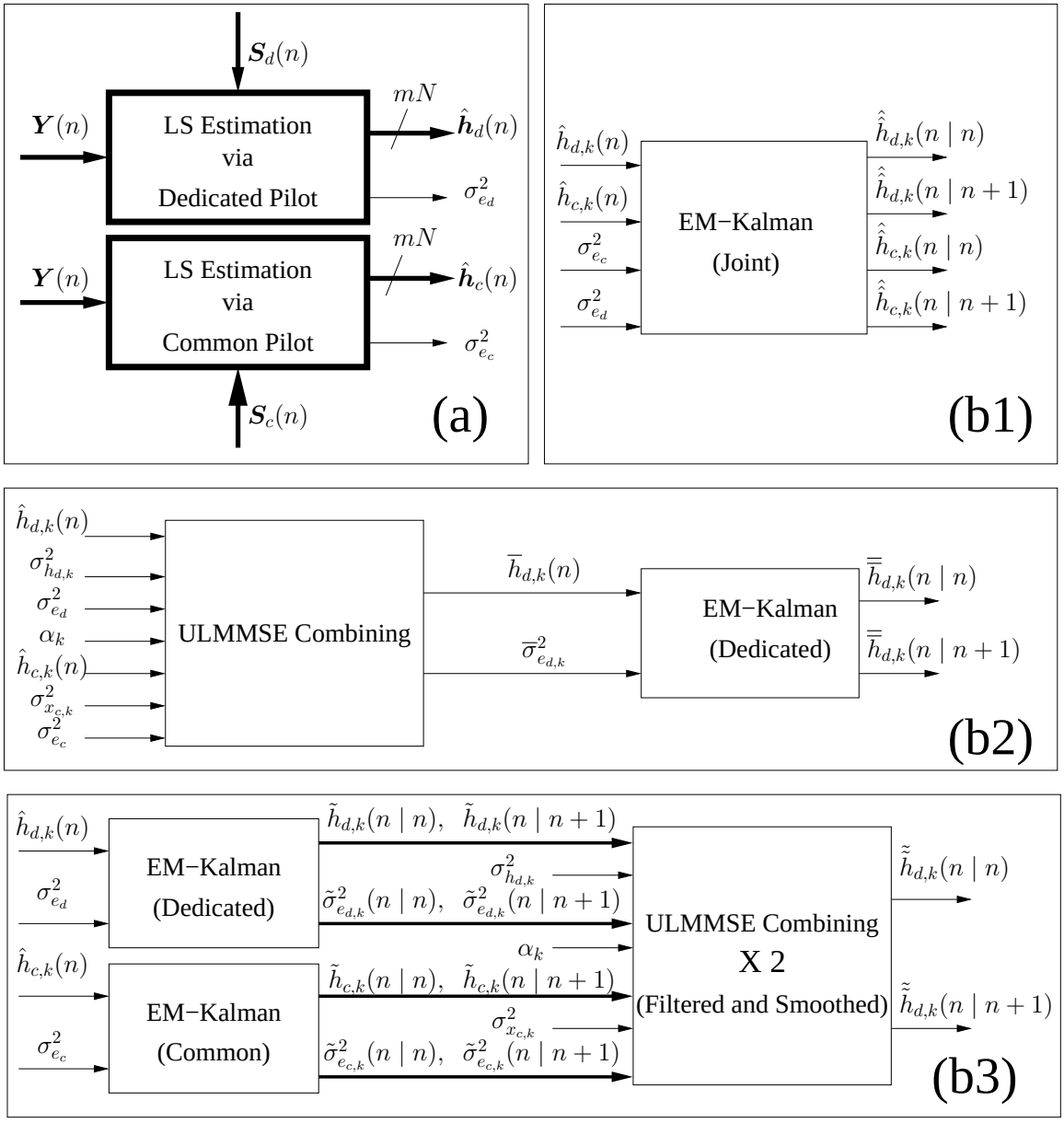


Figure 3.1: a: joint LS estimation of all channel taps), {b1,b2,b3}: {optimal scheme, suboptimal scheme 1, suboptimal scheme 2} for each tap

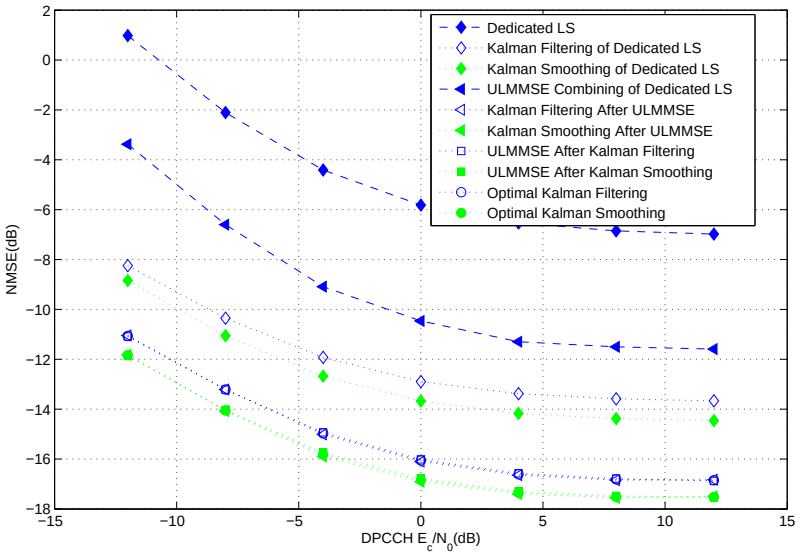


Figure 3.2: NMSE vs DPCCH E_c/N_0, $\zeta = 1$, $\rho = 0.99$

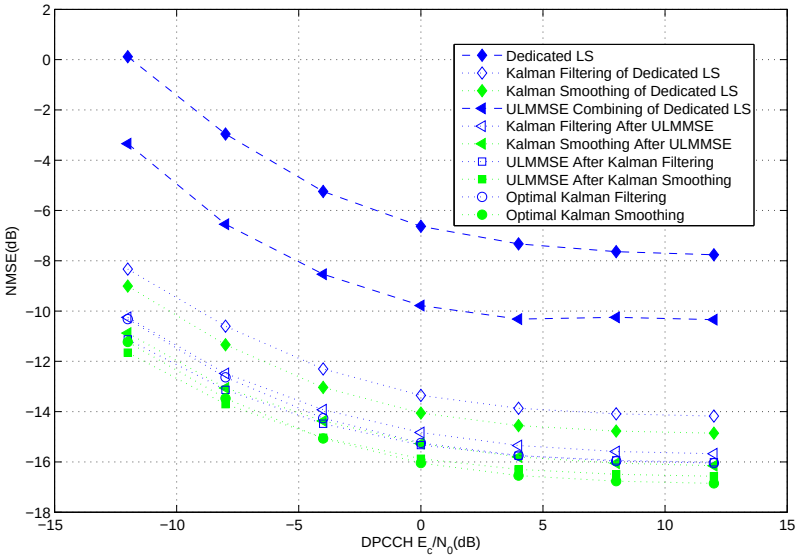


Figure 3.3: NMSE vs DPCCH E_c/N_0, $\zeta = 0.95$, $\rho = 0.9$

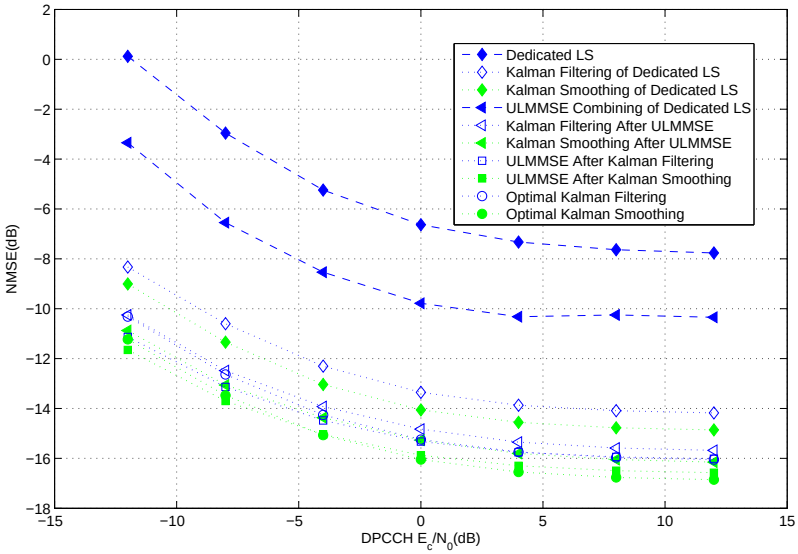


Figure 3.4: NMSE vs DPCCH E_c/N_0, $\zeta = 0.95$, $\rho = 0.99$

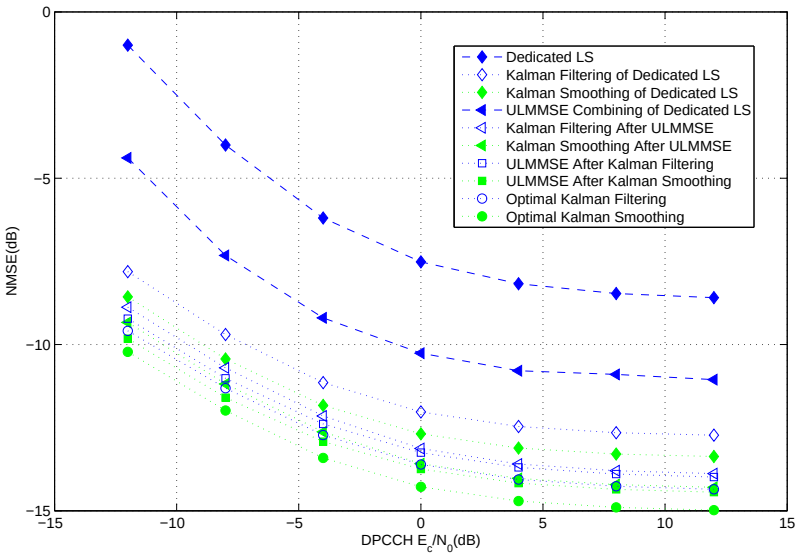


Figure 3.5: NMSE vs DPCCH E_c/N_0, $\zeta = 0.9$, $\rho = 0.9$

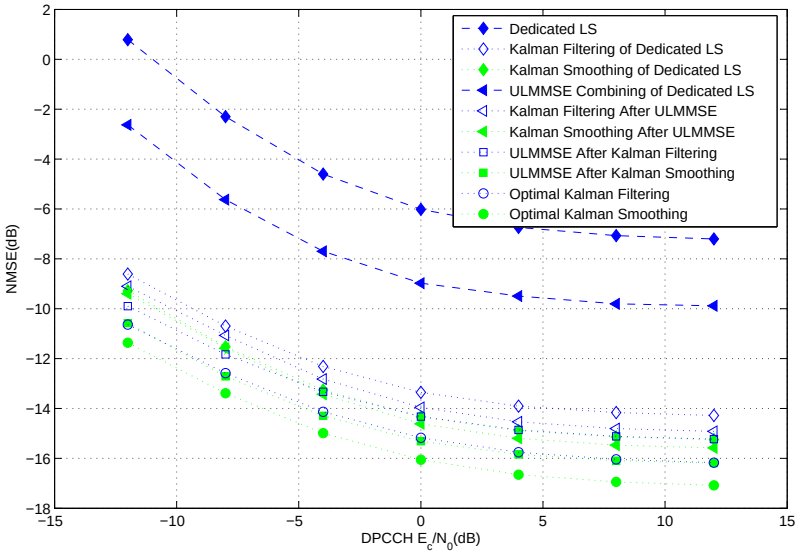


Figure 3.6: NMSE vs DPCCH E_c/N_0, $\zeta = 0.9$, $\rho = 0.99$

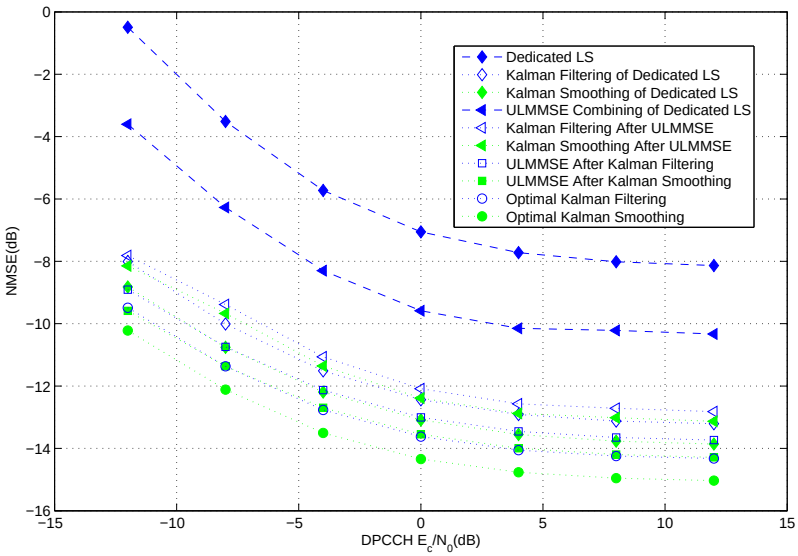


Figure 3.7: NMSE vs DPCCH E_c/N_0, $\zeta = 0.8$, $\rho = 0.9$

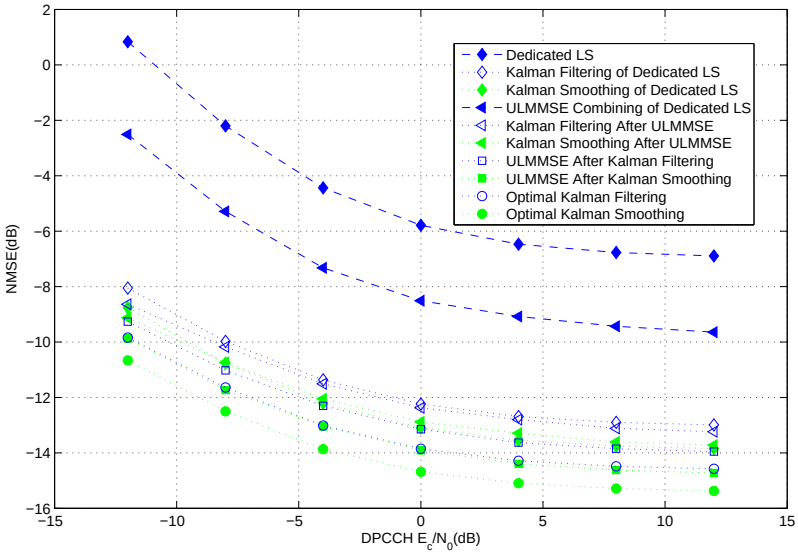


Figure 3.8: NMSE vs DPCCH E_c/N_0, $\zeta = 0.8$, $\rho = 0.99$

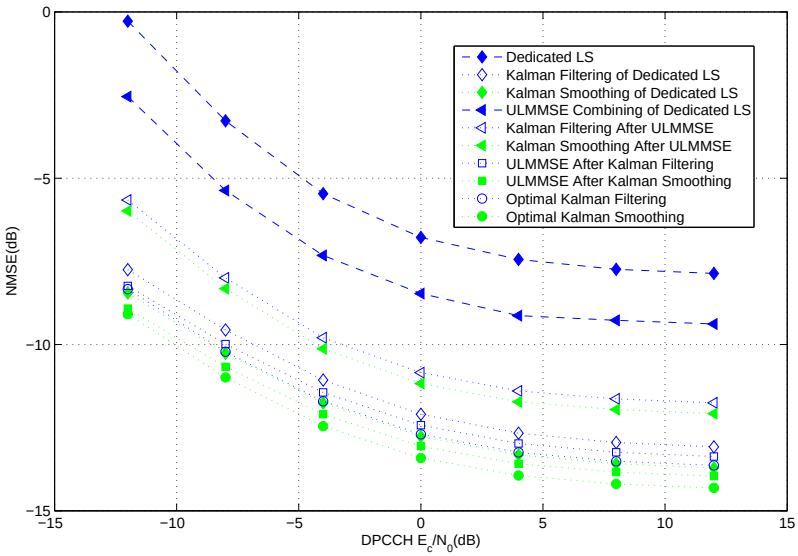


Figure 3.9: NMSE vs DPCCH E_c/N_0, $\zeta = 0.6$, $\rho = 0.9$

Part II

Chip Equalization

Chapter 4

Chip Level Adaptive Equalization for HSDPA

We consider a chip level decision-directed NLMS equalization scheme which targets estimating the total transmitted base station chip sequence in a decision-directed manner and using it as the desired response for equalizer adaptation. For this purpose, we explicitly use only the knowledge of the user-assigned HSPDSCH codes in order to obtain reliable signal components by hard decisions. By exploiting the equivalence between the actual multirate transmission in the sense of containing multiple spreading factors and the multicode pseudo-transmission at the single HSDPA spreading level we use also the estimated pseudo-symbols of other codes via LMMSE weightings. In addition to its reasonable complexity and Max-SINR achieving performance in realistic HSDPA working regimes, the proposed scheme also has the advantage of not requiring the channel parameters. We evaluate its performance by extensive simulations vis-à-vis the Griffiths equalizer which requires channel parameters.

4.1 Introduction

There are several ways to implement the Max-SINR equalizer.

One group of parametric methods target reliably estimating all the $\boldsymbol{h}$, σ_v^2 and σ_b^2 ingredients once every predetermined time period and calculating the $\boldsymbol{f}_o$ filter[1] from (1.31). The update time period depends on the rate of change of the channel, i.e. on the Doppler frequency, and depends on the birth and the death rate of the users. These methods have the advantage of precisely modeling the BS signal component $\sigma_b^2 \boldsymbol{\mathcal{T}}(\boldsymbol{h})\boldsymbol{\mathcal{T}}(\boldsymbol{h})^H$ in $\boldsymbol{R}_{yy}$. However they ignore the color of other-cells' interference by modeling it as white noise [16, 52, 53, 54].

Another group of semi-parametric methods might aim to avoid this drawback

[1]Throughout text $\boldsymbol{f}$ refers to $\boldsymbol{f}_o$ or $\tilde{\boldsymbol{f}}_o$ depending on the context

by calculating the $\boldsymbol{R}_{yy}$ statistics directly from the received data, see for example such a technique in a different receiver context in [55]. Although at first sight it looks attractive, the short term sample support is not sufficient to obtain this statistics precisely, especially in highly time-varying channel conditions. In addition to being too parametric, solutions from these two groups also require matrix-matrix multiplication and and matrix inversion operations which both have $O(m^3N^3)$ complexity if done in standard ways. One might argue that this filter update can be done at a low rate to decrease the complexity. In that case the receiver would not be able to track the fast varying channels. Even if only the low speed scenarios are considered, they are still not attractive for implementation neither with ASIC hardware nor with programmable vector processors. In hardware, they would occupy a lot of chip space. In software, they would put imbalanced load, making the processor MIPS scheduling troublesome. Due to all these reasons, implementations based on direct computations from the equalizer expression in Equation 1.31 are not preferable.

An alternative approach for equalizer implementation is adaptive filtering within which also there are several techniques associated with different optimization criteria [56, 57, 58, 59]. The two well-known adaptive techniques are RLS and LMS. RLS is not a very suitable method since it also has a high complexity of $O(m^2N^2)$. Moreover, in general it is not numerically stable, it cannot cope with non-stationary signals and it is negatively impacted by colored noise at its input since it inherently solves the deterministic least squares problem which requires white noise for convergence to Wiener (MMSE) solution [60, 61]. Unfortunately in wireless channels neither the received signal is stationary (due to time-varying channel) nor the noise, i.e. the additive interference, is white. LMS is on the contrary advantageous regarding all the mentioned aspects: It has low complexity of $O(mN)$, it is numerically stable and most importantly it is robust to modeling errors, disturbance variations and nonstationarities [60]. Due to these reasons, we restrict our focus to two chip level equalizers derived from the standard LMS algorithm.

4.2 Chip Level Adaptive Equalizers

In this section we look at chip-spaced filter implementations. Extensions to the poly-phase implementations of adaptive equalizers in the case of Rx-diversity (multiple antennas) and/or fractional sampling is straightforward, see for example a poly-phase symbol level LMS implementation in [56].

$\boldsymbol{R_{yy}}$ and $\boldsymbol{R_{by}}$ statistics are useful for LMMSE filtering purpose only for stationary $\boldsymbol{y}$, which unfortunately is not the case in time-varying channels. Still the LMMSE filtering equation is a good starting point for formulating LMS recursions. One can compute $\tilde{\boldsymbol{f}}_o$ exactly either directly from (1.31) or do it by the steepest descent method [62]. In the latter case, by starting from an initial filter weights assignment $\boldsymbol{f}_0$, one approaches to $\tilde{\boldsymbol{f}}_o$ iteratively as

$$\boldsymbol{f}_{l+1} = \boldsymbol{f}_l - \mu \nabla \boldsymbol{f}_l \tag{4.1}$$

by going in the opposite direction of the instantaneous MSE gradient vector

$$\nabla \boldsymbol{f}_l = \boldsymbol{f}_l \boldsymbol{R_{yy}} - \boldsymbol{R_{by}} \tag{4.2}$$

which is obtained from the standard Wiener filtering MSE expression [62].

By replacing the $\boldsymbol{R_{yy}}$ and $\boldsymbol{R_{by}}$ statistics by their respective instantaneous values $\boldsymbol{y}_l \boldsymbol{y}_l^H$ and $b[l]\boldsymbol{y}_l^H$, the standard LMS algorithm turns the explained *iterative* LMMSE scheme into a *data recursive* adaptive scheme which is adapted with every incoming sample (chip) as

$$\boldsymbol{f}_{l+1} = \boldsymbol{f}_l - \mu_l(\boldsymbol{f}_l \boldsymbol{y}_l - b[l-l_d])\boldsymbol{y}_l^H = \boldsymbol{f}_l + \mu_l e[l]\boldsymbol{y}_l^H \tag{4.3}$$

where l_d is the filter delay, μ_l is the step size (adapted as well), $b[l-l_d]$ is the desired response which we denote also as $d[l]$ and $\boldsymbol{y}_l = [y[l], y[l-1], \ldots, y[l-N+1]]^T$ is the input regression vector [62]. As seen from (4.3), the desired response $d[l]$ is the total BS transmitted signal, which unfortunately is not known.

Griffiths algorithm is a preferred method when a training sequence $d[l]$ is not available or is not reliable [63]. Similar to LMS, in equalizer context it also uses the total BS power and it is derived from the LMMSE equations (4.1) and (4.2), this time by replacing only the $\boldsymbol{R_{yy}}$ statistics with $\boldsymbol{y}_l \boldsymbol{y}_l^H$

$$\boldsymbol{f}_{l+1} = \boldsymbol{f}_l - \mu_l(\boldsymbol{f}_l \boldsymbol{y}_l \boldsymbol{y}_l^H - \sigma_b^2 \boldsymbol{h}_p^\dagger) \tag{4.4}$$

Normalized forms of LMS and Griffiths are obtained by normalizing the update terms with the input signal power:

$$\boldsymbol{f}_{l+1} = \boldsymbol{f}_l + \frac{\mu_l e[l]\boldsymbol{y}_l^H}{\boldsymbol{y}_l^H \boldsymbol{y}_l} \text{ (NLMS)} \tag{4.5}$$

$$\boldsymbol{f}_{l+1} = \boldsymbol{f}_l - \frac{\mu_l(\boldsymbol{f}_l \boldsymbol{y}_l \boldsymbol{y}_l^H - \sigma_b^2 \boldsymbol{h}_p^\dagger)}{\boldsymbol{y}_l^H \boldsymbol{y}_l} \text{ (N-Griffiths)} \tag{4.6}$$

N-Griffiths equalizer can be implemented directly from (4.6), however it requires the channel parameters. Implementation of NLMS equalizer is not that trivial since at first sight it seems that there is the single possibility of using the PCPICH as the desired signal. This conventional NLMS algorithm does not work well since the 10% of the BS signal power given to the PCPICH is too small compared to the interference level [59]. If one wants to obtain a more satisfactory performance from NLMS, one must find a way of exploiting more components from the transmitted total BS signal as the desired response.

4.3 Decision Directed HSDPA Equalizer

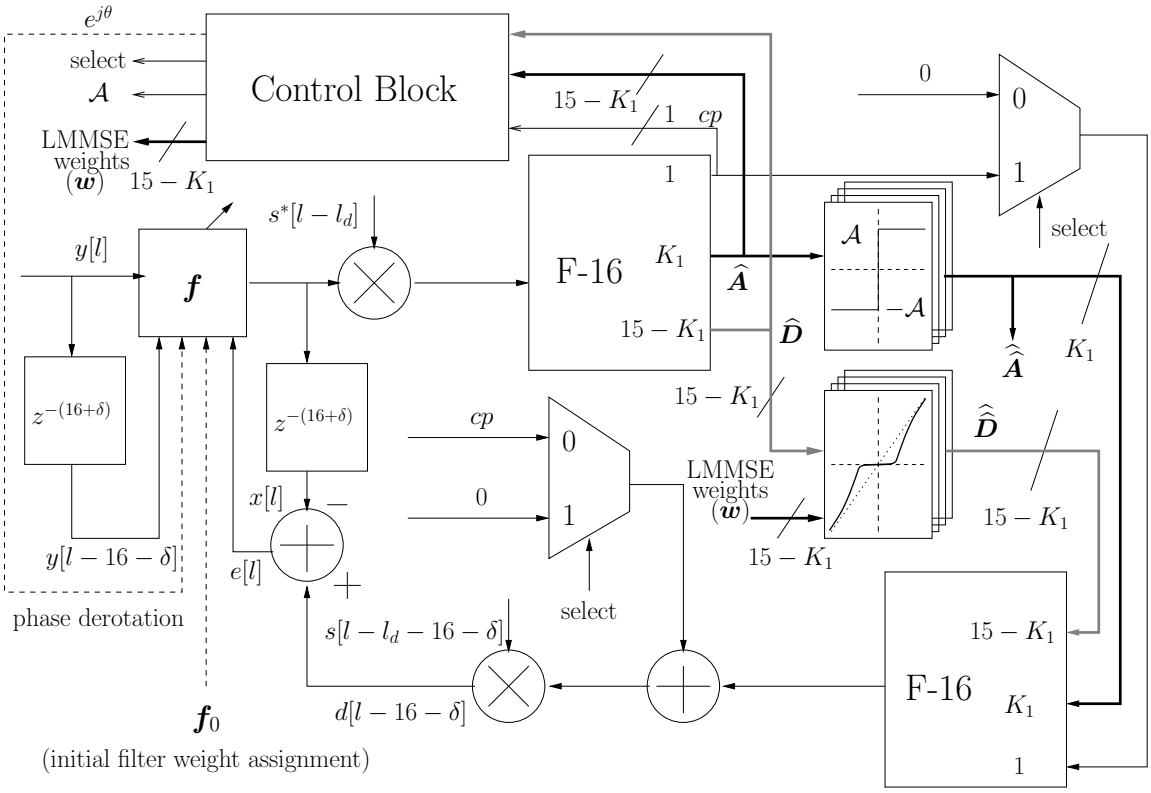


Figure 4.1: HSDPA-specific decision directed NLMS (HDD-NLMS) equalizer

Fig. 4.1 shows a schematic block diagram of our HDD-NLMS adaptive chip level equalizer. The scheme is motivated from the decision directed LMS equalization principle for single user ISI channels, which dates back to 1966 [3]. Since, in the context of CDMA downlink chip equalization, the decisions should be done on the transmitted total BS chip

sequence, this first obliges estimating all the user symbols by despreading and symbol post-processing such as hard decisions or LMMSE weightings and then returning back to chip level by respreading. These two cascade operations incur a delay of one HSPDSCH symbol period plus δ-chips-lasting feedback processing, i.e. $16+\delta$ chips, when the symbol decisions are constrained to SF-16. Therefore the same amount of delay should be applied to the input signal and the filter output. The goal is to adapt the NLMS equalizer by the desired sequence $d[l-16-\delta]$ fedback as the brute force estimate of the transmitted total BS chip sequence in the preceding symbol period. Although the data flows more than one symbol period ahead of the adaptation process, we can safely use the obtained filter weights due to the fact that this delay is a negligible time compared to the coherence time of typical wireless channels and hence the associated optimal equalizer weights do not change much during this period.

We use Fast Walsh Hadamard Transformation (FWHT) for efficiently implementing multiple despreading operations. If one wants to despread K codes with spreading factor L, using FWHT instead of K independent correlators decreases the complexity from KL units to $Llog_2(L)$ units. As long as $K > log_2(L)$, FWHT is advantageous. The crossover K value for SF-16 is $log_2(16) = 4$. Since in our system we are interested in despreading with all the 16 codes at SF-16 we use FWHT of length 16. We call it F-16 on the figure. F-16 outputs $\hat{\boldsymbol{A}}$ associated with HSPDSCHs of the user of interest are passed through hard decision blocks and fedforward as $\hat{\hat{\boldsymbol{A}}}$ to the channel decoder and other post processing units.

We use three different means for treating the F-16 corelator outputs in the upper filtering branch to feedback to the lower adaptation branch:

i. K_1 soft HSPDSCH symbol estimates $\widehat{\boldsymbol{A}}$ are passed through slicers, i.e. hard detected, and hence the resultant $\widehat{\widehat{\boldsymbol{A}}}$ are supposed to carry the most reliable components of the desired signal estimate. This is actually the case as long as mostly correct detections are made and as long as HSPDSCH symbol amplitudes are estimated precisely.

ii. The remaining correlator outputs $\widehat{\boldsymbol{D}}$, except the first one, are fedback as $\widehat{\widehat{\boldsymbol{D}}}$ scaled by seperate LMMSE weights. In fact we do not know a priori the active spreading codes in the OVSF code space which are spanned by the spreading codes of these $15-K_1$ correlators. However, as long as hard decisions or other nonlinear operations which definitely require the symbol constellations and the symbol amplitudes are not considered, one does not need to know the actual channelization (spreading) codes and one does not need to estimate their symbols. In this case it is equally sufficient to get pseudo-symbol estimates

reflected from the actual symbols residing at particular places in the OVSF hierarchy to the SF-16 level and to apply LMMSE weights on these pseudo-symbols.

Say the instantaneous power on any correlator output with index u is $|\hat{a}_u|^2$ and the noise-plus-interference variance is $\sigma^2_{n_u}$. Then the instantaneous LMMSE weight for that output will be $w_u = \frac{|\hat{a}_u|^2 - \sigma^2_{n_u}}{|\hat{a}_u|^2}$. The numerator term corresponds to the useful signal power and the denominator term corresponds to the sum of the useful signal power and the noise-plus-interference power. If the estimated LMMSE weight on any particular branch is negative, then it is replaced by zero. This latter situation is equivalent to excluding those outputs from the feedback operation and it occurs when the power at that particular branch is below $\sigma^2_{n_u}$.

iii. The first correlator output, i.e. the *cp* output from despreading with 16-ones-code, partially despreads PCPICH, PCCPCH and all the other active codes under the OVSF subtree rooted from the code $c_{16,0}$. There are two possible approaches here. The first option is excluding the output of this branch from the feedback operation but instead adding the PCPICH chip sequence in a hard manner since the PCPICH sequence is a known sequence. The advantage is that the added term is not noisy. However it has two disadvantages. First of all one needs to also estimate the PCPICH amplitude. Secondly and more importantly, once PCPICH is explicitly fedback, one cannot exploit the signal contributions from any other code under the OVSF subtree rooted from $c_{16,0}$. The second option is feeding back by LMMSE scaling as is done for the other remaining branches.

The multiplexing mechanism between the hard PCPICH addition and scaled linear feedback of the first correlator output requires defining a threshold value for the first correlator output power.

Lemma 4.3.1 *Let PCPICH power be P_{cp}, the instantaneous power at the first correlator output be $|\hat{a}_1|^2$ and the LMMSE weight be $w_1 = \frac{|\hat{a}_1|^2 - \sigma^2_{n1}}{|\hat{a}_1|^2}$. Then the optimal $|\hat{a}_1|^2$ threshold value for multiplexing PCPICH and the LMMSE weighted first correlator output is $P_{thr} = \frac{P_{cp}}{w_1^2} + 2\sigma^2_{n_1}$.*

Proof. The useful signal power at the LMMSE weighting output is $\left(|\hat{a}_1|^2 - \sigma^2_{n_1}\right) w_1^2$. The noise-plus-interference power at the LMMSE weighting output is $\sigma^2_{n_1} w_1^2$. We take the useful signal power as a reference. Then selection of pure PCPICH signal can be considered as an estimation with error variance equal to the power difference between the useful signal

power and P_{cp}. That is to say if PCPCH is selected then this selection has a constant *sample variance*. In order for the two options to have the same variance this variance should be equal to the noise-plus-interference power at the LMMSE weighting output

$$\left(|\hat{a}_1|^2 - \sigma_{n_1}^2\right) w_1^2 - P_{cp} = \sigma_{n_1}^2 w_1^2 \Rightarrow P_{thr} = \frac{P_{cp}}{w_1^2} + 2\sigma_{n_1}^2 \tag{4.7}$$

□

We add the pilot signal to the feedback path in hard manner if the power on the first branch is smaller than P_{thr} obtained in (4.7), which is estimated and updated once every CPICH symbol period in the *Control Block* in Fig. 4.1.

The overall feedback strategy improves the energy of the desired signal and allows for a better tracking of the channel. Moreover, this recursive process can be interpreted as a learning process also for the desired signal. With each recursion, the quality of filter weights and thus the detected or estimated feedback signal, i.e. the desired signal, is improved.

4.3.1 Misconvergence Problem

Any decision directed scheme is prone to misconvergence problem. This is a phenomenon which occurs when the equalizer locks to a rotated constellation state, does systematic errors all the time and cannot recover from there [58]. A common remedy is to have a backup solution such as a pure pilot-aided method or a constant modulus algorithm (CMA) [64, 65] which takes the turn when the equalizer diverges and gives the turn back to the decision-directed scheme when SINR conditions are again above an acceptable level [58, 66, 67, 68, 69]. Since we do not want to have any backup solution and we want instead to *avoid* misconvergence, in the *Control Block* we obtain a Super-PCPICH-Symbol, i.e. sum of a block of PCPICH-symbols, every 5 or 10 PCPICH symbol periods [2] and derotate the equalizer filter weights by an angle θ which is equal to the difference between the phase of the estimated Super-PCPICH Symbol and 45 degrees, which is the correct phase of the pilot signal. This, interestingly enough, adds a local zero-forcing (ZF) dimension to the global MMSE equalization problem.

The PCPICH tone is a significant element, not only for avoiding misconvergence most of the time but also bringing the filter back to convergence state if misconvergence cannot be avoided in deep fades. Consider the very initialization of the adaptation, for

[2] a Doppler-spread and noise dependent design parameter which can be taken less or more

example. We are using the channel matched filter (CMF), i.e. the Rake receiver in its FIR form for the initialization of the filter weights. This is a nice-to-have but not a strictly essential feature. Even if we start with the all-zeros filter weights, by the aid of the CPICH zero-forcing mechanism, the filter passes the transient phase and the filter output locks to the correct constellation. This of course takes longer than starting with the CMF.

4.4 Amplitude Estimation

For HDD-NLMS, in order to feedback hard detected HSPDSCH symbols and to apply LMMSE weightings, we need to have both noise-plus-interference variance estimators and amplitude estimators.

4.4.1 Amplitude Estimation for QPSK Symbols

Recalling the SINR expression in (1.25), the denominator part which denotes the noise-plus-interference term can be estimated from the following propositions:

Corollary 4.4.1 *The denominator term of the SINR expression in (1.25) does not depend on the code identity.*

Corollary 4.4.2 *The denominator term of the SINR expression in (1.25) is inversely proportional on the spreading factor.*

Corollary 4.4.3 *From Corollary 4.4.1 and Corollary 4.4.2 we can conclude that once the value of the SINR denominator term is found for one code, the same term for another code can be easily obtained by scaling the first term with the spreading factor ratio of the two codes.*

Remark 4.4.4 *The despreading functionality is real-valued.*

By using Remark 4.4.4, we can separate the input and output of the despreading block into real (I) and imaginary (Q) branches and reach to the equality in Figure 4.2.

Remark 4.4.5 *If one code is actively assigned to a user from the OVSF tree, then its parent or child codes cannot be assigned to any other user, i.e. they are inactive.*

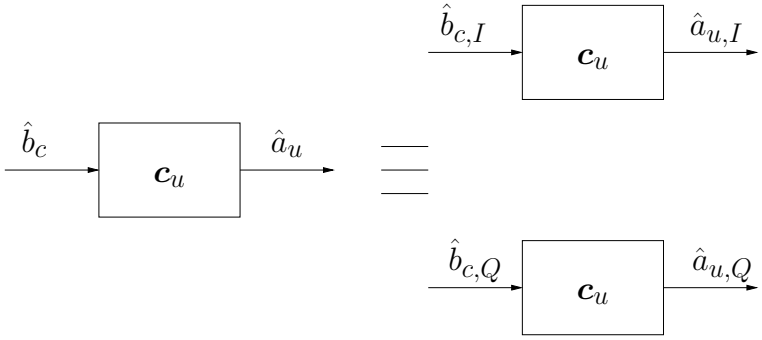


Figure 4.2: I-Q seperation (parallel implementation) of the despreading operation

Lemma 4.4.6 *Despreading with the two inactive child codes of $\boldsymbol{c}_u$, i.e. despreading with $\boldsymbol{c}_1 = [\boldsymbol{c}_u \boldsymbol{c}_u]$ and $\boldsymbol{c}_2 = [\boldsymbol{c}_u - \boldsymbol{c}_u]$ will give the two code multiplexed pseudo-symbol estimates reflected from the two consecutive symbols on $\boldsymbol{c}_u$ as*

$$\hat{a}_1(m) = \hat{a}_u(2m-1) + \hat{a}_u(2m) \tag{4.8}$$

$$\hat{a}_2(m) = \hat{a}_u(2m-1) - \hat{a}_u(2m) \tag{4.9}$$

where m denotes the symbol index.

Using the Remark 4.4.4, partitioning these pseudo-symbols into their I-Q components will give

$$\hat{a}_{1,I}(m) = \hat{a}_{u,I}(2m-1) + \hat{a}_{u,I}(2m) + \hat{n}_{u,I}(2m-1) + \hat{n}_{u,I}(2m) \tag{4.10}$$

$$= a_{1,I}(m) + n_{1,I}(m) \tag{4.11}$$

$$\hat{a}_{2,I}(m) = \hat{a}_{u,I}(2m-1) - \hat{a}_{u,I}(2m) + \hat{n}_{u,I}(2m-1) - \hat{n}_{u,I}(2m) \tag{4.12}$$

$$= a_{2,I}(m) + n_{2,I}(m) \tag{4.13}$$

Theorem 4.4.7 *Since the real part (imaginary part) of QPSK modulation has only two values i.e. a scaled form of {+1,-1}, either $a_{1,I}(m)$ or $a_{2,I}(m)$ has to be equal to zero. Both cannot be nonzero at the same time.*

Proof. There are four possible combinations of $a_{u,I}(2m-1)$ and $a_{u,I}(2m)$ for generating $a_{1,I}(m)$ and $a_{2,I}(m)$ as shown in Table 4.1

As is seen in Table 4.1, one of $a_{1,I}(m)$ or $a_{2,I}(m)$ is zero. □

Corollary 4.4.8 *Due to Corollary 4.4.1 and Corollary 4.4.2, $\sigma^2_{n_{1,I}} = \sigma^2_{n_{2,I}} = 2\sigma^2_{n_{u,I}}$.*

Table 4.1: Combinations of two consecutive QPSK symbols

$a_{u,I}(2m-1)$	$a_{u,I}(2m)$	$a_{1,I}(m)$	$a_{2,I}(m)$
1	1	2	0
1	-1	0	2
-1	1	0	-2
-1	-1	-2	0

Using Theorem 4.4.7 and Corollary 4.4.8, we can easily calculate the I (real) component of the noise-plus-interference sample variance as

$$\sigma^2_{n_{u,I}} = \frac{min\left\{|\hat{a}_{1,I}(m)|^2, |\hat{a}_{2,I}(m)|^2\right\}}{2} \tag{4.14}$$

Similar reasoning and procedure can be followed to obtain the Q (imaginary) component as

$$\sigma^2_{n_{u,Q}} = \frac{min\left\{|\hat{a}_{1,Q}(m)|^2, |\hat{a}_{2,Q}(m)|^2\right\}}{2} \tag{4.15}$$

Finally

$$\sigma^2_{n_u} = \sigma^2_{n_{u,I}} + \sigma^2_{n_{u,Q}} \tag{4.16}$$

Figure 4.3 shows a block diagram of the explained method.

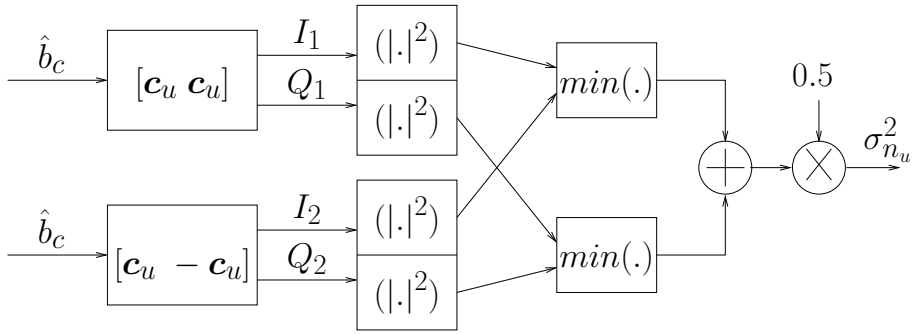


Figure 4.3: QPSK noise-plus-interference variance estimation

The quality of the estimated variance can be improved in several ways:

- by taking moving average of the estimated values during a number of consecutive symbol instants

- by repeating the procedure on a number of other known codes with QPSK modulation like some common codes as PCPICH, PCCPCH and then by doing weighted averaging among them. For this purpose one has to
  - first obtain the estimate of σ_n^2 from the $\sigma_{n_u}^2$ estimate over each known code by using the relation $\sigma_{n_u}^2 = L_u \sigma_n^2$
  - then do weighted averaging among the variances obtained in the first step. Weighting would be beneficial since, for example an estimate of σ_n^2 from a code with spreading factor L=128 would in ideal conditions have 8 times less variance than an estimate of σ_n^2 from a code with spreading factor L=16. On the other hand phase drift impacts at the synchronization modules might favour giving more weight to low spreading factors than their weight in ideal conditions.
- In principle both moving averaging and inter-code averaging should be applied.

The scheme can be applied on any active code with QPSK modulation, meaning that it can also be applied on any unused code as long as there is no activity on its parent or child codes. We can claim this since that unused code can be considered as a QPSK modulated code with zero power.

Instead of temporal application, the explained scheme can also be applied among couples of QPSK modulated codes with the same power at the same SF level as well. Adding and subtracting the two symbols will give one zero value for the real and imaginary parts. This is very much applicable among QPSK modulated HSPDSCH codes.

Once σ_n^2 is estimated and refined by the explained mechanisms, by using Corollary 4.4.2, it can be used in the amplitude estimation of any user‘s symbol.

Let symbol-plus-noise-plus-interference power for code u be $\sigma_{\hat{a}_u}^2$. Then subtracting the $\sigma_{n_u}^2$ estimate from $\sigma_{\hat{a}_u}^2$ will give the estimate of $\sigma_{\hat{a}_u}^2$. Finally the real and imaginary amplitude of the QPSK symbol can be obtained as $\mathcal{A}_{u,I} = \mathcal{A}_{u,Q} = \sqrt{\frac{\sigma_{\hat{a}_u}^2}{2}}$.

An amplitude estimation scheme for 16-QAM modulation is given in Appendix C.

4.5 Simulations and Conclusions

For simulations, we consider 3GPP-RAN4 compatible HSDPA service scenarios in the UMTS FDD downlink for mobile terminals from Category 7 and Category 8.

Table 4.2 shows the simulation settings.

Table 4.3 shows the ITU Vehicular-A channel power delay profile that we use for simulations [8].

Table 4.2: Simulation Settings

Parameters	**Settings**
Chip rate	3.84 Mcps
Number of HSPDSCH codes	10 (All assumed to be belonging to the user of interest)
Modulation scheme	QPSK
PCPICH power	10% of the BS power (-10dB)
PCCPCH power	6.3% of the BS power (-12dB)
Total power in the first subtree	25% of the the BS power
Total HSDSCH power (E_c)	25% or 50% of the BS power
$\hat{I}_{or}/I_{oc}$	6dB or 10dB
OCNS power	Remaining BS power randomly distributed to the 5 remaining codes at SF level 16
Equalizer tap spacing	1 chip
Number of receive antennas	1
Equalizer length	24
Equalizer adaptation rate	Once every 2 chips
Transmission pulse shape	Root-Raised-Cosine (rrc) with roll-off factor 0.22
Channel model	Jakes fading model ([70])
Channel power delay profile	ITU Vehicular A
Mobile speeds	30km/h and 120km/h
Channel update rate	Once every 16 chips

Table 4.3: Vehicular A Channel Power Delay Profile

Relative Delay [ns]	0	310	710	1090	1730	2510
Relative Mean Power [dB]	0	-1	-9	-10	-15	-20

Table 4.4: Channel Estimation NMSE Simulation Settings for Griffiths Equalizer

Channel Estimation Quality	30km/h		120km/h	
	$\hat{I}_{or}/I_{oc} = 10dB$	$\hat{I}_{or}/I_{oc} = 6dB$	$\hat{I}_{or}/I_{oc} = 10dB$	$\hat{I}_{or}/I_{oc} = 6dB$
Low Quality	-8	-7	-6	-5
Medium Quality	-12	-11	-10	-9
High Quality	-18	-16	-15	-13

We obtain the continuous time *transmission* channel by convolving the *propagation* channel with the pulse shape. We pass to the discrete time model by first sampling the continuous time channel and then truncating from two sides the tails that spread more than four chips far away from the centers of the channel components which are the convolution results of the pulse shape with the first and the last propagation channel paths. Therefore the propagation channel pulse shape cascade, i.e. the overall channel, has a length of 19 chips at 3.84 Mchips/sec transmission rate.

We applied the noise-variance estimation and QPSK amplitude estimation techniques explained in the text.

For both schemes we set the initial step size to $\mu_0 = 0.03$ and adapt it on the run by a method that looks at the inner product value between the gradient vectors at the present state and the previous state [71]:

$$\mu_{l+1} = \mu_l + \eta \Re\{< \nabla \boldsymbol{f}_l, \nabla \boldsymbol{f}_{l-1} >\} \tag{4.17}$$

This method simply tells that if the filter weights are updated in similar directions during two consecutive recursions (the phase is less than 90°), then it is an indication that there is still a long way to go in such directions to reach to the desired solution point (Wiener solution), so it makes sense to increase the step size. If the direction is changing, then the adaptive process is at steady state doing Brownian motions around the desired solution, so it is reasonable decrease the step size in order to decrease the amplitude of oscillations. As a single modification on the original scheme, in order for the mobile to recover from the outage states rapidly, when the SINR falls below a defined threshold such as $-6dB$ we freeze the step size adaptation and fix the step size to a high value such as 0.1 and restart adaptation when the SINR goes above a higher threshold such as $-3dB$.

On Figures 4.4 to 4.11 $\{\hat{I}_{or}, I_{oc}, E_c\}$ denote respectively {chip level received BS signal power, additive white noise power modeling also the intercell interference, total power assigned to HSPDSCH codes}.

On Figures 4.4 to 4.11 {D, C, GL, GM, GH, M} respectively denote {HDD-NLMS, CMF, Griffiths with a low quality channel estimator, Griffiths with a medium quality channel estimator, Griffiths with a high quality channel estimator, Max-SINR}.

CMF serves as the SNR bound and Max-SINR receiver serves as the SINR bound. Therefore, for CMF and Max-SINR we assumed that we have ideal channel information.

For Griffiths, however, to have a reasonably fair comparison with HDD-NLMS

scheme, we perturbed the correct channel parameters by adding random Gaussian noise complying with the normalized MSE values set in Table 4.4. Since the performance of the Griffiths equalizer depends on the quality of channel estimation we defined three performance categories for possible channel estimators. Without loss of generality we consider PCPICH-aided channel estimators working on blocks of received samples as *low quality*, channel estimators that also exploit the channel dynamics as *medium quality* and finally channel estimators that exploit both the channel dynamics and all the existing pilot sequences in the system as *high quality*. The values in Table 4.4 are some judiciously chosen values taking into account the performances of the channel estimation techniques obtained in Chapter 3. They can be modified to reflect the performance of any other channel estimation technique. In order to have more precise results, one should integrate the actual channel estimator which is considered for implementation into the system model.

For plotting convenience, we obtain BER results for each TTI, i.e. 3 UMTS slots, and we sample the instantaneously obtained SINR every slot.

As seen from Figure 4.4 to Figure 4.11, both HDD-NLMS and Griffiths perform much better than the CMF.

HDD-NLMS scheme performs significantly better than Griffiths in several conditions. Furthermore at high SNR regions, i.e. when far from deep fades, HDD-NLMS performance comes very close to the Max-SINR performance.

The impact of channel estimation quality is clearly seen when we compare the GL, GM and GH performances. Still the differences among them are not very significant especially at 30km/h mobile speed. Therefore we conclude that Griffiths is a very robust scheme against channel estimation errors.

HDD-NLMS performance is sometimes getting worse than other schemes in very deep fades. As explained in the text this is mostly attributable to not assigning a backup solution for HDD-NLMS in such cases. However one must at the same time note that during deep fades, the mobile terminal will most probably not be scheduled by the BS. Taking into account such realistic HSDPA schedulers we can exclude deep fades from comparisons. In all the other cases HDD-NLMS is a better performing solution. Moreover it does not require channel parameters.

HDD-NLMS has also some limitations. It cannot be directly used during inactive periods when the mobile does not receive any HSDPA service. However, since UMTS downlink is a code limited system, when the mobile is not scheduled Node-B will most

probably assign the same codes to the user who gets the service. Even if this is not the case it is easy to detect the active HSPDSCH codes since the code search space is only limited to 15 codes, the HSPDSCH codes are placed consecutively and the constellation is limited to only QPSK and 16-QAM modulations. Decision directed schemes are also known to have convergence problems with QAM modulations. Therefore Griffiths alone is perhaps a more proper solution for 16-QAM. However a better performing 16-QAM solution would be to multiplex the two, Griffiths serving as an eye-opener for HDD-NLMS.

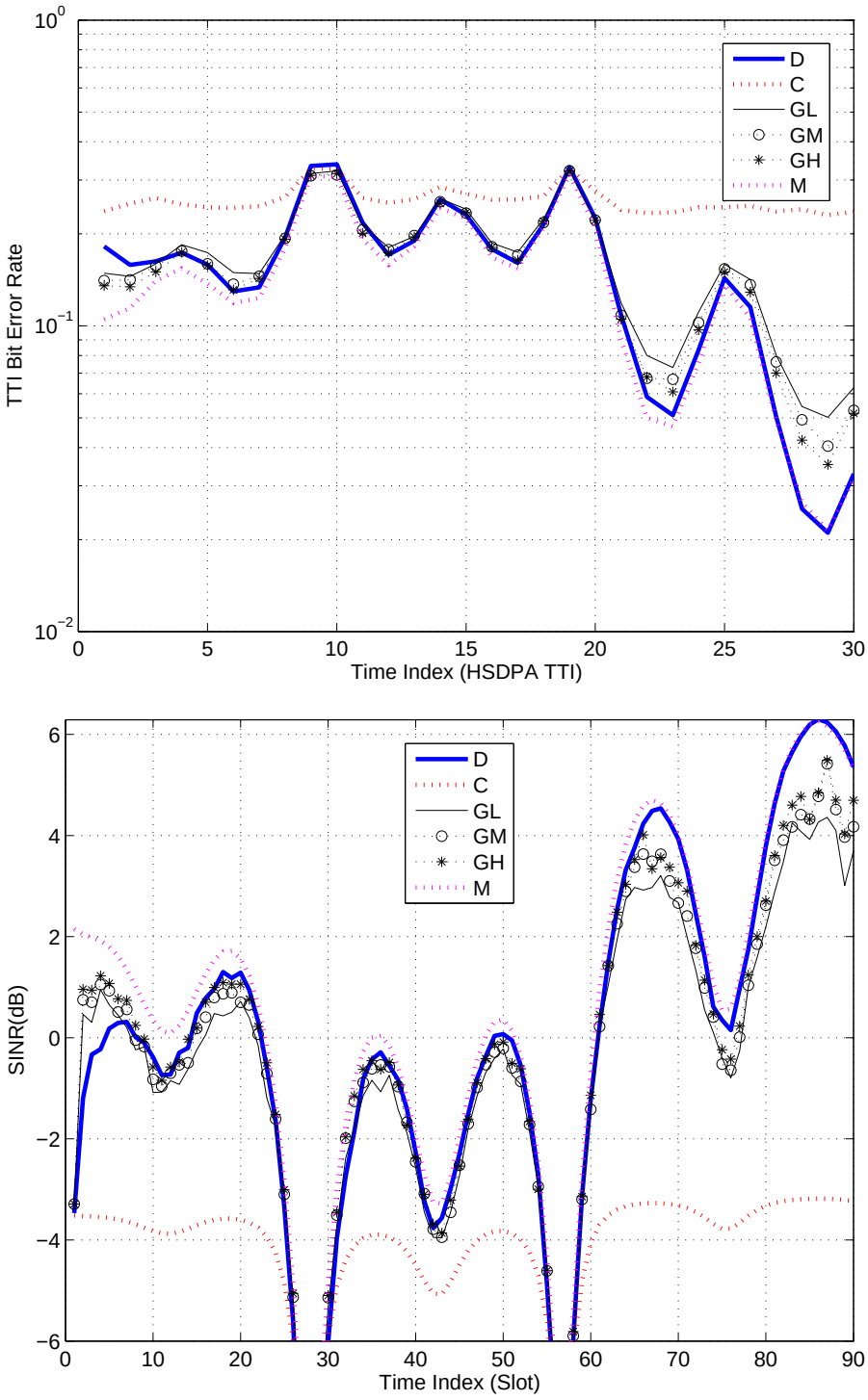


Figure 4.4: vA30 channel profile, 10 codes, $E_c/\hat{I}_{or} = -6dB$, $\hat{I}_{or}/I_{oc} = 10dB$

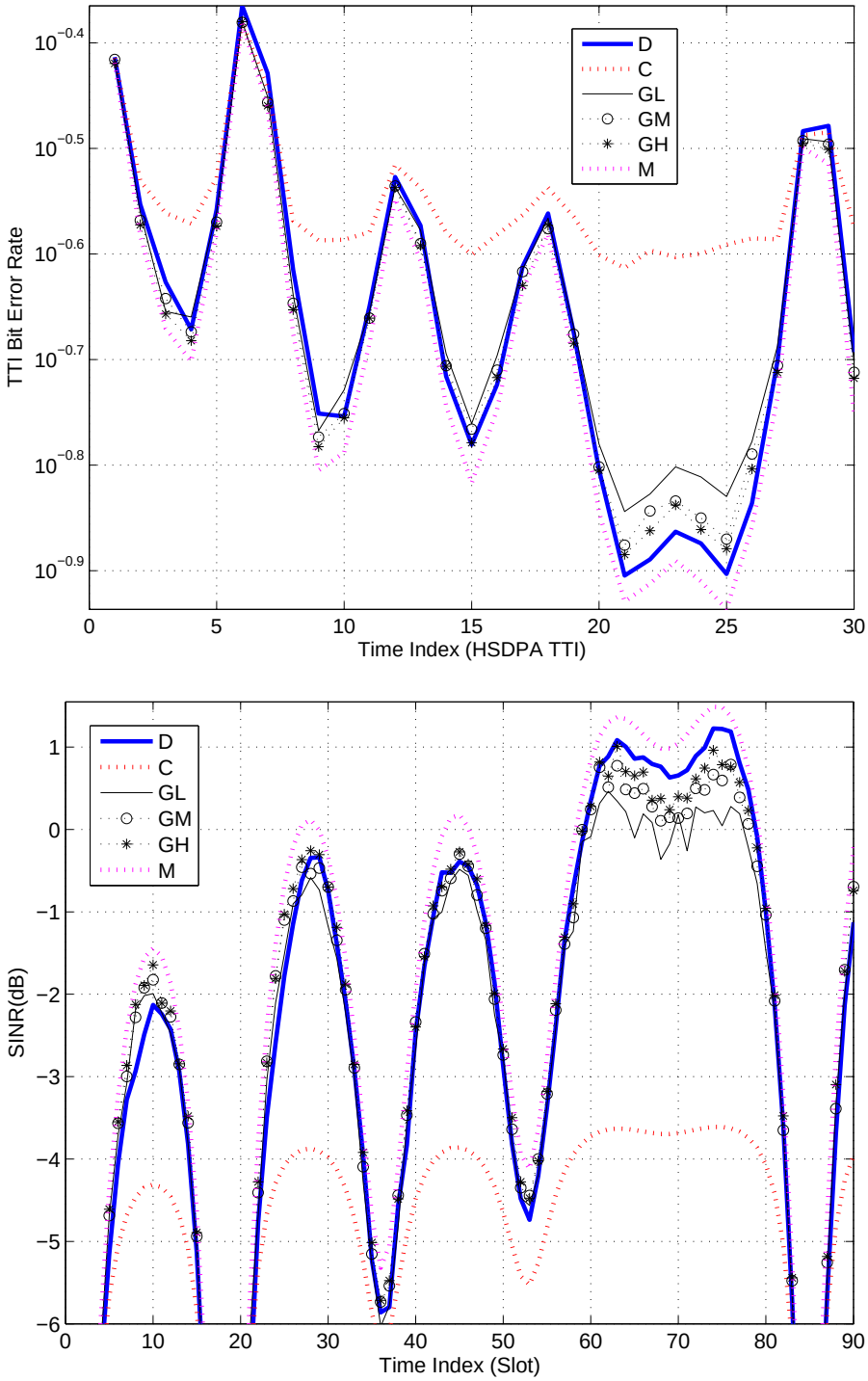


Figure 4.5: vA30 channel profile, 10 codes, $E_c/\hat{I}_{or} = -6dB$, $\hat{I}_{or}/I_{oc} = 6dB$

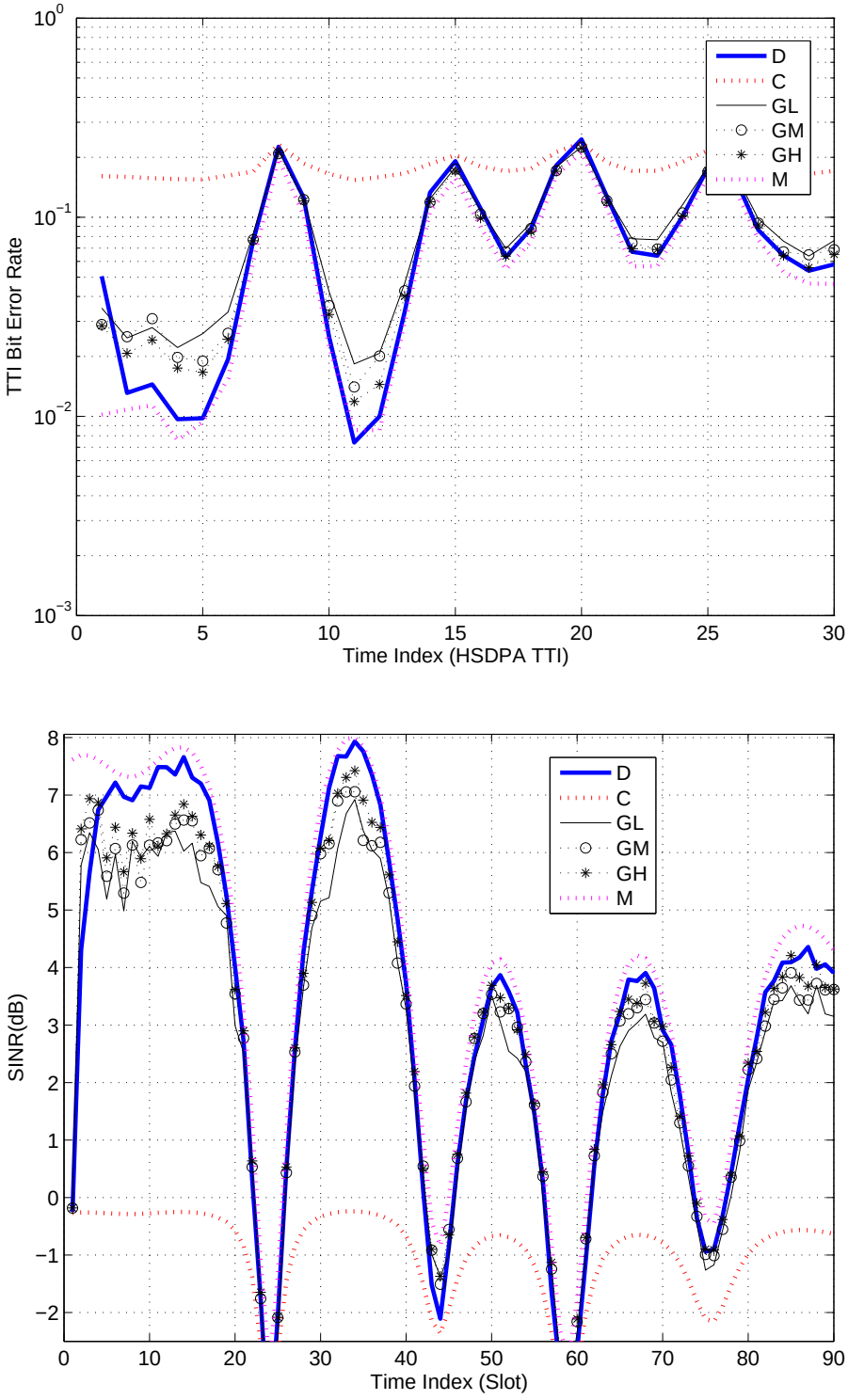


Figure 4.6: vA30 channel profile, 10 codes, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 10dB$

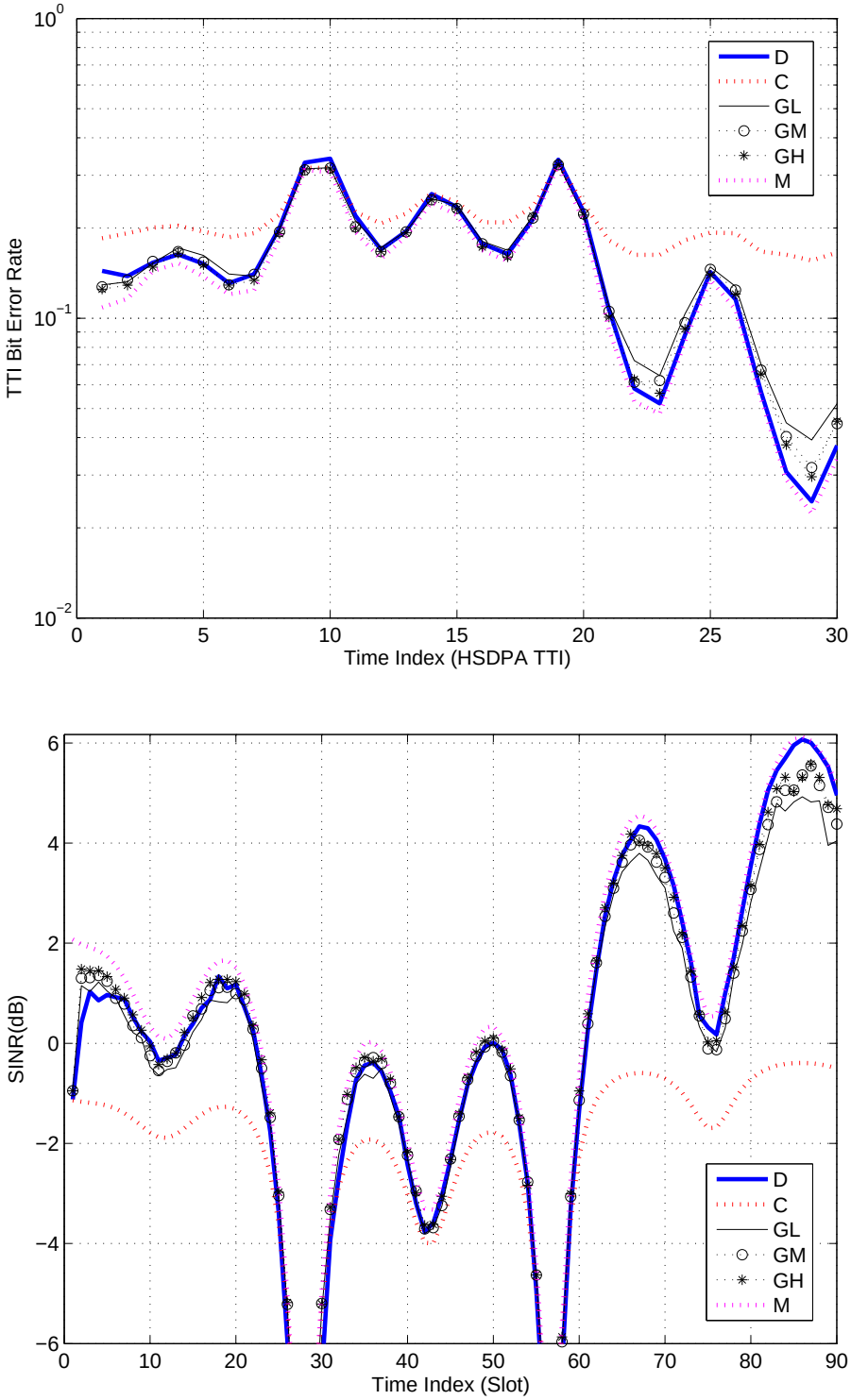


Figure 4.7: vA30 channel profile, 10 codes, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 6dB$

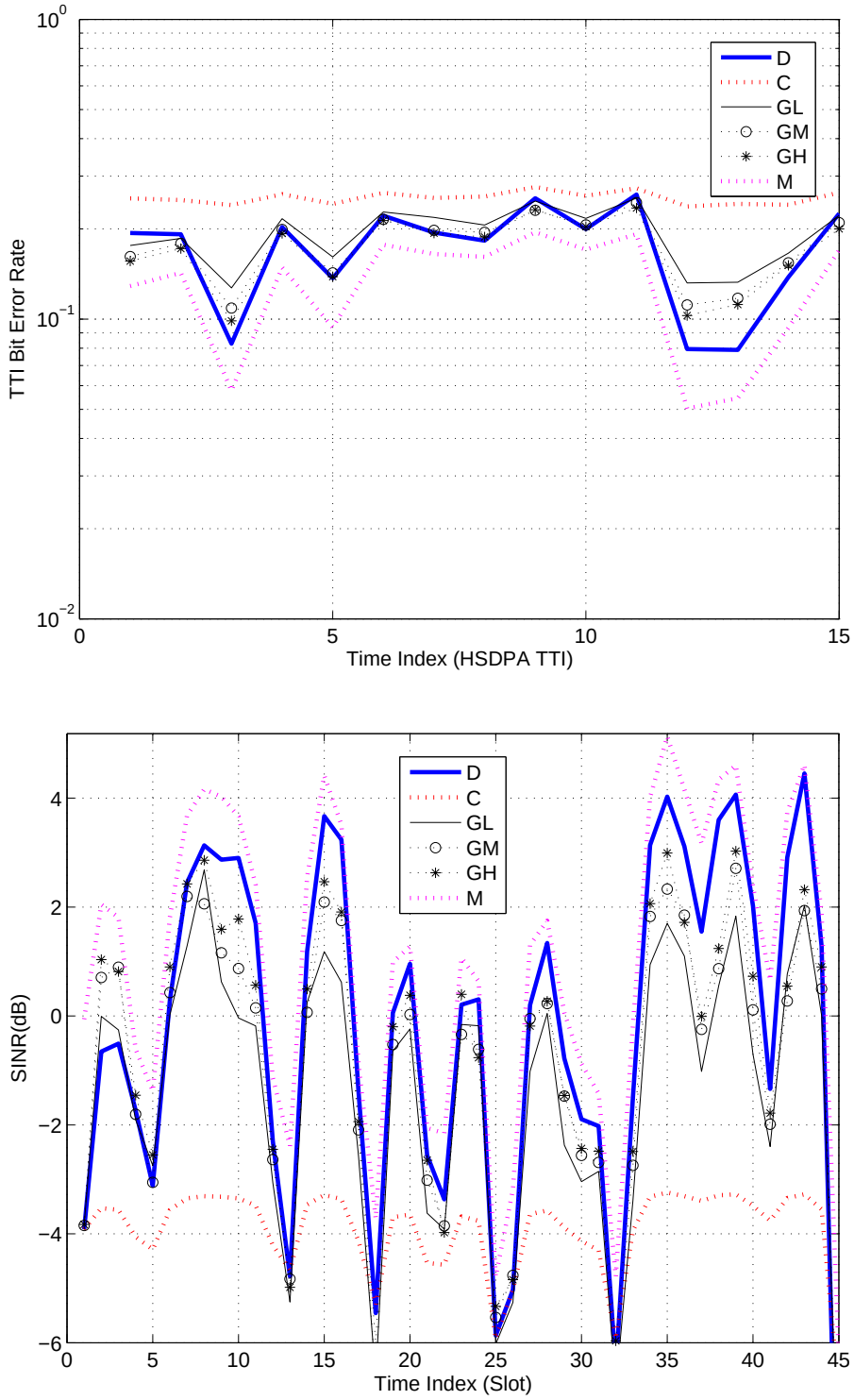


Figure 4.8: vA120 channel profile, 10 codes, $E_c/\hat{I}_{or} = -6dB$, $\hat{I}_{or}/I_{oc} = 10dB$

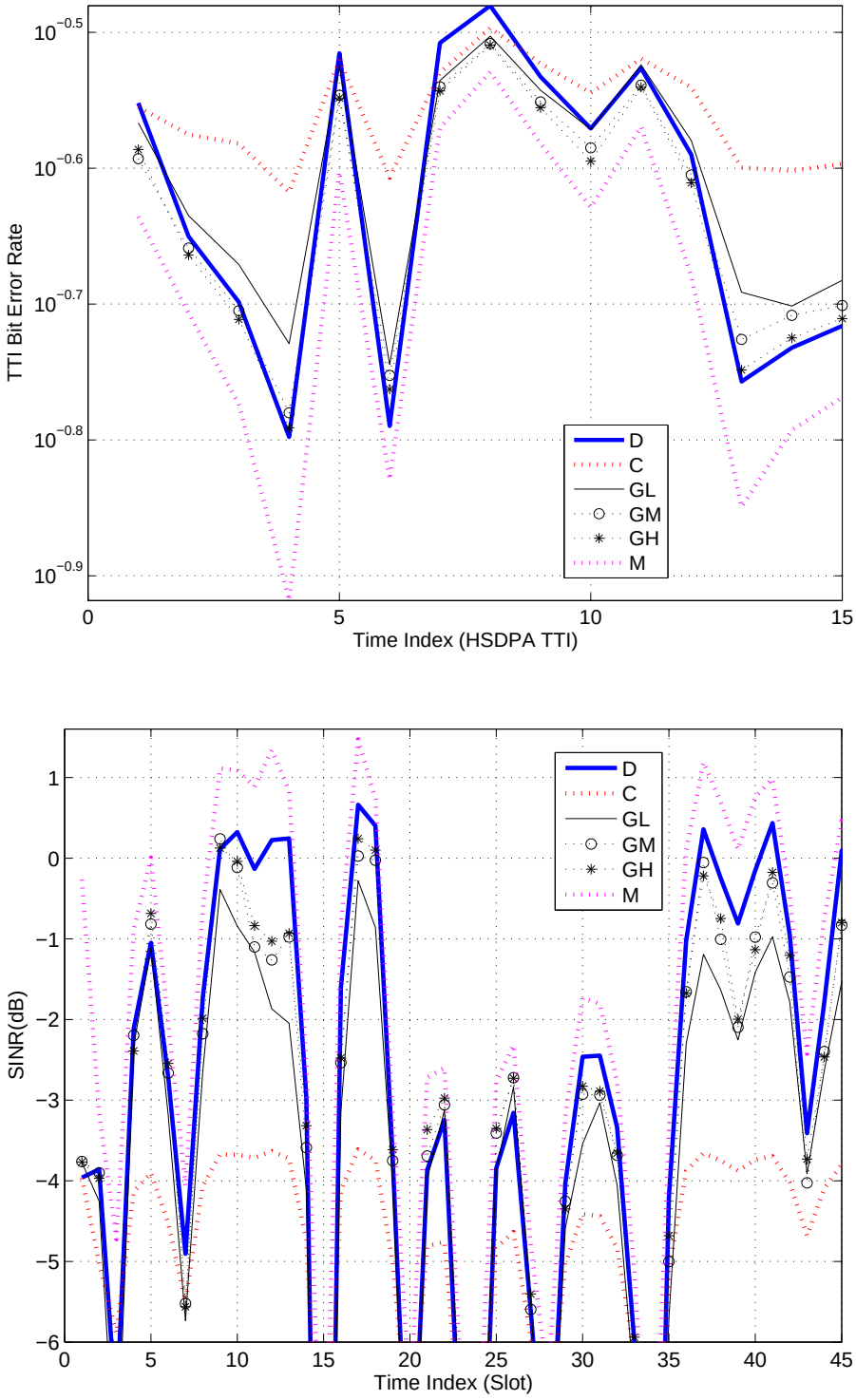


Figure 4.9: vA120 channel profile, 10 codes, $E_c/\hat{I}_{or} = -6dB$, $\hat{I}_{or}/I_{oc} = 6dB$

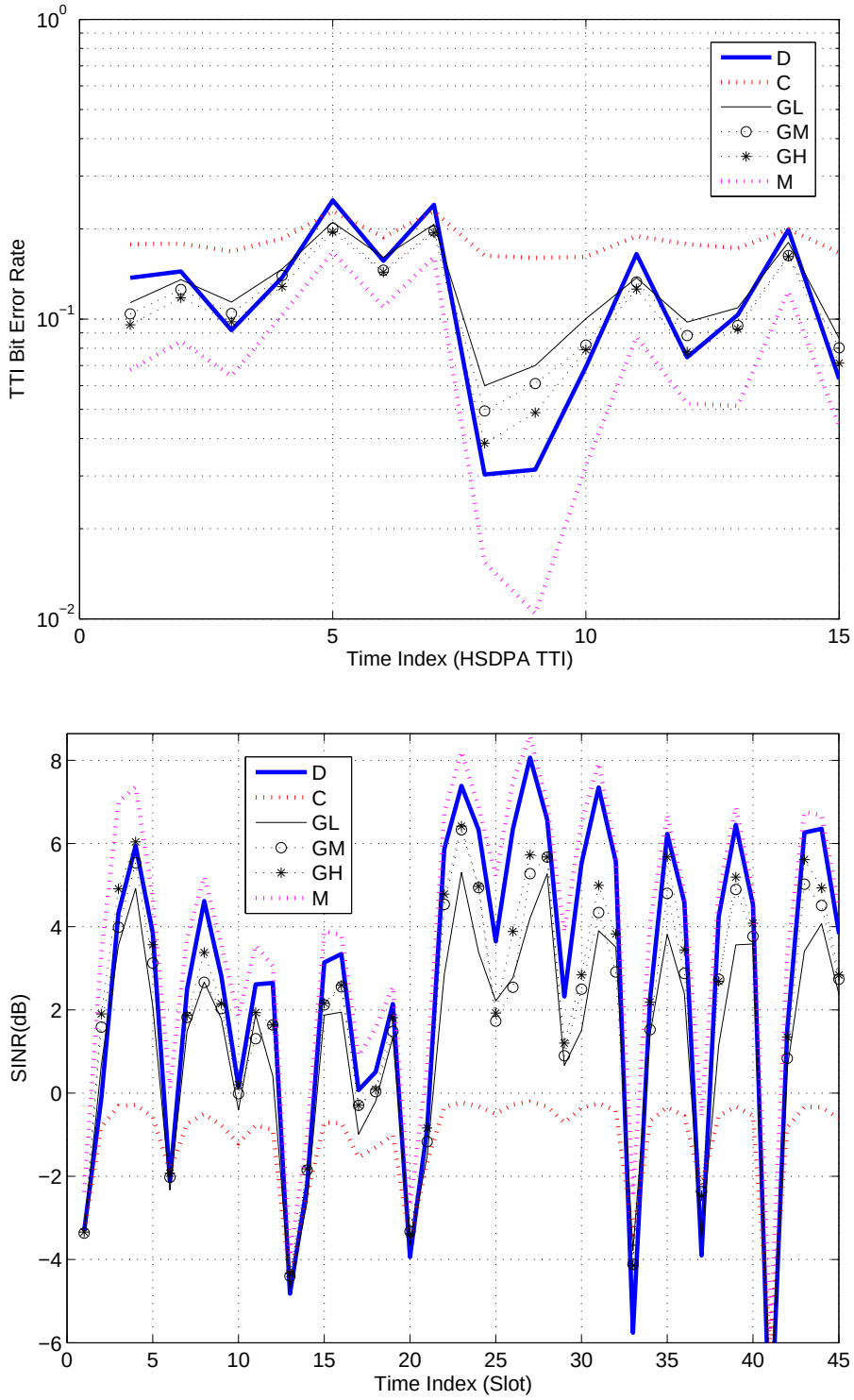


Figure 4.10: vA120 channel profile, 10 codes, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 10dB$

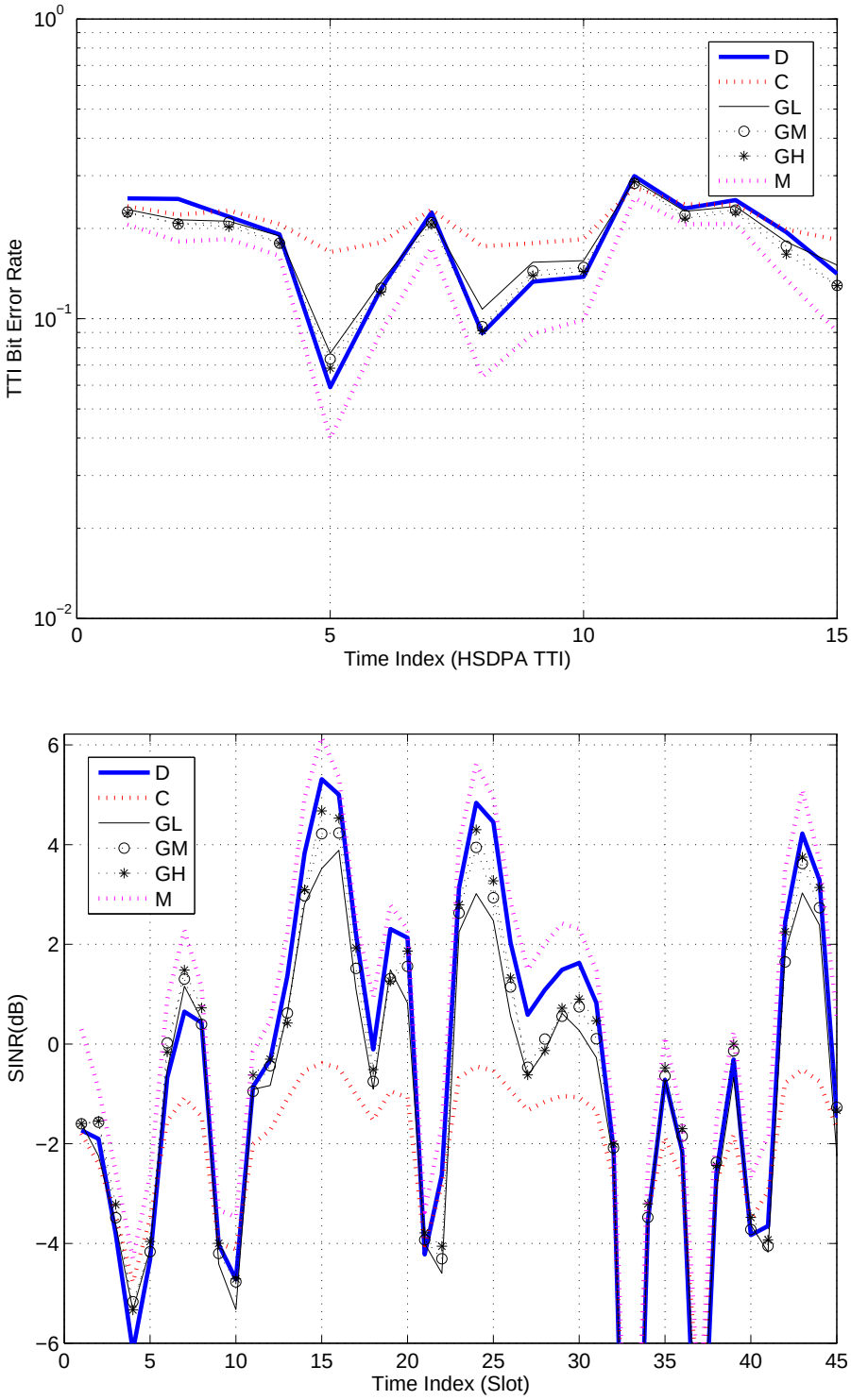


Figure 4.11: vA120 channel profile, 10 codes, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 6dB$

Chapter 5

Adaptive Equalization by Group Despreading

In this chapter we generalize the decision directed equalization concept by incorporating the despreading operation. Standard symbol level LMS algorithm which does full despreading with the PCPICH code is slow in tracking highly varying channels. Group despreading, i.e. despreading with partial PCPICH code is a means to increase the adaptation rate and hence to increase the tracking capability. However, contrary to the full despreading, this might even amplify the level of interference over the PCPICH tone instead of suppressing it. In order to avoid this situation as much as possible we propose a decision directed mechanism, which estimates the ingredients of the pseudo-symbol output of the partial despreading operation with hard decisions and LMMSE weightings. We further generalize the scheme to other possible implementations when mobile have access to HSPDSCH codes residing in particular locations of the OVSF.

5.1 Introduction

Unlike TDMA systems like GSM, pilot-aided equalizer design for CDMA systems is very problematic [59]. In TDMA systems common pilot signal is time-multiplexed with payload data. Therefore it is not interfered by any other same BS signal but only the co-channel interference coming from the other cells and the AWGN. However in CDMA systems like UMTS FDD downlink, pilot data, i.e. the PCPICH, is code-multiplexed with all the other existing users and the control channels [8]. Therefore, since affected by a high amount of interference, it cannot be used efficiently for training the equalizer weights at baud rate, i.e. at chip rate [59]. In order to remedy this situation Frank et.al considered first despreading the received signal with the PCPICH code, hence suppressing most of the

interference over the PCPICH signal [72]. Later Petre et.al extended it to the fractionally spaced implementations [56].

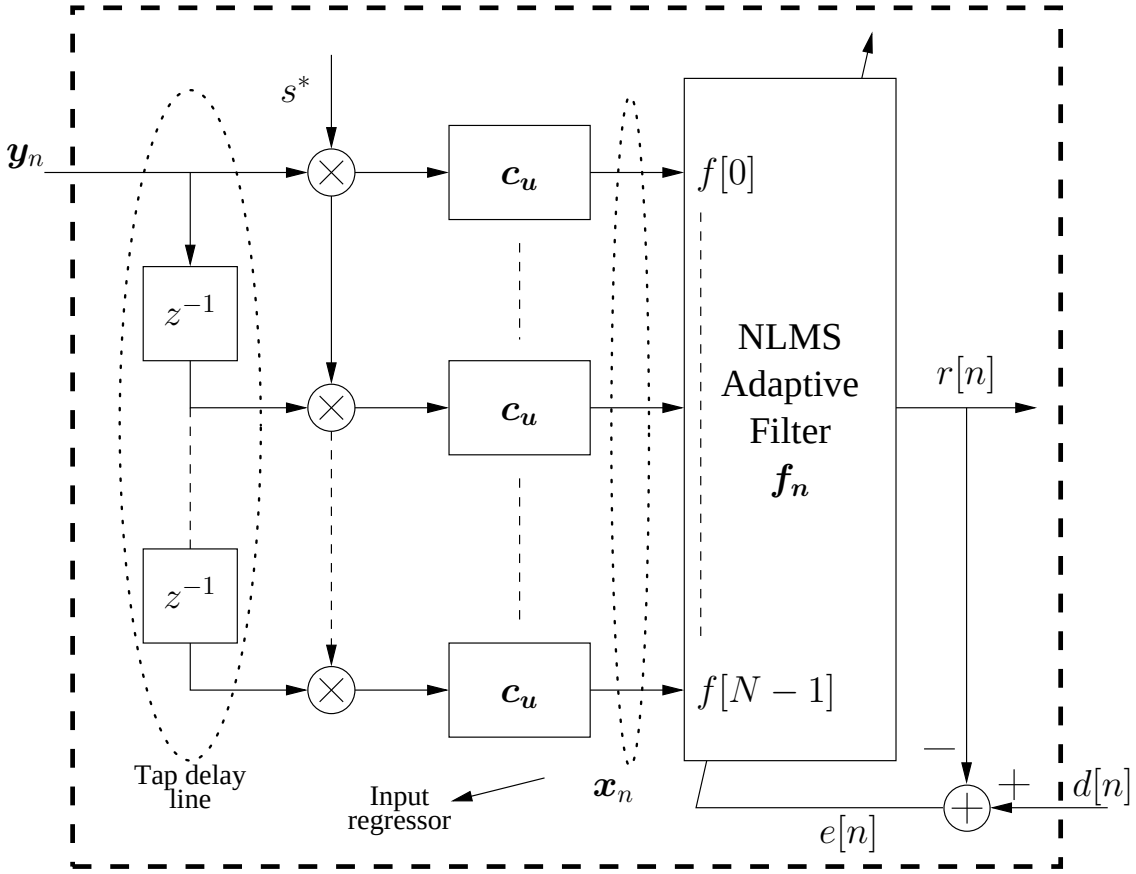


Figure 5.1: PCPICH Symbol Level NLMS Equalizer for WCDMA Downlink, $\boldsymbol{c}_u = \boldsymbol{c}_{256,0}$

Although this is quite an effective method in cases where the pilot code length is short, in UMTS FDD downlink, due to the long PCPICH despreading operation, adaptation can only be considered once every 256 chips. This drives the technique to be slow in both converging and tracking the highly time-varying channels [59, 72, 56, 58].

One possibility to increase the tracking capability of the scheme could be to do partial despreading, i.e. despreading with a parent code of PCPICH. However, if done so, unfortunately not only the PCPICH is partially despread but also the PCCPCH and all the child codes of that parent code. Therefore the amount of interference might even get amplified instead of being suppressed depending on whether there is significant amount of activity under the subtree originating from the chosen parent code or not.

In this section we propose a decision-directed receiver architecture which partially suppresses the interference coming from PCCPCH and other child codes and hence enables adapting the equalizer weights at higher rates than PCPICH symbol rate. The proposed receiver is all-time applicable to any WCDMA system where multi-access is realized via OVSF codes like the FDD downlink. Therefore, it can serve as a backup solution to the HDD-NLMS equalizer that we covered in Chapter 4 during the inactive periods when the mobile does not receive HSPDSCHs or as an eye-opener in 16-QAM mode.

We also elaborate on some possible generalizations and improvement possibilities of the scheme via the usage of side information like the knowledge of multiple HSPDSCH codes during the reception of HSDPA service. In fact the HDD-NLMS equalizer is a particular case of the category of schemes explained in this chapter when the parent code is chosen to be the root of the OVSF tree.

5.2 DD-NLMS Equalizer by Group Despreading

A basic schematic of the standard pilot-aided *symbol level* (N)LMS scheme is shown in Figure 5.1 [72]. This architecture is made up of a tap delay line, a descrambler and a despreader with the PCPICH channelization code $\boldsymbol{c}_{256,0}$ on each branch originating from the associated taps, a filter of length equal to the number of taps N and the standard LMS adaptation scheme. The adaptive filter input $\boldsymbol{x}_n$ serves as the input regressor, the filter output $r[n]$ is the PCPICH symbol estimate, $d[n]$ is the desired signal, i.e. the correct PCPICH symbol and $e[n]$ is the error signal. Adaptive filtering algorithm is the standard NLMS algorithm that trains via $\boldsymbol{x}_n$, $r[n]$, $e[n]$ and $d[n]$ [62]. Here we show the chip spaced implementation [72]. Fractionally spaced scheme is a straightforward extension [56].

Partial-despreading replacement of the full despreading over the PCPICH code corresponds to changing $SF = 256$ with for example $SF = 16$ when we effectively despread with the parent code $\boldsymbol{c}_{16,0}$. Doing so we obtain 16 consecutive $r[n]$ filter outputs during one PCPICH period of 256 chips and hence we can adapt 16 times more[1]. However, as explained before, this operation also despreads the PCCPCH and all the other active child codes of the parent code $\boldsymbol{c}_{16,0}$. This situation is demonstrated in Figure 5.2. The code subspace shown by the enclosed dashed region might contain some possible active codes with significant interfering power.

[1] n is the index of partially despread symbols

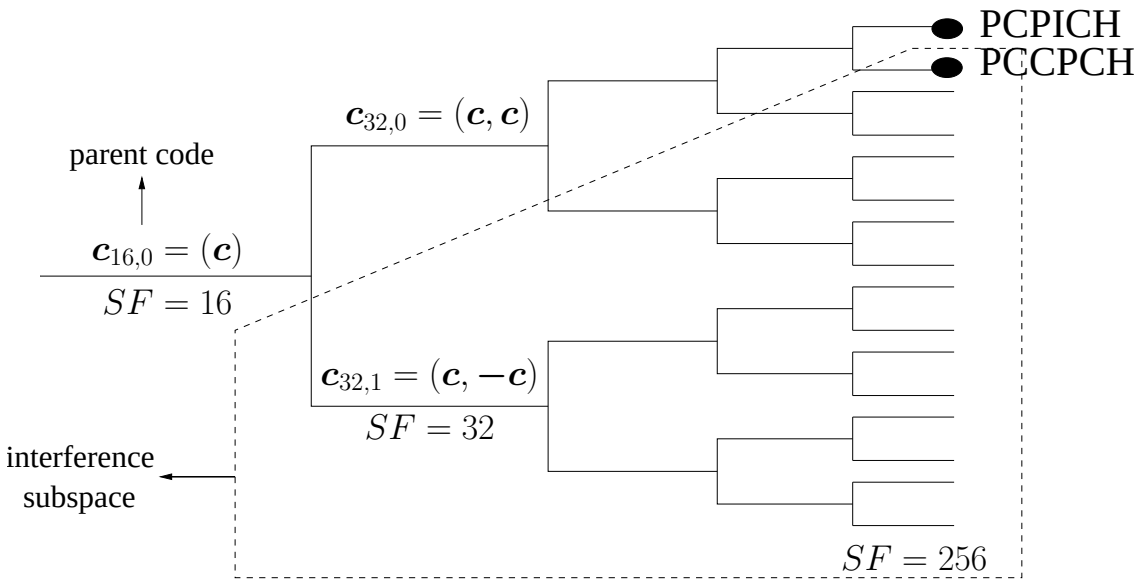


Figure 5.2: OVSF subtree rooted from $\boldsymbol{c}_{16,0}$ code

The total partially despread signal is equal to

$$r[n] = cp_{partial}[n] + pcch_{partial}[n] + \sum_i d_{i\_partial}[n] + v[n] \tag{5.1}$$

where $cp_{partial}[n]$ is the partially despread PCPICH symbol which serves as the desired response, $pcch_{partial}[n]$ is the partially despread interfering PCCPCH symbol, $\sum_i d_{i\_partial}[n]$ represents the partially despread interfering symbols from the other active codes in the interference subspace shown in Figure 5.2 and $v[n]$ is the sum of interference-plus-noise due to the combined effects of the multipath, the intercell interference and the thermal noise.

The drawback with the explained approach is clear. The original goal is to have only $cp_{partial}[n] + v[n]$ at the filter output, so that $cp_{partial}[n]$ serves as the desired response $d[n]$ and $v[n]$ serves as the error signal $e[n]$. However, unfortunately, $pcch_{partial}[n]$ and $\sum_i d_{i\_partial}[n]$ bring additional interference.

Our proposal to remedy the situation is based on the idea of recovering $pcch_{partial}[n]$ and $\sum_i d_{i\_partial}[n]$ from the interfering signals category as much as possible and putting them into the desired signal category. This is done by obtaining the sum of the estimates $\widehat{pcch}_{partial}[n]$, $\sum_i \hat{d}_{i\_partial}[n]$ and the partial PCPICH symbol estimate $\widehat{cp}_{partial}[n]$ as the desired signal estimate $\hat{d}[n]$ by an algorithm which is schematically demonstrated in Figure 6.2.

The algorithm runs on consecutive packets of received data spanning the elements associated with every PCPICH transmitted symbol. The processing window moves 256 chips in a sliding window manner for the next packet processing. For the *data filtering*, the adapted filter weights from the previous packet are used. For the initial packet filter weight assignment, we use the filter weights from the conventional Rake receiver. Below we explain the two phases of the algorithm in the *processing window with index n*.

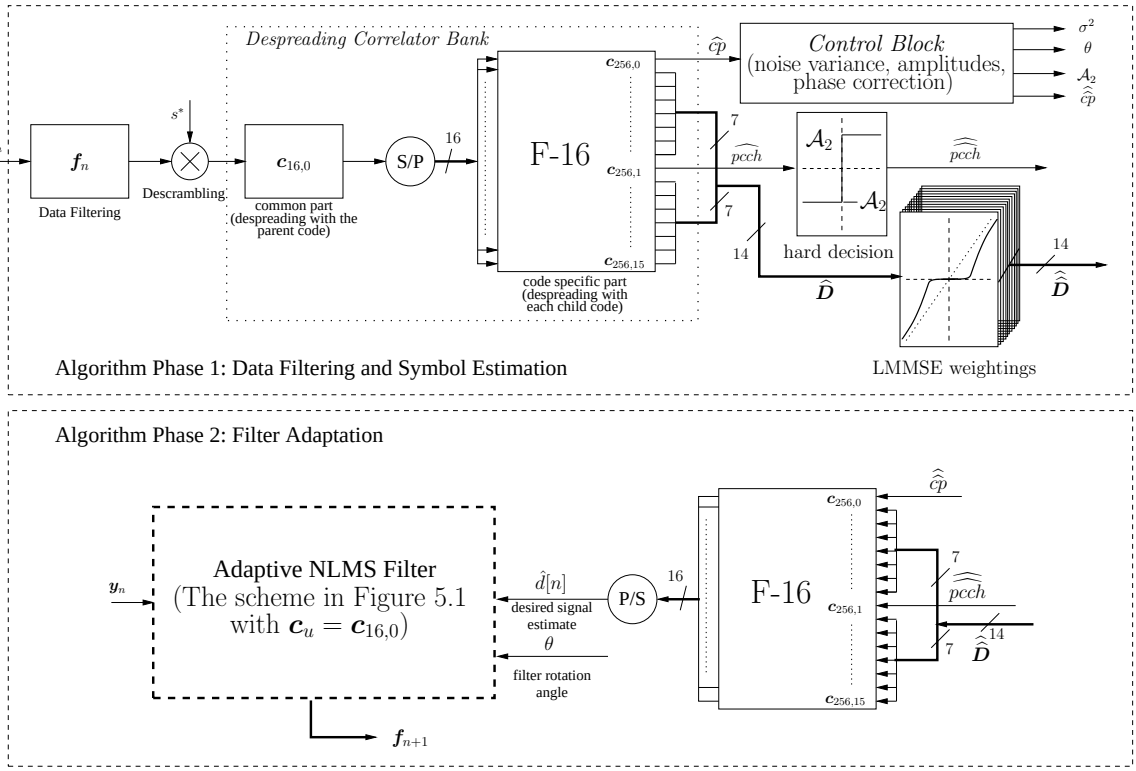


Figure 5.3: Decision-Directed NLMS Equalizer by Group Despreading

Algorithm Phase 1: Estimation of (Pseudo-)Symbols at SF-Level 256

As shown in the phase 1 part of Figure 6.2, we first pass the received chip rate signal through the FIR filter whose weights we obtain in the phase 2 part in the previous PCPICH symbol period. Then we descramble and despread with the 16 child codes of $\boldsymbol{c}_{16,0}$ at SF-level 256, i.e. $\{\boldsymbol{c}_{256,0}, \boldsymbol{c}_{256,1}, \ldots, \boldsymbol{c}_{256,15}\}$.

For the multiple despreading operations we apply a two-stage procedure. First we despread with the code of the parent code $\boldsymbol{c}_{16,0}$ and then do the second step multiple

correlations jointly via Fast Walsh Hadamard Transformation of size 16, i.e. F-16. When despreading is done independently with the 16 codes $\{\boldsymbol{c}_{256,0}, \boldsymbol{c}_{256,1}, \ldots, \boldsymbol{c}_{256,15}\}$, then the complexity is $256 * 16 = 4096$ units. The complexity of N-length FWHT is $Nlog_2N$. Therefore by the method we use, the complexity decreases to $256 + 16log_216 = 320$ units. After despreading, the first correlator output is the PCPICH linear symbol estimate $\widehat{cp}$, the second correlator output is the PCCPCH symbol estimate $\widehat{pcch}$ that appears at the 9^{th} place in the FWHT output which is just an output order difference from the OVSF code ordering and the rest $\widehat{\boldsymbol{D}}$ are the pseudo-symbol estimates reflected from any active code in the interference subspace indicated in Figure 5.2. S/P block represents the serial to parallel conversion operation.

Once the 16 linear (pseudo-)symbol estimates are obtained, we refine their quality by post processing elements. For the pilot element PCPICH, the symbol value is known. It is $1 + j$. However its real/imaginary amplitude is not known. This amplitude $\mathcal{A}_1$ and the symbol error variance σ^2 are estimated in the "Control Block" similar to the mechanism in Chapter 4. Once $\mathcal{A}_1$ is known, we have the final estimate for the PCPICH as $\widehat{cp} = \mathcal{A}_1(1+j)$. Similarly $\mathcal{A}_2$ is estimated by exploiting the knowledge of σ^2 and $\widehat{pcch}$ as $\mathcal{A}_2 = \sqrt{\frac{\left|\widehat{pcch}\right|^2 - \sigma^2}{2}}$. Knowing $\mathcal{A}_2$ we do hard decisions over the linear PCCPCH symbol estimates $\widehat{pcch}$ in order to obtain the refined estimates $\widehat{\widehat{pcch}}$. For each of the remaining codes we similarly estimate the received powers as $P_i = \left|\hat{d}_i\right|^2 - \sigma^2$ and we do linear minimum mean square error (LMMSE) weighting on them as $\widehat{\hat{d}}_i = \frac{P_i}{P_i + \sigma^2}\hat{d}_i$.

Algorithm Phase 2: Filter Adaptation

The final estimates of the Phase 1, i.e. $\{\widehat{\widehat{cp}}, \widehat{\widehat{pcch}}, \widehat{\widehat{\boldsymbol{D}}}\}$ are put to the corresponding F-16 ports and respread to SF-level 16 to obtain the 16 consecutive desired signal estimates in one PCPICH symbol period. These 16 values are used to adapt the NLMS equalizer weights 16 times in the same packet interval. Final equalizer weights serve as the next packet filter $\boldsymbol{f}_{n+1}$. P/S block represents parallel to serial conversion.

Similar to Chapter 4 we apply a misconvergence avoidance mechanism by using the estimated angle of the PCPICH symbol every 5 or 10 PCPICH symbol periods.

The proposed equalizer algorithm is applicable for both software and hardware implementations. In both cases, however, an input buffering of at least 256 chips is necessary and the total processing of the two stages must be completed in one PCPICH symbol period.

Therefore chip level processing should be done at least twice the chip rate, meaning that the hardware clock or software processor should work at least with the speed of $2 * 3.84$ Mhz $= 7.68$ Mhz.

Our treatment was using the pseudo-symbol of $\boldsymbol{c}_{16,0}$ as the desired signal for the NLMS algorithm via doing decision direction through the (pseudo)-symbols of 16 codes at SF-256. However this is only an example. It can be trivially generalized for the parent codes at other spreading factors. Higher spreading factors such as 32, 64 or 128, i.e. despreading with $\boldsymbol{c}_{32,0}$, $\boldsymbol{c}_{64,0}$ or $\boldsymbol{c}_{128,0}$ would be more suitable for low SNR and low speed conditions whereas lower spreading factors such as 2,4,8 would be more suitable for high SNR and high speed conditions.

Generalization of the Scheme to Smaller Window Sizes in HSDPA Service

The explained concept can be generalized to the usage of other other code groups from the OVSF hierarchy. Especially the knowledge of codes in the case of High Speed Downlink Packet Access (HSDPA) service is an opportunity to use hard decisions as is done over the PCCPCH in Figure 6.2. Consider the following scenarios:

Deployment Scenario 1:

Say there are 8 existing HSPDSCHs occupying the codes $\{\boldsymbol{c}_{16,8}$, $\boldsymbol{c}_{16,9}$, $\boldsymbol{c}_{16,10}$, $\boldsymbol{c}_{16,11}$, $\boldsymbol{c}_{16,12}$, $\boldsymbol{c}_{16,13}$, $\boldsymbol{c}_{16,14}$, $\boldsymbol{c}_{16,15}\}$. This means that they have a common parent code $\boldsymbol{c}_{2,1}$. Therefore one can

i. set the window size of the explained algorithm to 16

ii. despread with the 8 HSPDSCH codes $\{\boldsymbol{c}_{16,8}$, $\boldsymbol{c}_{16,9}$, $\boldsymbol{c}_{16,10}$, $\boldsymbol{c}_{16,11}$, $\boldsymbol{c}_{16,12}$, $\boldsymbol{c}_{16,13}$, $\boldsymbol{c}_{16,14}$, $\boldsymbol{c}_{16,15}\}$. For this purpose one can first do despreading with the parent code $\boldsymbol{c}_{2,1}$ as the first common part and then do F-8 over 8 consecutive values. This is similar to what we have done in the despreading correlator bank in Figure 6.2.

iii. do hard decisions over the eight HSDPA soft symbols.

iv. Feed the 8 hard decision values to F-8 input and obtain desired signal d every 2 chips. The corresponding SF for the adaptive filtering part in Figure 6.2 will be 2 and the despreading code will be $\boldsymbol{c}_{2,1}$. Since the window size in such a scheme will be much

less than 256 (in this case 2), it will be much more robust than the original scheme against highly varying channel conditions. Because the latency for the usage of filter weights for the next window will be only 2 chips.

Deployment Scenario 2:

Say this time there are 4 existing HSPDSCHs occupying the codes $\{\boldsymbol{c}_{16,4}$, $\boldsymbol{c}_{16,5}$, $\boldsymbol{c}_{16,6}$, $\boldsymbol{c}_{16,7}\}$. This means that they have a common parent code $\boldsymbol{c}_{4,1}$. Therefore one can

i. set the window size of the explained algorithm to 16

ii. despread with the 4 HSPDSCH codes

iii. do hard decisions over these four HSDPA soft symbols.

iv. Feed the 4 hard decision values to F-4 input and obtain desired signal d every 4 chips. The corresponding SF for the adaptive filtering part in Figure 6.2 will be 4 and the despreading code will be $\boldsymbol{c}_{4,1}$.

Deployment Scenario 3:

Say this time there are 3 existing HSPDSCHs occupying the codes $\{\boldsymbol{c}_{16,4}$, $\boldsymbol{c}_{16,95}$, $\boldsymbol{c}_{16,6}\}$, i.e. different from Scenario 2, $\boldsymbol{c}_{16,7}$ does not carry HSDPA service . In this case as the single difference one can do the LMMSE weighting over the soft estimate of $\boldsymbol{c}_{16,7}$ as was done over the 14 codes at SF level 256 in Figure 6.2.

5.3 Simulations and Conclusions

Most of the simulation settings are similar to Chapter 4 settings. Table 5.1 shows only the differences.

Table 5.1: Additional or Different Simulation Settings

Additional Parameters	**Settings**
$\hat{I}_{or}/I_{oc}$	10dB
Additional interference power (E_{int}) power in the first subtree	0% or 20% of the BS
Equalizer adaptation rate	Once every 16 chips

On Figures 5.4 to 5.11 {DS, C, M} respectively denote the receivers {DD-NLMS with group despreading, CMF, Max-SINR}.

CMF serves as the SNR bound and Max-SINR receiver serves as the SINR bound. Therefore, for CMF and Max-SINR we assumed that we have ideal channel information.

For plotting convenience, we obtain BER results for each TTI, i.e. 3 UMTS slots, and we sample the instantaneously obtained SINR every PCPICH symbol period.

By comparing the results from $E_{int}/\hat{I}_{or} = 0$ and $E_{int}/\hat{I}_{or} = 0.2$ we conclude that there is significant performance degradation when there is an high amount of interference. This clearly shows the importance of hard decision operation for feedback. Still the performance with noisy situation is much better than the CMF.

The scheme gives satisfactory performance even at 120km/h mobile speed.

We are expecting better performance from HSPDSCH-aided generalized schemes discussed in the text which have not yet been tested. As an ideal solution for HSDPA service, one can start with the first explained PCPICH and PCCPCH aided scheme and once a certain equalization quality is achieved one can switch to the HSPDSCH codes aided schemes. It is also possible to multiplex several of such DD-NLMS schemes and switch from one scheme to another depending on the instantaneous

- knowledge of codes
- power in the known code domain
- power in the unknown code domain
- $\hat{I}_{or}/I_{oc}$
- mobile speed

Besides the possibility of being used as stand-alone solutions, Group Despread DD-NLMS methods can also perfectly serve as backup solutions for the HDD-NLMS scheme covered in Chapter 4 for initialization purposes.

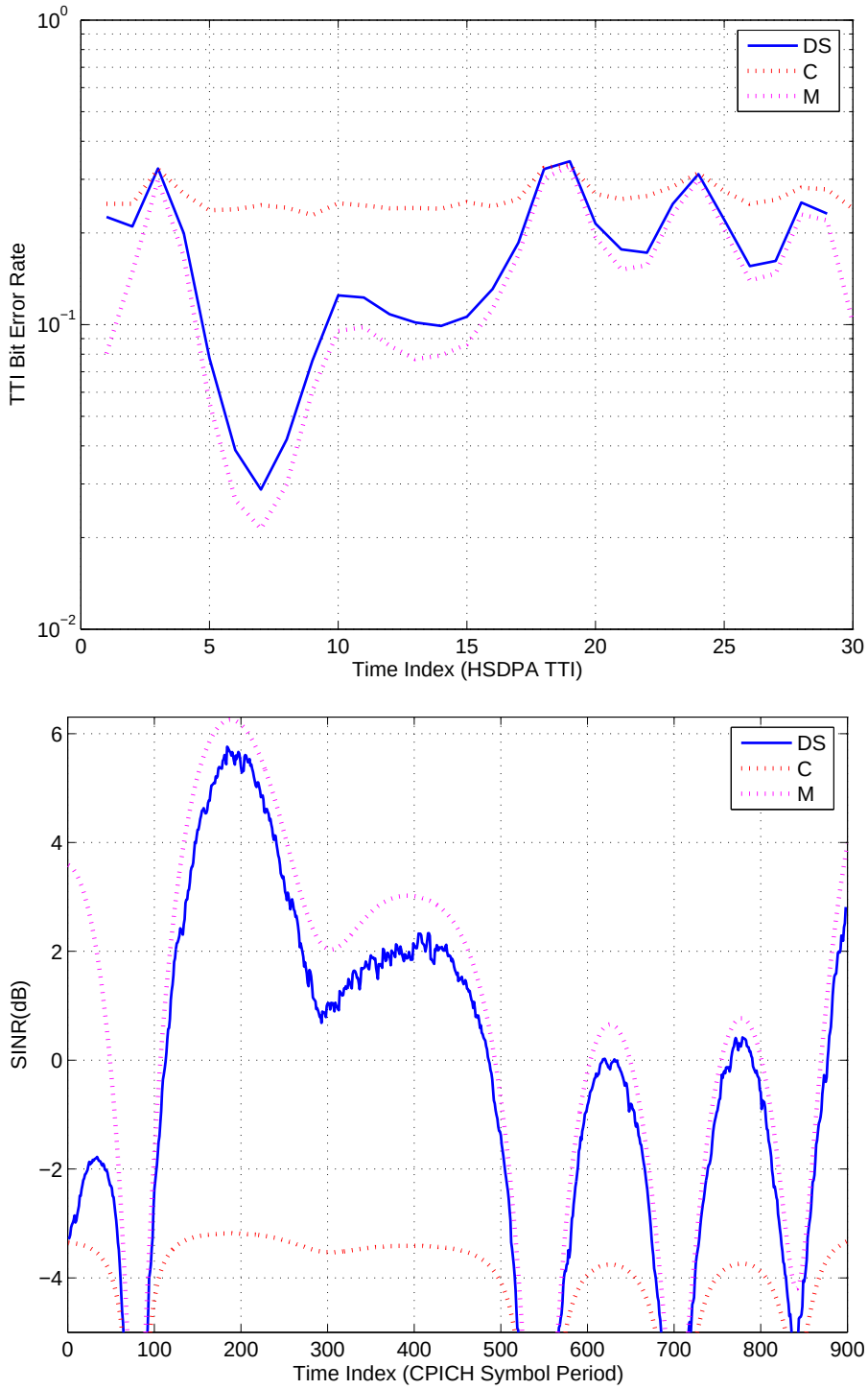


Figure 5.4: vA30 channel profile, $E_c/\hat{I}_{or} = -6dB$, $\hat{I}_{or}/I_{oc} = 10dB$, $E_{int}/\hat{I}_{or} = 0$

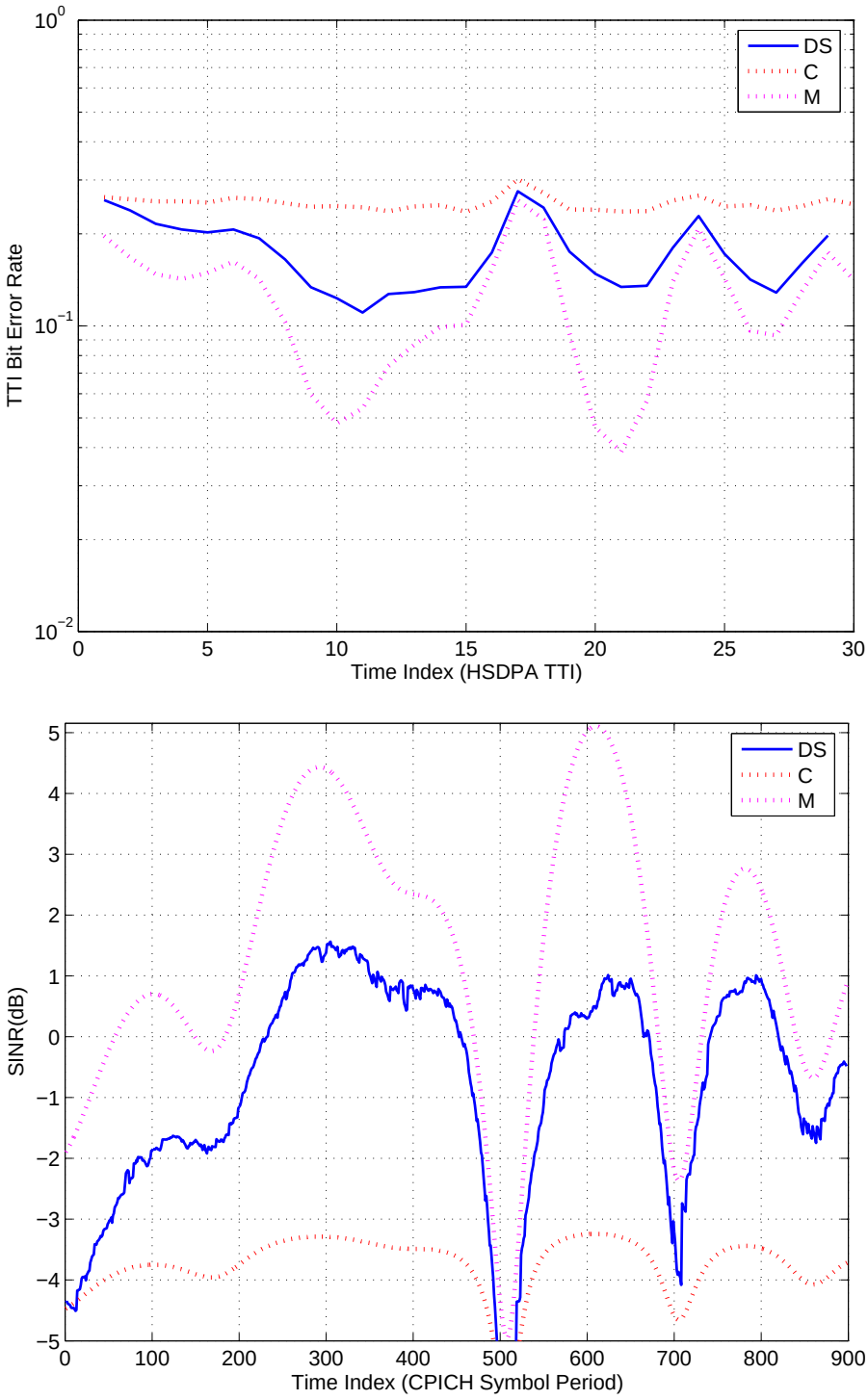


Figure 5.5: vA30 channel profile, $E_c/\hat{I}_{or} = -6dB$, $\hat{I}_{or}/I_{oc} = 10dB$, $E_{int}/\hat{I}_{or} = 0.2$

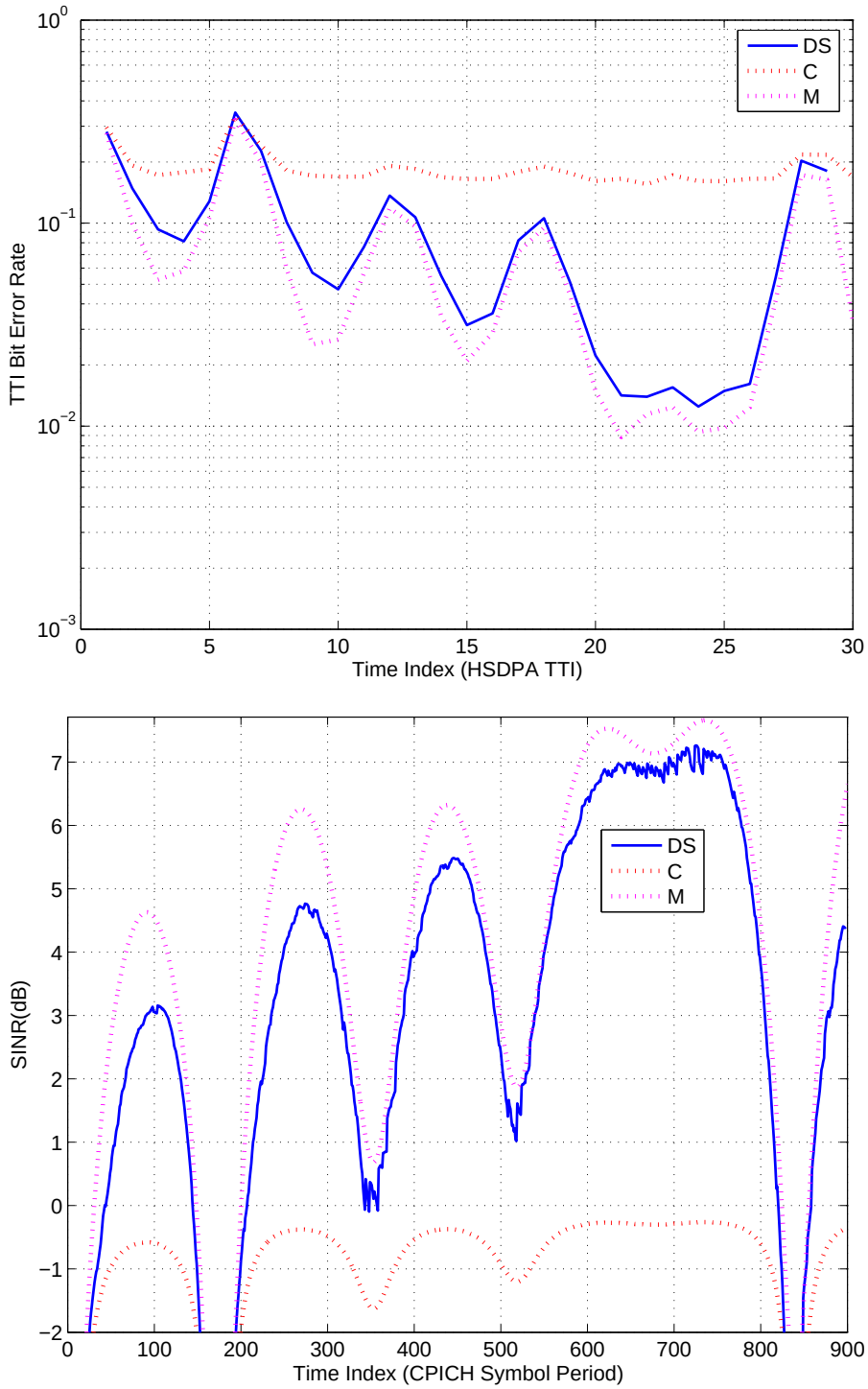


Figure 5.6: vA30 channel profile, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 10dB$, $E_{int}/\hat{I}_{or} = 0$

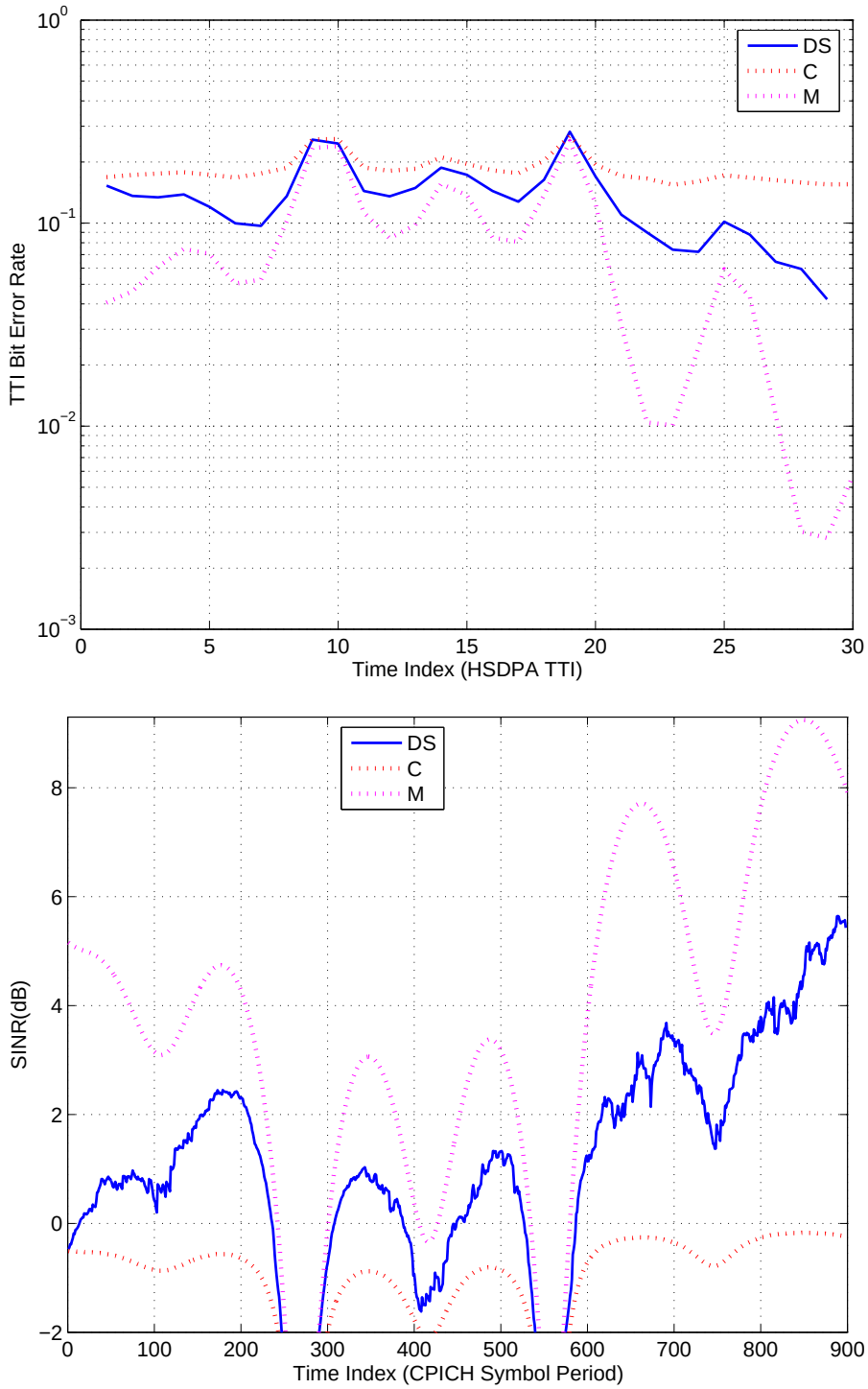


Figure 5.7: vA30 channel profile, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 10dB$, $E_{int}/\hat{I}_{or} = 0.2$

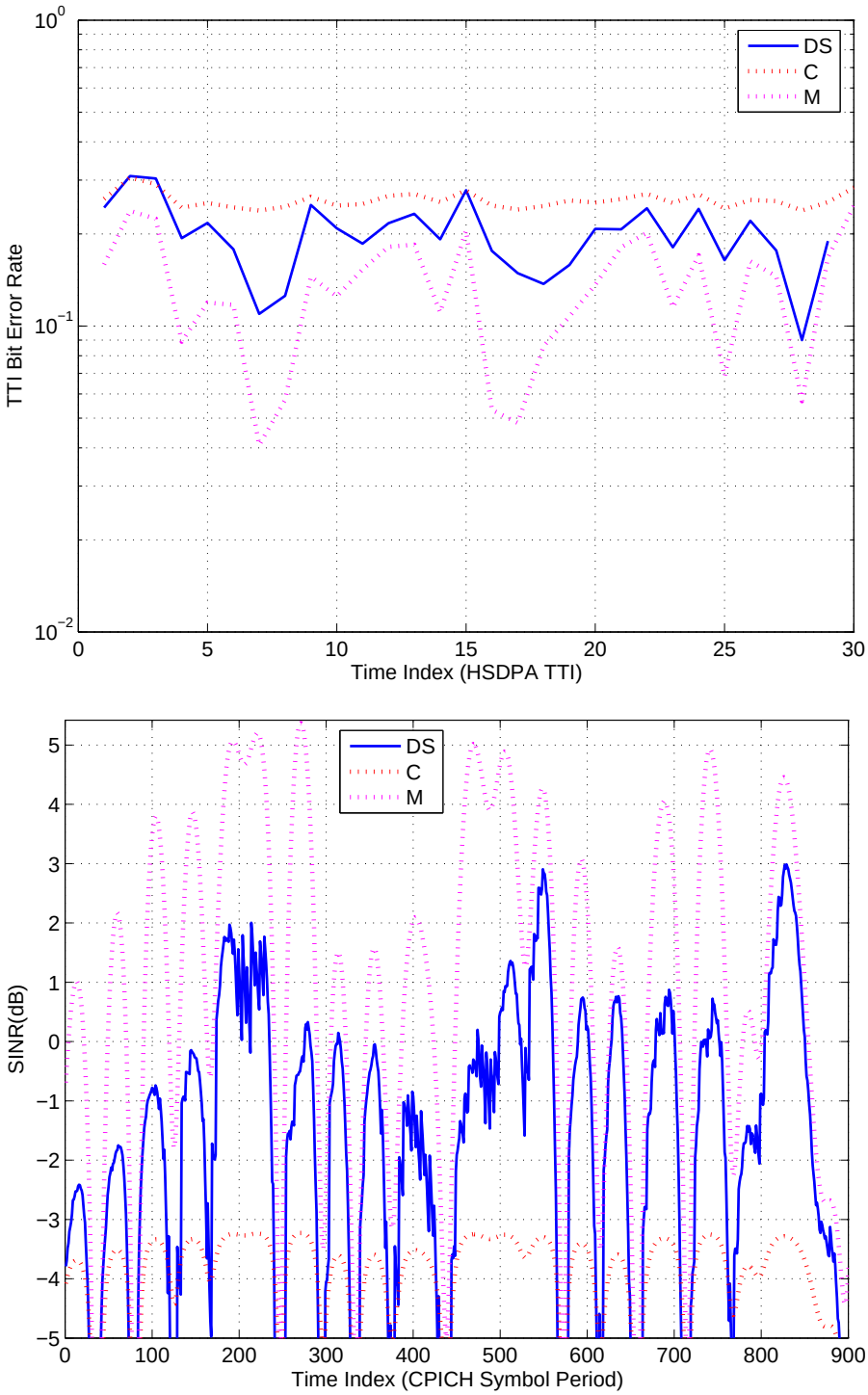


Figure 5.8: vA120 channel profile, $E_c/\hat{I}_{or} = -6dB$, $\hat{I}_{or}/I_{oc} = 10dB$, $E_{int}/\hat{I}_{or} = 0$

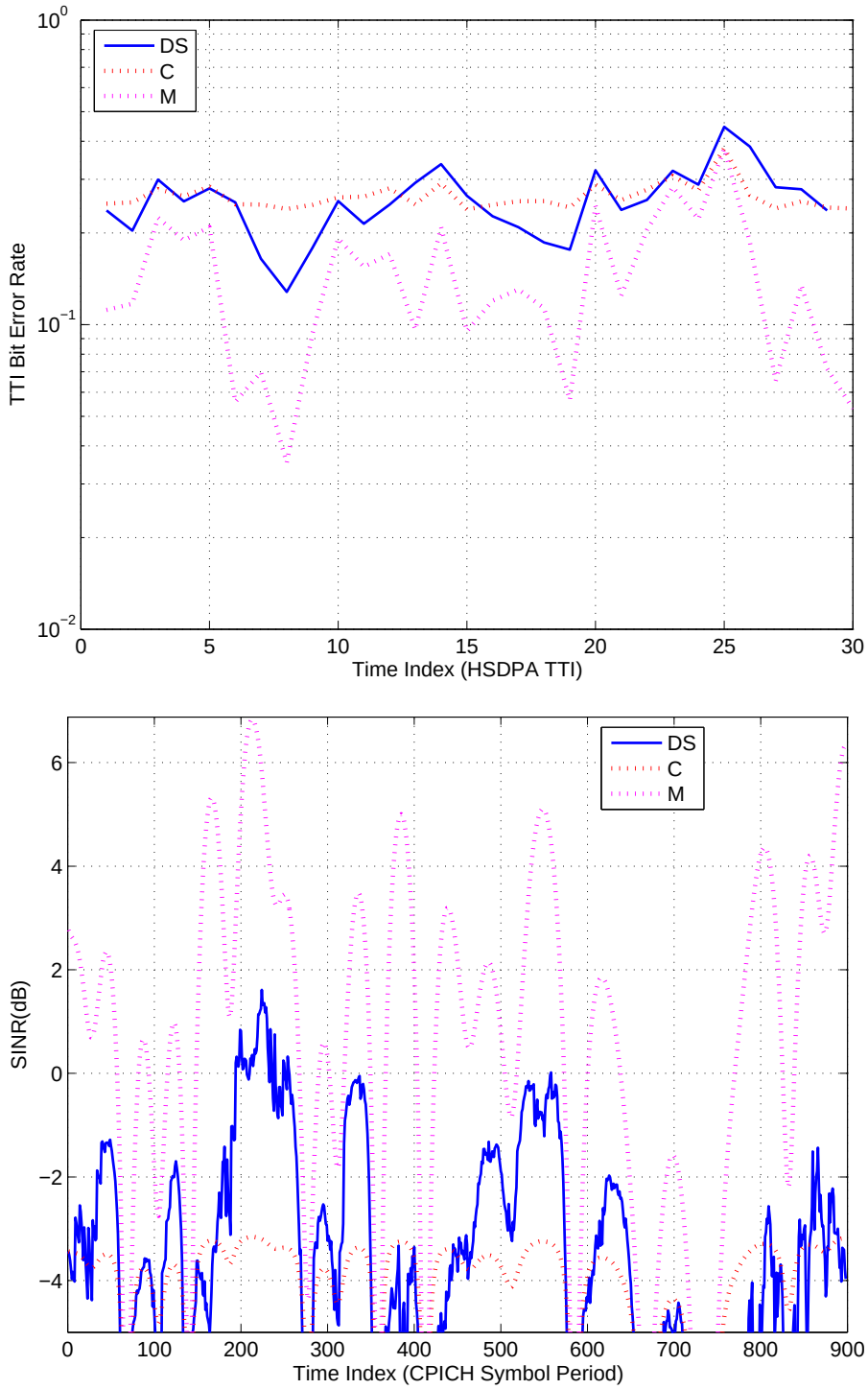


Figure 5.9: vA120 channel profile, $E_c/\hat{I}_{or} = -6dB$, $\hat{I}_{or}/I_{oc} = 10dB$, $E_{int}/\hat{I}_{or} = 0.2$

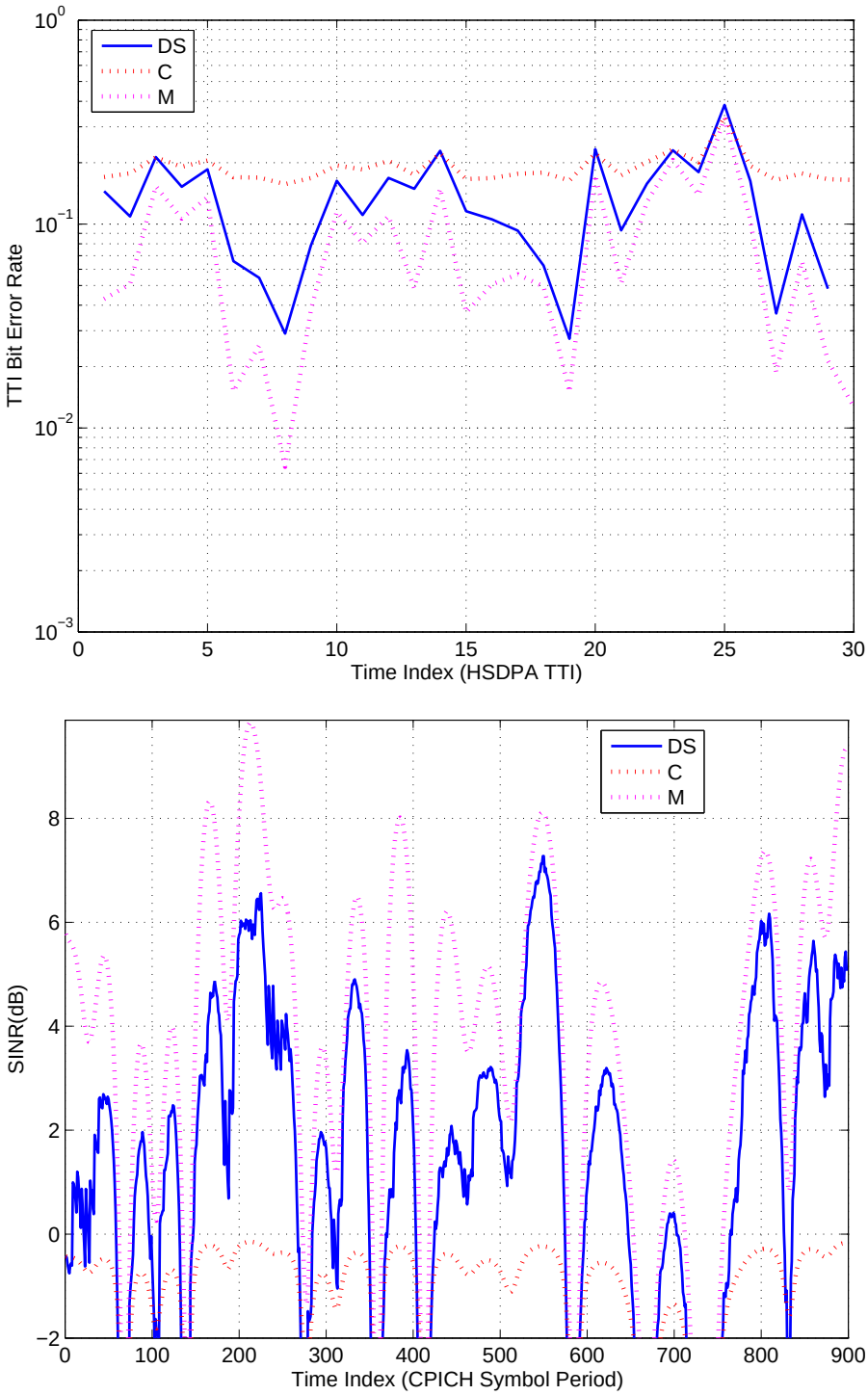


Figure 5.10: vA120 channel profile, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 10dB$, $E_{int}/\hat{I}_{or} = 0$

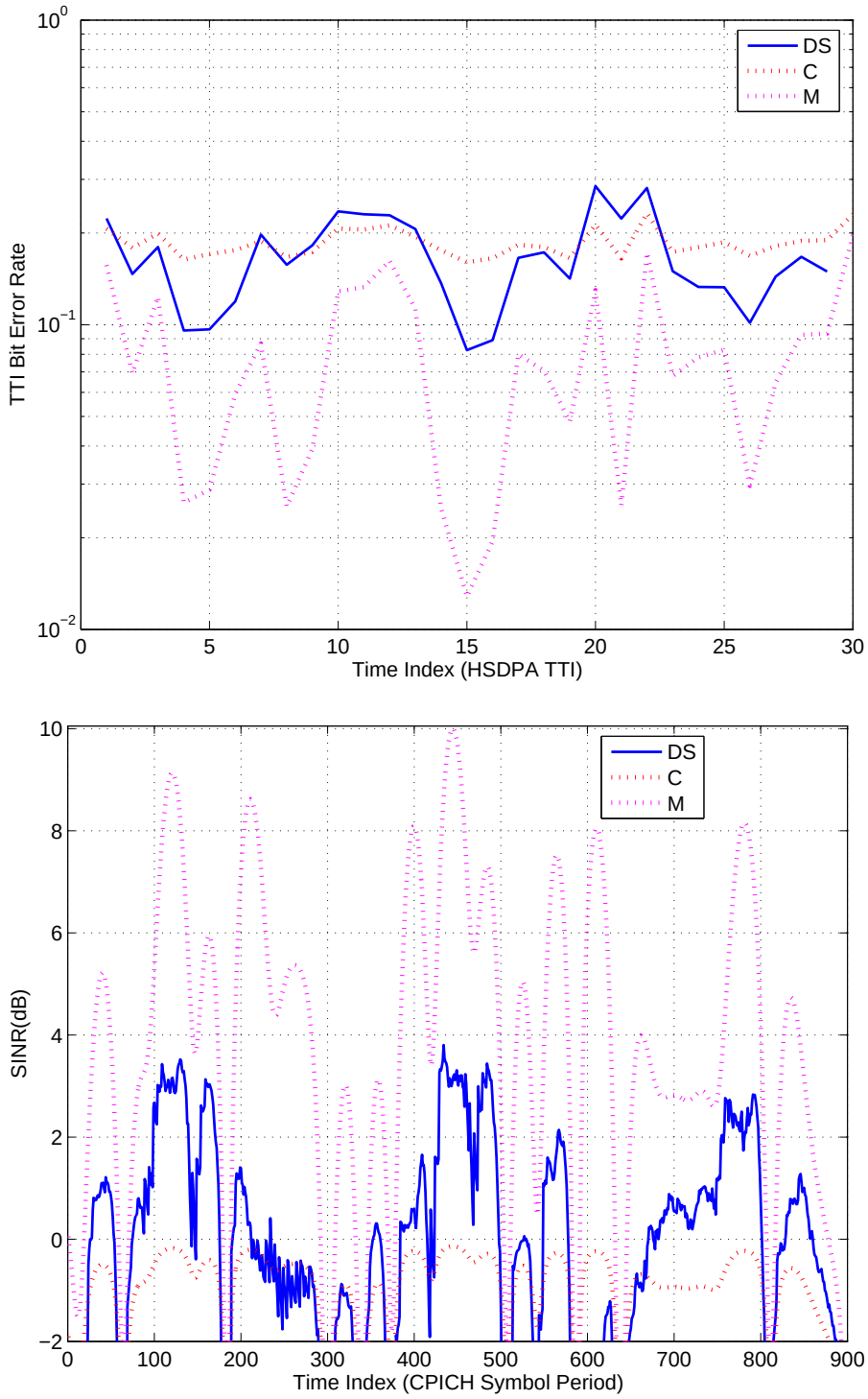


Figure 5.11: vA120 channel profile, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 10dB$,$E_{int}/\hat{I}_{or} = 0.2$

Chapter 6

Symbol Level Adaptive Equalization for HSDPA

Although a simple solution, pilot-symbols-trained LMS algorithm is not capable of tracking fast varying channels in UMTS FDD downlink due to insufficient adaptation rate. Pilot-chips-trained LMS adaptation is on the other hand much more prone to noise. These two phenomena manifest themselves in the two components of the adaptation excess mean square (EMSE). A compromise can be found by considering HSDPA symbol-level Griffiths or decision-directed equalization. These two methods enable adapting 16 times more frequently than the pilot-symbols-aided adaptation. They are also attractive for implementation since a modular approach can be adopted by exploiting either one or more of the available HSDPA code domains depending on the instantaneous channel quality and the performance requirements.

6.1 Introduction

In this Chapter we propose HSDPA symbol level adaptation schemes which benefit from the knowledge of multiple HSDSCH codes and their identical power and constellation properties.

6.2 Griffiths Equalization at HSDPA Symbol Level

Griffiths adaptation scheme is shown in Figure 6.1. The filtering delay is taken to be $N-1$ chips.

When only one HSDSCH code domain, say the first one, is used, Griffiths Equalization at HSDPA symbol level can be derived by again considering the Wiener solution as

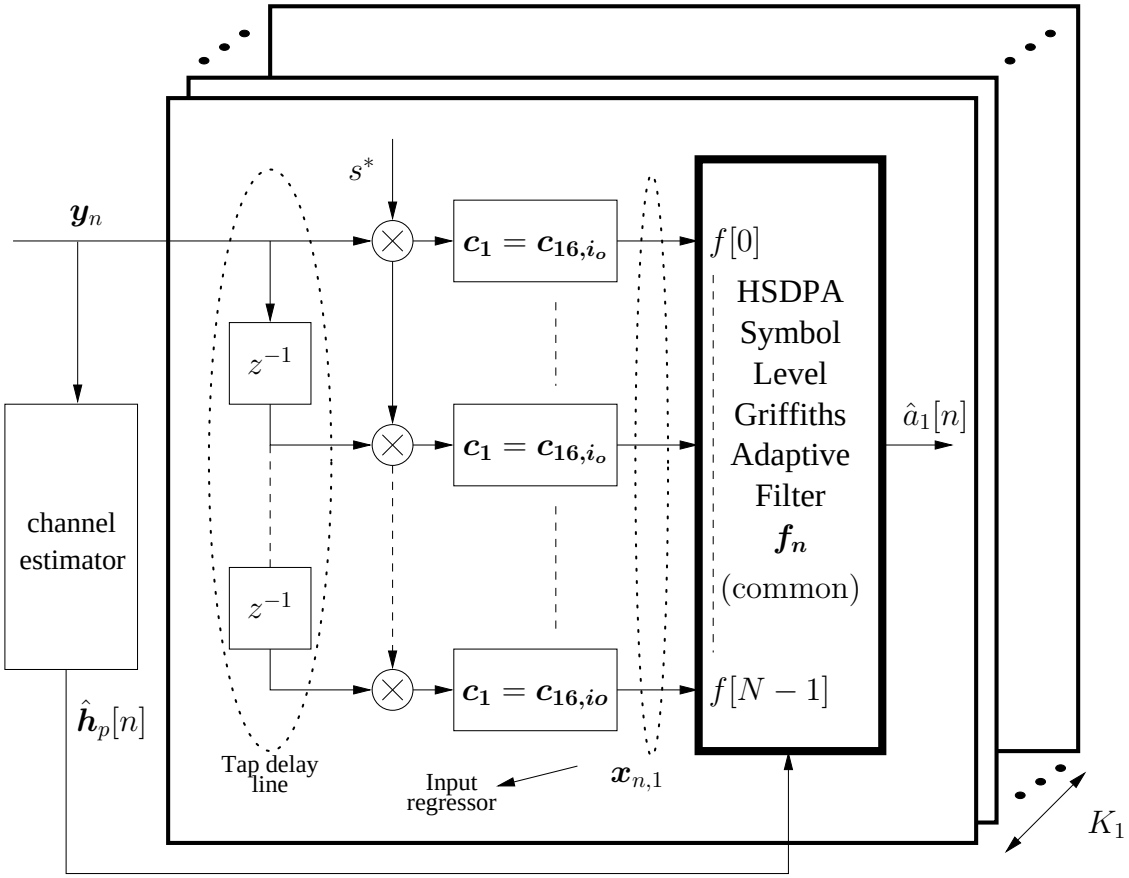


Figure 6.1: HSDPA Symbol Level N-Griffiths Equalizer

a starting point.

$$\tilde{\boldsymbol{f}}_o = \boldsymbol{R}_{a_1\boldsymbol{x_{n,1}}}\boldsymbol{R}^{-1}_{\boldsymbol{x_{n,1}}\boldsymbol{x_{n,1}}} \tag{6.1}$$

By making the connection between the cross-correlation term and the channel matched filter as

$$\boldsymbol{R}_{a_1\boldsymbol{x_{n,1}}} = \sigma^2_{a_1}\boldsymbol{h}^{\dagger}_p[n] \tag{6.2}$$

and replacing the regression vector covariance matrix by the instantaneous sample statistics, we reach to the Griffiths adaptation mechanism as

$$\begin{aligned}\boldsymbol{f}_{n+1} &= \boldsymbol{f}_n - \mu_{n,1}\nabla\boldsymbol{f}_{n,1} \\ &= \boldsymbol{f}_n - \mu_{n,1}(\boldsymbol{f}_n\boldsymbol{x}_{n,1}\boldsymbol{x}^H_{n,1} - \sigma^2_{a_1}\boldsymbol{h}^{\dagger}_p[n])\end{aligned} \tag{6.3}$$

where $\boldsymbol{f}_n$, $\mu_{n,1}$, $\nabla\boldsymbol{f}_{n,1}$, $\boldsymbol{x}_{n,1}$, respectively denote the filter weights, the step size, the MSE

gradient error vector and the input regression vector for the first code at the HSDPA symbol instant n.

The normalized form of the symbol level Griffiths adaptation scheme can be written as

$$\boldsymbol{f}_{n+1} = \boldsymbol{f}_n - \frac{\mu_{n,1}(\boldsymbol{f}_n \boldsymbol{x}_{n,1} \boldsymbol{x}_{n,1}^H - \sigma_b^2 \boldsymbol{h}_p^{\dagger}[n])}{\boldsymbol{x}_{n,1}^H \boldsymbol{x}_{n,1}} \tag{6.4}$$

6.3 Decision Directed Equalization at HSDPA Symbol Level

Decision-directed adaptation scheme is shown in Figure 6.2. All the amplitude estimation, step size adaptation, filter weights phase correction by the aid of pilot tone and the initial filter weights assignment procedures are similar to the explained ones in Chapter 4 and Chapter 5.

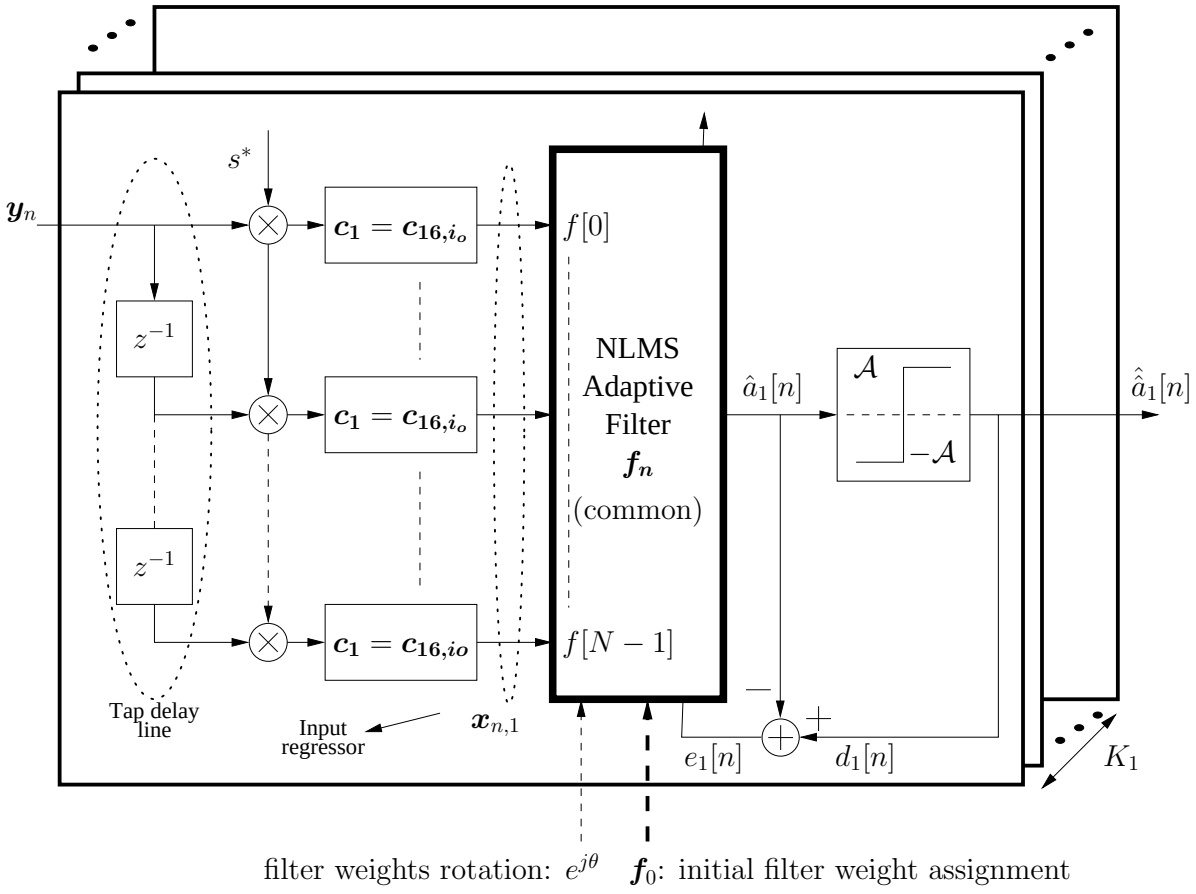


Figure 6.2: HSDPA Symbol Level Decision-Directed NLMS Equalizer

When again only one HSDSCH code domain is used, decision-directed LMS adaptation can be formulated as

$$\begin{aligned}\boldsymbol{f}_{n+1} &= \boldsymbol{f}_n - \mu_{n,1}\nabla\boldsymbol{f}_{n,1} \\ &= \boldsymbol{f}_n - \mu_{n,1}(\boldsymbol{f}_n\boldsymbol{x}_{n,1} - \hat{\hat{a}}_1[n])\boldsymbol{x}_{n,1}^H \\ &= \boldsymbol{f}_n + \mu_{n,1}e_1[n]\boldsymbol{x}_{n,1}^H \end{aligned} \tag{6.5}$$

where $\boldsymbol{f}_n$, $\mu_{n,1}$, $\nabla\boldsymbol{f}_{n,1}$, $\boldsymbol{x}_{n,1}$, $\hat{\hat{a}}_1[n]$ and $e_1[n]$ respectively denote the filter weights, the step size, MSE gradient error vector, input regression vector, the hard decided HSDSCH symbol which serves as the desired response and the error signal for the first code at HSDPA symbol instant n.

The normalized form of the symbol level DD-LMS adaptation scheme can be formulated as

$$\boldsymbol{f}_{n+1} = \boldsymbol{f}_n - \frac{\mu_{n,1}(\boldsymbol{f}_n\boldsymbol{x}_{n,1} - \hat{\hat{a}}_1[n])\boldsymbol{x}_{n,1}^H}{\boldsymbol{x}_{n,1}^H\boldsymbol{x}_{n,1}} \tag{6.6}$$

6.4 Extensions to Multiple Codes Usage

Existence of K_1 equal amplitude and equal power HSDSCH codes is an opportunity to adapt symbol level DD-LMS and symbol level Griffiths schemes by better gradient estimates as

$$\nabla\boldsymbol{f}_n = \sum_{i=1}^{K_1}\nabla\boldsymbol{f}_{n,i} \tag{6.7}$$

so that

$$\boldsymbol{f}_{n+1} = \boldsymbol{f}_n - \mu_n\nabla\boldsymbol{f}_n \tag{6.8}$$

This multi-code adaptation diversity comes from the fact that the input regression vectors $\boldsymbol{x}_{n,k}$, $k \in \{1, 2, \ldots, K_1\}$ at different code domains are uncorrelated.

The N-DDLMS and N-Griffiths adaptations are formulated as

$$\boldsymbol{f}_{n+1} = \boldsymbol{f}_n - \mu_n \sum_{i=1}^{i=K_1}\frac{\nabla\boldsymbol{f}_{n,i}}{\boldsymbol{x}_{n,i}^H\boldsymbol{x}_{n,i}} \tag{6.9}$$

The Mean Square Error (MSE) expression for (N)-LMS adaptation has two components: Minimum Mean Square Error (MMSE) and Excess Mean Square Error (EMSE) [62]. MMSE is the error floor performance of the Wiener Filter, which cannot be avoided. EMSE is the additional interference due to imperfect adaptation. It also has two ingredients as the

stochastic gradient noise due to only one instantaneous sample support for obtaining the required adaptation statistics and the lag noise due to the time variation of the channel. Using K_1 codes decreases the stochastic gradient noise K_1 times if step size is chosen as $\mu_n = \frac{\mu_{n,1}}{K_1}$. The optimal step size compromising the two ingredients of EMSE would be a value $\frac{\mu_{n,1}}{K_1} < \mu_n < \mu_{n,1}$, exact value depending on the mobile speed, i.e. the time variation of the channel and the instantaneous noise level.

6.5 Simulations and Conclusions

We look at the BER and SINR performances of the proposed HSDPA symbol level adaptation schemes with the same simulation settings used in Chapter 4.

On Figures 6.5 to 6.10 {D, C, SD, SGL, SGM, SGH, M, GL, GM, GH, P} respectively denote {DD-NLMS with all codes, CMF, one-code DD-NLMS, one-code Griffiths with a low quality channel estimator, one-code Griffiths with a medium quality channel estimator, one-code Griffiths with a high quality channel estimator, Max-SINR, Griffiths with all codes with a low quality channel estimator, Griffiths with all codes with a medium quality channel estimator, Griffiths with all codes with a high quality channel estimator, PCPICH symbol level NLMS}.

As was the case in the simulation results for chip level N-Griffiths in Chapter 4, the HSDPA symbol-level N-Griffiths schemes are also robust against channel estimation errors, hence there is negligible difference among the results obtained from different quality channel estimation supports.

Although PCPICH symbol level NLMS scheme performs better than CMF at 30km/h UE speed, it performs much worse at 120km/h. The crossover UE speed when CMF starts performing better is about 55km/h[1].

When all the HSDSCH code domains are exploited for adaptation, similar to chip level adaptation schemes in Chapter 4, DD-NLMS performs better than N-Griffiths in most of the cases, approaching the Max-SINR performance at high SNR regions but performs worse in low SNR regions. When only one code is used, however, N-Griffiths showed better speed of convergence characteristics than DD-NLMS. One can attribute this to the fact that the cross-correlation term in the adaptation of the N-Griffiths scheme is common for all the gradient vectors. Therefore there is more correlation between N-Griffiths filter update

[1] not shown in simulations

vectors and hence less difference between one code and multiple codes adaptation qualities, which in fact makes one code N-Griffiths implementation a very attractive solution due to its low complexity.

An ideal implementation strategy would be to adapt also the number of used codes and even more to switch between N-Griffiths and DD-NLMS schemes depending on the channel conditions, in particular N-Griffiths serving as an eye-opener for N-DDLMS. A more costly solution would be to concurrently run the N-Griffiths and DD-NLMS schemes and update filter weights by a weighted combination of their gradient vectors. This would in particular be useful with 16-QAM modulation. Such methods have been especially considered in literature between constant modulus (CM) and DD-LMS schemes, CM taking the role of the eye-opener [67].

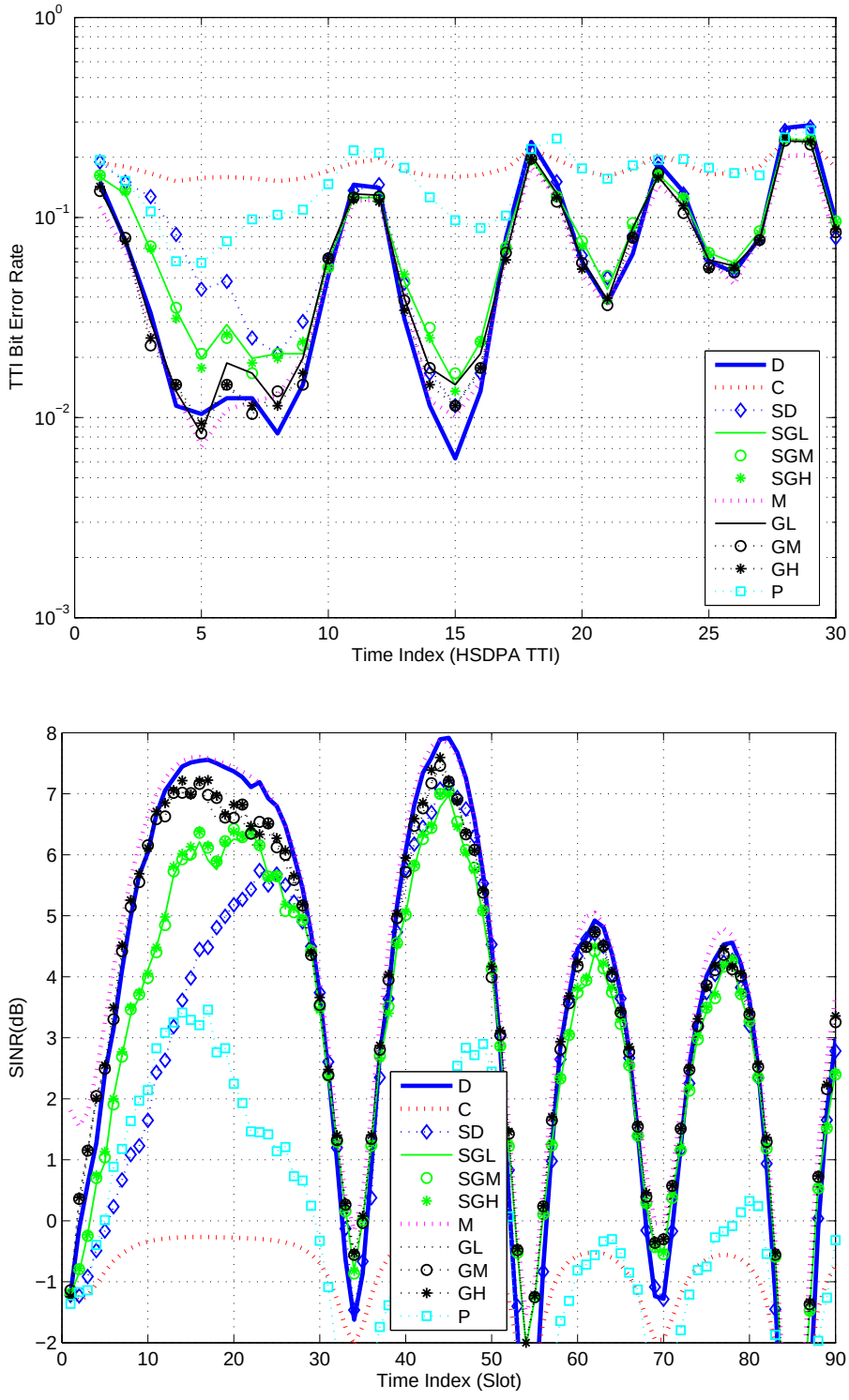


Figure 6.3: vA30 channel profile, 10 codes, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 10dB$

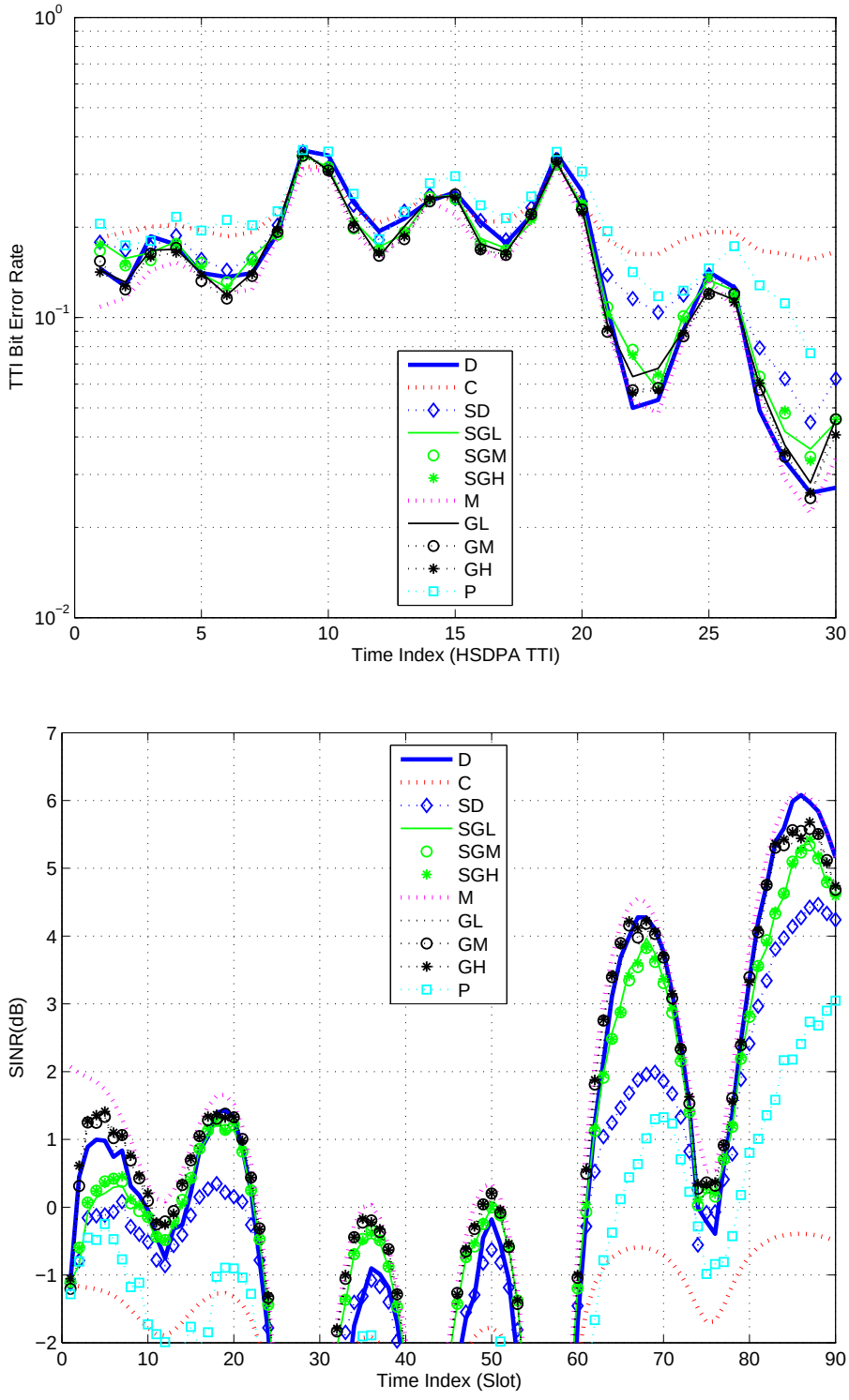


Figure 6.4: vA30 channel profile, 10 codes, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 6dB$

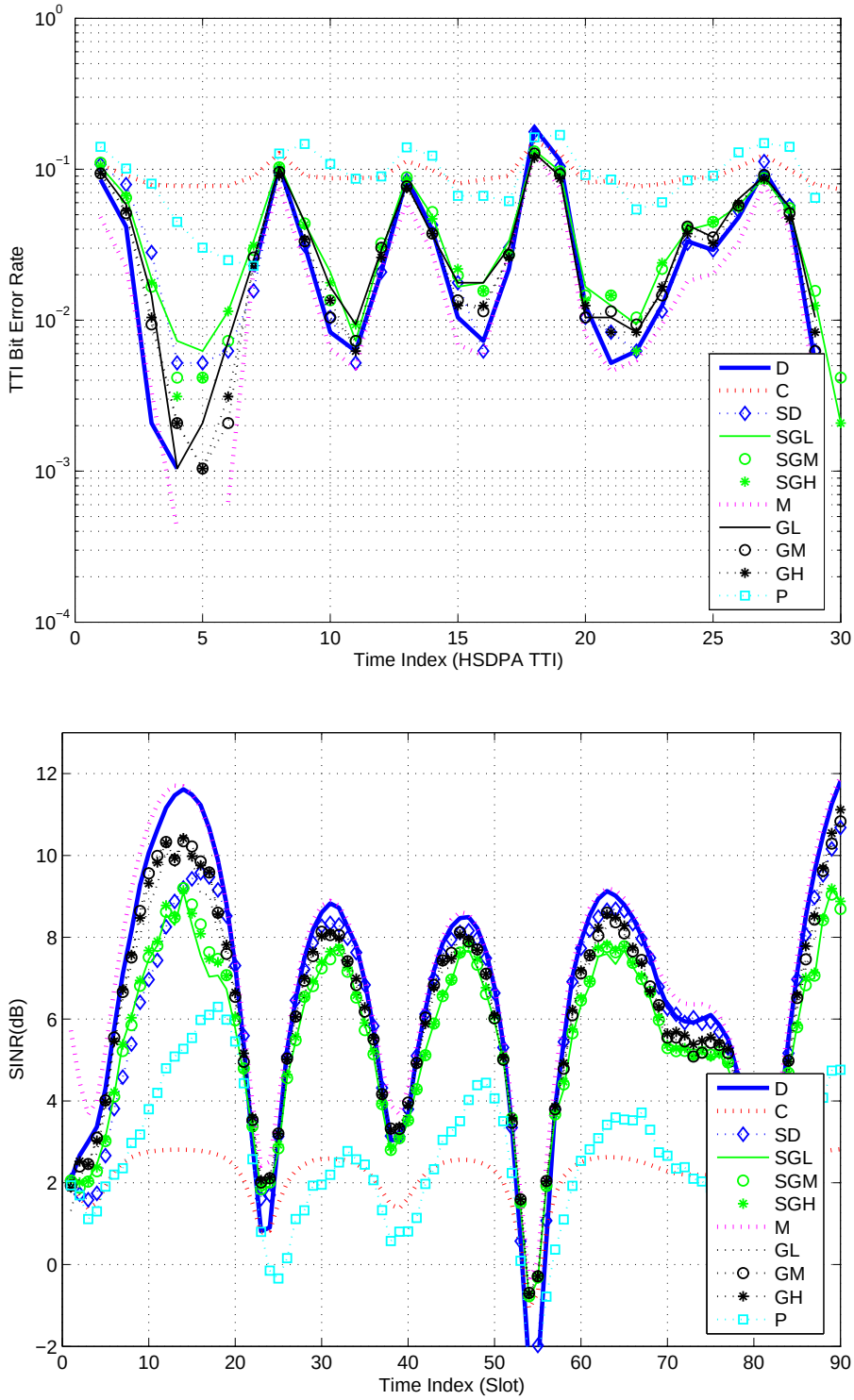


Figure 6.5: vA30 channel profile, 5 codes, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 10dB$

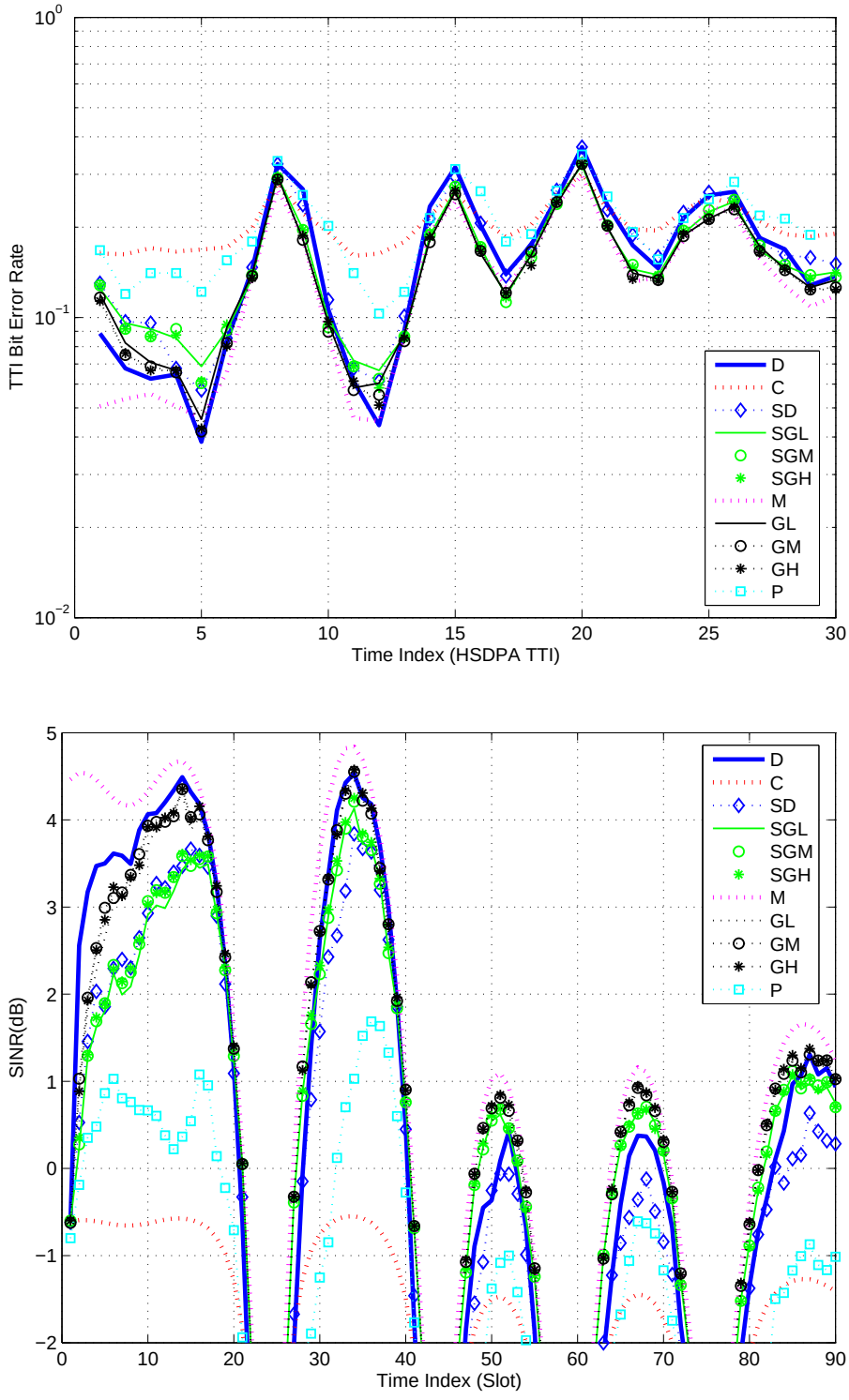


Figure 6.6: vA30 channel profile, 5 codes, $E_c/\hat{I}_{or} = -6dB$, $\hat{I}_{or}/I_{oc} = 6dB$

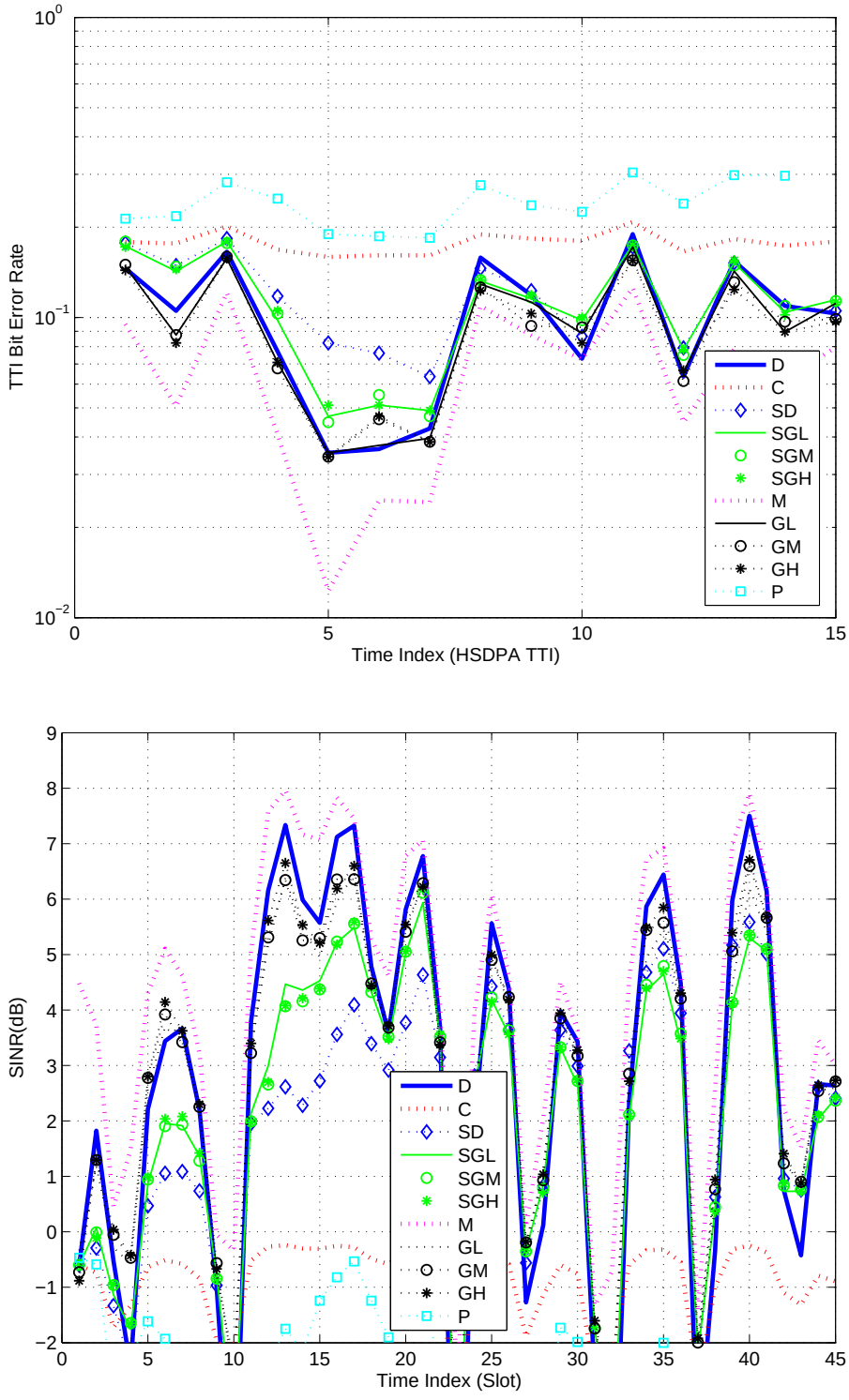


Figure 6.7: vA120 channel profile, 10 codes, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 10dB$

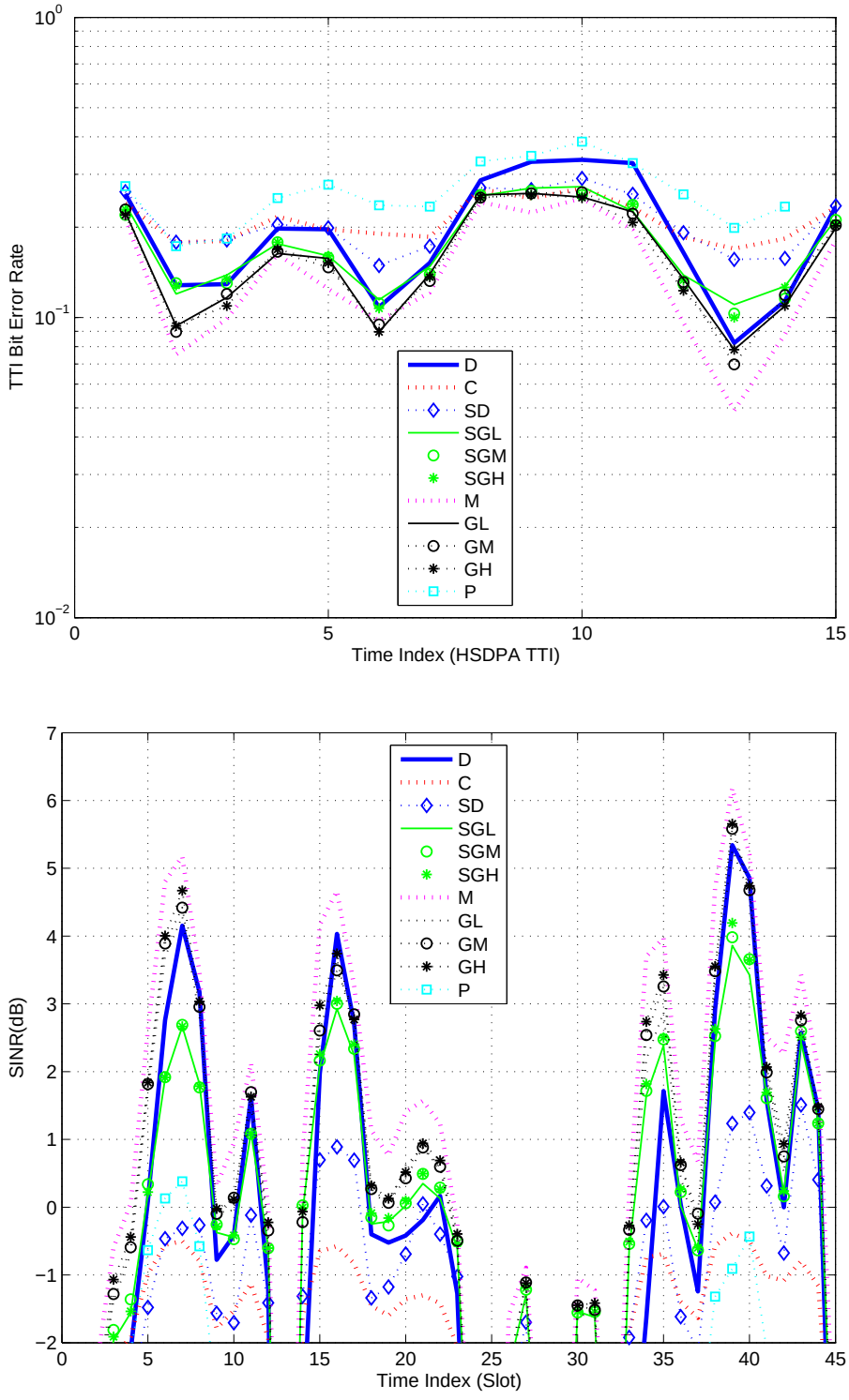


Figure 6.8: vA120 channel profile, 10 codes, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 6dB$

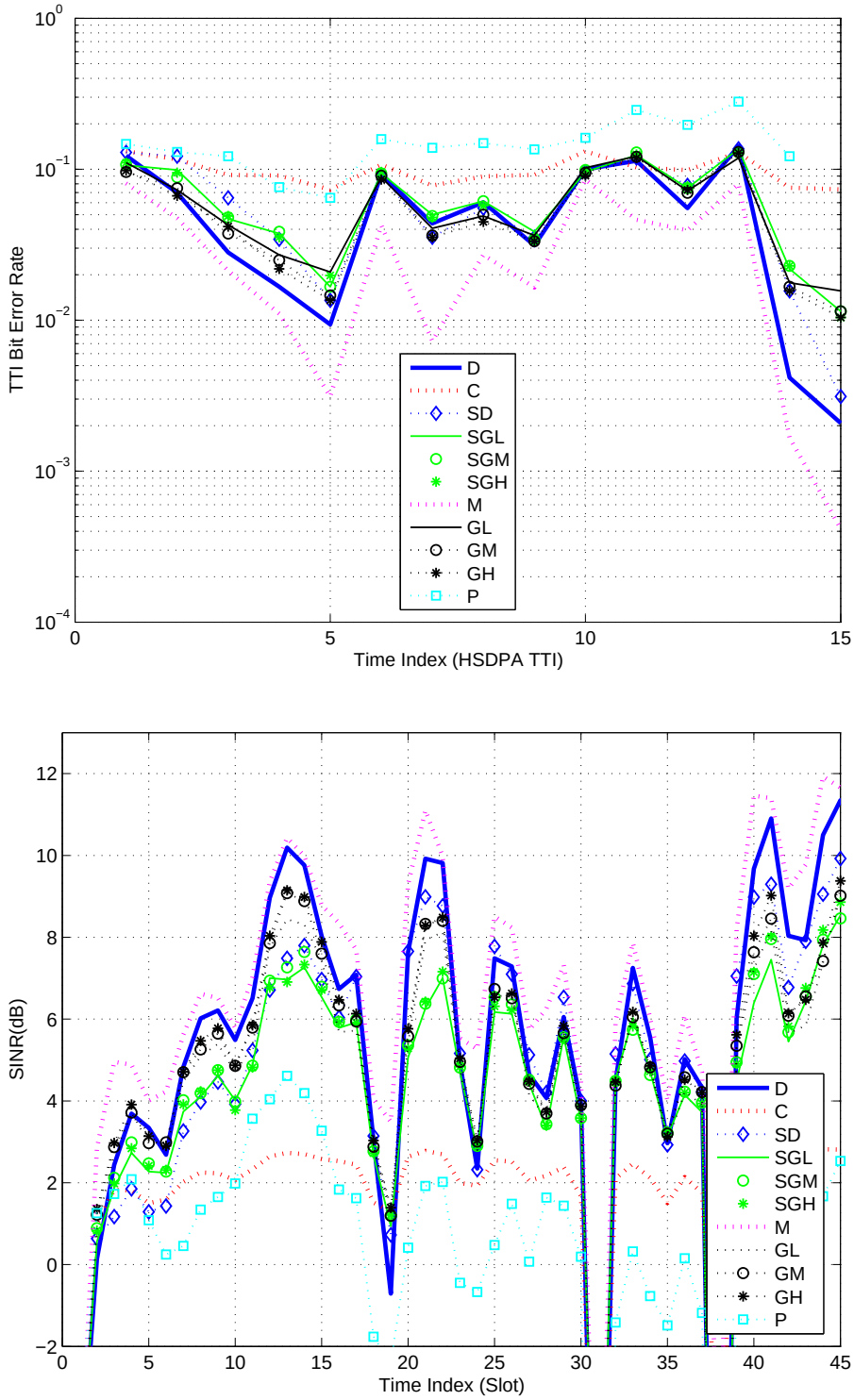


Figure 6.9: vA120 channel profile, 5 codes, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 10dB$

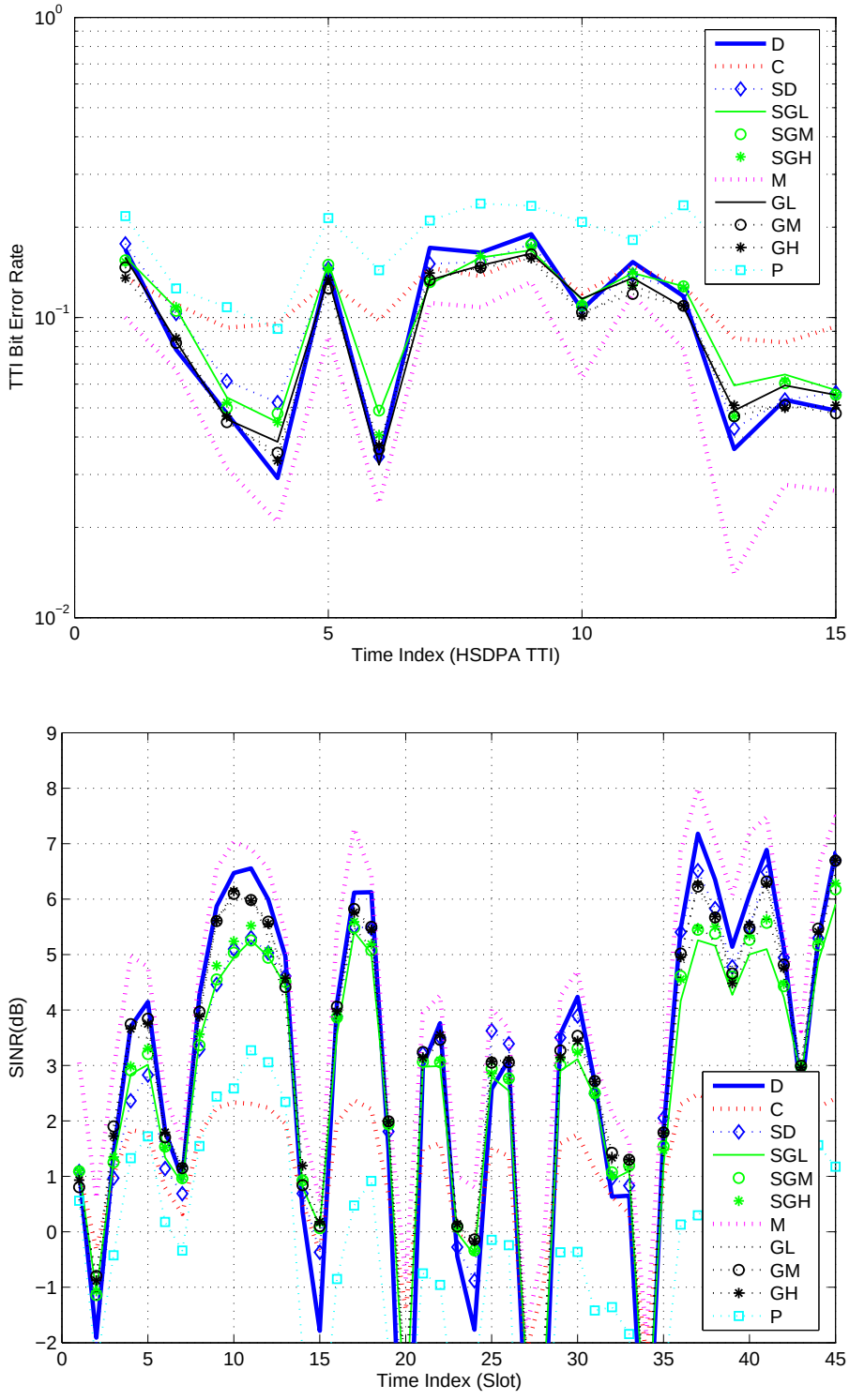


Figure 6.10: vA120 channel profile, 5 codes, $E_c/\hat{I}_{or} = -3dB$, $\hat{I}_{or}/I_{oc} = 6dB$

Part III

Multiuser Detection

Chapter 7

Iterative Receivers with Chip Equalizers

We consider iterative WCDMA receiver techniques for the UMTS FDD downlink. The popular LMMSE chip equalizer-correlator receiver we covered in the previous chapters does not exploit subspaces in partially loaded systems. This is in contrast to the symbol level LMMSE receiver, which is time-varying though, due to the scrambler, and hence too complex to implement. A compromise can be found by performing symbol level Multi-Stage Wiener Filtering (MSWF), which is an iterative solution in which the complexity per iteration becomes comparable to twice that of the RAKE receiver. Since the MSWF works best when the input is white, better performance is obtained if the RAKE in each MSWF stage gets replaced by a chip equalizer-correlator. One of the main contributions here is to point out that the chip equalizer benefits from a separate optimization in every stage. This is shown through a mix of analysis and simulation results.

7.1 Introduction

LMMSE receiver is complex for UMTS FDD mobile terminals since it not only requires inversion of a large user cross-correlation matrix but also needs the code and the amplitude knowledge of all the active users [73]. Furthermore, LMMSE solution changes every chip period due to aperiodic scrambling. The LMMSE *chip* equalizer-correlator is a suboptimal but much simpler alternative which is derived by modeling the scrambler as a stationary random sequence [16, 17]. Another suboptimal multiuser detector that *explicitly* focuses on subtracting the signals of interfering codes is the parallel interference cancellation (PIC) receiver [24]. It is well known that, under very relaxed cell loads, when the number of iterations goes to infinity, PIC might converge to the decorrelating receiver [22]. However,

provided that it converges, still the convergence rate is very slow and it requires many stages to obtain a reasonable performance. This is due to the existence of high cross-correlations among users, which in fact is a consequence of the low orthogonality factor obtained initially from the usage of Rake receiver in the front-end [74, 31, 30]. In this chapter, to at least guarantee the convergence in realistic loading factor situations and to increase the speed of convergence, we start the decorrelation operation, i.e. the zero forcing (ZF) *symbol* equalization from the output of LMMSE chip equalizer correlator front end receiver whose orthogonality factor is higher than the Rake receiver. For approximating this matrix inversion operation, we consider the polynomial expansion (PE) technique which is a better structured equivalent of PIC [28].

Till now interference cancellation has been considered somewhat reluctantly for the downlink since it unrealistically requires knowing the locations of active codes in the OVSF tree and the amounts of powers they carry. However the problem can be simplified by an equivalent modeling of the active multi-rate transmission system as a multi-code pseudo-transmission system at any chosen single SF-level L in the OVSF hierarchy. One toy example representing actually the UMTS-TDD case that contains SFs ranging from 1 to 16 is given in Fig. 7.1. In this example, the nodes corresponding to the active codes at SF-levels 4 and 8 are demonstrated by black bulbs. Their pseudo-equivalents at SF-level 16 (i.e. $L = 16$) are demonstrated by zig-zag pattern bulbs.

One can detect the existence or absence of pseudo-codes at the pseudo-level by comparing the powers at their correlator outputs with a noise-floor threshold [75]. These multiple correlations can be implemented with $O(L\,logL)$ complexity by using Fast Walsh Hadamard Transformation (FWHT). Unitary FWHTs (U-FWHT) with proper dimensions can be logically/physically exploited to see/implement the two-way transformations between actual symbol sequences corresponding to the known codes (e.g. HSDPA codes) at various SF-levels and their pseudo-symbol sequence equivalents at a single SF-level. Fig. 7.2 demonstrates the two-way transformations between L_2/L_1 consecutive (time-multiplexed) actual symbols a_i at level L_1 and L_2/L_1 parallel (code-multiplexed) pseudo-symbols $\tilde{a}_i$ at a larger SF-level L_2. $P_{\{L_2/L_1\}}/S$ and $S/P_{\{L_2/L_1\}}$ are parallel to serial and serial to parallel converters from/to a bus size L_2/L_1. When actual symbols reside at a higher SF-level, the two transformations have reverse roles.

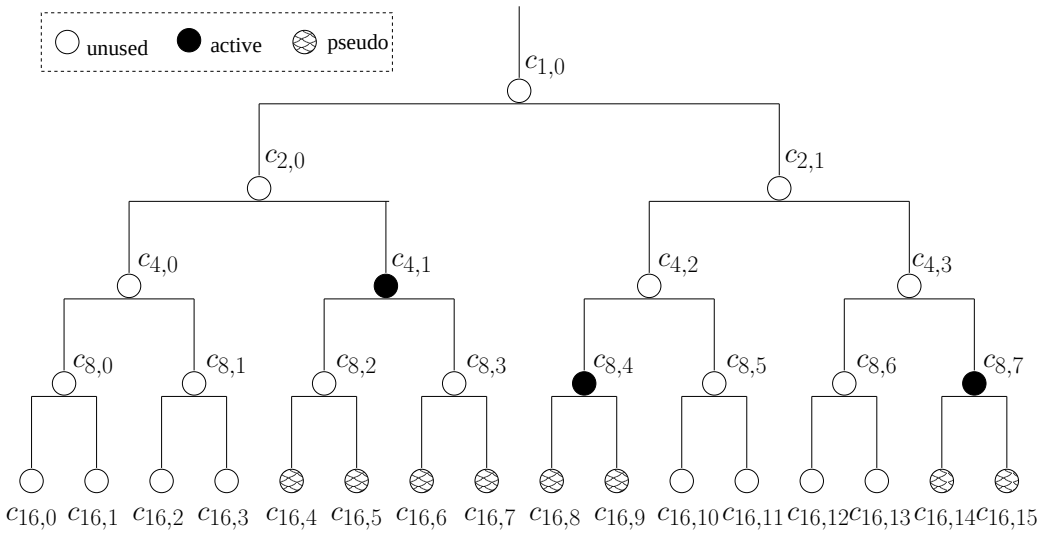


Figure 7.1: Equivalency of active-multirate and pseudo-multicode systems

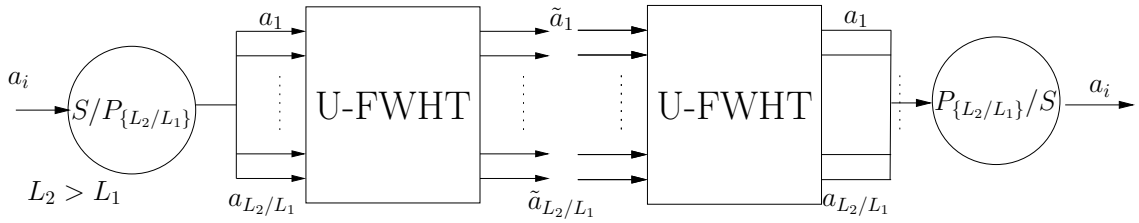


Figure 7.2: Transformations between actual and pseudo symbols

7.2 Polynomial Expansion Receiver

In this section, we develop parallel intracell interference canceling (IC) structures based on polynomial expansion (PE) technique which was initially proposed in [28]. We exploit the *pseudo-equivalency* concept at the highest *active* SF-level, SF-256, in the UMTS-FDD downlink for applying PE at this level. We ignore the existence of SF-512 since it is rarely used carrying control commands during an upload operation. The rationale for choosing the *highest active SF*, from now on called L, is to obtain the highest possible degree of freedom in determining the PE subspace. If any other level L_x were selected, then an activity on a child code of $c_{x,i}$, $i \in \{0, \ldots, L_x - 1\}$, say at a level $L_y > L_x$ on $c_{y,j}$, $j \in \{(L_y/L_x)i, \ldots, (L_y/L_x)(i+1) - 1\}$, would obligate the implicit inclusion of also all the other child codes of $c_{x,i}$ at level L_y by including $c_{x,i}$ in the PE. This would have an adverse effect of noise amplification.

Pseudo-codes might be used in place of the *unknown actual codes* since the actual symbol estimates and their powers are not necessary as long as the pseudo-symbols are treated linearly in interference cancellation. However, knowing or detecting the actual codes is an opportunity for exploiting hard or hyperbolic-tangent nonlinearities or even channel decoding and encoding to refine their symbol estimates [76, 77]. In the latter case, one can pass between the symbol blocks of known codes and their pseudo-equivalents at SF-256 by properly dimensioned FWHTs. By this way hybrid treatment, i.e. respective nonlinear and linear treatment of known and unknown codes, becomes possible.

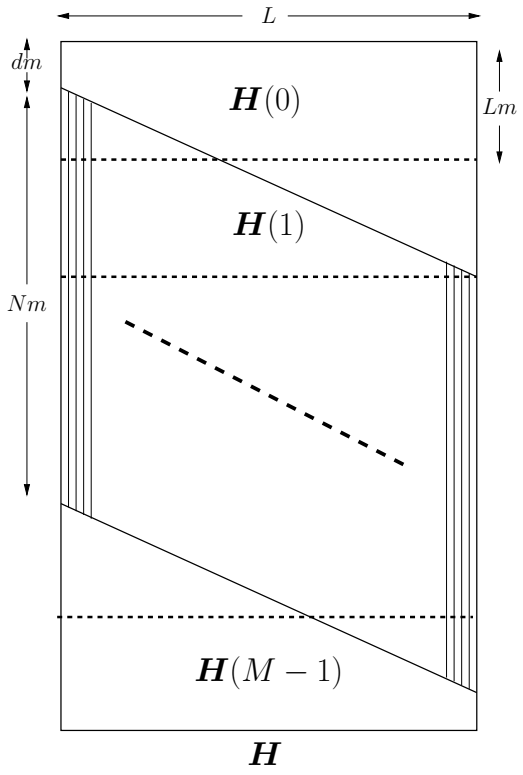


Figure 7.3: Channel impulse response of $\boldsymbol{H}(z)$.

We model the discrete time received signal over one pseudo-symbol period as

$$\boldsymbol{Y}[n] = \boldsymbol{H}(z)\boldsymbol{S}[n]\boldsymbol{C}\boldsymbol{A}[n] + \boldsymbol{V}[n] = \widetilde{\boldsymbol{G}}(n,z)\boldsymbol{A}[n] + \boldsymbol{V}[n]$$

representing the system at the symbol rate. As shown in Figure 7.3, $\boldsymbol{H}(z) = \sum_{i=0}^{M-1} \boldsymbol{H}[i]\, z^{-i}$ is the symbol rate $Lm \times L$ channel transfer function, z^{-1} being the symbol period delay operator. The block coefficients $\boldsymbol{H}(i)$ are the $M = \lceil \frac{L+N+d-1}{L} \rceil$ parts of the block Toeplitz

matrix with $m \times 1$ sized blocks, $\boldsymbol{h}$ being the first column whose top entries might be zero for it comprises the transmission delay d between the BS and the mobile terminal. In this representation, $\boldsymbol{H}[0]$ carries the signal part corresponding to $\boldsymbol{A}[n]$ where there is no user of interest inter-symbol interference (ISI) but only user of interest inter-chip interference (ICI) and multi-user interference (MUI). $\boldsymbol{H}(i), (i \in \{1, 2, \ldots, M-1\})$,similarly, carries the ICI and MUI from $\boldsymbol{A}[n-i]$. The $L \times L$ matrix $\boldsymbol{S}[n]$ is diagonal and contains the scrambler for symbol period n. The column vector $\boldsymbol{A}[n]$ contains the K (pseudo-)symbols and $\boldsymbol{C}$ is the $L \times K$ matrix of the K active codes.

Although it is possible to find an FIR left inverse filter for $\tilde{\boldsymbol{G}}(n, z)$ provided that $Lm \geq K$, this is not practical since $\tilde{\boldsymbol{G}}(n, z)$ is time-varying due to the aperiodicity of the scrambling. Therefore, we will introduce a less complex approximation to this inversion based on the polynomial expansion technique [28]. Instead of basing the receiver directly on the received signal, we shall first introduce a dimensionality reduction step from Lm to K by equalizing the channels with Linear Minimum Mean Square Error Zero Forcing (LMMSE-ZF) chip rate equalizers $\boldsymbol{F}(z)$ followed by a bank of correlators. LMMSE-ZF equalizer is the one among all possible ZF equalizers which minimizes the MSE at the output [78].

Let $\boldsymbol{X}[n]$ be the $K \times 1$ correlator output, which would correspond to the Rake receiver outputs if channel matched filters were used instead of channel equalizers. Then,

$$\begin{aligned} \boldsymbol{X}[n] &= \tilde{\boldsymbol{F}}(n, z)\boldsymbol{Y}[n] \\ &= \boldsymbol{C}^H \boldsymbol{S}^H[n]\boldsymbol{F}(z)(\tilde{\boldsymbol{G}}(n, z)\boldsymbol{A}[n] + \boldsymbol{V}[n]) \\ &= \boldsymbol{M}(n, z)\boldsymbol{A}[n] + \tilde{\boldsymbol{F}}(n, z)\boldsymbol{V}[n] \end{aligned}$$

where $\boldsymbol{M}(n, z) = \tilde{\boldsymbol{F}}(n, z)\tilde{\boldsymbol{G}}(n, z)$ and ZF equalization results in $\boldsymbol{F}(z)\boldsymbol{H}(z) = \boldsymbol{I}$. Hence,

$$\boldsymbol{M}(n, z) = \sum_{i=-\infty}^{\infty} \boldsymbol{M}[n, i]z^{-i} = \begin{bmatrix} \boldsymbol{I} & * \\ * & \boldsymbol{I} \end{bmatrix} \tag{7.1}$$

due to proper normalization of the code energies.

In order to obtain the estimate of $\boldsymbol{A}[n]$, we initially consider the processing of $\boldsymbol{X}[n]$ by a decorrelator as

$$\begin{aligned} \widehat{\boldsymbol{A}}[n] &= \boldsymbol{M}(n, z)^{-1}\boldsymbol{X}[n] \\ &= (\boldsymbol{I} - \overline{\boldsymbol{M}}(n, z))^{-1}\boldsymbol{X}[n]. \end{aligned} \tag{7.2}$$

The correlation matrix $\boldsymbol{M}(n,z)$ has a coefficient $\boldsymbol{M}[n,0]$ with a dominant unit diagonal in the sense that all other elements of the $\boldsymbol{M}[n,i]$ are much smaller than one in magnitude. Hence, the polynomial expansion approach suggests to develop $(\boldsymbol{I}-\overline{\boldsymbol{M}}(n,z))^{-1} = \sum_{i=0}^{\infty}\overline{\boldsymbol{M}}(n,z)^i$ up to some finite order, which leads to the iterative receiver as[1]

$$\begin{aligned}\widehat{\boldsymbol{A}}^{(-1)} &= 0 \;\;;\;\; i \geq 0\,.\\ \widehat{\boldsymbol{A}}^{(i)} &= \boldsymbol{X} + \overline{\boldsymbol{M}}\,\widehat{\boldsymbol{A}}^{(i-1)}\,,\\ &= \boldsymbol{X} + (\boldsymbol{I}-\boldsymbol{M})\,\widehat{\boldsymbol{A}}^{(i-1)}\,,\\ &= \widehat{\boldsymbol{A}}^{(i-1)} + \tilde{\boldsymbol{F}}^{i}(\boldsymbol{Y} - \tilde{\boldsymbol{G}}\widehat{\boldsymbol{A}}^{(i-1)})\,. \end{aligned} \tag{7.3}$$

The resultant iterative receiver architecture is given in Figure 7.4 where the numbers in paranthesis indicate the iteration indices. A practical receiver would be limited to a few orders, the quality of which depends on the degree of dominance of the static part of the diagonal of $\boldsymbol{M}(n,z)$ given in (7.3) with respect to the ICI carrying dynamic contents of the diagonal elements and multiuser interference (MUI) carrying off-diagonal elements.

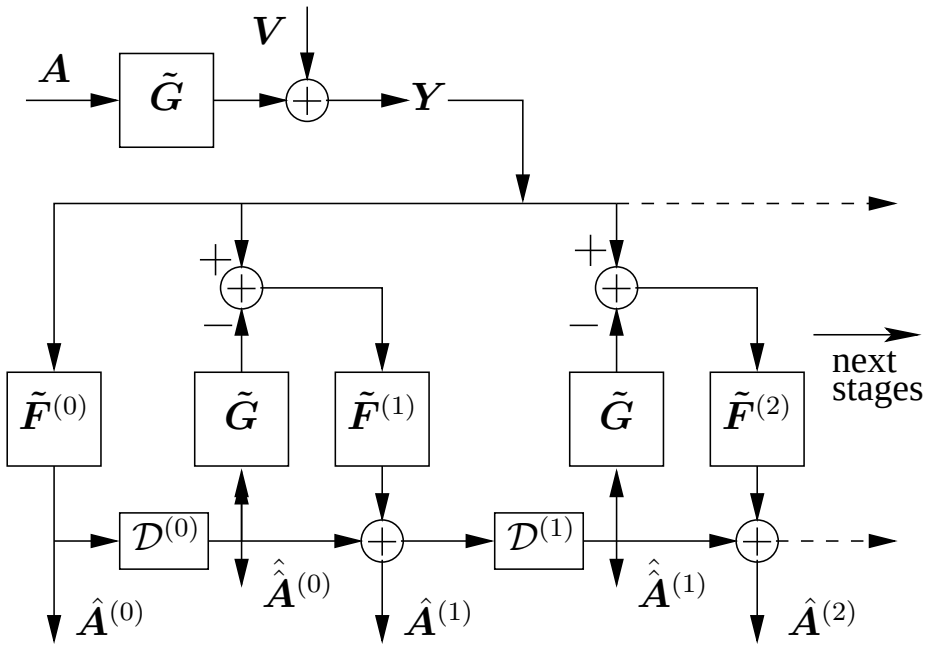


Figure 7.4: Polynomial expansion receiver

In an iterative PE approach, it is advantageous to replace several *local* receiver components obtained from *global* LMMSE-ZF formulation by their LMMSE counterparts.

[1] time indices are dropped for brevity

Such modifications should lead to smaller offdiagonal power and hence faster convergence of the iterations to an estimate that is closer to a global MMSE estimate. For example LMMSE-ZF chip equalizers can be replaced by LMMSE equalizers which, though perturb the orthogonal structure of the received signal from the BS, do not enhance as much the intercell interference plus noise [79]. Furthermore some symbol feedback functionalities $\mathcal{D}$ shown in Figure 7.5 such as LMMSE weighting factors, hard decisions, a variety of soft decisions like hyperbolic-tangent functionality or even channel decoding and encoding blocks can be introduced.

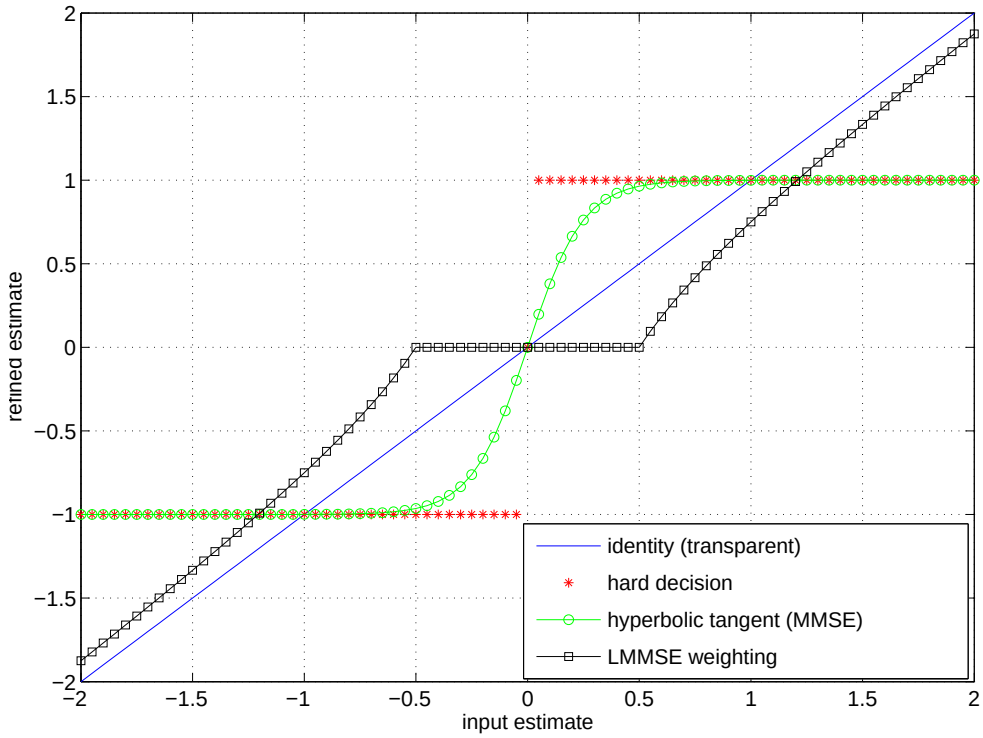


Figure 7.5: Feedback functionalities for real and imaginary parts of QPSK symbols which have 6dB SINR

7.3 Filter Adaptation

Figure 7.6 shows the open form of the receiver in Figure 7.4 where we clearly see the chip level blocks. In case the symbol feedback functionality $\mathcal{D}$ is the identity matrix, we

can further obtain a third equivalent architecture given in Figure 7.7 which, different from the previous two, iterates over the *chip* estimates at chip level filter outputs.

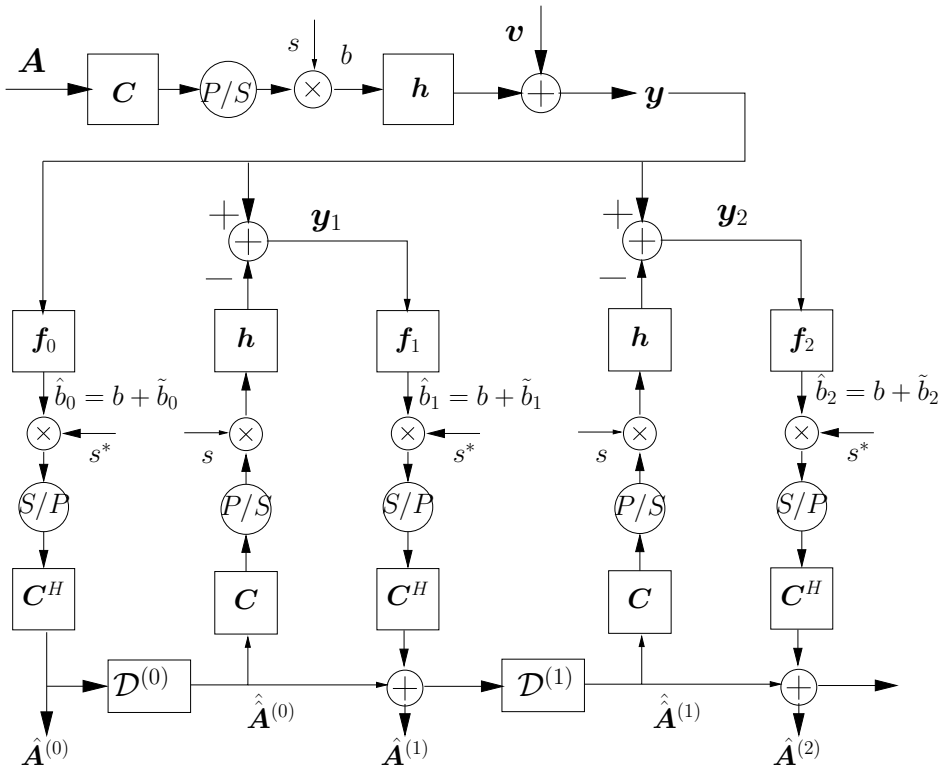


Figure 7.6: Polynomial expansion receiver open format

Since the projection operation $\boldsymbol{S}[n]\boldsymbol{C}\boldsymbol{C}^H\boldsymbol{S}^*[n]$ is not a chip level operation and is not convolutive it cannot be easily integrated into the filter optimization process. Nevertheless it has two nice properties: the diagonal part is the deterministic value $\mathcal{C}_l\boldsymbol{I}$ where $\mathcal{C}_l$ is the effective cell loading factor and the expected value of the non-diagonal part is zero. By considering only the diagonal parts of the local projection operations, we reach to the Multi-stage Wiener (LMMSE) filter adaptation procedure given in the equations group (7.5) where $\left\{\boldsymbol{\mathcal{X}_i}, \boldsymbol{\mathcal{Y}_i}, \tilde{\boldsymbol{B}}_i\right\}$ respectively denote {transfer function between the BS signal and the residual BS signal, transfer function for the intercell interference plus noise, the residual interference plus noise} at iteration i[2]. The Wiener (LMMSE) filter and the unbiased LMMSE filter are denoted by $\boldsymbol{F}_i^w$ and $\boldsymbol{F}_i$ respectively.

[2]Each bold variable in Section 7.3 has a (z) suffix which is dropped for brevity; † stands for z-transform para-conjugate operator meaning matched filter in the time domain

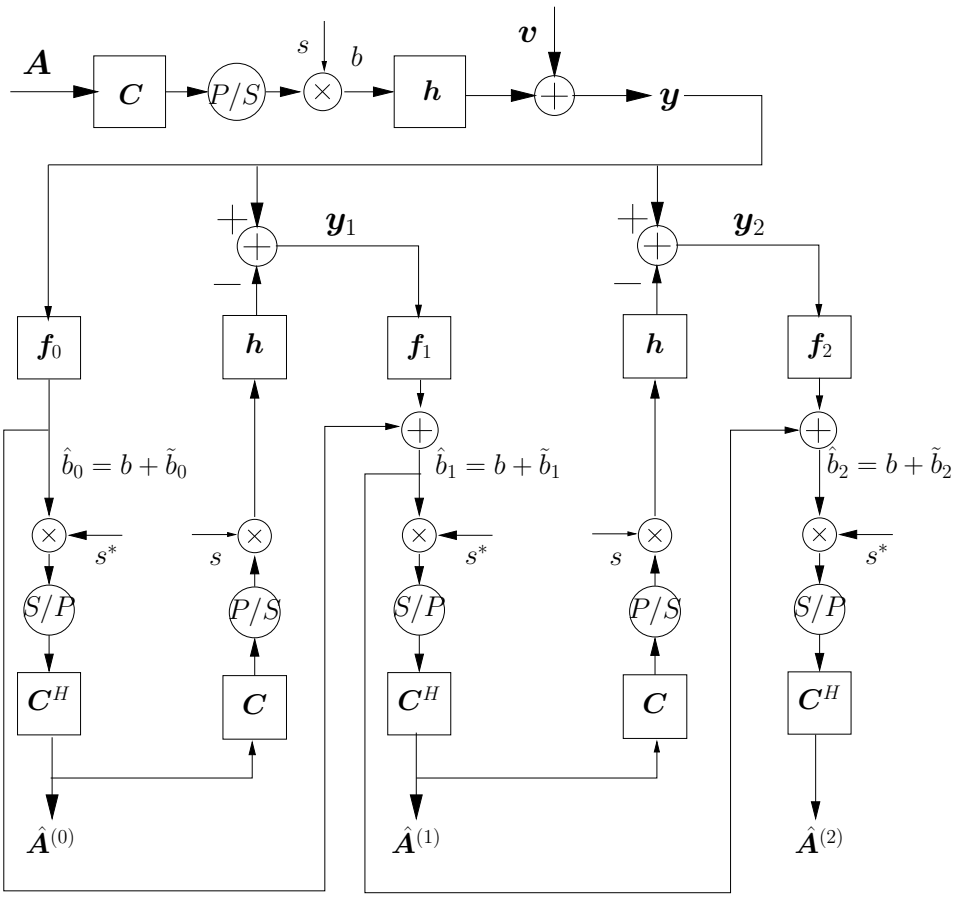


Figure 7.7: PE receiver equivalent chip estimate iterating model

The LMMSE optimization process output is the complete filter expression of $\boldsymbol{F}_i$ from which we derive its two ingredients $\boldsymbol{S}_{\tilde{b}_{i-1}\boldsymbol{y}_i}$ and $\boldsymbol{S}_{\boldsymbol{y}_i\boldsymbol{y}_i}$ by *factorization*. The structure of the factorized terms are clear guidelines for understanding that, when unbiased, the chip level filter $\boldsymbol{F}_i$ intends to estimate and subtract the *residual interference plus noise* term at the preceding iteration, which is expected to be also valid for systems with additional system components such as hard decisions. For example, if we consider the loop among the signals $\hat{b}_0$, $\boldsymbol{y}_1$ and $\hat{b}_1$ that contains the transfer functions $\boldsymbol{F}_1(z)$ and $\boldsymbol{H}(z)$, it estimates the residual signal $\tilde{b}_0$ and subtracts it from $\hat{b}_0$ which leads to the creation of the new residual signal $\tilde{b}_1$. The same reasoning holds for subsequent iterations where the amount of interference plus noise variance $\sigma^2_{\tilde{b}_i}$ is expected to decrease with increasing i as long as the spectral radius $\rho(\boldsymbol{I} - \mathcal{C}_l\boldsymbol{F}_i\boldsymbol{H}) < 1$.

INITIALIZATION (First Stage)

$$\boldsymbol{\mathcal{X}}_0 = \boldsymbol{F}_0\boldsymbol{H} - \boldsymbol{I}$$

$$\boldsymbol{\mathcal{Y}}_0 = \boldsymbol{F}_0$$

$$\tilde{\boldsymbol{B}}_0 = \boldsymbol{\mathcal{X}}_0\boldsymbol{B} + \boldsymbol{\mathcal{Y}}_0\boldsymbol{V}$$

ITERATIONS (Interference Cancellation Stages)

for $(i > 0)$ and $(i < i_{max})$

$$\boldsymbol{\mathcal{X}}_i = (\boldsymbol{I} - \mathcal{C}_l\boldsymbol{F}_i\boldsymbol{H})\boldsymbol{\mathcal{X}}_{i-1}$$

$$\boldsymbol{\mathcal{Y}}_i = (\boldsymbol{I} - \mathcal{C}_l\boldsymbol{F}_i\boldsymbol{H})\boldsymbol{\mathcal{Y}}_{i-1} + \boldsymbol{F}_i$$

$$\tilde{\boldsymbol{B}}_i = \boldsymbol{\mathcal{X}}_i\boldsymbol{B} + \boldsymbol{\mathcal{Y}}_i\boldsymbol{V}$$

$$\arg_{\boldsymbol{F}_i^w} \min \frac{1}{2\pi j} \oint \frac{dz}{z} \left(\boldsymbol{\mathcal{X}}_i \boldsymbol{\mathcal{X}}_i^{\dagger} \sigma_b^2 + \boldsymbol{\mathcal{Y}}_i \boldsymbol{\mathcal{Y}}_i^{\dagger} \sigma_v^2 \right) \tag{7.4}$$

$$\boldsymbol{F}_i^w = \boldsymbol{S}_{\tilde{b}_{i-1}\boldsymbol{y}_i} \boldsymbol{S}_{\boldsymbol{y}_i\boldsymbol{y}_i}^{-1}$$

$$\boldsymbol{S}_{\tilde{b}_{i-1}\boldsymbol{y}_i} = \mathcal{C}_l \boldsymbol{\mathcal{X}}_{i-1} \boldsymbol{\mathcal{X}}_{i-1}^{\dagger} \boldsymbol{H}^{\dagger} \sigma_b^2 - \boldsymbol{\mathcal{Y}}_{i-1} \left(\boldsymbol{I} - \mathcal{C}_l \boldsymbol{H} \boldsymbol{\mathcal{Y}}_{i-1} \right)^{\dagger} \sigma_v^2$$

$$\boldsymbol{S}_{\boldsymbol{y}_i\boldsymbol{y}_i} = \mathcal{C}_l^2 \boldsymbol{H} \boldsymbol{\mathcal{X}}_{i-1} \boldsymbol{\mathcal{X}}_{i-1}^{\dagger} \boldsymbol{H}^{\dagger} \sigma_b^2 + \left(\boldsymbol{I} - \mathcal{C}_l \boldsymbol{H} \boldsymbol{\mathcal{Y}}_{i-1} \right) \left(\boldsymbol{I} - \mathcal{C}_l \boldsymbol{H} \boldsymbol{\mathcal{Y}}_{i-1} \right)^{\dagger} \sigma_v^2$$

$$\boldsymbol{F}_i = \frac{2\pi j \boldsymbol{F}_i^w}{\oint \frac{dz}{z} \boldsymbol{F}_i^w \boldsymbol{H}} \text{: unbiasing operation} \tag{7.5}$$

end

In practice, LMMSE chip equalizer correlator blocks might also be implemented as Generalized Rake (G-Rake) receivers in which case, in each stage, filtering with $\boldsymbol{F}_i$ and $\boldsymbol{H}$ will be similar to the filtering part of the Rake receiver [80]. Hence, each iteration will have twice the complexity of that of Rake.

7.3.1 Impact of Symbol Feedback Nonlinearities on Filter Expressions

When hard decisions or hyperbolic tangent nonlinearities are used on a subset of codes, we see two alternatives to reflect their impact to filtering expressions. The first approach is to simply assume that the associated symbols are perfectly estimated and hence to exclude them after first stage. In this case, the only required changes are to consider $\mathcal{C}_l$ and σ_b^2 as respectively the loading factor and the sum chip variance of the remaining codes that are treated linearly. The second approach is to quantify the variances of the symbol

estimation errors after the nonlinearities at every stage by the scheme in Appendix D and introduce new additive Gaussian noise sources at those points with the obtained variances.

7.4 Intercell Interference Cancellation

Polynomial expansion receiver can be modified to include also the intercell interference cancellation. The filter adaptations, the changes in signal modeling and the architecture for cancelling the interference of one neighboring BS are given in equations group (7.6), equations group (7.7) and Figure 7.8. The scheme can be easily extended to cover any number of cells by increasing the sizes of vectors and matrices in the equations group (7.7).

INITIALIZATION (First Stage)

$$\boldsymbol{\mathcal{X}}_0 = \boldsymbol{F}_0\boldsymbol{H} - \boldsymbol{I}$$

$$\boldsymbol{\mathcal{Y}}_0 = \boldsymbol{F}_0$$

$$\tilde{\boldsymbol{B}}_0 = \boldsymbol{\mathcal{X}}_0\boldsymbol{B} + \boldsymbol{\mathcal{Y}}_0\boldsymbol{V}$$

ITERATIONS (Interference Cancellation Stages)

for $(i > 0)$ and $(i < i_{max})$

$$\boldsymbol{\mathcal{X}}_i = (\boldsymbol{I} - \boldsymbol{F}_i\boldsymbol{H}\boldsymbol{\mathcal{C}}_l)\boldsymbol{\mathcal{X}}_{i-1}$$

$$\boldsymbol{\mathcal{Y}}_i = (\boldsymbol{I} - \boldsymbol{F}_i\boldsymbol{H}\boldsymbol{\mathcal{C}}_l)\boldsymbol{\mathcal{Y}}_{i-1} + \boldsymbol{F}_i$$

$$\tilde{\boldsymbol{B}}_i = \boldsymbol{\mathcal{X}}_i\boldsymbol{B} + \boldsymbol{\mathcal{Y}}_i\boldsymbol{V}$$

$$\begin{aligned}
&\arg_{\boldsymbol{F}^w_{1,i}} \min \frac{1}{2\pi j}\oint \frac{dz}{z}\left(\boldsymbol{\mathcal{X}}_{1,i}\boldsymbol{\Sigma}_b^2\boldsymbol{\mathcal{X}}_{1,i}^\dagger + \boldsymbol{\mathcal{Y}}_{1,i}\boldsymbol{\mathcal{Y}}_{1,i}^\dagger\sigma_v^2\right) \\
&\arg_{\boldsymbol{F}^w_{2,i}} \min \frac{1}{2\pi j}\oint \frac{dz}{z}\left(\boldsymbol{\mathcal{X}}_{2,i}\boldsymbol{\Sigma}_b^2\boldsymbol{\mathcal{X}}_{2,i}^\dagger + \boldsymbol{\mathcal{Y}}_{2,i}\boldsymbol{\mathcal{Y}}_{2,i}^\dagger\sigma_v^2\right)
\end{aligned} \tag{7.6}$$

$$\boldsymbol{F}^w_{1,i} = \boldsymbol{S}_{\tilde{b}_{1,i-1}\boldsymbol{y}_i}\boldsymbol{S}^{-1}_{\boldsymbol{y}_i\boldsymbol{y}_i} \qquad \boldsymbol{F}^w_{2,i} = \boldsymbol{S}_{\tilde{b}_{2,i-1}\boldsymbol{y}_i}\boldsymbol{S}^{-1}_{\boldsymbol{y}_i\boldsymbol{y}_i}$$

$$\boldsymbol{S}_{\tilde{b}_{1,i-1}\boldsymbol{y}_i} = \mathcal{C}_{l_1}\boldsymbol{\mathcal{X}}_{1,i-1}\boldsymbol{\Sigma}_b^2\boldsymbol{\mathcal{X}}_{1,i-1}^\dagger\boldsymbol{H}_1^\dagger - \boldsymbol{\mathcal{Y}}_{1,i-1}\left(\boldsymbol{I} - \mathcal{C}_{l_1}\boldsymbol{H}_1\boldsymbol{\mathcal{Y}}_{1,i-1}\right)^\dagger\sigma_v^2$$

$$\boldsymbol{S}_{\tilde{b}_{2,i-1}\boldsymbol{y}_i} = \mathcal{C}_{l_2}\boldsymbol{\mathcal{X}}_{2,i-1}\boldsymbol{\Sigma}_b^2\boldsymbol{\mathcal{X}}_{2,i-1}^\dagger\boldsymbol{H}_2^\dagger - \boldsymbol{\mathcal{Y}}_{2,i-1}\left(\boldsymbol{I} - \mathcal{C}_{l_2}\boldsymbol{H}_2\boldsymbol{\mathcal{Y}}_{2,i-1}\right)^\dagger\sigma_v^2$$

$$\boldsymbol{S}_{\boldsymbol{y}_i\boldsymbol{y}_i} = \boldsymbol{H}\boldsymbol{\mathcal{C}}_l\boldsymbol{\mathcal{X}}_{i-1}\boldsymbol{\Sigma}_b^2\boldsymbol{\mathcal{X}}_{i-1}^\dagger\boldsymbol{\mathcal{C}}_l\boldsymbol{H}^\dagger + \left(\boldsymbol{I} - \boldsymbol{H}\boldsymbol{\mathcal{C}}_l\boldsymbol{\mathcal{Y}}_{i-1}\right)\left(\boldsymbol{I} - \boldsymbol{H}\boldsymbol{\mathcal{C}}_l\boldsymbol{\mathcal{Y}}_{i-1}\right)^\dagger\sigma_v^2$$

$$\boldsymbol{F}_{1,i} = \frac{2\pi j\boldsymbol{F}^w_{1,i}}{\oint \frac{dz}{z}\boldsymbol{F}^w_{1,i}\boldsymbol{H}_1} \qquad \boldsymbol{F}_{2,i} = \frac{2\pi j\boldsymbol{F}^w_{2,i}}{\oint \frac{dz}{z}\boldsymbol{F}^w_{2,i}\boldsymbol{H}_2}$$

end

$$
\begin{aligned}
\boldsymbol{A}[n] &\longrightarrow \begin{bmatrix} \boldsymbol{A}_1[n] \\ \boldsymbol{A}_2[n] \end{bmatrix} : \text{vector of symbols} \\
\boldsymbol{C} &\longrightarrow \begin{bmatrix} \boldsymbol{C}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{C}_2 \end{bmatrix} : \text{channelization codes} \\
\boldsymbol{S}[n] &\longrightarrow \begin{bmatrix} \boldsymbol{S}_1[n] & \mathbf{0} \\ \mathbf{0} & \boldsymbol{S}_2[n] \end{bmatrix} : \text{scrambling} \qquad (7.7) \\
\boldsymbol{B} &\longrightarrow \begin{bmatrix} \boldsymbol{B}_1 \\ \boldsymbol{B}_2 \end{bmatrix} : \text{transmitted chip sequences} \\
\sigma_b^2 &\longrightarrow \boldsymbol{\Sigma}_b^2 = \begin{bmatrix} \sigma_{b_1}^2 & 0 \\ 0 & \sigma_{b_2}^2 \end{bmatrix} : \text{chip level signal covariance} \\
\boldsymbol{H}(z) &\longrightarrow \begin{bmatrix} \boldsymbol{H}_1(z) & \boldsymbol{H}_2(z) \end{bmatrix} : \text{chip rate channel} \\
\boldsymbol{F}_i(z) &\longrightarrow \begin{bmatrix} \boldsymbol{F}_{1,i}(z) \\ \boldsymbol{F}_{2,i}(z) \end{bmatrix} : \text{chip level equalizers at iteration i} \\
\tilde{\boldsymbol{G}}(n,z) &\longrightarrow \begin{bmatrix} \tilde{\boldsymbol{G}}_1(n,z) & \tilde{\boldsymbol{G}}_2(n,z) \end{bmatrix} = \begin{bmatrix} \boldsymbol{H}_1(z)\boldsymbol{S}_1[n]\boldsymbol{C}_1 & \boldsymbol{H}_2(z)\boldsymbol{S}_2[n]\boldsymbol{C}_2 \end{bmatrix} : \text{symbol rate channel} \\
\tilde{\boldsymbol{F}}^{(i)}(n,z) &\longrightarrow \begin{bmatrix} \tilde{\boldsymbol{F}}_1^{(i)}(n,z) \\ \tilde{\boldsymbol{F}}_2^{(i)}(n,z) \end{bmatrix} : \text{symbol level equalizers at iteration i} \\
\boldsymbol{\mathcal{X}}_i &\longrightarrow \begin{bmatrix} \boldsymbol{\mathcal{X}}_{1,1,i} & \boldsymbol{\mathcal{X}}_{1,2,i} \\ \boldsymbol{\mathcal{X}}_{2,1,i} & \boldsymbol{\mathcal{X}}_{2,2,i} \end{bmatrix} : \text{interference transfer function} \\
\boldsymbol{\mathcal{X}}_{1,i} &= \begin{bmatrix} \boldsymbol{\mathcal{X}}_{1,1,i} & \boldsymbol{\mathcal{X}}_{1,2,i} \end{bmatrix} : \text{interference transfer function for the first BS signal} \\
\boldsymbol{\mathcal{X}}_{2,i} &= \begin{bmatrix} \boldsymbol{\mathcal{X}}_{2,1,i} & \boldsymbol{\mathcal{X}}_{2,2,i} \end{bmatrix} : \text{interference transfer function for the second BS signal} \\
\boldsymbol{\mathcal{Y}}_i &\longrightarrow \begin{bmatrix} \boldsymbol{\mathcal{Y}}_{1,i} \\ \boldsymbol{\mathcal{Y}}_{2,i} \end{bmatrix} : \text{noise transfer function} \\
\mathcal{C}_l &\longrightarrow \boldsymbol{\mathcal{C}}_l = \begin{bmatrix} \mathcal{C}_{l_1} & 0 \\ 0 & \mathcal{C}_{l_2} \end{bmatrix} : \text{loading factors} \\
\boldsymbol{Y}[n] &\longrightarrow \begin{bmatrix} \boldsymbol{H}_1(z)\boldsymbol{S}_1[n]\boldsymbol{C}_1 & \boldsymbol{H}_2(z)\boldsymbol{S}_2[n]\boldsymbol{C}_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_1[n] \\ \boldsymbol{A}_2[n] \end{bmatrix} + \boldsymbol{V}[n] = \tilde{\boldsymbol{G}}(n,z)\ \boldsymbol{A}[n]\ + \boldsymbol{V}[n]
\end{aligned}
$$

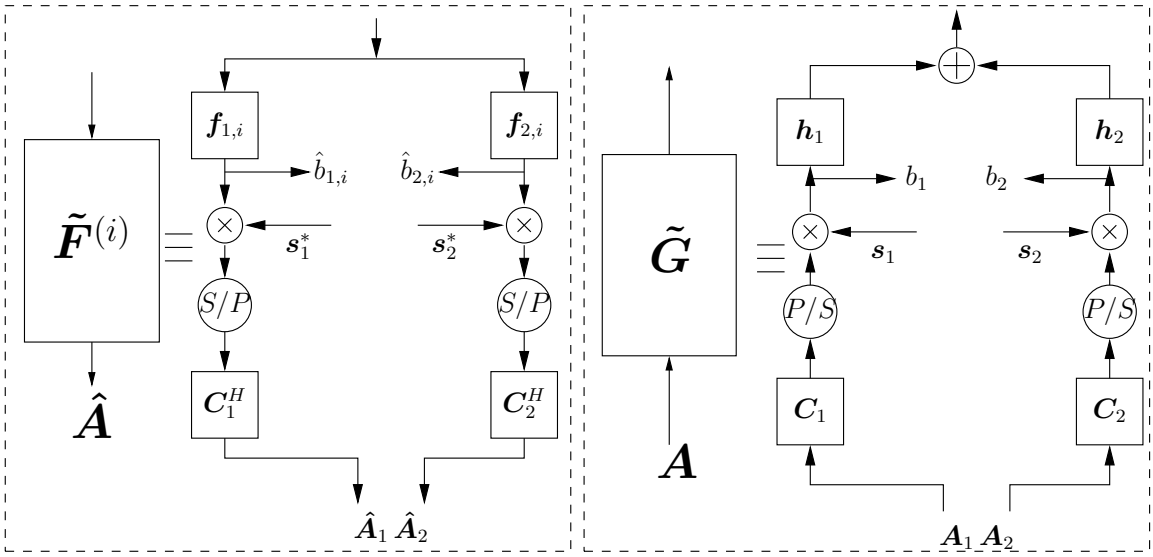


Figure 7.8: Symbol level transfer function blocks and their chip level equivalents

7.5 Simulations and Conclusions

For the simulations we consider only intracell interference cancellation.

We take a high speed downlink packet access (HSDPA) scenario in the UMTS-FDD downlink [8]. We consider 5 HSDPA codes at SF-16 assigned to the UE each consuming 8% of the base station power. The PCPICH pilot tone at SF-256 consumes 10% power. There is the PCCPCH code at SF-256 that consumes 4% power. To effectively model all the rest multirate user codes that we do not know, we place 46 pseudo-codes at level 256 each having 1% power. So in total, 5 HSDSCH codes at SF-16 being equivalent to 80 pseudo-codes at SF-256, the system is effectively 50% loaded with 128 (pseudo-)codes at SF-256, i.e. $\mathcal{C}_l = 0.5$. Although, in practice, the pseudo-codes should be detected by a method explained in the text, for the moment, we assume that they are known. We also assume perfect knowledge of the channel. An oversampling factor of 2 and one receive antenna is used [3]. Static propagation channel parameters are randomly generated from the ITU Vehicular-A power delay profile. Pulse shape is the UMTS-standard, root-raised cosine with a roll-off factor of 0.22. Therefore the propagation channel, pulse shape cascade (i.e. the overall channel) has a length of 19 chips at 3.84 Mchips/sec transmission rate. Symbols

[3]The order of filtering and rechanneling operations have an impact on the noise term in case of polyphase filtering which we neglect for the moment

are QPSK. $\hat{I}_{or}/I_{oc}$ denotes the received base station power to intercell interference plus noise power ratio. We took the average SINR result of 5 HSDPA codes over 100 realizations of one UMTS slot (160 symbol period) transmissions.

Figure 7.9 shows the performances of the PE scheme with various different chip level filter usages and iterations from one to three. The legends indicate the used filters with iteration order. For example F0-F1-F2 means optimized filters are used in all the stages; F0-Rake-Rake hybrid scheme means first stage filter is LMMSE chip equalizer and subsequent two are Rake receivers; Rake-Rake-Rake corresponds to the *conventional* linear PIC. Many other variants different from the shown ones can also be used. As is expected Rake receiver performs the worst. The conventional Linear PIC with only Rake receivers starts diverging after first iteration especially in the I_{or}/I_{oc} values below 10dB. This is consistent with the literature since it is well known that, for guaranteeing the converge of the LPIC, loading factor should be lower than %17 [81][4]. The scheme which uses only F0 does not improve significantly after second iteration. Using Rake receivers after F0 performs very well. As expected adapting the filters at all iterations performs the best. Such a scheme obtains almost the same performance of F0-Rake-Rake in one less iteration, i.e. with configuration F0-F1. At low $\hat{I}_{or}/I_{oc}$ values which reflect the cell edge situations, the performance of the first iteration is better than the second one. This is due to the fact that at low SNR regions the gain from the interference reduction is not sufficient to compensate the loss from noise amplification, since the iterative scheme is still a decorrelation. One might also attribute this to the well-known *ping-pong* effect for LPIC [82].

Figure 7.10 shows the performances when we apply hard decisions on the 5 HSDPA codes which have an effective loading impact $\mathcal{C}_{HSDPA} = \frac{5}{16}$. With the assumption of correct decisions we subtract $\mathcal{C}_{HSDPA}$ from the overall cell load of 0.5 and apply the $\mathcal{C}_l = \frac{3}{16}$ value in the filter adaptation process in (7.5). In this case using Rake receivers after first stage equalization catches up with the optimized filters after three stages. We also observe that conventional PIC also starts getting into a convergence trend. It is not however explicit from the $\boldsymbol{\mathcal{X}_i}$ and $\boldsymbol{\mathcal{Y}_i}$ expressions why things should improve despite the fact that the $\mathcal{C}_l$ value decreases resulting in lower iteration gain in chip estimation. Due to this fact one would at first sight expect an insufficient interference reduction to compensate the amplified noise. This is however not the case due to the fact that almost all of the ingredients of

[4] in the random CDMA, flat fading case

the additional noise term coming from the previous iteration is in the subspace belonging to the codes whose symbols are estimated linearly whereas the final SINR performance metric is computed on codes such as HSDPA codes which are treated by hard decisions. In the full linear treatment however, the additional noise that traverses the iterations with amplification is in the whole signal space. Therefore when hard decisions are applied there is an implicit reduction of additional noise by a factor $\frac{\mathcal{C}_l}{\mathcal{C}_l+\mathcal{C}_{HSDPA}}$. These interpretations seem to be conflicting with the chip equalizer adaptation expressions where we ignored the non-diagonal part of the projection operation $\boldsymbol{S}[n]\boldsymbol{C}\boldsymbol{C}^H\boldsymbol{S}^*[n]$ in order to recover from the dependence on codes. For the interpretations of performances at symbol levels however one has to look from a different perspective, taking into account the code knowledge.

Comparing Figure 7.9 and Figure 7.10 we observe that at medium and high $\hat{I}_{or}/I_{oc}$ working regions hard decisions increase the obtained SINR by 1 to 3 dB. At low $\hat{I}_{or}/I_{oc}$ regions there is no gain, which is understandable since in those regions hard decisions are not reliable.

We next look at the orthogonality factor histograms of the considered receivers by randomly generating 10^4 static channels from the Vehicular A power delay profile with and $\hat{I}_{or}/I_{oc}$ value of 10dB. Figure 7.11 shows the histograms for the CMF and LMMSE equalizer. We see that, besides giving worse median OF, CMF might also give OFs less than 0.4. In Figure 7.12, Figure 7.13 and Figure 7.14 we respectively see the trend of OFs obtained from all CMF usage, CMF usage after first stage equalization and all chip level LMMSE equalizer usage in LPIC iterations. In order to obtain them, we first compute the $\|\boldsymbol{\mathcal{X}}_i\|^2$ and pass to OF as $\alpha_i = \frac{1}{1+\|\boldsymbol{\mathcal{X}}_i\|^2}$ since all the filters are unbiased. The histograms in Figure 7.12 clearly demonstrate the problem with conventional LPIC. From median value perspective the OF improves after the first stage and then starts degrading. Even more important concern is the widening of OF range. After four iterations there are even channel cases where OF is close to zero. The histograms in Figure 7.13 demonstrate the importance of LMMSE chip equalization as a starting point. Although there are still a very little amount of corner cases leading to small OFs, the overall performance is at an acceptable level. Finally the histograms in Figure 7.13 clearly indicate the strength of using optimized chip equalizers at all stages. Not only the median value but also the worst case OF improves with every iteration. In brief we can say that when the mobile knows multiple codes as in the HSDPA service, applying Rake receivers after a first stage equalization stage is a proper choice. In the case of only one code however it is beneficial to adapt filters at every stage.

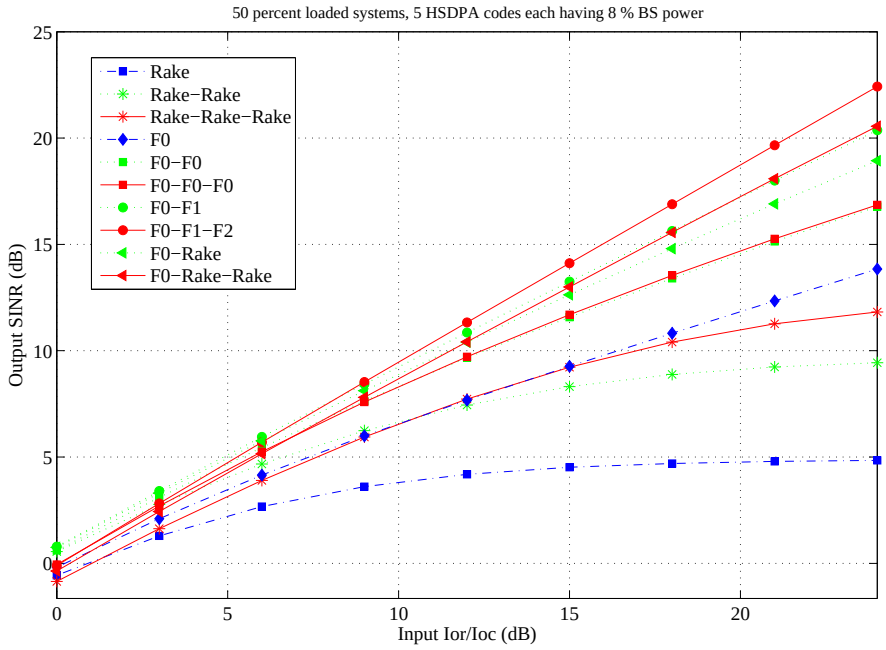


Figure 7.9: SINR vs $\hat{I}_{or}/I_{oc}$ linear decisions results, Vehicular A channel, N=19

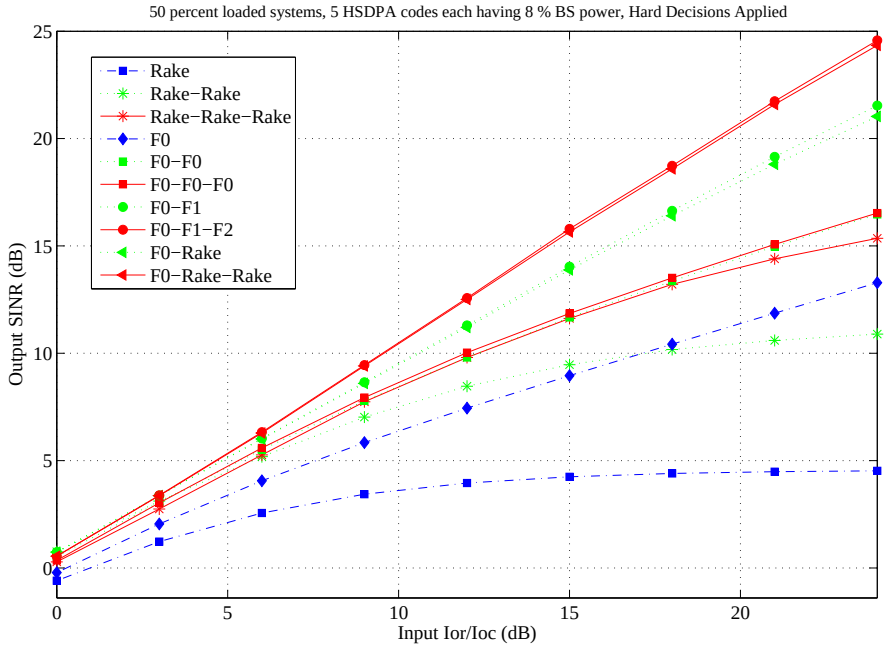


Figure 7.10: SINR vs $\hat{I}_{or}/I_{oc}$ hard decisions results, Vehicular A channel, N=19

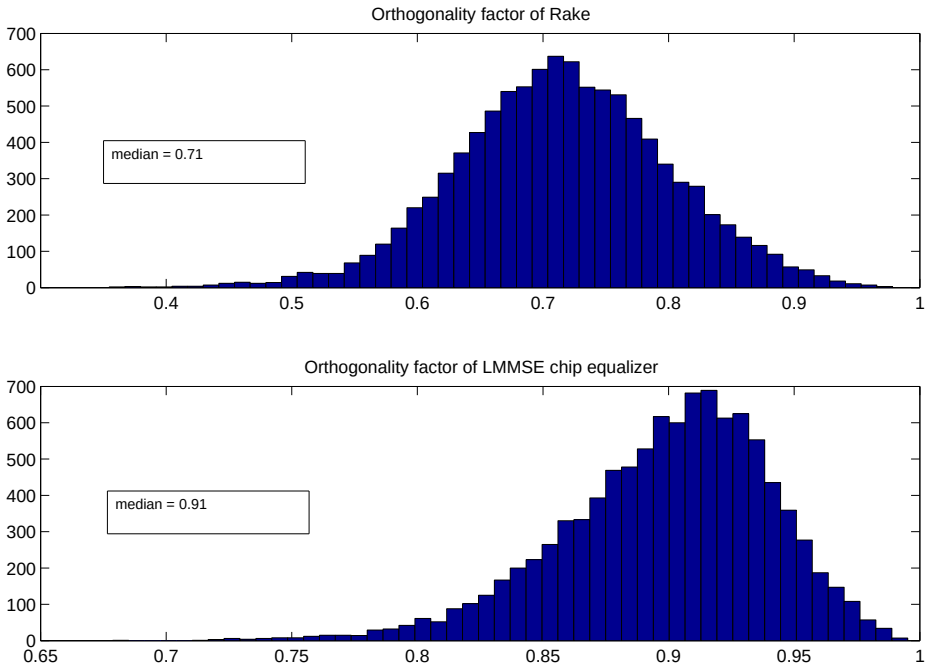


Figure 7.11: Orthogonality factor histograms of 2-phase CMF and 2-phase LMMSE chip equalizer in Vehicular A channel with $\hat{I}_{or}/I_{oc} = 10dB$

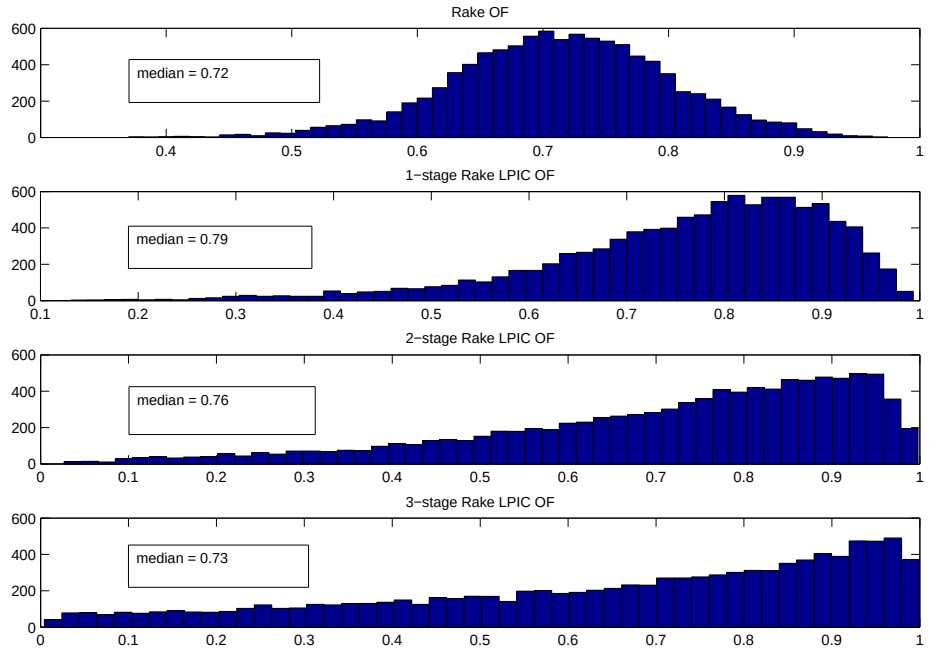


Figure 7.12: Orthogonality factor histogram of conventional LPIC with 2-phase CMF in the Vehicular A channel

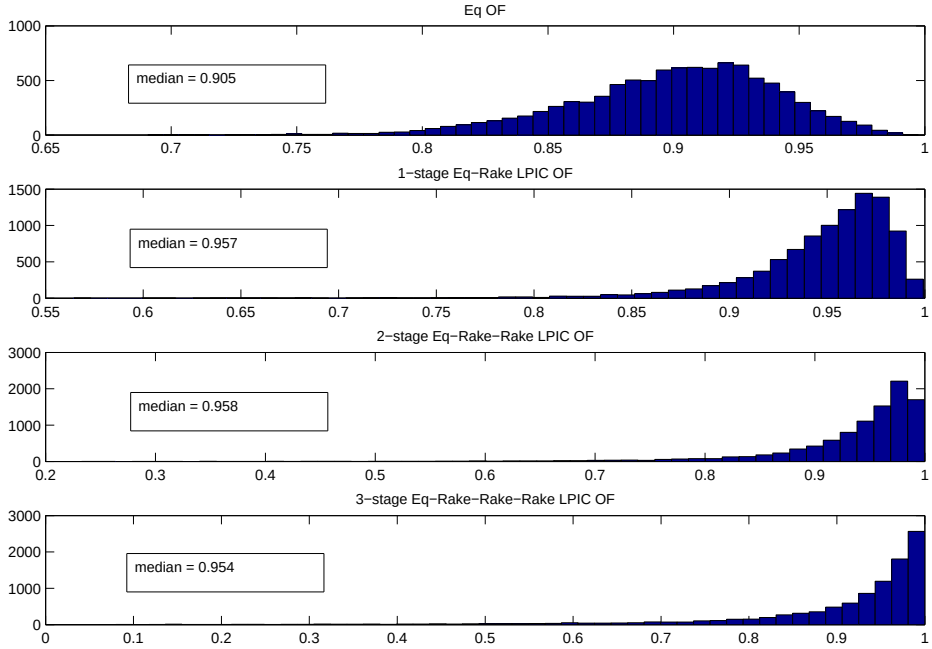


Figure 7.13: Orthogonality factor histogram of LPIC with first stage 2-phase LMMSE chip equalizer followed by 2-phase CMFs in the Vehicular A channel with $\hat{I}_{or}/I_{oc} = 10dB$

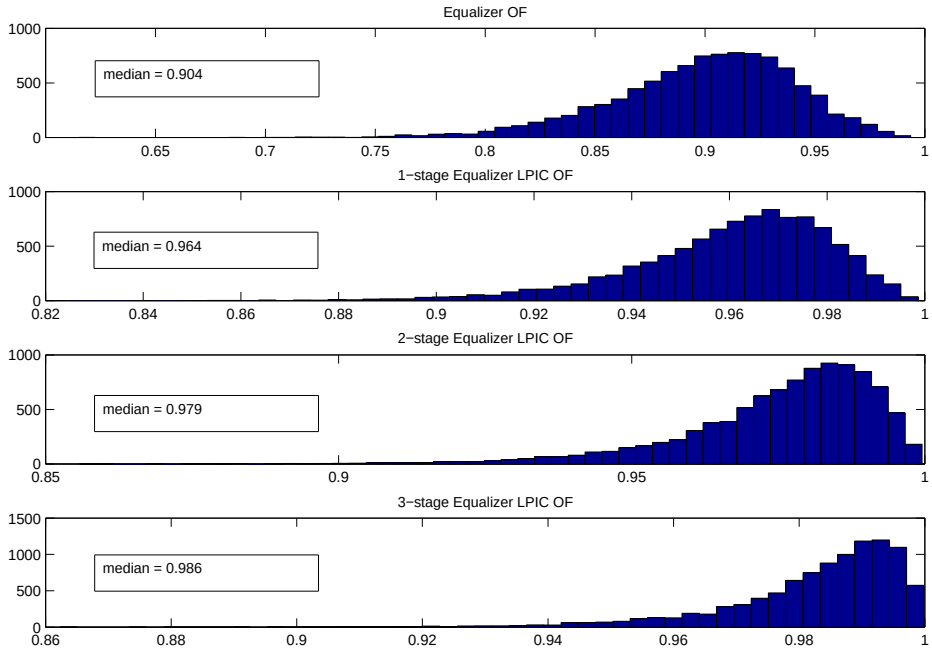


Figure 7.14: Orthogonality factor histogram of LPIC with 2-phase LMMSE chip equalizers in all stages in the Vehicular A channel with $\hat{I}_{or}/I_{oc} = 10dB$

Chapter 8

Conclusions and Future Work

In this final chapter we first compare the complexities of the proposed equalization schemes in an embedded vector processor (EVP) [5, 6]. We conclude that symbol level implementations of Griffiths and Decision-Directed LMS are in many situations significantly less complex than their chip level implementations. Then we finalize the thesis by a listing of possible future work.

8.1 Embedded Vector Processor

In this section we briefly review the features of the embedded vector processor (EVP) given in Figure 8.1 which we consider as the target implementation medium for our equalizer proposals [5, 6]. Its main advantage compared to conventional DSP devices is the ability to process large amounts of parallel vector data at a time with however a bit more limited scalar operations capability. Compared to ASICS, it is more flexible, i.e. good for functionality updates but might be slightly less optimized in terms of area and power consumption. On the other hand, it might even in some situations be more area efficient than ASICs since it has shared architectural blocks among different units whereas in ASICs components doing different tasks are mostly designed separately. Taking into account all the pros and cons it is perhaps one of the most ideal implementation platforms for UMTS development, which requires flexibility and abundant vector operations.

The main data holding units of EVP called vector registers contain 256 bits with which EVP uses the principle of vector parallelism to process multiple smaller data with vector sizes 8 bits, 16 bits or 32 bits. For instance, in UMTS modems, 8 bits is the most common unit for real or imaginary sample representations. In this case, one vector register

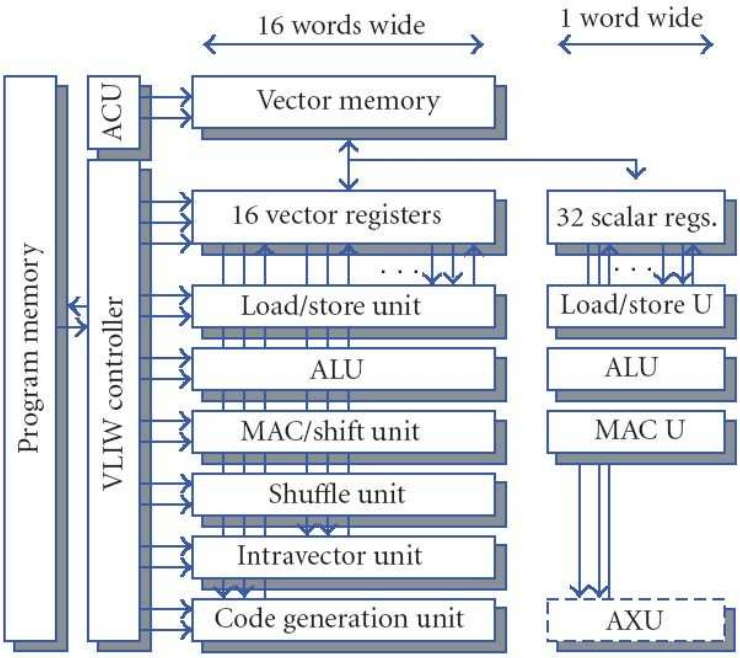


Figure 8.1: EVP Architecture (from [5] by the courtesy of authors)

can host 16 complex samples which EVP can process in one clock cycle.

In addition to the vector parallelism, EVP also uses a very long instruction word (VLIW) model which supports another parallelism among multiple vector functional units (FUs) of a single instruction multiple data (SIMD) unit, i.e. the third column of Figure 8.1. This second parallelism is perhaps the most important property to judge the complexity of an algorithm for EVP. If an algorithm can be programmed in such a way that the load is distributed as uniformly as possible on different FUs, it becomes advantageous w.r.t. to other alternatives which do not have that property. In other words, for EVP, complexity, i.e. the MIPS consumption, depends mostly on the most loaded FU. VLIW instruction may also contain instructions for scalar functional units shown in the right-hand side of Figure 8.1.

8.1.1 EVP Functional Units

Arithmetic Logic Unit (ALU) handles operations such as vector-vector additions, vector-negations, finding the maximum or minimum element of a vector, merging vectors.

Multiply and Accumulate (MAC) unit, as the name implies, is responsible for

vector-vector multiplications besides vector moving and copying tasks and taking some additions and subtractions operations load from the shoulders of ALU in computationally intensive cases.

Vector load and store (LSU) unit transfers vector register contents from/to the memory.

Vector shuffle unit (SHU) can shuffle a vector on per-element basis.

The code generation unit (CGU) is a specific block for CDMA systems which can generate 16 consecutive scrambled channelization code chips in one clock cycle.

Intra-vector operations unit (IVU) is useful for operations such as adding or subtracting the elements of a vector, finding the maximum and minimum values of a vector and their positions and applying boolean operations among elements. It is in particular useful in CDMA for despreading operations. As will be explained later, this is the distinguishing factor for equalizer algorithm complexity.

8.2 EVP Complexity of Equalizers

For complexity assessments we only consider the functionalities which have the most impact. These are data filtering, filter adaptation and despreading. Channel estimation overhead of the Griffiths algorithm or the supporting feedback functionalities of decision-directed schemes have much less complexity.

Although for simulations we considered 24 taps, for complexity evaluations we take 32 taps.

As said before, UMTS data length is 8 bits. The compliant unit complexities of the major operations and the utilized FUs are given in Table 8.1. When we map the values

Table 8.1: Units of complexity for mostly used operations

	Cycles	Functional Unit	Data Representation
Data Filtering	8	MAC	16 bits to avoid overflow
Filter Adaptation	8	MAC	16 bits to avoid overflow
Despreading one HSDPA code	1	IVU	8 bits
Walsh Hadamard Transform FWHT-16	11	MAC	first two lattices 8 bits last two lattices 16 bits

in Table 8.1 to chip level and symbol level equalizers we obtain the results in Table 8.2. In

the end we made a load balancing by transferring some of the load from IVUs to MACs. Note that reverse transfer is not possible.

Table 8.2: Cost per HSDPA period of 16 chips

	Chip Level Eq		Symbol Level Eq	
	Multiplicity	Cost	Multiplicity	Cost
Data Filtering	16	128	K_1	$8K_1$
Filter Adaptation	16	128	K_1	$8K_1$
Seperate Despreading of HSDPA codes	-	-	$32K_1$	$32K_1$
FWHT-16	1	11	-	-
Total Cost	267MAC		$16K_1$ MAC + $32K_1$ IVU $\equiv 24K_1$(MAC+IVU)	

From the obtained results we conclude that the cross-over point is $K_1 = 11$. In real life scenarios $K_1 \leq 11$ would be the case most of the time. Moreover, HSDPA terminals from categories 5 and 6 support up to 5, terminals from categories 7 and 8 support up to 10 and hence only very high end terminals from categories 9 and 10 support up to 15, i.e. more than 11 codes. In fact load balancing is a bit pessimistic consideration. In practice EVP is not only used for the equalization purpose. For other purposes also MAC is most of the time the overused and hence the more precious entity. Therefore , perhaps it is better to compare the complexity only from the MAC usage perspective. Accordingly one can conclude that symbol level equalizers are advantageous w.r.t. chip level equalizers in EVP implementation even if we use all the 15 code subspaces. Assuming that on average 5 HSPDSCH codes are used, then with the pessimistic approach symbol level equalization will have 120 (MAC+IVU) complexity which is 45% of chip level equalization. With the optimistic approach the symbol level equalization MAC complexity will be 80 units which is 30% of the chip level equalization MAC complexity. The complexity of HSDPA symbol level equalization can be further decreased by considering a subset of codes for adaptation.

8.3 Future Work

Possible extensions on the covered channel estimation schemes are

- benefiting from 3 or more pilot sequences in case of one or more SCPICH assignments during fixed beamforming,

- handling also the channel variations within the slot in high speeds,
- taking into account the correlations among FIR channel taps,
- sparsification and hybrid treatment of different taps,
- incorporating them into the context of semi-blind techniques such as in [83, 84].

Possible performance improvement or complexity reduction means on the covered adaptive chip equalizers are

- concurrent adaptation of N-Griffiths and HDD-NLMS schemes which would be particularly useful in 16-QAM mode,
- incorporating a computationally intensive scheme for better filter initialization or fading recovery,
- considering schemes for partial updating of filter weights to reduce the complexity as in [85].
- sparsification of filter weights

A resonable complete work would be to unify the proposed channel estimation and channel equalization schemes with the proposed iterative multiuser detector. Furthermore it is possible to improve the channel estimation quality along with the iterations of the receiver benefiting from the reduced interference levels over the pilot sequences. This can be combined with the coordinated adaptive estimation of the filter weights in different stages which is another open issue worth investigating.

We conclude by giving our perspective on the practical implementation issues associated with using equalizers instead of channel matched filters (Rake receivers) in the context of the considered iterative receiver. In practice, due to computation budget constraints, the allowed number of iterations will be quite limited. Due to interactions with the channel decoding unit and the higher layer processes, in some cases there might be no convincing argument to go for more than one or two iterations. Furthermore, the number of used iterations perhaps should be decided on the run. Namely, one starts iterating and passes to the next iteration if a need really occurs. With such a modular and dynamic deployment reasoning, it is imperative to put the effort to stay close to optimal till each stage, in particular in the first few stages. Therefore the first stage has to use a chip level

LMMSE equalizer. All the effort in the literature, however, has been put on using Rake receivers at all stages, focusing on optimizing the introduced weighting factors among all the stages [28, 29]. Hence another possible future work would be to compare our results with the ones in the literature in terms of both performance and complexity.

Appendix A

Stationarity Results for Oversampled Systems

In this section, we give the main results from the stationarity analysis of oversampled systems, see [86] for more details and proofs.

Consider the linear transmission system with the discrete time stationary input sequence $a[i]$ and the continuous time output $x(t)$ as

$$x(t) = \sum_{i=-\infty}^{\infty} a[i]h(t - iT) + v(t), \tag{A.1}$$

where $h(t)$ is the overall channel response which is the convolution of the transmission pulse shape and the propagation channel. Then the following results apply:

Lemma 1 *The continuous time process $x(t)$ is cyclostationary with symbol period T.*

Lemma 2 *The discrete time sequence $x[i]$ obtained by sampling $x(t)$ at symbol rate is stationary in the wide sense.*

Lemma 3 *The discrete time sequence $x[i]$ obtained by sampling $x(t)$ at an integer multiple m of symbol rate is cyclostationary with symbol period. However, when we group the m received samples for each symbol period in vectors as*

$$\boldsymbol{y}[k] = \left[\ x[km]\ x[km+1]\ \ldots\ x[km+m-1]\ \right]^T, \tag{A.2}$$

then $\boldsymbol{y}[k]$ is vector stationary; namely each of its m elements $y_j[k] = x[km+j]$ is wide sense stationary.

Appendix B

Fractionally-spaced Equalization of Polyphase Channels

In this section, we briefly discuss the channel equalization of SIMO (single input, multi output) and MIMO (multi input, multi output) systems which are applicable to the cases of oversampling with respect to the symbol period and/or reception by multiple sensors. We will only be concerned with the cases of oversampling at twice the symbol rate with one or two sensors but every derivation is also applicable to the composite of other sampling rates and other number of receive antennas, see [79].

When we sample the received signal $r(t)$ of a single source at rate $2/T$, then the resultant sequence $r[n]$ and the channel response $h[n]$ can be decomposed into their even and odd components at symbol rate as $r_e[n] = r[2n]$, $r_o[n] = r[2n+1]$, $h_e[n] = h[2n]$ and $h_o[n] = h[2n+1]$. Accordingly, FIR I/O relations between the transmitted symbol sequence $a[n]$ and the two received sequences are as $r_e[n] = a[n] * h_e[n]$ and $r_o[n] = a[n] * h_o[n]$. It is possible to equalize this polyphase channel by the multichannel setup using two FIR filters $f_e[n]$ and $f_o[n]$ s.t.

$$f_e[n] * h_e[n] + f_o[n] * h_o[n] = \delta[n-d]. \tag{B.1}$$

We will now show the equivalent 2-phase representation of this multichannel equalization setup. The z-transform of the 2-phase channel response at sampling rate $2/T$ can be written as the 2-phase composition of the z-transforms of 2 single channel responses at symbol rate as

$$\boldsymbol{H}[z] = H_e[z^2] + z^{-1}H_o[z^2]. \tag{B.2}$$

Corresponding fractionally-spaced equalizer can be written as

$$\boldsymbol{F}[z] = z^{-2}F_e[z^2] + z^{-1}F_o[z^2] \tag{B.3}$$

taking into account the compensation of one-cycle delay between the two phases, causality requirement and the correct synchronization of down-sampling instants with the symbol boundaries. Hence, polyphase equalizer-channel cascade response as shown in Fig. B.1 upper branch is equal to

$$\begin{aligned}\boldsymbol{G}_p[z] = z^{-2}(F_e[z^2]H_e[z^2] + F_o[z^2]H_o[z^2]) + \\ z^{-1}(F_o[z^2]H_e[z^2] + z^{-2}F_e[z^2]H_o[z^2]).\end{aligned} \tag{B.4}$$

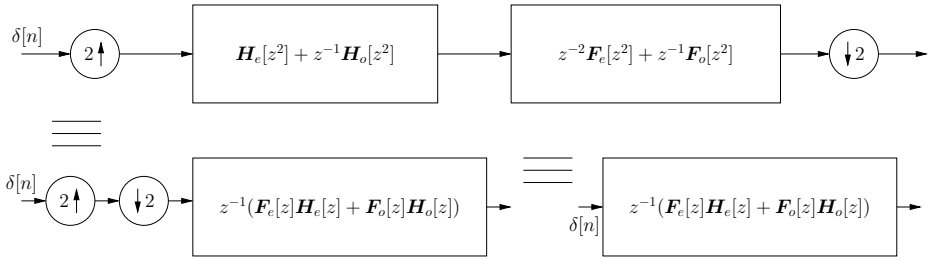


Figure B.1: Polyphase channel equalization structure

Transferring $\boldsymbol{G}_p[z]$ to the right side of the downsampler as

$$\boldsymbol{G}[z] = 1/2\ (\boldsymbol{G}_p[z^{1/2}] + \boldsymbol{G}_p[-z^{1/2}]), \tag{B.5}$$

the contribution of the second summation term in (B.4) and upsampler&downsampler cascade disappear. Hence, we are left with the symbol level total transfer function

$$\boldsymbol{G}[z] = z^{-1}(\boldsymbol{F}_e[z]\boldsymbol{H}_e[z] + \boldsymbol{F}_o[z]\boldsymbol{H}_o[z]). \tag{B.6}$$

At this moment, zero forcing equalization problem can be reformulated as

$$\boldsymbol{F}_e[z]\boldsymbol{H}_e[z] + \boldsymbol{F}_o[z]\boldsymbol{H}_o[z] = z^{-d+1}, d\ \epsilon\ \{1, 2, \ldots\} \tag{B.7}$$

which is the z-domain equivalent of (B.1) with the only difference that due to the causality constraint, the channel can only be equalized with a non-zero delay. This is due to the phase offset between multichannels in the oversampling setup which would possibly not exist for the synchronized antenna arrays.

Now that we have proved the equivalency of the polyphase and multichannel setups, we next consider the corresponding linear system of fractionally-spaced equalization (deconvolution) equations in the discrete time domain which can be represented as a multiplication of the block Toeplitz channel convolution matrix[1] with the equalizer as

$$\mathcal{T}(H_N)\boldsymbol{f}_{2M} = [\mathcal{T}(H_o)\ \mathcal{T}(H_e)]\ [\boldsymbol{f}_e^T\ \boldsymbol{f}_o^T]^T = \widetilde{\boldsymbol{e}}_{d-1} \tag{B.8}$$

where

$$\mathcal{T}(H_N) = \begin{bmatrix} h_o[0] & \dots & 0 & h_e[0] & \dots & 0 \\ h_o[1] & \ddots & \vdots & h_e[1] & \ddots & \vdots \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ h_o[N-1] & \ddots & 0 & h_e[N-1] & \ddots & 0 \\ 0 & \ddots & h_o[0] & 0 & \ddots & h_e[0] \\ \vdots & \ddots & h_o[1] & \vdots & \ddots & h_e[1] \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & h_o[N-1] & 0 & \dots & h_e[N-1] \end{bmatrix}, \tag{B.9}$$

$$\boldsymbol{f}_{2M} = \begin{bmatrix} f_e[0]\ f_e[1]\ \dots\ f_e[M-1]\ f_o[0]\ f_o[1]\ \dots\ f_o[M-1] \end{bmatrix}^T \tag{B.10}$$

and $\widetilde{\boldsymbol{e}}_{d-1}$ is the unit column delay vector having a one at the d^{th} position as

$$\widetilde{\boldsymbol{e}}_{d-1} = \begin{bmatrix} \overbrace{0}^{position\ 1} & \dots & 0 & \overbrace{1}^{position\ d} & 0 & \dots & 0 \end{bmatrix} \tag{B.11}$$

with the dimensions $(M+N-1)\times 2M$, $2M\times 1$ and $(M+N-1)\times 1$ respectively. This is a linear system of $(M+N-1)$ equations and $2M$ unknowns. For the solution to exist, the number of equations should be less than or equal to the number of unknowns,namely the system should be exactly determined or underdetermined. This is here to say that $M \geq (N-1)$ is a sufficient condition to equalize single input systems by sampling at twice the symbol rate and besides several fast algorithms exist for the solution of this kind of Toeplitz structures.

Now consider a synchronised two-seperate-input system such that composite received sequence is equal to

$$x(t) = \sum_{i=-\infty}^{\infty} a_1[i]h_1(t-iT) + a_2[i]h_2(t-iT) + v(t). \tag{B.12}$$

[1]Notice the order change of even and odd blocks due to compensation of phase offset.

Unlike the previous single input system, it is here impossible to completely cancel the other user signal with a one sensor equalizer that samples the received signal at twice the symbol rate since the linear system has $2(M+N-1)$ equations and $2M$ unknowns, namely being overdetermined. Hence, one needs to go to either higher sampling rates or use multi-sensors. However, oversampling by more than two is not feasible in practical systems due to bandwidth constraints, see [79]. Therefore, we will consider oversampling at twice the symbol rate with two sensors and proceed with derivations similar to the single input case.

Interleaving the received samples of the two sensors, $r^a[n]$ and $r^b[n]$, we obtain a vector signal at the symbol rate as[2]

$$\boldsymbol{r}[n] = \left[\ r_e^a[n]\ r_o^a[n]\ r_e^b[n]\ r_o^b[n]\ \right]^T \tag{B.13}$$

where again

$$r_e^a[n] = r^a[2n],\ r_o^a[n] = r^a[2n+1],\ r_e^b[n] = r^b[2n],\ r_o^b[n] = r^b[2n+1] \tag{B.14}$$

$$\begin{aligned} h_{1e}^a[n] = h_1^a[2n],\ h_{1o}^a[n] = h_1^a[2n+1],\ h_{1e}^b[n] = h_1^b[2n],\ h_{1o}^b[n] = h_1^b[2n+1] \\ h_{2e}^a[n] = h_2^a[2n],\ h_{2o}^a[n] = h_2^a[2n+1],\ h_{2e}^b[n] = h_2^b[2n],\ h_{2o}^b[n] = h_2^b[2n+1] \end{aligned} \tag{B.15}$$

and accordingly,

$$\begin{aligned} r_e^a[n] &= a_1[n] * h_{1e}^a[n] + a_2[n] * h_{2e}^a[n] \\ r_o^a[n] &= a_1[n] * h_{1o}^a[n] + a_2[n] * h_{2o}^a[n] \\ r_e^b[n] &= a_1[n] * h_{1e}^b[n] + a_2[n] * h_{2e}^b[n] \\ r_o^b[n] &= a_1[n] * h_{1o}^b[n] + a_2[n] * h_{2o}^b[n] \end{aligned} \tag{B.16}$$

holds. For each of the two users, the equivalent 4-phase channel transfer functions, corresponding fractionally spaced equalizers, multichannel representations of the resultant transfer functions and linear systems of fractionally-spaced equalization (deconvolution) equations in the discrete time domain can be formulated as

$$\begin{aligned} \boldsymbol{H}_1[z] &= H_{1e}^a[z^2] + z^{-1}H_{1o}^a[z^2] + z^{-2}H_{1e}^b[z^2] + z^{-3}H_{1o}^b[z^2] \\ \boldsymbol{H}_2[z] &= H_{2e}^a[z^2] + z^{-1}H_{2o}^a[z^2] + z^{-2}H_{2e}^b[z^2] + z^{-3}H_{2o}^b[z^2] \end{aligned} \tag{B.17}$$

[2] In this section superscripts ^{a} and ^{b} differentiate the sensors.

$$\begin{aligned}\boldsymbol{F}_1[z] &= z^{-4}F_{1e}^a[z^2] + z^{-3}F_{1o}^a[z^2] + z^{-2}F_{1e}^b[z^2] + z^{-1}F_{1o}^b[z^2] \\ \boldsymbol{F}_2[z] &= z^{-4}F_{2e}^a[z^2] + z^{-3}F_{2o}^a[z^2] + z^{-2}F_{2e}^b[z^2] + z^{-1}F_{2o}^b[z^2]\end{aligned} \tag{B.18}$$

$$\begin{aligned}\boldsymbol{G}_1[z] &= z^{-1}(\boldsymbol{F}_{1e}^a[z]\boldsymbol{H}_{1e}^a[z] + \boldsymbol{F}_{1o}^a[z]\boldsymbol{H}_{1o}^a[z] + \boldsymbol{F}F_{1e}^b[z]\boldsymbol{H}_{1e}^b[z] + \boldsymbol{F}_{1o}^b[z]\boldsymbol{H}_{1o}^b[z]) \\ \boldsymbol{G}_2[z] &= z^{-1}(\boldsymbol{F}_{2e}^a[z]\boldsymbol{H}_{2e}^a[z] + \boldsymbol{F}_{2o}^a[z]\boldsymbol{H}_{2o}^a[z] + \boldsymbol{F}_{2e}^b[z]\boldsymbol{H}_{2e}^b[z] + \boldsymbol{F}_{2o}^b[z]\boldsymbol{H}_{2o}^b[z])\end{aligned} \tag{B.19}$$

$$\mathcal{T}_1(H_{2N})\boldsymbol{F}_{1,4M} = \begin{bmatrix}\mathcal{T}(H_{1o}^b) & \mathcal{T}(H_{1e}^b) & \mathcal{T}(H_{1o}^a) & \mathcal{T}(H_{1e}^a) \\ \mathcal{T}(H_{2o}^b) & \mathcal{T}(H_{2e}^b) & \mathcal{T}(H_{2o}^a) & \mathcal{T}(H_{2e}^a)\end{bmatrix}\begin{bmatrix}\boldsymbol{F}_{1e}^a \\ \boldsymbol{F}_{1o}^a \\ \boldsymbol{F}_{1e}^b \\ \boldsymbol{F}_{1o}^b\end{bmatrix} = \widetilde{\boldsymbol{e}e}_{d1-1}$$

$$\mathcal{T}_2(H_{2N})\boldsymbol{F}_{2,4M} = \begin{bmatrix}\mathcal{T}(H_{2o}^b) & \mathcal{T}(H_{2e}^b) & \mathcal{T}(H_{2o}^a) & \mathcal{T}(H_{2e}^a) \\ \mathcal{T}(H_{1o}^b) & \mathcal{T}(H_{1e}^b) & \mathcal{T}(H_{1o}^a) & \mathcal{T}(H_{1e}^a)\end{bmatrix}\begin{bmatrix}\boldsymbol{F}_{2e}^a \\ \boldsymbol{F}_{2o}^a \\ \boldsymbol{F}_{2e}^b \\ \boldsymbol{F}_{2o}^b\end{bmatrix} = \widetilde{\boldsymbol{e}}_{d2-1} \tag{B.20}$$

respectively. These are underdetermined systems of $2(M + N - 1)$ equations with $4M$ unknowns and hence $M \geq (N - 1)$ is a sufficient condition to equalize 2-input systems by oversampling at twice the symbol rate with two sensors.

Appendix C

16-QAM Amplitude Estimation

Symbols from 16-QAM alphabet have real and imaginary components from the equiprobable set $\{\pm 1, \pm 3\}$. The transmitter gain, the channel gain and the receiver gain all together effectively scale the symbols by a time-varying factor, say $\mathcal{A}$.

One of the most important changes which influences almost all the newly applied HSDPA techniques is that there is practically no fast power control on HSPDSCHs. In general all the instantaneously available BS power is shared among the HSPDSCHs. Therefore instantaneous channel quality (CQ) and the received power, and hence the amplitude, might vary rapidly in time. Varying channel conditions and the power variations at the AGC (automatic gain control) module of the receiver makes the temporal power variation even higher. This makes the 16-QAM modulation amplitude estimation quite a difficult task. Past methods focusing on this problem make the equiprobability assumption on the elements of the symbol alphabet [87, 88, 89]. Since the sample support for amplitude estimation has to be very limited in order to adapt to highly changing conditions, the equiprobability assumption is not a valid assumption. In this section we propose an iterative algorithm that tries to remedy this situation.

The initialization of the algorithm is done by the fake assumption that equiprobability holds. For this first step we bring together the real and imaginary components of N soft symbol estimates of all the HSPDSCH codes in a pool of $2N$ elements. Then the second order moment, i.e. the expected power, of these elements is obtained as

$$E\left[|\hat{a}_{u,I}|^2\right] = 5\mathcal{A}^2 + \sigma^2_{n_{u,I}} = 5\mathcal{A}^2 + \frac{\sigma^2_{n_u}}{2} \tag{C.1}$$

From Equation C.1 we can obtain the first step coarse amplitude estimate as

$$\hat{\mathcal{A}}_0 = \sqrt{\frac{|\hat{a}_{u,I}|^2 - \frac{\sigma^2_{n_u}}{2}}{5}} \tag{C.2}$$

In the second step we use the coarse amplitude estimate $\hat{\mathcal{A}}_0$ and the noise-plus-interference variance estimate $\sigma^2_{n_u}$ obtained in section 4.4.1 for the purpose of separating the in-total $2N$ real and imaginary symbol samples into two bins, one containing N_1 elements from the set {-1,1} set and the other containing N_2 elements from the {-3,3} set such that $N_1 + N_2 = 2N$. Call B_1 the first bin and B_2 the second bin. The estimate $2\hat{\mathcal{A}}_0$ serves as the initial decision boundary Thr_1. The samples below Thr_1 are put to B_1 and the samples above Thr_1 are put to B_2. The typical working region of 16-QAM in HSDPA services is above the SINR level of $15dB$. Therefore the probability of elements from $\{\pm\mathcal{A}\}$ falling into B_2 and the probability of elements from $\{\pm 3\mathcal{A}\}$ falling into B_1 is as low as 10^{-4}. Then we construct a "system of two equations" as

$$\hat{\mathcal{A}}_1^2 + \frac{\sigma^2_{n_u}}{2} = x_1 \tag{C.3}$$

$$9\hat{\mathcal{A}}_1^2 + \frac{\sigma^2_{n_u}}{2} = x_2 \tag{C.4}$$

where

$$x_1 = \sum_{k=B_1} \frac{r_k^2}{N_1}, \quad x_2 = \sum_{k=B_2} \frac{r_k^2}{N_2}$$

Since the noise variance estimation from section 4.4.1 is quite reliable we plug its value in corresponding places of the above two equations. Then the next iteration signal power estimates for the real and imaginary components can be computed by using the Least Squares method as

$$\underbrace{\begin{bmatrix} 1 \\ 9 \end{bmatrix}}_{\boldsymbol{U}} \hat{\mathcal{A}}_1^2 = \underbrace{\begin{bmatrix} x_1 - \frac{\sigma^2_{n_u}}{2} \\ x_2 - \frac{\sigma^2_{n_u}}{2} \end{bmatrix}}_{\tilde{\boldsymbol{X}}_1} \Rightarrow \hat{\mathcal{A}}_1^2 = \left(\boldsymbol{U}^H \boldsymbol{U}\right)^{-1} \boldsymbol{U}^H \tilde{\boldsymbol{X}}_1 = \frac{[1 \;\; 9]}{82} \tilde{\boldsymbol{X}}_1 = \frac{x_1 + 9x_2 - 5\sigma^2_{n_u}}{82}$$

For this iteration an alternative approach could be to estimate both $\hat{\mathcal{A}}_1^2$ and $\sigma^2_{n_u}$ from the system of two equations in when there is no reliable $\sigma^2_{n_u}$ estimate from the previous stage.

The updated amplitude estimate $\hat{\mathcal{A}}_1$ after the first iteration serves for determining the new decision boundary $Thr_2 = 2\hat{\mathcal{A}}_1$. The samples are again partitioned into B_1 and B_2 bins accordingly and the new amplitude estimate $\hat{\mathcal{A}}_2$ is obtained in a way similar to the first

iteration. This process is iterated until convergence, which takes at most 5-6 iterations. The process can either be stopped when the allowed maximum number of iterations is reached or when the obtained N_1 value in two consecutive iterations are the same. The latter case means that the process has already converged and there is no advantage in iterating any more.

The explained iterative process is iterating between estimating $\hat{\mathcal{A}}_i$ and N_1 by using one for the estimation of the other. In fact the estimation of $\hat{\mathcal{A}}$ fits to the *expectation* and the estimation of N_1 (via Thr_i) fits to the *maximization* parts of the Expectation Maximization (EM) algorithm [51, 90].

In order to decrease the complexity of the iterative EM scheme one can a priori consider that even if the number of elements in {-1,1} set and {-3,3} set, i.e. N_1 and N_2, are different they cannot differ too much. Say one set contains at least 35 % of all the elements. From this assumption one can a priori calculate (decide for) two other *fixed* thresholds Thr_{low} and Thr_{high}. The elements below Thr_{low} always stay in B_1 and the elements above Thr_{high} always stay in B_2. Only the elements between Thr_{low} and Thr_{high} are allowed to change bins during the EM iterations. Sorting the elements falling between Thr_{low} and Thr_{high} at the initialization stage can further decrease the complexity of the process. Because when the elements are sorted, determining the elements from the dynamic set that fall below the new Thr_i will be just sliding the cursor to left and right. This concept is schematically demonstrated in Figure C.1.

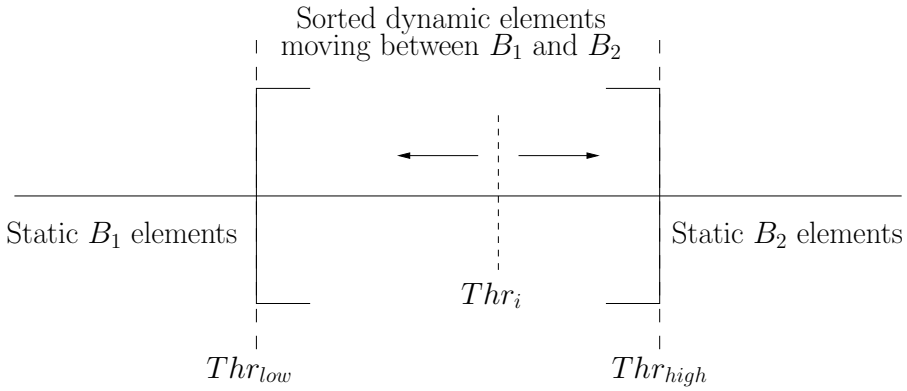


Figure C.1: Narrowed Search Region

Appendix D

Quantification of SINR Gains From Using Symbol Nonlinearities

LMMSE receiver is optimum in the mean square error sense among all the linear receivers [73]. However it is quite complex to implement in UMTS FDD mobile terminals since it needs the code and the amplitude knowledge of all the active users. Furthermore LMMSE solution changes for every chip due to the existence of aperiodic scrambling sequence. The *chip-level* LMMSE equalizer-correlator structure is a suboptimal but much simpler alternative which is derived by modeling the scrambler as a stationary random sequence [16, 17]. It is highly considered for near-future systems such as high speed downlink packet access service (HSDPA) [12]. Therefore, there is an active interest for investigating its performance. Chaufray et.al did an asymptotic analysis for its baud-rate implementation, which is valid for large, i.e more than 128, spreading factors (SFs) and for equalizer lengths equal to the SF [91]. The signal to interference plus noise ratio (SINR) formula obtained from the analysis in [16, 17] is simpler and it covers all SFs, all equalizer lengths and also the fractionally-spaced implementations. In this chapter we expand this SINR analysis by including two nonlinearities, hard decision and MMSE estimator, which refine the linear symbol estimates.

D.1 Downlink Transmitter and Receiver Model

Fig.D.1 shows the discrete time baseband communications model of the UMTS FDD downlink. We define {user identifier, user symbol index, chip index, sample index} as $\{k,\ n,\ l,\ \tilde{l}\}$ and {upsampler, downsampler, user spreading factor} as $\{\uparrow,\ \downarrow,\ L_k\}$. User

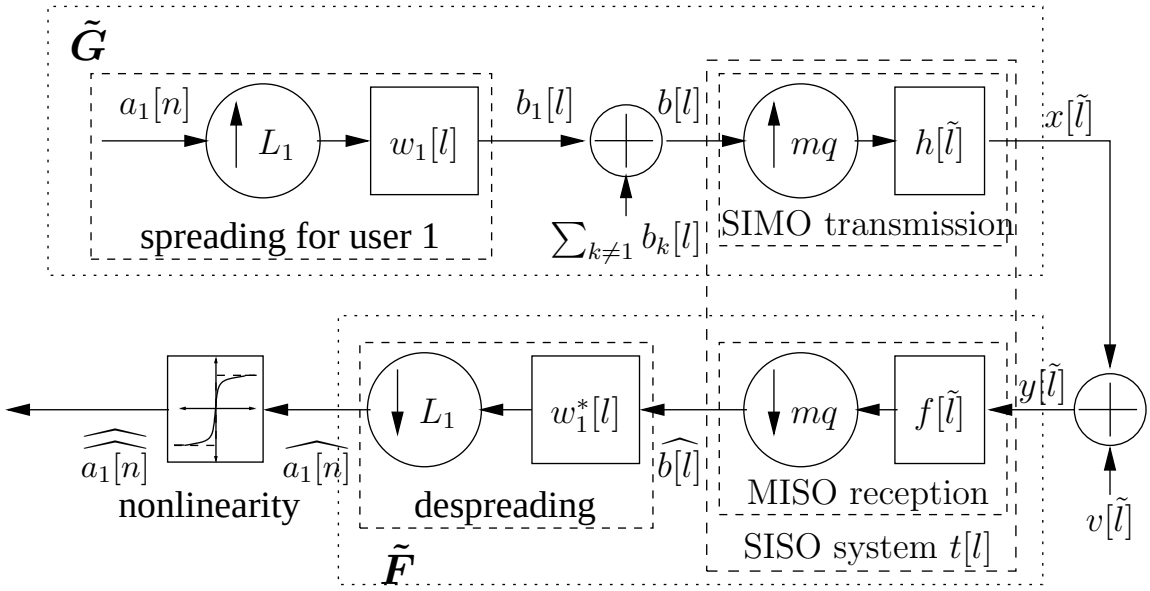


Figure D.1: Downlink communications model

spreading codes are generated by multiplying the periodic, unit energy, orthogonal Walsh-Hadamard user channelization codes $c_k[l \ mod \ L_k]$ with the BS-specific, aperiodic, random, unit amplitude, complex scrambling sequence $s[l]$ as $w_k[l] = c_k[l \ mod \ L_k]s[l]$. User of interest has the index $k = 1$. When there is no beamforming, not only the intended user chip sequence but also those of other users connected to the same BS, i.e $b[l] = \sum_k b_k[l]$, pass through a common chip-rate finite impulse response (FIR) channel $\boldsymbol{h}$ of length P which turns into a *polyphase channel* (a single input multi output (SIMO) system with memory) of length mqP in the presence of multiple (q) antennas or oversampling (by an integer factor m) at the mobile side. Continuous time counterpart of this discrete channel is the overall convolution of the root-raised-cosine pulse shape that has 0.22 roll-off factor (rrc-0.22), the propagation channel and the antialiasing filter at the receiver front end. The received signal is the superposition of the channel-distorted BS signal $x[\tilde{l}]$ and an additional *noise plus intercell interference* term $v[\tilde{l}]$ which in practice is assumed to be approximately *white*: $y[\tilde{l}] = x[\tilde{l}] + v[\tilde{l}]$. We first equalize the $\boldsymbol{h}$ channel by passing the received signal through a linear *unbiased* LMMSE $\boldsymbol{f}$ filter, without loss of generality, of length equal to the channel length mqP and afterwards through a downsampling operation (decimation) by a factor of mq. This combined operation is equivalent to a multi input single output (MISO) system with memory. Then, the resultant signal, the estimate of the synchronous BS chip sequence

$\widehat{b[l]}$, is passed through a correlator to obtain the linear $\widehat{a_1[n]}$ estimates. In the last stage, we refine the linear estimates by passing them through a nonlinear functionality such as hard decision or MMSE estimator.

D.2 SINR of LMMSE Equalizer-Correlator

The chip rate single input single output (SISO) *overall* FIR channel $\boldsymbol{t}$ of length $(2P-1)$ is the mq times downsampled form of the convolution of the $\boldsymbol{h}$ and $\boldsymbol{f}$ FIR filters ($\boldsymbol{t} = (\boldsymbol{h} * \boldsymbol{f}) \downarrow mq$). The central tap $t(P)$ which is equal to 1 due to *unbiasedness* is the orthogonal useful signal carrying part where the correlator synchronizes and the rest $(t(i), i \in \{1, \ldots, P-1, P+1, \ldots, 2P-1\})$ are the non-orthogonal interference carrying taps.

The SINR of $\widehat{a_1}$ symbol estimate, index n dropped, can be written as:

$$\begin{aligned} SINR &= \frac{|t(P)|^2 L_1 \sigma_{b_1}^2}{(\|\boldsymbol{t}\|^2 - |t(P)|^2)\sigma_b^2 + \sigma_{v_c}^2} = \frac{L_1 \sigma_{b_1}^2}{(\frac{1}{\gamma} - 1)\sigma_b^2 + \sigma_{v_c}^2} \\ &= \frac{\sigma_{a_1}^2}{\sigma_{n_1}^2} \end{aligned} \tag{D.1}$$

where $\{\sigma_{b_1}^2, \sigma_{a_1}^2, \sigma_{v_c}^2, n_1, \sigma_{n_1}^2\}$ represent {user of interest chip power, user of interest symbol power, colored noise variance equal to $\|\boldsymbol{f}\|^2\sigma_v^2$, noise plus interference at the output of the correlator, variance of n_1}. The term

$$\gamma = |t(P)|^2 / \|\boldsymbol{t}\|^2 = 1/\|\boldsymbol{t}\|^2 \tag{D.2}$$

is the *orthogonality factor* which has been treated in the literature for the Rake receiver variants [30, 31]. We also define the filter output residual interference factor as

$$\kappa = \|\boldsymbol{t}\|^2 - |t(P)|^2 = \|\boldsymbol{t}\|^2 - 1 \tag{D.3}$$

D.3 Combined Analysis with Nonlinearities

Linear symbol estimates can be partitioned into real and complex parts:

$$\widehat{a_1} = a_1 + n_1 = a_1^R + n_1^R + j(a_1^I + n_1^I) = \chi_1 + j\psi_1.$$

When the symbols belong to $\boldsymbol{W} = \{W_1, \ldots, W_M\}$ alphabet with $W_m = \mho_m + j\,\aleph_m$ elements, MMSE symbol estimator can be derived as:

$$\begin{aligned} \widehat{\widehat{a_1}}(\widehat{a_1}) &= E\{a_1 \mid \widehat{a_1}\} = \sum_{m=1}^{M} W_m\, P(W_m \mid \widehat{a_1}) \\ &= \frac{\sum_{m=1}^{M} W_m\, P(\widehat{a_1} \mid W_m)}{\sum_{m=1}^{M} P(\widehat{a_1} \mid W_m)}. \end{aligned} \tag{D.4}$$

It has been shown that multiuser interference at the output of multiuser receivers is approximately Gaussian distributed [37, 36]. In this case,

$$P(\widehat{a_1} \mid W_m) = \frac{1}{\pi\sigma_{n_1}^2} e^{-\frac{(\chi_1 - \mho_m)^2}{\sigma_{n_1}^2}}\; e^{-\frac{(\psi_1 - \aleph_m)^2}{\sigma_{n_1}^2}}. \tag{D.5}$$

Considering only QPSK constellation for the rest of the paper and plugging (D.5) in (D.4), we obtain the hyperbolic tangent estimator as:

$$\widehat{\widehat{\widehat{a_1^{thyp}}}} = \frac{\alpha_1}{\sqrt{2}} \tanh(\sqrt{2}\frac{\alpha_1}{\sigma_{n_1}^2}\chi_1) + j\;\frac{\alpha_1}{\sqrt{2}} \tanh(\sqrt{2}\frac{\alpha_1}{\sigma_{n_1}^2}\psi_1).$$

Note that $\sigma_{n_1}^2$ can be estimated, for example, as the power obtained by despreading with an unused code in the system. In order to practically implement this estimator, we still need to first estimate the symbol *power* β_1 and then the symbol *amplitude* α_1. By block averaging, β_1 can be estimated as

$$\widehat{\beta_1} = \frac{\sum_{n=1}^{N} \mid \widehat{a_1[n]} \mid^2}{N} - \sigma_{n_1}^2 = \frac{\Upsilon_1}{N} - \sigma_{n_1}^2 \tag{D.6}$$

where N is the number of symbols in the *power estimation interval*. Since Υ_1 is the sum of squares of $2N$ independent Gaussian random variables, half with mean $\pm a_1^R$ and half with $\pm a_1^I$, all having the variance $\sigma_{n_1}^2/2$, it has a *non-central chi-squared* distribution with $2N$ degrees of freedom and the noncentrality parameter $(N\sigma_{a_1}^2)$. By *central limit theorem*, for large N (say $N \geq 30$), it can be approximated by a Gaussian distribution. The mean and variance of this approximate Gaussian distribution are equal to [92]

$$\mu_{\Upsilon_1} = N(\sigma_{a_1}^2 + \sigma_{n_1}^2)\,; \sigma_{\Upsilon_1}^2 = N\sigma_{n_1}^2(\sigma_{n_1}^2 + 2\sigma_{a_1}^2).$$

Therefore $\hat{\beta}_1$ is also Gaussian distributed with mean and variance

$$\mu_{\widehat{\beta_1}} = \sigma_{a_1}^2\,; \sigma_{\widehat{\beta_1^2}} = \frac{\sigma_{n_1}^2(\sigma_{n_1}^2 + 2\sigma_{a_1}^2)}{N}.$$

However the left tail of this distribution goes into the negative region which does not make sense for a power expression and hence the correct distribution is the left-truncated Gaussian distribution with an impulse at 0 equal in amplitude to the cumulative distribution of the truncated part:

$$f(\widehat{\beta}_1) = \frac{1}{\sqrt{2\pi}\,\sigma_{\widehat{\beta}_1}} e^{-\frac{(\widehat{\beta}_1 - \mu_{\widehat{\beta}_1})^2}{2\sigma^2_{\widehat{\beta}_1}}} + Q(\frac{\mu_{\widehat{\beta}_1}}{\sigma_{\widehat{\beta}_1}})\delta(\widehat{\beta}_1).$$

From this, one can obtain the probability density function (pdf) of the real and imaginary *amplitudes* $\breve{\alpha}_1$ as

$$\begin{aligned} g(\breve{\alpha}_1) &= 4\breve{\alpha}_1 f(2\breve{\alpha}_1{}^2)\ ;\ \breve{\alpha}_1 = \widehat{\alpha_1}/\sqrt{2} = \sqrt{\widehat{\beta}_1/2} \\ g(\breve{\alpha}_1) &= \frac{4\breve{\alpha}_1}{\sqrt{2\pi}\,\sigma_{\widehat{\beta}_1}} e^{-\frac{(2\breve{\alpha}_1{}^2 - \mu_{\widehat{\beta}_1})^2}{2\sigma^2_{\widehat{\beta}_1}}} + Q(\frac{\mu_{\widehat{\beta}_1}}{\sigma_{\widehat{\beta}_1}})\delta(\breve{\alpha}_1). \end{aligned}$$

Having obtained the amplitude pdfs, we can estimate the interference plus noise variance at the output of the hyperbolic tangent estimators as:

$$\begin{aligned} \sigma^2_{n_{1thyp}} &= 2\int_{0^-}^{\infty} g(\breve{\alpha}_1) \int_{-\infty}^{\infty} |\ \alpha_1^R - \breve{\alpha}_1 tanh(\frac{\breve{\alpha}_1 x}{\sigma^2_{\breve{n}_1}})\ |^2 \\ &\quad \mathcal{N}_x(\alpha_1^R, \sigma^2_{\breve{n}_1})\ dx\, d\breve{\alpha}_1 \end{aligned} \tag{D.7}$$

where $\sigma^2_{\breve{n}_1} = \sigma^2_{n_1}/2$ due to the symmetry of real and imaginary parts and $\mathcal{N}_x(\alpha_1^R, \sigma^2_{\breve{n}_1})$ is the normal distribution of x with {mean,variance} $= \{\alpha_1^R, \sigma^2_{\breve{n}_1}\}$. Inner integral calculates the interference plus noise variance for each particular amplitude estimate. Outer integral weights the variances over the amplitude estimate pdf. Since the result is valid for the real or the imaginary part, we multiply it by 2. The $\sigma^2_{n_1 hard}$ variance at the hard-decision output can be similarly calculated by replacing $tanh(\breve{\alpha}_1 x/\sigma^2_{\breve{n}_1})$ in (D.7) with $sign(x)$. Finally, plugging the estimated output noise-plus-interference variances $\sigma^2_{n_1 hard}$ and $\sigma^2_{n_1 thyp}$ in place of $\sigma^2_{n_1}$ in (D.1), hard decision and hyperbolic tangent estimator SINR values can be obtained.

For the sake of simpler presentation from this point on we only consider the hard decisions.

The analysis done in this section can be useful for nonlinear multiuser detection (MUD) purposes. However the real time implementation of the whole analysis can be considered complicated. Therefore here we propose a simpler off-line alternative. The eventual required value is in fact the ratio $\iota = \frac{\sigma^2_{n_1}}{\sigma^2_{n_1 hard}} = \frac{SINR_2}{SINR_1}$ which we call here as the nonlinear SINR gain. This variable in turn depends only on N and on the average $SINR_1$

value at the hard decision input. $SINR_1$ value can be obtained, for example, from (D.6) as $\widehat{\beta}_1/\sigma^2_{n_1}$. Once N is fixed for a particular application, one can obtain the analysis plots of ι vs $SINR_1$ and prepare a look-up table or fit it to a polynomial of sufficient order, say 2. Then the only requirement for a real time implementation would be to obtain the average SINR value.

D.4 Simulations and Conclusions

We model 8 codes that carry QPSK symbols with $L_k = 16$ and equal variance, i.e an HSDPA scenario with 50% loading factor. We consider the static form of ITU Vehicular-A *propagation* channel. We obtain the continuous time *transmission* channel by convolving the *propagation* channel with rrc-0.22 pulse shape. We pass to the discrete time model by first sampling the continuous time channel and then truncating from two sides the tails that spread more than four chips far away from the centers of the channel components corresponding to the convolutions of the pulse shape with the first and the last propagation channel paths. The resultant $\boldsymbol{h}$ channel length becomes $P = 19$ due to the UMTS chip rate of 3.84 Mchips/sec. At the receiver side, an oversampling factor of $m = 2$ and one receive antenna $q = 1$ are considered.

We assume that we only know $\boldsymbol{h}$. To be fully compliant with practical implementations, all the other parameters are estimated. We randomly generate $\boldsymbol{h}$ and user symbols 100 times for each E_b/N_o value and take the average of the SINR results to obtain Figure D.2. We estimate symbol powers and amplitudes over a block length of N=150.

As expected, hyperbolic tangent outputs have the highest SINR values, hard decision performing $1dB$ to $2dB$ worse. There is a large performance gap between the linear scheme and the two nonlinear schemes which increases when one goes to very large or very small E_b/N_o regions. The nonlinear SINR gains of hard decisions and hyperbolic tangent nonlinearities and their second order polynomial approximation plots are also shown on Figure D.2.

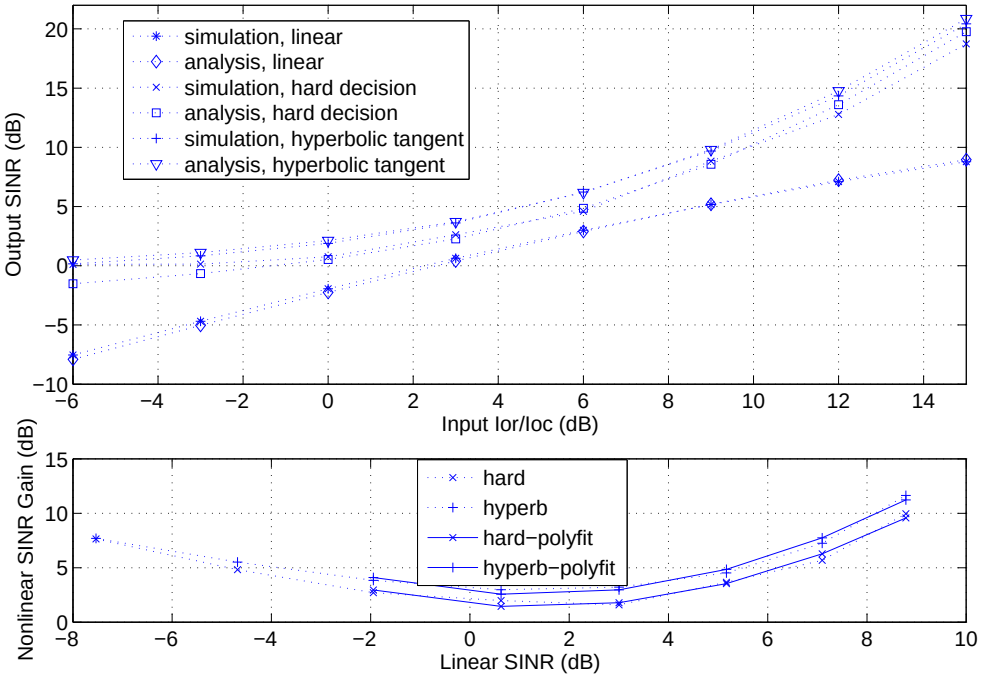


Figure D.2: SINR vs $\hat{I}_{or}/I_{oc}$ results, Vehicular A channel, P=19

Appendix E

Résumé de thèse

E.1 Introduction

Cette thèse a pour objet l'étude des communications lien descendant (*downlink*) du mode FDD (Frequency Division Duplexed, division de fréquence duplexée) de UMTS.

La station de base est la source de transmissions pour sa cellule. Les signaux transmis depuis différentes cellules sont distingués les uns des autres par l'attribution de différents codes de brouillage pseudo-aléatoires (*pseudo-random scrambling codes*) qui sont répétés pour chaque *frame* UMTS de 38400 puces, et donc sont connus comme codes à long recouvrement (*long* overlay codes).

L'accès multiple des utilisateurs d'une même cellule logique est réalisé par un schéma CDMA qui utilise des codes orthogonaux courts de "canalisation" (*short orthogonal channelization codes*) provenant de différents niveaux de l'arbre du code OVSF illustré figure E.1. Chaque niveau contient des codes correspondant aux colonnes de la transformation de Walsh-Hadamard (WHT) avec les bonnes tailles. Un code de "canalisation" associé à un utilisateur est utilisé péridioquement pour la transmission de chaque symbole.

Le modèle de transmission lien descendant *baseband* du mode UMTS-FDD avec support de HSDPA est illustré figure E.2.

Au niveau de l'émetteur, le premier groupe de séquences de symboles modulés K_1 (i.i.d QPSK ou 16-QAM) $\{a_1[n], a_2[n], \ldots, a_{K_1}[n]\}$ qui appartiennent à la transmission HSDPA sont tout d'abord "upsamplés" par un facteur 16 et ensuite convolués avec leurs codes de "canalisation" *unit-amplitude* respectifs $\{\boldsymbol{c}_{16,i_o}, \boldsymbol{c}_{16,i_o+1}, \ldots, \boldsymbol{c}_{16,i_o+K_1-1}\}$ illustrés à la figure E.1. Tous les symboles HSPDSCH ont la même énergie et le même schéma de modulation. Le deuxième groupe de transmission multi-vitesse (multi-rate)

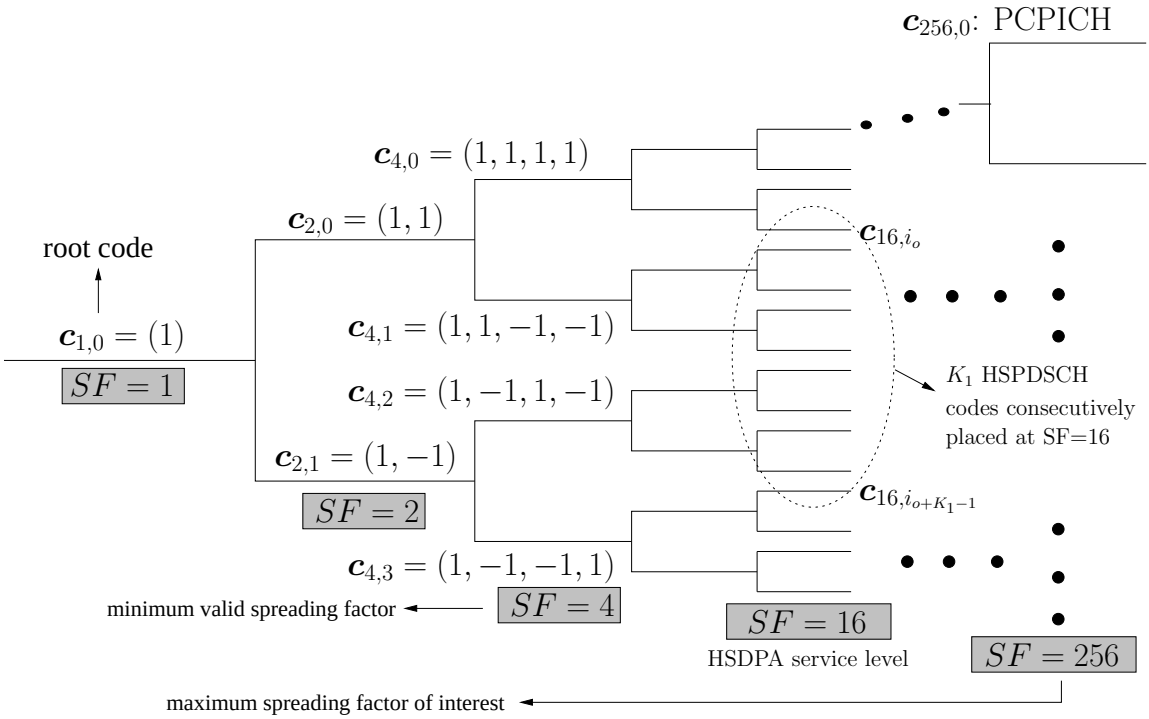


Figure E.1: Schéma partiel de l'arbre du code OVSF

$\{\tilde{a}_1[n_1], \tilde{a}_2[n_2], \ldots, \tilde{a}_{K_2}[n_{K_2}]\}$ représentant les canaux physiques dédiés (Dedicated Physical CHannels, DPCHs), HSSCCHs et d'autres canaux de contrôle sont de la même manière "upsamplés" et convolués avec leur codes de "canalisation" respectifs $\left\{\boldsymbol{c}_{L_1,i_1}, \boldsymbol{c}_{L_2,i_2}, \ldots, \boldsymbol{c}_{L_{K_2},i_{K_2}}\right\}$.

La séquence somme de toutes les séquences "chip" générées est multipliée avec la séquence de brouillage apériodique "unit-energy" et "BS-spécifique" $s[l]$. La séquence "BS chip" résultante effective $b[l]$ est transmise au canal.

E.2 Estimation de canal

Comme illustré dans la figure E.3, le système fournit au moins deux séquences pilotes. DPCH fournit des pilotes temps multiplexés avec les données utiles (payload data). PCPICH fournit des pilotes continus.

En l'absence de "beamforming" Tx, les canaux DPCH et PCPICH sont les mêmes (mis à part un facteur réel dû à l'offset d'énergie Tx). En conséquence, la plupart du temps PCPICH est utilisé pour l'estimation de canaux DPCH. En présence de "beamforming" Tx

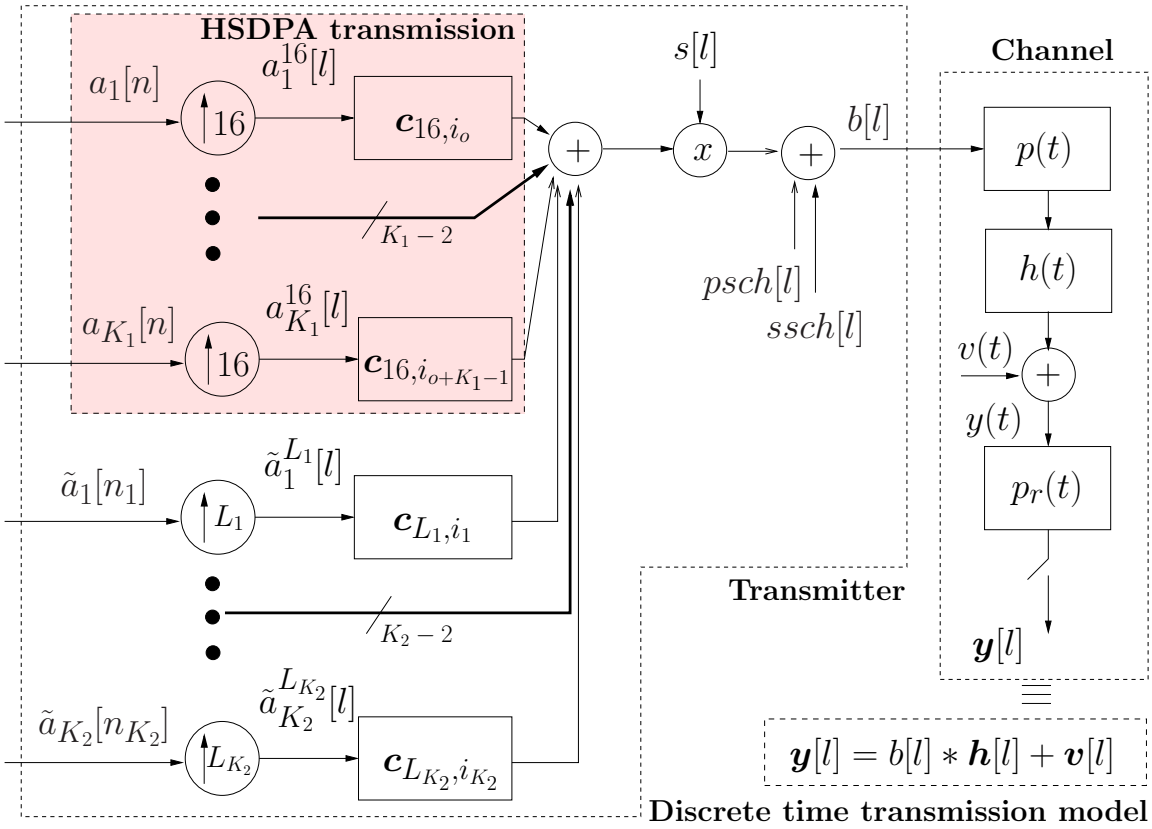


Figure E.2: Modèle de transmission lien descendant *baseband* de UMTS

le canal DPCH est en général différent du canal PCPICH. En conséquence, PCPICH n'est généralement pas considéré comme utile pour l'estimation de canal DPCH. La question est : quelle séquence pilote doit-on choisir pour l'estimation de canal.

Doit-on utiliser des pilotes dédiés ?

- Ils permettent d'estimer le canal pendant le "beamforming" Tx dédié.
- Des séquences pilotes courtes peuvent cependant apporter une mauvaise précision de l'estimation.
- De plus, la détection de variations de canal sur les données nécessite de l'interpolation et/ou de la prédiction et donc augmente la complexité.

Doit-on utiliser des pilotes courants ?

- La fourniture continue de "chips" d'entraînement amène une haute précision de l'estimation

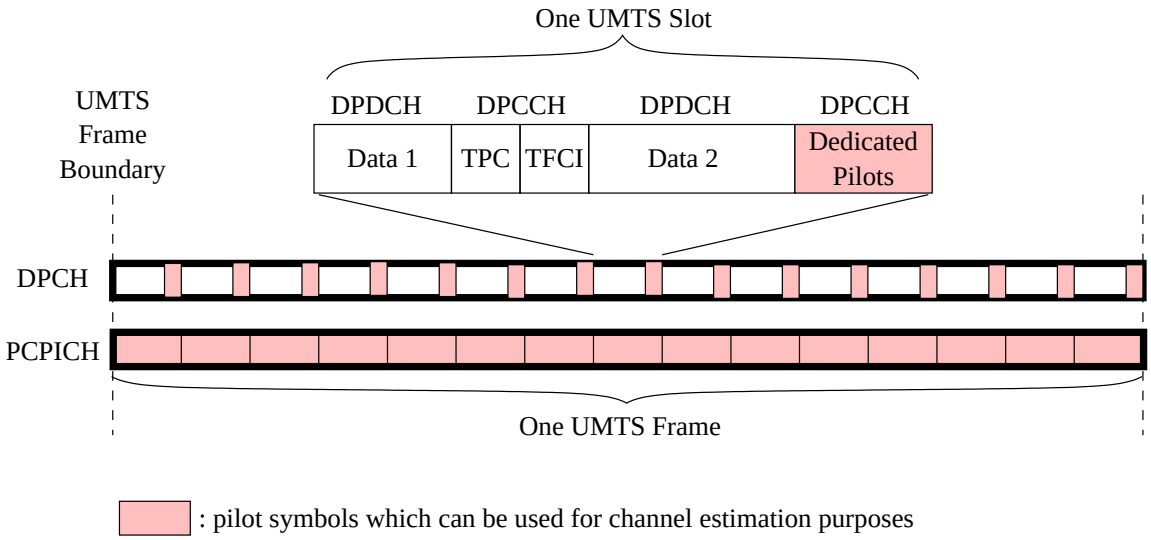


Figure E.3: Séquences pilotes disponibles

et une détection des variations de canal simple, ce qui est une bonne chose.

- Cependant, il ne peut pas être utilisé seul en présence de “beamforming” Tx dédié.

Ne peut-on utiliser les deux ?

Des résultats de tests “field” propriétaires montrent que c’est possible. Parce-que les deux canaux de propagation véhiculant PCPICH et DPCH sont hautement corrélés (effet Doppler partagé, délais de chemin, etc.). C’est donc une opportunité d’exploiter les deux sources d’entraînement pour l’estimation de canal dédié à l’utilisateur, ce qui est spécialement bénéfique pendant le mode “beamforming” Tx.

La méthode d’estimation de canal que nous proposons a deux étapes.

E.2.1 Première étape : estimation de canal “moindres carrés”

Cela correspond à la phase de mesure du mécanisme de filtrage de Kalman que nous mentionnerons plus tard.

Le canal FIR associé à DPCH peut être représenté par

$$\boldsymbol{h}_d(n) = [h_{d,0}(n), \ldots, h_{d,k}(n), \ldots, h_{d,N-1}(n)]^T$$

L’estimé LS “in-slot” du canal associé à DPCH se résoud de

$$\hat{\boldsymbol{h}}_d(n) = \arg\min_{\boldsymbol{h}_d} \|\boldsymbol{Y}(n) - \boldsymbol{S}_d(n)\boldsymbol{h}_d(n)\|^2 \approx \beta_d^{-1}\boldsymbol{S}_d^H(n)\boldsymbol{Y}(n)$$

- n: index de "slot"
- k: index de canal "tap", N: index de canal "tap" maximal
- $\boldsymbol{S}_d(n)$: matrice de bloc de Hankel comprenant la séquence pilote "chip" dédiée
- $\boldsymbol{Y}[n]$: vecteur de symboles reçu durant le "slot" n
- β_d: énergie du pilote dédié pendant le "slot" n
- L'estimation des canaux $\boldsymbol{h}_c(n)$ associés à PCPICH est faite de la même manière

E.2.2 Deuxième étape : combiner les stratégies (de filtrage)

Après la première étape, nous avons l'estimation LS (least-square, moindres carrés) des canaux dédiés et courants bien que le but final soit l'estimation du canal dédié.

À partir de ce moment, nous traitons chaque "tap" de canal indépendamment.

Au premier instant nous considérons la combinaison des estimations LS obtenues à la première étape dans un sens LMMSE non biaisé pour décroître les variances d'erreur des estimations des "tap" de canal DPCH. Un autre degré de liberté provient de la dynamique temporelle du canal. En ajustant les variations du "tap" du canal DPCH à des modèles autorégressifs (AR) d'ordre suffisant, on peut appliquer un filtrage de Kalman (KF) sur les estimations des canaux dédiés obtenus à chaque "slot". Un processus AR de premier ordre est suffisant quand on veut faire correspondre la variation du canal et le déplacement (shift) et la bande passante (BW) Doppler. L'approche la plus propre est de bénéficier des deux dimensions.

E.2.3 Trois approches différentes au filtrage de Kalman

- D'abord faire une combinaison LMMSE et ensuite un filtrage de Kalman
  - sous-optimal, mais est le plus simple.
  - un seul état dans le filtre de Kalman
- Filtrage de Kalman conjoint des deux

  - l'approche optimale dans le sens MMSE
  - deux états dans le filtre de Kalman

- D'abord appliquer un filtre de Kalman séparément aux "taps" des canaux DPCH et PCPICH et ensuite faire une combinaison MMSE des résultats
  - sous-optimal à cause du bruit coloré aprés KF
  - deux état de Kalman et donc même complexité que Kalman conjoint (non préférable)

- En général toutes les méthodes sont implémentables
  - la complexité est proportionnelle au nombre de "taps" de canal
  - le filtrage de Kalman pour chaque "tap" a au plus deux états

E.2.4 Combinaison non-biaisée LMMSE des estimations de canal LS

Définir $h_{c,k}(n)$ sur la base de $h_{d,k}(n)$ comme

$$h_{c,k}(n) = \alpha_k h_{d,k}(n) \quad + \quad \underbrace{x_{c,k}(n)}_{\text{erreur d'estimation MMSE}} \quad , \quad k = 0, 1, \ldots, N-1$$

Nous arrivons au problème de combinaison ULMMSE

$$\arg\min_{\boldsymbol{f}_k} \mathrm{E}|h_{d,k}(n) - \boldsymbol{f}_k \left[\hat{h}_{d,k}(n)\ \hat{h}_{c,k}(n)\right]^T|^2 \quad \text{s.t. } \boldsymbol{f}_k[1\ \alpha_k]^T = 1$$

Pour implémenter le schéma, nous avons besoin de connaître certains paramètres.

- Les estimations LS des "taps" de canal DPCH ont une variance d'erreur identique (même chose pour les "taps" de canal PCPICH). On peut donc estimer $\sigma^2_{e_{d,k}}$ et $\sigma^2_{e_{c,k}}$ en surestimant légèrement les longueurs des canaux DPCH et PCPICH et en prenant l'énergie à la queue des estimées des canaux en tant que variances d'erreur

- α_k et $\sigma^2_{x_{c,k}}$ peuvent être estimés par une technique simple de "covariance matching" ou par l'algorithme EM dans le contexte KF que nous expliquerons plus tard.

E.2.5 Filtrage de Kalman

Les équations d'espace-état (state-space) et les paramètres du modèle pour KF peuvent être résumés par

$$\boldsymbol{h}_k(n) = \begin{bmatrix} h_{d,k}(n) \\ h_{c,k}(n) \end{bmatrix} : \text{Vecteur d'Etat Courant (Present State Vector)}$$

$\boldsymbol{h}_k(n+1) = \rho_k \boldsymbol{h}_k(n) + \boldsymbol{B}\boldsymbol{u}(n)$: Processus de Transition d'Etat (State Transition Process)

$$\boldsymbol{B} = \sqrt{1-\left|\rho_k^2\right|} \begin{bmatrix} 1 & 0 \\ \alpha_k & \sqrt{1-\frac{\sigma^2_{h_{d,k}}}{\sigma^2_{h_{c,k}}}\left|\alpha_k^2\right|} \end{bmatrix} : \text{Gain d'Entrée (Input Gain)}$$

$$\boldsymbol{u}(n) = \begin{bmatrix} \Delta h_{d,k}(n) \\ \Delta h_{c,k}(n) \end{bmatrix} : \text{Vecteur d'Entrée (Input Vector)}$$

$$\boldsymbol{R_{uu}} = \begin{bmatrix} \sigma^2_{\Delta h_{d,k}} & 0 \\ 0 & \sigma^2_{\Delta h_{c,k}} \end{bmatrix} : \text{Covariance d'Entrée (Input Covariance)}$$

$\boldsymbol{Q} = \boldsymbol{B}\boldsymbol{R_{uu}}\boldsymbol{B}^H$: Covariance du Bruit du Processus (Process Noise Covariance)

$$\boldsymbol{R_{ww}} = \begin{bmatrix} \sigma^2_{e_{d,k}} & 0 \\ 0 & \sigma^2_{e_{c,k}} \end{bmatrix} : \text{Covariance du Bruit de Mesure (Measurement Noise Covariance)}$$

E.2.6 Estimation des Paramètres du Modèle de Filtre de Kalman

- Pour KF, $\{\rho, \boldsymbol{Q}, \boldsymbol{R}_{ww}\}$ sont nécessaires
- Estimation de $\{\rho, \boldsymbol{Q}\}$: via l'algorithme EM dans le framework de lissage de Kalman "fixed-lag" (fixed-lag Kalman smoothing) (avec seulement un seul pas de délai) qui est illustré dans la figure E.4

L'algorithme de maximisation d'espérance (Expectation Maximization algorithm, EM) est la méthode privilégiée quand le problème étudié souffre de données incomplètes et que la solution nécessite à la fois de compléter ces données et d'estimer certains paramètres. L'algorithme itère entre la phase E (E-phase) qui est le calcul "log-likelihood" espéré de la donnée manquante (imputée) en utilisant à la fois les données observées et les estimées actuelles des paramètres et la phase M (M-phase) qui calcule la valeur la plus probable (maximum likelihood value, ML) des paramètres en conditionnant les données imputées comme

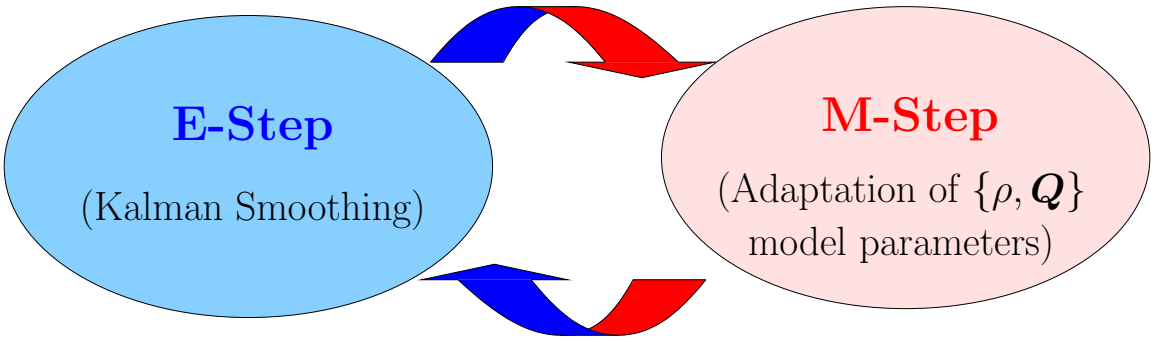


Figure E.4: Une Itération de l'algorithme de maximisation d'espérance (Expectation Maximization Algorithm)

si elles étaient correctes. Dans cette section, les paramètres de canal $\boldsymbol{h}(n) = [h_d(n)\ h_c(n)]^T$ sont les données manquantes et $\{\hat{\rho}, \hat{\boldsymbol{Q}}\}$ sont les seules estimées de paramètres nécessaires. Adapter le mécanisme EM au contexte du filtrage de Kalman nécessite aussi du lissage (smoothing) dans la phase E.

E.2.7 Simulations et Conclusions

Dans cette section, nous donnons des résultats de simulation à la figure E.5 dans un cas où le coefficient de corrélation temporelle ρ et le coefficient de corrélation spatiale normalisé α sont tous les deux fixés à 0.9 pour toutes les "taps" de canal.

Table E.1: Simulation Settings

Parameters	**Settings**
Chip rate	3.84 Mcps $\Rightarrow T_c = 260$ns
Sampling rate	twice chip rate
Modulation scheme	QPSK
PCPICH power	10% of the BS power (-10dB)
DPCH power	5% of the BS power (-13dB) effectively 20% when beamforming gain is taken as 4
DPCH slot occupation	20%
OCNS power	(3GPP compliant) remaining BS power randomly distributed to a number of codes at SF level 128
Transmission pulse shape	Root-Raised-Cosine (rrc) with roll-off factor 0.22
Channel model	AR(1) process, WSSUS over multi-paths
Channel power delay profile	ITU Vehicular A

Table E.2: Vehicular A Channel Power Delay Profile

Relative Delay [ns]	0	310	710	1090	1730	2510
Relative Mean Power [dB]	0	-1	-9	-10	-15	-20

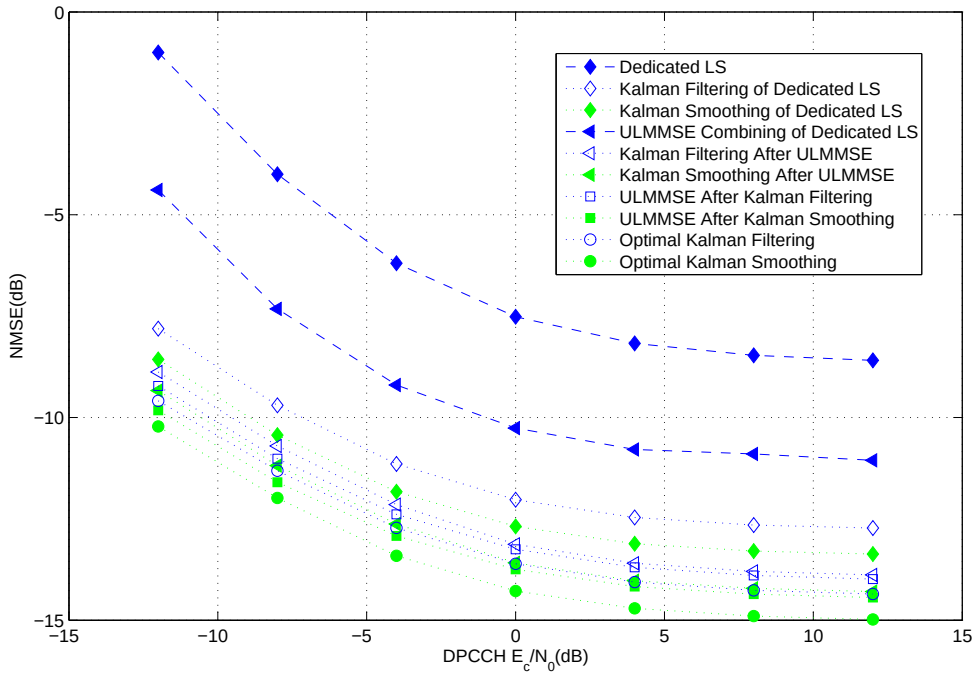


Figure E.5: $\rho = 0.9$ (90km/h), $\zeta = 0.9$

- la combinaison LMMSE des estimées de canal moindres carrés (Least Squares) apporte une amélioration pour des “cross correlations” raisonnablement élevées
- le filtrage de Kalman sur les estimées LS DPCH est meilleur
- le filtrage de Kalman joint est le meilleur (optimal au sens MMSE) mais dans le même temps le plus complexe
- Utiliser les deux séquences pilotes est aussi attractif pour le cas “non-beamforming”, spécialement pour les bords des cellules, et donc peut être utilisé pour augmenter la couverture
- Le lissage (smoothing) (sur les passes avant et arrière, forward and backward pass)

augmente les performances en rapport avec le filtrage (seulement dans la passe avant) dans tous les cas.

- Le traitement séparé des “taps” est la clé pour réduire la complexité

E.3 Annulation des interférences

Nous commençons par donner une relation utile. Comme montré sur la figure E.6 avec un example simplifié, un lien descendant (downlink) CDMA “multirate” dans le sens de multiples facteurs d’éparpillement (spreading) peut être aussi bien représenté par une pseudo-transmission “multicode” avec un seul facteur d’éparpillement. Dans un but de détection multi-utilisateurs cette relation est utilisée pour modéliser le système avec un facteur d’éparpillement de 256. En mettant simplement une banque de corrélateurs (correlator bank) à ce niveau, on détecte la présence ou l’absence de pseudo-symboles en comparant les énergies de sortie à un niveau d’énergie seuil de bruit. Tant que ces pseudo-symboles sont traités linéairement, il n’est pas important de savoir où se trouve le symbole réel.

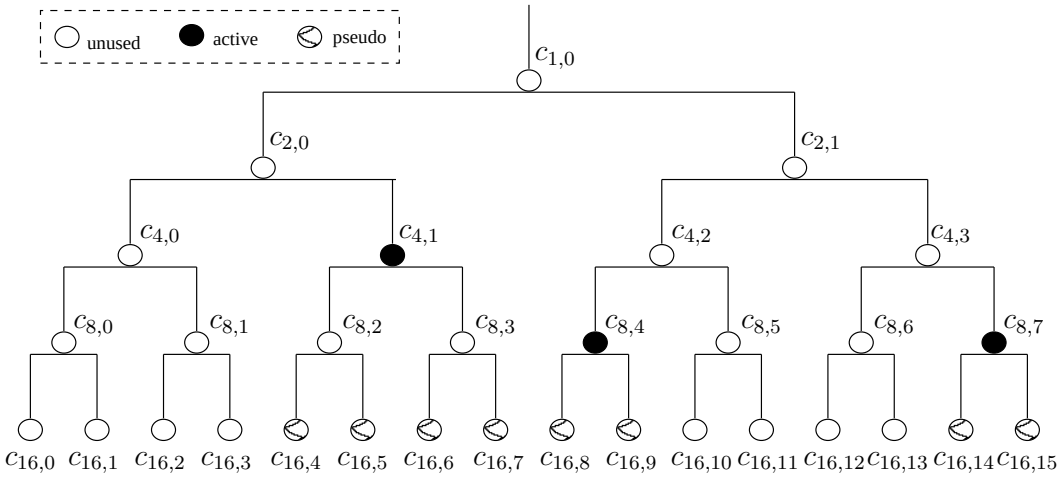


Figure E.6: Équivalence de “Active-Multirate” et “Pseudo-Multicode”

E.3.1 État de l’art des récepteurs

Le détecteur multi-utilisateurs optimal en termes de taux d’erreur sur symbole (symbol error rate, SER) minimal est l’estimation de séquence à probabilité maximale (Maximum Likelihood Sequence Estimation, MLSE) qui est une procédure de recherche

exhaustive sur l'alphabet des symboles de toutes les séquences transmises possibles de tous les utilisateurs avec le critère de minimisation

$$\hat{\boldsymbol{d}}_{ML} = \arg \min_{\boldsymbol{d} \in \mathcal{X}^{MK}} \left\| \boldsymbol{Y} - \overbrace{\boldsymbol{\mathcal{T}}(\boldsymbol{h})\boldsymbol{S}\boldsymbol{C}}^{\tilde{\boldsymbol{G}}} \underbrace{\boldsymbol{\mathcal{A}}\boldsymbol{d}}_{\boldsymbol{A}} \right\|^2 \tag{E.1}$$

où $\mathcal{X}$, K, $\boldsymbol{Y}$, $\tilde{\boldsymbol{G}}$, $\boldsymbol{\mathcal{T}}(\boldsymbol{h})$, $\boldsymbol{S}$, $\boldsymbol{C}$, $\boldsymbol{d}$, $\boldsymbol{\mathcal{A}}$ et $\boldsymbol{A}$ dénotent respectivement l'alphabet des symboles[1], le nombre d'utilisateurs, le bloc "sample" reçu portant la sortie de canal de transmission de M périodes de symbole, la fonction de transfert du canal niveau symbole, la matrice de convolution du canal, la matrice de brouillage diagonale, la matrice des codes de "canalisation" diagonale par blocs, MK vecteurs de symboles désirés "unit-amplitude", la matrice diagonale des amplitudes de symbole et le vecteur des symboles diagonaux "amplitude-scaled". Puisque ce critère est contraint par un alphabet fini, il est NP-hard[2]. Une des approches sous-optimale mais plus simple est de relâcher la contrainte d'alphabet fini par une fausse correspondance de $\boldsymbol{d}$ depuis l'ensemble fini $\mathcal{X}^{MK}$ vers $\mathbb{C}^{MK}$ qui transforme le problème LS non-linéaire de (E.1) dans un problème LS linéaire

$$\hat{\boldsymbol{A}}_{LS} = \arg \min_{\boldsymbol{A} \in \mathbb{C}^{MK}} \left\| \boldsymbol{Y} - \tilde{\boldsymbol{G}}\boldsymbol{A} \right\|^2 \tag{E.2}$$

dont la solution est

$$\hat{\boldsymbol{A}}_{LS} = \tilde{\boldsymbol{F}}_{Dec}\boldsymbol{Y} = \left(\overbrace{\tilde{\boldsymbol{G}}^H \tilde{\boldsymbol{G}}}^{\boldsymbol{R}} \right)^{-1} \tilde{\boldsymbol{G}}^H \boldsymbol{Y} = \boldsymbol{R}^{-1}\boldsymbol{X} \tag{E.3}$$

où $\boldsymbol{X}$ et $\boldsymbol{R}$ dénotent respectivement les estimées des symboles de sortie des banques de filtre de correspondance utilisateur unique (single user matched filter, SUMF) et la matrice de "cross-correlation" des symboles.

Bien qu'il "déconvolve" complètement $\tilde{\boldsymbol{G}}$, le décorrélateur $\tilde{\boldsymbol{F}}_{Dec}$ amplifie le terme de bruit $\boldsymbol{V}$. Une meilleure approche est l'estimateur LMMSE qui modélise $\boldsymbol{A}$ comme un vecteur aléatoire Gaussien et résoud le critère de coût

$$\tilde{\boldsymbol{F}}_{LMMSE} = \arg_{\tilde{\boldsymbol{F}}} \min_{\boldsymbol{A} \in \mathbb{C}^{MK}} \mathbb{E} \left(\tilde{\boldsymbol{F}}\boldsymbol{Y} - \boldsymbol{A} \right) \left(\tilde{\boldsymbol{F}}\boldsymbol{Y} - \boldsymbol{A} \right)^H \tag{E.4}$$

avec la solution

$$\tilde{\boldsymbol{F}}_{LMMSE} = \left(\tilde{\boldsymbol{G}}^H \tilde{\boldsymbol{G}} + \sigma_v^2 \boldsymbol{\mathcal{A}}^{-2} \right)^{-1} \tilde{\boldsymbol{G}}^H \tag{E.5}$$

[1] représentant le cas simple d'une même constellation pour tous les utilisateurs

[2] un problème de décision qui est au moins aussi dur que n'importe quel problème dont la solution est *vérifiable* avec une complexité polynomiale

qui en plus du cas du décorrélateur nécessite aussi l'amplitude des symboles $\boldsymbol{\mathcal{A}}$.

À la fois le décorrélateur et le récepteur LMMSE sont très complexes en raison du fait qu'ils nécessitent des opérations d'inversion de matrice avec une complexité en $O(M^3K^3)$. Par conséquent, des approximations de rang réduit de l'opération d'inversion de matrice ont été beaucoup examinées dans la littérature avec des techniques itératives. Nous nous basons seulement sur la famille dite annulation d'interférences parallèle (Parallel Interference Cancellation, PIC) qui est la contrepartie des itérations de Jacobi pour les solutions itératives de systèmes linéaires d'équations.

Le LPIC conventionnel correspond à l'utilisation d'itérations de Jacobi pour les solutions de systèmes linéaires d'équations. En scindant l'expression $\boldsymbol{R}$ de (E.3) en deux parties $\boldsymbol{I}$ et $(\boldsymbol{R}-\boldsymbol{I})$, on peut arriver à la solution de décorrélation itérative suivante[3]

$$\hat{\boldsymbol{A}}_{LS}^{(i)} = (\boldsymbol{I}-\boldsymbol{R})\hat{\boldsymbol{A}}_{LS}^{(i-1)} + \boldsymbol{X} \tag{E.6}$$

Les itérations convergent tant que le rayon spectral $\rho(\boldsymbol{I}-\boldsymbol{R})$ est plus petit que 1, ce qui n'est pas garanti[4].

E.3.2 Récepteur à expansion polynomiale

Au premier instant, nous considérons le remplacement de $\tilde{\boldsymbol{G}}^H$ par $\tilde{\boldsymbol{F}}$ avec des égalisateurs MMSE-ZF en tant que filtres "chip rate". Définir $\boldsymbol{M} = \tilde{\boldsymbol{F}}\tilde{\boldsymbol{G}}$ tel que $\boldsymbol{X} = \tilde{\boldsymbol{F}}\boldsymbol{Y}$ amène aux équations itératives depuis les expansions polynomiales suivantes

$$\begin{aligned}
\widehat{\boldsymbol{A}}^{(-1)} &= 0 \;\; ; \;\; i \geq 0 \\
\widehat{\boldsymbol{A}}^{(i)} &= (\boldsymbol{I}-\boldsymbol{M})\,\widehat{\boldsymbol{A}}^{(i-1)} + \boldsymbol{X} \\
&= \widehat{\boldsymbol{A}}^{(i-1)} + \tilde{\boldsymbol{F}}(\boldsymbol{Y} - \tilde{\boldsymbol{G}}\widehat{\boldsymbol{A}}^{(i-1)})
\end{aligned}$$

En dernière étape, nous remplaçons $\tilde{\boldsymbol{F}}$ par $\tilde{\boldsymbol{F}}^i$ pour refléter les optimisations LMMSE locales à l'itération i. L'architecture résultat avec les blocs "symbol rate" et les blocs équivalents "chip rate" sont donnés figure E.7 et figure E.8 respectivement.

Dans le cas où les fonctionnalités "symbol feedback" $\mathcal{D}^{(i)}$ sont des matrices identité, on peut déplacer les opérations d'addition vers l'avant des blocs de débrouillage (descrambling) ce qui amène au modèle "chip iterating" de la figure E.9.

[3] de manière similaire, en scindant $\boldsymbol{M}$ de (1.38) pour le récepteur LMMSE

[4] $\rho(\boldsymbol{X}) = \max\{|\lambda|, \lambda \in \Lambda(\boldsymbol{X})\}$ où $\Lambda(\boldsymbol{X})$ est la matrice des valeurs propres (eigenvalue matrix) de $\boldsymbol{X}$

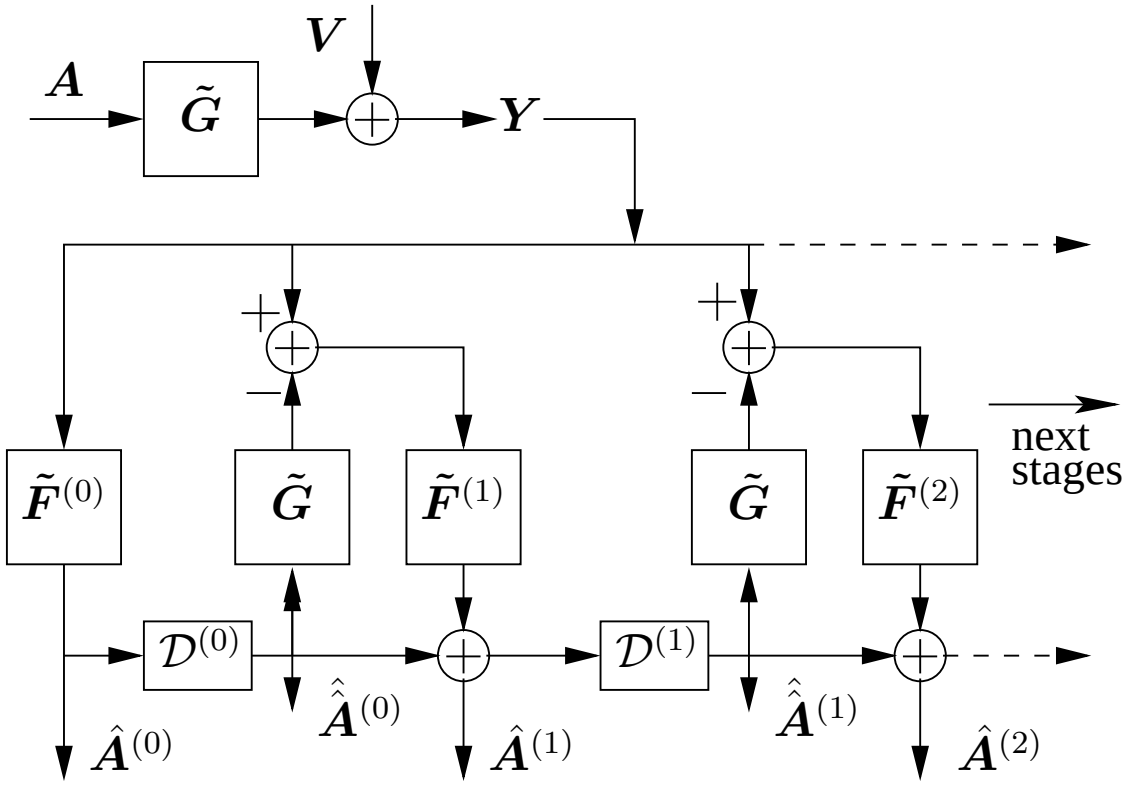


Figure E.7: Récepteur PE avec les blocs "symbol rate"

L'étape suivante consiste à s'émanciper de la dépendance au code dans l'architecture. Pour cela nous utilisons le fait que la cascade "descrambling-despreading-respreading-rescrambling" est approximativement une multiplication scalaire avec le facteur de charge (loading factor) *effectif*. En remplaçant simplement ces blocs avec ce facteur d'échelle, on obtient l'architecture de récepteur de la figure E.10. Puisqu'il s'agit maintenant d'un système complètement convolutif, il est facile d'optimiser progressivement les filtres dans toutes les étapes en fixant les filtres des étapes initiales. Quand des décisions difficiles (hard decisions) sont utilisées, $\mathcal{C}_l$ est ajusté pour refleter uniquement le sous-espace de code traité linéairement.

E.3.3 Simulations

Les valeur de simulation sont les suivantes :

- Canal de propagation : "ITU Vehicular A"

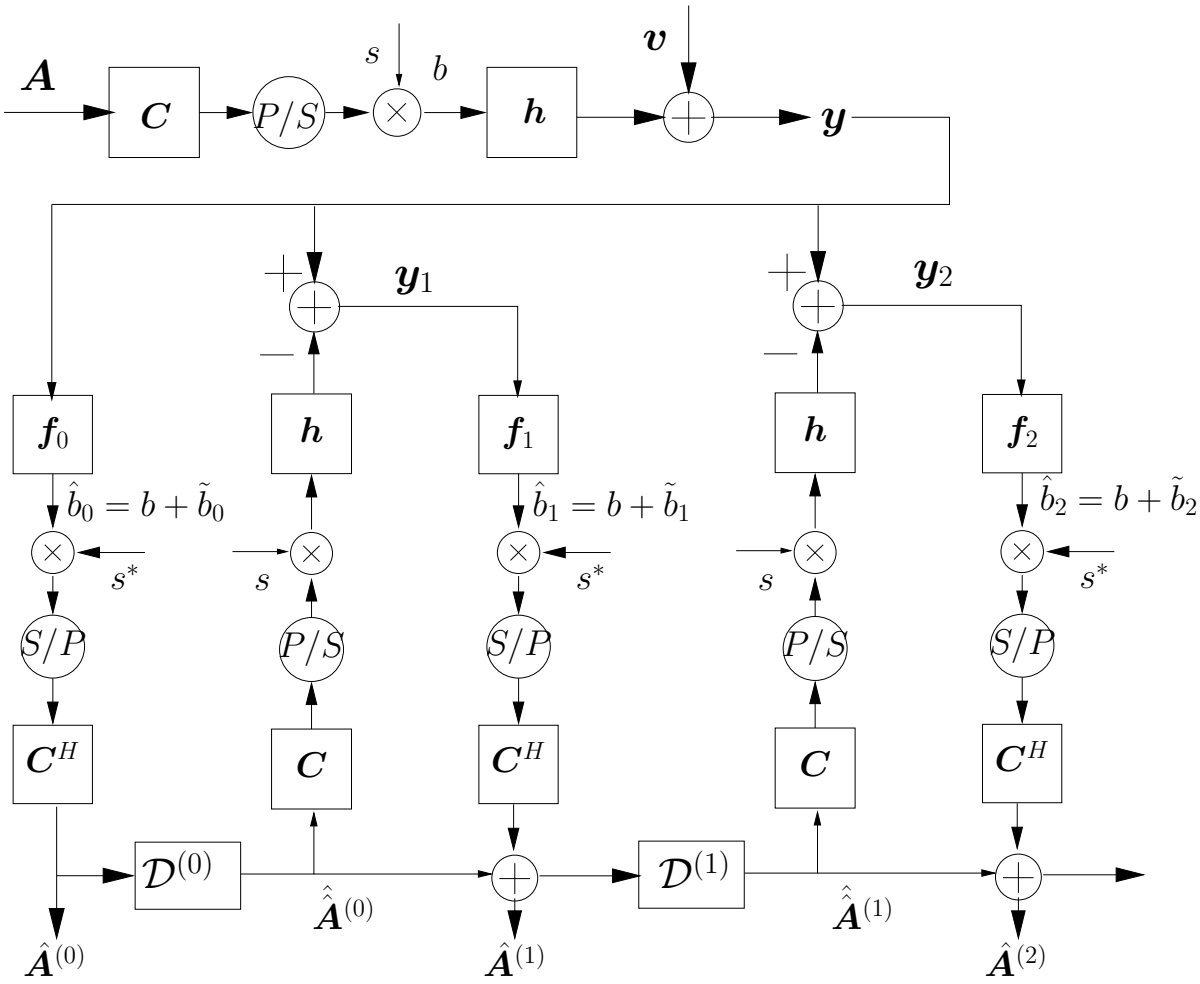


Figure E.8: Format ouvert de récepteur PE avec les blocs équivalents “chip rate”

- “sampling” à deux fois le “chip rate”
- symboles QPSK
- 5 HSDSCH codes à SF-16: chacun consommant 8% du “BS power”
- “Pilot tone” à SF-256: consommant 10% du “BS power”
- PCCPCH à SF-256: consommant 4% du “BS power”
- 46 canaux à SF-256: consommation totale 46% du “BS power”
- Charge effective à SF-256 = 50%

Les figures E.11 et E.12 montrent les résultats SINR de différents schémas d'utilisation des filtres pour l'utilisation respectivement du "feedback" identité et des décisions difficiles (hard decisions).

Une autre mesure intéressante est le facteur d'orthogonalité (orthogonality factor, OF) qui est le rapport de l'énergie de canal effective utile (l'énergie du "tap" réel à l'instant du délai d'égalisation) et de l'énergie du canal effective totale. Un exemple démonstratif avec comme canal effectif la cascade du canal et du filtre de canal assorti est donné à la figure E.13.

Les figure E.14, E.15 et E.16 montrent respectivement la progression des itérations en utilisant des récepteurs "all Rake", en utilisant un égalisateur à la première étape (et "Rake" aux étapes suivantes) et en utilisant des égalisateurs à toutes les étapes. Il est clair que d'adapter les égalisateurs à chaque étape donne les meilleurs résultats.

E.3.4 Conclusions

- On s'attend à ce que des opérations LMMSE locales aident à approcher le LMMSE global
- Une adaptation séparée à chaque étape est très utile dans le traitement linéaire (quand seulement un code est connu)
- Quand des décisions difficiles (hard decisions) sont appliquées, un récepteur Rake après l'égalisation de première étape "chip" est un choix judicieux (quand des codes multiples sont connus, comme dans HSDPA)
- Dans la pratique
  - seulement un petit nombre d'itérations autorisé
  - Le nombre d'itération est décidé en cours de route
  - Nécessité de fournir l'effort de rester proche de l'optimal à chaque étape (en particulier les quelques premières étapes)
  - La première étape doit utiliser un égalisateur LMMSE "chip level"
  - ... de préférence à des méthodes existantes qui investissent seulement dans "Rake" à toutes les étapes

E.4 Perspectives

Estimation de canal

- bénéficier de 3 ou plus séquences pilotes dans le cas d'une ou de plusieurs assignations SCPICH pendant un "beamforming" fixe,
- prendre en compte la corrélation entre les "taps" de canal FIR,
- "sparsification" et traitement hybride de différents "taps",
- les incorporer dans le contexte de techniques semi-aveugles

Détection multi-utilisateurs

- Extensions triviales pour inclure aussi l'annulation d'interférences inter-cellulaires
- Intégration des schémas proposés d'estimation de canal et d'égalisation de canal[5] de manière coordonnée
- Amélioration de l'estimation de canal en même temps que les itérations du récepteur en bénéficiant des niveaux réduits d'interférences

[5]Cette thèse a aussi couvert quelques implémentations adaptatives de l'égalisateur "chip" qui ne sont pas expliquées dans ce résumé

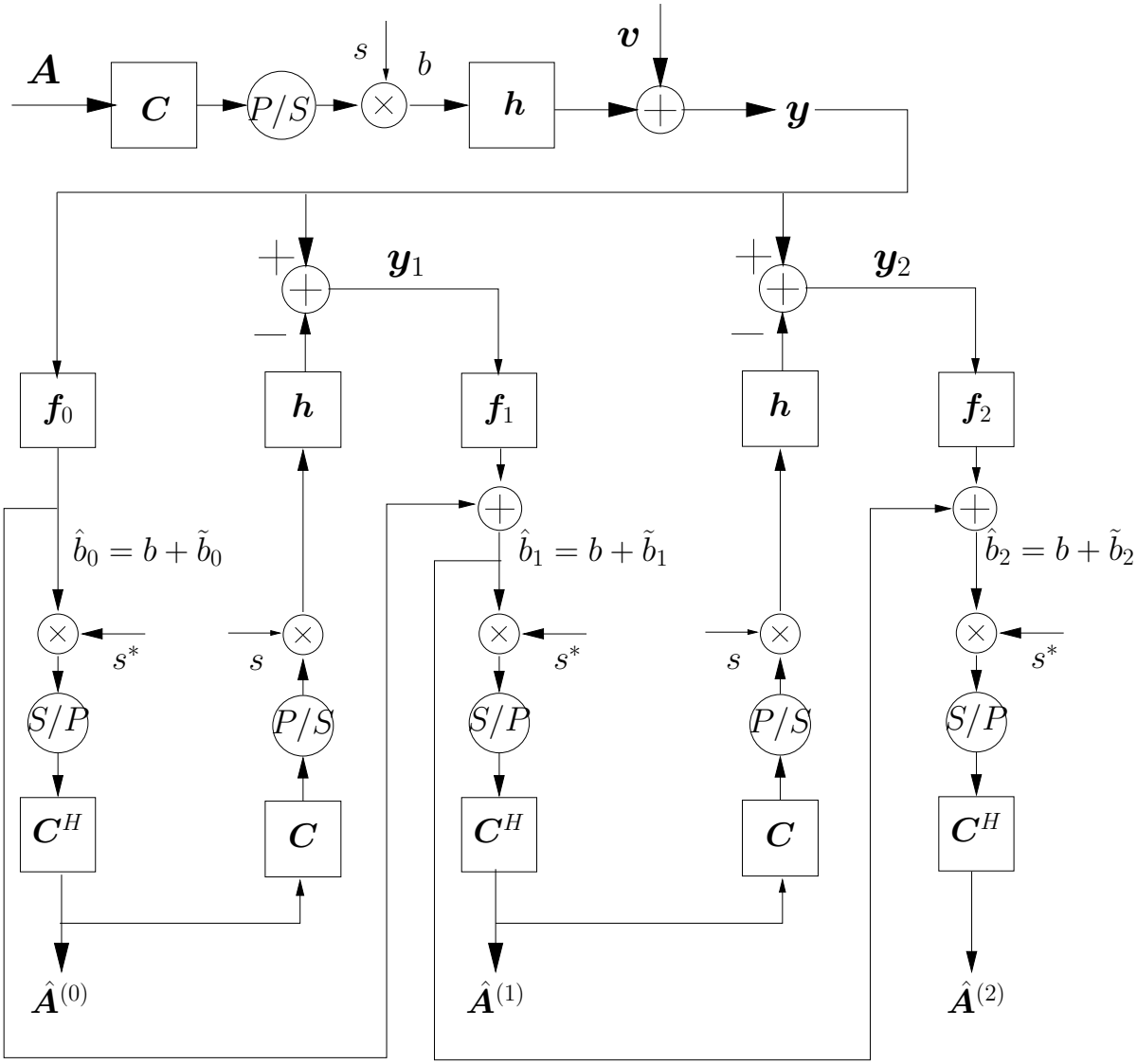


Figure E.9: Modèle “chip iterating” équivalent en cas de “feedback” identité

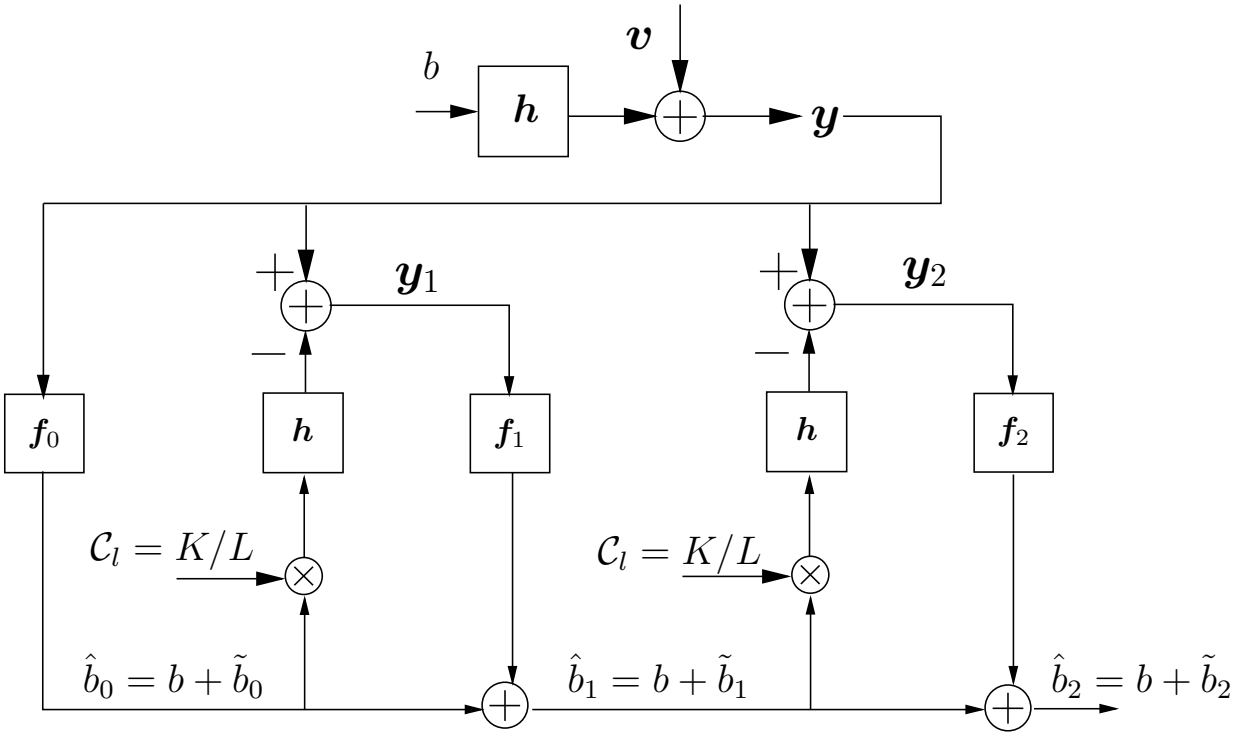


Figure E.10: Modèle convolutif approché “chip rate”

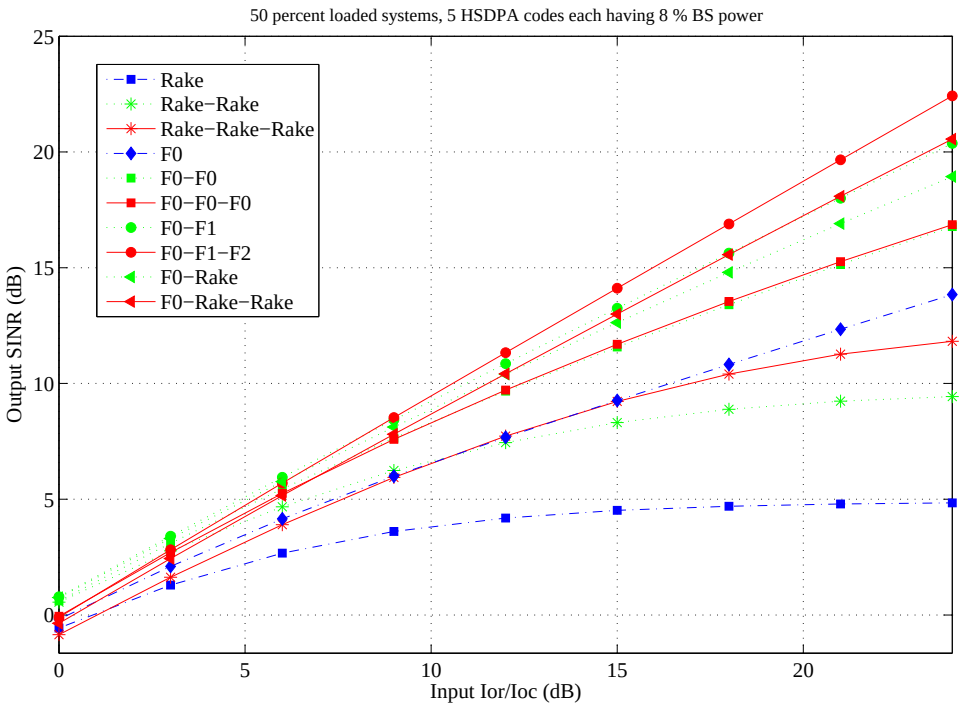


Figure E.11: SINR vs $\hat{I}_{or}/I_{oc}$ linear decisions results, VA-ch, N=19

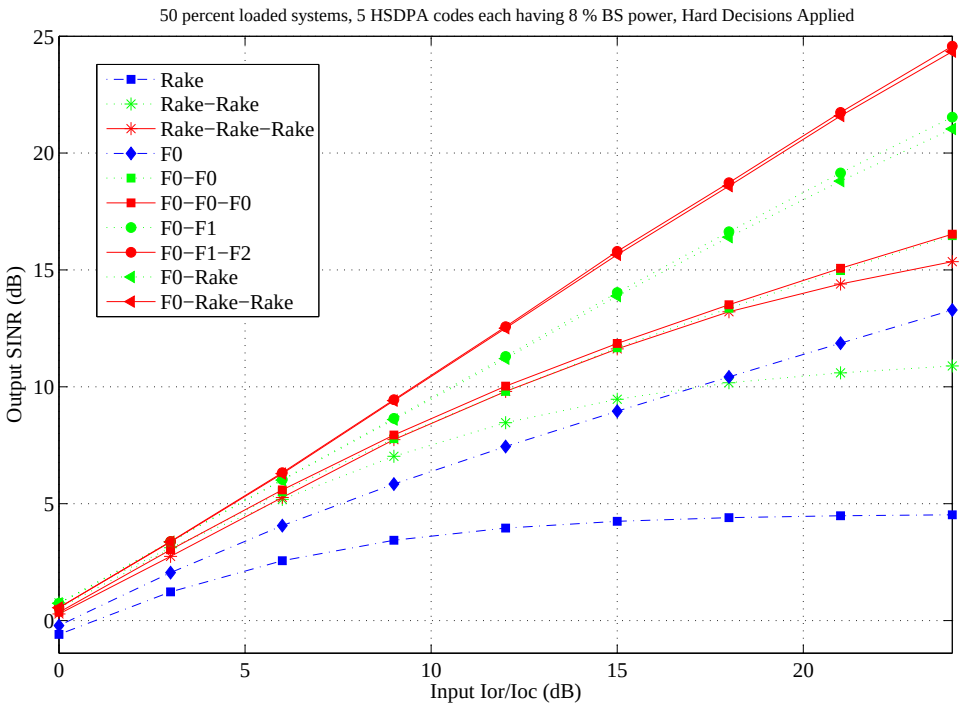


Figure E.12: SINR vs $\hat{I}_{or}/I_{oc}$ hard decisions results, VA-ch, N=19

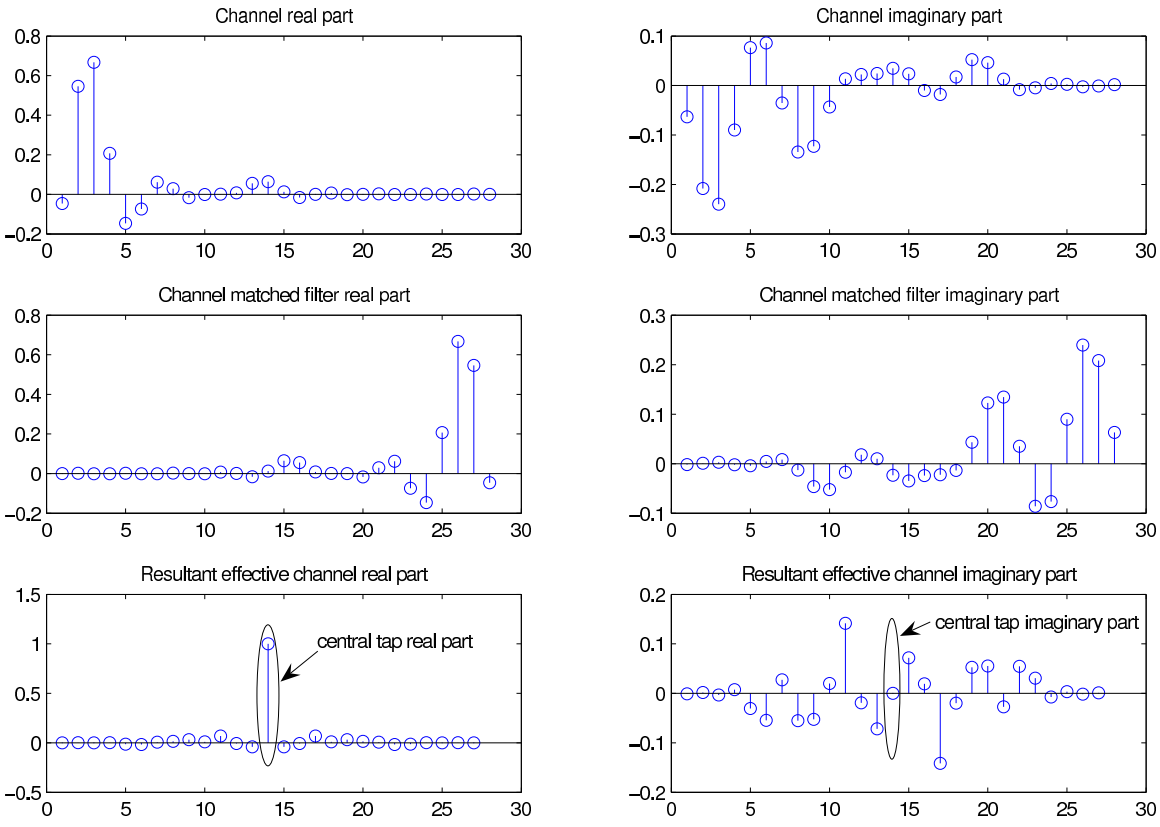


Figure E.13: Exemple OF démonstratif à la sortie du filtre de canal assorti

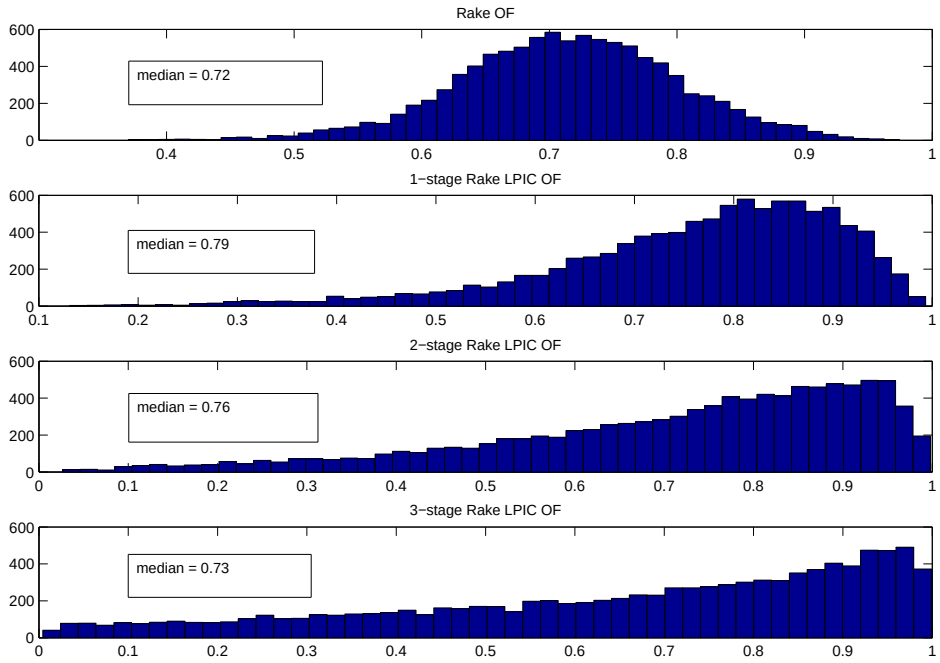


Figure E.14: Orthogonality factor histogram of conventional LPIC with 2-phase CMF in the Vehicular A channel

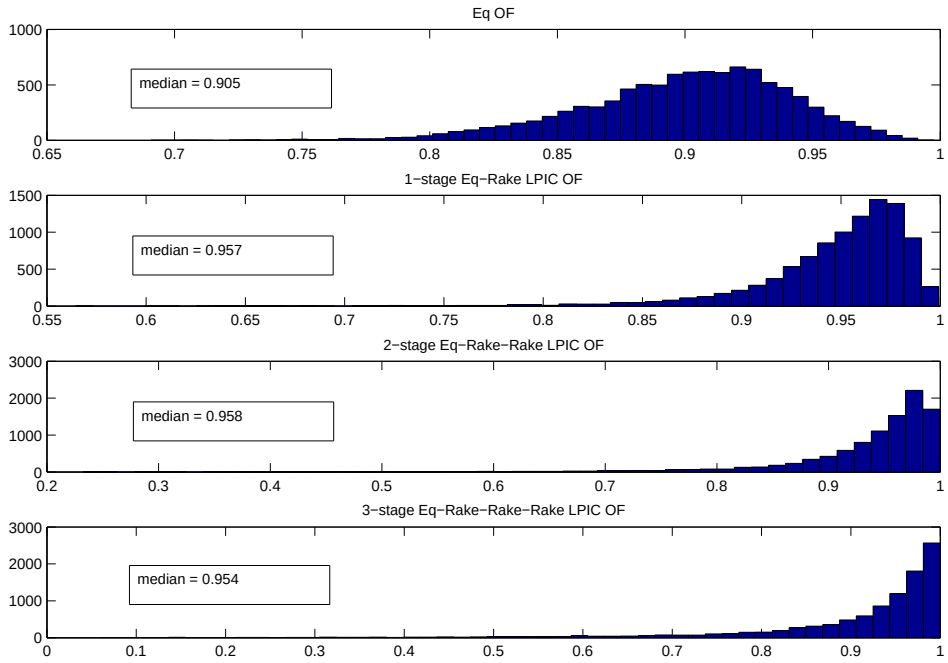


Figure E.15: Orthogonality factor histogram of LPIC with first stage 2-phase LMMSE chip equalizer followed by 2-phase CMFs in the Vehicular A channel with $\hat{I}_{or}/I_{oc} = 10dB$

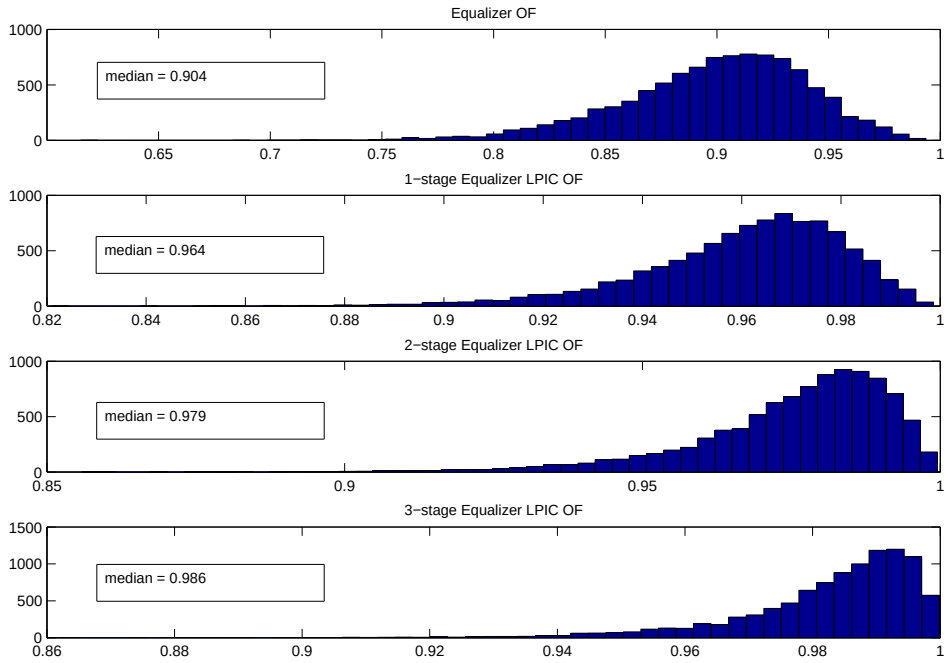


Figure E.16: Orthogonality factor histogram of LPIC with 2-phase LMMSE chip equalizers in all stages in the Vehicular A channel with $\hat{I}_{or}/I_{oc} = 10dB$

References

[1] *A Brief History of Communications.* IEEE History Center at Rutgers University, IEEE Communications Society, 2002.

[2] B. Widrow and M. Hoff, "Adaptive switching circuits," *IRE WESCON Conv. Rec.*, vol. 4, pp. 96–104, 1960.

[3] R. Lucky, "Techniques for adaptive equalization of digital communication systems," *Bell Sys. Tech, Journal*, vol. 45, pp. 255–286, February 1966.

[4] S. Verdú, "Minimum probability of error for asynchronous Gaussian multiple access channels," *IEEE Transactions on Information Theory*, vol. IT-32, pp. 85–96, January 1986.

[5] K. Berkel, F. Heinle, P. Meuwissen, K. Moerman, and M. Weiss, "Vector processing as an enabler for software-defined radio in handheld devices," *EURASIP Journal on Applied Signal Processing*, vol. 16, pp. 2613–2625, 2005.

[6] ——, "Vector processing as an enabler for software-defined radio in handsets from 3G+WLAN onwards," in *Software Defined Radio Technical Conference and Product Exposition*, Phoenix, Arizona, November 2004.

[7] "QoS concept and architecture," 3GPP, [Online]. Available: http://www.3gpp.org/ftp/Specs, Tech. Rep. TS 23.107.

[8] "User equipment (UE) radio transmission and reception (FDD)," 3GPP, [Online]. Available: http://www.3gpp.org/ftp/Specs, Tech. Rep. TS 25.101.

[9] "Physical layer procedures (FDD)," 3GPP, [Online]. Available: http://www.3gpp.org/ftp/Specs, Tech. Rep. TS 25.214.

[10] J. F. Kurose and K. W. Rose, *Computer Networking: A top down approach featuring the Internet.* Addison-Wesley, 2003.

[11] W. Stallings, *Data and Computer Communications, Seventh Edition*, 7th ed. Prentice Hall, May 2003.

[12] "Physical layer aspects of HSDPA," 3GPP, [Online].
Available: http://www.3gpp.org/ftp/Specs, Tech. Rep. TS 25.848.

[13] "UTRA high speed downlink packet access; overall UTRAN description," 3GPP, [Online]. Available: http://www.3gpp.org/ftp/Specs, Tech. Rep. TS25.855.

[14] H. Holma and A. Toskala, *WCDMA for UMTS.* John Wiley and Sons, 2000.

[15] P. Frenger, S. Parkvall, and E. Dahlman, "The evolution of WCDMA towards higher speed downlink packet data access," in *Proc. of the Vehicular Technology Conf.*, October 2001.

[16] M. Lenardi and D. T. M. Slock, "SINR maximizing equalizer receiver for DS-CDMA," in *Proc. of the EUSIPCO Conf.*, Tampere, Finland, September 2000.

[17] ——, "A RAKE structured SINR maximizing mobile receiver for the WCDMA downlink," in *Proc. of the Asilomar Conf. on Signals, Systems and Computers*, Pacific Grove, California, USA, November 2001.

[18] J. G. Proakis, *Digital Communications*, 3rd ed. NY: McGraw-Hill, 1995.

[19] T. S. Rappaport, *Wireless Communications - Principles and Practice.* Upper Saddle River, NJ: Prentice Hall, 1996.

[20] C. Cozzo, G. E. Bottomley, and A. S. Khayrallah, "Rake receiver finger placement for realistic channels," in *Proc. of the Wireless Communications and Networking Conference*, 2004, pp. 316–321.

[21] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication.* Cambridge University Press, May 2005.

[22] R. Lupas and S. Verdú, "Linear multiuser detectors for synchronous code-division multiple-access channels," *IEEE Transactions on Information Theory*, vol. IT-35, pp. 123–136, January 1989.

[23] S. Verdú, *Multiuser Detection.* Cambridge University Press, 1998.

[24] M. Varanasi and B. Aazhang, "Multistage detection in asynchronous code-division multiple-access communications," *IEEE Transactions on Communications Theory*, vol. 38, pp. 509–519, Apr 1990.

[25] A. Duel-Hallen, "Decorrelating decision-feedback multiuser detector for synchronous code-division multiple-access channel," *IEEE Transactions on Communications*, vol. 41, no. 2, pp. 285–290, February 1993.

[26] G. Strang, *Introduction to Linear Algebra*, 3rd ed. Wellesley-Cambridge Press, June 1998.

[27] H. P. Decell, "An application of the Cayley-Hamilton theorem to generalized matrix inversion," *Jr. SIAM Review*, vol. 7, No.4, pp. 526–528, October 1965.

[28] S. Moshavi, E. Kanterakis, and D. L. Schilling, "Multistage linear receivers for DS-CDMA systems," *International Journal of Wireless Information Networks*, vol. 3,No.1, 1996.

[29] R. R. Muller and S. Verdú, "Design and analysis of low complexity interference mitigation on vector channels," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 8, pp. 1429–1441, August 2001.

[30] K. Pedersen and P. Mogensen, "The downlink orthogonality factors influence on WCDMA system performance," in *Proc. of the Vehicular Technology Conf.*, September 2002.

[31] N. Mehta, L. Greenstein, T. Willis, and Z. Kostic, "Analysis and results for the orthogonality factor in WCDMA downlinks," in *Proc. of the Vehicular Technology Conf.*, May 2002.

[32] K. Zayana and B. Guisnet, "Measurements and modelisation of shadowing cross-correlations between two base stations," in *Proc. of the ICUPC*, Rome, Italy, October 1998.

[33] L. Greenstein, Y. Yeh, V. Erceg, and M. V. Clark, "A new path gain/delay-spread propagation model for digital cellular channels," *IEEE Transactions on Vehicular Technology*, vol. 46, no.2, May 1997.

[34] J. Sadowsky, D. Yellin, S. Moshavi, and Y. Perets, "Cancellation accuracy in CDMA pilot interference cancellation," in *Proc. of the Vehicular Technology Conf.*, April 2003.

[35] C. F. Report, "Digital mobile radio towards future generation systems," COST Telecom Secretariat, European Commission, Brussels, Belgium, Tech. Rep., 1999.

[36] J. Zhang, E. Chong, and D. Tse, "Output MAI distributions of linear MMSE multiuser receivers in DS-CDMA systems," *IEEE Transactions on Information Theory*, vol. 47, no.3, March 2001.

[37] H. Poor and S. Verdú, "Probability of error in MMSE multiuser detection," *IEEE Transactions on Information Theory*, vol. 43, no.3, May 1997.

[38] J. Sadowsky, D. Yellin, S. Moshavi, and Y. Perets, "Capacity gains from pilot cancellation in CDMA networks," in *Proc. of the Wireless Communications and Networking Conf.*, March 2003.

[39] "Technical specifications," 3GPP, [Online]. Available: http://www.3gpp.org, Tech. Rep.

[40] J. Bsltersee, G. Fock, P. Schultz-Rittich, and H. Meyr, "Performance analysis of phasor estimation algorithms for FDD-UMTS RAKE receiver," in *IEEE 6th Symp. on Spread Spectrum Technologies and Applications*, September 2000.

[41] M. Lenardi and D. Slock, "Estimation of time-varying wireless channels and application to the UMTS WCDMA FDD downlink," in *Proc. of the European Wireless Comm. Tech. conf.*, Firenze, Italy, February 2002.

[42] M. Benthin and K. Kammayer, "Influence of channel estimation on the performance of a coherent DS-CDMA system," *IEEE Trans. on Vehic. Tech.*, vol. 46, no.2, pp. 262–268, May 1997.

[43] M. Usada, Y. Ishikawa, and S. Onoe, "Optimizing the number of dedicated pilot symbols for forward link in WCDMA systems," in *Proc. of the IEEE Vehic. Tech. Conf.*, Spring 2000.

[44] K. A. Qaraqe and S. Roe, "Channel estimation algorithms for third generation WCDMA communication systems," in *Proc. of the IEEE Vehic. Tech. Conf.*, Fall 2001.

[45] G. Montalbano and D. Slock, "Joint common-dedicated pilots based estimation of time-varying channels for W-CDMA receivers," in *Proc. of the IEEE Vehic. Tech. Conf.*, October 2003.

[46] B. Musicus, "Iterative algorithms for optimal signal reconstruction and parameter identification given noisy and incomplete data," Ph.D. dissertation, Mass. Inst. Tech., September 1982.

[47] R. Shumway and D. Stoffer, "An approach to time series smoothing and forecasting using the EM algorithm," *J. Time Series Anal.*, vol. 3, no.4, pp. 253–264, 1982.

[48] E. Weinstein, A. Oppenheim, M. Feder, and J. Buck, "Iterative and sequential algorithms for multisensor signal enhancement," *IEEE Trans. Signal Proc.*, vol. 42, no.4, pp. 846–859, April 1994.

[49] V. Digalakis, J. Rohlicek, and M. Ostendorf, "ML estimation of a stochastic linear system with the EM algorithm and its application to speech recognition," *IEEE Trans. Speech and Audio Proc.*, vol. 1, no.4, October 1993.

[50] W. Gao, S.Tsai, and J. Lehnert, "Diversity cobining for DS/SS systems with time-varying correlated fading branches," *IEEE Trans. Communications*, vol. 51, no.2, February 2003.

[51] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. B*, vol. 39, no.1, pp. 1–38, 1977.

[52] M. Lenardi, A. Medles, and D. T. M. Slock, "A SINR maximizing RAKE receiver for DS-CDMA downlinks," in *Proc. of the Asilomar Conf. on Signals, Systems and Computers*, Pacific Grove, California - USA, November 2000.

[53] L. Mailaender, "Low complexity implementation of CDMA downlink equalization," in *Proc. of the Int. Conf on 3G Mobile Communications Technologies*, March 2001, pp. 396–400.

[54] T. Krauss and M. Zoltowski, "Chip-level MMSE equalization at the edge of the cell," in *Proc. of the Wireless Communications and Networking Conf.*, September 2000, pp. 386 – 392.

[55] W. Chen and U. Mitra, "An improved blind adaptive MMSE receiver for fast fading DS-CDMA channels," in *Proc. of the Globecom Conf.*, San Antonio, November 2001, pp. 758–762.

[56] F. Petre, M. Moonen, M. Engels, B. Gyselinckx, and H.D.Man, "Pilot-aided adaptive chip equalizer receiver for interference suppression in DS-CDMA forward link," in *Proc. of the Vehicular Technology Conf.*, Sept 2000, pp. 303–308.

[57] M. Hadef, S. Weiss, and M. Rupp, "Adaptive blind multiuser DS-CDMA," *Electronics Letters*, vol. 41, pp. 1184–1186, October 2005.

[58] A. Margetts and P. Schniter, "Adaptive chip-rate equalization of downlink multirate wideband CDMA," *IEEE Trans. on Signal Processing*, vol. 53, no.6, June 2005.

[59] K. Hooli, M. Juntti, M. Heikkila, P. Komulainen, M. Latva-aho, and J. Lilleberg, "Chip-level channel equalization in WCDMA downlink," *Eurasip Journal on Applied Signal Processing*, no. no.8, pp. 757–770, August 2002.

[60] B. Hassibi, *On the Robustness of LMS Filters*, least-mean-square adaptive filters ed. John Wiley and Sons, 2003, ch. 4.

[61] O. Macchi, *Adaptive Processing : The Least Mean Squares Approach with Applications in Transmission.* John Wiley and Sons, 1995.

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

[63] M. Heikkila, P.Komulainen, and J. Lilleberg, "Interference suppression in CDMA downlink through adaptive channel equalization," in *Proc. of the Vehicular Technology Conf.*, September 1999.

[64] Y. Sato, "A method of self-recovering equalization for multilevel amplitude-modulations systems," *IEEE Trans. on Communications*, pp. 679–682, June 1975.

[65] J. R. Treichler and B. G. Agee, "A new approach to multipath correction of constant modulus signals," *IEEE Trans. Acoust. Speech and Signal Processing*, vol. ASSP-31, pp. 459–472, April 1983.

[66] L. Litwin, M. Zoltowski, T. Endres, and S. Hulyalkar, "Blended CMA: smooth, adaptive transfer from CMA to DD-LMS," in *Proc. of the Wireless Communications and Networking Conference*, New Orleans, LA-USA, September 1999.

[67] F. D. Castro, M. D. Castro, and D. Arantes, "Concurrent blind deconvolution for channel equalization," in *Proc. of the IEEE International Conf. on Comm.*, Helsinki, Finland, June 2001, pp. 366–371.

[68] S. Chen and E. Chng, "Concurrent constant modulus algorithm and soft decision directed scheme for fractionally-spaced blind equalization," in *Proc. of the IEEE International Conf. on Comm.*, Paris-France, June 2004, pp. 2342 – 2346.

[69] C. Papadias and D. Slock, "On the decision-directed equalization of constant modulus signals," in *Proc. of the Asilomar Conf. on Signals, Systems and Computers*, Pacific Grove, CA, November 1994, pp. 1423–1427.

[70] W. Jakes, *Microwave Mobile Communications*. New York, Wiley, 1974.

[71] V. Mathews and Z.Xie, "A stochastic gradient adaptive filter with gradient adaptive step size," *IEEE Trans. on Signal Processing*, vol. 41, no.6, June 1993.

[72] C. D. Frank and E. Visotsky, "Adaptive interference suppression for direct-sequence CDMA systems with long spreading codes," in *Proc. of the Allerton Conference on Communication, Control, and Computing*, Urbana-Champaign, IL, September 1998.

[73] U. Madhow and M. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA," *IEEE Trans. on Communications Theory*, vol. 42, pp. 3178–3188, December 1994.

[74] R. M. Buehrer, S. P. Nicoloso, and S. Gollamudi, "Linear versus nonlinear interference cancellation," *Journal of Communications and Networks*, vol. 1, June 1999.

[75] M. Madcour, S. Gupta, and Y. Wang, "Successive interference cancellation algorithms for downlink W-CDMA communications," *IEEE Transactions on Wireless Communications*, vol. 1, no. 1, January 2002.

[76] D. Divsalar, M. Simon, and D. Raphelli, "Improved parallel interference cancellation," *IEEE Transactions on Communications Theory*, vol. 46, pp. 258–268, February 1998.

[77] R. Irmer, A. Nahler, and G. Fettweis, "On the impact of soft decision functions on the performance of multistage parallel interference cancelers for CDMA systems," in *Proc. Vehicular Tech. Conf.*, Rhodes, Greece, May 2001.

[78] I. Ghauri and D. T. M. Slock, "MMSE-ZF receiver and blind adaptation for multirate CDMA," in *Proc. Vehicular Technology Conf.*, September 1999.

[79] C. Papadias and D. Slock, "Fractionally spaced equalization of linear polyphase channels and related blind techniques based on multichannel linear prediction," *IEEE Transactions on Signal Processing*, vol. 47, no.3, March 1999.

[80] Y. P. Wang and G. E. Bottomley, "Generalized Rake reception for cancelling interference from multiple base stations," in *Proc. Vehicular Tech. Conf.*, Boston, Massachusset - USA, September 2000.

[81] A. Grant and C. Schlegel, "Convergence of linear interference cancellation multiuser receivers," *IEEE Transaction on Communications*, vol. 49, no. 10, October 2001.

[82] L. K. Rasmussen and I. J. Oppermann, "Ping-pong effects in linear parallel interference cancellation for CDMA," *IEEE Trans. on Wireless Comm.*, vol. 2, no. 2, pp. 357–363, March 2003.

[83] E. de Carvalho and D. T. M. Slock, "Crámer-Rao bounds for semi-blind, blind and training sequence based channel estimation," in *Proc. 1st IEEE Conf. on Signal Processing Advances in Wireless Communications*, Paris, France, April 1997, pp. 129–132.

[84] E. Aktas and U. Mitra, "Semiblind channel estimation for CDMA systems with parallel data and pilot signals," *IEEE Trans. on Communications*, vol. 52, no. 7, pp. 1102–1112, July 2004.

[85] S. Douglas, "Adaptive filters employing partial updates," *IEEE Trans. Circuits and Systems II*, vol. 44, pp. 209–216, March 1997.

[86] C. Papadias, "Methods for blind equalization and identification of linear channels," Ph.D. dissertation, l'Ecole Nationale Superieure des Telecommunications, March 1995.

[87] H. Xu, Z. Li, and H. Zheng, "A non-data-aided SNR estimation algorithm for QAM signals," in *Communications, Circuits and Systems Conf.*, June 2004, pp. 999 – 1003.

[88] T. Summers and S. Wilson, "SNR mismatch and online estimation in turbo decoding," *IEEE Trans. On Communications*, vol. 46,no.4, April 1998.

[89] P. Gao and C. Tepedelenlioglu, "SNR estimation for nonconstant modulus constellations," *IEEE Transactions on Signal Processing*, vol. 53, pp. 865 – 870, March 2005.

[90] T. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, vol. 13, pp. 47–60, November 1996.

[91] J. Chaufray, W. Hachem, and P. Loubaton, "Asymptotic analysis of optimum and sub-optimum CDMA downlink MMSE receivers," *IEEE Transactions on Information Theory*, vol. 50, no.11, pp. 2620–2638, November 2004.

[92] D. Marcuse, "Derivation of analytical expressions for the bit-error probability in lightwave systems with optical amplifiers," *Journal of Lightwave Technology*, vol. 8, no.12, December 1990.