**KATHOLIEKE UNIVERSITEIT LEUVEN**
FACULTEIT DER TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELEKTROTECHNIEK
Kardinaal Mercierlaan 94, 3001 Leuven (Heverlee)

# SIGNAL PROCESSING ALGORITHMS FOR CDMA-BASED WIRELESS COMMUNICATIONS

Promotor:
Prof. Dr. ir. M. Moonen

Proefschrift voorgedragen tot
het behalen van het doctoraat
in de toegepaste wetenschappen

door

**Geert LEUS**

Mei 2000

**KATHOLIEKE UNIVERSITEIT LEUVEN**
FACULTEIT DER TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELEKTROTECHNIEK
Kardinaal Mercierlaan 94, 3001 Leuven (Heverlee)

# SIGNAL PROCESSING ALGORITHMS FOR CDMA-BASED WIRELESS COMMUNICATIONS

Jury:
Prof. Dr. ir. E. Aernoudt, voorzitter
Prof. Dr. ir. M. Moonen, promotor
Prof. Dr. ir. B. De Moor
Prof. Dr. ir. S. Van Huffel
Prof. Dr. ir. L. Vandendorpe (U.C.L.)
Prof. Dr. ir. J. Vandewalle
Prof. Dr. ir. E. Van Lil
Prof. Dr. ir. A.-J. van der Veen (T.U. Delft)

Proefschrift voorgedragen tot
het behalen van het doctoraat
in de toegepaste wetenschappen

door

**Geert LEUS**

U.D.C. 681.3*I54 Mei 2000

D/2000/7515/25

ISBN 90-5682-255-1

# Voorwoord

Bij het begin van deze thesis zou ik graag een aantal mensen willen bedanken die rechtstreeks of onrechtstreeks hebben bijgedragen tot de goede afloop van mijn doctoraat. In de eerste plaats gaat mijn oprechte dank uit naar mijn promotor Prof. Marc Moonen, die mij de mogelijkheid geboden heeft om bij de onderzoeksgroep SISTA een doctoraat te beginnen. Zijn inzichten en constructieve kritiek hebben dit onderzoek in goede banen geleid. Ik dank Marc omdat hij mij in alle vrijheid onderzoek heeft laten uitvoeren, getuige hiervan de kans die ik heb gekregen om tijdens mijn doctoraat een studieverblijf aan Stanford University te verrichten. Marc was voor mij de ideale promotor.

Prof. Bart De Moor en Prof. Sabine Van Huffel zou ik willen bedanken voor de kostbare tijd die zij aan het nalezen van deze thesis hebben besteed en voor hun nuttige opmerkingen die hebben bijgedragen tot een betere begrijpbaarheid van dit werk.

I would also like to thank Prof. Luc Vandendorpe (U.C.L.) for his interest in my research and for carefully proofreading this thesis, despite his busy schedule.

Verder gaat mijn dank uit naar de andere leden van de jury, Prof. Joos Vandewalle, Prof. Emmanuel Van Lil en Prof. Alle-Jan van der Veen (T.U. Delft). Prof. Joos Vandewalle wil ik bedanken voor zijn grenzeloze inzet om van SISTA een welvarende onderzoeksgroep te maken. Prof. Emmanuel Van Lil dank ik voor zijn onmiddellijke bereidheid om in de jury te willen zetelen, niettegenstaande onze waarschijnlijk verschillende interesses. Verder wil ik Prof. Alle-Jan van der Veen bedanken voor zijn medewerking en zijn eigen hoogstaand onderzoekswerk dat mij gedurende de afgelopen jaren sterk geïnspireerd heeft.

Verder bedank ik ook Prof. Etienne Aernoudt voor het aanvaarden van het voorzitterschap van de jury.

Deze thesis is mede tot stand gekomen dankzij de stimulerende werksfeer die er binnen SISTA aanwezig is. Vandaar dat ik graag alle leden van SISTA zou willen bedanken. Ik denk hierbij in het bijzonder aan mijn oude en nieuwe bureaugenoten, Katleen Van Acker, Katrien De Cock, Johan Cambré, Jan Schier,

Koen Eneman en Olivier Rousseaux. Katleen Van Acker en Piet Vandaele zou ik ook willen bedanken voor de verschillende nuttige discussies die ik met hen gevoerd heb. Verder wil ik graag Geert Rombouts en Bart De Schutter bedanken voor hun vakkundige hulp wanneer ik weer eens geconfronteerd werd met één of ander computerprobleem. Ook onze secretaresse, Ida Tassens, wil ik bedanken voor haar administratieve bijstand en natuurlijk ook dank aan Bart Motmans voor zijn organisatorische werk binnen SISTA.

I would also like to thank Prof. Arogyaswami Paulraj for giving me the opportunity to visit Stanford University for a couple of months. During this study leave I had some very fruitful discussions with David Gesbert resulting in some interesting research results. Many thanks David. Ik denk ook met plezier terug aan de nuttige gesprekken die ik met Patrick Vandenameele (IMEC) gevoerd heb gedurende dit studieverblijf.

Ook het samenwerkingsverband tussen SISTA en Alcatel hebben mij in de loop van mijn doctoraat geïnspireerd. Vandaar mijn dank aan de volgende Alcatel werknemers: Thierry Pollet, Olivier van de Wiel (nu bij ETSI), Marc Van Bladel en Stan Claes.

Tenslotte wil in nog mijn familie en vrienden bedanken voor alles wat ze voor mij tijdens dit doctoraatswerk betekend hebben. In het bijzonder dank ik mijn ouders voor de mogelijkheden die zij mij geboden hebben en voor hun onvoorwaardelijke steun en vertrouwen.

# Abstract

Wireless communication systems rely on a *multiple-access* technique, i.e., a mechanism to divide the common transmission medium among different users. *Code-division multiple-access* (CDMA) is a multiple-access technique that has received considerable attention in recent years. In a CDMA system, each user spreads his information-bearing signal into a wideband signal, using specific code information. All users then transmit their wideband signal within the same frequency and time channel. This thesis deals with the development of receivers for various CDMA systems. Digital signal processing plays a central role in this development.

In recent literature, so-called *multi-user receivers* have become very popular. These receivers take into account the full structure of the *multi-user interference* (MUI), i.e., the interference originating from the other users. However, they have a rather high computational complexity. In the first part of this thesis, we therefore design new alternatives to multi-user receivers, yielding a lower computational complexity and a comparable performance. These alternatives are based on the concept of block spreading and have the remarkable property to completely remove the MUI, without using any channel information.

In the second part of this thesis, we will look at CDMA systems with multiple receive antennas (*spatial oversampling*). A popular receiver for such a system first applies some kind of space-time processing to transform the received wideband signals into an estimate of the wideband signal transmitted by the desired user and then reverses the spreading operation corresponding to this user. This receiver is known as the space-time RAKE receiver. We show how the desired user's code information can be used to design space-time RAKE receivers in a *blind* fashion, i.e., without relying on training sequences. Block processing as well as adaptive processing methods are considered.

# Korte Inhoud

Draadloze communicatiesystemen steunen op een techniek om *meervoudige toegang* te bieden, d.w.z., een mechanisme om het gemeenschappelijke transmissiemedium te verdelen onder verschillende gebruikers. *Codegebaseerde meervoudige toegang* (CDMA of *code-division multiple-access*) is een techniek om meervoudige toegang te bieden die de recente jaren heel wat aandacht heeft gekregen. In een CDMA systeem zet iedere gebruiker zijn informatiesignaal om in een breedbandsignaal, gebruik makende van specifieke code-informatie. Al de gebruikers sturen dan hun breedbandsignaal door via hetzelfde frequentie- en tijdskanaal. Deze thesis behandelt de ontwikkeling van ontvangers voor diverse CDMA systemen. Digitale signaalverwerking speelt een centrale rol in deze ontwikkeling.

In de recente literatuur zijn de zogenaamde *meergebruiker-ontvangers* heel populair geworden. Deze ontvangers houden rekening met de volledige structuur van de *meergebruiker-interferentie* (MUI of *multi-user interference*), d.w.z., de interferentie afkomstig van andere gebruikers. Ze zijn echter nogal rekenintensief. Daarom ontwikkelen we in het eerste deel van deze thesis alternatieven voor meergebruiker-ontvangers, die minder rekenintensief zijn en een vergelijkbare performantie hebben. Deze alternatieven zijn gebaseerd op het concept van blokspreiding en hebben de opmerkelijke eigenschap de MUI volledig te onderdrukken, zonder gebruik te maken van kanaalinformatie.

In het tweede deel van deze thesis kijken we naar CDMA systemen met meerdere ontvangstantennes (*spatiale overbemonstering*). Een populaire ontvanger voor zo een systeem past eerst een soort van ruimte-tijd verwerking toe om de ontvangen breedbandsignalen om te zetten in een schatting van het breedbandsignaal uitgezonden door de gewenste gebruiker en keert dan de spreidingsoperatie uitgevoerd door deze gebruiker om. Deze ontvanger wordt een ruimte-tijd RAKE-ontvanger genoemd. We tonen aan hoe de code-informatie van de gewenste gebruiker kan gebruikt worden om ruimte-tijd RAKE-ontvangers te ontwerpen op een *blinde* manier, d.w.z., zonder te steunen op trainingssequenties. Zowel blokverwerkingsmethoden als adaptieve verwerkingsmethoden worden beschouwd.

# Glossary

## Mathematical Notation

| | |
|---|---|
| $\mathbf{x}$ | vector $\mathbf{x}$ |
| $\mathbf{X}$ | matrix $\mathbf{X}$ |
| $\mathbf{X}^T$ | transpose of matrix $\mathbf{X}$ |
| $\mathbf{X}^H$ | Hermitian transpose of matrix $\mathbf{X}$ |
| $\mathbf{X}^*$ | complex conjugate of matrix $\mathbf{X}$ |
| $\mathbf{X}^{-1}$ | inverse of matrix $\mathbf{X}$ |
| $\mathbf{X}^\dagger$ | pseudoinverse of matrix $\mathbf{X}$ |
| $\mathrm{tr}\{\mathbf{X}\}$ | trace of matrix $\mathbf{X}$ |
| $\|\mathbf{X}\|$ | Frobenius norm of matrix $\mathbf{X}$ |
| $\mathrm{fc}\{\mathbf{X}\}$ | matrix obtained by flipping the columns of matrix $\mathbf{X}$ |
| $\mathrm{fr}\{\mathbf{X}\}$ | matrix obtained by flipping the rows of matrix $\mathbf{X}$ |
| $\mathrm{diag}\{\mathbf{x}\}$ | square diagonal matrix with $\mathbf{x}$ as diagonal. |
| $\mathbf{X}(k,l)$ | element on the $k$th row and $l$th column of matrix $\mathbf{X}$ |
| $\mathbf{X}(k:l,:)$ | rows $k$ up to $l$ of matrix $\mathbf{X}$ |
| $\mathbf{X}(:,k:l)$ | columns $k$ up to $l$ of matrix $\mathbf{X}$ |
| $\mathbf{0}$ | zero vector |
| $\mathbf{O}$ | zero matrix |
| $\mathbf{1}_n$ | unit vector with a one in the $n$th entry |
| $\mathbf{I}$ | identity matrix |
| $\mathbf{O}_n$ | $n \times n$ zero matrix |
| $\mathbf{I}_n$ | $n \times n$ identity matrix |
| $\mathbf{J}_n$ | $n \times n$ cyclic shift matrix with ones in the first subdiagonal and in the upper right entry |
| $\mathbf{D}_n$ | $n \times n$ diagonal matrix with diagonal elements $-1, +1, -1, +1, \ldots$ |
| $\mathcal{F}_P$ | $P \times P$ DFT matrix |
| $\mathcal{I}_P$ | $P \times P$ IDFT matrix |
| $\star$ | don't care matrix |
| $\overset{\star}{\diagdown}$ | a square upper triangular don't care matrix |
| $\mathrm{proj}_{\Upsilon}\{x\}$ | mapping of $x$ to the closest (in Euclidean distance) element of $\Upsilon$ |
| $\Re\{x\}$ | real part of $x$ |

| | |
|---|---|
| $\Im\{x\}$ | imaginary part of $x$ |
| $\hat{x}$ | estimate of $x$ |
| $\underline{\hat{x}}$ | hard estimate of $x$ |
| $\lvert x \rvert$ | absolute value of $x$ |
| $\lfloor x \rfloor$ | largest integer smaller or equal to $x \in \mathbb{R}$ |
| $\lceil x \rceil$ | smallest integer larger or equal to $x \in \mathbb{R}$ |
| $\mathrm{E}\{x\}$ | expectation of random variable $x$ |
| $x \bmod y$ | remainder after dividing $x \in \mathbb{R}$ by $y \in \mathbb{R}$ |
| $\mathbb{R}$ | the set of real numbers |
| $\mathbb{C}$ | the set of complex numbers |

## Acronyms and Abbreviations

| | |
|---|---|
| ACMA | Analytical Constant Modulus Algorithm |
| AMOUR | A Mutually Orthogonal Usercode Receiver |
| AWGN | Additive White Gaussian Noise |
| BER | Bit Error Rate |
| BM | Block Method |
| BP | Block Processing |
| BPSK | Binary Phase Shift Keying |
| CDMA | Code-Division Multiple-Access |
| DF | Decision Feedback |
| DFT | Discrete Fourier Transformation |
| DMT | Discrete Multi-Tone |
| DMT-CDMA | Discrete Multi-Tone CDMA |
| DMT-CDMA-BS | Discrete Multi-Tone CDMA based on Block Spreading |
| DMT-DS-CDMA | Discrete Multi-Tone Direct-Sequence CDMA |
| DS-CDMA | Direct-Sequence CDMA |
| DS-CDMA-BS | Direct-Sequence CDMA based on Block Spreading |
| EWAM | Exponentially Weighted Adaptive Method |
| FDD | Frequency-Division Duplexing |
| FDMA | Frequency-Division Multiple-Access |
| FEQ | Frequency domain EQualizer |
| FFT | Fast Fourier Transformation |
| FIR | Finite Impulse Response |
| GSM | Global System for Mobile communications |
| ICI | Inter-Chip Interference |
| ICBI | Inter-Chip-Block Interference |
| IDFT | Inverse Discrete Fourier Transformation |
| IFFT | Inverse Fast Fourier Transformation |
| ILSP | Iterative Least-Squares with Projection |
| IS-95 | Interim Standard 95 |
| ISI | Inter-Symbol Interference |

| | |
|---|---|
| ITI | Inter-Tone Interference |
| LS | Least Squares |
| LV-AMOUR | Lagrange-Vandermonde AMOUR |
| LV-CDMA | Lagrange-Vandermonde CDMA |
| MF | Matched Filter |
| MIMO | Multi-Input Multi-Output |
| MMSE | Minimum Mean-Square Error |
| MMSE-DF | Minimum Mean-Square Error Decision Feedback |
| MOE | Minimum Output Energy |
| MRE | Mutually Referenced Equalizer |
| MSE | Mean Square-Error |
| MT | Multi-Tone |
| MUI | Multi-User Interference |
| MVDR | Minimum Variance Distortionless Response |
| NMSE | Normalized Mean Square-Error |
| OFDM | Orthogonal Frequency-Division Multiplexing |
| PSK | Phase Shift Keying |
| QPSK | Quadrature Phase Shift Keying |
| QRD | QR Decomposition |
| RACMA | Real Analytical Constant Modulus Algorithm |
| RC | Real Constellation |
| SDMA | Spatial-Division Multiple-Access |
| SIMO | Single-Input Multi-Output |
| SINR | Signal-to-Interference-plus-Noise Ratio |
| SNR | Signal-to-Noise Ratio |
| SOS | Second-Order Statistics |
| SSI | SubSpace Intersection |
| s.t. | subject to |
| ST | Space-Time |
| ST-RAKE | Space-Time RAKE |
| SVD | Singular Value Decomposition |
| TD | Transmit Diversity |
| TDD | Time-Division Duplexing |
| TDMA | Time-Division Multiple-Access |
| TEQ | Time domain EQualizer |
| TLS | Total Least Squares |
| VL-AMOUR | Vandermonde-Lagrange AMOUR |
| VL-CDMA | Vandermonde-Lagrange CDMA |
| WAP | Windowed Adaptive Processing |
| WAM | Windowed Adaptive Method |
| W-CDMA | Wideband CDMA |
| w.l.o.g. | without loss of generality |
| w.r.t. | with respect to |
| ZF | Zero-Forcing |
| ZF-DF | Zero-Forcing Decision Feedback |

# Contents

# Signaalverwerkingsalgoritmen voor CDMA-Gebaseerde Draadloze Communicatie

## Samenvatting

## Hoofdstuk 1. Inleiding

De laatste jaren is de markt voor draadloze communicatiesystemen enorm gegroeid. De oorzaak hiervan ligt voornamelijk bij de volgende drie aantrekkelijke kenmerken van draadloze systemen:

- Voor de meeste toepassingen is een draadloos systeem goedkoper dan een bedraad systeem.
- Draadloze systemen laten communicatie toe op plaatsen waar het moeilijk is om draden te leggen.
- Draadloze systemen laten mobiele gebruikers toe.

Een cruciaal onderdeel van gelijk welk draadloos communicatiesysteem is de techniek om *meervoudige toegang* (*multiple access*) te bieden, d.w.z., het mechanisme om het gemeenschappelijke transmissiemedium te verdelen onder verschillende gebruikers. De basistechnieken om *meervoudige toegang* te bieden zijn *frequentiegebaseerde meervoudige toegang* (FDMA of *frequency-division multiple-access*), *tijdsgebaseerde meervoudige toegang* (TDMA of *time-division multiple-access*) en *codegebaseerde meervoudige toegang* (CDMA of *code division multiple-access*). Het onderwerp van deze thesis is de ontwikkeling van

ontvangers voor diverse CDMA systemen. Digitale signaalverwerking speelt een centrale rol in de ontwikkeling van deze ontvangers.

Draadloze systemen zijn meestal cellulair. De operator deelt het grondgebied op in verschillende cellen en plaatst een basisstation in het middelpunt van iedere cel. Bijgevolg zijn er twee richtingen van communicatie mogelijk: de communicatie van het basisstation naar de gebruikers (*forward link* of *downlink*) en de communicatie van de gebruikers naar het basisstation (*reverse link* of *uplink*). In deze thesis zullen we enkel aandacht besteden aan de communicatie van de gebruikers naar het basisstation.

Aan iedere cel van een cellulair communicatiesysteem wordt een zekere frequentieband toegekend, die kan hergebruikt worden in andere cellen. De verhouding van het totaal aantal cellen over het aantal cellen die dezelfde frequentieband gebruiken wordt *frequentiehergebruikfactor* genoemd. Merk op dat de bandbreedte die aan een cel wordt toegewezen gelijk is aan de bandbreedte van het beschikbare frequentiespectrum gedeeld door de frequentiehergebruikfactor. Met dit in ons achterhoofd bespreken we nu de de drie basistechnieken om meervoudige toegang te bieden (zie figuur 1.1).

In een FDMA of TDMA systeem communiceert iedere gebruiker binnen dezelfde cel via een uniek frequentie- of tijdskanaal. Bijgevolg moet er geen rekening gehouden worden met intracel-interferentie. Omdat er meestal gebruik gemaakt wordt van een frequentiehergebruikfactor van zeven, is de afstand tussen twee cellen die dezelfde frequentieband toegewezen krijgen groot (zie figuur 1.2). De intercel-interferentie is daarom verwaarloosbaar. Het huidige GSM systeem is gebaseerd op een combinatie van FDMA en TDMA.

In een CDMA systeem, daarentegen, communiceert iedere gebruiker binnen dezelfde cel via eenzelfde frequentie- en tijdskanaal. Bijgevolg moet er rekening gehouden worden met intracel-interferentie. Omdat er meestal gebruik gemaakt wordt van een frequentiehergebruikfactor van één (deze eigenschap wordt de *universele frequentiehergebruikeigenschap* genoemd), is de afstand tussen twee cellen die dezelfde frequentieband toegewezen krijgen klein (zie figuur 1.2). De intercel-interferentie is daarom niet verwaarloosbaar. De vraag is dan hoe de meergebruiker-interferentie (MUI of *multi-user interference*), gedefinieerd als de som van de intracel- en intercel-interferentie, kan onderdrukt worden. Het antwoord op deze vraag volgt uit het feit dat, in een CDMA systeem, iedere gebruiker zijn informatiesignaal omzet in een breedbandsignaal, gebruik makende van specifieke code-informatie. De ontvanger kan dan bijvoorbeeld de spreidingsoperatie uitgevoerd door de gewenste gebruiker omkeren, in de veronderstelling dat de code-informatie van deze gebruiker gekend is. Als al de code-informatie willekeurig gekozen is, zal deze operatie het ontvangen breedbandsignaal omzetten in een smalbandsignaal (met dezelfde bandbreedte als de informatiesignalen), dat veel meer vermogen bevat afkomstig van de gewenste gebruiker dan van de andere gebruikers. Dit laat ons toe om het gewenste

informatiesignaal te reconstrueren (zie figuur 1.3). Om de residuele MUI na deze operatie zo klein mogelijk te houden, moet iedere gebruiker het laagst mogelijke vermogen uitsturen dat nodig is voor een betrouwbare transmissie. Dit vereist een strikte vermogenscontrole. Als we geen strikte vermogenscontrole toepassen, zullen gebruikers die zich dicht bij het basisstation bevinden, en dus ontvangen worden met een hoog vermogen, een nefaste invloed hebben op de detectie van gebruikers die zich ver van het basisstation bevinden, en dus ontvangen worden met een laag vermogen. Dit is het zogenaamde *near-far*-probleem. We kunnen echter ook meer geavanceerde ontvangers gebruiken, die geen strikte vermogenscontrole vereisen (i.p.v. de ontvanger die simpelweg de spreidingsoperatie uitgevoerd door de gewenste gebruiker omkeert). Merk op dat sommige van deze ontvangers gebaseerd zijn op al de code-informatie, en niet alleen op de code-informatie van de gewenste gebruiker. Welke ontvangsttechniek we ook gebruiken, CDMA heeft een aantal interessante voordelen t.o.v. FDMA of TDMA:

- In het geval van spraaktransmissie is een gebruiker minder dan de helft van de tijd actief. In een FDMA of TDMA systeem, waar iedere gebruiker een uniek frequentie- of tijdskanaal toegewezen krijgt, veroorzaakt dit een groot capaciteitsverlies, dat enkel kan opgelost worden door een andere gebruiker toegang te verlenen tot het kanaal van de stille gebruiker. Dit is echter enkel mogelijk als de stille periode lang genoeg duurt. In een CDMA systeem, waar iedere gebruiker hetzelfde frequentie- en tijdskanaal krijgt toegewezen, komt dit probleem niet voor.

- In een FDMA of TDMA systeem bezetten aangrenzende cellen verschillende frequentiebanden (frequentiehergebruikfactor van zeven). Dus even nadat een gebruiker de grens tussen twee cellen heeft overgestoken, verbreekt hij de verbinding met het basisstation van de eerste cel en begint hij te communiceren met het basisstation van de tweede cel. Dit wordt *hard handoff* genoemd. De universele frequentiehergebruikeigenschap van een CDMA systeem, daarentegen, laat het gebruik van *soft handoff* toe. Even voordat een gebruiker de grens tussen twee cellen zal oversteken, begint hij te communiceren met het basisstation van de tweede cel, zonder de verbinding met het basisstation van de eerste cel te verbreken. Dit zorgt voor een soort van ontvangstdiversiteit die kan gebruikt worden om de performantie van het systeem te verbeteren.

- Ten gevolge van verstrooiing door de omgeving wordt een uitgezonden signaal ontvangen via verschillende paden (zie figuur 1.4). De ontvangen signalen die horen bij de verschillende paden zijn allemaal kopies van hetzelfde uitgezonden signaal met verschillende amplituden, fasen en vertragingen. Vermits in een CDMA systeem iedere gebruiker een grotere bandbreedte toegewezen krijgt dan in een FDMA of TDMA systeem, worden de paden beter geïsoleerd. Bijgevolg kan in een CDMA systeem

de meerpad-interferentie (*multi-path interference*) efficiënter onderdrukt worden dan in een FDMA of TDMA systeem.

In een CDMA systeem gaat iedere gebruiker zijn datasymboolsequentie met snelheid $1/T$ (de datasymboolsnelheid) omzetten in een chipsequentie met snelheid $N/T$ (de chipsnelheid), gebruik makende van specifieke code-informatie. Merk op dat $N \geq 1$ de spreidingsfactor wordt genoemd. De manier waarop deze spreidingsoperatie uitgevoerd wordt hangt af van het soort CDMA. Al de gebruikers sturen dan hun chipsequentie door via hetzelfde frequentie- en tijdskanaal. Tenslotte probeert de ontvanger de datasymboolsequentie van de gewenste gebruiker te reconstrueren, gebaseerd op ofwel de kennis van de code-informatie van de gewenste gebruiker ofwel de kennis van al de code-informatie.

In dit hoofdstuk worden diverse CDMA systemen geïntroduceerd. Eerst besteden we aandacht aan het meest populaire type van CDMA, namelijk *direct-sequence CDMA* (DS-CDMA). DS-CDMA wordt momenteel gebruikt in het IS-95 systeem [Vit95], maar het is ook de techniek die in de meeste ontwerpen van derdegeneratiesystemen aangewend wordt om meervoudige toegang te bieden [OP98]. Dergelijke ontwerpen van derdegeneratiesystemen worden *wideband CDMA* (W-CDMA) ontwerpen genoemd. In dit hoofdstuk wordt een uitgebreid overzicht gegeven van bestaande ontvangers voor een DS-CDMA systeem. Eerst bespreken we de RAKE-ontvanger, ontwikkeld voor een enkele ontvangstantenne. Daarna bestuderen we de ruimte-tijd RAKE-ontvanger, een uitbreiding van de RAKE-ontvanger voor meerdere ontvangstantennes. Tenslotte besteden we aandacht aan zogenaamde meergebruiker-ontvangers. Er wordt in dit hoofdstuk ook een sectie gewijd aan zogenaamde blinde ontvangers, die geen trainingssequenties nodig hebben om een gebruiker te detecteren. Na de bespreking van het populaire DS-CDMA systeem, besteden we aandacht aan het DS-CDMA systeem gebaseerd op blokspreiding (DS-CDMA-BS systeem), geïntroduceerd door Cirpan en Tsatsanis [CT97]. De spreidingsoperatie die in een DS-CDMA-BS systeem gebruikt wordt kan geïnterpreteerd worden als de spreidingsoperatie die in een DS-CDMA systeem gebruikt wordt, gevolgd door *chipinterleaving* [CT97]. Merk op dat de auteurs van [CT97] het DS-CDMA-BS systeem enkel hebben geïntroduceerd om blinde meergebruiker-kanaalschatting te vergemakkelijken. Zij hebben geen speciale ontvangerstructuren voorgesteld voor zo een systeem. In dit hoofdstuk tonen wij aan dat een DS-CDMA-BS systeem een opmerkelijke eigenschap bezit die aanleiding zal geven tot de ontwikkeling van nieuwe ontvangers voor een DS-CDMA-BS systeem. Deze eigenschap is voor zover wij weten nog nooit eerder opgemerkt.

Vervolgens bespreken we in dit hoofdstuk een aantal CDMA systemen gebaseerd op DMT-modulatie (DMT: *discrete multi-tone*). MT-modulatie (MT: *multi-tone*) is een aantrekkelijke modulatietechniek die momenteel heel wat aandacht krijgt [Bin90]. Hoewel men zou verwachten dat zo een techniek het beschikbare frequentiespectrum zou opdelen in verschillende frequentiebanden die

mekaar niet overlappen, deelt MT-modulatie dit spectrum op in overlappende frequentiebanden met bepaalde orthogonaliteitseigenschappen. De middelpunten van deze banden noemen we de tonen. DMT-modulatie is de discrete implementatie van MT-modulatie en maakt meestal gebruik van een zogenaamde cyclische prefix (*cyclic prefix*). DMT-modulatie heeft twee interessante eigenschappen. Eerst en vooral kan iedere toon optimaal (wat betreft performantie of capaciteit) gemoduleerd worden, als we de signaalruisverhouding (SNR) op die toon kennen. Ten tweede laat DMT-modulatie een eenvoudige kanaalegalisatie toe.

DMT-gebaseerde CDMA systemen combineren de voordelen van DMT-modulatie met CDMA. Er zijn echter verschillende mogelijkheden om DMT-modulatie met CDMA te combineren. Een eerste mogelijkheid bestaat erin om op de DMT-gemoduleerde datasymboolsequentie dezelfde spreidingsoperatie uit te voeren als in een DS-CDMA systeem. We bekomen dan het DMT-CDMA systeem, geïntroduceerd door Vandendorpe [Van95]. Een tweede mogelijkheid bestaat erin om op de DMT-gemoduleerde datasymboolsequentie dezelfde spreidingsoperatie uit te voeren als in een DS-CDMA-BS systeem. We bekomen dan het nieuwe DMT-CDMA-BS systeem. Merk op dat we meestal geen gebruik maken van een cyclische prefix in een DMT-CDMA of DMT-CDMA-BS systeem. Een eenvoudige manier om een ontvanger voor een DMT-CDMA of DMT-CDMA-BS systeem te ontwerpen is een ontvanger voor een DS-CDMA of DS-CDMA-BS systeem te nemen en daarachter een DMT-demodulator te plaatsen. We geven zo een ontvanger in deze thesis dezelfde naam als de overeenkomstige ontvanger voor een DS-CDMA of DS-CDMA-BS systeem.

Een laatste mogelijkheid, die we in dit proefschrift zullen beschouwen, bestaat erin op de verschillende tonen van een DMT-systeem dezelfde spreidingsoperatie uit te voeren als in een DS-CDMA systeem. We bekomen dan het zogenaamde DMT-DS-CDMA systeem, geïntroduceerd door DaSilva en Sousa [DS94]. Merk op dat we in een DMT-DS-CDMA systeem meestal wel gebruik maken van een cylcische prefix.

Tenslotte wordt in dit hoofdstuk een algemeen overzicht gegeven van deze thesis. De structuur van de thesis is weergegeven in figuur 1.15. De thesis bestaat uit twee delen. In deel I ontwikkelen we alternatieven voor meergebruikerontvangers, die minder rekenintensief zijn en een vergelijkbare performantie hebben. Deze alternatieven, die gebaseerd zijn op het concept van blokspreiding, worden ontworpen voor een enkele ontvangstantenne en een ICI-beperkt quasi-synchroon propagatiemodel. In deel II bestuderen we deterministische blinde ruimte-tijd RAKE-ontvangers voor een DS-CDMA systeem met meerdere ontvangstantennes. We besteden daarbij eerst aandacht aan blokverwerking en daarna aan adaptieve verwerking. Tenslotte tonen we aan hoe de ontwikkelde deterministische blinde ruimte-tijd RAKE-ontvangers ook kunnen toegepast worden voor een DMT-DS-CDMA systeem.

## Hoofdstuk 2. Basisconcepten

Dit hoofdstuk introduceert een aantal basisconcepten die in deze thesis zullen gebruikt worden. We beginnen dit hoofdstuk met een wiskundige beschrijving van de spreidingsoperatie voor de verschillende CDMA systemen die besproken werden in het vorige hoofdstuk. Eerst concentreren we ons op het DS-CDMA systeem en het DS-CDMA-BS systeem. Daarna beschouwen we een aantal gerelateerde DMT-gebaseerde CDMA systemen, zoals het DMT-CDMA systeem, het DMT-CDMA-BS systeem en het DMT-DS-CDMA systeem.

Vervolgens bespreken we een aantal datamodellen die de ontvangen informatie beschrijven. Voor de eenvoud veronderstellen we hierbij dat er aan de ontvangstzijde bemonsterd wordt aan de chipsnelheid. De modellen die men zo bekomt kunnen eenvoudig uitgebreid worden naar datamodellen voor een bemonstering aan een veelvoud van de chipsnelheid. Eerst worden een aantal datamodellen voorgesteld gebaseerd op een enkele ontvangstantenne. Hierbij wordt speciale aandacht besteed aan datamodellen voor een ICI-beperkt quasi-synchroon propagatiemodel. Daarna worden datamodellen voorgesteld gebaseerd op meerdere ontvangstantennes.

## Deel I. Ontvangers voor Blokspreidingssystemen: Alternatieven voor Meergebruiker-Ontvangers

In een DS-CDMA en DMT-CDMA systeem worden hoge performanties bereikt met zogenaamde meergebruiker-ontvangers. Deze ontvangers zijn echter nogal rekenintensief. Daarom zoeken we in deel I naar alternatieven voor meergebruiker-ontvangers, waarbij we mikken op een lagere complexiteit en een vergelijkbare performantie. Voor een enkele ontvangstantenne en een ICI-beperkt quasi-synchroon propagatiemodel vormen nieuwe ontvangers die we ontwikkelen voor een DS-CDMA-BS en DMT-CDMA-BS systeem zulke alternatieven.

## Hoofdstuk 3. Ontvangers voor een DS-CDMA Systeem

In dit hoofdstuk geven we een overzicht van een aantal bestaande ontvangers voor een DS-CDMA systeem. Alhoewel deze ontvangers kunnen ontwikkeld worden voor meer algemene scenario's, veronderstellen we hier een enkele ontvangstantenne en een ICI-beperkt quasi-synchroon propagatiemodel. Verder maken we gebruik van korte codesequenties.

Eerst wordt de RAKE-ontvanger beschreven. Deze bestaat uit een bank van correlators, gevolgd door een lineaire combinator, die de uitgangen van de correlators lineair combineert. Daarna beschrijven we de lineaire meergebruiker-egalisator. In de context van de lineaire meergebruiker-egalisator stellen we tenslotte een blind meergebruiker-kanaalschattingsalgoritme voor.

## Hoofdstuk 4. Ontvangers voor een DS-CDMA-BS Systeem

In dit hoofdstuk ontwikkelen we een aantal nieuwe ontvangers voor een DS-CDMA-BS systeem. We veronderstellen weer een enkele ontvangstantenne en een ICI-beperkt quasi-synchroon propagatiemodel, en maken gebruik van korte codesequenties.

Eerst wordt de blok-RAKE-ontvanger geïntroduceerd. Deze bestaat uit een bank van blokcorrelators, gevolgd door een lineaire blokcombinator, die de uitgangen van de blokcorrelators lineair combineert. Daarna introduceren we de MUI-vrije ontvanger, die de MUI volledig onderdrukt, zonder gebruik te maken van kanaalinformatie. De MUI-vrije ontvanger is eigenlijk een aangepaste blok-RAKE-ontvanger, die bestaat uit een bank van aangepaste blokcorrelators, gevolgd door een lineaire blokcombinator, die de uitgangen van de aangepaste blokcorrelators lineair combineert. De MUI-vrije werking wordt bekomen door gebruik te maken van een zogenaamde shift-orthogonale set van codesequenties, waarop deze ontvanger is gebaseerd. Om de MUI-vrije ontvanger bruikbaar te maken in de praktijk, ontwikkelen we een ontwerpstrategie voor een shift-orthogonale set van codesequenties. Deze ontwerpstrategie kan beschouwd worden als één van de belangrijkste bijdragen van dit hoofdstuk. In de context van de MUI-vrije ontvanger stellen we tenslotte een blind ééngebruiker-kanaalschattingsalgoritme voor.

## Hoofdstuk 5. Complexiteit en Simulatieresultaten

Dit hoofdstuk vergelijkt de nieuwe MUI-vrije ontvanger, ontwikkeld voor een DS-CDMA-BS systeem, met de bestaande lineaire meergebruiker-egalisator, ontwikkeld voor een DS-CDMA systeem. Eerst vergelijken we de complexiteit van de MUI-vrije ontvanger met de complexiteit van de lineaire meergebruiker-egalisator. We veronderstellen daarbij dat het ontwerp van de ontvanger gebaseerd is op het overeenkomstig blind kanaalschattingsalgoritme, niettegenstaande dezelfde conclusies kunnen getrokken worden uit andere ontwerptech-

nieken, zoals bijvoorbeeld trainingsgebaseerde ontwerptechnieken. We tonen aan dat de complexiteit van de MUI-vrije ontvanger veel kleiner is dan de complexiteit van de lineaire meergebruiker-egalisator. Bovendien is ook de complexiteit van een volledige bank van MUI-vrije ontvangers (één voor elke gebruiker) veel kleiner dan de complexiteit van een volledige bank van lineaire meergebruiker-egalisatoren (één voor elke gebruiker), hoewel deze laatste complexiteit van dezelfde grootte orde is als de complexiteit van één enkele lineaire meergebruiker-egalisator. Verder illustreren we dat de performantie van de MUI-vrije ontvanger vergelijkbaar is met die van de lineaire meergebruiker-egalisator. Dit zowel voor het geval de kanalen gekend zijn als voor het geval ze geschat worden met behulp van het overeenkomstig blind kanaalschattingsalgoritme.

Vervolgens vergelijken we in dit hoofdstuk de MUI-vrije ontvanger met de overeenkomstige FFT-gebaseerde VL-AMOUR zender-ontvanger [WSGB99]. We tonen aan dat de laatste een duurdere zenderstructuur heeft dan de spreidingsoperatie die in een DS-CDMA-BS systeem gebruikt wordt en een duurdere ontvangerstructuur heeft dan de MUI-vrije ontvanger. Voor het geval de kanalen gekend zijn tonen simulatieresultaten aan dat de performantie van de MUI-vrije ontvanger vergelijkbaar is met de performantie van de overeenkomstige FFT-gebaseerde VL-AMOUR zender-ontvanger.

## Hoofdstuk 6. Ontvangers voor een DMT-CDMA en DMT-CDMA-BS Systeem

We vestigen nu onze aandacht op DMT-gebaseerde CDMA systemen. In overeenstemming met hoofdstuk 3 geven we eerst een overzicht van een aantal bestaande ontvangers voor een DMT-CDMA systeem, zoals de RAKE-ontvanger en de lineaire meergebruiker-egalisator. Elk van deze ontvangers bestaat simpelweg uit de overeenkomstige ontvanger voor een DS-CDMA systeem gevolgd door een DMT-demodulator. In overeenstemming met hoofdstuk 4 ontwikkelen we daarna een aantal nieuwe ontvangers voor een DMT-CDMA-BS systeem, zoals de blok-RAKE-ontvanger en de MUI-vrije ontvanger, die de MUI volledig onderdrukt, zonder gebruik te maken van kanaalinformatie. Elk van deze ontvangers bestaat opnieuw simpelweg uit de overeenkomstige ontvanger voor een DS-CDMA-BS systeem gevolgd door een DMT-demodulator. De belangrijkste bijdrage van dit hoofdstuk is echter de ontwikkeling van de MUI/ITI-vrije ontvanger, die naast de MUI ook de intertoon-interferentie (ITI of *inter-tone interference*) volledig onderdrukt, zonder gebruik te maken van kanaalinformatie. Deze ontvanger wordt bekomen door de lineaire blokcombinator van de MUI-vrije ontvanger te vervangen door een eenvoudigere operatie die niet gebaseerd is op kanaalinformatie en er voor zorgt dat de ITI volledig onderdrukt wordt. Aan de uitgang van de DMT-demodulator moet iedere toon dan wel

nog aangepast worden met behulp van een frequentiedomeinegalisator (FEQ of *frequency domain equalizer*) bestaande uit één coëfficiënt.

Dan vergelijken we de complexiteit van de MUI-vrije ontvanger met de complexiteit van de lineaire meergebruiker-egalisator. Zoals in hoofdstuk 5, maken we hierbij de veronderstelling dat het ontwerp van de ontvanger gebaseerd is op het overeenkomstig blind kanaalschattingsalgoritme. Als we dezelfde redenering volgen als in hoofdstuk 5 is het eenvoudig om aan te tonen dat de complexiteit van de MUI-vrije ontvanger veel kleiner is dan de complexiteit van de lineaire meergebruiker-egalisator. Merk verder op dat de MUI/ITI-vrije ontvanger goedkoper is dan de MUI-vrije ontvanger, omdat het gebruik van de lineaire blokcombinator vermeden wordt.

Uit de simulatieresultaten kunnen we de volgende conclusies trekken:

- Als we overgaan van een scenario waarbij het uitgestuurde vermogen voor elke toon hetzelfde is, naar een scenario waarbij het uitgestuurde vermogen voor elke toon wordt aangepast aan het vermogen van het kanaal op de overeenkomstige frequentie, dan neemt de performantie van de MUI-vrije ontvanger en de MUI/ITI-vrije ontvanger toe, terwijl de performantie van de lineaire meergebruiker-egalisator afneemt.

- De MUI-vrije ontvanger en de MUI/ITI-vrije ontvanger, gebaseerd op een scenario waarbij het uitgestuurde vermogen voor elke toon wordt aangepast aan het vermogen van het kanaal op de overeenkomstige frequentie (beste scenario voor deze ontvangers), hebben een performantie die respectievelijk beter is dan en vergelijkbaar is met die van de lineaire meergebruiker-egalisator, gebaseerd op een scenario waarbij het uitgestuurde vermogen voor elke toon hetzelfde is (beste scenario voor deze ontvanger).

# Deel II. Deterministische Blinde Ruimte-Tijd RAKE-Ontvangers

In deel II bestuderen we deterministische blinde ruimte-tijd RAKE-ontvangers voor een DS-CDMA systeem met meerdere ontvangstantennes. In een blokverwerkingscontext vergen deze ontvangers een veel kleinere hoeveelheid ontvangen samples dan stochastische blinde ruimte-tijd RAKE-ontvangers. Adaptieve implementaties van deterministische blinde ruimte-tijd RAKE-ontvangers zijn dus geschikt om te gebruiken in een sterk tijdsvariërende omgeving.

# Hoofdstuk 7. Deterministische Blinde Blokverwerking

In dit hoofdstuk ontwikkelen we deterministische blinde ruimte-tijd RAKE-ontvangers gebaseerd op blokverwerking. We beschouwen hiervoor zowel een deelruimteaanpak als een niet-deelruimteaanpak. Het verschil tussen deze twee aanpakken is dat de deelruimteaanpak gebaseerd is op een deelruimteontbinding, terwijl de niet-deelruimteaanpak dit niet is. We ontwikkelen blokverwerkingsmethoden voor zowel directe blinde symboolschatting als directe blinde egalisatorschatting, met oog voor rekenintensiviteit. De voorgestelde blokverwerkingsmethoden voor directe blinde symboolschatting zijn voor zover wij weten nog nooit eerder bestudeerd geweest in de literatuur. Sommige van de voorgestelde blokverwerkingsmethoden voor directe blinde egalisatorschatting zijn daarentegen wel al bestudeerd geweest in de literatuur, maar enkel in hun eenvoudigste vorm. Er bestaan ook interessante verbanden tussen de voorgestelde methoden, aan de ene kant, en de SSI-methoden (SSI: *subspace intersection*) van [LX97, vdVTP97] en de MRE-methode (MRE: *mutually referenced equalizer*) van [GDM97], die allen ontwikkeld werden voor een TDMA systeem (geen codering), aan de andere kant.

De voorgestelde blokverwerkingsmethoden zijn toepasbaar voor gelijk welk modulatieformaat van de datasymbolen. Indien de datasymbolen tot een reëel constellatiediagramma behoren, kunnen we deze methoden aanpassen om het reële karakter van de datasymbolen uit te buiten. De methoden die zo ontstaan noemen we RC-methoden (RC: *real constellation*).

In dit hoofdstuk wordt ook beschreven hoe de voorgestelde blokverwerkingsmethoden kunnen aangepast worden, indien zenddiversiteit gebruikt wordt (meerdere zendantennes per gebruiker). We spreiden dan dezelfde datasymboolsequentie met verschillende codesequenties en versturen de resulterende chipsequenties via verschillende zendantennes. De methoden die zo ontstaan noemen we TD-methoden (TD: *transmit diversity*).

Hoe kleiner de spreidingsfactor $N$, hoe groter de informatiesnelheid voor eenzelfde bandbreedte of hoe kleiner de bandbreedte voor eenzelfde informatiesnelheid. Daarom bestuderen we in dit hoofdstuk het geval $N = 1$. Dit geval noemt men codemodulatie, omdat een datasymboolsequentie dan gemoduleerd (en niet gespreid) wordt met een codesequentie. We tonen aan dat de methoden met een polynomiale beperking niet toepasbaar zijn voor $N = 1$. De RC-methoden met een polynomiale beperking zijn daarentegen ook niet toepasbaar voor $N = 1$, als de codesequenties reëel zijn, maar zijn wel toepasbaar voor $N = 1$, als de codesequenties complex zijn. De TD-methoden met een polynomiale beperking zijn tenslotte altijd toepasbaar voor $N = 1$.

Simulatieresultaten tonen aan dat de voorgestelde methoden robuust zijn tegen

een overschatting van de systeemorde. Bovendien zien we dat een beperkte kennis van de kanaalorde kan leiden tot een betere performantie, niettegenstaande we theoretisch gezien geen informatie over de kanaalorde nodig hebben. Tenslotte hebben we geïllustreerd dat, wanneer er geen ICI aanwezig is, de RC- en TD-methoden gebaseerd op codemodulatie goedkope en relatief performante alternatieven vormen voor bestaande signaalscheidingsalgoritmen die niet gebaseerd zijn op codering, zoals ILSP (*iterative least squares with projection*) en RACMA (*real analytical constant modulus algorithm*).

## Hoofdstuk 8. Deterministische Blinde Adaptieve Verwerking

In dit hoofdstuk ontwikkelen we deterministische blinde ruimte-tijd RAKE-ontvangers gebaseerd op adaptieve verwerking. Eerst wordt een gevensterde adaptieve verwerking besproken. We beschouwen hiervoor weer zowel een deelruimteaanpak als een niet-deelruimteaanpak. Bij de ontwikkeling van de gevensterde adaptieve verwerkingsmethoden wordt enkel aandacht besteed aan directe blinde symboolschatting. Dan wordt een exponentieel gewogen adaptieve verwerking besproken. We beschouwen hiervoor enkel een niet-deelruimteaanpak omdat dit de enige zinvolle aanpak is. Bij de ontwikkeling van de exponentieel gewogen adaptieve verwerkingsmethoden wordt weer enkel aandacht besteed aan directe blinde symboolschatting.

We hebben aangetoond dat adaptieve verwerking in sommige gevallen minder complex is dan blokverwerking. Simulaties illustreren dat de voorgestelde adaptieve methoden goede resultaten halen voor zowel een tijdsinvariante als een tijdsvariërende omgeving. Merk op dat in zo een tijdsvariërende omgeving de performantie van een blokverwerkingsalgoritme uitgevoerd op een grote hoeveelheid ontvangen samples niet meer aanvaardbaar is.

## Hoofdstuk 9. Uitbreidingen voor een DMT-DS-CDMA Systeem

De deterministische blinde ruimte-tijd RAKE-ontvangers gebaseerd op blokverwerking, voorgesteld in hoofdstuk 7, zijn ook toepasbaar voor een DMT-DS-CDMA systeem (hetzelfde kan gezegd worden voor de deterministische blinde ruimte-tijd RAKE-ontvangers gebaseerd op adaptieve verwerking, voorgesteld in hoofdstuk 8). Deze uitbreiding wordt uitgelegd in dit hoofdstuk.

We tonen aan dat het gebruik van een cyclische prefix de ICI in een DMT-

DS-CDMA systeem reduceert. Als het systeem bijvoorbeeld synchroon is, dan is het mogelijk om dit systeem op een zodanige manier te bekijken dat het lijkt alsof iedere kanaalorde gereduceerd wordt met de lengte van de cyclische prefix (als de kanaalorde groter is dan de lengte van de cyclische prefix) of gereduceerd wordt tot nul (als de kanaalorde kleiner is dan of gelijk is aan de lengte van de cyclische prefix). De geobserveerde ICI-reductie zorgt ervoor dat de complexiteit van de blokverwerkingsalgoritmen, voorgesteld in hoofdstuk 7, kan gereduceerd worden wanneer ze toegepast worden voor een bepaalde toon van een DMT-DS-CDMA systeem t.o.v. het geval ze toegepast worden voor het gerelateerde DS-CDMA systeem. Bovendien tonen simulatieresultaten aan dat deze complexiteitsreductie zelfs samengaat met een verbeterde performantie.

## Hoofdstuk 10. Conclusies en Verder Onderzoek

In deel I hebben we alternatieven voor meergebruiker-ontvangers ontwikkeld, die minder rekenintensief zijn en een vergelijkbare performantie hebben. Deze alternatieven, die gebaseerd zijn op het concept van blokspreiding, werden ontworpen voor een enkele ontvangstantenne en een ICI-beperkt quasi-synchroon propagatiemodel.

In deel II hebben we deterministische blinde ruimte-tijd RAKE-ontvangers bestudeerd voor een DS-CDMA systeem met meerdere ontvangstantennes. We hebben daarbij eerst aandacht besteed aan blokverwerking en daarna aan adaptieve verwerking. Tenslotte hebben we aangetoond hoe de ontwikkelde deterministische blinde ruimte-tijd RAKE-ontvangers ook kunnen toegepast worden voor een DMT-DS-CDMA systeem.

De nieuwe ontvangers voor blokspreidingssystemen, ontwikkeld in deel I, steunen op het gebruik van een enkele ontvangstantenne. Het zou interessant zijn om te onderzoeken hoe deze ontvangers kunnen uitgebreid worden naar methoden gebaseerd op meerdere ontvangstantennes, met als doel het aantal gebruikers dat het systeem kan onderhouden te verhogen.

In deel I hebben we gezien dat de MUI-vrije werking van de MUI-vrije ontvanger wordt bekomen door gebruik te maken van een shift-orthogonale set van codesequenties. Op basis van zo een shift-orthogonale set van codesequenties kunnen we gemakkelijk een orthogonale set van codesequenties construeren. Op het eerste gezicht laten de codesequenties die tot zo een set behoren een meer uniforme spreiding toe dan bijvoorbeeld Walsh-Hadamard codesequenties, die ook orthogonaal zijn. Vandaar dat men mag verwachten dat ze goede kandidaten zijn voor asynchrone DS-CDMA communicatie. De vraag is nu of ze betere kandidaten zijn dan bijvoorbeeld Gold codesequenties.

W-CDMA ontwerpen [OP98] voorzien het gebruik van trainingssequenties om

de gebruikers te detecteren. In die context zou het nuttig zijn om na te gaan hoe de blinde ruimte-tijd RAKE-ontvangers, die bestudeerd werden in deel II, kunnen gecombineerd worden met trainingsgebaseerde technieken.

Een ander aantrekkelijk onderzoeksonderwerp is de uitbreiding van de blinde ruimte-tijd RAKE-ontvangers gebaseerd op zenddiversiteit, die geïntroduceerd werden in deel II, naar blinde ontvangers voor systemen gebaseerd op andere (meer optimale) ruimte-tijd coderingstechnieken.

# Chapter 1

# Introduction

The last few years, there has been a rapidly growing interest in wireless communication systems. This is mainly caused by the following three attractive features of wireless systems:

- For most applications, a wireless system is cheaper than a wired system.
- Wireless systems allow communication in places where it is difficult to make wired connections.
- Wireless systems allow mobile users.

The crucial part of any wireless communication system is the *multiple-access* technique, i.e., the mechanism to share the common transmission medium between several users. The basic multiple-access techniques are *frequency-division multiple-access* (FDMA), *time-division multiple-access* (TDMA) and *code-division multiple-access* (CDMA). This thesis deals with the development of receivers for various CDMA systems. Digital signal processing plays a central role in the design of these receivers.

## 1.1 Multiple-Access Principles

Wireless communication systems are usually organized in a cellular fashion. The wireless operator divides the covered area into different cells and places a base station in the center of each cell. As a result, two directions of communication are possible: the communication from the base station to the users, called the *forward link* or *downlink*, and the communication from the users

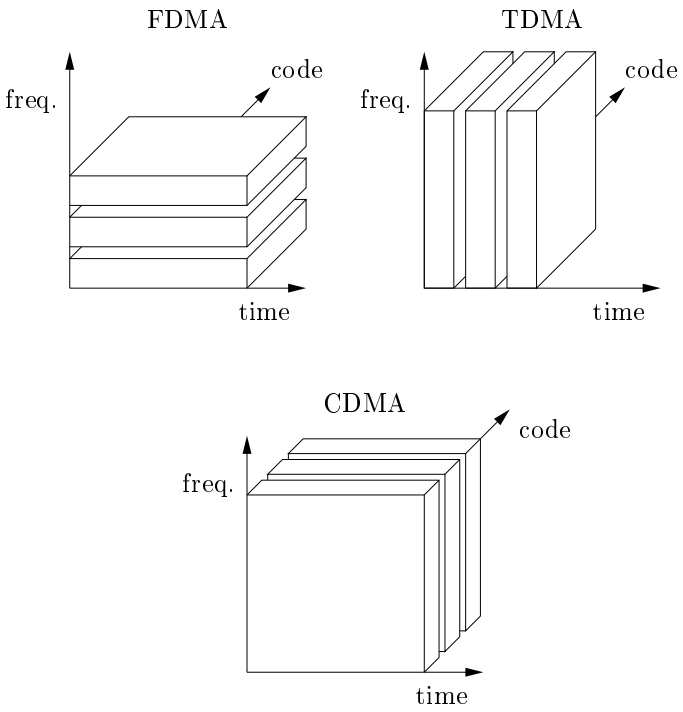



Figure 1.1: Basic multiple-access techniques.

to the base station, called the *reverse link* or *uplink*. These two directions of communication are usually separated in frequency, which is referred to as *frequency division duplexing* (FDD), or in time, which is referred to as *time division duplexing* (TDD). Most second generation systems, such as the GSM system [MP92] and the IS-95 system [Vit95], as well as most third generation system proposals [OP98] are based on FDD. From now on, we will only focus on uplink communication.

Each cell of a cellular system is assigned a certain frequency band, which can be reused by other cells. The ratio of the total number of cells over the number of cells using the same frequency band is called the *frequency reuse factor*. Note that the width of the frequency band used by a cell is equal to the width of the available frequency spectrum divided by the frequency reuse factor. Keeping this in mind, we now discuss the three basic multiple-access techniques (see **figure 1.1**).

In a *frequency-division multiple-access* (FDMA) or *time-division multiple-access* (TDMA) system, each user within the same cell operates in a unique frequency or time channel. As a result, there is no intra-cell interference. The frequency

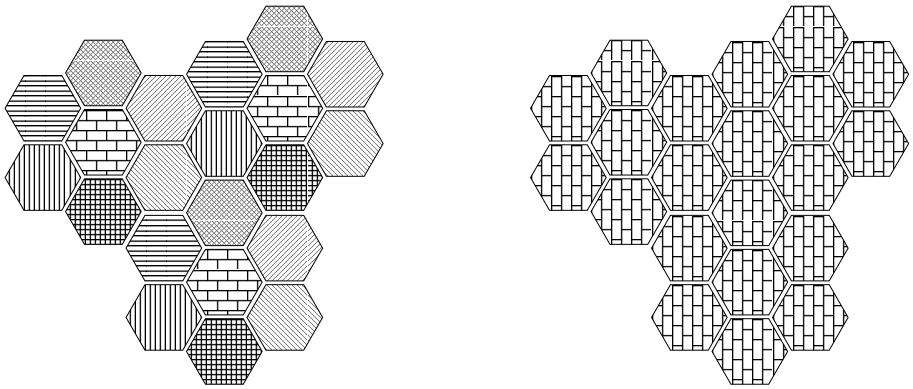


Figure 1.2: Cellular wireless communication system with a frequency reuse factor of seven, in the left figure, and one, in the right figure. All the cells that are hatched in a similar fashion use the same frequency band.

reuse factor that is typically used is seven, which means that the distance between two cells using the same frequency band is large (see **figure 1.2**). Hence, the inter-cell interference is negligible. The current GSM system [MP92] is based on a combination of FDMA and TDMA. Each frequency channel in a GSM system has a bandwidth of 200 kHz and is divided into eight time channels.

In a *code-division multiple-access* CDMA system, on the other hand, each user within the same cell operates in the same frequency and time channel. As a result, there is intra-cell interference. The frequency reuse factor that is typically used is one (this feature is called *universal frequency reuse*), which means that the distance between two cells using the same frequency band is small (see **figure 1.2**). Hence, the inter-cell interference is not negligible. The question then is how the *multi-user interference* (MUI), defined as the sum of the intra-cell and inter-cell interference, can be suppressed. The answer originates from the fact that, in a CDMA system, each user spreads his information-bearing signal into a wideband signal, using specific code information. The receiver can then for instance reverse the spreading operation corresponding to the desired user, under the assumption that this user's code information is known. If all the code information is randomly chosen, this operation will despread the received wideband signal into a smallband signal (with the same bandwidth as information-bearing signals), that contains much more power from the desired user than from the other users. This allows us to reconstruct the desired user's information-bearing signal. This multiple-access principle of CDMA is illustrated in **figure 1.3**. To keep the residual MUI after despreading as low as possible, each user has to emit the least amount of power that is needed for reliable transmission. This requires strict power control. If we do not employ

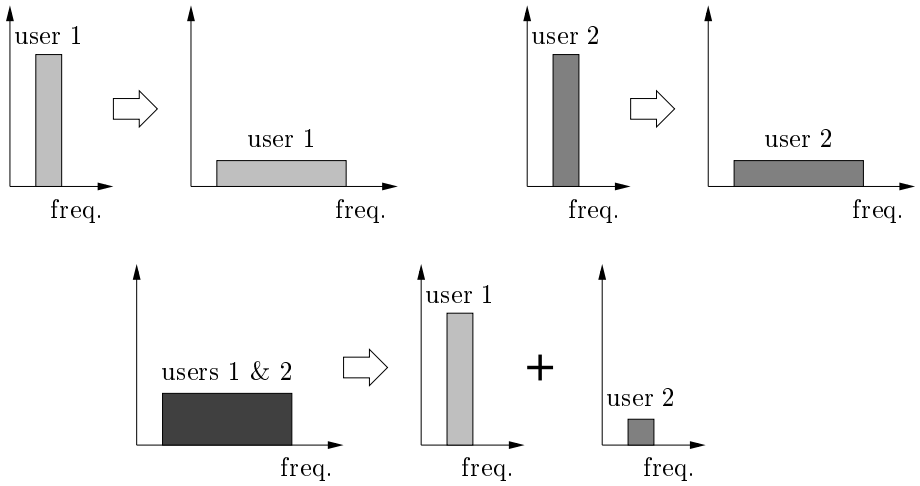



Figure 1.3: Multiple-access principle of CDMA. In the top figure, two users spread their information-bearing signal into a wideband signal before transmitting it. In the bottom figure, the receiver then reverses the spreading operation corresponding to user 1.

strict power control, users near the base station, which are received with high power, will have a significant impact on the detection of users far from the base station, which are received with low power. This problem is known as the *near-far problem*. However, we can also use more advanced receivers, that do not require strict power control (instead of the receiver that simply reverses the spreading operation corresponding to the desired user). Note that some of these receivers are based on all the code information, and not only on the desired user's code information. Whatever the receiver we use, CDMA has a number of interesting advantages over FDMA or TDMA:

- For voice transmission, a user is active less than half the time. In an FDMA or TDMA system, where each user is assigned a unique frequency or time channel, this causes a great loss of capacity, which can only be resolved by giving another user access to the channel of the silent user. However, this is only possible when the silent period is long enough. In a CDMA system, where each user is assigned the same frequency and time channel, this problem does not occur.
- In an FDMA or TDMA system, contiguous cells are assigned different frequency bands (frequency reuse factor of 7). Hence, somewhat after a user has crossed the border between two cells, he drops the link with the first cell's base station and starts communicating with the second

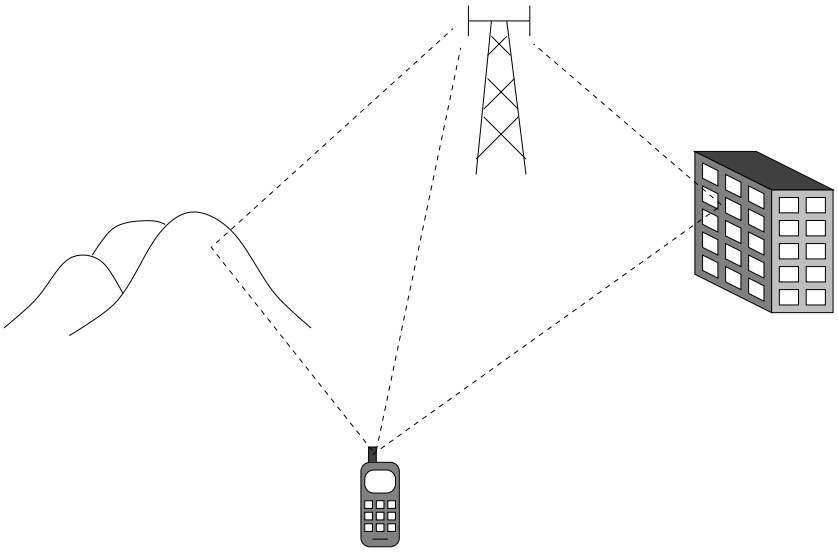


Figure 1.4: Multi-path propagation environment.

cell's base station. This is called *hard handoff*. The universal frequency reuse feature of CDMA, on the other hand, allows the use of *soft handoff*. Somewhat before a user will cross the border between two cells, he starts communicating with the second cell's base station, without dropping the link with the first cell's base station. This creates a kind of receiver diversity that can be exploited to improve the performance.

- Due to scattering, a transmitted signal will be received from a number of different paths (see **figure 1.4**). The received signals corresponding to different paths are all copies of the same transmitted signal with different gains, phases and delays. Since in a CDMA system each user is usually assigned a larger bandwidth than in an FDMA or TDMA system, the different paths are better isolated. Hence, in a CDMA system the multi-path interference can be suppressed more efficiently than in an FDMA or TDMA system.

## 1.2 More Details on CDMA

In a digital communication system, a bit stream is usually mapped onto a so-called *data symbol sequence*, with data symbols belonging to some finite alphabet (each data symbol represents a fixed number of bits). In a CDMA

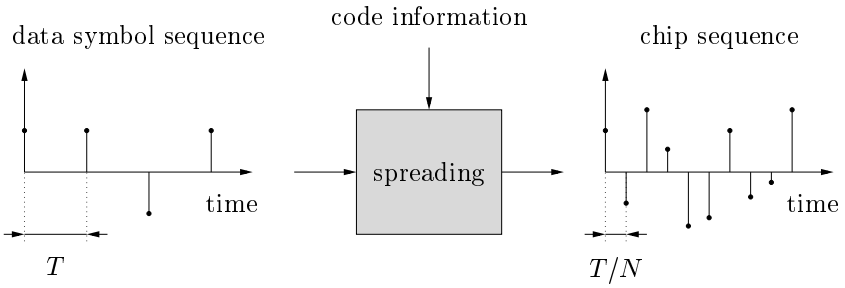



Figure 1.5: Spreading operation used in a CDMA system.

system, each user spreads his *data symbol sequence* with rate $1/T$ (the *data symbol rate*) into a so-called *chip sequence* with rate $N/T$ (the *chip rate*), using specific code information (see **figure 1.5**). Note that $N \geq 1$ is called the *spreading factor*. The way in which this spreading operation is carried out depends on the type of CDMA. All users then transmit their chip sequence within the same frequency and time channel. The receiver finally tries to recover the desired user's data symbol sequence, based on either the knowledge of the desired user's code information or the knowledge of all the code information. For simplicity, we will always consider chip rate sampling at the receiver side (see section 2.2.3).

Before describing other types of CDMA, which we will consider in this thesis, we start with a discussion of the most popular type of CDMA, namely direct-sequence CDMA (DS-CDMA). DS-CDMA is currently used in the IS-95 system [Vit95], but it is also the multiple-access technique that is employed in most third generation system proposals [OP98]. Such third generation system proposals are referred to as wideband CDMA (W-CDMA) proposals. The IS-95 system has a bandwidth of 1.25 MHz, while the W-CDMA proposals aim at bandwidths of 5 MHz or more.

## 1.3 DS-CDMA System

In a *DS-CDMA system*, each user spreads his data symbol sequence into a chip sequence by taking the following steps (see **figure 1.6**):

1. Spread the *data symbol sequence* by a factor $N$ using a specific code sequence (spreader):
   a. Repeat each data symbol $N$ times.
   b. Multiply the resulting symbol sequence with the code sequence.

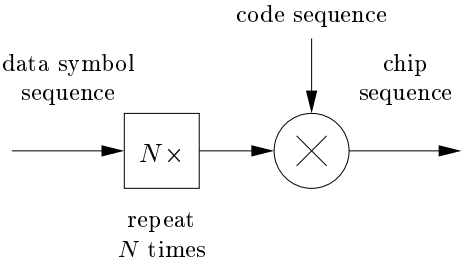



Figure 1.6: Spreading operation used in a DS-CDMA system.

**Remark 1.3.1** If the code sequences spread successive data symbols in the same way (the spreading operation is periodic from data symbol to data symbol) they are called *short* or *periodic*. If the code sequences spread successive data symbols in a different way (the spreading operation is aperiodic from data symbol to data symbol) they are called *long* or *aperiodic*. □

At the receiver side we can then reverse the spreading operation corresponding to the desired user. For short as well as long code sequences this is done as follows:

1. Sample the received signal at chip rate.
2. Despread the resulting *received sequence* by a factor $N$ using the desired user's code sequence (correlator; see **figure 1.7**):
   a. Multiply the received sequence with the complex conjugate code sequence.
   b. Sum $N$ successive symbols.
   c. Downsample by a factor $N$.
3. Scale the resulting symbol sequence in the appropriate way.

In the following example we demonstrate how the correlator of step 2 depicted in figure 1.7 can suppress the multi-user interference (MUI), i.e., the interference from other users, and the inter-chip interference (ICI), i.e., the interference from neighboring chips due to the multi-path propagation environment (see figure 1.4):

**Example 1.3.2** We assume that all the users are synchronous and have a discrete-time channel of order $L-1$ ($1 < L \ll N$). For this case, the received sequence consists of the superposition of 1 scaled version and $L-1$ scaled and

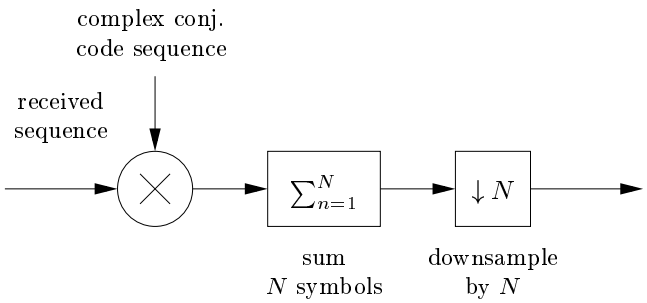



Figure 1.7: Correlator.

delayed versions of the desired user's chip sequence, plus for every interfering user the superposition of 1 scaled version and $L-1$ scaled and delayed versions of this interfering user's chip sequence (see **figure 1.8**). If the crosscorrelation function between the desired user's code sequence and any interfering user's code sequence is small at delays 0 to $L-1$, the correlator will only put a small part of the interfering user's received power in the resulting symbol sequence (of course, the higher the interfering user's received power, the more interference we obtain from this user; this causes the earlier described near-far problem) and hence the correlator will suppress the MUI. Note that only when $L = 1$ (no ICI) and all the code sequences are orthogonal, the correlator will completely remove all the interfering users' received power (no near-far problem). However, as indicated above, we consider $1 < L \ll N$ in this example. If the autocorrelation function of the desired user's code sequence at delays 1 to $L-1$ is small, the correlator will only put a small part of the power of the $L-1$ scaled and delayed versions of the desired user's chip sequence in the resulting symbol sequence and hence the correlator will suppress the ICI. □

Next to the very simple receiver we have discussed above, many other receivers for a DS-CDMA system have been developed in the literature. In the following two sections, we will give an overview of these receivers.

## 1.4 Receivers for a DS-CDMA System

In this section, we review a number of existing receivers for a DS-CDMA system. The most popular receiver is the *conventional receiver* or so-called *RAKE receiver* (Price and Green [PG58])[1].

[1] The conventional receiver is also called *RAKE receiver*, because it has the structure of a rake.

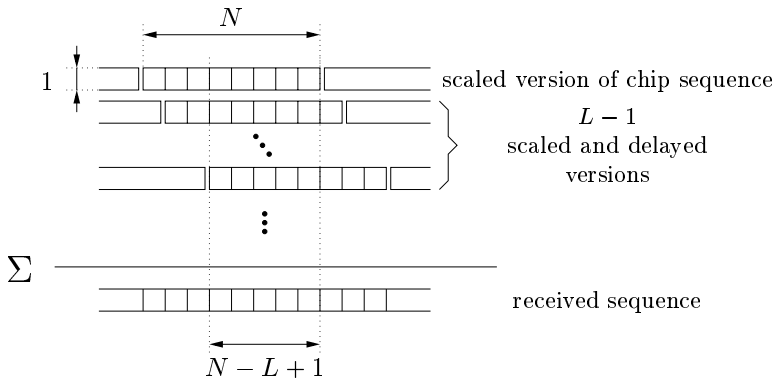



Figure 1.8: Structure of the received sequence.

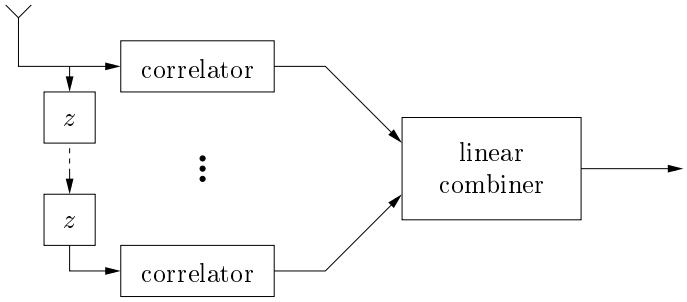



Figure 1.9: RAKE receiver.

### 1.4.1 RAKE Receiver

The RAKE receiver is based on the use of a single receive antenna. It consists of a bank of correlators for the desired user, where each correlator (see figure 1.7) despreads the received sequence matched to a different channel tap, using the desired user's code sequence, followed by a linear combiner, which linearly combines the different correlator outputs in order to generate an estimate of the desired user's data symbol sequence (see **figure 1.9**). Note that the *structure* of the RAKE receiver only requires the knowledge of the desired user's code sequence. RAKE receivers can be developed for short as well as long code sequences.

Very popular is the matched filter linear combiner, resulting in the so-called *coherent RAKE receiver* [Vit95]. The coherent RAKE receiver maximizes the signal-to-noise ratio (SNR), but it suffers from the near-far problem. Therefore,

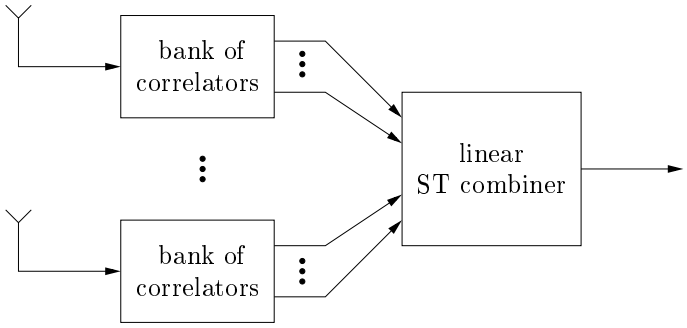



Figure 1.10: ST-RAKE receiver.

research has been focusing on the development of receivers that are robust to this near-far problem. This has lead to the design of *space-time RAKE (ST-RAKE) receivers* and *multi-user receivers*. For the latter we make a distinction between *multi-user RAKE receivers* and *multi-user equalizers*.

### 1.4.2 ST-RAKE Receiver

The ST-RAKE receiver is an extension of the RAKE receiver for multiple receive antennas. It consists of a set of banks of correlators for the desired user (one bank at each receive antenna), followed by a linear ST combiner, which linearly combines the different correlator outputs in order to generate an estimate of the desired user's data symbol sequence (see **figure 1.10**). Note that the *structure* of the ST-RAKE receiver only requires the knowledge of the desired user's code sequence. ST-RAKE receivers can be developed for short as well as long code sequences.

A well-known way to look at this ST-RAKE receiver is as a *linear ST chip equalizer*, which operates on the different received sequences in order to generate an estimate of the desired user's chip sequence, followed by a correlator, which despreads the linear ST chip equalizer output. Many different linear ST chip equalizers can be applied. The most popular one is the matched filter linear ST chip equalizer, resulting in the so-called *coherent ST-RAKE receiver* [SNXP93, NPK94]. Compared to the coherent RAKE receiver, the coherent ST-RAKE receiver also maximizes the signal-to-noise ratio (SNR), but it is less sensitive to the near-far problem. However, instead of applying a matched filter linear ST chip equalizer, we can more effectively exploit the *spatial diversity* provided by the different receive antennas, suppressing the ICI (if present) and MUI to a greater extent. This can for instance be done using the maximal signal-to-interference-plus-noise ratio (SINR) linear

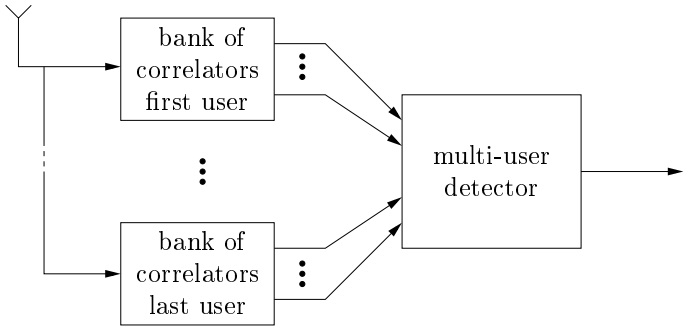



Figure 1.11: Multi-user RAKE receiver.

ST chip equalizer [RZ97, CZ98], which maximizes the SINR. Special cases of this equalizer are the minimum mean-square error (MMSE) linear ST chip equalizer [LZ97, WLLZ98] and the minimum variance distortionless response (MVDR) linear ST chip equalizer [WLLZ98].

### 1.4.3 Multi-User RAKE Receiver

The multi-user RAKE receiver is based on the use of a single receive antenna. It is assumed that the number of users is smaller than or equal to the spreading factor. The multi-user RAKE receiver consists of a set of banks of correlators (one bank for each user), followed by a *multi-user detector*, which operates on the different correlator outputs in order to generate an estimate of the desired user's data symbol sequence (see **figure 1.11**). Note that the *structure* of the multi-user RAKE receiver requires the knowledge of all the code sequences. In principle, short as well as long code sequences can be used. However, for long code sequences the multi-user detector would have to change from data symbol to data symbol. Hence, multi-user RAKE receivers are usually based on the use of short code sequences.

#### Optimal Multi-User Detector

The first multi-user detector developed was the optimal (maximum likelihood) multi-user detector, presented by Verdu in [Ver86] (multi-user extension of the approach proposed by Forney, Jr. [For92]).

**Suboptimal Multi-User Detector**

Since the optimal multi-user detector has a complexity that is exponential in the number of users, several less complex suboptimal multi-user detectors have been developed, among them the *linear* and *non-linear* multi-user detectors.

- *Linear Multi-User Detector:*

  The most popular linear multi-user detectors are the zero-forcing (ZF) linear multi-user detector (also known as the decorrelating linear multi-user detector) [LV89, LV90, XSR90, ZB96, ZB95, KM95] and the MMSE linear multi-user detector [XSR90]. Note that for asynchronous communication and/or in the presence of ICI, there exists no finite-length ZF linear multi-user detector.

  **Remark 1.4.1** In case multiple receive antennas are present, we can use a bank of linear multi-user detectors (one linear multi-user detector at each receive antenna), followed by a linear combiner, which linearly combines the different linear multi-user detector outputs in order to generate an estimate of the desired user's data symbol sequence [Zvo96]. □

- *Non-Linear Multi-User Detector:*

  One type of non-linear multi-user detectors consists of decision feedback (DF) implementations of linear multi-user detectors. ZF-DF and MMSE-DF multi-user detectors can be found in [DH95, XSR90].

  Next to the DF multi-user detectors, there also exists subtractive interference multi-user detectors, such as the successive interference cancellation multi-user detector [PH94] and the parallel interference cancellation multi-user detector [VA90]. Successive interference cancellation works well in a bad power-controlled situation, while parallel interference cancellation works well in a good power-controlled situation.

### 1.4.4 Multi-User Equalizer

A multi-user equalizer is based on the use of a single receive antenna or multiple receive antennas. It is assumed that the number of users is smaller than the spreading factor or the product of the spreading factor with the number of receive antennas. A multi-user equalizer directly operates on the received sequence or the different received sequences in order to generate an estimate of the desired user's data symbol sequence, without imposing any specific structure. Hence, the *structure* of the multi-user equalizer does not imply the knowledge of any code sequences. In principle, short as well as long code sequences can be used. However, like before, for long code sequences the multi-user equalizer

would have to change from data symbol to data symbol. Hence, multi-user equalizers are usually based on the use of short code sequences. Mostly suboptimal multi-user equalizers are discussed in literature.

#### Suboptimal Multi-User Equalizer

Like the suboptimal multi-user detectors, the suboptimal multi-user equalizers can be divided into two classes, namely the *linear* and *non-linear* multi-user equalizers.

- *Linear Multi-User Equalizer:*

  The ZF linear multi-user equalizer is presented in [TG96, TG97, WP98, KB93, KKKB96]. The MMSE linear multi-user equalizer is presented in [MH94, Mil95, KKKB96, SD99, GSP98]. Finally, the MVDR linear multi-user equalizer is presented in [WP98, XT98]. Note that under certain identifiability conditions (Tsatsanis and Giannakis [TG96]), there always exists a finite-length ZF linear multi-user equalizer.

  For a single-receive antenna, finite-length linear multi-user equalizers have a better performance than the corresponding finite-length linear multi-user detectors. This is because a multi-user equalizer does not impose a specific receiver structure, while a multi-user RAKE receiver does.

- *Non-Linear Multi-User Equalizer:*

  Most non-linear multi-user equalizers discussed in literature are DF implementations of linear multi-user equalizers. ZF-DF and MMSE-DF multi-user equalizers are presented by Klein et al [KKKB96].

## 1.5 Blind Receivers for a DS-CDMA System

For DS-CDMA systems, recent research aims at the development of *blind* receivers. These receivers do not require any training sequences to detect a user. Note that training sequences usually consume a considerable part of the transmission time. Moreover, if a training sequence is present, blind receivers can generally be adapted to incorporate the knowledge of this sequence, resulting into so-called *semi-blind* receivers, which generally perform better than their training-based counterparts. For a DS-CDMA system, mainly blind ST-RAKE receivers and blind multi-user equalizers have been investigated. For both receiver structures the underlying data model is multi-input multi-output (MIMO).

- For the ST-RAKE receiver, the number of inputs is equal to the number of users and the number of outputs is equal to the number of receive antennas.
- For the multi-user equalizer, the number of inputs is also equal to the number of users and the number of outputs is equal to the spreading factor or the product of the spreading factor with the number of receive antennas. In this context, note that when the channels are time invariant, the MIMO data model underlying the multi-user equalizer will also be time invariant, if the code sequences are short, while it will change from data symbol to data symbol, if the code sequences are long.

We can make a distinction between *deterministic* and *stochastic* blind receivers. Deterministic blind receivers exploit deterministic properties of the data model. They can usually be divided into two classes: the *subspace* deterministic blind receivers (they are based on a subspace decomposition) and the *non-subspace* deterministic blind receivers (they are not based on a subspace decomposition). Stochastic blind receivers, on the other hand, exploit statistical properties of the data model. Since the work of Tong et al [TXK94], it has become clear that for single-input multi-output (SIMO) systems blind detection based on second order statistics (SOS) is feasible. For MIMO systems, on the other hand, blind detection based on SOS is only feasible, if we can exploit some additional information. In a DS-CDMA system this additional information can be the code information and therefore most of the recently developed stochastic blind ST-RAKE receivers and stochastic blind multi-user equalizers are based on SOS.

### 1.5.1 Blind ST-RAKE Receivers

We can make a distinction between blind ST-RAKE receivers based on blind channel estimation and those based on direct blind equalizer estimation.

#### Blind Channel Estimation

Within the framework of the ST-RAKE receiver, SOS blind multi-user channel estimation algorithms are discussed in [SNXP93, LZ97, WLLZ98]. These algorithms estimate the channel as the principal eigenvector of a difference between two autocorrelation matrices: a pre-correlation autocorrelation matrix (calculated before correlation with the desired user's code sequence) and a post-correlation autocorrelation matrix (calculated after correlation with the desired user's code sequence).

Estimating the desired user's channel with one of the above methods only

requires the desired user's code sequence, namely through the fact that these algorithms make use of the data at the output of the set of banks of correlators (one bank at each receive antenna). Based on the knowledge of the desired user's channel we can then easily design the desired user's matched filter linear ST chip equalizer and based on the knowledge of all the channels, we can then easily design the MMSE or MVDR linear ST chip equalizer.

**Remark 1.5.1** Note that it is also possible to compute the desired user's MMSE or MVDR linear ST chip equalizer based only on the knowledge of the desired user's channel [LZ97, WLLZ98]. □

#### Direct Blind Equalizer Estimation

Non-subspace deterministic blind linear ST chip equalizer estimation algorithms are presented in [LX96a, LZ97].

SOS blind maximal SINR linear ST chip equalizer estimation algorithms are presented in [RZ97, CZ98]. These algorithms are related to the SOS blind ST-RAKE channel estimation algorithms, discussed above.

All the above linear ST chip equalizer estimation algorithms only require the knowledge of the desired user's code sequence.

### 1.5.2 Blind Multi-User Equalizers

As already mentioned, when the channels are time invariant, the MIMO data model underlying the multi-user equalizer will also be time invariant, if the code sequences are short, while it will change from data symbol to data symbol, if the code sequences are long. That is why most blind multi-user equalizers are based on the use of short code sequences. In this section, we give an overview of these blind multi-user equalizers. Like before, we can make a distinction between blind multi-user equalizers based on blind channel estimation and those based on direct blind equalizer estimation.

#### Blind Channel Estimation

Within the framework of the linear multi-user equalizer, subspace deterministic blind multi-user channel estimation algorithms are presented in [BA96, LX96b, TX97, TG97, WP98] (although these algorithms are usually introduced using SOS, they are basically deterministic algorithms). These algorithms are strongly related to the work that has been carried out for SIMO systems [MDCM95, AMCG+97, HAMW97].

Within the framework of the linear multi-user equalizer, SOS blind multi-user channel estimation algorithms are presented in [SD99, XT99]. In [SD99], a linear prediction approach is used. Related linear prediction approaches for SIMO systems are discussed in [Din97, AMML97, GD97]. Correlation matching is discussed in [XT99]. Related correlation matching approaches for SIMO systems are discussed in [GH97, ZT97a, ZT97b]. Note that the SOS blind multi-user channel estimation algorithms discussed above have milder identifiability conditions than the subspace deterministic blind multi-user channel estimation algorithms.

Estimating the desired user's channel with one of the above methods only requires the desired user's code sequence. Based on the knowledge of all the channels and code sequences, we can then easily design the desired user's ZF, MMSE or MVDR linear multi-user equalizer.

**Remark 1.5.2** Note that it is also possible to compute the desired user's ZF, MMSE or MVDR linear multi-user equalizer based only on the knowledge of the desired user's channel and code sequence [WP98, SD99]. □

#### Direct Blind Equalizer Estimation

SOS constrained minimum output energy (MOE) linear multi-user equalizer estimation algorithms are discussed in [Tsa97, TX98]. In [TX98], Capon beamforming ideas are used to calculate the optimal constraint. In this case, the constrained MOE linear multi-user equalizer approaches the maximal SINR linear multi-user equalizer, for decreasing noise. For SIMO systems, a similar algorithm is presented in [XT98]. An SOS MMSE linear multi-user equalizer estimation algorithm is developed by Gesbert et al [GSP98]. A part of this algorithm is related with the subspace deterministic blind multi-user channel estimation algorithms.

All the above linear multi-user equalizer estimation algorithms only require the knowledge of the desired user's code sequence.

## 1.6 DS-CDMA-BS System

After this discussion of the well-known DS-CDMA system, we now focus on the *DS-CDMA system based on block spreading (DS-CDMA-BS system)*, introduced by Cirpan and Tsatsanis [CT97]. The spreading operation used in a DS-CDMA-BS system can be interpreted as the spreading operation used in a DS-CDMA system followed by *chip interleaving* [CT97]. Note that the authors of [CT97] have only introduced the DS-CDMA-BS system to facilitate blind

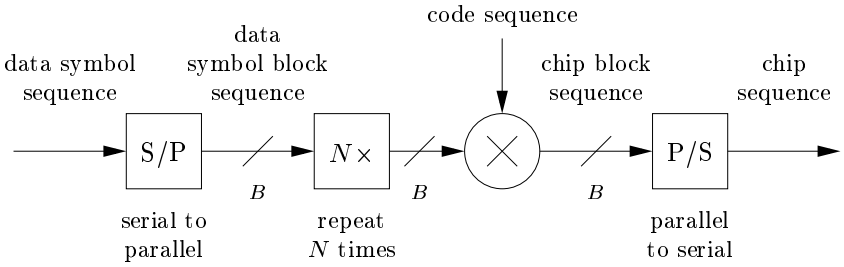



Figure 1.12: Spreading operation used in a DS-CDMA-BS system.

multi-user channel estimation. They did not propose special receiver structures for such a system.

In a DS-CDMA-BS system, each user spreads his data symbol sequence into a chip sequence by taking the following steps (see **figure 1.12**):

1. Block the *data symbol sequence* (block size $B$).
2. Spread the resulting *data symbol block sequence* by a factor $N$ using a specific code sequence (block spreader):
   a. Repeat each data symbol block $N$ times.
   b. Multiply the resulting block sequence with the code sequence.
3. Deblock the resulting *chip block sequence* (block size $B$).

Note that remark 1.3.1 still holds (replace data symbol by data symbol block). At the receiver side we can then reverse the spreading operation corresponding to the desired user. For short as well as long code sequences this is done as follows:

1. Sample the received signal at chip rate.
2. Block the resulting *received sequence* (block size $B$).
3. Despread the resulting *received block sequence* by a factor $N$ with the desired user's code sequence (block correlator; see **figure 1.13**):
   a. Multiply the received block sequence with the complex conjugate code sequence.
   b. Sum $N$ successive blocks.
   c. Downsample by a factor $N$.

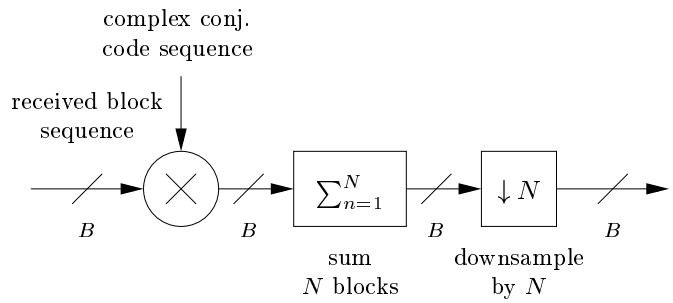



Figure 1.13: Block correlator.

4. Transform the resulting block sequence in the appropriate way.

5. Deblock the resulting block sequence (block size $B$).

In the following example we demonstrate how the block correlator of step 3 depicted in figure 1.13 can suppress the multi-user interference (MUI), i.e., the interference from other users, and the inter-chip-block interference (ICBI), i.e., the interference from neighboring chip blocks due to the multi-path propagation environment (see figure 1.4):

**Example 1.6.1** We assume that all the users are synchronous and have a discrete-time channel of order $L-1$ ($1 < L \ll N$). For this case, the received block sequence consists of the sum of 1 transformed version and $\lceil (L-1)/B \rceil$ transformed and delayed versions of the desired user's chip block sequence, plus for every interfering user the sum of 1 transformed version and $\lceil (L-1)/B \rceil$ transformed and delayed versions of this interfering user's chip block sequence (see **figure 1.14**). If the crosscorrelation function between the desired user's code sequence and any interfering user's code sequence is small at delays 0 to $\lceil (L-1)/B \rceil$, the block correlator will only put a small part of the interfering user's received power in the resulting block sequence (of course, the higher the interfering user's received power, the more interference; this causes the earlier described near-far problem) and hence the block correlator will suppress the MUI. Note that only when $L = 1$ (no ICBI) and all the code sequences are orthogonal, the block correlator will completely remove all the interfering users' received power (no near-far problem). However, as indicated above, we consider $1 < L \ll N$ in this example. If the autocorrelation function of the desired user's code sequence at delays 1 to $\lceil (L-1)/B \rceil$ is small, the block correlator will only put a small part of the power of the $\lceil (L-1)/B \rceil$ transformed and delayed versions of the desired user's chip block sequence in the resulting block sequence and hence the block correlator will suppress the ICBI. □

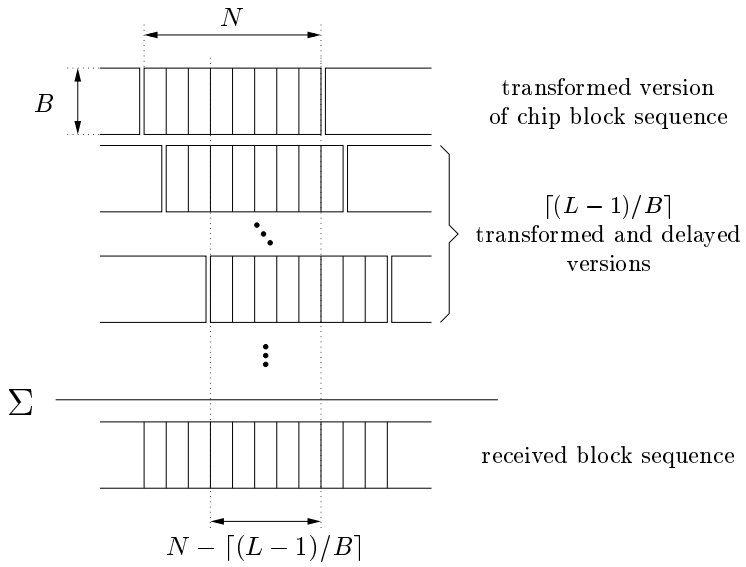



Figure 1.14: Structure of the received block sequence.

Comparing examples 1.6.1 and 1.3.2 (see figures 1.14 and 1.8), we notice that for the DS-CDMA-BS system there exist $N - \lceil (L-1)/B \rceil$ received sample blocks that are only affected by one data symbol block per user, while for the DS-CDMA system there exist $N-L+1$ received samples that are only affected by one data symbol per user (remember we considered synchronous users in examples 1.6.1 and 1.3.2). For the DS-CDMA-BS system, this means that if we would modify the block correlator in such a way that only those $N-\lceil (L-1)/B \rceil$ received sample blocks are taken into account, it is possible to design a special set of code sequences such that the block correlator completely removes the MUI and ICBI. Similarly, for the DS-CDMA system, this means that if we would modify the correlator in such a way that only those $N-L+1$ received samples are taken into account, it is possible to design a special set of code sequences such that the correlator completely removes the MUI and ICI. For the DS-CDMA-BS system, this would require that all the $(N-\lceil (L-1)/B \rceil) \times 1$ subvectors of the $J$ ($J$ is the number of users) code vectors of size $N \times 1$ should be orthogonal. Since there are $J(\lceil (L-1)/B \rceil + 1)$ such subvectors, this would limit $J$ to $(N - \lceil (L-1)/B \rceil)/(\lceil (L-1)/B \rceil + 1)$. Similarly, for the DS-CDMA system, this would require that all the $(N-L+1) \times 1$ subvectors of the $J$ ($J$ is the number of users) code vectors of size $N \times 1$ should be orthogonal. Since there are $JL$ such subvectors, this would limit $J$ to $(N-L+1)/L$. As a result, for the DS-CDMA-BS system this limitation on the number of users would be acceptable for any $L$ if we take $B \geq L-1$, while for the DS-CDMA system this limitation on the number of users would only be acceptable for

$L = 2$ (remember we considered $1 < L \ll N$ in examples 1.6.1 and 1.3.2). *This remarkable feature of a DS-CDMA-BS system, which has to the best of our knowledge not been recognized so far, will be the key to the development of new receivers for a DS-CDMA-BS system.*

## 1.7 DMT-Based CDMA Systems

Multi-tone (MT) modulation, also known as multi-carrier (MC) modulation or orthogonal frequency-division multiplexing (OFDM), is an attractive modulation technique that has become very popular lately [Bin90]. Although one would expect that such a technique divides the available frequency band into different non-overlapping frequency channels, MT modulation divides this band into different *overlapping* frequency channels with specific orthogonality properties. The centers of these bands are called the tones. Discrete multi-tone (DMT) modulation is the discrete implementation of MT modulation. It transforms the data symbol sequence into a so-called *DMT-modulated data symbol sequence* by taking the following steps:

1. Block the *data symbol sequence* (block size $P$).
2. Transform the resulting *DMT symbol sequence* (DMT symbol size $P$) by the inverse discrete Fourier transformation (IDFT). Hence, the $P$ *DMT subsymbol sequences*, which are contained in the DMT symbol sequence, will modulate $P$ different tones.
3. Add a *cyclic prefix* of length $\mu$ ($\mu \ll P$).
4. Deblock the resulting block sequence (block size $P + \mu$).

DMT modulation has two interesting properties. First of all, each tone can be optimally (in the sense of performance or capacity) modulated, if we know the signal-to-noise ratio (SNR) on this tone. Second, DMT modulation allows a simple channel equalization. For instance, if the channel order is shorter than or equal to the cyclic prefix, the inter-symbol interference (ISI) w.r.t. the DMT symbol sequence, i.e., the interference from neighboring DMT symbols, and the inter-tone interference (ITI), i.e., the interference from other DMT subsymbols in the same DMT symbol, can be completely removed by a DMT demodulator, which removes the cyclic prefix and performs a discrete Fourier transformation (DFT). However, in some applications, such as digital subscriber line (DSL) modems (wired), the channel order is usually much longer than the cyclic prefix. An existing approach to solve this problem is to shorten the channel using a linear time domain equalizer (TEQ) [ADC96, LCC95, MYR96, VBM95, VKS96]. A general disadvantage of this approach is that the TEQ equalizes all the

tones in a combined fashion and as a result limits the performance of the system. Therefore, we have recently proposed a new approach to solve this problem [VALM+99]. The main idea of this approach is to equalize each tone separately, using a linear frequency domain equalizer (FEQ) per tone. We have shown in [VALM+99], that for the same operation window the optimal FEQ approach always results in a higher capacity than the optimal TEQ approach, while their complexities in transmission mode are comparable. Note hereby that the optimal TEQ approach is almost never used, since the related criterion is highly complex. Therefore this criterion is usually simplified to an MSE criterion, which results in a suboptimal TEQ approach. The criterion related to the optimal FEQ approach, on the other hand, does not have to be simplified since it already corresponds to an MSE criterion. For a further discussion on this FEQ approach, see [VALM+00, VALMP00, VALMP99, VAPLM99].

We will now present some DMT-based CDMA systems that combine the advantages of DMT modulation with CDMA.

One possibility is to perform the spreading operation used in a DS-CDMA system on the DMT-modulated data symbol sequence. We then obtain the *DMT-CDMA system*, introduced by Vandendorpe [Van95]. Another possibility is to perform the spreading operation used in a DS-CDMA-BS system on the DMT-modulated data symbol sequence. We then obtain the new *DMT-CDMA system based on block spreading (DMT-CDMA-BS system)*. Note that we usually do not add a cyclic prefix in a DMT-CDMA or DMT-CDMA-BS system. For both possibilities, the spacing between the different tones is $1/(PT)$. A simple way to design a receiver for a DMT-CDMA or DMT-CDMA-BS system is to take a receiver for a DS-CDMA or DS-CDMA-BS system and to put a DMT demodulator behind it. In this thesis, we will give such a receiver the same name as the corresponding receiver for a DS-CDMA or DS-CDMA-BS system.

A last possibility, we will consider in this work, is to perform the spreading operation used in a DS-CDMA system on the different DMT subsymbol sequences. We then obtain the *DMT-DS-CDMA system*, introduced by DaSilva and Sousa [DS94]. In contrast with the previously described DMT-based CDMA systems, we do consider a cyclic prefix in a DMT-DS-CDMA system. The actual spreading factor therefore is $N(P+\mu)/P$. However, since the length of the cyclic prefix $\mu$ is generally much smaller than the DMT symbol size $P$, this actual spreading factor can be very well approximated by $N$. The spacing between the different tones now is $N/(PT)$.

**Remark 1.7.1** Note that there also exists a CDMA system that spreads each data symbol over the different tones. This system is usually referred to as the *MC-CDMA system*, introduced in 1993 [YLF93, FP93]. However, we will not consider this system in this work. □

## 1.8 Thesis Survey and Contributions

In this section, we give a general overview of the thesis. The structure of the thesis is depicted in **figure 1.15**.

In *chapter 2*, we will give a mathematical description of the spreading operation for the different CDMA systems discussed in this chapter. We also present some interesting data models describing the received information.

For a DS-CDMA and DMT-CDMA system, multi-user receivers achieve high performances. However, they are rather complex. In **part I**, we will therefore look for alternatives to multi-user receivers, aiming at a lower complexity and a comparable performance. For a single receive antenna and an ICI-limited quasi-synchronous propagation model, new receivers we develop for a DS-CDMA-BS and DMT-CDMA-BS system constitute such alternatives. In order to make a fair comparison, existing receivers for a DS-CDMA and DMT-CDMA system are also introduced for a single receive antenna and an ICI-limited quasi-synchronous propagation model.

In *chapter 3*, we will review some existing receivers for a DS-CDMA system, such as the RAKE receiver and the linear multi-user equalizer. Note that these receivers are well-known in the wireless communication community.

In *chapter 4*, we will develop new receivers for a DS-CDMA-BS system, such as the block RAKE receiver and the MUI-free receiver, which completely removes the MUI, without using any channel information. Note that these receivers are own contributions.

The complexity and performance of the MUI-free receiver and the linear multi-user equalizer will then be compared in *chapter 5*. In the same chapter, we will also compare the MUI-free receiver with the corresponding FFT-based Vandermonde-Lagrange AMOUR (VL-AMOUR) transceiver [WSGB99].

In *chapter 6*, we will present different receivers for a DMT-CDMA and DMT-CDMA-BS system. For a DMT-CDMA system, we will review some existing receivers, such as the RAKE receiver and the linear multi-user equalizer. Each of these receivers basically is a cascade of the corresponding receiver for a DS-CDMA system and a DMT demodulator. For a DMT-CDMA-BS system, we will develop new receivers, such as the block RAKE receiver and the MUI-free receiver, which completely removes the MUI, without using any channel information. Again, each of these receivers basically is a cascade of the corresponding receiver for a DS-CDMA-BS system and a DMT demodulator. However, the main contribution of this chapter is the development of the MUI/ITI-free receiver, which completely removes the MUI and ITI, without using any channel information. To conclude, we will compare the complexity and performance of the MUI-free receiver and the MUI/ITI-free receiver with the complexity and

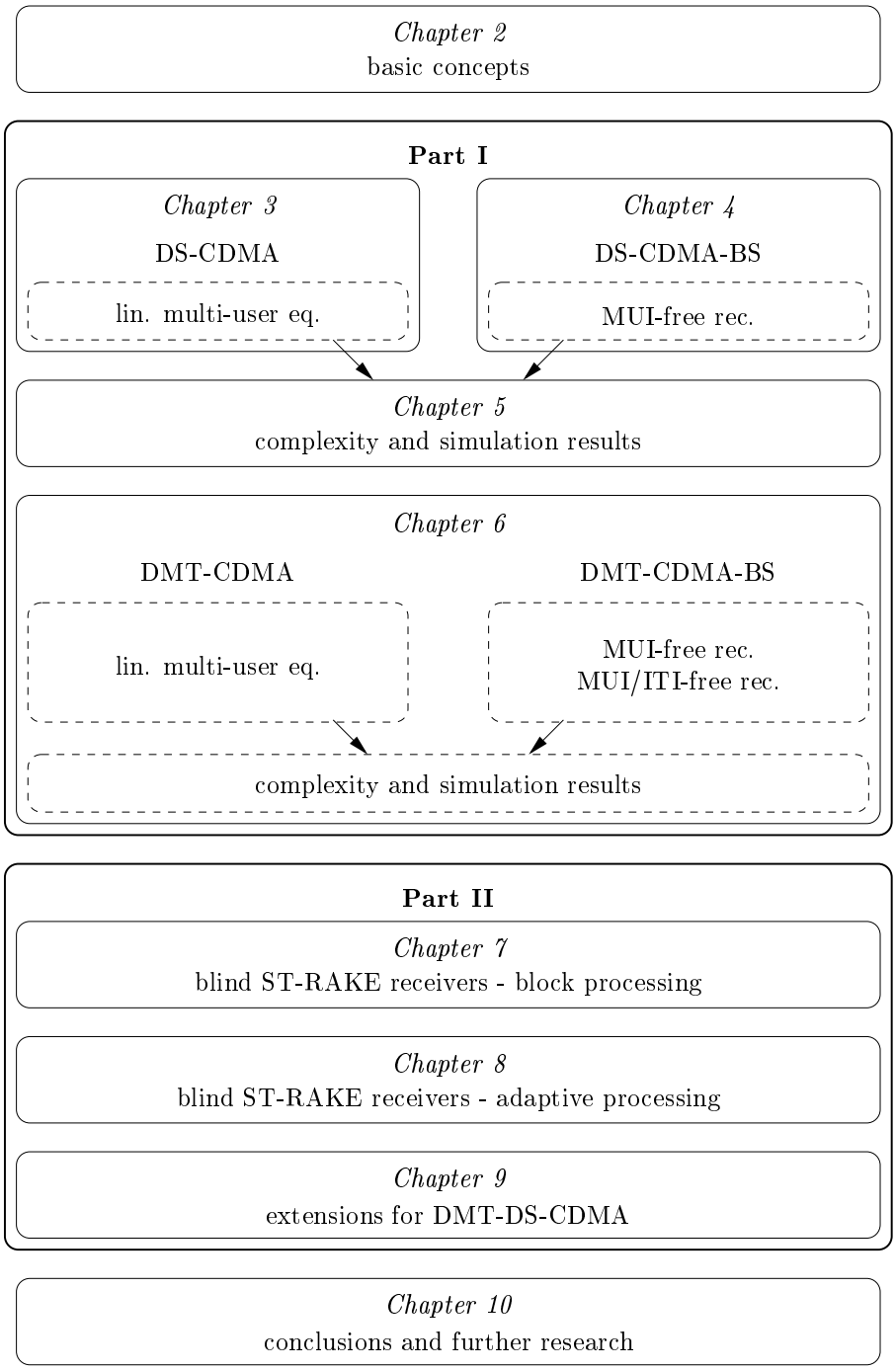



Figure 1.15: Structure of the thesis.

performance of the linear multi-user equalizer.

The publications related to part I are

- G. Leus and M. Moonen, *MUI-Free Receiver for a Synchronous DS-CDMA System Based on Block Spreading in the Presence of Frequency-Selective Fading*, IEEE Transactions on Signal Processing, accepted for publication.
- G. Leus and M. Moonen, *MUI-Free Receiver for a Shift-Orthogonal Quasi-Synchronous DS-CDMA System Based on Block Spreading in Frequency-Selective Fading*, in Proc. of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Istanbul, Turkey, June 2000, accepted for publication.
- G. Leus and M. Moonen, *Block Spreading for Discrete Multi-Tone CDMA Systems in the Presence of Frequency-Selective Fading*, in Proc. of the Asilomar Conference on Signals, Systems and Computers, Pacific Grove, California, October 1999.
- G. Leus and M. Moonen, *Fully Decorrelating Blind Equalization for Synchronous DS-CDMA Systems*, in Proc. of the IEEE Digital Signal Processing Workshop, Bryce Canyon, Utah, August 1998.

In **part II**, deterministic blind space-time RAKE (ST-RAKE) receivers for a DS-CDMA system with multiple receive antennas are developed. In a block processing context, these require much smaller blocks of received samples than stochastic blind ST-RAKE receivers. As a result, adaptive implementations can be used in a highly time-varying environment.

In chapter 7, we discuss deterministic blind ST-RAKE receivers based on *block processing*. Direct blind symbol estimation as well as direct blind equalizer estimation will be focused on. The proposed block processing methods based on direct blind symbol estimation have to the best of our knowledge not yet been studied in the literature. Some of the proposed block processing methods based on direct blind equalizer estimation, on the other hand, have already been studied in the literature, but only in their most simplest form.

In chapter 8, we then present some deterministic blind ST-RAKE receivers based on *adaptive processing*. We make a distinction between *windowed* and *exponentially-weighted* adaptive processing. Focus here will only be on direct blind symbol estimation. The proposed adaptive algorithms are all new contributions.

Note that in both chapters, we present *subspace methods* as well as *non-subspace methods*. Interesting complexity and performance comparisons of the proposed methods are given.

Finally, in *chapter 9*, we show how the deterministic blind ST-RAKE receivers based on block processing, presented in chapter 7, can also be applied on a DMT-DS-CDMA system (similar reasonings are possible for the deterministic blind ST-RAKE receivers based on adaptive processing, presented in chapter 8). We examine how the complexity and performance of the block processing methods, presented in chapter 7, behave when they are carried out for a tone of a DMT-DS-CDMA system w.r.t. the case they are carried out for the related DS-CDMA system.

Publications related to part II are

- G. Leus and M. Moonen, *Viterbi and RLS Decoding for Deterministic Blind Symbol Estimation in DS-CDMA Wireless Communication*, Signal Processing, Vol. 8, No. 5, pages 745–771, May 2000.
- G. Leus, P. Vandaele and M. Moonen, *Deterministic Blind Modulation-Induced Source Separation for Digital Wireless Communications*, IEEE Transactions on Signal Processing, submitted for publication.
- G. Leus, P. Vandaele and M. Moonen, *Per Tone Blind Signal Separation for a DMT-DS-CDMA System*, in Proc. of the European Signal Processing Conference (EUSIPCO), Tampere, Finland, September 2000, accepted for publication.
- P. Vandaele, G. Leus and M. Moonen, *A Non-Iterative Blind Signal Separation Algorithm Based on Transmit Diversity and Coding*, in Proc. of the Asilomar Conference on Signals, Systems and Computers, Pacific Grove, California, October 1999.
- P. Vandaele, G. Leus and M. Moonen, *A Non-Iterative Blind Binary Signal Separation Algorithm Based on Linear Coding*, in Proc. of the IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Annapolis, Maryland, May 1999, pages 98–101.
- G. Leus and M. Moonen, *Adaptive Blind Equalization for Asynchronous DS-CDMA Systems Based on RLS*, in Proc. of the European Signal Processing Conference (EUSIPCO), Rhodes, Greece, September 1998, pages 765–768.
- G. Leus and M. Moonen, *An Adaptive Blind Receiver for Asynchronous DS-CDMA Based on Recursive SVD and RLS*, in Proc. of the ProRISC/IEEE Workshop on Circuits, Systems and Signal Processing, Mierlo, The Netherlands, November 1997, pages 351–358.
- G. Leus and M. Moonen, *An Adaptive Blind Receiver for Asynchronous DS-CDMA Based on Recursive SVD and Viterbi Decoding*, in Proc. of the IEEE Benelux Symposium on Communications and Vehicular Technology, Enschede, The Netherlands, October 1997, pages 40–47.

- G. Leus and M. Moonen, *Adaptive Blind Equalization for Synchronous DS-CDMA Systems*, in Proc. of the COST 254 Workshop on Emerging Techniques for Communication Terminals, Toulouse, France, July 1997, pages 373–377.

In *chapter 10*, we finally summarize the main conclusions and give some suggestions for further research.

# Chapter 2

# Basic Concepts

In this chapter, we will explain some basic concepts that will be used throughout this thesis. In section 2.1, we give a mathematical description of the spreading operation for the different CDMA systems discussed in the previous chapter. First, we consider the well-known DS-CDMA system [Vit95]. Then we focus on the DS-CDMA system based on block spreading (DS-CDMA-BS system), presented in [CT97]. Next to these systems, we also consider some DMT-based CDMA systems, such as the DMT-CDMA system, presented in [Van95], and the new DMT-CDMA system based on block spreading (DMT-CDMA-BS system). A final DMT-based CDMA system we look at is the DMT-DS-CDMA system, presented in [DS94]. In section 2.2, we then discuss some data models describing the received information. The data models we present for a DS-CDMA or DMT-CDMA system employing short code sequences have already been considered in the literature. However, the data models we present for a DS-CDMA-BS or DMT-CDMA-BS system are new. Conclusions are given in section 2.3.

## 2.1 CDMA Systems

In a digital communication system, a bit stream is usually mapped onto a so-called *data symbol sequence*, with data symbols belonging to some finite alphabet $\Omega$ (each data symbol represents a fixed number of bits). Focusing for instance on phase shift keying (PSK), $\theta$-PSK modulation will map a block of $\log_2(\theta)$ bits onto one of $\theta$ data symbols, which have the same modulus and equidistant phases. Binary phase shift keying (BPSK) corresponds to $\theta = 2$, while quadrature phase shift keying (QPSK) corresponds to $\theta = 4$ (see **figure 2.1**).

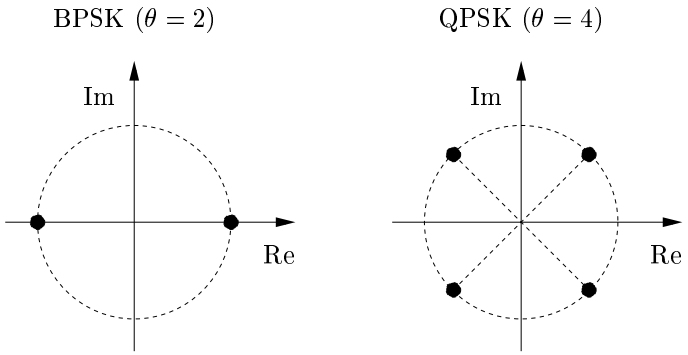



Figure 2.1: PSK constellation diagrams.

In a CDMA system, the $j$th user ($j = 1, 2, \ldots, J$) spreads his data symbol sequence $s_j[k]$ with rate $1/T$ (the *data symbol rate*) into a so-called chip sequence $x_j[n]$ with rate $N/T$ (the *chip rate*), using specific code information. Note that $N \geq 1$ is called the *spreading factor*. All users then transmit their chip sequence within the same frequency and time channel. In this section, we give a mathematical description of the above spreading operation for the different CDMA systems discussed in the previous chapter. First, we consider the well-known DS-CDMA system [Vit95] and the DS-CDMA-BS system [CT97]. Then, we look at some DMT-based CDMA systems, such as the DMT-CDMA system [Van95], the new DMT-CDMA-BS system and, finally, the DMT-DS-CDMA system [DS94].

### 2.1.1 DS-CDMA System

In a DS-CDMA system [Vit95], the data symbol sequence $s_j[k]$ is spread by a factor $N$ with the length-$\rho N$ ($\rho \geq 1$) code sequence $c_j[n]$ ($c_j[n] \neq 0$, for $n = 0, 1, \ldots, \rho N - 1$, and $c_j[n] = 0$, for $n < 0$ and $n \geq \rho N$), resulting into the desired chip sequence $x_j[n]$:

$$x_j[n] = s_j[k] c_j[n \bmod \rho N], \quad \text{with } k = \left\lfloor \frac{n}{N} \right\rfloor. \tag{2.1}$$

This spreading operation is depicted in **figure 2.2**.

**Remark 2.1.1** If $\rho = 1$, successive data symbols are spread in the same way (the spreading operation is periodic from data symbol to data symbol). From remark 1.3.1, we know that such code sequences are called *short* or *periodic*. If $\rho > 1$, successive data symbols are spread in a different way (the spreading operation is aperiodic from data symbol to data symbol). From remark 1.3.1, we know that such code sequences are called *long* or *aperiodic*. □

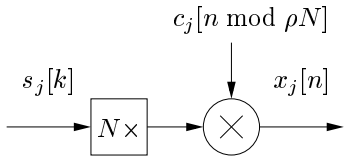



Figure 2.2: Spreading operation used in a DS-CDMA system.

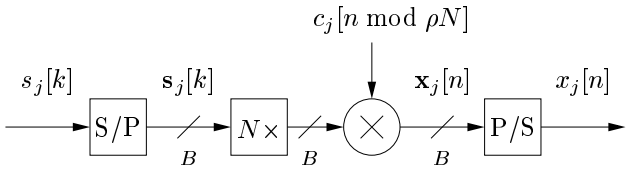



Figure 2.3: Spreading operation used in a DS-CDMA-BS system.

### 2.1.2 DS-CDMA-BS System

In a DS-CDMA-BS system [CT97], the data symbol sequence $s_j[k]$ is first blocked (block size $B$), leading to the following data symbol block sequence:

$$\mathbf{s}_j[k] = \Big[ s_j[kB] \quad s_j[kB+1] \quad \cdots \quad s_j[(k+1)B-1] \Big]^T .$$

This data symbol block sequence $\mathbf{s}_j[k]$ is then spread by a factor $N$ with the length-$\rho N$ ($\rho \geq 1$) code sequence $c_j[n]$ ($c_j[n] \neq 0$, for $n = 0, 1, \ldots, \rho N - 1$, and $c_j[n] = 0$, for $n < 0$ and $n \geq \rho N$), resulting into the chip block sequence $\mathbf{x}_j[n]$, given by

$$\mathbf{x}_j[n] = \mathbf{s}_j[k] c_j[n \bmod \rho N], \quad \text{with } k = \left\lfloor \frac{n}{N} \right\rfloor .$$

The desired chip sequence $x_j[n]$ is then obtained by deblocking $\mathbf{x}_j[n]$ (block size $B$):

$$\mathbf{x}_j[n] = \Big[ x_j[nB] \quad x_j[nB+1] \quad \cdots \quad x_j[(n+1)B-1] \Big]^T .$$

This spreading operation is depicted in **figure 2.3**. Note that remark 2.1.1 still holds (replace data symbol by data symbol block).

### 2.1.3 DMT-Based CDMA Systems

DMT-based CDMA systems aim at combining the advantages of DMT modulation (see section 1.7) with CDMA. In a DMT-based CDMA system, the

data symbol sequence $s_j[k]$ is first blocked (block size $P$), leading to the DMT symbol sequence $\tilde{\mathbf{r}}_j^{dmt}[\kappa]$ (DMT symbol size $P$):

$$\tilde{\mathbf{r}}_j^{dmt}[\kappa] = \begin{bmatrix} s_j[\kappa P] & s_j[\kappa P+1] & \cdots & s_j[(\kappa+1)P-1] \end{bmatrix}^T.$$

Defining the DMT subsymbol sequence for the $p$th tone ($p = 1, 2, \ldots, P$) as

$$\tilde{r}_{j,p}[\kappa] = s_j[\kappa P + p - 1],$$

this DMT symbol sequence $\tilde{\mathbf{r}}_j^{dmt}[\kappa]$ can also be written as

$$\tilde{\mathbf{r}}_j^{dmt}[\kappa] = \begin{bmatrix} \tilde{r}_{j,1}[\kappa] & \tilde{r}_{j,2}[\kappa] & \cdots & \tilde{r}_{j,P}[\kappa] \end{bmatrix}^T.$$

### DMT-CDMA System and DMT-CDMA-BS System

A first way to proceed, leading to a DMT-CDMA system or a DMT-CDMA-BS system, is to perform an inverse discrete Fourier transformation (IDFT) on the DMT symbol sequence $\tilde{\mathbf{r}}_j^{dmt}[\kappa]$:

$$\mathbf{r}_j^{dmt}[\kappa] = \sqrt{P}\mathcal{I}_P \tilde{\mathbf{r}}_j^{dmt}[\kappa],$$

where $\mathcal{I}_P$ represents the $P \times P$ IDFT matrix. Note that we do not add a cyclic prefix here. Deblocking $\mathbf{r}_j^{dmt}[\kappa]$ (block size $P$) results into the DMT-modulated data symbol sequence $r_j[k]$:

$$\mathbf{r}_j^{dmt}[\kappa] = \begin{bmatrix} r_j[\kappa P] & r_j[\kappa P+1] & \cdots & r_j[(\kappa+1)P-1] \end{bmatrix}^T.$$

*DMT-CDMA System:*

In a DMT-CDMA system [Van95], we then perform the spreading operation used in a DS-CDMA system (see figure 2.2) on the DMT-modulated data symbol sequence. The DMT-modulated data symbol sequence $r_j[k]$ is spread by a factor $N$ with the length-$\rho N$ ($\rho \geq 1$) code sequence $c_j[n]$ ($c_j[n] \neq 0$, for $n = 0, 1, \ldots, \rho N - 1$, and $c_j[n] = 0$, for $n < 0$ and $n \geq \rho N$), resulting into the desired chip sequence $x_j[n]$:

$$x_j[n] = r_j[k] c_j[n \bmod \rho N], \quad \text{with } k = \left\lfloor \frac{n}{N} \right\rfloor.$$

The spreading operation corresponding to the DMT-CDMA system is illustrated in **figure 2.4**.

*DMT-CDMA-BS System:*

In a DMT-CDMA-BS system, we then perform the spreading operation used in a DS-CDMA-BS system (see figure 2.3) on the DMT-modulated data symbol

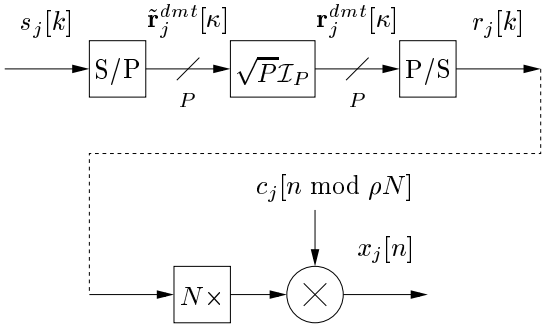



Figure 2.4: Spreading operation used in a DMT-CDMA system.

sequence. The DMT-modulated data symbol sequence $r_j[k]$ is first blocked (block size $B$), leading to the following DMT-modulated data symbol block sequence:

$$\mathbf{r}_j[k] = \begin{bmatrix} r_j[kB] & r_j[kB+1] & \cdots & r_j[(k+1)B-1] \end{bmatrix}^T.$$

This DMT-modulated data symbol block sequence $\mathbf{r}_j[k]$ is then spread by a factor $N$ with the length-$\rho N$ ($\rho \geq 1$) code sequence $c_j[n]$ ($c_j[n] \neq 0$, for $n = 0, 1, \ldots, \rho N - 1$, and $c_j[n] = 0$, for $n < 0$ and $n \geq \rho N$), resulting into the chip block sequence $\mathbf{x}_j[n]$, given by

$$\mathbf{x}_j[n] = \mathbf{r}_j[k] c_j[n \bmod \rho N], \quad \text{with } k = \left\lfloor \frac{n}{N} \right\rfloor.$$

The desired chip sequence $x_j[n]$ is then obtained by deblocking $\mathbf{x}_j[n]$ (block size $B$):

$$\mathbf{x}_j[n] = \begin{bmatrix} x_j[nB] & x_j[nB+1] & \cdots & x_j[(n+1)B-1] \end{bmatrix}^T.$$

The spreading operation corresponding to the DMT-CDMA-BS system is illustrated in **figure 2.5**.

### DMT-DS-CDMA System

A second way to proceed, leading to a DMT-DS-CDMA system [DS94], is to perform the spreading operation used in a DS-CDMA system (see figure 2.2) on each of the DMT subsymbol sequences. The DMT subsymbol sequence $\tilde{r}_{j,p}[\kappa]$ is spread by a factor $N$ with the length-$\rho N$ ($\rho \geq 1$) code sequence $c_{j,p}[\nu]$ ($c_{j,p}[\nu] \neq 0$, for $\nu = 0, 1, \ldots, \rho N - 1$, and $c_{j,p}[\nu] = 0$, for $\nu < 0$ and $\nu \geq \rho N$), resulting into

$$\tilde{x}_{j,p}[\nu] = \tilde{r}_{j,p}[\kappa] c_{j,p}[\nu \bmod \rho N], \text{ with } \kappa = \lfloor \nu / N \rfloor. \tag{2.2}$$

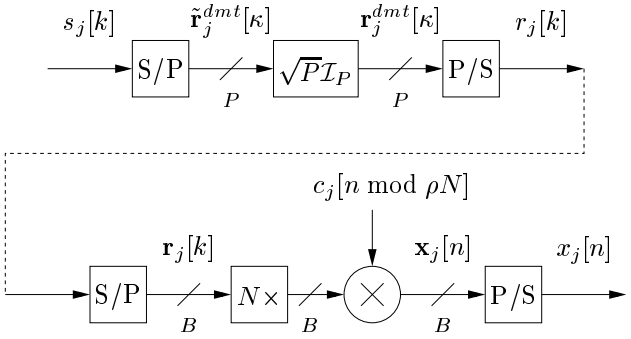



Figure 2.5: Spreading operation used in a DMT-CDMA-BS system.

Combining the results for all $P$ tones, we obtain

$$\tilde{\mathbf{x}}_j^{dmt}[\nu] = \begin{bmatrix} \tilde{x}_{j,1}[\nu] & \tilde{x}_{j,2}[\nu] & \cdots & \tilde{x}_{j,P}[\nu] \end{bmatrix}^T.$$

We then perform an IDFT and add a cyclic prefix of length $\mu$:

$$\mathbf{x}_j^{dmt}[\nu] = \mathbf{P}_t \sqrt{P}\mathcal{I}_P \tilde{\mathbf{x}}_j^{dmt}[\nu],$$

where $\mathbf{P}_t$ is the $(P+\mu) \times P$ matrix, given by

$$\mathbf{P}_t = \left[\begin{array}{c|c} \mathbf{O} & \mathbf{I}_\mu \\ \hline \multicolumn{2}{c}{\mathbf{I}_P} \end{array}\right].$$

The desired chip sequence $x_j[n]$ is then obtained by deblocking $\mathbf{x}_j^{dmt}[\nu]$ (block size $P+\mu$):

$$\mathbf{x}_j^{dmt}[\nu] = \begin{bmatrix} x_j[\nu(P+\mu)] & x_j[\nu(P+\mu)+1] & \cdots & x_j[(\nu+1)(P+\mu)-1] \end{bmatrix}^T.$$

The spreading operation corresponding to the DMT-DS-CDMA system is illustrated in **figure 2.6**. Note that because we use a cyclic prefix of length $\mu$ the actual spreading factor now is $N(P+\mu)/P$. However, since the length of the cyclic prefix $\mu$ is generally much smaller than the DMT symbol size $P$, this actual spreading factor can be very well approximated by $N$.

### 2.1.4 Short Code Sequences

Here, we introduce some useful notations for a short code sequence. These notations will be used in part I. For a short code sequence $c_j[n]$ ($\rho = 1$), we define the $N \times 1$ code vector $\mathbf{c}_j$ as

$$\mathbf{c}_j = \begin{bmatrix} c_j[0] & c_j[1] & \cdots & c_j[N-1] \end{bmatrix}^T.$$

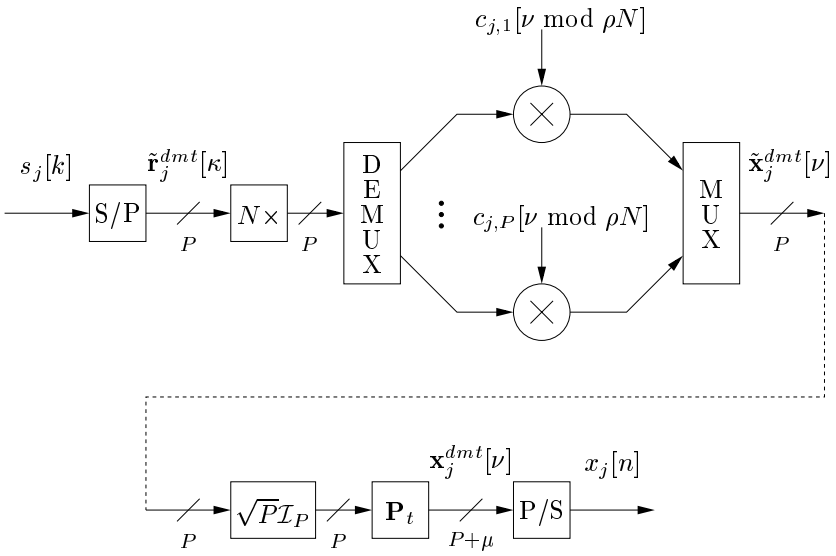



Figure 2.6: Spreading operation used in a DMT-DS-CDMA system.

We further define the $(N-n)\times 1$ code vector $\mathbf{c}_j^{\uparrow(n)}$ by removing the upper $n$ entries of $\mathbf{c}_j$:

$$\mathbf{c}_j^{\uparrow(n)} = \Big[c_j[n] \quad c_j[n+1] \quad \cdots \quad c_j[N-1]\Big]^T \tag{2.3}$$

and the $(N-n)\times 1$ code vector $\mathbf{c}_j^{\downarrow(n)}$ by removing the lower $n$ entries of $\mathbf{c}_j$:

$$\mathbf{c}_j^{\downarrow(n)} = \Big[c_j[0] \quad c_j[1] \quad \cdots \quad c_j[N-1-n]\Big]^T. \tag{2.4}$$

## 2.2 Data Models

In the previous section, we have described the spreading operation for various CDMA systems. In this section, we present some data models describing the received information. This is done for a single receive antenna as well as for multiple receive antennas. We do not give an actual description of the data models but only aim at an equivalent baseband description, which is common practice. The data models we present for a DS-CDMA or DMT-CDMA system employing short code sequences are well-know. The data models we present for a DS-CDMA-BS or DMT-CDMA-BS system, on the other hand, are new. The goal of developing these data models is to reveal some interesting features that can be used for the receiver design.

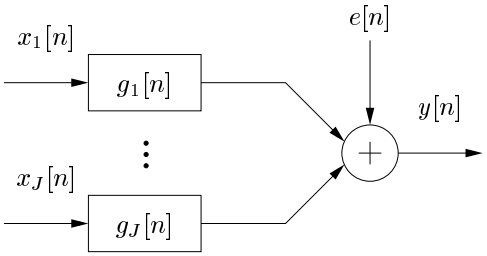



Figure 2.7: Single receive antenna data model.

Note that we assume that the multi-path propagation environment is slowly time varying (slowly *fading*), such that the *coherence time* (the time over which the multi-path propagation environment is essentially time invariant) is much larger than the data symbol period $T$. Hence, we can model the multi-path propagation environment as time invariant within the observation window.

### 2.2.1 Single Receive Antenna

Assume that our system is equipped with 1 receive antenna and let $g_j(t)$ denote the continuous-time channel from the $j$th user to the receive antenna, including the transmit and receive filters. The received signal can then be written as

$$y(t) = \sum_{j=1}^{J} \sum_{n=-\infty}^{+\infty} g_j(t - nT/N)x_j[n] + e(t),$$

where $e(t)$ is the continuous-time additive noise. If we sample at the chip rate $N/T$, we then obtain the following received sequence

$$y[n] = y(nT/N) = \sum_{j=1}^{J} \sum_{n'=-\infty}^{+\infty} g_j[n']x_j[n - n'] + e[n], \tag{2.5}$$

where $e[n] = e(nT/N)$ is the discrete-time additive noise and $g_j[n] = g_j(nT/N)$ is the discrete-time channel from the $j$th user to the receive antenna. We model $g_j[n]$ as a finite impulse response (FIR) filter of order $L_j$ with delay index $\delta_j$ ($g_j[n] \neq 0$, for $n = \delta_j$ and $n = \delta_j + L_j$, and $g_j[n] = 0$, for $n < \delta_j$ and $n > \delta_j + L_j$). Note that the larger the order of $g_j[n]$, the more inter-chip interference (ICI) for the $j$th user. The above described data model (a single-channel model: 1 channel per user) is depicted in **figure 2.7**.

*DS-CDMA or DMT-CDMA System Employing Short Code Sequences:*

Assume we have a DS-CDMA or DMT-CDMA system employing short code sequences. We can then define the discrete-time *composite* channel from the

$j$th user to the receive antenna as

$$h_j[n] = \sum_{n'=-\infty}^{+\infty} g_j[n']c_j[n-n']. \tag{2.6}$$

Since $g_j[n]$ is an FIR filter of order $L_j$ with delay index $\delta_j$, $h_j[n]$ is an FIR filter of order $L_j + N - 1$ with delay index $\delta_j$ ($h_j[n] \neq 0$, for $n = \delta_j$ and $n = \delta_j + L_j + N - 1$, and $h_j[n] = 0$, for $n < \delta_j$ and $n > \delta_j + L_j + N - 1$). If we then introduce the following received vector sequence:

$$\mathbf{y}^{vec}[k] = \Big[y[kN] \quad y[kN+1] \quad \cdots \quad y[(k+1)N-1]\Big]^T,$$

we can write

$$\mathbf{y}^{vec}[k] = \sum_{j=1}^{J} \sum_{k'=-\infty}^{+\infty} \mathbf{h}_j^{vec}[k']s_j[k-k'] + \mathbf{e}^{vec}[k] \quad \text{(for DS-CDMA)}, \tag{2.7}$$

$$\mathbf{y}^{vec}[k] = \sum_{j=1}^{J} \sum_{k'=-\infty}^{+\infty} \mathbf{h}_j^{vec}[k']r_j[k-k'] + \mathbf{e}^{vec}[k] \quad \text{(for DMT-CDMA)}, \tag{2.8}$$

where $\mathbf{e}^{vec}[k]$ is similarly defined as $\mathbf{y}^{vec}[k]$ and $\mathbf{h}_j^{vec}[k]$ is the discrete-time $N \times 1$ composite vector channel from the $j$th user to the receive antenna, given by

$$\mathbf{h}_j^{vec}[k] = \Big[h_j[kN] \quad h_j[kN+1] \quad \cdots \quad h_j[(k+1)N-1]\Big]^T.$$

Since $h_j[n]$ is an FIR filter of order $L_j + N - 1$ with delay index $\delta_j$, $\mathbf{h}_j^{vec}[k]$ is an $N \times 1$ FIR vector filter of order $\lfloor(\delta_j + L_j + N - 1)/N\rfloor - \lfloor\delta_j/N\rfloor$ with delay index $\lfloor\delta_j/N\rfloor$ ($\mathbf{h}_j^{vec}[k] \neq \mathbf{0}$, for $k = \lfloor\delta_j/N\rfloor$ and $k = \lfloor(\delta_j + L_j + N - 1)/N\rfloor$, and $\mathbf{h}_j^{vec}[k] = \mathbf{0}$, for $k < \lfloor\delta_j/N\rfloor$ and $k > \lfloor(\delta_j + L_j + N - 1)/N\rfloor$). Note that the larger the order of $\mathbf{h}_j^{vec}[k]$, the more inter-symbol interference (ISI) for the $j$th user. The ISI we are talking about here is the ISI w.r.t. the data symbol sequence $s_j[k]$ (for DS-CDMA) or w.r.t. the DMT-modulated data symbol sequence $r_j[k]$ (for DMT-CDMA). The above described data model (a multi-channel model with $N$ channels per user) is depicted in **figure 2.8**.

*DS-CDMA-BS or DMT-CDMA-BS System:*

Assume we have a DS-CDMA-BS or DMT-CDMA-BS system. If we then block the received sequence $y[n]$ (block size $B$), leading to the following received block sequence:

$$\mathbf{y}[n] = \Big[y[nB] \quad y[nB+1] \quad \cdots \quad y[(n+1)B-1]\Big]^T,$$

we can write

$$\mathbf{y}[n] = \sum_{j=1}^{J} \sum_{n'=-\infty}^{\infty} \mathbf{G}_j[n']\mathbf{x}_j[n-n'] + \mathbf{e}[n], \tag{2.9}$$

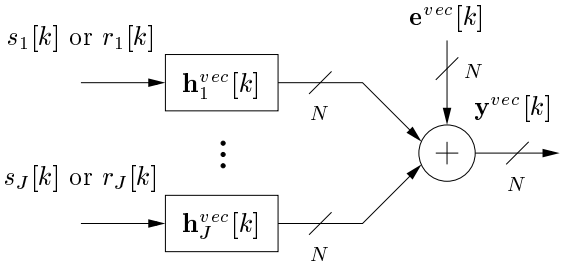



Figure 2.8: Single receive antenna data model for a DS-CDMA or DMT-CDMA system employing short code sequences.

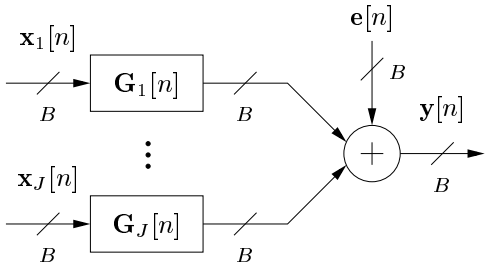



Figure 2.9: Single receive antenna data model for a DS-CDMA-BS or DMT-CDMA-BS system.

where $\mathbf{e}[n]$ is similarly defined as $\mathbf{y}[n]$ and $\mathbf{G}_j[n]$ is the discrete-time $B \times B$ matrix channel from the $j$th user to the receive antenna, given by

$$\mathbf{G}_j[n] = \begin{bmatrix} g_j[Bn] & g_j[Bn-1] & \cdots & g_j[B(n-1)+1] \\ g_j[Bn+1] & g_j[Bn] & \cdots & g_j[B(n-1)+2] \\ \vdots & \vdots & \ddots & \vdots \\ g_j[B(n+1)-1] & g_j[B(n+1)-2] & \cdots & g_j[Bn] \end{bmatrix}.$$

Since $g_j[n]$ is an FIR filter of order $L_j$ with delay index $\delta_j$, $\mathbf{G}_j[n]$ is a $B \times B$ FIR matrix filter of order $\lceil (\delta_j + L_j)/B \rceil - \lfloor \delta_j/B \rfloor$ with delay index $\lfloor \delta_j/B \rfloor$ ($\mathbf{G}_j[n] \neq \mathbf{O}_B$, for $n = \lfloor \delta_j/B \rfloor$ and $n = \lceil (\delta_j + L_j)/B \rceil$, and $\mathbf{G}_j[n] = \mathbf{O}_B$, for $n < \lfloor \delta_j/B \rfloor$ and $n > \lceil (\delta_j + L_j)/B \rceil$). Note that the larger the order of $\mathbf{G}_j[n]$, the more inter-chip-block interference (ICBI) for the $j$th user. The above described data model (a single-matrix-channel model: 1 matrix channel per user) is depicted in **figure 2.9**.

### ICI-Limited Quasi-Synchronous Propagation Model

Denoting the maximal channel order as

$$L_{max} = \max_j \{L_j\},$$

we define an *ICI-limited quasi-synchronous propagation model* as a propagation model, for which we can find a parameter $L$, such that

$$0 \leq \delta_j < \delta_j + L_j < L \ll N, \quad \text{for } j = 1, 2, \ldots, J. \tag{2.10}$$

Note that we always assume there is ICI present. Then, $g_j[n]$ can be modeled by an FIR filter of order $L-1$ with delay index 0 (overmodeling can occur) and the data model (2.5) can be written as

$$y[n] = \sum_{j=1}^{J} \sum_{n'=0}^{L-1} g_j[n'] x_j[n-n'] + e[n]. \tag{2.11}$$

We then also define the $L \times 1$ channel vector for the $j$th user as

$$\mathbf{g}_j = \begin{bmatrix} g_j[0] & g_j[1] & \cdots & g_j[L-1] \end{bmatrix}^T. \tag{2.12}$$

*DS-CDMA or DMT-CDMA System Employing Short Code Sequences:*

When an ICI-limited quasi-synchronous propagation model is considered for a DS-CDMA or DMT-CDMA system employing short code sequences, $\mathbf{h}_j^{vec}[k]$ becomes an $N \times 1$ FIR vector filter of order 1 with delay index 0, where

$$\mathbf{h}_j^{vec}[0] = \begin{bmatrix} h_j[0] & h_j[1] & \cdots & h_j[N-1] \end{bmatrix}^T,$$

$$\mathbf{h}_j^{vec}[1] = \begin{bmatrix} h_j[N] & h_j[N+1] & \cdots & h_j[N+L-2] & 0 & \cdots & 0 \end{bmatrix}^T.$$

The data models (2.7) and (2.8) can then be written as

$$\mathbf{y}^{vec}[k] = \sum_{j=1}^{J} \sum_{k'=0}^{1} \mathbf{h}_j^{vec}[k'] s_j[k-k'] + \mathbf{e}^{vec}[k] \quad \text{(for DS-CDMA)},$$

$$\mathbf{y}^{vec}[k] = \sum_{j=1}^{J} \sum_{k'=0}^{1} \mathbf{h}_j^{vec}[k'] r_j[k-k'] + \mathbf{e}^{vec}[k] \quad \text{(for DMT-CDMA)}.$$

Removing the first $L-1$ elements of $\mathbf{y}^{vec}[k]$:

$$\mathbf{y}^{cut}[k] = \begin{bmatrix} y[kN+L-1] & y[kN+L] & \cdots & y[(k+1)N-1] \end{bmatrix}^T,$$

we can completely remove the ISI for every user [LX96b] and we obtain

$$\mathbf{y}^{cut}[k] = \sum_{j=1}^{J} \mathbf{h}_j^{cut} s_j[k] + \mathbf{e}^{cut}[k] \quad \text{(for DS-CDMA)},$$

$$\mathbf{y}^{cut}[k] = \sum_{j=1}^{J} \mathbf{h}_j^{cut} r_j[k] + \mathbf{e}^{cut}[k] \quad \text{(for DMT-CDMA)},$$

where $\mathbf{e}^{cut}[k]$ is similarly defined as $\mathbf{y}^{cut}[k]$ and $\mathbf{h}_j^{cut}$ is the $(N-L+1)\times 1$ composite channel vector for the $j$th user, given by

$$\mathbf{h}_j^{cut} = \begin{bmatrix} h_j[L-1] & h_j[L] & \cdots & h_j[N-1] \end{bmatrix}^T. \tag{2.13}$$

Note that $\mathbf{y}^{cut}[k]$ can also be written as

$$\mathbf{y}^{cut}[k] = \mathbf{H}\mathbf{s}[k] + \mathbf{e}^{cut}[k] \quad \text{(for DS-CDMA)}, \tag{2.14}$$

$$\mathbf{y}^{cut}[k] = \mathbf{H}\mathbf{r}[k] + \mathbf{e}^{cut}[k] \quad \text{(for DMT-CDMA)}, \tag{2.15}$$

where $\mathbf{H}$ is the $(N-L+1)\times J$ composite channel matrix, given by

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1^{cut} & \mathbf{h}_2^{cut} & \cdots & \mathbf{h}_J^{cut} \end{bmatrix},$$

and $\mathbf{s}[k]$ and $\mathbf{r}[k]$ are defined as

$$\mathbf{s}[k] = \begin{bmatrix} s_1[k] & s_2[k] & \cdots & s_J[k] \end{bmatrix}^T,$$

$$\mathbf{r}[k] = \begin{bmatrix} r_1[k] & r_2[k] & \cdots & r_J[k] \end{bmatrix}^T.$$

*DS-CDMA-BS or DMT-CDMA-BS System:*

When an ICI-limited quasi-synchronous propagation model is considered for a DS-CDMA-BS or DMT-CDMA-BS system, we usually take the block size $B$ equal to the parameter $L$. Then $\mathbf{G}_j[n]$ becomes an $L\times L$ FIR matrix filter of order 1 with delay index 0, where

$$\mathbf{G}_j[0] = \begin{bmatrix} g_j[0] & 0 & \cdots & 0 \\ g_j[1] & g_j[0] & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_j[L-1] & g_j[L-2] & \cdots & g_j[0] \end{bmatrix},$$

$$\mathbf{G}_j[1] = \begin{bmatrix} 0 & g_j[L-1] & \cdots & g_j[1] \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & g_j[L-1] \\ 0 & 0 & \cdots & 0 \end{bmatrix}.$$

The data model (2.9) can then be written as

$$\mathbf{y}[n] = \sum_{j=1}^{J} \sum_{n'=0}^{1} \mathbf{G}_j[n']\mathbf{x}_j[n-n'] + \mathbf{e}[n]. \tag{2.16}$$

### 2.2.2 Multiple Receive Antennas (Spatial Oversampling)

Assume that our system is equipped with $M$ receive antennas and let $g_j^{(m)}(t)$ denote the continuous-time channel from the $j$th user to the $m$th receive antenna, including the transmit and receive filters. The received signal at the $m$th receive antenna can then be written as

$$y^{(m)}(t) = \sum_{j=1}^{J} \sum_{n=-\infty}^{+\infty} g_j^{(m)}(t - nT/N)x_j[n] + e^{(m)}(t),$$

where $e^{(m)}(t)$ is the continuous-time additive noise at the $m$th receive antenna. If we sample at the chip rate $N/T$, we then obtain the following received sequence at the $m$th receive antenna

$$y^{(m)}[n] = y^{(m)}(nT/N) = \sum_{j=1}^{J} \sum_{n'=-\infty}^{+\infty} g_j^{(m)}[n-n']x_j[n'] + e^{(m)}[n],$$

where $e^{(m)}[n] = e^{(m)}(nT/N)$ is the discrete-time additive noise at the $m$th receive antenna and $g_j^{(m)}[n] = g_j^{(m)}(nT/N)$ is the discrete-time channel from the $j$th user to the $m$th receive antenna. Stacking the received samples obtained from the $M$ receive antennas:

$$\mathbf{y}^{st}[n] = \begin{bmatrix} y^{(1)}[n] & y^{(2)}[n] & \cdots & y^{(M)}[n] \end{bmatrix}^T,$$

we can write

$$\mathbf{y}^{st}[n] = \sum_{j=1}^{J} \sum_{n'=-\infty}^{+\infty} \mathbf{g}_j^{st}[n']x_j[n-n'] + \mathbf{e}^{st}[n], \tag{2.17}$$

where $\mathbf{e}^{st}[n]$ is similarly defined as $\mathbf{y}^{st}[n]$ and $\mathbf{g}_j^{st}[n]$ is the discrete-time $M \times 1$ vector channel from the $j$th user to the $M$ receive antennas, given by

$$\mathbf{g}_j^{st}[n] = \begin{bmatrix} g_j^{(1)}[n] & g_j^{(2)}[n] & \cdots & g_j^{(M)}[n] \end{bmatrix}^T. \tag{2.18}$$

We model $\mathbf{g}_j^{st}[n]$ as an $M \times 1$ FIR vector filter of order $L_j$ with delay index $\delta_j$ ($\mathbf{g}_j^{st}[n] \neq \mathbf{0}$, for $n = \delta_j$ and $n = \delta_j + L_j$, and $\mathbf{g}_j^{st}[n] = \mathbf{0}$, for $n < \delta_j$ and $n > \delta_j + L_j$). Note that the larger $L_j$, the more ICI for the $j$th user. The

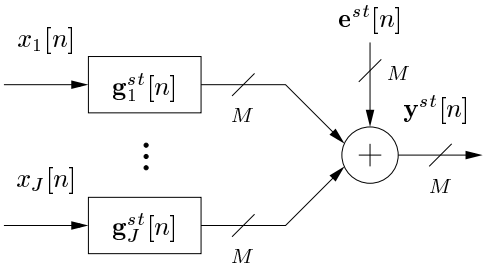



Figure 2.10: Multiple receive antenna data model.

above described data model (a multi-channel model with $M$ channels per user) is depicted in **figure 2.10**.

Note that for a DS-CDMA or DMT-CDMA system employing short code sequences and for a DS-CDMA-BS or DMT-CDMA-BS system we could proceed in a similar way as in section 2.2.1. Then, the data models described in (2.7) and (2.8) would become multi-channel models with $MN$ channels per user and the data model described in (2.9) would become a multi-matrix-channel model with $M$ matrix channels per user. However, since these models will not be used in this thesis, they will not be described.

### 2.2.3 Temporal Oversampling

Until now, we have always considered chip rate sampling at the receiver side. Since in practice, the transmit and receive filters satisfy the Nyquist criterion for a rate of $N/T$ and have a bandwidth that is somewhat higher than $N/T$ (usually root raised cosine filters are used [Pro89]), chip rate sampling allows us to capture all the transmitted information that is present in the received information if the multi-path propagation environment does not destroy the Nyquist character of the transmit and receive filters and we sample at the correct time instances. Since this situation is not very likely to occur, we should actually sample at a rate that is higher than the chip rate $N/T$. This is called *temporal oversampling* [Ung76]. Sampling at $S$ times the chip rate, the data model described in (2.5) would become a multi-channel model with $S$ channels per user, the data models described in (2.7) and (2.8) would become multi-channel models with $NS$ channels per user, the data model described in (2.9) would become a multi-matrix channel model with $S$ matrix channels per user and, finally, the data model described in (2.17) would become a multi-channel model with $MS$ channels per user. However, for simplicity reasons, we will always consider chip rate sampling at the receiver side.

## 2.3 Conclusions

In this chapter, we have explained some basic concepts that will be used throughout this thesis. First, we have described the spreading operation for various CDMA systems. Then, we have discussed some data models describing the received information.

# Part I

# Receivers for Block Spreading Systems: Alternatives to Multi-User Receivers

For a DS-CDMA and DMT-CDMA system, multi-user receivers achieve high performances. However, they are rather complex. In this part of the thesis, we therefore look for alternatives to multi-user receivers, aiming at a lower complexity and a comparable performance. For a single receive antenna and an ICI-limited quasi-synchronous propagation model, new receivers we develop for a DS-CDMA-BS and DMT-CDMA-BS system constitute such alternatives. In order to make a fair comparison, existing receivers for a DS-CDMA and DMT-CDMA system are also introduced for a single receive antenna and an ICI-limited quasi-synchronous propagation model.

In *chapter 3*, we review some existing receivers for a DS-CDMA system, such as the RAKE receiver and the linear multi-user equalizer.

In *chapter 4*, we then develop new receivers for a DS-CDMA-BS system, such as the block RAKE receiver and the MUI-free receiver, which completely removes the MUI, without using any channel information.

Next, we compare the complexity and performance of the MUI-free receiver with the complexity and performance of the linear multi-user equalizer in *chapter 5*. This chapter also contains a comparison of the MUI-free receiver with the corresponding FFT-based Vandermonde-Lagrange AMOUR (VL-AMOUR) transceiver [WSGB99].

In the last chapter of this part, namely *chapter 6*, we present different receivers for a DMT-CDMA and DMT-CDMA-BS system. For a DMT-CDMA system, we review some existing receivers, such as the RAKE receiver and the linear multi-user equalizer. Each of these receivers basically is a cascade of the corresponding receiver for a DS-CDMA system and a DMT demodulator. For a DMT-CDMA-BS system, we develop new receivers, such as the block RAKE receiver and the MUI-free receiver, which completely removes the MUI, without using any channel information. Again, each of these receivers basically is a cascade of the corresponding receiver for a DS-CDMA-BS system and a DMT demodulator. However, the main contribution of this chapter is the development of the MUI/ITI-free receiver, which completely removes the MUI and ITI, without using any channel information. To conclude, we compare the complexity and performance of the MUI-free receiver and the MUI/ITI-free receiver with the complexity and performance of the linear multi-user equalizer.

# Chapter 3

# Receivers for a DS-CDMA System

In this chapter, we review some existing receivers for a DS-CDMA system. Although the receivers we discuss can be developed for a more general scenario, we consider a single receive antenna and an ICI-limited quasi-synchronous propagation model. We further use short code sequences.

Details on the used data model are given in section 3.1. In section 3.2, the RAKE receiver is discussed. This receiver consists of a bank of correlators, followed by a linear combiner, which linearly combines the correlator outputs. In section 3.3, we then discuss the linear multi-user equalizer. Within the framework of the linear multi-user equalizer, we then present a subspace deterministic blind multi-user channel estimation algorithm in section 3.4. We also give a performance analysis of this algorithm and show how the channel gain can be estimated. Note that this blind channel estimation approach is presented in [LX96b], including the performance analysis and channel gain estimation. We end with some conclusions in section 3.5.

## 3.1 Data Model

We consider a single receive antenna and an ICI-limited quasi-synchronous propagation model (see section 2.2.1). We further use short code sequences (see section 2.1.4). The data models that will be used are given by (2.11)

and (2.14), which are repeated here for convenience:

$$y[n] = \sum_{j=1}^{J} \sum_{n'=0}^{L-1} g_j[n'] x_j[n-n'] + e[n], \tag{3.1}$$

$$\mathbf{y}^{cut}[k] = \mathbf{H}\mathbf{s}[k] + \mathbf{e}^{cut}[k]. \tag{3.2}$$

## 3.2 RAKE Receiver

The RAKE receiver (Price and Green [PG58]) consists of a bank of correlators, followed by a linear combiner, which linearly combines the correlator outputs.

### 3.2.1 Bank of Correlators

Since $g_j[n]$ can be modeled by an FIR filter of order $L-1$ with delay index 0 (overmodeling can occur), we apply a bank of $L$ correlators. The $(l+1)$th $(l = 0, 1, \ldots, L-1)$ correlator of the $j$th user despreads the received sequence $y[n]$ matched to $g_j[l]$, using the code sequence $c_j[n]$. The output of the $(l+1)$th correlator can be written as

$$\begin{aligned} z_{j,l}[k] &= \sum_{n=0}^{N-1} c_j^*[n] y[kN+n+l] \\ &= \mathbf{c}_j^H \mathbf{c}_j g_j[l] s_j[k] + p_{j,l}[k] + q_{j,l}[k] + n_{j,l}[k]. \end{aligned} \tag{3.3}$$

In these formulas $p_{j,l}[k]$ represents the *residual ICI*:

$$\begin{aligned} p_{j,l}[k] = \sum_{\substack{n=\delta_j \\ n \neq l}}^{\delta_j + L_j} & \mathbf{c}_j^{\downarrow(\nu)H} \mathbf{c}_j^{\uparrow(\nu)} g_j[n] s_j[k+\kappa] \\ & + \mathbf{c}_j^{\uparrow(N-\nu)H} \mathbf{c}_j^{\downarrow(N-\nu)} g_j[n] s_j[k+\kappa+1], \end{aligned}$$

with

$$\kappa = \left\lfloor \frac{l-n}{N} \right\rfloor \text{ and } \nu = l - n - \kappa N,$$

$q_{j,l}[k]$ represents the *residual MUI*:

$$q_{j,l}[k] = \sum_{\substack{j'=1 \\ j' \neq j}}^{J} \sum_{n=\delta_{j'}}^{\delta_{j'}+L_{j'}} \mathbf{c}_j^{\downarrow(\nu)H} \mathbf{c}_{j'}^{\uparrow(\nu)} g_{j'}[n] s_{j'}[k+\kappa]$$

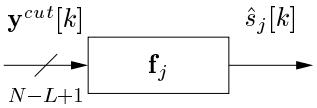



Figure 3.1: Linear multi-user equalizer for a DS-CDMA system.

$$+ \mathbf{c}_j^{\uparrow(N-\nu)H} \mathbf{c}_{j'}^{\downarrow(N-\nu)} g_{j'}[n] s_{j'}[k+\kappa+1],$$

with

$$\kappa = \left\lfloor \frac{l-n}{N} \right\rfloor \text{ and } \nu = l - n - \kappa N,$$

and $n_{j,l}[k]$ represents the *residual additive noise*:

$$n_{j,l}[k] = \sum_{n=0}^{N-1} c_j^*[n] e[kN+n+l].$$

### 3.2.2 Linear Combiner

We then linearly combine the $L$ correlator outputs $\{z_{j,l}[k]\}_{l=0}^{L-1}$ to suppress the remaining residual ICI and MUI:

$$\hat{s}_j[k] = f_{j,0} z_{j,0}[k] + f_{j,1} z_{j,1}[k] + \cdots + f_{j,L-1} z_{j,L-1}[k],$$

where $\{f_{j,l}\}_{l=0}^{L-1}$ are the linear combiner weights. We can for instance use the MF linear combiner, resulting in the coherent RAKE receiver [Vit95]. We then take

$$f_{j,l} = \frac{g_j^*[l]}{\|\mathbf{g}_j\| \|\mathbf{c}_j\|}, \quad \text{for } l = 0, 1, \ldots, L-1.$$

## 3.3 Linear Multi-User Equalizer

We perform a linear multi-user equalization on $\mathbf{y}^{cut}[k]$ to suppress the remaining MUI (the ISI is removed by construction):

$$\hat{s}_j[k] = \mathbf{f}_j \mathbf{y}^{cut}[k] = \mathbf{f}_j \mathbf{H} \mathbf{s}[k] + \mathbf{f}_j \mathbf{e}^{cut}[k],$$

where $\mathbf{f}_j$ is the $1 \times (N-L+1)$ linear multi-user equalizer for the $j$th user. The linear multi-user equalizer is illustrated in **figure 3.1**.

We now focus on the calculation of the signal-to-interference-plus-noise ratio (SINR) and bit error rate (BER) for the $j$th user at the output of the linear multi-user equalizer $\mathbf{f}_j$. First of all, observe that

$$\mathbf{R}_{\mathbf{y}^{cut}} = \mathrm{E}\{\mathbf{y}^{cut}[k]\mathbf{y}^{cutH}[k]\} = \mathbf{H}\mathbf{R}_{\mathbf{s}}\mathbf{H}^H + \mathbf{R}_{\mathbf{e}^{cut}}, \tag{3.4}$$

where $\mathbf{R}_{\mathbf{e}^{cut}}$ is similarly defined as $\mathbf{R}_{\mathbf{y}^{cut}}$ and

$$\mathbf{R}_{\mathbf{s}} = \mathrm{E}\{\mathbf{s}[k]\mathbf{s}^H[k]\}.$$

If we then define the $1 \times J$ vector $\mathbf{a}_j$ as

$$\mathbf{a}_j = \mathbf{f}_j\mathbf{H}$$

and the $1 \times J$ vector $\breve{\mathbf{a}}_j$ as the vector that is obtained by zeroing the $j$th element of $\mathbf{a}_j$, the SINR and BER for the $j$th user at the output of a linear multi-user equalizer $\mathbf{f}_j$ can be expressed as

$$SINR_j = \frac{|\mathbf{a}_j(j)|^2\mathbf{R}_{\mathbf{s}}(j,j)}{\breve{\mathbf{a}}_j\mathbf{R}_{\mathbf{s}}\breve{\mathbf{a}}_j^H + \mathbf{f}_j\mathbf{R}_{\mathbf{e}^{cut}}\mathbf{f}_j^H}, \tag{3.5}$$

$$BER_j \approx \mathrm{Q}\left\{SINR_j\right\}, \tag{3.6}$$

where $\mathrm{Q}\{\cdot\}$ is the well-known BER for signaling over an additive white Gaussian noise (AWGN) channel as a function of the signal-to-noise ratio (SNR) per data symbol for the chosen modulation scheme [Pro89]. Note that the approximation in (3.6) becomes an equality if the interference plus noise at the output of the linear multi-user equalizer $\mathbf{f}_j$ is Gaussian. We now discuss two linear multi-user equalizers in more detail: the ZF linear multi-user equalizer [TG96, TG97, WP98, KB93, KKKB96] and the MMSE linear multi-user equalizer [MH94, Mil95, KKKB96, SD99, GSP98]. To guarantee the existence of the ZF linear multi-user equalizer, we make the following assumption:

**Assumption 3.3.1** The $(N-L+1) \times J$ composite channel matrix $\mathbf{H}$ has full column rank $J$. □

This assumption requires that

$$N - L + 1 \geq J.$$

*ZF Linear Multi-User Equalizer:*

We say that a linear multi-user equalizer $\mathbf{f}_j$ is ZF, if

$$\mathbf{a}_j = \mathbf{f}_j\mathbf{H} = \mathbf{1}_j.$$

The ZF linear multi-user equalizer (we take the one that minimizes the MSE $\mathrm{E}\{\|\hat{s}_j[k] - s_j[k]\|^2\}$) can be expressed as

$$\mathbf{f}_{j,ZF} = \mathbf{1}_j(\mathbf{H}^H\mathbf{R}_{\mathbf{e}^{cut}}^{-1}\mathbf{H})^{-1}\mathbf{H}^H\mathbf{R}_{\mathbf{e}^{cut}}^{-1}. \tag{3.7}$$

If the data symbol sequences $\{s_j[k]\}_{j=1}^{J}$ are mutually uncorrelated and zero-mean with variance 1 ($\mathbf{R_s} = \mathbf{I}_J$) and the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$ ($\mathbf{R}_{\mathbf{e}^{cut}} = \sigma_e^2 \mathbf{I}_{N-L+1}$), (3.7), (3.5) and (3.6) become

$$\begin{aligned} \mathbf{f}_{j,ZF} &= \mathbf{1}_j (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H, \\ SINR_{j,ZF} &= \frac{1}{\sigma_e^2 [(\mathbf{H}^H \mathbf{H})^{-1}](j,j)}, \\ BER_{j,ZF} &\approx \mathrm{Q}\{SINR_{j,ZF}\}. \end{aligned} \tag{3.8}$$

The derivation of these formulas is presented in [KKKB96]. Note that when $\mathbf{H}$ is badly conditioned, the ZF linear multi-user receiver will boost the noise (high denominator in formula for $SINR_{j,ZF}$).

**Remark 3.3.2** It is clear that calculating the desired user's ZF linear multi-user equalizer according to (3.8) requires the knowledge of all the channels and code sequences. However, it is also possible to compute the desired user's ZF linear multi-user equalizer based only on the knowledge of the desired user's channel and code sequence, and the noise variance [WP98]. □

*MMSE Linear Multi-User Equalizer:*

We say that a linear multi-user equalizer $\mathbf{f}_j$ is MMSE, if the MSE

$$\mathrm{E}\{\|\hat{s}_j[k] - s_j[k]\|^2\}$$

is minimized. The MMSE linear multi-user equalizer is given by

$$\mathbf{f}_{j,MMSE} = \mathbf{1}_j (\mathbf{H}^H \mathbf{R}_{\mathbf{e}^{cut}}^{-1} \mathbf{H} + \mathbf{R}_{\mathbf{s}}^{-1})^{-1} \mathbf{H}^H \mathbf{R}_{\mathbf{e}^{cut}}^{-1}. \tag{3.9}$$

If the data symbol sequences $\{s_j[k]\}_{j=1}^{J}$ are mutually uncorrelated and zero-mean with variance 1 ($\mathbf{R_s} = \mathbf{I}_J$) and the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$ ($\mathbf{R}_{\mathbf{e}^{cut}} = \sigma_e^2 \mathbf{I}_{N-L+1}$), (3.9), (3.5) and (3.6) become

$$\begin{aligned} \mathbf{f}_{j,MMSE} &= \mathbf{1}_j (\mathbf{H}^H \mathbf{H} + \sigma_e^2 \mathbf{I}_J)^{-1} \mathbf{H}^H, \\ SINR_{j,MMSE} &= \frac{[(\mathbf{I}_J + \sigma_e^2 (\mathbf{H}^H \mathbf{H})^{-1})^{-1}](j,j)}{1 - [(\mathbf{I}_J + \sigma_e^2 (\mathbf{H}^H \mathbf{H})^{-1})^{-1}](j,j)}, \\ BER_{j,MMSE} &\approx \mathrm{Q}\{SINR_{j,MMSE}\}. \end{aligned} \tag{3.10}$$

The derivation of these formulas is again presented in [KKKB96]. Note that the MMSE linear multi-user equalizer is the linear multi-user equalizer that maximizes the SINR. Hence, we have $SINR_{j,MMSE} \leq SINR_{j,ZF}$, where the equality is obtained if $\sigma_e^2$ goes to zero.

**Remark 3.3.3** It is clear that calculating the desired user's MMSE linear multi-user equalizer according to (3.10) requires the knowledge of all the channels and code sequences, and the noise variance. However, it is also possible

to compute the desired user's MMSE linear multi-user equalizer based only on the knowledge of the desired user's channel and code sequence [SD99]. □

## 3.4 Blind Channel Estimation

Within the framework of the linear multi-user equalizer, we present a subspace deterministic blind multi-user channel estimation algorithm. This algorithm is performed on the model (3.2). It is labeled as *subspace*, since it is based on a subspace decomposition. We first describe how the algorithm works and then give a performance analysis. We finally show how the channel gain can be estimated. Note that this blind channel estimation approach is presented in [LX96b], including the performance analysis and channel gain estimation.

### 3.4.1 Algorithm

In this section, we describe how the algorithm works [LX96b]. For a burst length of $K$, we first define the $(N-L+1)\times K$ matrix $\mathbf{Y}$ as

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}^{cut}[0] & \mathbf{y}^{cut}[1] & \cdots & \mathbf{y}^{cut}[K-1] \end{bmatrix}.$$

Using (3.2) we then obtain

$$\mathbf{Y} = \mathbf{H}\mathbf{S} + \mathbf{E}, \tag{3.11}$$

where $\mathbf{E}$ is similarly defined as $\mathbf{Y}$ and $\mathbf{S}$ is the $J \times K$ matrix given by

$$\mathbf{S} = \begin{bmatrix} \mathbf{s}[0] & \mathbf{s}[1] & \cdots & \mathbf{s}[K-1] \end{bmatrix}.$$

To develop the algorithm, we make the following assumption:

**Assumption 3.4.1** The $(N-L+1)\times J$ composite channel matrix $\mathbf{H}$ has full column rank $J$ and has more rows than columns. □

This assumption requires that

$$N - L + 1 > J.$$

Further, we introduce a condition on the inputs:

**Assumption 3.4.2** The $J \times K$ matrix $\mathbf{S}$ has full row rank $J$. □

This assumption requires that

$$K \geq J.$$

Let us first, for the sake of clarity, assume that there is no additive noise present in $\mathbf{Y}$. Calculating the singular value decomposition (SVD) of $\mathbf{Y}$ then leads to

$$\mathbf{Y} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H,$$

where $\mathbf{\Sigma}$ is a diagonal matrix (diagonal elements in descending order) of the same size as $\mathbf{Y}$ and $\mathbf{U}$ and $\mathbf{V}$ are square unitary matrices. Because of assumptions 3.4.1 and 3.4.2, the $(N-L+1) \times K$ matrix $\mathbf{Y}$ has rank $J$ and, defining the $(N-L+1) \times (N-L+1-J)$ matrix $\mathbf{U}^n$ as the last $N-L+1-J$ columns of $\mathbf{U}$ (because of assumption 3.4.1, $\mathbf{U}^n$ is not empty), the columns of $\mathbf{U}^n$ form an orthonormal basis of the left null space of $\mathbf{H}$:

$$\mathbf{U}^{nH}\mathbf{H} = \mathbf{O}.$$

For the $j$th user we can then write

$$\mathbf{U}^{nH}\mathbf{h}_j^{cut} = \mathbf{0}. \tag{3.12}$$

Because of (2.6) and (2.13), we can express $\mathbf{h}_j^{cut}$ as

$$\mathbf{h}_j^{cut} = \mathbf{C}_j\mathbf{g}_j, \tag{3.13}$$

where $\mathbf{C}_j$ is the $(N-L+1) \times L$ matrix, given by

$$\mathbf{C}_j = \begin{bmatrix} c_j[L-1] & \cdots & c_j[0] \\ c_j[L] & \cdots & c_j[1] \\ \vdots & & \vdots \\ c_j[N-1] & \cdots & c_j[N-L] \end{bmatrix},$$

and $\mathbf{g}_j$ is defined in (2.12). Hence, (3.12) can be rewritten as

$$\mathbf{U}^{nH}\mathbf{C}_j\mathbf{g}_j = \mathbf{0}. \tag{3.14}$$

This equation may allow us to identify $g_j[n]$.

We introduce the following condition:

**Condition 3.4.3** The intersection of the column spaces of $\mathbf{H}$ and $\mathbf{C}_j$ has dimension 1 and $\mathbf{C}_j$ has full column rank $L$. □

Using this condition, we have the following identifiability result [LX96b]:

**Theorem 3.4.4** *Under assumptions* (3.4.1) *and* (3.4.2)*, the channel* $g_j[n]$ *can be uniquely (up to a complex scaling factor) determined from* (3.14)*, if and only if condition 3.4.3 is satisfied.*

**Proof:** See [LX96b]. □

Under assumptions (3.4.1) and (3.4.2), it is easy to show that condition 3.4.3 is equivalent with $\mathbf{C}_j^H \mathbf{U}^n$ having a one-dimensional left null space or equivalently having rank $L-1$. It is then clear that this can only be satisfied if

$$N - L - J \geq L.$$

Let us now assume that additive noise is present in $\mathbf{Y}$. Calculating the SVD of $\mathbf{Y}$ then leads to

$$\mathbf{Y} = \hat{\mathbf{U}}\hat{\mathbf{\Sigma}}\hat{\mathbf{V}}^H,$$

where $\hat{\mathbf{\Sigma}}$ is a diagonal matrix (diagonal elements in descending order) of the same size as $\hat{\mathbf{Y}}$ and $\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$ are square unitary matrices. Defining the $(N-L+1)\times(N-L+1-J)$ matrix $\hat{\mathbf{U}}^n$ as the last $N-L+1-J$ columns of $\hat{\mathbf{U}}$, we then consider the following minimization problem:

$$\boxed{\begin{array}{c}\min_{\mathbf{g}_j}\{\|\hat{\mathbf{U}}^{nH}\mathbf{C}_j\mathbf{g}_j\|^2\},\\ \text{s.t. a non-triviality constraint imposed on } \mathbf{g}_j.\end{array}}$$

If we impose a unit norm constraint on $\mathbf{g}_j$ (this constraint is mostly used), then the left singular vector of $\mathbf{C}_j^H\hat{\mathbf{U}}^n$ corresponding to the smallest singular value represents a possible solution. Since this vector can be interpreted as an estimate of $\mathbf{g}_j^o$, the left singular vector of $\mathbf{C}_j^H\mathbf{U}^n$ corresponding to the smallest singular value, we will denote this solution as $\hat{\mathbf{g}}_j^o$. As an estimate for the channel vector $\mathbf{g}_j$ we then consider

$$\hat{\mathbf{g}}_j = \hat{\gamma}_j\hat{\mathbf{g}}_j^o, \tag{3.15}$$

where $\hat{\gamma}_j$ is an estimate of $\gamma_j$, which is given by $\mathbf{g}_j = \gamma_j\mathbf{g}_j^o$. This $\gamma_j$ can be estimated from some short known headers that are transmitted, or we can blindly estimate $|\gamma_j|$, which is equal to the channel gain $\|\mathbf{g}_j\|$ (see section 3.4.3), and use an appropriate differential modulation scheme to get rid of the phase ambiguity [Pro89]. However, for simplicity we will estimate $\gamma_j$ as

$$\hat{\gamma}_j = \mathbf{g}_j\hat{\mathbf{g}}_j^{o\dagger}.$$

This leads to an estimate $\hat{\mathbf{g}}_j$ that is optimal in the LS sense (since $\mathbf{g}_j$ is unknown, this is of course not feasible in practice).

1. compute the SVD of $\mathbf{Y}$:
$$\mathbf{Y} = \hat{\mathbf{U}}\hat{\mathbf{\Sigma}}\hat{\mathbf{V}}^H$$
2. collect the last $N - L + 1 - J$ columns of $\hat{\mathbf{U}}$: $\hat{\mathbf{U}}^n$
3. construct $\mathbf{C}_j$ as
$$\mathbf{C}_j = \begin{bmatrix} c_j[L-1] & \cdots & c_j[0] \\ c_j[L] & \cdots & c_j[1] \\ \vdots & & \vdots \\ c_j[N-1] & \cdots & c_j[N-L] \end{bmatrix}$$
4. solve
$$\min_{\mathbf{g}_j} \{\|\hat{\mathbf{U}}^{n\,H}\mathbf{C}_j\mathbf{g}_j\|^2\}, \quad \text{s.t. } \|\mathbf{g}_j\|^2 = 1$$

Table 3.1: Blind multi-user channel estimation algorithm.

**Remark 3.4.5** Note that the estimation of the desired user's channel only requires the desired user's code sequence. □

A brief outline of the algorithm is given in **table 3.1**.

### 3.4.2 Performance Analysis

The performance analysis is given in the following theorem [LX96b]:

**Theorem 3.4.6** *Assume that the channel vector estimate $\hat{\mathbf{g}}_j$ is obtained as in (3.15), with $\hat{\gamma}_j = \gamma_j$. Further, assume that the data symbol sequences $\{s_j[k]\}_{j=1}^J$ are mutually uncorrelated and zero-mean with variance 1 and the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$. Then, only considering the first order approximation of $\Delta\mathbf{g}_j = \hat{\mathbf{g}}_j - \mathbf{g}_j$ in $\mathbf{E}$, the bias and the normalized MSE (NMSE) of $\hat{\mathbf{g}}_j$ can be expressed as*

$$bias_j = \mathbf{0},$$
$$NMSE_j \approx \frac{\sigma_e^2}{K\|\mathbf{g}_j\|^2}\|\mathbf{T}_j^{\dagger H}\|^2,$$

*with $\mathbf{T}_j = \mathbf{C}_j^H\mathbf{U}^n$.*

**Proof:** Using a result from subspace perturbation analysis [LLV93], the first order approximation of $\Delta\hat{\mathbf{U}}^n = \hat{\mathbf{U}}^n - \mathbf{U}^n$ in $\mathbf{E}$ can be written as

$$\Delta^{(1)}\mathbf{U}^n = -\mathbf{Y}^{\dagger H}\mathbf{E}^H\mathbf{U}^n.$$

If we then define $\mathbf{T}_j = \mathbf{C}_j^H \mathbf{U}^n$ and $\hat{\mathbf{T}}_j = \mathbf{C}_j^H \hat{\mathbf{U}}^n$, the first order approximation of $\Delta\mathbf{T}_j = \hat{\mathbf{T}}_j - \mathbf{T}_j$ in $\mathbf{E}$ can be written as

$$\Delta^{(1)}\mathbf{T}_j = \mathbf{C}_j^H \Delta^{(1)}\mathbf{U}^n.$$

If $\hat{\mathbf{g}}_j$ is obtained as in (3.15), with $\hat{\gamma}_j = \gamma_j$, then we can use the same result from subspace perturbation analysis as before to determine the first order approximation of $\Delta\mathbf{g}_j = \hat{\mathbf{g}}_j - \mathbf{g}_j$ in $\mathbf{E}$:

$$\begin{aligned}\Delta^{(1)}\mathbf{g}_j &= -\mathbf{T}_j^{\dagger H}\Delta^{(1)}\mathbf{T}_j^H\mathbf{g}_j \\ &= -\mathbf{T}_j^{\dagger H}\Delta^{(1)}\mathbf{U}^{nH}\mathbf{C}_j\mathbf{g}_j \\ &= \mathbf{T}_j^{\dagger H}\mathbf{U}^{nH}\mathbf{E}\mathbf{Y}^{\dagger}\mathbf{h}_j.\end{aligned}$$

Only considering the first order approximation of $\Delta\mathbf{g}_j$ in $\mathbf{E}$, the bias and the MSE of $\hat{\mathbf{g}}_j$ can be expressed as

$$bias_j = \mathrm{E}\{\Delta^{(1)}\mathbf{g}_j\} \quad \text{and} \quad MSE_j = \mathrm{E}\{\Delta^{(1)}\mathbf{g}_j^H\Delta^{(1)}\mathbf{g}_j\}.$$

Assume that the data symbol sequences $\{s_j[k]\}_{j=1}^J$ are mutually uncorrelated and zero-mean with variance 1 and the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$. The bias of $\hat{\mathbf{g}}_j$ then becomes

$$bias_j = \mathrm{E}\{\Delta^{(1)}\mathbf{g}_j\} = \mathbf{0}.$$

The MSE of $\hat{\mathbf{g}}_j$ then becomes

$$\begin{aligned}MSE_j &= \mathrm{E}\{\Delta^{(1)}\mathbf{g}_j^H\Delta^{(1)}\mathbf{g}_j\} \\ &= \sigma_e^2\|\mathbf{T}_j^{\dagger H}\|^2\mathrm{E}\{\|\mathbf{Y}^{\dagger}\mathbf{h}_j\|^2\}.\end{aligned}$$

Following a similar reasoning as in [LLV93], we can further prove that

$$\mathrm{E}\{\|\mathbf{Y}^{\dagger}\mathbf{h}_j\|^2\} \approx \frac{1}{K}.$$

Therefore, the NMSE of $\hat{\mathbf{g}}_j$ can be written as

$$NMSE_j \approx \frac{\sigma_e^2}{K\|\mathbf{g}_j\|^2}\|\mathbf{T}_j^{\dagger H}\|^2. \qquad \square$$

### 3.4.3 Channel Gain Estimation

In this section, we show how to estimate $|\gamma_j|$, which is equal to the channel gain $\|\mathbf{g}_j\|$, from $\mathbf{Y}$ [LX96b]. Assume that the data symbol sequences $\{s_j[k]\}_{j=1}^J$ are mutually uncorrelated and zero-mean with variance 1 ($\mathbf{R_s} = \mathbf{I}_J$) and the

additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$ ($\mathbf{R}_{\mathbf{e}^{cut}} = \sigma_e^2 \mathbf{I}_{N-L+1}$). Then, we can write (3.4) as

$$\begin{aligned}\mathbf{R}_{\mathbf{y}^{cut}} &= \mathbf{H}\mathbf{H}^H + \sigma_e^2 \mathbf{I}_{N-L+1} \\ &= \mathbf{H}^o \begin{bmatrix} \gamma_1^2 & & \\ & \ddots & \\ & & \gamma_J^2 \end{bmatrix} \mathbf{H}^{oH} + \sigma_e^2 \mathbf{I}_{N-L+1},\end{aligned}$$

where $\mathbf{H}^o$ is similarly defined as $\mathbf{H}$, using $\{\mathbf{h}_j^{cuto} = \mathbf{C}_j \mathbf{g}_j^o\}_{j=1}^J$ instead of $\{\mathbf{h}_j^{cut} = \mathbf{C}_j \mathbf{g}_j\}_{j=1}^J$. Hence, we can write

$$|\gamma_j|^2 = \mathbf{f}_{j,ZF}^o (\mathbf{R}_{\mathbf{y}^{cut}} - \sigma_e^2 \mathbf{I}_{N-L+1}) \mathbf{f}_{j,ZF}^{oH},$$

where $\mathbf{f}_{j,ZF}^o$ is similarly defined as $\mathbf{f}_{j,ZF}$ (see (3.8)), using $\mathbf{H}^o$ instead of $\mathbf{H}$. This leads to the following estimate of $|\gamma_j|^2$:

$$|\hat{\gamma}_j|^2 = \hat{\mathbf{f}}_{j,ZF}^o (\hat{\mathbf{R}}_{\mathbf{y}^{cut}} - \hat{\sigma}_e^2 \mathbf{I}_{N-L+1}) \hat{\mathbf{f}}_{j,ZF}^{oH}, \tag{3.16}$$

where $\hat{\mathbf{f}}_{j,ZF}^o$ is similarly defined as $\mathbf{f}_{j,ZF}^o$, using $\hat{\mathbf{H}}^o$ instead of $\mathbf{H}^o$, with $\hat{\mathbf{H}}^o$ similarly defined as $\mathbf{H}^o$, using $\{\hat{\mathbf{h}}_j^{cuto} = \mathbf{C}_j \hat{\mathbf{g}}_j^o\}_{j=1}^J$ instead of $\{\mathbf{h}_j^{cuto} = \mathbf{C}_j \mathbf{g}_j^o\}_{j=1}^J$. Further $\hat{\mathbf{R}}_{\mathbf{y}^{cut}}$ is defined as

$$\hat{\mathbf{R}}_{\mathbf{y}^{cut}} = \frac{1}{K} \mathbf{Y}\mathbf{Y}^H$$

and $\hat{\sigma}_e^2$ is for instance obtained as the average of the $N - L + 1 - J$ smallest eigenvalues of $\hat{\mathbf{R}}_{\mathbf{y}^{cut}}$. Note that this estimate of the noise variance can also be used in the MMSE linear multi-user equalizer, when the noise variance $\sigma_e^2$ is unknown.

## 3.5 Conclusions

In this chapter, we have reviewed some existing receivers for a DS-CDMA system. First, the RAKE receiver has been discussed. Then, we have discussed the linear multi-user equalizer. Within the framework of the linear multi-user equalizer, we have further presented a subspace deterministic blind multi-user channel estimation algorithm. Complexity issues and simulation results will be presented in chapter 5.

# Chapter 4

# Receivers for a DS-CDMA-BS System

In this chapter, we develop new receivers for a DS-CDMA-BS system. Like in the previous chapter, we consider a single receive antenna and an ICI-limited quasi-synchronous propagation model. We further use short code sequences.

Details on the used data model are given in section 4.1. In section 4.2, the block RAKE receiver is introduced. This receiver consists of a bank of block correlators, followed by a linear block combiner, which linearly combines the block correlator outputs. In section 4.3, we then introduce the MUI-free receiver, which completely removes the MUI, without using any channel information. This receiver basically is a modified block RAKE receiver, which consists of a bank of modified block correlators, followed by a linear block combiner, which linearly combines the modified block correlator outputs. The MUI-free operation is obtained by the use of a shift-orthogonal set of code sequences, on which this receiver is based. To make the MUI-free receiver useful in practice, we develop a design strategy for a shift-orthogonal set of code sequences in section 4.4. This design strategy can be considered as one of the main contributions of this chapter. Within the framework of the MUI-free receiver, we then present a subspace deterministic blind single-user channel estimation algorithm in section 4.5. We also give a performance analysis of this algorithm and show how the channel gain can be estimated. Note that, in a different context, a similar blind channel estimation approach is presented in [TG97]. However, no performance analysis or channel gain estimation is presented there. We end with some conclusions in section 4.6.

## 4.1 Data Model

We consider a single receive antenna and an ICI-limited quasi-synchronous propagation model (see section 2.2.1). We further use short code sequences (see section 2.1.4). As proposed in section 2.2.1, when an ICI-limited quasi-synchronous propagation model is considered for a DS-CDMA-BS system, we take the block size $B$ equal to the parameter $L$ (see (2.10)). The data model that will be used is given by (2.16), which is repeated here for convenience:

$$\mathbf{y}[n] = \sum_{j=1}^{J} \sum_{n'=0}^{1} \mathbf{G}_j[n']\mathbf{x}_j[n-n'] + \mathbf{e}[n]. \tag{4.1}$$

## 4.2 Block RAKE Receiver

The block RAKE receiver consists of a bank of block correlators, followed by a linear block combiner, which linearly combines the block correlator outputs.

### 4.2.1 Bank of Block Correlators

Since $\mathbf{G}_j[n]$ is an FIR matrix filter of order 1 with delay index 0, we apply a bank of 2 block correlators. The first block correlator of the $j$th user despreads the received block sequence $\mathbf{y}[n]$ matched to $\mathbf{G}_j[0]$, using the code sequence $c_j[n]$, while the second block correlator of the $j$th user despreads the received block sequence $\mathbf{y}[n]$ matched to $\mathbf{G}_j[1]$, using the code sequence $c_j[n]$. The output of the first block correlator can be written as

$$\begin{aligned}\mathbf{z}_{j,0}[k] &= \sum_{n=0}^{N-1} c_j^*[n]\mathbf{y}[kN+n] \\ &= \mathbf{c}_j^H \mathbf{c}_j \mathbf{G}_j[0]\mathbf{s}_j[k] + \mathbf{p}_{j,0}[k] + \mathbf{q}_{j,0}[k] + \mathbf{n}_{j,0}[k],\end{aligned} \tag{4.2}$$

while the output of the second block correlator can be written as

$$\begin{aligned}\mathbf{z}_{j,1}[k] &= \sum_{n=0}^{N-1} c_j^*[n]\mathbf{y}[kN+n+1] \\ &= \mathbf{c}_j^H \mathbf{c}_j \mathbf{G}_j[1]\mathbf{s}_j[k] + \mathbf{p}_{j,1}[k] + \mathbf{q}_{j,1}[k] + \mathbf{n}_{j,1}[k].\end{aligned} \tag{4.3}$$

In these formulas $\mathbf{p}_{j,0}[k]$ and $\mathbf{p}_{j,1}[k]$ represent the *residual ICBI*:

$$\begin{aligned}\mathbf{p}_{j,0}[k] &= \mathbf{c}_j^{\downarrow(N-1)H}\mathbf{c}_j^{\uparrow(N-1)}\mathbf{G}_j[1]\mathbf{s}_j[k-1] + \mathbf{c}_j^{\uparrow(1)H}\mathbf{c}_j^{\downarrow(1)}\mathbf{G}_j[1]\mathbf{s}_j[k], \\ \mathbf{p}_{j,1}[k] &= \mathbf{c}_j^{\downarrow(1)H}\mathbf{c}_j^{\uparrow(1)}\mathbf{G}_j[0]\mathbf{s}_j[k] + \mathbf{c}_j^{\uparrow(N-1)H}\mathbf{c}_j^{\downarrow(N-1)}\mathbf{G}_j[0]\mathbf{s}_j[k+1],\end{aligned}$$

$\mathbf{q}_{j,0}[k]$ and $\mathbf{q}_{j,1}[k]$ represent the *residual MUI*:

$$\mathbf{q}_{j,0}[k] = \sum_{\substack{j'=1 \\ j' \neq j}}^{J} \mathbf{c}_j^H \mathbf{c}_{j'} \mathbf{G}_{j'}[0]\mathbf{s}_{j'}[k] + \mathbf{c}_j^{\downarrow(N-1)H} \mathbf{c}_{j'}^{\uparrow(N-1)} \mathbf{G}_{j'}[1]\mathbf{s}_{j'}[k-1] + \mathbf{c}_j^{\uparrow(1)H} \mathbf{c}_{j'}^{\downarrow(1)} \mathbf{G}_{j'}[1]\mathbf{s}_{j'}[k],$$

$$\mathbf{q}_{j,1}[k] = \sum_{\substack{j'=1 \\ j' \neq j}}^{J} \mathbf{c}_j^{\downarrow(1)H} \mathbf{c}_{j'}^{\uparrow(1)} \mathbf{G}_{j'}[0]\mathbf{s}_{j'}[k] + \mathbf{c}_j^{\uparrow(N-1)H} \mathbf{c}_{j'}^{\downarrow(N-1)} \mathbf{G}_{j'}[0]\mathbf{s}_{j'}[k+1] + \mathbf{c}_j^H \mathbf{c}_{j'} \mathbf{G}_{j'}[1]\mathbf{s}_{j'}[k]$$

and $\mathbf{n}_{j,0}[k]$ and $\mathbf{n}_{j,1}[k]$ represent the *residual additive noise*:

$$\mathbf{n}_{j,0}[k] = \sum_{n=0}^{N-1} c_j^*[n]\mathbf{e}[kN+n],$$

$$\mathbf{n}_{j,1}[k] = \sum_{n=0}^{N-1} c_j^*[n]\mathbf{e}[kN+n+1].$$

### 4.2.2 Linear Block Combiner

We then linearly combine the 2 block correlator outputs $\mathbf{z}_{j,0}[k]$ and $\mathbf{z}_{j,1}[k]$ to suppress the remaining residual ICBI and MUI, and the remaining ISI w.r.t. the symbols in $\mathbf{s}_j[k]$:

$$\hat{\mathbf{s}}_j[k] = \mathbf{F}_{j,0}\mathbf{z}_{j,0}[k] + \mathbf{F}_{j,1}\mathbf{z}_{j,1}[k],$$

where $\mathbf{F}_{j,0}$ and $\mathbf{F}_{j,1}$ are the $L \times L$ linear block combiner weights. We can for instance use the MF linear block combiner, resulting in the coherent block RAKE receiver. We then take

$$\mathbf{F}_{j,0} = \frac{\mathbf{G}_j^H[0]}{\|\mathbf{g}_j\|\|\mathbf{c}_j\|}, \quad \mathbf{F}_{j,1} = \frac{\mathbf{G}_j^H[1]}{\|\mathbf{g}_j\|\|\mathbf{c}_j\|}.$$

## 4.3 MUI-Free Receiver

The MUI-free receiver basically is a modified block RAKE receiver, which consists of a bank of modified block correlators, followed by a linear block combiner, which linearly combines the modified block correlator outputs. The MUI-free operation is obtained by the use of a shift-orthogonal set of code sequences, on which this receiver is based.

### 4.3.1 Bank of Modified Block Correlators

Instead of applying a bank of 2 block correlators, as in the previous section, we here apply a bank of 2 modified block correlators. In contrast with the 2 block correlators, the 2 modified block correlators use 2 slightly different code sequences to perform the block despreading. The first modified block correlator for the $j$th user despreads the received block sequence $\mathbf{y}[n]$ matched to $\mathbf{G}_j[0]$, using the code sequence $c_j^{\uparrow(1)}[n]$, where

$$c_j^{\uparrow(1)}[n] = \begin{cases} 0, & \text{for } n = 0 \\ c_j[n], & \text{for } n \neq 0 \end{cases},$$

while the second modified block correlator for the $j$th user despreads the received block sequence $\mathbf{y}[n]$ matched to $\mathbf{G}_j[1]$, using the code sequence $c_j^{\downarrow(1)}[n]$, where

$$c_j^{\downarrow(1)}[n] = \begin{cases} c_j[n], & \text{for } n \neq N-1 \\ 0, & \text{for } n = N-1 \end{cases}.$$

Note the correspondence with the notations (2.3) and (2.4). The output of the first modified block correlator can be written as

$$\begin{aligned} \mathbf{z}_{j,0}^{mod}[k] &= \sum_{n=0}^{N-1} c_j^{\uparrow(1)*}[n]\mathbf{y}[kN+n] = \sum_{n=1}^{N-1} c_j^{*}[n]\mathbf{y}[kN+n] \\ &= \mathbf{c}_j^{\uparrow(1)H}\mathbf{c}_j^{\uparrow(1)}\mathbf{G}_j[0]\mathbf{s}_j[k] + \mathbf{p}_{j,0}^{mod}[k] + \mathbf{q}_{j,0}^{mod}[k] + \mathbf{n}_{j,0}^{mod}[k], \end{aligned} \tag{4.4}$$

while the output of the second modified block correlator can be written as

$$\begin{aligned} \mathbf{z}_{j,1}^{mod}[k] &= \sum_{n=0}^{N-1} c_j^{\downarrow(1)*}[n]\mathbf{y}[kN+n+1] = \sum_{n=0}^{N-2} c_j^{*}[n]\mathbf{y}[kN+n+1] \\ &= \mathbf{c}_j^{\downarrow(1)H}\mathbf{c}_j^{\downarrow(1)}\mathbf{G}_j[1]\mathbf{s}_j[k] + \mathbf{p}_{j,1}^{mod}[k] + \mathbf{q}_{j,1}^{mod}[k] + \mathbf{n}_{j,1}^{mod}[k]. \end{aligned} \tag{4.5}$$

In these formulas $\mathbf{p}_{j,0}^{mod}[k]$ and $\mathbf{p}_{j,1}^{mod}[k]$ represent the *residual ICBI*:

$$\begin{aligned} \mathbf{p}_{j,0}^{mod}[k] &= \mathbf{c}_j^{\uparrow(1)H}\mathbf{c}_j^{\downarrow(1)}\mathbf{G}_j[1]\mathbf{s}_j[k], \\ \mathbf{p}_{j,1}^{mod}[k] &= \mathbf{c}_j^{\downarrow(1)H}\mathbf{c}_j^{\uparrow(1)}\mathbf{G}_j[0]\mathbf{s}_j[k], \end{aligned}$$

$\mathbf{q}_{j,0}^{mod}[k]$ and $\mathbf{q}_{j,1}^{mod}[k]$ represent the *residual MUI*:

$$\mathbf{q}_{j,0}^{mod}[k] = \sum_{\substack{j'=1 \\ j'\neq j}}^{J} \mathbf{c}_j^{\uparrow(1)H}\mathbf{c}_{j'}^{\uparrow(1)}\mathbf{G}_{j'}[0]\mathbf{s}_{j'}[k] + \mathbf{c}_j^{\uparrow(1)H}\mathbf{c}_{j'}^{\downarrow(1)}\mathbf{G}_{j'}[1]\mathbf{s}_{j'}[k],$$

$$\mathbf{q}_{j,1}^{mod}[k] = \sum_{\substack{j'=1 \\ j' \neq j}}^{J} \mathbf{c}_j^{\downarrow(1)H} \mathbf{c}_{j'}^{\uparrow(1)} \mathbf{G}_{j'}[0]\mathbf{s}_{j'}[k] + \mathbf{c}_j^{\downarrow(1)H} \mathbf{c}_{j'}^{\downarrow(1)} \mathbf{G}_{j'}[1]\mathbf{s}_{j'}[k]$$

and $\mathbf{n}_{j,0}^{mod}[k]$ and $\mathbf{n}_{j,1}^{mod}[k]$ represent the *residual additive noise*:

$$\mathbf{n}_{j,0}^{mod}[k] = \sum_{n=0}^{N-1} c_j^{\uparrow(1)*}[n]\mathbf{e}[kN+n] = \sum_{n=1}^{N-1} c_j^{*}[n]\mathbf{e}[kN+n],$$
$$\mathbf{n}_{j,1}^{mod}[k] = \sum_{n=0}^{N-1} c_j^{\downarrow(1)*}[n]\mathbf{e}[kN+n+1] = \sum_{n=0}^{N-2} c_j^{*}[n]\mathbf{e}[kN+n+1].$$

Compared to the residual ICBI and MUI formulas presented in the previous section, the above residual ICBI and MUI formulas only contain terms in the time index $k$.

Assume now that a shift orthogonal set of code sequences is used:

**Definition 4.3.1** A set of $J$ length-$N$ code sequences $\{c_j[n]\}_{j=1}^{J}$ is shift-orthogonal, if and only if

$$\begin{cases} \mathbf{c}_j^{\uparrow(1)H}\mathbf{c}_{j'}^{\uparrow(1)} = \mathbf{c}_j^{\downarrow(1)H}\mathbf{c}_{j'}^{\downarrow(1)} = \eta\delta[j-j'] \\ \mathbf{c}_j^{\uparrow(1)H}\mathbf{c}_{j'}^{\downarrow(1)} = \mathbf{c}_j^{\downarrow(1)H}\mathbf{c}_{j'}^{\uparrow(1)} = 0 \end{cases}, \quad \text{for } j, j' = 1, 2, \ldots, J, \tag{4.6}$$

where $\eta = (N-1)/N$. □

**Remark 4.3.2** The condition (4.6) can only be satisfied if $J \leq \lfloor (N-1)/2 \rfloor$, since $2J$ code vectors of size $(N-1) \times 1$ can only be mutually orthogonal if $2J \leq N-1$. □

Then, it is clear that the 2 modified block correlators completely remove the ICBI and MUI, and (4.4) and (4.5) can be written as

$$\mathbf{z}_{j,0}^{mod}[k] = \eta\mathbf{G}_j[0]\mathbf{s}_j[k] + \mathbf{n}_{j,0}^{mod}[k], \tag{4.7}$$
$$\mathbf{z}_{j,1}^{mod}[k] = \eta\mathbf{G}_j[1]\mathbf{s}_j[k] + \mathbf{n}_{j,1}^{mod}[k]. \tag{4.8}$$

Stacking the 2 modified block correlator outputs $\mathbf{z}_{j,0}^{mod}[k]$ and $\mathbf{z}_{j,1}^{mod}[k]$:

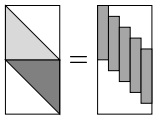
$$\mathbf{z}_j[k] = \left[\mathbf{z}_{j,0}^{modT}[k] \quad \mathbf{z}_{j,1}^{modT}[k]\right]^T,$$

we obtain

$$\mathbf{z}_j[k] = \eta\mathbf{G}_j\mathbf{s}_j[k] + \mathbf{n}_j[k], \tag{4.9}$$

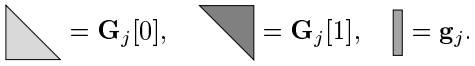
where $\mathbf{n}_j[k]$ is similarly defined as $\mathbf{z}_j[k]$ and $\mathbf{G}_j$ is the $2L \times L$ channel matrix for the $j$th user, given by

$$\mathbf{G}_j = \quad = \quad , \tag{4.10}$$



with

$= \mathbf{G}_j[0], \quad = \mathbf{G}_j[1], \quad = \mathbf{g}_j.$



Note that if the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$, we obtain

$$\mathbf{R}_{\mathbf{n}_j} = \mathrm{E}\{\mathbf{n}_j[k]\mathbf{n}_j^H[k]\} = \eta\sigma_e^2\mathbf{I}_{2L}.$$

### 4.3.2 Linear Block Combiner

We then linearly combine the 2 modified block correlator outputs $\mathbf{z}_{j,0}^{mod}[k]$ and $\mathbf{z}_{j,1}^{mod}[k]$ to suppress the remaining ISI w.r.t. the symbols in $\mathbf{s}_j[k]$:

$$\begin{aligned}\hat{\mathbf{s}}_j[k] &= \mathbf{F}_{j,0}\mathbf{z}_{j,0}^{mod}[k] + \mathbf{F}_{j,1}\mathbf{z}_{j,1}^{mod}[k] \\ &= \eta\left(\mathbf{F}_{j,0}\mathbf{G}_j[0] + \mathbf{F}_{j,1}\mathbf{G}_j[1]\right)\mathbf{s}_j[k] + \mathbf{F}_{j,0}\mathbf{n}_{j,0}^{mod}[k] + \mathbf{F}_{j,1}\mathbf{n}_{j,1}^{mod}[k],\end{aligned}$$

where $\mathbf{F}_{j,0}$ and $\mathbf{F}_{j,1}$ are the $L \times L$ linear block combiner weights. Using the notation of (4.9), we can also write

$$\hat{\mathbf{s}}_j[k] = \mathbf{F}_j\mathbf{z}_j[k] = \eta\mathbf{F}_j\mathbf{G}_j\mathbf{s}_j[k] + \mathbf{F}_j\mathbf{n}_j[k],$$

where $\mathbf{F}_j$ is the $L \times 2L$ linear block combiner for the $j$th user, given by

$$\mathbf{F}_j = \begin{bmatrix}\mathbf{F}_{j,0} & \mathbf{F}_{j,1}\end{bmatrix}.$$

The MUI-free receiver is illustrated in **figure 4.1**.

We now focus on the calculation of the signal-to-interference-plus-noise ratio (SINR) and bit error rate (BER) for the $j$th user at the output of the linear block combiner $\mathbf{F}_j$. First of all, observe that

$$\mathbf{R}_{\mathbf{z}_j} = \mathrm{E}\{\mathbf{z}_j[k]\mathbf{z}_j^H[k]\} = \eta^2\mathbf{G}_j\mathbf{R}_{\mathbf{s}_j}\mathbf{G}_j^H + \mathbf{R}_{\mathbf{n}_j}, \tag{4.11}$$

where $\mathbf{R}_{\mathbf{n}_j}$ is similarly defined as $\mathbf{R}_{\mathbf{z}_j}$ and

$$\mathbf{R}_{\mathbf{s}_j} = \mathrm{E}\{\mathbf{s}_j[k]\mathbf{s}_j^H[k]\}.$$

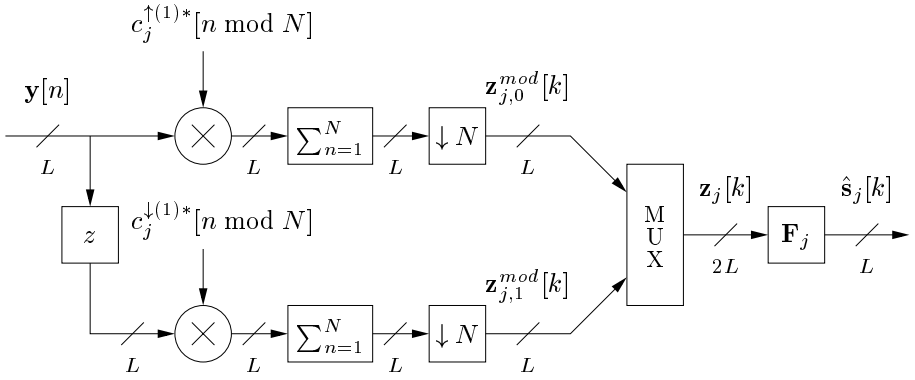



Figure 4.1: MUI-free receiver for a DS-CDMA-BS system.

If we then define the $L \times L$ matrix $\mathbf{A}_j$ as

$$\mathbf{A}_j = \eta \mathbf{F}_j \mathbf{G}_j$$

and the $L \times L$ matrix $\breve{\mathbf{A}}_j$ as the matrix that is obtained by zeroing the diagonal elements of $\mathbf{A}_j$, the SINR and BER for the $j$th user at the output of the linear block combiner $\mathbf{F}_j$ can be expressed as

$$SINR_j = \frac{\frac{1}{L}\sum_{l=1}^{L} |\mathbf{A}_j(l,l)|^2 \mathbf{R}_{\mathbf{s}_j}(l,l)}{\frac{1}{L}\sum_{l=1}^{L} ([\breve{\mathbf{A}}_j \mathbf{R}_{\mathbf{s}_j} \breve{\mathbf{A}}_j^H](l,l) + [\mathbf{F}_j \mathbf{R}_{\mathbf{n}_j} \mathbf{F}_j^H](l,l))}, \tag{4.12}$$

$$BER_j \approx \frac{1}{L}\sum_{l=1}^{L} \mathrm{Q}\left\{ \frac{|\mathbf{A}_j(l,l)|^2 \mathbf{R}_{\mathbf{s}_j}(l,l)}{[\breve{\mathbf{A}}_j \mathbf{R}_{\mathbf{s}_j} \breve{\mathbf{A}}_j^H](l,l) + [\mathbf{F}_j \mathbf{R}_{\mathbf{n}_j} \mathbf{F}_j^H](l,l)} \right\}, \tag{4.13}$$

where $\mathrm{Q}\{\cdot\}$ is the well-known BER for signaling over an additive white Gaussian noise (AWGN) channel as a function of the signal-to-noise ratio (SNR) per data symbol for the chosen modulation scheme [Pro89]. Note that the approximation in (4.13) becomes an equality if the interference plus noise at the output of the linear block combiner $\mathbf{F}_j$ is Gaussian. We now discuss two linear block combiners in more detail: the ZF linear block combiner and the MMSE linear block combiner. Because of the specific structure of $\mathbf{G}_j$ (see (4.10)), we have the following property:

**Property 4.3.3** The $2L \times L$ channel matrix $\mathbf{G}_j$ has full column rank $L$. □

From this property, it is clear that we do not have to make an assumption similar to assumption 3.3.1, to guarantee the existence of the ZF linear block combiner.

*ZF Linear Block Combiner:*

We say that a linear block combiner $\mathbf{F}_j$ is ZF, if

$$\mathbf{A}_j = \eta \mathbf{F}_j \mathbf{G}_j = \mathbf{I}_L.$$

The ZF linear block combiner (we take the one that minimizes the MSE $\mathrm{E}\{\|\hat{\mathbf{s}}_j[k] - \mathbf{s}_j[k]\|^2\}$) can be expressed as

$$\mathbf{F}_{j,ZF} = \eta^{-1}(\mathbf{G}_j^H \mathbf{R}_{\mathbf{n}_j}^{-1} \mathbf{G}_j)^{-1} \mathbf{G}_j^H \mathbf{R}_{\mathbf{n}_j}^{-1}. \tag{4.14}$$

If the data symbol sequence $s_j[k]$ is zero-mean white with variance 1 ($\mathbf{R}_{\mathbf{s}_j} = \mathbf{I}_L$) and the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$ ($\mathbf{R}_{\mathbf{n}_j} = \eta\sigma_e^2 \mathbf{I}_{2L}$), (4.14), (4.12) and (4.13) become

$$\mathbf{F}_{j,ZF} = \eta^{-1}(\mathbf{G}_j^H \mathbf{G}_j)^{-1} \mathbf{G}_j^H, \tag{4.15}$$

$$SINR_{j,ZF} = \frac{1}{\eta^{-1}\sigma_e^2 \dfrac{1}{L} \displaystyle\sum_{l=1}^{L} [(\mathbf{G}_j^H \mathbf{G}_j)^{-1}](l,l)},$$

$$BER_{j,ZF} \approx \frac{1}{L} \sum_{l=1}^{L} \mathrm{Q}\left\{ \frac{1}{\eta^{-1}\sigma_e^2 [(\mathbf{G}_j^H \mathbf{G}_j)^{-1}](l,l)} \right\}.$$

The derivation of these formulas is based on the work of [KKKB96]. Note that when $\mathbf{G}_j$ is badly conditioned, the ZF linear block combiner will boost the noise (high denominator in formula for $SINR_{j,ZF}$).

**Remark 4.3.4** Note that calculating the desired user's ZF linear block combiner according to (4.15) only requires the knowledge of the desired user's channel, as MUI has been eliminated by proper code design. □

*MMSE Linear Block Combiner:*

We say that a linear block combiner $\mathbf{F}_j$ is MMSE, if the MSE

$$\mathrm{E}\{\|\hat{\mathbf{s}}_j[k] - \mathbf{s}_j[k]\|^2\}$$

is minimized. The MMSE linear block combiner is given by

$$\mathbf{F}_{j,MMSE} = (\eta^2 \mathbf{G}_j^H \mathbf{R}_{\mathbf{n}_j}^{-1} \mathbf{G}_j + \mathbf{R}_{\mathbf{s}_j\mathbf{s}_j}^{-1})^{-1} \eta \mathbf{G}_j^H \mathbf{R}_{\mathbf{n}_j}^{-1}. \tag{4.16}$$

If the data symbol sequence $s_j[k]$ is zero-mean white with variance 1 ($\mathbf{R}_{\mathbf{s}_j} = \mathbf{I}_L$) and the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$ ($\mathbf{R}_{\mathbf{n}_j} = \eta\sigma_e^2 \mathbf{I}_{2L}$), (4.16), (4.12) and (4.13) become

$$\mathbf{F}_{j,MMSE} = (\eta \mathbf{G}_j^H \mathbf{G}_j + \sigma_e^2 \mathbf{I}_L)^{-1} \mathbf{G}_j^H, \tag{4.17}$$

$$SINR_{j,MMSE} = \frac{\frac{1}{L}\sum_{l=1}^{L}[(\mathbf{I}_L + \eta^{-1}\sigma_e^2(\mathbf{G}_j^H\mathbf{G}_j)^{-1})^{-1}](l,l)}{\frac{1}{L}\sum_{l=1}^{L}1 - [(\mathbf{I}_L + \eta^{-1}\sigma_e^2(\mathbf{G}_j^H\mathbf{G}_j)^{-1})^{-1}](l,l)},$$

$$BER_{j,MMSE} \approx \frac{1}{L}\sum_{l=1}^{L}\mathrm{Q}\left\{\frac{[(\mathbf{I}_L + \eta^{-1}\sigma_e^2(\mathbf{G}_j^H\mathbf{G}_j)^{-1})^{-1}](l,l)}{1 - [(\mathbf{I}_L + \eta^{-1}\sigma_e^2(\mathbf{G}_j^H\mathbf{G}_j)^{-1})^{-1}](l,l)}\right\}.$$

The derivation of these formulas is again based on the work of [KKKB96]. Note that the MMSE linear block combiner is the linear block combiner that maximizes the SINR. Hence, we have $SINR_{j,MMSE} \leq SINR_{j,ZF}$, where the equality is obtained if $\sigma_e^2$ goes to zero.

**Remark 4.3.5** Note that calculating the desired user's MMSE linear block combiner according to (4.17) only requires the knowledge of the desired user's channel and the noise variance, as MUI has been eliminated by proper code design. □

## 4.4 Shift-Orthogonal Code Design

In this section, we develop a design strategy for a shift-orthogonal set of code sequences, which makes the MUI-free receiver useful in practice. Note that this design strategy can be considered as one of the main contributions of this chapter.

For $N = d + 1$ and $J = \lfloor (N-1)/2 \rfloor = d/2$ ($d$ power of 2), we show how to design a shift-orthogonal set of $J$ length-$N$ code sequences $\{c_j[n]\}_{j=1}^{J}$ (see definition 4.3.1). Defining the $n \times n$ cyclic shift matrix $\mathbf{J}_n$ as

$$\mathbf{J}_n = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}, \tag{4.18}$$

we first introduce the following definition:

**Definition 4.4.1** A $d \times d/2$ ($d$ power of 2) matrix $\mathbf{C}_d$ is admissible, if and only if $\mathbf{C}_d^H\mathbf{C}_d = d\mathbf{I}_{d/2}$ and $\mathbf{C}_d^H\mathbf{J}_d\mathbf{C}_d = \mathbf{O}_{d/2}$. □

We can then state the following theorem:

**Theorem 4.4.2** *If a $d \times d/2$ ($d$ power of 2) matrix $\mathbf{C}_d$ is admissible, then, for $N = d + 1$ and $J = \lfloor (N-1)/2 \rfloor = d/2$, the following set of $J$ length-$N$ code sequences $\{c_j[n]\}_{j=1}^{J}$ is shift-orthogonal:*

$$c_j[n] = \begin{cases} (1/\sqrt{N})\mathbf{C}_d(N-1, j), & \text{for } n = 0 \\ (1/\sqrt{N})\mathbf{C}_d(n, j), & \text{for } n = 1, 2, \ldots, N-1 \\ 0, & \text{for } n < 0 \text{ and } n \geq N \end{cases},$$
$$\text{for } j = 1, 2, \ldots, J. \qquad (4.19)$$

**Proof:** The proof follows directly from definitions 4.3.1 and 4.4.1, and from the fact that $\mathbf{c}_j^{\uparrow(1)} = (1/\sqrt{N})\mathbf{C}_d(:, j)$ and $\mathbf{c}_j^{\downarrow(1)} = (1/\sqrt{N})\mathbf{J}_d\mathbf{C}_d(:, j)$ □

Hence, it only remains to show how to design an admissible $d \times d/2$ ($d$ power of 2) matrix $\mathbf{C}_d$. Defining the $n \times n$ diagonal matrix $\mathbf{D}_n$ as

$$\mathbf{D}_n = \begin{bmatrix} -1 & 0 & 0 & 0 & \cdots \\ 0 & 1 & 0 & 0 & \cdots \\ 0 & 0 & -1 & 0 & \cdots \\ 0 & 0 & 0 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix},$$

we therefore introduce the following definition:

**Definition 4.4.3** A $d \times d/2$ ($d$ power of 2) matrix $\mathbf{C}_d = \begin{bmatrix} \mathbf{C}_{d,1} & \mathbf{C}_{d,2} \end{bmatrix}$, where $\mathbf{C}_{d,1}$ and $\mathbf{C}_{d,2}$ have size $d \times d/4$, is reducible, if and only if $\mathbf{C}_{d,2} = \mathbf{D}_d\mathbf{C}_{d,1}$. □

This gives rise to the following theorem.

**Theorem 4.4.4** *If a $d \times d/2$ ($d$ power of 2) matrix $\mathbf{C}_d = \begin{bmatrix} \mathbf{C}_{d,1} & \mathbf{C}_{d,2} \end{bmatrix}$, where $\mathbf{C}_{d,1}$ and $\mathbf{C}_{d,2}$ have size $d \times d/4$, is admissible and reducible, then the $2d \times d$ matrix*

$$\mathbf{C}_{2d} = \begin{bmatrix} \mathbf{C}_{2d,1} & \mathbf{C}_{2d,2} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{d,1} & \mathbf{J}_d\mathbf{C}_{d,1} & \mathbf{C}_{d,2} & -\mathbf{J}_d\mathbf{C}_{d,2} \\ \mathbf{C}_{d,2} & -\mathbf{J}_d\mathbf{C}_{d,2} & \mathbf{C}_{d,1} & \mathbf{J}_d\mathbf{C}_{d,1} \end{bmatrix},$$

*where $\mathbf{C}_{2d,1}$ and $\mathbf{C}_{2d,2}$ have size $2d \times d/2$, is also admissible and reducible.*

**Proof:** Assume the $d \times d/2$ ($d$ power of 2) matrix $\mathbf{C}_d = \begin{bmatrix} \mathbf{C}_{d,1} & \mathbf{C}_{d,2} \end{bmatrix}$, where $\mathbf{C}_{d,1}$ and $\mathbf{C}_{d,2}$ have size $d \times d/4$, is admissible and reducible. Using $\mathbf{C}_{d,2} =$

$\mathbf{D}_d\mathbf{C}_{d,1}$, it is easy to check that $\mathbf{C}_{2d,2} = \mathbf{D}_{2d}\mathbf{C}_{2d,1}$. Next, using $\mathbf{C}_d^H\mathbf{C}_d = d\mathbf{I}_{d/2}$ and $\mathbf{C}_d^H\mathbf{J}_d\mathbf{C}_d = \mathbf{O}_{d/2}$, it can be shown by simple calculation that $\mathbf{C}_{2d}^H\mathbf{C}_{2d} = 2d\mathbf{I}_d$. Finally, using $\mathbf{C}_d^H\mathbf{C}_d = d\mathbf{I}_{d/2}$, $\mathbf{C}_d^H\mathbf{J}_d\mathbf{C}_d = \mathbf{O}_{d/2}$ and $\mathbf{C}_{d,2} = \mathbf{D}_d\mathbf{C}_{d,1}$, we now prove that $\mathbf{C}_{2d}^H\mathbf{J}_{2d}\mathbf{C}_{2d} = \mathbf{O}_d$. Defining the $n \times n$ matrix $\mathbf{T}_n$ as

$$\mathbf{T}_n = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix},$$

we first observe that

$$\mathbf{C}_{2d}^H\mathbf{J}_{2d}\mathbf{C}_{2d} = \begin{bmatrix} \mathbf{C}_{d,1}^H & \mathbf{C}_{d,2}^H \\ \mathbf{C}_{d,1}^H\mathbf{J}_d^H & -\mathbf{C}_{d,2}^H\mathbf{J}_d^H \\ \mathbf{C}_{d,2}^H & \mathbf{C}_{d,1}^H \\ -\mathbf{C}_{d,2}^H\mathbf{J}_d^H & \mathbf{C}_{d,1}^H\mathbf{J}_d^H \end{bmatrix} \begin{bmatrix} \mathbf{J}_d - \mathbf{T}_d & \mathbf{T}_d \\ \mathbf{T}_d & \mathbf{J}_d - \mathbf{T}_d \end{bmatrix} \cdot \begin{bmatrix} \mathbf{C}_{d,1} & \mathbf{J}_d\mathbf{C}_{d,1} & \mathbf{C}_{d,2} & -\mathbf{J}_d\mathbf{C}_{d,2} \\ \mathbf{C}_{d,2} & -\mathbf{J}_d\mathbf{C}_{d,2} & \mathbf{C}_{d,1} & \mathbf{J}_d\mathbf{C}_{d,1} \end{bmatrix}.$$

Writing $\mathbf{C}_{2d}^H\mathbf{J}_{2d}\mathbf{C}_{2d}$ as

$$\mathbf{C}_{2d}^H\mathbf{J}_{2d}\mathbf{C}_{2d} = \begin{bmatrix} \mathbf{B}_{1,1} & \mathbf{B}_{1,2} & \mathbf{B}_{1,3} & \mathbf{B}_{1,4} \\ \mathbf{B}_{2,1} & \mathbf{B}_{2,2} & \mathbf{B}_{2,3} & \mathbf{B}_{2,4} \\ \mathbf{B}_{3,1} & \mathbf{B}_{3,2} & \mathbf{B}_{3,3} & \mathbf{B}_{3,4} \\ \mathbf{B}_{4,1} & \mathbf{B}_{4,2} & \mathbf{B}_{4,3} & \mathbf{B}_{4,4} \end{bmatrix},$$

where $\mathbf{B}_{m,n}$ $(m,n = 1,2,3,4)$ is the $d/4 \times d/4$ submatrix of $\mathbf{C}_{2d}^H\mathbf{J}_{2d}\mathbf{C}_{2d}$ at position $(m,n)$, it is then clear that the $d/4 \times d/4$ submatrices $\mathbf{B}_{3,1}$, $\mathbf{B}_{3,2}$, $\mathbf{B}_{3,3}$, $\mathbf{B}_{3,4}$, $\mathbf{B}_{4,1}$, $\mathbf{B}_{4,2}$, $\mathbf{B}_{4,3}$ and $\mathbf{B}_{4,4}$ are respectively equal to the $d/4 \times d/4$ submatrices $\mathbf{B}_{1,3}$, $\mathbf{B}_{1,4}$, $\mathbf{B}_{1,1}$, $\mathbf{B}_{1,2}$, $\mathbf{B}_{2,3}$, $\mathbf{B}_{2,4}$, $\mathbf{B}_{2,1}$ and $\mathbf{B}_{2,2}$. Hence, only the latter have to be calculated:

Submatrix $\mathbf{B}_{1,1}$:

$$\begin{aligned} \mathbf{B}_{1,1} &= \mathbf{C}_{d,1}^H(\mathbf{J}_d - \mathbf{T}_d)\mathbf{C}_{d,1} + \mathbf{C}_{d,2}^H\mathbf{T}_d\mathbf{C}_{d,1} \\ &\quad + \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{C}_{d,2} + \mathbf{C}_{d,2}^H(\mathbf{J}_d - \mathbf{T}_d)\mathbf{C}_{d,2} \\ &= -\mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{C}_{d,1} + \mathbf{C}_{d,1}^H\mathbf{D}_d\mathbf{T}_d\mathbf{C}_{d,1} \\ &\quad + \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{D}_d\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{D}_d\mathbf{T}_d\mathbf{D}_d\mathbf{C}_{d,1} \\ &= -\mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{C}_{d,1} \\ &\quad + \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{C}_{d,1} + \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{C}_{d,1} = \mathbf{O}_{d/4}. \end{aligned}$$

Submatrix $\mathbf{B}_{1,2}$:

$$\begin{aligned}\mathbf{B}_{1,2} &= \mathbf{C}_{d,1}^H(\mathbf{J}_d - \mathbf{T}_d)\mathbf{J}_d\mathbf{C}_{d,1} + \mathbf{C}_{d,2}^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} \\ &\quad - \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,2} - \mathbf{C}_{d,2}^H(\mathbf{J}_d - \mathbf{T}_d)\mathbf{J}_d\mathbf{C}_{d,2} \\ &= \mathbf{C}_{d,1}^H\mathbf{J}_d^2\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} + \mathbf{C}_{d,1}^H\mathbf{D}_d\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} \\ &\quad - \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{J}_d\mathbf{D}_d\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{D}_d\mathbf{J}_d^2\mathbf{D}_d\mathbf{C}_{d,1} + \mathbf{C}_{d,1}^H\mathbf{D}_d\mathbf{T}_d\mathbf{J}_d\mathbf{D}_d\mathbf{C}_{d,1} \\ &= \mathbf{C}_{d,1}^H\mathbf{J}_d^2\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} \\ &\quad + \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{J}_d^2\mathbf{C}_{d,1} + \mathbf{C}_{d,1}^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} = \mathbf{O}_{d/4}.\end{aligned}$$

Submatrix $\mathbf{B}_{1,3}$: similar calculation as for $\mathbf{B}_{1,1}$.
Submatrix $\mathbf{B}_{1,4}$: similar calculation as for $\mathbf{B}_{1,2}$.
Submatrix $\mathbf{B}_{2,1}$:

$$\begin{aligned}\mathbf{B}_{2,1} &= \mathbf{C}_{d,1}^H\mathbf{J}_d^H(\mathbf{J}_d - \mathbf{T}_d)\mathbf{C}_{d,1} - \mathbf{C}_{d,2}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{C}_{d,1} \\ &\quad + \mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{C}_{d,2} - \mathbf{C}_{d,2}^H\mathbf{J}_d^H(\mathbf{J}_d - \mathbf{T}_d)\mathbf{C}_{d,2} \\ &= d\mathbf{I}_{d/4} - \mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{D}_d\mathbf{J}_d^H\mathbf{T}_d\mathbf{C}_{d,1} \\ &\quad + \mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{D}_d\mathbf{C}_{d,1} - d\mathbf{I}_{d/4} + \mathbf{C}_{d,1}^H\mathbf{D}_d\mathbf{J}_d^H\mathbf{T}_d\mathbf{D}_d\mathbf{C}_{d,1} \\ &= -\mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{C}_{d,1} \\ &\quad + \mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{C}_{d,1} + \mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{C}_{d,1} = \mathbf{O}_{d/4}.\end{aligned}$$

Submatrix $\mathbf{B}_{2,2}$:

$$\begin{aligned}\mathbf{B}_{2,2} &= \mathbf{C}_{d,1}^H\mathbf{J}_d^H(\mathbf{J}_d - \mathbf{T}_d)\mathbf{J}_d\mathbf{C}_{d,1} - \mathbf{C}_{d,2}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} \\ &\quad - \mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,2} + \mathbf{C}_{d,2}^H\mathbf{J}_d^H(\mathbf{J}_d - \mathbf{T}_d)\mathbf{J}_d\mathbf{C}_{d,2} \\ &= -\mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{D}_d\mathbf{J}_d^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} \\ &\quad - \mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{J}_d\mathbf{D}_d\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{D}_d\mathbf{J}_d^H\mathbf{T}_d\mathbf{J}_d\mathbf{D}_d\mathbf{C}_{d,1} \\ &= -\mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} - \mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} \\ &\quad + \mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} + \mathbf{C}_{d,1}^H\mathbf{J}_d^H\mathbf{T}_d\mathbf{J}_d\mathbf{C}_{d,1} = \mathbf{O}_{d/4}.\end{aligned}$$

Submatrix $\mathbf{B}_{2,3}$: similar calculation as for $\mathbf{B}_{2,1}$.
Submatrix $\mathbf{B}_{2,4}$: similar calculation as for $\mathbf{B}_{2,2}$.

From these calculations it is clear that $\mathbf{C}_{2d}^H\mathbf{J}_{2d}\mathbf{C}_{2d} = \mathbf{O}_d$, which concludes the proof. □

Based on this theorem we can design an admissible and reducible $d \times d/2$ ($d$

power of 2) matrix $\mathbf{C}_d$ in a recursive way, starting for instance with

$$\mathbf{C}_4 = \begin{bmatrix} +1 & -1 \\ +1 & +1 \\ -1 & +1 \\ +1 & +1 \end{bmatrix} \quad \text{or} \quad \mathbf{C}_4 = \frac{1}{\sqrt{2}} \begin{bmatrix} +1-i & -1+i \\ +1+i & +1+i \\ -1+i & +1-i \\ +1+i & +1+i \end{bmatrix},$$

resulting into a set of BPSK or QPSK code sequences, respectively. Observe from theorem 4.4.2 that the $J$ length-$N$ code sequences, defined in (4.19), are actually based on $J$ length-$(N-1)$ code sequences, extended with a cyclic prefix of 1 code symbol ($c_j[0] = c_j[N-1]$). The insertion of a cyclic prefix is a well known procedure in DMT systems [Bin90].

## 4.5 Blind Channel Estimation

Within the framework of the MUI-free receiver, we present a subspace deterministic blind single-user channel estimation algorithm. This algorithm is performed on the model (4.9). It is labeled as *subspace*, since it is based on a subspace decomposition. We first describe how the algorithm works and then give a performance analysis. We finally show how the channel gain can be estimated. Note that, in a different context, a similar approach is presented in [TG97]. However, no performance analysis or channel gain estimation is presented there.

### 4.5.1 Algorithm

In this section, we describe how the algorithm works [TG97]. For a burst length of $K$ (multiple of $L$), we first define the $2L \times K/L$ matrix $\mathbf{Z}_j$ as

$$\mathbf{Z}_j = \begin{bmatrix} \mathbf{z}_j[0] & \mathbf{z}_j[1] & \cdots & \mathbf{z}_j[K/L-1] \end{bmatrix}.$$

Using (4.9) we then obtain

$$\mathbf{Z}_j = \eta \mathbf{G}_j \mathbf{S}_j + \mathbf{N}_j, \tag{4.20}$$

where $\mathbf{N}_j$ is similarly defined as $\mathbf{Z}_j$ and $\mathbf{S}_j$ is the $L \times K/L$ matrix given by

$$\mathbf{S}_j = \begin{bmatrix} \mathbf{s}_j[0] & \mathbf{s}_j[1] & \cdots & \mathbf{s}_j[K/L-1] \end{bmatrix}.$$

Because of the specific structure of $\mathbf{G}_j$ (see (4.10)), we have the following property:

**Property 4.5.1** The $2L \times L$ channel matrix $\mathbf{G}_j$ has full column rank $L$ and has more rows than columns. □

From this property, it is clear that we do not have to make an assumption similar to assumption 3.4.1, to develop the algorithm. Further, we introduce a condition on the input:

**Assumption 4.5.2** The $L \times K/L$ matrix $\mathbf{S}_j$ has full row rank $L$. □

This assumption requires that

$$K/L \geq L.$$

Let us first, for the sake of clarity, assume that there is no additive noise present in $\mathbf{Z}_j$. Calculating the singular value decomposition (SVD) of $\mathbf{Z}_j$ then leads to

$$\mathbf{Z}_j = \mathbf{U}_j \mathbf{\Sigma}_j \mathbf{V}_j^H,$$

where $\mathbf{\Sigma}_j$ is a diagonal matrix (diagonal elements in descending order) of the same size as $\mathbf{Z}_j$ and $\mathbf{U}_j$ and $\mathbf{V}_j$ are square unitary matrices. Because of property 4.5.1 and assumption 4.5.2, the $2L \times K/L$ matrix $\mathbf{Z}_j$ has rank $L$ and, defining the $2L \times L$ matrix $\mathbf{U}_j^n$ as the collection of the $L$ last columns of $\mathbf{U}_j$ (because of property 4.5.1, $\mathbf{U}_j^n$ is not empty), the columns of $\mathbf{U}_j^n$ form an orthonormal basis of the left null space of $\mathbf{G}_j$:

$$\mathbf{U}_j^{nH} \mathbf{G}_j = \mathbf{O}_L.$$

This equation can be rewritten as

$$\mathcal{U}_j^{nH} \mathbf{g}_j = \mathbf{0}, \tag{4.21}$$

where $\mathbf{g}_j$ is defined in (2.12) and $\mathcal{U}_j^n$ is the $L \times L^2$ matrix, given by

$$\mathcal{U}_j^n = \begin{bmatrix} \mathbf{U}_{j,1}^n & \mathbf{U}_{j,2}^n & \cdots & \mathbf{U}_{j,L}^n \end{bmatrix},$$

with $\mathbf{U}_{j,l}^n$ $(l = 1, 2, \ldots, L)$ the $L \times L$ matrix, given by

$$\mathbf{U}_{j,l}^n = \mathbf{U}_j^n(l : l + L - 1, :).$$

This equation may allow us to identify $g_j[n]$.

We have the following identifiability result [TG97]:

**Theorem 4.5.3** *Under assumption* (4.5.2)*, the channel* $g_j[n]$ *can be uniquely (up to a complex scaling factor) determined from* (4.21).

**Proof:** See [TG97]. □

Let us now assume that additive noise is present in $\mathbf{Z}_j$. Calculating the SVD of $\mathbf{Z}_j$ then leads to

$$\mathbf{Z}_j = \hat{\mathbf{U}}_j \hat{\mathbf{\Sigma}}_j \hat{\mathbf{V}}_j^H,$$

where $\hat{\mathbf{\Sigma}}_j$ is a diagonal matrix (diagonal elements in descending order) of the same size as $\mathbf{Z}_j$ and $\hat{\mathbf{U}}_j$ and $\hat{\mathbf{V}}_j$ are square unitary matrices. Defining the $2L \times L$ matrix $\hat{\mathbf{U}}_j^n$ as the collection of the $L$ last columns of $\hat{\mathbf{U}}_j$, we then consider the following minimization problem:

$$\boxed{\begin{array}{c} \min_{\mathbf{g}_j} \{ \| \hat{\mathcal{U}}_j^{nH} \mathbf{g}_j \|^2 \}, \\ \text{s.t. a non-triviality constraint imposed on } \mathbf{g}_j, \end{array}}$$

where $\hat{\mathcal{U}}_j^n$ is the $L \times L^2$ matrix, given by

$$\hat{\mathcal{U}}_j^n = \begin{bmatrix} \hat{\mathbf{U}}_{j,1}^n & \hat{\mathbf{U}}_{j,2}^n & \cdots & \hat{\mathbf{U}}_{j,L}^n \end{bmatrix},$$

with $\hat{\mathbf{U}}_{j,l}^n$ $(l = 1, 2, \ldots, L)$ the $L \times L$ matrix, given by

$$\hat{\mathbf{U}}_{j,l}^n = \hat{\mathbf{U}}_j^n (l : l + L - 1, :).$$

If we impose a unit norm constraint on $\mathbf{g}_j$ (this constraint is mostly used), then the left singular vector of $\hat{\mathcal{U}}_j^n$ corresponding to the smallest singular value represents a possible solution. Since this vector can be interpreted as an estimate of $\mathbf{g}_j^o$, the left singular vector of $\mathcal{U}_j^n$ corresponding to the smallest singular value, we will denote this solution as $\hat{\mathbf{g}}_j^o$. As an estimate for the channel vector $\mathbf{g}_j$ we then consider

$$\hat{\mathbf{g}}_j = \hat{\gamma}_j \hat{\mathbf{g}}_j^o, \tag{4.22}$$

where $\hat{\gamma}_j$ is an estimate of $\gamma_j$, which is given by $\mathbf{g}_j = \gamma_j \mathbf{g}_j^o$. Like in the previous chapter, this $\gamma_j$ can be estimated from some short known headers that are transmitted, or we can blindly estimate $|\gamma_j|$, which is equal to the channel gain $\|\mathbf{g}_j\|$ (see section 4.5.3), and use an appropriate differential modulation scheme to get rid of the phase ambiguity [Pro89]. However, for simplicity we will estimate $\gamma_j$ as

$$\hat{\gamma}_j = \mathbf{g}_j \hat{\mathbf{g}}_j^{o\dagger}.$$

This leads to an estimate $\hat{\mathbf{g}}_j$ that is optimal in the LS sense (since $\mathbf{g}_j$ is unknown, this is of course not feasible in practice).

1. compute the SVD of $\mathbf{Z}_j$:
$$\mathbf{Z}_j = \hat{\mathbf{U}}_j \hat{\mathbf{\Sigma}}_j \hat{\mathbf{V}}_j^H$$
2. collect the last $L$ columns of $\hat{\mathbf{U}}_j$: $\hat{\mathbf{U}}_j^n$
3. for $l = 1, 2, \ldots, L$
   - construct $\hat{\mathbf{U}}_{j,l}^n$ as
$$\hat{\mathbf{U}}_{j,l}^n = \hat{\mathbf{U}}_j^n(l : l + L - 1, :)$$
4. construct $\hat{\mathcal{U}}_j^n$ as
$$\hat{\mathcal{U}}_j^n = \begin{bmatrix} \hat{\mathbf{U}}_{j,1}^n & \hat{\mathbf{U}}_{j,2}^n & \cdots & \hat{\mathbf{U}}_{j,L}^n \end{bmatrix}$$
5. solve
$$\min_{\mathbf{g}_j} \{ \| \mathcal{U}_j^{nH} \mathbf{g}_j \|^2 \}, \quad \text{s.t. } \|\mathbf{g}_j\|^2 = 1$$

Table 4.1: Blind single-user channel estimation algorithm.

**Remark 4.5.4** Note that the estimation of the desired user's channel only requires the desired user's code sequence, namely through the fact that this algorithm makes use of the data at the output of the bank of 2 modified block correlators. □

A brief outline of the algorithm is given in **table 4.1**.

### 4.5.2 Performance Analysis

The performance analysis is given in the following theorem.

**Theorem 4.5.5** *Assume that the channel vector estimate $\hat{\mathbf{g}}_j$ is obtained as in (4.22), with $\hat{\gamma}_j = \gamma_j$. Further assume that the data symbol sequence $s_j[k]$ is zero-mean white with variance 1 and the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$. Then, only considering the first order approximation of $\Delta\mathbf{g}_j = \hat{\mathbf{g}}_j - \mathbf{g}_j$ in $\mathbf{N}_j$, the bias and the normalized MSE (NMSE) of $\hat{\mathbf{g}}_j$ can be expressed as*

$$bias_j = \mathbf{0},$$
$$NMSE_j \approx \frac{L\sigma_e^2}{K\eta\|\mathbf{g}_j\|^2} \left\| \mathcal{U}_j^{n\dagger H} \right\|^2.$$

**Proof:** Using a result from subspace perturbation analysis [LLV93], the first order approximation of $\Delta\mathbf{U}_j^n = \hat{\mathbf{U}}_j^n - \mathbf{U}_j^n$ in $\mathbf{N}_j$ can be written as

$$\Delta^{(1)}\mathbf{U}_j^n = -\mathbf{Z}_j^{\dagger H}\mathbf{N}_j^H\mathbf{U}_j^n.$$

The first order approximation of $\Delta\mathbf{U}_{j,l}^n = \hat{\mathbf{U}}_{j,l}^n - \mathbf{U}_{j,l}^n$ $(l = 1, 2, \ldots, L)$ in $\mathbf{N}_j$ is therefore given by

$$\Delta^{(1)}\mathbf{U}_{j,l}^n = -\mathbf{Z}_{j,l}^{\dagger H}\mathbf{N}_j^H\mathbf{U}_j^n,$$

where

$$\mathbf{Z}_{j,l}^{\dagger} = \mathbf{Z}_j^{\dagger}(:, l : l + L - 1).$$

Thus the first order approximation of $\Delta\mathcal{U}_j^n = \hat{\mathcal{U}}_j^n - \mathcal{U}_j^n$ in $\mathbf{N}_j$ is

$$\begin{aligned}\Delta^{(1)}\mathcal{U}_j^n &= \begin{bmatrix}\Delta^{(1)}\mathbf{U}_{j,1}^n & \cdots & \Delta^{(1)}\mathbf{U}_{j,L}^n\end{bmatrix} \\ &= -\begin{bmatrix}\mathbf{Z}_{j,1}^{\dagger H}\mathbf{N}_j^H\mathbf{U}_j^n & \cdots & \mathbf{Z}_{j,L}^{\dagger H}\mathbf{N}_j^H\mathbf{U}_j^n\end{bmatrix}.\end{aligned}$$

If $\hat{\mathbf{g}}_j$ is obtained as in (4.22), with $\hat{\gamma}_j = \gamma_j$, then we can use the same result from subspace perturbation analysis as before to determine the first order approximation of $\Delta\mathbf{g}_j = \hat{\mathbf{g}}_j - \mathbf{g}_j$ in $\mathbf{N}_j$:

$$\begin{aligned}\Delta^{(1)}\mathbf{g}_j &= -\mathcal{U}_j^{n\dagger H}\Delta^{(1)}\mathcal{U}_j^{nH}\mathbf{g}_j \\ &= \mathcal{U}_j^{n\dagger H}\begin{bmatrix}\mathbf{U}_j^{nH}\mathbf{N}_j\mathbf{Z}_{j,1}^{\dagger} \\ \vdots \\ \mathbf{U}_j^{nH}\mathbf{N}_j\mathbf{Z}_{j,L}^{\dagger}\end{bmatrix}\mathbf{g}_j \\ &= \begin{bmatrix}\mathcal{U}_{j,1}^{n\dagger H}\mathbf{U}_j^{nH} & \cdots & \mathcal{U}_{j,L}^{n\dagger H}\mathbf{U}_j^{nH}\end{bmatrix}\begin{bmatrix}\mathbf{N}_j & & \\ & \ddots & \\ & & \mathbf{N}_j\end{bmatrix}\begin{bmatrix}\mathbf{Z}_{j,1}^{\dagger}\mathbf{g}_j \\ \vdots \\ \mathbf{Z}_{j,L}^{\dagger}\mathbf{g}_j\end{bmatrix},\end{aligned}$$

where $\mathcal{U}_{j,l}^{n\dagger}$ $(l = 1, 2, \ldots, L)$ is the $L \times L$ matrix, given by

$$\mathcal{U}_{j,l}^{n\dagger} = \mathcal{U}_j^{n\dagger}((l-1)L + 1 : lL, :).$$

Only considering the first order approximation of $\Delta\mathbf{g}_j$ in $\mathbf{N}_j$, the bias and the MSE of $\hat{\mathbf{g}}_j$ can be expressed as

$$bias_j = \mathrm{E}\{\Delta^{(1)}\mathbf{g}_j\} \quad \text{and} \quad MSE_j = \mathrm{E}\{\Delta^{(1)}\mathbf{g}_j^H\Delta^{(1)}\mathbf{g}_j\}.$$

Assume that the data symbol sequence $s_j[k]$ is zero-mean white with variance 1 and the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$. The bias of $\hat{\mathbf{g}}_j$ then becomes

$$bias_j = \mathrm{E}\{\Delta^{(1)}\mathbf{g}_j\} = \mathbf{0}.$$

The MSE of $\hat{\mathbf{g}}_j$ then becomes

$$\begin{aligned} MSE_j &= \mathrm{E}\{\Delta^{(1)}\mathbf{g}_j^H\Delta^{(1)}\mathbf{g}_j\} \\ &= \eta\sigma_e^2\sum_{l=1}^{L}\Big(\|\mathcal{U}_{j,l}^{n\dagger H}\|^2\mathrm{E}\{\|\mathbf{Z}_{j,l}^{\dagger}\mathbf{g}_j\|^2\} \\ &\quad + \sum_{\substack{l'=1\\ l'\neq l}}^{L}\mathrm{tr}\{\mathcal{U}_{j,l}^{n\dagger H}\mathcal{U}_{j,l'}^{n\dagger}\}\mathrm{E}\{\mathbf{g}_j^H\mathbf{Z}_{j,l}^{\dagger H}\mathbf{Z}_{j,l'}^{\dagger}\mathbf{g}_j\}\Big). \end{aligned}$$

Following a similar reasoning as in [LLV93], we can further prove that

$$\begin{cases} \mathrm{E}\{\|\mathbf{Z}_{j,l}^{\dagger}\mathbf{g}_j\|^2\} \approx \dfrac{L}{K\eta^2}, & \text{for } l = 1, 2, \ldots, L \\ \mathrm{E}\{\mathbf{g}_j^H\mathbf{Z}_{j,l}^{\dagger H}\mathbf{Z}_{j,l'}^{\dagger}\mathbf{g}_j\} \approx 0, & \text{for } l, l' = 1, 2, \ldots, L, \text{ with } l \neq l' \end{cases}.$$

Therefore, the NMSE of $\hat{\mathbf{g}}_j$ can be written as

$$NMSE_j \approx \frac{L\sigma_e^2}{K\eta\|\mathbf{g}_j\|^2}\sum_{l=1}^{L}\|\mathcal{U}_{j,l}^{n\dagger H}\|^2 = \frac{L\sigma_e^2}{K\eta\|\mathbf{g}_j\|^2}\|\mathcal{U}_j^{n\dagger H}\|^2. \qquad \square$$

### 4.5.3 Channel Gain Estimation

In this section, we show how to estimate $|\gamma_j|$, which is equal to the channel gain $\|\mathbf{g}_j\|$, from $\mathbf{Z}_j$. Assume that the data symbol sequence $s_j[k]$ is zero-mean white with variance 1 ($\mathbf{R}_{\mathbf{s}_j} = \mathbf{I}_L$) and the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$ ($\mathbf{R}_{\mathbf{n}_j} = \eta\sigma_e^2\mathbf{I}_{2L}$). Then, we can write (4.11) as

$$\mathbf{R}_{\mathbf{z}_j} = \mathrm{E}\{\mathbf{z}_j[k]\mathbf{z}_j^H[k]\} = \eta^2\mathbf{G}_j\mathbf{G}_j^H + \eta\sigma_e^2\mathbf{I}_{2L} = \eta^2|\gamma_j|^2\mathbf{G}_j^o\mathbf{G}_j^{oH} + \eta\sigma_e^2\mathbf{I}_{2L},$$

where $\mathbf{G}_j^o$ is similarly defined as $\mathbf{G}_j$, using $\mathbf{g}_j^o$ instead of $\mathbf{g}_j$. Hence, we can write

$$|\gamma_j|^2 = \frac{1}{L}\mathrm{tr}\{\mathbf{F}_{j,ZF}^o(\mathbf{R}_{\mathbf{z}_j} - \eta\sigma_e^2\mathbf{I}_{2L})\mathbf{F}_{j,ZF}^{oH}\},$$

where $\mathbf{F}_{j,ZF}^o$ is similarly defined as $\mathbf{F}_{j,ZF}$ (see (4.15)), using $\mathbf{G}_j^o$ instead of $\mathbf{G}_j$. This leads to the following estimate of $|\gamma_j|^2$:

$$|\hat{\gamma}_j|^2 = \frac{1}{L}\mathrm{tr}\{\hat{\mathbf{F}}_{j,ZF}^o(\hat{\mathbf{R}}_{\mathbf{z}_j} - \eta\hat{\sigma}_e^2\mathbf{I}_{2L})\hat{\mathbf{F}}_{j,ZF}^{oH}\}, \tag{4.23}$$

where $\hat{\mathbf{F}}_{j,ZF}^o$ is similarly defined as $\mathbf{F}_{j,ZF}^o$, using $\hat{\mathbf{G}}_j^o$ instead of $\mathbf{G}_j^o$, with $\hat{\mathbf{G}}_j^o$ similarly defined as $\mathbf{G}_j^o$, using $\hat{\mathbf{g}}_j^o$ instead of $\mathbf{g}_j^o$. Further, $\hat{\mathbf{R}}_{\mathbf{z}_j}$ is defined as

$$\hat{\mathbf{R}}_{\mathbf{z}_j} = \frac{L}{K}\mathbf{Z}_j\mathbf{Z}_j^H$$

and $\hat{\sigma}_e^2$ is for instance obtained as $\eta^{-1}$ times the average of the $L$ smallest eigenvalues of $\hat{\mathbf{R}}_{\mathbf{z}_j}$. Note that this estimate of the noise variance can also be used in the MMSE linear block combiner, when the noise variance $\sigma_e^2$ is unknown.

## 4.6 Conclusions

In this chapter, we have developed new receivers for a DS-CDMA-BS system. First, the block RAKE receiver has been introduced. Then, we have introduced the MUI-free receiver, which completely removes the MUI, without using any channel information. Within the framework of the MUI-free receiver, we have further presented a subspace deterministic blind single-user channel estimation algorithm. Complexity issues and simulation results will be presented in chapter 5.

# Chapter 5

# Complexity and Simulation Results

In this chapter, we compare the new MUI-free receiver, developed for a DS-CDMA-BS system, with the existing linear multi-user equalizer, developed for a DS-CDMA system. In section 5.1, we compare the complexity of the MUI-free receiver with the complexity of the linear multi-user equalizer. To make this comparison, we assume that the receiver design is based on the appropriate subspace deterministic blind channel estimation algorithm, although the same conclusions can be drawn for other types of receiver design. In section 5.2, we compare the performance of the MUI-free receiver with ZF respectively MMSE linear block combining with the performance of the ZF respectively MMSE linear multi-user equalizer. We give a performance comparison for the case the channels and the noise variance (necessary for both MMSE receivers) are known as well as for the case the channels are estimated using the appropriate subspace deterministic blind channel estimation algorithm and the noise variance is estimated as in the corresponding blind channel gain estimation algorithm. In this chapter, we also compare the MUI-free receiver with the corresponding FFT-based Vandermonde-Lagrange AMOUR (VL-AMOUR) transceiver [WSGB99]. This is done in section 5.3. Conclusions are drawn in section 5.4.

## 5.1 Complexity

In this section, we compare the complexity of the MUI-free receiver (section 4.3) with the complexity of the linear multi-user equalizer (section 3.3). We assume that the receiver design is based on subspace deterministic blind channel es-

timation (sections 4.5 and 3.4). Note that with complexity we mean the required number of *flops* (floating point operations). All the complexity measures we present in this section are based on any standard method that computes the short singular vectors of a matrix (see for instance [GVL89]). As in the previous two chapters, we consider a single receive antenna ICI-limited quasi-synchronous propagation model (see (2.10)). We assume that the burst length $K$ is very large.

Let us first focus on the MUI-free receiver. The design of the $j$th user's MUI-free receiver is determined by the design of the $j$th user's linear block combiner. From (4.15) and (4.17), it is clear that the design of the $j$th user's linear block combiner only requires the knowledge of the $j$th user's channel and the noise variance (if an MMSE linear block combiner is considered). To estimate the $j$th user's channel we use the subspace deterministic blind single-user channel estimation algorithm of section 4.5, which only requires the $j$th user's code sequence, namely through the fact that this algorithm makes use of the data at the output of the bank of 2 modified block correlators. For this, we first have to compute the subspace decomposition of the $2L \times K/L$ matrix $\mathbf{Z}_j$, which has a complexity of $\mathrm{O}\{LK\}$ flops. Then we have to compute the left singular vector of the $L \times L^2$ matrix $\hat{\mathcal{U}}_j^n$ corresponding to the smallest singular value, which has a complexity of $\mathrm{O}\{L^4\}$ flops. Hence, the complexity to estimate the $j$th user's channel is $\mathrm{O}\{LK\}$ flops (since $K$ is very large). Next, we have to calculate the pseudo-inverse of the $2L \times L$ matrix $\mathbf{G}_j$ (we focus on a ZF linear block combiner), which results in a complexity of $\mathrm{O}\{L^3\}$ flops. Note that the estimation of the noise variance, necessary to design an MMSE linear block combiner, does not increase this complexity very much. To conclude, the design of the $j$th user's linear block combiner has a complexity of $\mathrm{O}\{LK\}$ flops (since $K$ is very large). Hence, the design of the $j$th user's MUI-free receiver also has a complexity of $\mathrm{O}\{LK\}$ flops. Note that the design of a complete bank of $J$ MUI-free receivers (e.g., at a base station) results in a complexity of $\mathrm{O}\{JLK\}$ flops.

Let us next focus on the linear multi-user equalizer. From (3.8) and (3.10), it is clear that the design of the $j$th user's linear multi-user equalizer requires the knowledge of all the channels and code sequences and the additive noise (if an MMSE linear multi-user equalizer is considered). To estimate the $j$th user's channel, we use the subspace deterministic blind multi-user channel estimation algorithm of section 3.4, which only requires the $j$th user's code sequence. For this, we first have to compute a subspace decomposition of the $(N - L + 1) \times K$ matrix $\mathbf{Y}$, which has a complexity of $\mathrm{O}\{N^2K\}$ flops. Note that this subspace decomposition is user-independent and therefore still results into a complexity of $\mathrm{O}\{N^2K\}$ flops, if we want to estimate all the channels. Then we have to compute the left singular vector of the $L \times (N - L + 1 - J)$ matrix $\mathbf{C}_j^H \hat{\mathbf{U}}^n$ corresponding to the smallest singular value, which has a complexity of $\mathrm{O}\{L^2(N - L + 1 - J)\}$ flops. Note that this subspace decomposition is

user-dependent and therefore results into a complexity of $\mathrm{O}\{JL^2(N-L+1-J)\}$ flops, if we want to estimate all the channels. Hence, the complexity to estimate all the channels is $\mathrm{O}\{N^2K\}$ flops (since $K$ is very large). Next, we have to calculate the pseudo-inverse of the $(N-L+1)\times J$ matrix $\mathbf{H}$ (we focus on a ZF linear multi-user equalizer), which results in a complexity of $\mathrm{O}\{NJ^2\}$ flops. Note that the estimation of the noise variance, necessary to design an MMSE linear multi-user equalizer, does not increase this complexity very much. To conclude, the design of the $j$th user's linear multi-user equalizer has a complexity of $\mathrm{O}\{N^2K\}$ flops (since $K$ is very large). Note that the design of a complete bank of $J$ multi-user equalizers (e.g., at a base station) still results in a complexity of $\mathrm{O}\{N^2K\}$ flops.

**Remark 5.1.1** Note that even if we follow the approach discussed in remarks 3.3.2 and 3.3.3, it is not possible to reduce this complexity of $\mathrm{O}\{N^2K\}$ flops. □

Comparing the obtained complexities, we observe that the complexity to design an MUI-free receiver ($\mathcal{O}\{LK\}$ flops) is *much smaller* than the complexity to design a linear multi-user equalizer ($\mathrm{O}\{N^2K\}$ flops). This is because $L \ll N$ (see (2.10)). Moreover, the complexity to design a complete bank of $J$ MUI-free receivers ($\mathrm{O}\{JLK\}$ flops) is also much smaller than the complexity to design a complete bank of $J$ linear multi-user equalizers ($\mathrm{O}\{N^2K\}$ flops), although the latter is of the same order as the complexity to design only one linear multi-user equalizer. This is because $J$ is of the same order as $N$ and $L \ll N$ (see (2.10)).

Finally, note that the estimation of the desired user's data symbol sequence based on the calculated MUI-free receiver or linear multi-user equalizer results in a complexity per data symbol period of $\mathrm{O}\{N\}$ flops (this complexity is more or less the same for both receivers).

## 5.2 Simulation Results

In this section, we perform some simulations in Matlab to compare the performance of the MUI-free receiver with ZF respectively MMSE linear block combining (section 4.3) with the performance of the ZF respectively MMSE linear multi-user equalizer (section 3.3). We consider an 8-user DS-CDMA-BS and DS-CDMA system ($J=8$) with a spreading factor $N=17$. We design a shift-orthogonal set of $J$ length-$N$ code sequences $\{c_j[n]\}_{j=1}^{J}$, as explained in section 4.4, and use this set of code sequences for the DS-CDMA-BS system as

| $n$ | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| $g_1[n]$ | $+0.1513 - 0.1291i$ | $-0.1327 + 0.6884i$ | $-0.3550 - 0.1626i$ | $+0.1228 + 0.5490i$ |
| $g_2[n]$ | $-0.2588 + 0.4124i$ | $+0.3581 - 0.0455i$ | $-0.4241 + 0.5462i$ | $-0.3760 - 0.1147i$ |
| $g_3[n]$ | $+0.4066 - 0.5422i$ | $+0.5321 + 0.2291i$ | $+0.4098 + 0.1556i$ | $+0.1024 - 0.0487i$ |
| $g_4[n]$ | $-0.2163 + 0.2912i$ | $-0.5332 - 0.4866i$ | $-0.0811 - 0.4955i$ | $+0.0781 + 0.2985i$ |
| $g_5[n]$ | $-0.3083 + 0.0279i$ | $+0.8654 + 0.0949i$ | $-0.0394 - 0.0426i$ | $+0.0033 - 0.3779i$ |
| $g_6[n]$ | $-0.0638 - 0.5494i$ | $+0.2137 + 0.3858i$ | $+0.2448 + 0.5350i$ | $+0.1602 + 0.3574i$ |
| $g_7[n]$ | $+0.1898 - 0.1522i$ | $+0.1045 - 0.1316i$ | $+0.3813 - 0.3089i$ | $-0.5015 - 0.6482i$ |
| $g_8[n]$ | $-0.0609 + 0.7390i$ | $-0.5202 + 0.0305i$ | $-0.1328 - 0.3271i$ | $-0.1992 - 0.1195i$ |

Table 5.1: Normalized channels.

well as for the DS-CDMA system. We start from (see section 4.4 for details)

$$\mathbf{C}_4 = \begin{bmatrix} +1 & -1 \\ +1 & +1 \\ -1 & +1 \\ +1 & +1 \end{bmatrix},$$

resulting into a set of BPSK code sequences. For every user $j$, we generate a random channel $g_j[n]$ of order $L_j = 3$ with delay index $\delta_j = 0$. We assume $L = 4$ (note that (2.10) is then satisfied). The resulting set of normalized channels is listed in **table 5.1**. We assume that the data symbol sequences $\{s_j[k]\}_{j=1}^{J}$ are QPSK-modulated mutually uncorrelated and zero-mean white with variance 1. We further assume that the additive noise $e[n]$ is zero-mean white Gaussian with variance $\sigma_e^2$. We define the received energy per data symbol period ($T$) for the $j$th user as

$$E_j = \|\mathbf{g}_j\|^2$$

and we assume that all users interfering with this $j$th user have the same received energy per data symbol period:

$$E_{j'} = E, \quad \text{for } j' = 1, \ldots, j-1, j+1, \ldots, J.$$

We then define the signal-to-noise ratio (SNR) and the near-far ratio (NFR) for the $j$th user at the input of the receiver as

$$SNR_j = \frac{E_j}{\sigma_e^2} \quad \text{and} \quad NFR_j = \frac{E}{E_j}.$$

For all simulations we will conduct 5000 trials using bursts of $K = 200$ data symbols.

*Simulation 1:* First, we compare the performance of the MUI-free receiver with ZF respectively MMSE linear block combining with the performance of the ZF respectively MMSE linear multi-user equalizer, under the assumption that all

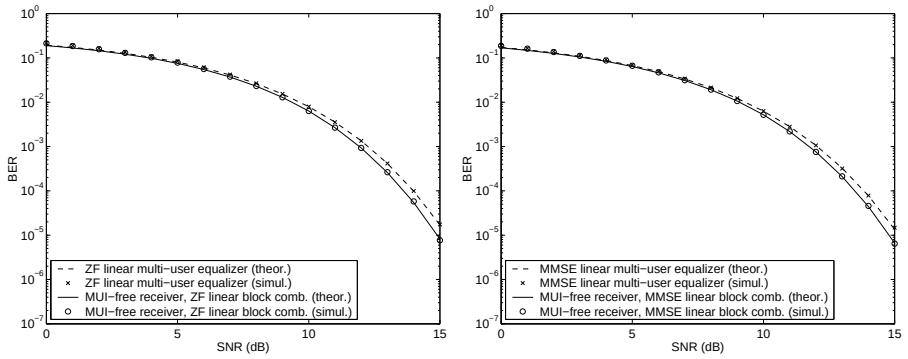



Figure 5.1: Average theoretical and simulated BER per user as a function of the SNR for an NFR of 0 dB (known channels and noise variance).

the channels $\{g_j[n]\}_{j=1}^{J}$ and the noise variance $\sigma_e^2$ (necessary for both MMSE receivers) are known. **Figure 5.1** shows the average theoretical and simulated BER per user as a function of the SNR for an NFR of 0 dB. **Figure 5.2** shows the same results as a function of the NFR for an SNR of 10 dB. First of all, we observe that the simulation results are well predicted by the theoretical results. As mentioned in sections 4.3 and 3.3, the theoretical predictions are based on the assumption that the interference plus noise at the output of the receiver is Gaussian. If the interference at the output of the receiver is different from zero it will not be exactly Gaussian. Since the ZF receivers remove all the received interference, their performance is exactly predicted. The MMSE receivers, on the other hand, do not remove all the received interference and their performance is not exactly predicted. However, for this particular example, the interference at the output of the MMSE receivers is almost zero and the difference between the simulation results and the theoretical results is hardly visible. Second, we know from section 4.3 that an MUI-free receiver should always be NFR-independent[1], irrespective of the type of linear block combining. This is clearly illustrated in figure 5.2. From this figure it is also clear that the ZF linear multi-user equalizer is NFR-independent, while the MMSE linear multi-user equalizer is not. For a very high NFR the performance of the MMSE linear multi-user equalizer is equal to the performance of the ZF linear multi-user equalizer. For a very low NFR the performance of the MMSE linear multi-user equalizer approaches the performance of the coherent RAKE receiver, which is, for such a very low NFR, better than the performance of the ZF linear multi-user equalizer. Note that the NFR-region where the transition

[1] Note that a near-far resistant receiver (see [Ver86]) is not exactly the same as an NFR-independent receiver. NFR-independence implies near-far resistance, since near-far resistance can be interpreted as NFR-independence for vanishing noise. However, near-far resistance does not necessarily imply NFR-independence.

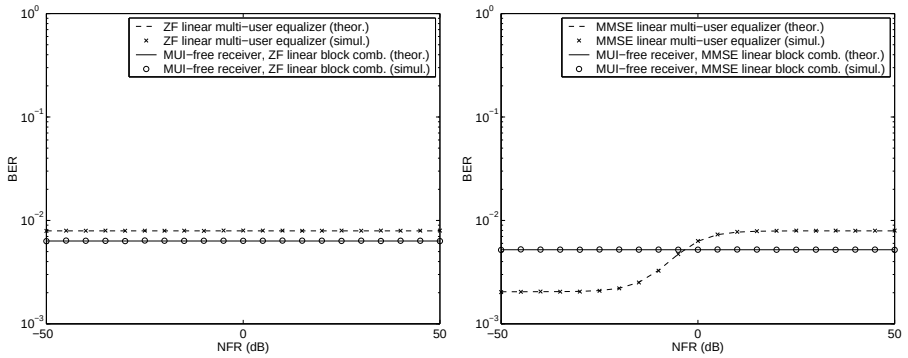



Figure 5.2: Average theoretical and simulated BER per user as a function of the NFR for an SNR of 10 dB (known channels and noise variance).

takes place is positioned around the inverse of the SNR. Further, we observe that the performance of the MUI-free receiver with ZF linear block combining is comparable with the performance of the ZF linear multi-user equalizer. We know that the performance of the MUI-free receiver with MMSE linear block combining is better than the performance of the MUI-free receiver with ZF linear block combining and that the difference between those performances decreases with the SNR. Therefore, the performance of the MUI-free receiver with MMSE linear block combining is also better than the performance of the MMSE linear multi-user equalizer at very high NFR (see figure 5.2) and the difference between those performances also decreases with the SNR. The somewhat surprising result that the MUI-free receiver can perform better than the linear multi-user equalizer is due to the fact that the linear multi-user equalizer we consider here only has length $N - L + 1 = 14$. Hence, such a linear multi-user equalizer removes $L - 1 = 3$ out of $N = 17$ received samples, while in the MUI-free receiver every modified block correlator only removes 1 out of $N = 17$ received samples (since every modified block correlator only removes 1 out of $N = 17$ received sample blocks). The performance of a much longer (hence, much more expensive) linear multi-user equalizer would be somewhat better than the performance of the MUI-free receiver. Finally, **figure 5.3** shows the theoretical and simulated SINR as a function of the desired user for an SNR of 10 dB and an NFR of 0 dB. Again, we observe that the simulation results are well predicted by the theoretical results.

*Simulation 2:* Next, we compare the performance of the subspace deterministic blind single-user channel estimation algorithm of section 4.5 with the performance of the subspace deterministic blind multi-user channel estimation algorithm of section 3.4. **Figure 5.4** shows the average theoretical and simulated NMSE per user of the channel estimates as a function of the SNR for an NFR of

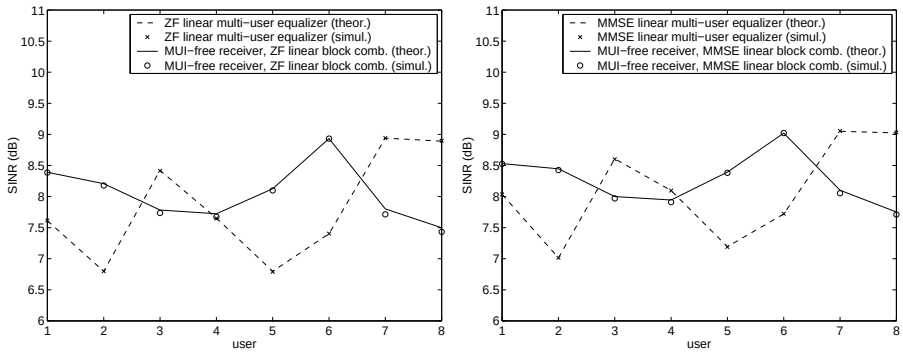



Figure 5.3: Theoretical and simulated SINR as a function of the desired user for an SNR of 10 dB and an NFR of 0 dB (known channels and noise variance).

0 dB. **Figure 5.5** shows the same results as a function of the NFR for an SNR of 20 dB. First of all we see that the theoretical performances of the single-user and multi-user algorithm are comparable. From figure 5.4 we observe that for an SNR above 10 dB the simulation results are well predicted by the theoretical results (below 10 dB the additive noise influence is too large to predict the simulation results using only first order subspace perturbation analysis). From figure 5.5 we notice that for the single-user algorithm the above statement is true for all values of the NFR, while for the multi-user algorithm it is only true for an NFR above the inverse of the SNR. When the NFR drops below the inverse of the SNR, the $(N - L + 1) \times (N - L + 1 - J)$ matrix $\hat{\mathbf{U}}^n$, that is used in section 3.4.1 to estimate the channel vector $\mathbf{g}_j$, is greatly influenced by additive noise and the first order subspace perturbation analysis is not accurate enough to predict the simulation results. We know from section 4.5 that the single-user algorithm should always be NFR-independent. This is clearly illustrated in figure 5.5. The multi-user algorithm, on the other hand, only appears to be theoretically NFR-independent. The simulation results show that this is not the case in practice (see discussion above).

*Simulation 3:* We now repeat simulation 1 but for each receiver we will use the appropriate subspace deterministic blind channel estimation algorithm. Moreover, for each of both MMSE receivers we will use the appropriate estimate of the noise variance $\sigma_e^2$ (see sections 4.5.3 and 3.4.3). **Figure 5.6** shows the average theoretical (known channels and noise variance) and simulated (estimated channels and noise variance) BER per user as a function of the SNR for an NFR of 0 dB. **Figure 5.7** shows the same results as a function of the NFR for an SNR of 10 dB. We observe that estimating the channels and the noise variance (necessary for both MMSE receivers) decreases the performances of all receivers. Note that the design of the desired user's ZF or MMSE lin-

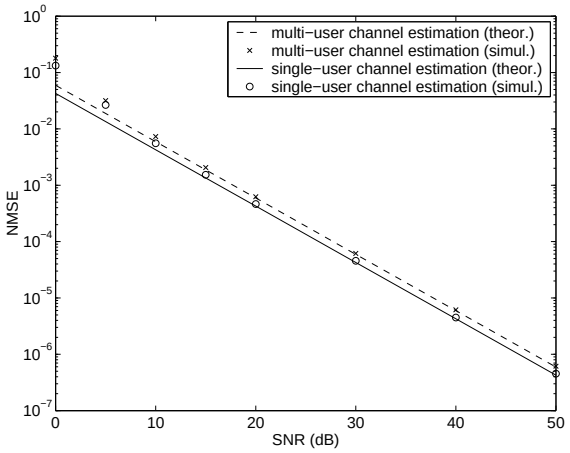



Figure 5.4: Average theoretical and simulated NMSE per user of the channel estimates as a function of the SNR for an NFR of 0 dB.

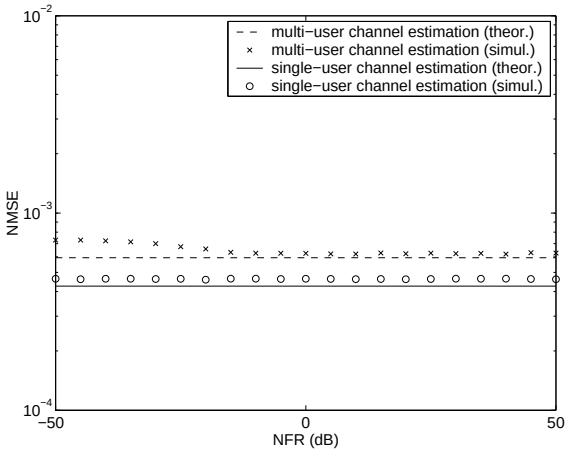



Figure 5.5: Average theoretical and simulated NMSE per user of the channel estimates as a function of the NFR for an SNR of 20 dB.

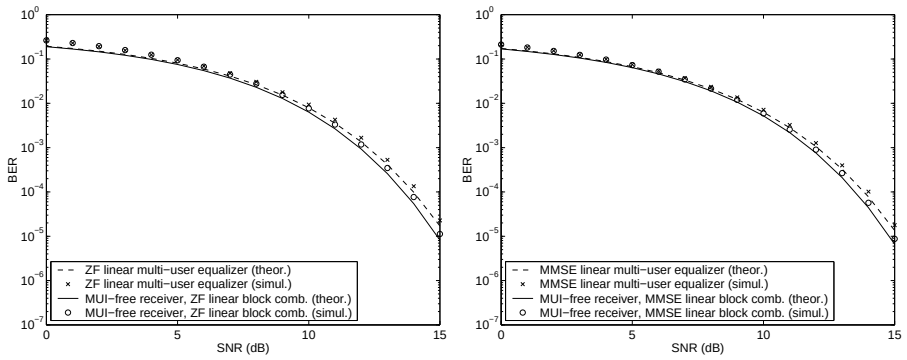



Figure 5.6: Average theoretical (known channels and noise variance) and simulated (estimated channels and noise variance) BER per user as a function of the SNR for an NFR of 0 dB.

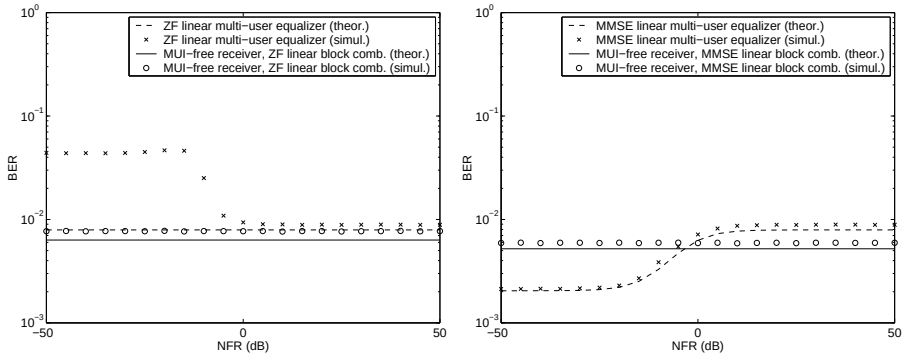



Figure 5.7: Average theoretical (known channels and noise variance) and simulated (estimated channels and noise variance) BER per user as a function of the NFR for an SNR of 10 dB.

ear multi-user equalizer requires the knowledge of all the channels (we do not consider the approach discussed in remarks 3.3.2 and 3.3.3). When the NFR drops below the inverse of the SNR, the channel estimates of the interfering users become less accurate. For the ZF linear multi-user equalizer, this property increases the difference in BER with simulation 1. For the MMSE linear multi-user equalizer, this property does not change the difference in BER with simulation 1, since the MMSE linear multi-user equalizer is, in that region of the NFR, more steered by the additive noise.

## 5.3 MUI-Free Receiver versus VL-AMOUR

In this section, we compare the MUI-free receiver with the corresponding FFT-based Vandermonde-Lagrange AMOUR (VL-AMOUR) transceiver [WSGB99].

The VL-AMOUR transceiver was first presented in [WSGB99]. Like the MUI-free receiver, the structure of VL-AMOUR also removes the MUI, without using any channel information. Note that the Vandermonde-Lagrange AMOUR (VL-AMOUR) transceiver has some flexibility advantages over the earlier developed Lagrange-Vandermonde AMOUR (LV-AMOUR) transceiver [GWSB99]. Further, note that the VL-AMOUR and LV-AMOUR transceiver can be considered as a generalization of the VL-CDMA [SBG99] and LV-CDMA [SG99] transceiver, respectively.

In this section, we focus on an FFT-based VL-AMOUR transceiver (only fast Fourier transform (FFT) and inverse fast Fourier transform (IFFT) operations are used). To make a fair comparison with the MUI-free receiver, we assume that the transmitter of the FFT-based VL-AMOUR transforms the data symbol block sequence $\mathbf{s}_j[k]$ with block size $L$ into a symbol block sequence with block size $NL$, applying an operation based on an $(N-1)L$-points IFFT and inserting a cyclic prefix of length $L$. Every user will be assigned a distinct set of $2L$ equidistant tones. Note that this transmitter scheme is *more complex in operation* than the spreading operation used in a DS-CDMA-BS system (because of the large IFFT). The receiver of the FFT-based VL-AMOUR then removes the cyclic prefix of length $L$ and applies an operation based on an $(N-1)L$-points FFT. This leads to a data model that resembles (4.9). Next, a linear block combiner is applied, like in the MUI-free receiver. Note that this receiver scheme is *more complex in operation* than the MUI-free receiver (because of the large FFT). The FFT-based VL-AMOUR transceiver described above is referred to as the FFT-based VL-AMOUR transceiver that corresponds to the MUI-free receiver.

**Remark 5.3.1** As shown in section 4.4, it is possible to design a shift-orthogonal set of *constant modulus* code sequences (e.g., BPSK or QPSK code se-

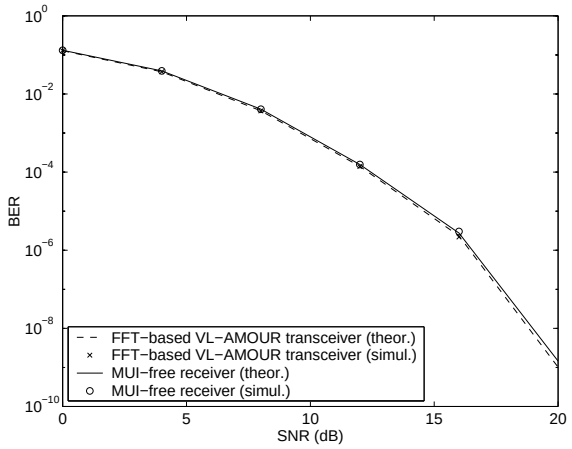



Figure 5.8: Average theoretical and simulated BER as a function of the SNR for an NFR of 0 dB (known channels).

quences). This has the advantage that, when the data symbols are constant modulus, the chip sequences will also be constant modulus. Note that *this is certainly not the case for the VL-AMOUR transceiver.* □

We now compare the performance of the MUI-free receiver with ZF linear block combining with the performance of the corresponding FFT-based VL-AMOUR transceiver with ZF linear block combining, under the assumption that all the channels $\{g_j[n]\}_{j=1}^{J}$ are known. We consider the same simulation conditions as in section 5.2, except that we now assume BPSK modulation instead of QPSK modulation and that we now consider 100 different channel realization and 500 trials per channel instead of only one channel realization and 5000 trials. **Figure 5.8** shows the average theoretical and simulated BER per user as a function of the SNR for an NFR of 0 dB. It is clear that the performance of the MUI-free receiver with ZF linear block combining is comparable with the performance of the corresponding FFT-based VL-AMOUR receiver with ZF linear block combining.

## 5.4 Conclusions

In this chapter, we have compared the new MUI-free receiver, developed for a DS-CDMA-BS system, with the existing linear multi-user equalizer, developed for a DS-CDMA system. First of all, it has been shown that *the complexity of the MUI-free receiver is much smaller than the complexity of the linear*

*multi-user equalizer*, if the receiver design is based on the appropriate subspace deterministic blind channel estimation algorithm. Moreover, *the complexity of a complete bank of MUI-free receivers is also much smaller than the complexity of a complete bank of linear multi-user equalizers*, although the latter is of the same order as the complexity of only one linear multi-user equalizer. Next, we have illustrated that *the performance of the MUI-free receiver with ZF respectively MMSE linear block combining is comparable with the performance of the ZF respectively MMSE linear multi-user equalizer*. This for the case the channels and the noise variance (necessary for both MMSE receivers) are known as well as for the case the channels are estimated using the appropriate subspace deterministic blind channel estimation algorithm and the noise variance is estimated as in the corresponding blind channel gain estimation algorithm. In this chapter, we have also compared the MUI-free receiver with the corresponding FFT-based VL-AMOUR transceiver [WSGB99]. We have shown that *the latter uses a transmitter scheme that is more complex in operation than the spreading operation used in a DS-CDMA-BS system and uses a receiver scheme that is more complex in operation than the MUI-free receiver*. For the case the channels are known, simulation results illustrate that *the performance of the MUI-free receiver with ZF linear block combining is comparable with the performance of the corresponding FFT-based VL-AMOUR transceiver with ZF linear block combining*.

# Chapter 6

# Receivers for a DMT-CDMA and DMT-CDMA-BS System

In this chapter, we present different receivers for a DMT-CDMA and DMT-CDMA-BS system. In section 6.1, we review some existing receivers for a DMT-CDMA system. First, the RAKE receiver is discussed. Then, we discuss the linear multi-user equalizer. Each of these receivers simply consists of the corresponding receiver for a DS-CDMA system followed by a DMT demodulator. In section 6.2, we develop new receivers for a DMT-CDMA-BS system. First, the block RAKE receiver is introduced. Then, we introduce the MUI-free receiver, which completely removes the MUI, without using any channel information. Again, each of these receivers simply consists of the corresponding receiver for a DS-CDMA-BS system followed by a DMT demodulator. Finally, we develop the MUI/ITI-free receiver, which completely removes the MUI and inter-tone interference (ITI), without using any channel information. The description of the MUI/ITI-free receiver can be considered as the main contribution of this chapter. Complexity issues and simulation results are discussed in section 6.3. Conclusions are given in section 6.4.

## 6.1 DMT-CDMA System

In this section, we review some existing receivers for a DMT-CDMA system. Although the receivers we discuss can be developed for a more general scenario, we consider a single receive antenna and an ICI-limited quasi-synchronous prop-

agation model. We further use short code sequences.

Details on the used data model are given in section 6.1.1. In section 6.1.2, the RAKE receiver is discussed. In section 6.1.3, we then discuss the linear multi-user equalizer. Each of these receivers basically is a cascade of the corresponding receiver for a DS-CDMA system and a DMT demodulator.

### 6.1.1 Data Model

We consider a single receive antenna and an ICI-limited quasi-synchronous propagation model (see section 2.2.1). We further use short code sequences (see section 2.1.4). The data models that will be used are given by (2.11) and (2.15), which are repeated here for convenience:

$$y[n] = \sum_{j=1}^{J} \sum_{n'=0}^{L-1} g_j[n'] x_j[n-n'] + e[n], \tag{6.1}$$

$$\mathbf{y}^{cut}[k] = \mathbf{H}\mathbf{r}[k] + \mathbf{e}^{cut}[k]. \tag{6.2}$$

### 6.1.2 RAKE Receiver

First, we apply a RAKE receiver developed for a DS-CDMA system. The bank of $L$ correlators results in the outputs $\{z_{j,l}[k]\}_{l=0}^{L-1}$ (see (3.3), where $s_j[k]$ should be replaced by $r_j[k]$). We then linearly combine these correlator outputs to suppress the remaining residual ICI and MUI:

$$\hat{r}_j[k] = f_{j,0} z_{j,0}[k] + f_{j,1}[k] z_{j,1}[k] + \cdots + f_{j,L-1} z_{j,L-1}[k],$$

where $\{f_{j,l-1}\}_{l=1}^{L}$ are the linear combiner weights.

Finally, we perform a discrete Fourier transformation (DFT):

$$\hat{\tilde{\mathbf{r}}}_j^{dmt}[\kappa] = 1/\sqrt{P}\,\mathcal{F}_P \hat{\mathbf{r}}_j^{dmt}[\kappa],$$

where $\mathcal{F}_P$ represents the $P \times P$ DFT matrix and

$$\hat{\mathbf{r}}_j^{dmt}[\kappa] = \begin{bmatrix} \hat{r}_j[\kappa P] & \hat{r}_j[\kappa P + 1] & \cdots & \hat{r}_j[(\kappa+1)P - 1] \end{bmatrix}^T.$$

### 6.1.3 Linear Multi-User Equalizer

First, we apply a linear multi-user equalizer developed for a DS-CDMA system. We perform a linear multi-user equalization on $\mathbf{y}^{cut}[k]$ to suppress the

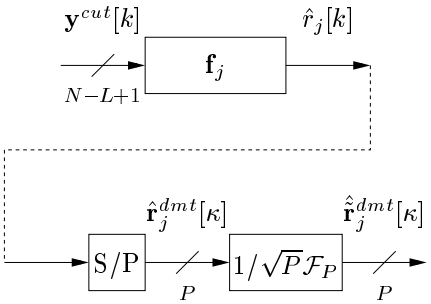



Figure 6.1: Linear multi-user equalizer for a DMT-CDMA system.

remaining MUI (the ISI is removed by construction):

$$\hat{r}_j[k] = \mathbf{f}_j \mathbf{y}^{cut}[k],$$

where $\mathbf{f}_j$ is the $1 \times (N - L + 1)$ linear multi-user equalizer for the $j$th user.

Finally, we perform a DFT:

$$\hat{\hat{\mathbf{r}}}_j^{dmt}[\kappa] = 1/\sqrt{P}\mathcal{F}_P \hat{\mathbf{r}}_j^{dmt}[\kappa],$$

where

$$\hat{\mathbf{r}}_j^{dmt}[\kappa] = \Big[\hat{r}_j[\kappa P] \quad \hat{r}_j[\kappa P + 1] \quad \cdots \quad \hat{r}_j[(\kappa + 1)P - 1]\Big]^T.$$

The linear multi-user equalizer is illustrated in **figure 6.1**.

## 6.2 DMT-CDMA-BS System

In this section, we develop new receivers for a DMT-CDMA-BS system. Like in the previous section, we consider a single receive antenna and an ICI-limited quasi-synchronous propagation model. We further use short code sequences.

Details on the used data model are given in section 6.2.1. In section 6.2.2, the block RAKE receiver is introduced. In section 6.2.3, we then introduce the MUI-free receiver, which completely removes the MUI, without using any channel information. Each of these receivers basically is a cascade of the corresponding receiver for a DS-CDMA-BS system and a DMT demodulator. In section 6.2.4, we finally develop the MUI/ITI-free receiver, which completely removes the MUI and ITI, without using any channel information. This receiver is obtained by replacing the linear block combiner of the MUI-free receiver by a simpler operation that is not based on channel information and that guarantees that the DMT demodulator following this operation completely removes

the ITI. Note that after the DMT demodulator we still have to rotate and scale every tone in the appropriate way by means of a 1-tap frequency domain equalizer (FEQ). The description of the MUI/ITI-free receiver can be considered as the main contribution of this chapter.

### 6.2.1 Data Model

We consider a single receive antenna and an ICI-limited quasi-synchronous propagation model (see section 2.2.1). We further use short code sequences (see section 2.1.4). As proposed in section 2.2.1, when an ICI-limited quasi-synchronous propagation model is considered for a DMT-CDMA-BS system, we take the block size $B$ equal to the parameter $L$ (see (2.10)). The data model that will be used is given by (2.16), which is repeated here for convenience:

$$\mathbf{y}[n] = \sum_{j=1}^{J} \sum_{n'=0}^{1} \mathbf{G}_j[n'] \mathbf{x}_j[n-n'] + \mathbf{e}[n]. \tag{6.3}$$

We also assume that the DMT symbol size $P$ is a multiple of the block size $B = L$.

### 6.2.2 Block RAKE Receiver

First, we apply a block RAKE receiver developed for a DS-CDMA-BS system. The bank of 2 block correlators results in the outputs $\mathbf{z}_{j,0}[k]$ and $\mathbf{z}_{j,1}[k]$ (see (4.2) and (4.3), where $\mathbf{s}_j[k]$ should be replaced by $\mathbf{r}_j[k]$). We then linearly combine these block correlator outputs to suppress the remaining residual ICBI and MUI, and the remaining ISI w.r.t. the symbols in $\mathbf{r}_j[k]$:

$$\hat{\mathbf{r}}_j[k] = \mathbf{F}_{j,0}\mathbf{z}_{j,0}[k] + \mathbf{F}_{j,1}\mathbf{z}_{j,1}[k],$$

where $\mathbf{F}_{j,0}$ and $\mathbf{F}_{j,1}$ are the $L \times L$ linear block combiner weights.

Finally, we perform a discrete Fourier transformation (DFT):

$$\hat{\tilde{\mathbf{r}}}_j^{dmt}[\kappa] = 1/\sqrt{P}\mathcal{F}_P \hat{\mathbf{b}}_j^{dmt}[\kappa],$$

where $\mathcal{F}_P$ represents the $P \times P$ DFT matrix and

$$\hat{\mathbf{r}}_j^{dmt}[\kappa] = \begin{bmatrix} \hat{\mathbf{r}}_j[\kappa P] & \hat{\mathbf{r}}_j[\kappa P + 1] & \cdots & \hat{\mathbf{r}}_j[(\kappa+1)P - 1] \end{bmatrix}^T.$$

### 6.2.3 MUI-Free Receiver

First, we apply an MUI-Free receiver developed for a DS-CDMA-BS system. The bank of 2 modified block correlators results in the outputs $\mathbf{z}_{j,0}^{mod}[k]$ and

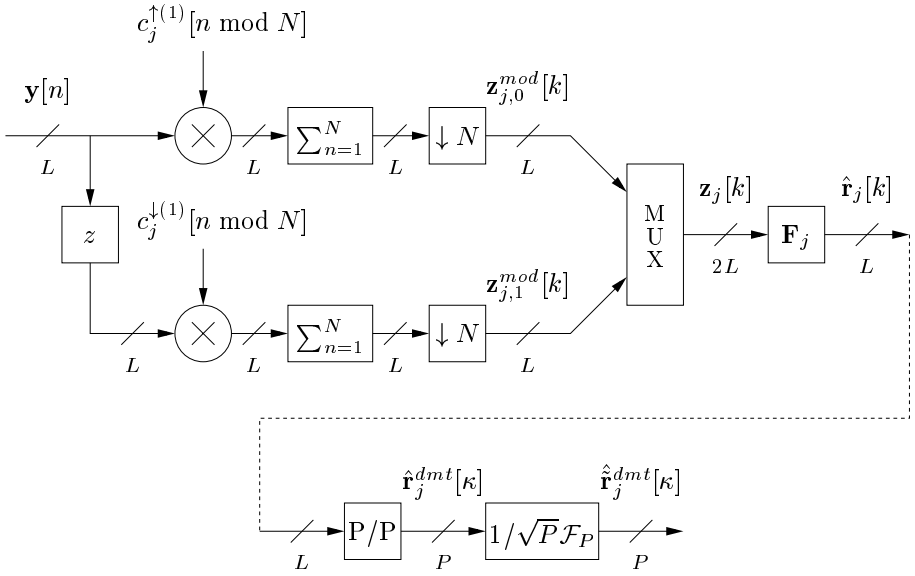



Figure 6.2: MUI-free receiver for a DMT-CDMA-BS system.

$\mathbf{z}_{j,1}^{mod}[k]$ (see (4.7) and (4.8), where $\mathbf{s}_j[k]$ should be replaced by $\mathbf{r}_j[k]$). We then linearly combine these modified block correlator outputs to suppress the remaining ISI w.r.t. the symbols in $\mathbf{r}_j[k]$:

$$\hat{\mathbf{r}}_j[k] = \mathbf{F}_{j,0}\mathbf{z}_{j,0}^{mod}[k] + \mathbf{F}_{j,1}\mathbf{z}_{j,1}^{mod}[k],$$

where $\mathbf{F}_{j,0}$ and $\mathbf{F}_{j,1}$ are the $L \times L$ linear block combiner weights.

Finally, we perform a DFT:

$$\hat{\hat{\mathbf{r}}}_j^{dmt}[\kappa] = 1/\sqrt{P}\mathcal{F}_P\hat{\mathbf{r}}_j^{dmt}[\kappa],$$

where

$$\hat{\mathbf{r}}_j^{dmt}[\kappa] = \Big[\hat{\mathbf{r}}_j[\kappa P] \quad \hat{\mathbf{r}}_j[\kappa P + 1] \quad \cdots \quad \hat{\mathbf{r}}_j[(\kappa+1)P - 1]\Big]^T.$$

The MUI-free receiver is illustrated in **figure 6.2**.

### 6.2.4 MUI/ITI-Free Receiver

The MUI/ITI-free receiver is obtained by replacing the linear block combiner of the MUI-free receiver by a simpler operation that is not based on channel information and that guarantees that the DMT demodulator following this operation completely removes the ITI. Note that after the DMT demodulator

we still have to rotate and scale every tone in the appropriate way by means of a 1-tap frequency domain equalizer (FEQ).

Consider the MUI-free receiver presented in the previous section. Instead of combining the 2 modified block correlator outputs $\mathbf{z}_{j,0}^{mod}[k]$ and $\mathbf{z}_{j,1}^{mod}[k]$ (see (4.7) and (4.8), where $\mathbf{s}_j[k]$ should be replaced by $\mathbf{r}_j[k]$), we now construct

$$\mathbf{z}_{j,0}^{dmt}[\kappa] = \left[\mathbf{z}_{j,0}^{modT}[\kappa P/L] \quad \mathbf{z}_{j,0}^{modT}[\kappa P/L+1] \quad \cdots \quad \mathbf{z}_{j,0}^{modT}[(\kappa+1)P/L-1]\right]^T,$$
$$\mathbf{z}_{j,1}^{dmt}[\kappa] = \left[\mathbf{z}_{j,1}^{modT}[\kappa P/L] \quad \mathbf{z}_{j,1}^{modT}[\kappa P/L+1] \quad \cdots \quad \mathbf{z}_{j,1}^{modT}[(\kappa+1)P/L-1]\right]^T.$$

Further, applying a simple transformation on $\mathbf{z}_{j,1}^{dmt}[\kappa]$:

$$\bar{\mathbf{z}}_{j,1}^{dmt}[\kappa] = \mathbf{J}_P^L \mathbf{z}_{j,1}^{dmt}[\kappa],$$

where $\mathbf{J}_n$ is defined in (4.18), we obtain

$$\bar{\mathbf{z}}_{j,1}^{dmt}[\kappa] =$$
$$\left[\mathbf{z}_{j,1}^{modT}[(\kappa+1)P/L-1] \quad \mathbf{z}_{j,1}^{modT}[\kappa P/L] \quad \cdots \quad \mathbf{z}_{j,1}^{modT}[(\kappa+1)P/L-2]\right]^T.$$

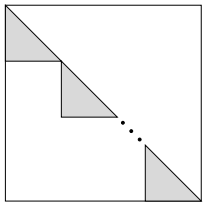
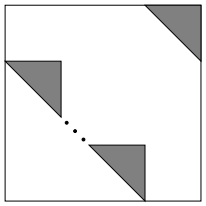
Defining

$$\mathbf{r}_j^{dmt}[\kappa] = \left[\mathbf{r}_j^T[\kappa P/L] \quad \mathbf{r}_j^T[\kappa P/L+1] \quad \cdots \quad \mathbf{r}_j^T[(\kappa+1)P/L-1]\right]^T,$$

it is then clear from (4.7) and (4.8) (replace $\mathbf{s}_j[k]$ by $\mathbf{r}_j[k]$) that

$$\mathbf{z}_{j,0}^{dmt}[\kappa] = \eta \mathbf{G}_{j,0}^{dmt} \mathbf{r}_j^{dmt}[\kappa] + \mathbf{n}_{j,0}^{dmt}[\kappa],$$
$$\bar{\mathbf{z}}_{j,1}^{dmt}[\kappa] = \eta \bar{\mathbf{G}}_{j,1}^{dmt} \mathbf{r}_j^{dmt}[\kappa] + \bar{\mathbf{n}}_{j,1}^{dmt}[\kappa],$$

where $\mathbf{n}_{j,0}^{dmt}[\kappa]$ is similarly defined as $\mathbf{z}_{j,0}^{dmt}[\kappa]$, $\bar{\mathbf{n}}_{j,1}^{dmt}[\kappa]$ is similarly defined as $\bar{\mathbf{z}}_{j,1}^{dmt}[\kappa]$ and $\mathbf{G}_{j,0}^{dmt}$ and $\bar{\mathbf{G}}_{j,1}^{dmt}$ are the $P \times P$ channel matrices for the $j$th user, given by

$\mathbf{G}_{j,0}^{dmt} =$  , $\bar{\mathbf{G}}_{j,1}^{dmt} =$  ,

with

 $= \mathbf{G}_j[0]$,  $= \mathbf{G}_j[1]$.

If we then simply sum $\mathbf{z}_{j,0}^{dmt}[\kappa]$ and $\bar{\mathbf{z}}_{j,1}^{dmt}[\kappa]$

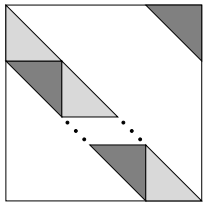
$$\mathbf{z}_j^{dmt}[\kappa] = \mathbf{z}_{j,0}^{dmt}[\kappa] + \bar{\mathbf{z}}_{j,1}^{dmt}[\kappa],$$

we obtain

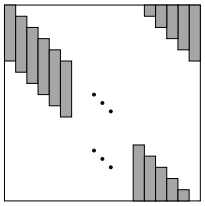
$$\mathbf{z}_j^{dmt}[\kappa] = \eta \mathbf{G}_j^{dmt} \mathbf{r}_j^{dmt}[\kappa] + \mathbf{n}_j^{dmt}[\kappa],$$

where $\mathbf{n}_j^{dmt}[\kappa]$ is similarly defined as $\mathbf{z}_j^{dmt}[\kappa]$ and $\mathbf{G}_j^{dmt}$ is the $P \times P$ channel matrix for the $j$th user, given by

$$\mathbf{G}_j^{dmt} = \mathbf{G}_{j,0}^{dmt} + \bar{\mathbf{G}}_{j,1}^{dmt} =$$



$=$



, (6.4)

with

$= \mathbf{G}_j[0],$

$= \mathbf{G}_j[1],$

$= \mathbf{g}_j.$

Note that if the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$, we obtain

$$\mathbf{R}_{\mathbf{n}_j^{dmt}} = \mathrm{E}\{\mathbf{n}_j^{dmt}[\kappa]\mathbf{n}_j^{dmtH}[\kappa]\} = 2\eta\sigma_e^2 \mathbf{I}_P.$$

We then perform a DFT:

$$\begin{aligned}\tilde{\mathbf{z}}_j^{dmt}[\kappa] &= 1/\sqrt{P}\mathcal{F}_P \mathbf{z}_j^{dmt}[\kappa] \\ &= \eta \mathcal{F}_P \mathbf{G}_j^{dmt} \mathcal{I}_P \tilde{\mathbf{r}}_j^{dmt}[\kappa] + 1/\sqrt{P}\mathcal{F}_P \mathbf{n}_j^{dmt}[\kappa].\end{aligned}$$

Because $\mathbf{G}_j^{dmt}$ is a circular matrix (see (6.4)), it is then clear that the ITI is completely removed:

$$\tilde{\mathbf{z}}_j^{dmt}[\kappa] = \eta \mathrm{diag}\{\mathcal{F}_P \mathbf{g}_j^{dmt}\} \tilde{\mathbf{r}}_j^{dmt}[\kappa] + 1/\sqrt{P}\mathcal{F}_P \mathbf{n}_j^{dmt}[\kappa],$$

where $\mathbf{g}_j^{dmt}$ is the $P \times 1$ channel vector for the $j$th user, given by

$$\mathbf{g}_j^{dmt} = \begin{bmatrix} g_j[0] & g_j[1] & \cdots g_j[P-1] \end{bmatrix}^T.$$

Finally, every tone is rotated and scaled in the appropriate way by means of a linear 1-tap frequency domain equalizer (FEQ). As an estimate for $\tilde{\mathbf{r}}_j^{dmt}[\kappa]$, we then obtain

$$\hat{\tilde{\mathbf{r}}}_j^{dmt}[\kappa] = \mathbf{\Lambda}_j \tilde{\mathbf{z}}_j^{dmt}[\kappa]$$

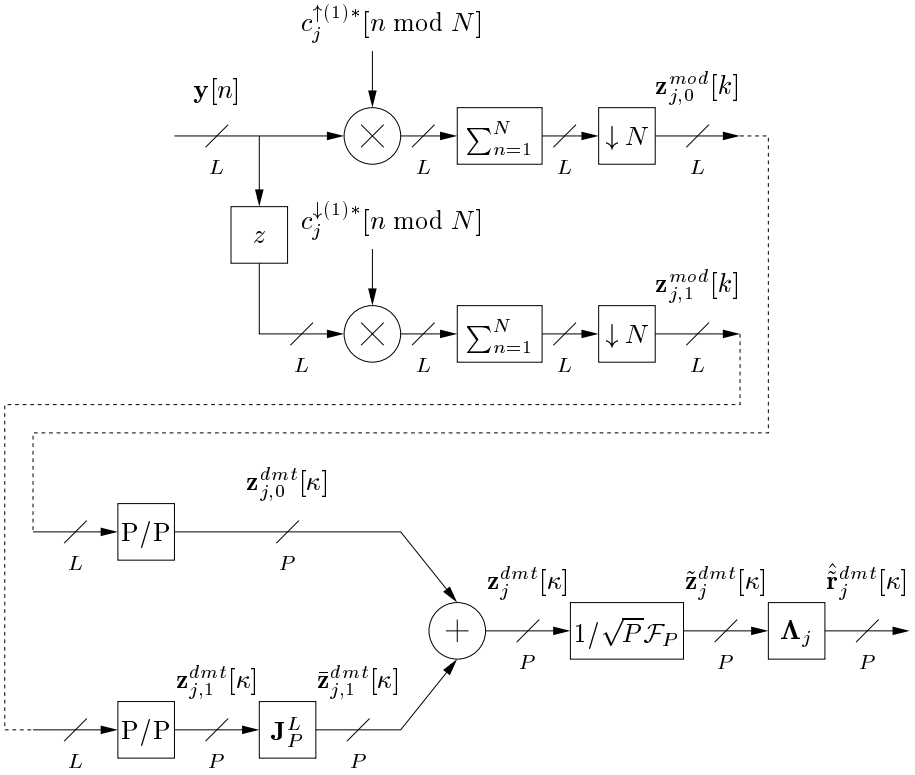



Figure 6.3: MUI/ITI-free receiver for a DMT-CDMA-BS system.

$$= \eta \mathbf{\Lambda}_j \text{diag}\{\mathcal{F}_P \mathbf{g}_j^{dmt}\} \tilde{\mathbf{r}}_j^{dmt}[\kappa] + 1/\sqrt{P} \mathbf{\Lambda}_j \mathcal{F}_P \mathbf{n}_j^{dmt}[\kappa],$$

where $\mathbf{\Lambda}_j$ is a $P \times P$ diagonal matrix whose diagonal elements are the linear 1-tap FEQ's for the $j$th user. The MUI/ITI-free receiver is illustrated in **figure 6.3**. We can use ZF or MMSE linear 1-tap FEQ's.

**Remark 6.2.1** If the DMT subsymbol sequences $\{\tilde{r}_{j,p}[\kappa]\}_{p=1}^{P}$ are PSK-modulated and the additive noise $e[n]$ is zero-mean white with variance $\sigma_e^2$ ($\mathbf{R}_{\mathbf{n}_j^{dmt}} = 2\eta\sigma_e^2 \mathbf{I}_P$), the ZF and MMSE linear 1-tap FEQ's for the $j$th user have the same performance and can, without any loss in performance, be replaced by

$$\mathbf{\Lambda}_j = \text{diag}\{\mathcal{F}_P \mathbf{g}_j^{dmt}\}^H. \tag{6.5}$$

□

## 6.3 Complexity and Simulation Results

We first compare the complexity of the MUI-free receiver with the complexity of the linear multi-user equalizer. Like in the previous chapter, we assume that the receiver design is based on subspace deterministic blind channel estimation. Following the same reasoning as in the previous chapter, it can be shown that the complexity of the MUI-free receiver is much smaller than the complexity of the linear multi-user equalizer. Further, note that the MUI/ITI-free receiver is cheaper than the MUI-free receiver, because the use of the linear block combiner is avoided.

We now perform some simulations in Matlab to compare the performance of the MUI-free receiver with ZF linear block combining and the MUI/ITI-free receiver with the performance of the ZF linear multi-user equalizer. We consider an 8-user DMT-CDMA-BS and DMT-CDMA system ($J = 8$) with spreading factor $N = 17$ and a DMT symbol size $P = 8$. We design a shift-orthogonal set of $J$ length-$N$ code sequences $\{c_j[n]\}_{j=1}^{J}$, as explained in section 4.4, and use this set of code sequences for the DMT-CDMA-BS system as well as for the DMT-CDMA system. We start from (see section 4.4 for details)

$$\mathbf{C}_4 = \begin{bmatrix} +1 & -1 \\ +1 & +1 \\ -1 & +1 \\ +1 & +1 \end{bmatrix},$$

resulting into a set of BPSK code sequences. For every user $j$, we generate a random channel $g_j[n]$ of order $L_j = 3$ with delay index $\delta_j = 0$. We assume $L = 4$ (note that (2.10) is then satisfied). The resulting set of normalized

channels is listed in table 5.1. We assume that the additive noise $e[n]$ is zero-mean white Gaussian with variance $\sigma_e^2$. Introducing the $P \times P$ diagonal matrix

$$\mathbf{P}_j = \text{diag}\{\mathcal{F}_P \mathbf{g}_j^{dmt}\}^H \text{diag}\{\mathcal{F}_P \mathbf{g}_j^{dmt}\},$$

we define the received energy per DMT symbol period ($PT$) for the $j$th user as

$$E_j^{dmt} = \text{E}\{\tilde{\mathbf{r}}_j^{dmtH}[\kappa] \mathbf{P}_j \tilde{\mathbf{r}}_j^{dmt}[\kappa]\}$$

and we assume that all users interfering with this $j$th user have the same received energy per DMT symbol period:

$$E_{j'}^{dmt} = E^{dmt}, \quad \text{for } j' = 1, \ldots, j-1, j+1, \ldots, J.$$

We then define the signal-to-noise ratio (SNR) and the near-far ratio (NFR) for the $j$th user at the input of the receiver as

$$SNR_j = \frac{E_j^{dmt}}{P\sigma_e^2} \quad \text{and} \quad NFR_j = \frac{E^{dmt}}{E_j^{dmt}}.$$

We consider two scenarios:

1. Every DMT subsymbol sequence $\tilde{r}_{j,p}[\kappa]$ is QPSK-modulated with variance 1. Note that $E_j^{dmt}$ then becomes

$$E_j^{dmt} = P\|\mathbf{g}_j^{dmt}\|^2.$$

2. Every DMT subsymbol sequence $\tilde{r}_{j,p}[\kappa]$ is QPSK-modulated with variance $\|\boldsymbol{g}_j^{dmt}\|^2 \mathbf{P}_j^{-1}(p,p)$. Note that $E_j^{dmt}$ then again becomes

$$E_j^{dmt} = P\|\mathbf{g}_j^{dmt}\|^2.$$

Note that the second scenario does not require a feedback path from the receiver to the transmitter, if we consider a time division duplexing (TDD) approach (uplink and downlink channels are the same), while it does require a feedback path from the receiver to the transmitter, if we consider a frequency division duplexing (FDD) approach (uplink and downlink channels are different). Obviously, the first scenario does not require a feedback path, irrespective of the type of duplexing. For all simulations we will conduct 5000 trials using bursts of $K = 200$ data symbols.

*Simulation 1:* First, we compare the performance of the MUI-free receiver with ZF linear block combining with the performance of the ZF linear multi-user equalizer, under the assumption that all the channels $\{g_j[n]\}_{j=1}^J$ are known. **Figure 6.4** shows the average BER per user as a function of the SNR for

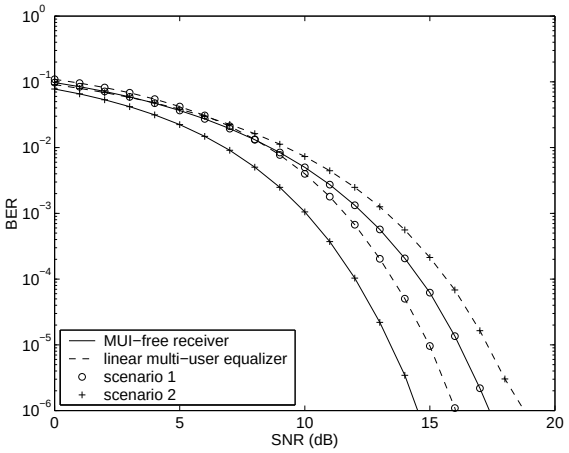



Figure 6.4: Average BER per user as a function of the SNR for an NFR of 0 dB (known channels).

an NFR of 0 dB. For scenario 1, the MUI-free receiver with ZF linear block combining performs somewhat worse than the ZF linear multi-user equalizer. However, if we switch from scenario 1 to scenario 2, the performance of the MUI-free receiver with ZF linear block combining improves, while the performance of the ZF linear multi-user equalizer deteriorates. The first can be explained by the fact that the MUI-free receiver removes the MUI, keeping the noise white, and then removes the ISI, coloring the noise in a way that only depends on the desired user's channel, while the latter can be explained by the fact that the ZF linear multi-user equalizer removes the MUI (ISI is removed by construction), coloring the noise in a way that depends on all channels. As a result, the performance of the MUI-free receiver with ZF linear block combining for scenario 2 (best scenario for this receiver) is better than the performance of the ZF linear multi-user equalizer for scenario 1 (best scenario for this receiver).

*Simulation 2:* Next, we compare the performance of the MUI/ITI-free receiver with the performance of the ZF linear multi-user equalizer, again under the assumption that all the channels $\{g_j[n]\}_{j=1}^{J}$ are known. Note that for the MUI/ITI-free receiver we take linear 1-tap FEQ's as in (6.5). **Figure 6.5** shows the average BER per user as a function of the SNR for an NFR of 0 dB. For scenario 1, the MUI/ITI-free receiver performs worse than the ZF linear multi-user equalizer. However, if we switch from scenario 1 to scenario 2, the performance of the MUI/ITI-free receiver improves, while the performance of the ZF linear multi-user equalizer deteriorates (as already mentioned). The first can be explained by the fact that the MUI/ITI-free receiver removes the

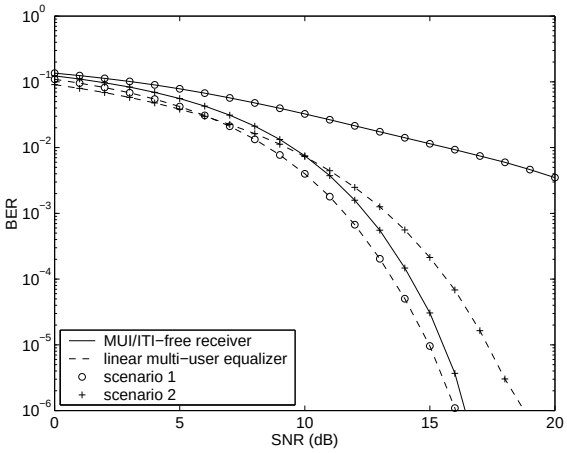



Figure 6.5: Average BER per user as a function of the SNR for an NFR of 0 dB (known channels).

MUI and ITI, keeping the noise white, and then rotates and scales every tone in the appropriate way, coloring the noise in a way that only depends on the desired user's channel, while the latter can be explained by the fact that the ZF linear multi-user equalizer removes the MUI (ISI is removed by construction), coloring the noise in a way that depends on all channels (as already mentioned). As a result, the performance of the MUI/ITI-free receiver for scenario 2 (best scenario for this receiver) is comparable with the performance of the ZF linear multi-user equalizer for scenario 1 (best scenario for this receiver).

## 6.4 Conclusions

In this chapter, we have presented different receivers for a DMT-CDMA and DMT-CDMA-BS system. For the DMT-CDMA system, we have reviewed some existing receivers. First, the RAKE receiver has been discussed. Then, we have discussed the linear multi-user equalizer. For the DMT-CDMA-BS system, we have developed new receivers. First, the block RAKE receiver has been introduced. Then, we have introduced the MUI-free receiver, which completely removes the MUI, without using any channel information. Finally, we have developed the MUI/ITI-free receiver, which completely removes the MUI and ITI, without using any channel information. The description of the MUI/ITI-free receiver can be considered as the main contribution of this chapter.

The complexity of the MUI-free receiver is much smaller than the complexity of

the linear multi-user equalizer, if the receiver design is based on the appropriate subspace deterministic blind channel estimation algorithm. Further, note that the MUI/ITI-free receiver is cheaper than the MUI-free receiver, because the use of the linear block combiner is avoided.

Interesting conclusions from the simulation results are the following:

- Switching from a scenario where the power of every DMT subsymbol sequence is the same, to a scenario where the power of every DMT subsymbol sequence is adjusted to the power of the channel at the corresponding frequency, the performance of the MUI-free receiver and the MUI/ITI-free receiver improves, while the performance of the linear multi-user equalizer deteriorates.
- The performance of the MUI-free receiver and the MUI/ITI-free receiver, considering a scenario where the power of every DMT subsymbol sequence is adjusted to the power of the channel at the corresponding frequency (best scenario for these receivers), is respectively *better than* and *comparable to* the performance of the linear multi-user equalizer, considering a scenario where the power of every DMT subsymbol sequence is the same (best scenario for this receiver).

# Part II

# Deterministic Blind Space-Time RAKE Receivers

In this part of the thesis, *deterministic* blind space-time RAKE (ST-RAKE) receivers for a DS-CDMA system with multiple receive antennas are developed. In a block processing context, these require much smaller blocks of received samples than stochastic blind ST-RAKE receivers. As a result, adaptive implementations can be used in a highly time-varying environment.

In *chapter 7*, we discuss deterministic blind ST-RAKE receivers based on *block processing*. *Direct blind symbol estimation* as well as *direct blind equalizer estimation* will be focused on. In *chapter 8*, we then present some deterministic blind ST-RAKE receivers based on *adaptive processing*. We make a distinction between *windowed* and *exponentially-weighted* adaptive processing. Focus here will only be on direct blind symbol estimation. In both chapters, we present *subspace methods* as well as *non-subspace methods*. Interesting complexity and performance comparisons of the proposed methods are given.

In *chapter 9*, we finally show how the deterministic blind ST-RAKE receivers based on block processing, presented in chapter 7, can also be applied on a DMT-DS-CDMA system (similar reasonings are possible for the deterministic blind ST-RAKE receivers based on adaptive processing, presented in chapter 8). We show that the use of a cyclic prefix reduces the ICI in a DMT-DS-CDMA system and that, as a result of this reduction of ICI, the computational complexity of the block processing methods, presented in chapter 7, can be reduced when they are carried out for a tone of a DMT-DS-CDMA system w.r.t. the case they are carried out for the related DS-CDMA system. Moreover, simulation results show that this reduction in computational complexity even comes with a performance improvement.

# Chapter 7

# Deterministic Blind Block Processing

In this chapter, we develop deterministic blind ST-RAKE receivers based on block processing. The data model is introduced in section 7.1. In section 7.2, the basic idea behind block processing is explained. We consider a subspace framework as well as a non-subspace framework. The difference between these two frameworks is that the subspace framework is based on the use of a subspace decomposition, while the non-subspace framework is not. In sections 7.3 and 7.4, we then discuss different block processing methods for direct blind symbol estimation and direct blind equalizer estimation, respectively. Section 7.5 further discusses the proposed methods. The proposed block processing methods based on direct blind symbol estimation have to the best of our knowledge not yet been studied in the literature. Some of the proposed block processing methods based on direct blind equalizer estimation, on the other hand, have already been studied in the literature, but only in their most simplest form. There also exist some interesting relations between the proposed methods, on one side, and the subspace intersection (SSI) methods of [LX97, vdVTP97] and the mutually referenced equalizer (MRE) method of [GDM97], which are all developed for a TDMA system (no coding), on the other side. A computational complexity analysis is given in section 7.6.

The methods presented in sections 7.3 and 7.4 are applicable for any modulation format. However, when the data symbols belong to a *real constellation*, we can adapt these methods to exploit the real character of the data symbols, which is discussed in section 7.7.

In section 7.8, we also describe how the methods presented in sections 7.3 and 7.4 can be adapted when *transmit diversity* is used (multiple transmit

antennas per user). We then spread the same data symbol sequence with different code sequences and transmit the resulting chip sequences through different antennas.

The smaller the spreading factor $N$, the larger the data symbol rate for a fixed bandwidth or the smaller the bandwidth for a fixed data symbol rate. Hence, it is worthwhile to investigate the case $N = 1$, which is done in section 7.9. This case is referred to as *code modulation*, since a data symbol sequence is then modulated (and not spread) with a code sequence.

In section 7.10 we show some simulation results. Conclusions are drawn in section 7.11.

## 7.1 Data Model

We consider a DS-CDMA system with $M$ receive antennas. The data model that will be used is given by (2.17), which is repeated here for convenience:

$$\mathbf{y}^{st}[n] = \sum_{j=1}^{J} \sum_{n'=-\infty}^{+\infty} \mathbf{g}_j^{st}[n'] x_j[n-n'] + \mathbf{e}^{st}[n]. \tag{7.1}$$

**Remark 7.1.1** Note that the results obtained here can easily be generalized for the case the spatial oversampling is replaced by or combined with *temporal oversampling* (see section 2.2.3). □

Let us then introduce the following $(Q+1)M \times \beta N$ output matrix (with Hankel structure):

$$\mathbf{Y}_n = \begin{bmatrix} \mathbf{y}^{st}[n] & \mathbf{y}^{st}[n+1] & \cdots & \mathbf{y}^{st}[n+\beta N-1] \\ \vdots & \vdots & & \vdots \\ \mathbf{y}^{st}[n+Q] & \mathbf{y}^{st}[n+Q+1] & \cdots & \mathbf{y}^{st}[n+Q+\beta N-1] \end{bmatrix},$$

where $\beta > 0$ is the amount of data symbols 'under consideration' (this will be specified later) and $Q \geq 0$ determines the amount of *temporal smoothing*. This output matrix can be written as

$$\mathbf{Y}_n = \sum_{j=1}^{J} \mathcal{G}_j \mathbf{X}_{j,n} + \mathbf{E}_n, \tag{7.2}$$

where $\mathbf{E}_n$ is similarly defined as $\mathbf{Y}_n$, $\mathcal{G}_j$ is the $(Q+1)M \times r_j$ ($r_j = Q+1+L_j$) channel matrix (with Toeplitz structure) for the $j$th user, given by

$$\mathcal{G}_j = \begin{bmatrix} \mathbf{g}_j^{st}[\delta_j + L_j] & \cdots & \mathbf{g}_j^{st}[\delta_j] & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{g}_j^{st}[\delta_j + L_j] & \cdots & \mathbf{g}_j^{st}[\delta_j] & \cdots & \mathbf{0} \\ & & \ddots & & \ddots & \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{g}_j^{st}[\delta_j + L_j] & \cdots & \mathbf{g}_j^{st}[\delta_j] \end{bmatrix}, \tag{7.3}$$

and $\mathbf{X}_{j,n}$ is the following $r_j \times \beta N$ input matrix (with Hankel structure) for the $j$th user:

$$\mathbf{X}_{j,n} = \begin{bmatrix} \mathbf{x}_{j,n-\delta_j-L_j}^T & \cdots & \mathbf{x}_{j,n-\delta_j+Q}^T \end{bmatrix}^T,$$

with

$$\mathbf{x}_{j,n} = \begin{bmatrix} x_j[n] & x_j[n+1] & \cdots & x_j[n+\beta N-1] \end{bmatrix}. \tag{7.4}$$

Note that (7.2) can also be written as

$$\mathbf{Y}_n = \mathcal{G}\mathbf{X}_n + \mathbf{E}_n, \tag{7.5}$$

where $\mathcal{G}$ is the $(Q+1)M \times r$ ($r = \sum_{j=1}^{J} r_j$) channel matrix, given by

$$\mathcal{G} = \begin{bmatrix} \mathcal{G}_1 & \cdots & \mathcal{G}_J \end{bmatrix}, \tag{7.6}$$

and $\mathbf{X}_n$ is the following $r \times \beta N$ input matrix:

$$\mathbf{X}_n = \begin{bmatrix} \mathbf{X}_{1,n}^T & \cdots & \mathbf{X}_{J,n}^T \end{bmatrix}^T.$$

We consider a burst length of $K$ data symbols ($s_j[k] \neq 0$, for $k = 0, 1, \ldots, K-1$, and $s_j[k] = 0$, for $k < 0$ and $k \geq K$) and constant modulus code symbols:

$$|c_j[n]| = \frac{1}{\sqrt{N}}, \quad \text{for } n = 0, 1, \ldots, \rho N - 1. \tag{7.7}$$

Note that we do not restrict ourselves to the use of short code sequences. We assume that the first user ($j = 1$) is desired and that $\delta_1$ is known at the receiver. We further assume w.l.o.g. that $\delta_1 = 0$.

## 7.2 Preliminaries

For the block processing approach, we consider the data model presented in the previous section, with $n = a$, where $a$ represents the *processing delay*, and

$\beta = K$, where $K$ is the burst length. Let us then define

$$\mathbf{s}_1 = \begin{bmatrix} s_1[0] & s_1[1] & \cdots & s_1[K-1] \end{bmatrix}. \tag{7.8}$$

Using (2.1) and (7.4), it is clear that this vector is 'contained' in $\mathbf{x}_{1,0}$:

$$\mathbf{x}_{1,0} = \begin{bmatrix} s_1[0]\mathbf{c}_1[0] & \cdots & s_1[K-1]\mathbf{c}_1[K-1] \end{bmatrix} = \mathbf{s}_1\mathbf{C}_1, \tag{7.9}$$

where $\mathbf{c}_1[k]$ is the code vector used to spread the data symbol $s_1[k]$ (spreading factor $N$):

$$\mathbf{c}_1[k] = \tag{7.10}$$
$$\begin{bmatrix} c_1[(k \bmod \rho)N] & c_1[(k \bmod \rho)N+1] & \cdots & c_1[(k \bmod \rho)N+N-1] \end{bmatrix},$$

and $\mathbf{C}_1$ is the $K \times KN$ code matrix, defined as

$$\mathbf{C}_1 = \begin{bmatrix} \mathbf{c}_1[0] & & \\ & \ddots & \\ & & \mathbf{c}_1[K-1] \end{bmatrix}. \tag{7.11}$$

The vector $\mathbf{x}_{1,0}$ is a row of every input matrix from the set $\{\mathbf{X}_a\}_{a=-Q}^{L_1}$ and is therefore 'contained' in every output matrix from the set $\{\mathbf{Y}_a\}_{a=-Q}^{L_1}$. The block processing problem addressed here is to compute $\mathbf{s}_1$ from $\{\mathbf{Y}_a\}_{a=A_1}^{A_2}$, with $-Q \le A_1 \le A_2 \le L_1$ ($A = A_2 - A_1 + 1$), based only on the knowledge of the code sequence $c_1[n]$. The solutions we will describe for this problem are based on the following assumptions:

**Assumption 7.2.1** The channel matrix $\mathcal{G}$ has full column rank $r$ ($r$ is then called the system order). □

**Assumption 7.2.2** The input matrix $\mathbf{X}_a$ has full row rank $r$, for $a = A_1, A_1 + 1, \ldots, A_2$. □

The first assumption requires that

$$(Q+1)M \ge r.$$

Since $r = \sum_{j=1}^{J} r_j = (Q+1)J + \sum_{j=1}^{J} L_j$, this is equivalent with

$$(Q+1)(M-J) \ge \sum_{j=1}^{J} L_j. \tag{7.12}$$

The second assumption, on the other hand, requires that

$$KN \geq r.$$

To solve the above block processing problem, we consider a subspace framework as well as a non-subspace framework. The subspace framework is based on the use of a subspace decomposition, while the non-subspace framework is not.

### 7.2.1 Subspace Framework

Let us first, for the sake of clarity, assume that there is no additive noise present in $\mathbf{Y}_a$ ($a = A_1, A_1+1, \ldots, A_2$). Calculating the SVD of $\mathbf{Y}_a$ then leads to:

$$\mathbf{Y}_a = \mathbf{U}_a \mathbf{\Sigma}_a \mathbf{V}_a^H,$$

where $\mathbf{\Sigma}_a$ is a diagonal matrix (diagonal elements in descending order) of the same size as $\mathbf{Y}_a$ and $\mathbf{U}_a$ and $\mathbf{V}_a$ are square unitary matrices. Because of assumptions 7.2.1 and 7.2.2, the $(Q+1)M \times KN$ matrix $\mathbf{Y}_a$ has rank $r$ and, defining the $KN \times r$ matrix $\mathbf{V}_a^s$ as the collection of the first $r$ columns of $\mathbf{V}_a$ (we define the collection of the last $KN-r$ columns of $\mathbf{V}_a$ as $\mathbf{V}_a^n$), the rows of $\mathbf{V}_a^{sH}$ form an orthonormal basis for the row space of $\mathbf{X}_a$. Hence, there exists a $1 \times r$ linear chip equalizer $\mathbf{f}_{1,a}^s$, for which

$$\mathbf{f}_{1,a}^s \mathbf{V}_a^{sH} - \mathbf{x}_{1,0} = \mathbf{0}, \tag{7.13}$$

and this linear chip equalizer $\mathbf{f}_{1,a}^s$ is unique. The known code information that is in $\mathbf{x}_{1,0}$ (see (7.9)) then allows us to write (7.13) as

$$\mathbf{f}_{1,a}^s \mathbf{V}_a^{sH} - \mathbf{s}_1 \mathbf{C}_1 = \mathbf{0}.$$

This can be derived for the processing delays $a = A_1, A_1+1, \ldots, A_2$. All these results can then be combined, leading to

$$f_1^s \mathcal{V}^{sH} - \mathbf{s}_1 \mathcal{C}_1 = \mathbf{0}, \tag{7.14}$$

where $f_1^s$ is the $1 \times Ar$ linear *super chip equalizer*, defined as

$$f_1^s = \begin{bmatrix} \mathbf{f}_{1,A_1}^s & \cdots & \mathbf{f}_{1,A_2}^s \end{bmatrix},$$

$\mathcal{V}^s$ is the $AKN \times Ar$ matrix, given by

$$\mathcal{V}^s = \begin{bmatrix} \mathbf{V}_{A_1}^s & & \\ & \ddots & \\ & & \mathbf{V}_{A_2}^s \end{bmatrix},$$

and $\mathcal{C}_1$ is the $K \times AKN$ matrix, given by

$$\mathcal{C}_1 = \begin{bmatrix} \mathbf{C}_1 & \cdots & \mathbf{C}_1 \end{bmatrix}. \tag{7.15}$$

For a space $\Gamma$ that contains the vector $\mathbf{s}_1$ (e.g., $\Omega^{1\times K}$, $\mathbb{C}^{1\times K}$ or $\mathbb{R}^{1\times K}$), we then introduce the following condition:

**Condition 7.2.3** For any vector $\mathbf{s}'_1$ in $\Gamma$, linearly independent of $\mathbf{s}_1$, there exists an input matrix $\mathbf{X}_a$, with $A_1 \leq a \leq A_2$, such that $\mathbf{x}'_{1,0} = \mathbf{s}'_1\mathbf{C}_1$ does not lie in the row space of $\mathbf{X}_a$. □

Using this condition, we have the following identifiability result:

**Theorem 7.2.4** *Under assumptions 7.2.1 and 7.2.2,* (7.14) *has a unique (up to a scaling factor admitted by* $\Gamma$*) solution* $\mathbf{s}_1$ *in* $\Gamma$ *for the data symbols, if and only if condition 7.2.3 is satisfied.*

**Proof:** Under assumptions 7.2.1 and 7.2.2, we know that $\mathbf{s}_1$ is a solution of (7.14) in $\Gamma$ for the data symbols. We now prove that condition 7.2.3 is a necessary and sufficient condition for $\mathbf{s}_1$ to be a unique (up to a scaling factor admitted by $\Gamma$) solution in $\Gamma$ for the data symbols.

We first prove that condition 7.2.3 is a necessary condition. Suppose that there exists a vector $\mathbf{s}'_1$ in $\Gamma$, linearly independent of $\mathbf{s}_1$, such that $\mathbf{x}'_{1,0} = \mathbf{s}'_1\mathbf{C}_1$ lies in the row space of $\mathbf{X}_a$, for $a = A_1, A_1+1, \ldots, A_2$. Because the rows of $\mathbf{V}_a^{sH}$ form an orthonormal basis for the row space of $\mathbf{X}_a$, for $a = A_1, A_1+1, \ldots, A_2$ (this is due to assumptions 7.2.1 and 7.2.2), it is then clear that there exists a set of linear chip equalizers $\{\mathbf{f}'^s_{1,a}\}_{a=A_1}^{A_2}$, such that

$$\mathbf{f}'^s_{1,a}\mathbf{V}_a^{sH} - \mathbf{x}'_{1,0} = \mathbf{f}'^s_{1,a}\mathbf{V}_a^{sH} - \mathbf{s}'_1\mathbf{C}_1 = \mathbf{0},$$
$$\text{for } a = A_1, A_1+1, \ldots, A_2.$$

This means that $\mathbf{s}'_1$ is also a solution of (7.14) in $\Gamma$ for the data symbols.

We then prove that condition 7.2.3 is a sufficient condition. Suppose that there exists a vector $\mathbf{s}'_1$ in $\Gamma$, linearly independent of $\mathbf{s}_1$, such that $\mathbf{s}'_1$ is also a solution of (7.14) in $\Gamma$ for the data symbols. This means that there exists a set of linear chip equalizers $\{\mathbf{f}'^s_{1,a}\}_{a=A_1}^{A_2}$, such that

$$\mathbf{f}'^s_{1,a}\mathbf{V}_a^{sH} - \mathbf{s}'_1\mathbf{C}_1 = \mathbf{f}'^s_{1,a}\mathbf{V}_a^{sH} - \mathbf{x}'_{1,0} = \mathbf{0},$$
$$\text{for } a = A_1, A_1+1, \ldots, A_2.$$

Because the rows of $\mathbf{V}_a^{sH}$ form an orthonormal basis for the row space of $\mathbf{X}_a$, for $a = A_1, A_1+1, \ldots, A_2$ (this is due to assumptions 7.2.1 and 7.2.2), it is then

clear that $\mathbf{x}'_{1,0} = \mathbf{s}'_1\mathbf{C}_1$ lies in the row space of $\mathbf{X}_a$, for $a = A_1, A_1+1, \ldots, A_2$. This concludes the proof. □

Under assumption 7.2.2, it is easy to show that condition 7.2.3 is equivalent with

$$\begin{bmatrix} \mathcal{X} \\ \mathcal{C}_1 \end{bmatrix}$$

having a one-dimensional left null space or equivalently having rank $Ar+K-1$, where

$$\mathcal{X} = \begin{bmatrix} \mathbf{X}_{A_1} & & \\ & \ddots & \\ & & \mathbf{X}_{A_2} \end{bmatrix}.$$

It is clear that this can only be satisfied if

$$A(KN - r) \geq K - 1. \tag{7.16}$$

**Remark 7.2.5** Assume that $L_1 \leq L_j$, for $j = 2, 3, \ldots, J$. Further, assume that there exists a vector $\mathbf{s}'_1$ in $\Gamma$, linearly independent of $\mathbf{s}_1$, such that $\mathbf{x}'_{1,0} = \mathbf{s}'_1\mathbf{C}_1$ is linearly dependent of $\{\mathbf{x}_{j,\delta_j}\}_{j=1}^{J}$. Then, it is clear that, because the vectors $\{\mathbf{x}_{j,\delta_j}\}_{j=1}^{J}$ are all rows of every input matrix from the set $\{\mathbf{X}_a\}_{a=-Q}^{L_1}$, condition 7.2.3 will never be satisfied, irrespective of $A_1$ and $A_2$, with $-Q \leq A_1 \leq A_2 \leq L_1$. This means that if condition 7.2.3 is not satisfied for $A = 1$, it could very well not be satisfied for $A > 1$. That is why the design of $K$ and $N$ should better not be related to $A$. Instead of (7.16), we therefore take

$$K(N-1) \geq r - 1 \tag{7.17}$$

as a design criterion for $K$ and $N$. □

Let us now assume that additive noise is present in $\mathbf{Y}_a$ ($a = A_1, A_1+1, \ldots, A_2$). Calculating the SVD of $\mathbf{Y}_a$ then leads to

$$\mathbf{Y}_a = \hat{\mathbf{U}}_a\hat{\mathbf{\Sigma}}_a\hat{\mathbf{V}}_a^H,$$

where $\hat{\mathbf{\Sigma}}_a$ is a diagonal matrix (diagonal elements in descending order) of the same size as $\mathbf{Y}_a$ and $\hat{\mathbf{U}}_a$ and $\hat{\mathbf{V}}_a$ are square unitary matrices. For an estimate $\hat{r}$ of the system order $r$, let us then define the $KN \times \hat{r}$ matrix $\hat{\mathbf{V}}_a^s$ as the collection of the first $\hat{r}$ columns of $\hat{\mathbf{V}}_a$ (we define the collection of the last $KN-\hat{r}$ columns of $\hat{\mathbf{V}}_a$ as $\hat{\mathbf{V}}_a^n$). We then solve the following minimization problem:

$$\boxed{\begin{array}{c} \min_{\mathbf{s}_1, \hat{f}_1^s} \{\|\hat{f}_1^s\hat{\mathcal{V}}^{sH} - \mathbf{s}_1\mathcal{C}_1\|^2\}, \\ \text{s.t. a non-triviality constraint imposed on } \mathbf{s}_1 \text{ and/or } \hat{f}_1^s, \end{array}} \tag{7.18}$$

where $\hat{f}_1^s$ represents a $1 \times A\hat{r}$ linear super chip equalizer and $\hat{\mathcal{V}}^s$ is the $AKN \times A\hat{r}$ matrix, given by

$$\hat{\mathcal{V}}^s = \begin{bmatrix} \hat{\mathbf{V}}^s_{A_1} & & \\ & \ddots & \\ & & \hat{\mathbf{V}}^s_{A_2} \end{bmatrix}.$$

### 7.2.2 Non-Subspace Framework

Let us first, for the sake of clarity, assume that there is no additive noise present in $\mathbf{Y}_a$ $(a = A_1, A_1+1, \ldots, A_2)$. Because of assumptions 7.2.1 and 7.2.2, the rows of $\mathbf{Y}_a$ span the row space of $\mathbf{X}_a$. Hence, there exists a $1 \times (Q+1)M$ linear chip equalizer $\mathbf{f}_{1,a}$, for which

$$\mathbf{f}_{1,a}\mathbf{Y}_a - \mathbf{x}_{1,0} = \mathbf{0}, \tag{7.19}$$

and this linear chip equalizer $\mathbf{f}_{1,a}$ is a ZF linear chip equalizer with $(Q+1)M - r$ degrees of freedom (hence, this linear chip equalizer $\mathbf{f}_{1,a}$ is only unique when $(Q+1)M = r$). The known code information that is in $\mathbf{x}_{1,0}$ (see (7.9)) then allows us to write (7.19) as

$$\mathbf{f}_{1,a}\mathbf{Y}_a - \mathbf{s}_1\mathbf{C}_1 = \mathbf{0}.$$

This can be derived for the processing delays $a = A_1, A_1+1, \ldots, A_2$. All these results can then be combined, leading to

$$f_1\mathcal{Y} - \mathbf{s}_1\mathcal{C}_1 = \mathbf{0}, \tag{7.20}$$

where $f_1$ is the $1 \times A(Q+1)M$ linear super chip equalizer, defined as

$$f_1 = \begin{bmatrix} \mathbf{f}_{1,A_1} & \cdots & \mathbf{f}_{1,A_2} \end{bmatrix},$$

and $\mathcal{Y}$ is the $A(Q+1)M \times AKN$ matrix, given by

$$\mathcal{Y} = \begin{bmatrix} \mathbf{Y}_{A_1} & & \\ & \ddots & \\ & & \mathbf{Y}_{A_2} \end{bmatrix}.$$

It is easy to show that the identifiability result presented in theorem 7.2.4 also holds for (7.20).

Let us now assume that additive noise is present in $\mathbf{Y}_a$ $(a = A_1, A_1+1, \ldots, A_2)$. We then solve the following minimization problem:

$$\boxed{\begin{array}{c} \min_{\mathbf{s}_1, f_1}\{\|f_1\mathcal{Y} - \mathbf{s}_1\mathcal{C}_1\|^2\}, \\ \text{s.t. a non-triviality constraint imposed on } \mathbf{s}_1 \text{ and/or } f_1. \end{array}} \tag{7.21}$$

**Remark 7.2.6** Like in (7.18), if we do not impose a non-triviality constraint on $\mathbf{s}_1$, we have to impose a non-triviality constraint on $f_1$. However, in contrast with (7.18), where a non-triviality constraint for $\hat{f}_1^s$ should be a constraint that prevents $\hat{f}_1^s$ from being $\mathbf{0}$, a non-triviality constraint for $f_1$ should be a constraint that prevents $f_1^H$ from lying in the left null space of $\mathcal{G}$. □

### 7.2.3 Constraints

In the following two sections, we discuss different block processing methods based on (7.18) and (7.21). We focus on direct blind symbol estimation as well as direct blind equalizer estimation and make use of the following constraints:

C1 Monic constraint on the data symbols:

$$s_1[0] = 1.$$

C2 Unit norm constraint on the data symbols:

$$\|\mathbf{s}_1\|^2 = 1.$$

C3 Unit norm constraint on the linear super chip equalizer:

$$\|\hat{f}_1^s\|^2 = 1 \quad \text{or} \quad \|f_1\|^2 = 1.$$

Note that in the non-subspace framework constraint C3 does not prevent $f_1^H$ from lying in the left null space of $\mathcal{G}$ as we discussed in remark 7.2.6. However, it makes the minimization problem easier to solve. We come back to this in remark 7.4.4.

Applying constraint C1 will result in a least squares (LS) algorithm. Applying constraint C2 or C3 will result in a total least squares (TLS) algorithm (or mixed LS/TLS algorithm).

## 7.3 Direct Blind Symbol Estimation

In this section, we discuss two block processing methods for direct blind symbol estimation.

### 7.3.1 Block Method BM1

This method is based on the elimination of $\hat{f}_1^s$ from (7.18). Hence, we do not impose a non-triviality constraint on $\hat{f}_1^s$. Solving (7.18) for $\hat{f}_1^s$ leads to the

following relation:

$$\hat{f}_1^s = \mathbf{s}_1 \mathcal{C}_1 \hat{\mathcal{V}}^s.$$

Substituting the above result in (7.18), we obtain the following minimization problem:

$$\begin{array}{c} \min\limits_{\mathbf{s}_1} \{\|\mathbf{s}_1 \mathcal{C}_1 (\mathbf{I} - \hat{\mathcal{V}}^s \hat{\mathcal{V}}^{sH})\|^2\}, \\ \text{s.t. a non-triviality constraint imposed on } \mathbf{s}_1, \end{array} \tag{7.22}$$

which is equivalent to

$$\boxed{\begin{array}{c} \min\limits_{\mathbf{s}_1} \{\|\mathbf{s}_1 \mathcal{C}_1 \hat{\mathcal{V}}^n\|^2\}, \\ \text{s.t. a non-triviality constraint imposed on } \mathbf{s}_1, \end{array}} \tag{7.23}$$

where $\hat{\mathcal{V}}^n$ is the $AKN \times A(KN - \hat{r})$ matrix, given by

$$\hat{\mathcal{V}}^n = \begin{bmatrix} \hat{\mathbf{V}}_{A_1}^n & & \\ & \ddots & \\ & & \hat{\mathbf{V}}_{A_2}^n \end{bmatrix}.$$

Let us, for instance, focus on constraints C1 and C2.

**Example 7.3.1 (constraint C1)** If we use constraint C1 as a non-triviality constraint, (7.23) becomes the following minimization problem:

$$\boxed{\min_{\dot{\mathbf{s}}_1} \{\|\dot{\mathbf{s}}_1 \mathcal{C}_1(2:K,:)\hat{\mathcal{V}}^n - \mathcal{C}_1(1,:)\hat{\mathcal{V}}^n\|^2\},} \tag{7.24}$$

where $\dot{\mathbf{s}}_1$ is defined as

$$\dot{\mathbf{s}}_1 = \begin{bmatrix} s_1[1] & s_1[2] & \cdots & s_1[K-1] \end{bmatrix}.$$

This LS problem can be solved by an LS algorithm. Applying the Householder QRD [GVL89], the computational complexity is

$$8(K-1)^2(A(KN - \hat{r}) - (K-1)/3) \text{ flops.} \tag{7.25}$$

□

**Example 7.3.2 (constraint C2)** If we use constraint C2 as a non-triviality constraint, (7.23) becomes the following minimization problem:

$$\boxed{\min_{\mathbf{s}_1} \{\|\mathbf{s}_1 \mathcal{C}_1 \hat{\mathcal{V}}^n\|^2\}, \quad \text{s.t. } \|\mathbf{s}_1\|^2 = 1.} \tag{7.26}$$

A solution for $\mathbf{s}_1$ of this problem is given by the Hermitian transpose of the left singular vector of $\mathcal{C}_1\hat{\mathcal{V}}^n$ corresponding to the smallest singular value. This problem resembles a TLS problem and can be solved by a TLS algorithm [VHV91]. Applying the SVD approach presented in [Cha82], the computational complexity is

$$8A(KN-\hat{r})K^2+25K^3 \text{ flops.} \tag{7.27}$$

Because $\mathcal{C}_1\mathcal{C}_1^H = A\mathbf{I}$ (see (7.7)), the norm in (7.22) can also be written as

$$\|\mathbf{s}_1\mathcal{C}_1(\mathbf{I}-\hat{\mathcal{V}}^s\hat{\mathcal{V}}^{sH})\|^2 = A\|\mathbf{s}_1\|^2 - \|\mathbf{s}_1\mathcal{C}_1\hat{\mathcal{V}}^s\|^2.$$

Therefore, if we use constraint C2 as a non-triviality constraint, (7.22) can be written as

$$\boxed{\max_{\mathbf{s}_1}\{\|\mathbf{s}_1\mathcal{C}_1\hat{\mathcal{V}}^s\|^2\}, \quad \text{s.t. } \|\mathbf{s}_1\|^2=1.} \tag{7.28}$$

A solution for $\mathbf{s}_1$ of this problem is given by the Hermitian transpose of the left singular vector of $\mathcal{C}_1\hat{\mathcal{V}}^s$ corresponding to the largest singular value. Since (7.28) is equivalent to (7.26), the left singular vector of $\mathcal{C}_1\hat{\mathcal{V}}^n$ corresponding to the smallest singular value is equal to the left singular vector of $\mathcal{C}_1\hat{\mathcal{V}}^s$ corresponding to the largest singular value (see also Appendix A of [vdVTP97]). We refer to (7.28) as the *indirect version* of method BM1 with constraint C2 (as opposed to the *direct version* of (7.26)). Using the SVD approach presented in [Cha82], the computational complexity associated with this indirect version is

$$\begin{cases} 8A\hat{r}K^2+25K^3 \text{ flops}, & \text{if } K \le A\hat{r} \\ 8KA^2\hat{r}^2+25A^3\hat{r}^3 \text{ flops}, & \text{if } K > A\hat{r} \end{cases}. \tag{7.29}$$

Note that this indirect version can lead to a reduction in computational complexity compared to the direct version. □

### 7.3.2 Block Method BM2

We consider (7.21) as a minimization problem in $\mathbf{s}_1$. Let us, for instance, focus on constraint C1.

**Example 7.3.3 (constraint C1)** If we use constraint C1 as a non-triviality constraint, (7.21) becomes the following minimization problem:

$$\boxed{\min_{\dot{\mathbf{s}}_1, f_1}\{\| \begin{bmatrix} f_1 & \dot{\mathbf{s}}_1 \end{bmatrix} \begin{bmatrix} \mathcal{Y} \\ -\mathcal{C}_1(2:K,:) \end{bmatrix} - \mathcal{C}_1(1,:)\|^2\}.} \tag{7.30}$$

This LS problem can be solved by an LS algorithm (only the last $K-1$ components of the solution of (7.30) are of interest). Applying the Householder QRD [GVL89], the computational complexity is

$$8(A(Q+1)M+K-1)^2(AKN-(A(Q+1)M+K-1)/3) \text{ flops.} \quad (7.31)$$

□

## 7.4 Direct Blind Equalizer Estimation

In this section, we discuss three block processing methods for direct blind equalizer estimation.

### 7.4.1 Block Method BM3

This method is based on the elimination of $\mathbf{s}_1$ from (7.18). Hence, we do not impose a non-triviality constraint on $\mathbf{s}_1$. Solving (7.18) for $\mathbf{s}_1$ leads to the following relation:

$$\mathbf{s}_1 = \hat{f}_1^s \hat{\mathcal{V}}^{sH} \mathcal{C}_1^{\dagger}.$$

Since $\mathcal{C}_1^{\dagger} = A^{-1}\mathcal{C}_1^H$ (see (7.7)), this can also be written as

$$\mathbf{s}_1 = A^{-1} \hat{f}_1^s \hat{\mathcal{V}}^{sH} \mathcal{C}_1^H. \quad (7.32)$$

Substituting the above result in (7.18), we obtain

$$\boxed{\begin{array}{c} \min_{\hat{f}_1^s} \{\|\hat{f}_1^s \hat{\mathcal{V}}^{sH}(\mathbf{I} - A^{-1}\mathcal{C}_1^H\mathcal{C}_1)\|^2\}, \\ \text{s.t. a non-triviality constraint imposed on } \hat{f}_1^s. \end{array}} \quad (7.33)$$

Let us, for instance, focus on constraint C3.

**Example 7.4.1 (constraint C3)** If we use constraint C3 as a non-triviality constraint, (7.33) becomes the following minimization problem:

$$\boxed{\min_{\hat{f}_1^s} \{\|\hat{f}_1^s \hat{\mathcal{V}}^{sH}(\mathbf{I} - A^{-1}\mathcal{C}_1^H\mathcal{C}_1)\|^2\}, \quad \text{s.t. } \|\hat{f}_1^s\|^2 = 1.} \quad (7.34)$$

A solution for $\hat{f}_1^s$ of this problem is given by the Hermitian transpose of the left singular vector of $\hat{\mathcal{V}}^{sH}(\mathbf{I} - A^{-1}\mathcal{C}_1^H\mathcal{C}_1)$ corresponding to the smallest singular value. This problem again resembles a TLS problem and can be solved by a

TLS algorithm [VHV91]. Applying the SVD approach presented in [Cha82], the computational complexity is

$$8AKNA^2\hat{r}^2 + 25A^3\hat{r}^3 \text{ flops.} \tag{7.35}$$

Like in example 7.3.2, because $\mathcal{C}_1\mathcal{C}_1^H = A\mathbf{I}$ (see (7.7)), the norm in (7.33) can also be written as

$$\|\hat{f}_1^s\hat{\mathcal{V}}^{sH}(\mathbf{I} - A^{-1}\mathcal{C}_1^H\mathcal{C}_1)\|^2 = \|\hat{f}_1^s\|^2 - A^{-1}\|\hat{f}_1^s\hat{\mathcal{V}}^{sH}\mathcal{C}_1^H\|^2.$$

Therefore, if we use constraint C3 as a non-triviality constraint, (7.33) can be written as

$$\boxed{\max_{\hat{f}_1^s}\{\|\hat{f}_1^s\hat{\mathcal{V}}^{sH}\mathcal{C}_1^H\|^2\}, \quad \text{s.t. } \|\hat{f}_1^s\|^2 = 1.} \tag{7.36}$$

A solution for $\hat{f}_1^s$ of this problem is given by the Hermitian transpose of the left singular vector of $\hat{\mathcal{V}}^{sH}\mathcal{C}_1^H$ corresponding to the largest singular value. Since (7.36) is equivalent to (7.34), the left singular vector of $\hat{\mathcal{V}}^{sH}(\mathbf{I}-A^{-1}\mathcal{C}_1^H\mathcal{C}_1)$ corresponding to the smallest singular value is equal to the left singular vector of $\hat{\mathcal{V}}^{sH}\mathcal{C}_1^H$ corresponding to the largest singular value (see also Appendix A of [vdVTP97]). We refer to (7.36) as the *indirect version* of method BM3 with constraint C3 (as opposed to the *direct version* of (7.34)). Using the SVD approach presented in [Cha82], the computational complexity associated with this indirect version is the same as (7.29). Note that this indirect version can lead to a reduction in computational complexity compared to the direct version. □

A solution for $\mathbf{s}_1$ can then be found by despreading the output of the linear super chip equalizer $\hat{f}_1^s$, which does not increase the complexity very much. This despreading operation can be described as

$$\mathbf{s}_1 = A^{-1}\hat{f}_1^s\hat{\mathcal{V}}^{sH}\mathcal{C}_1^H. \tag{7.37}$$

where the factor $A^{-1}$ is due to (7.7). Note that this operation corresponds to (7.32).

**Remark 7.4.2** From example 7.4.1, we know that the Hermitian transpose of the left singular vector of $\hat{\mathcal{V}}^{sH}\mathcal{C}_1^H$ corresponding to the largest singular value represents a solution for $\hat{f}_1^s$ of (7.36). It is then clear that the corresponding solution for $\mathbf{s}_1$, calculated using (7.37), is up to a real scaling factor equal to the Hermitian transpose of the left singular vector of $\mathcal{C}_1\hat{\mathcal{V}}^s$ corresponding to the largest singular value. From example 7.3.2, we then know that this solution for $\mathbf{s}_1$ (when normalized) also represents a solution of (7.28). As a result, *method BM3 with constraint C3 has exactly the same performance as method BM1 with constraint C2.* □

### 7.4.2 Block Method BM4

This method is based on the elimination of $\mathbf{s}_1$ from (7.21). Hence, we do not impose a non-triviality constraint on $\mathbf{s}_1$. Solving (7.21) for $\mathbf{s}_1$ leads to the following relation:

$$\mathbf{s}_1 = f_1 \mathcal{Y} \mathcal{C}_1^{\dagger}.$$

Since $\mathcal{C}_1^{\dagger} = A^{-1}\mathcal{C}_1^H$ (this is due to (7.7)), this can also be written as

$$\mathbf{s}_1 = A^{-1} f_1 \mathcal{Y} \mathcal{C}_1^H. \tag{7.38}$$

Substituting the above result in (7.21), we obtain

$$\boxed{\begin{array}{c} \min_{f_1}\{\|f_1 \mathcal{Y}(\mathbf{I} - A^{-1}\mathcal{C}_1^H\mathcal{C}_1)\|^2\}, \\ \text{s.t. a non-triviality constraint imposed on } f_1. \end{array}} \tag{7.39}$$

Let us, for instance, focus on constraint C3.

**Example 7.4.3 (constraint C3)** If we use constraint C3 as a non-triviality constraint, (7.39) becomes the following minimization problem:

$$\boxed{\min_{f_1}\{\|f_1 \mathcal{Y}(\mathbf{I} - A^{-1}\mathcal{C}_1^H\mathcal{C}_1)\|^2\}, \quad \text{s.t. } \|f_1\|^2 = 1.} \tag{7.40}$$

A solution for $f_1$ is then given by the Hermitian transpose of the left singular vector of $\mathcal{Y}(\mathbf{I} - A^{-1}\mathcal{C}_1^H\mathcal{C}_1)$ corresponding to the smallest singular value. This problem once more resembles a TLS problem and can be solved by a TLS algorithm [VHV91]. Applying the SVD approach presented in [Cha82], the computational complexity is given by

$$8AKNA^2(Q+1)^2M^2 + 25A^3(Q+1)^3M^3 \text{ flops}. \tag{7.41}$$

□

A solution for $\mathbf{s}_1$ can then be found by despreading the output of the linear super chip equalizer $f_1$, which does not increase the complexity very much. This despreading operation can be described as

$$\mathbf{s}_1 = A^{-1} f_1 \mathcal{Y} \mathcal{C}_1^H, \tag{7.42}$$

where the factor $A^{-1}$ is again due to (7.7). Note that this operation corresponds to (7.38).

**Remark 7.4.4** It is clear that method BM4 is strongly related to method BM3. However, in contrast with method BM3, method BM4 has to deal with the problem discussed in remark 7.2.6. Since method BM4 solves (7.21) without imposing a non-triviality constraint on $\mathbf{s}_1$, we have to impose a constraint on $f_1$ that prevents $f_1^H$ from lying in the left null space of $\mathcal{G}$. Constraint C3 does not prevent this and as a result *method BM4 with constraint C3 does not have such a good performance* (see the simulation results presented in section 7.10). Imposing the constraint $\|f_1\mathcal{Y}\|^2 = 1$ shows robustness against this problem (in the noiseless case, $f_1^H$ can then not lie in the left null space of $\mathcal{G}$). However, method BM4 with constraint $\|f_1\mathcal{Y}\|^2 = 1$ has actually already been discussed, since it is easy to show that *method BM4 with constraint* $\|f_1\mathcal{Y}\|^2 = 1$ *is equivalent with method BM3 with constraint C3, taking* $\hat{r} = (Q+1)M$ (assume that $\mathbf{Y}_a$ has full row rank $(Q+1)M$, for $a = A_1, A_1+1, \ldots, A_2$). □

### 7.4.3 Block Method BM5

We consider (7.21) as a minimization problem in $f_1$. Let us, for instance, focus on constraint C1.

**Example 7.4.5 (constraint C1)** If we use constraint C1 as a non-triviality constraint, (7.21) becomes the following minimization problem:

$$\boxed{\min_{\dot{\mathbf{s}}_1, f_1}\{\| \begin{bmatrix} \dot{\mathbf{s}}_1 & f_1 \end{bmatrix} \begin{bmatrix} -\mathcal{C}_1(2:K,:) \\ \mathcal{Y}_k \end{bmatrix} - \mathcal{C}_{1,k}(1,:)\|^2\}.} \tag{7.43}$$

This LS problem can be solved by an LS algorithm (only the last $A(Q+1)M$ components of the solution of (7.43) are of interest). Applying the Householder QRD [GVL89], the computational complexity is the same as (7.31). □

A solution for $\mathbf{s}_1$ can then again be found by despreading the output of the linear super chip equalizer $f_1$ (see (7.42)). As already mentioned, this operation does not increase the computational complexity very much.

**Remark 7.4.6** If (7.21) is solved for $f_1$ using a non-triviality constraint imposed on $\mathbf{s}_1$ (as in example 7.4.5), the solution for $\mathbf{s}_1$, computed using (7.42), does not necessarily satisfy this non-triviality constraint. □

## 7.5 Further Discussion

### 7.5.1 Symbol Estimation

Applying constraint C1, C2 or C3, we still have to transform the obtained solution for $\mathbf{s}_1$ to a hard estimate of $\mathbf{s}_1$. This is explained next. The obtained solution for $\mathbf{s}_1$ can be interpreted as an estimate of $\mathbf{s}_1^o$, the solution for $\mathbf{s}_1$ in the noiseless case, taking $\hat{r} = r$ in the subspace framework. Hence, we denote this solution as $\hat{\mathbf{s}}_1^o$. As an estimate of $\mathbf{s}_1$ we then consider

$$\hat{\mathbf{s}}_1 = \hat{\alpha}_1 \hat{\mathbf{s}}_1^o,$$

where $\hat{\alpha}_1$ is an estimate of $\alpha_1$, which is given by $\mathbf{s}_1 = \alpha_1 \mathbf{s}_1^o$. In practice $\alpha_1$ can be estimated from some short known headers in the data symbol sequence. However, for simplicity we will estimate $\alpha_1$ as

$$\hat{\alpha}_1 = \mathbf{s}_1 \hat{\mathbf{s}}_1^{o\dagger}.$$

This leads to an estimate $\hat{\mathbf{s}}_1$ that is optimal in the LS sense (since $\mathbf{s}_1$ is unknown, this is of course not feasible in practice). A hard estimate $\underline{\hat{\mathbf{s}}}_1$ of $\mathbf{s}_1$ is then calculated as

$$\underline{\hat{\mathbf{s}}}_1 = \underset{\Omega^{1\times K}}{\text{proj}}\{\hat{\mathbf{s}}_1\},$$

where

$$\underset{\Upsilon}{\text{proj}}\{\cdot\}$$

represents a mapping on the closest (in Euclidean distance) element of $\Upsilon$.

### 7.5.2 Relations with Existing Algorithms

The proposed block processing methods based on direct blind symbol estimation have to the best of our knowledge not yet been studied in the literature. Some of the proposed block processing methods based on direct blind equalizer estimation, on the other hand, have already been studied in the literature. In [LX96a], method BM5 with constraint C1 is discussed, while in [LZ97], method BM4 with constraint C3 is discussed. However, note that only one processing delay ($A = 1$) is considered in [LX96a, LZ97], while we consider the general case of $A \geq 1$.

Next, we show some relations with existing algorithms for a **TDMA system with a single user per time channel** (no coding). To model such a system, let us consider $J = 1$, $N = 1$, $c_j[n] = 1$, for $n = 0, 1, \ldots, \rho - 1$, and $c_j[n] = 0$,

for $n < 0$ and $n \geq \rho$ (the value of $\rho \geq 1$ is unimportant). This means that the $K \times AK$ matrix $\mathcal{C}_1$, given by (7.15), looks like

$$\mathcal{C}_1 = \begin{bmatrix} \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} \end{bmatrix}.$$

The direct version of method BM1 with constraint C2 (see (7.26)) then becomes

$$\min_{\mathbf{s}_1} \{\|\mathbf{s}_1 \hat{\mathcal{W}}^n\|^2\}, \quad \text{s.t. } \|\mathbf{s}_1\|^2 = 1, \tag{7.44}$$

where $\hat{\mathcal{W}}^n$ is the $K \times A(K - \hat{r})$ matrix, given by

$$\hat{\mathcal{W}}^n = \begin{bmatrix} \hat{\mathbf{V}}^n_{A_1} & \hat{\mathbf{V}}^n_{A_1+1} & \cdots & \hat{\mathbf{V}}^n_{A_2} \end{bmatrix}.$$

This method corresponds to the single-user version of the subspace intersection (SSI) method presented in [LX97]. The indirect version of method BM1 with constraint C2 (see (7.28)) then becomes

$$\max_{\mathbf{s}_1} \{\|\mathbf{s}_1 \hat{\mathcal{W}}^s\|^2\}, \quad \text{s.t. } \|\mathbf{s}_1\|^2 = 1, \tag{7.45}$$

where $\hat{\mathcal{W}}^s$ is the $K \times A\hat{r}$ matrix, given by

$$\hat{\mathcal{W}}^s = \begin{bmatrix} \hat{\mathbf{V}}^s_{A_1} & \hat{\mathbf{V}}^s_{A_1+1} & \cdots & \hat{\mathbf{V}}^s_{A_2} \end{bmatrix}.$$

This method corresponds to the single-user version of the subspace intersection (SSI) method presented in [vdVTP97].

**Remark 7.5.1** For the TDMA system with a single user per time channel (no coding), it is easy to check that, under assumption 7.2.2, assumption 7.2.3 (see the identifiability result presented in theorem 7.2.4) can only be satisfied for $A_1 = -Q$ and $A_2 = L_1$ (see also [Van99]). Hence, channel order information is required in this case. □

Finally, method BM4 (see (7.39)) then becomes

$$\begin{gathered} \min_{f_1} \{\|f_1 \mathcal{Z}\|^2\}, \\ \text{s.t. a non-triviality constraint imposed on } f_1, \end{gathered} \tag{7.46}$$

where $\mathcal{Z}$ is the $A(Q+1)M \times AK$ matrix, given by

$$\mathcal{Z} = A^{-1} \begin{bmatrix} (A-1)\mathbf{Y}_{A_1} & -\mathbf{Y}_{A_1} & \cdots & -\mathbf{Y}_{A_1} \\ -\mathbf{Y}_{A_1+1} & (A-1)\mathbf{Y}_{A_1+1} & \cdots & -\mathbf{Y}_{A_1+1} \\ \vdots & \vdots & \ddots & \vdots \\ -\mathbf{Y}_{A_2} & -\mathbf{Y}_{A_2} & \cdots & (A-1)\mathbf{Y}_{A_2} \end{bmatrix}.$$

As we will show next, this method is equivalent to the mutually referenced equalizer (MRE) method of [GDM97] (see also [GT99]). The basic idea behind the MRE method is that a set of linear equalizers is applied, the outputs of which serve as 'training sequences' for each other. Basically, this boils down to the following minimization problem [GDM97]:

$$\begin{aligned} &\min_{f_1}\{\|f_1\mathcal{Z}^{mre}\|^2\}, \\ &\text{s.t. a non-triviality constraint imposed on } f_1, \end{aligned} \tag{7.47}$$

where $\mathcal{Z}^{mre}$ is the $A(Q+1)M \times A(A-1)K/2$ matrix, given by

$$\mathcal{Z}^{mre} = \begin{bmatrix} \mathcal{Z}_{A_1}^{mre} & \mathcal{Z}_{A_1+1}^{mre} & \cdots & \mathcal{Z}_{A_2}^{mre} \end{bmatrix},$$

with $\mathcal{Z}_a^{mre}$ $(a = A_1, A_1+1, \ldots, A_2)$ the $A(Q+1)M \times (A_2-a+1)K$ matrix, given by

$$\mathcal{Z}_a^{mre} = \begin{bmatrix} \mathbf{O} & \mathbf{O} & \cdots & \mathbf{O} \\ \vdots & \vdots & & \vdots \\ \mathbf{O} & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{Y}_a & \mathbf{Y}_a & \cdots & \mathbf{Y}_a \\ -\mathbf{Y}_{a+1} & & & \\ & -\mathbf{Y}_{a+2} & & \\ & & \ddots & \\ & & & -\mathbf{Y}_{A_2} \end{bmatrix}.$$

The equivalence between (7.46) and (7.47) can now easily be explained by the fact that

$$A\mathcal{Z}\mathcal{Z}^H = \mathcal{Z}^{mre}\mathcal{Z}^{mreH} = \begin{bmatrix} (A-1)\mathbf{R}_{A_1} & -\mathbf{R}_{A_1,A_1+1} & \cdots & -\mathbf{R}_{A_1,A_2} \\ -\mathbf{R}_{A_1+1,A_1} & (A-1)\mathbf{R}_{A_1+1} & \cdots & -\mathbf{R}_{A_1+1,A_2} \\ \vdots & \vdots & \ddots & \vdots \\ -\mathbf{R}_{A_2,A_1} & -\mathbf{R}_{A_2,A_1+1} & \cdots & (A-1)\mathbf{R}_{A_2} \end{bmatrix},$$

where $\mathbf{R}_a = \mathbf{Y}_a\mathbf{Y}_a^H$, for $a = A_1, A_1+1, \ldots, A_2$, and $\mathbf{R}_{a,a'} = \mathbf{Y}_a\mathbf{Y}_{a'}^H$, for $a, a' = A_1, A_1+1, \ldots, A_2$, with $a \neq a'$. Note that remark 7.5.1 also applies to the MRE method. In [GvdVP99], it was shown that the MRE method with constraint $\|f_1\mathcal{Y}\|^2 = 1$ has exactly the same performance as the single-user version of the SSI method presented in [vdVTP97], taking $\hat{r} = (Q+1)M$ (assume that $\mathbf{Y}_a$ has full row rank $(Q+1)M$, for $a = A_1, A_1+1, \ldots, A_2$). *The combination of the results presented in remarks 7.4.2 and 7.4.4 therefore extends the claim made in [GvdVP99] to a DS-CDMA system.*

For a **TDMA system with multiple users per time channel** (no coding), methods similar to the direct and indirect version of method BM1 with constraint C2 are discussed in [LX97, vdVTP97]. If the users have equal channel orders, these methods consist of two steps. In the first step, the ISI for every user is suppressed. This step is related to the procedure carried out for a TDMA system with a single user per time channel (see (7.44) and (7.45)). In the second step, the MUI present in the resulting data model is suppressed. For this step we can make use of several deterministic blind source separation methods that have been proposed in the literature:

- A well known iterative algorithm that exploits the finite alphabet property of the data symbols is the ILSP algorithm of [TVP96]. However, this algorithm does not necessarily converge to the global minimum. Therefore, several random initializations or one computationally expensive initialization (see the non-iterative algorithms below) may be required to find the actual global minimum.

- Another iterative algorithm that exploits the finite alphabet property of the data symbols can be found in [HX97]. This algorithm, which sequentially estimates each signal, is less complex than ILSP. However, like the ILSP algorithm, it does not necessarily converge to the global minimum.

- Interesting non-iterative algorithms can be found in [vdVP96] (ACMA) for constant modulus constellations and in [vdV97] (RACMA), [AMR95] for a BPSK constellation. Although near-optimum these approaches are computationally expensive.

- Finally, in [LG99], we have proposed a new simple source separation algorithm for a BPSK constellation. Like the algorithms of [vdV97, AMR95], this algorithm is non-iterative. However, in contrast with all the algorithms discussed above it is recursive in time, i.e., the BPSK data symbols can be estimated on the fly.

If the users have unequal channel orders, difficulties occur and a cumbersome iterative procedure is required (see [LX97, vdVTP97]).

The major drawback of the algorithms presented in [LX97, vdVTP97] is that they are rather *sensitive to system order and channel order mismatch* (the latter is related with remark 7.5.1). *In a DS-CDMA system, the use of coding prevents these problems.*

| BM | constr. | complexity (flops) |
|---|---|---|
| BM1 | C1 | $8(K-1)^2(A(KN-\hat{r})-(K-1)/3)$ |
| BM1 | C2 | direct version: $8A(KN-\hat{r})K^2+25K^3$ |
| | | indirect version: $\begin{cases} 8A\hat{r}K^2+25K^3, & \text{if } K \leq A\hat{r} \\ 8KA^2\hat{r}^2+25A^3\hat{r}^3, & \text{if } K > A\hat{r} \end{cases}$ |
| BM2 | C1 | $8(A(Q+1)M+K-1)^2(AKN-(A(Q+1)M+K-1)/3)$ |
| BM3 | C3 | direct version: $8AKNA^2\hat{r}^2+25A^3\hat{r}^3$ |
| | | indirect version: $\begin{cases} 8A\hat{r}K^2+25K^3, & \text{if } K \leq A\hat{r} \\ 8KA^2\hat{r}^2+25A^3\hat{r}^3, & \text{if } K > A\hat{r} \end{cases}$ |
| BM4 | C3 | $8AKNA^2(Q+1)^2M^2+25A^3(Q+1)^3M^3$ |
| BM5 | C1 | $8(A(Q+1)M+K-1)^2(AKN-(A(Q+1)M+K-1)/3)$ |

Table 7.1: Computational complexity of the block processing methods.

### 7.5.3 Other Constraints

We can also make use of a finite alphabet constraint on the data symbols:

$$\mathbf{s}_1 \in \Omega^{1\times K}.$$

Applying this constraint for one of the methods discussed in sections 7.3 and 7.4 will result in an enumeration algorithm.

## 7.6 Computational Complexity

The computational complexities derived in sections 7.3 and 7.4 are summarized in **table 7.1**. Note that for the subspace block processing methods (methods BM1 and BM3), this complexity does not yet include the computational complexity of the subspace preprocessing step. A subspace block processing method requires the set of matrices $\{\hat{\mathbf{V}}_a\}_{a=A_1}^{A_2}$. Two subspace preprocessing approaches can be used (we will refer to them as BP subspace preprocessing approaches since they are used for block processing). The first approach computes the SVD of $\mathbf{Y}_a$, for $a = A_1, A_1+1, \ldots, A_2$, from scratch, e.g., using the algorithm of [DBMS79]. This approach has the following computational complexity:

$$A(16K^2N^2(Q+1)M + 12KN(Q+1)^2M^2) \text{ flops.} \tag{7.48}$$

The second approach first computes the SVD of $\mathbf{Y}_{A_1}$ from scratch and then computes the SVD of $\mathbf{Y}_a$, for $a = A_1+1, A_1+2, \ldots, A_2$, in a recursive fashion, e.g., using a modified version of the algorithm of [MDPM95] that computes the SVD based on a QR up- and downdating step followed by $n_{it}$ Jacobi iterations. This approach has the following computational complexity:

$$\begin{aligned} &16K^2N^2(Q+1)M + 12KN(Q+1)^2M^2 \\ &+(A-1)18K^2N^2 + (A-1)(29+36n_{it})(Q+1)^2M^2 \\ &+(A-1)(24+18n_{it})KN(Q+1)M \text{ flops.} \end{aligned} \tag{7.49}$$

Note that for $A = 1$, the complexity measures of the first and second approach are obviously exactly the same.

**Remark 7.6.1** Note that if $\hat{r} = (Q+1)M$ the complexity of the two BP subspace preprocessing approaches can be decreased because $\hat{\mathbf{V}}_a$ can then be computed from the QRD of $\mathbf{Y}_a$. However, this will not be considered in this work. □

**Remark 7.6.2** To obtain the simulation results presented in section 7.10, we will always make use of the first approach. The results for the second approach are approximately the same, even for $n_{it} = 1$. □

*Numerical example:* We consider the following system parameters: $J = 2$, $M = 4$, $L_1 = 2$, $L_2 = 4$ and $N = 2$. Note that these parameters correspond to system 1 of section 7.10. We further take $Q = 4$.

First, we compare the computational complexity of the two BP subspace preprocessing approaches discussed above (for the second approach, we take $n_{it} = 1$, which usually is sufficient). The results are shown in **figure 7.1**. From this figure it is clear that, for $A > 1$, the second approach is cheaper than the first approach. Note that the complexity of the second approach is almost independent of $A$.

Next, we compare the computational complexity of the different block processing methods discussed in sections 7.3 and 7.4. We only consider $A = 1$ and $A = 3$. For the subspace block processing methods (methods BM1 and BM3) we examine two values of $\hat{r}$: $\hat{r} = r$ (when $r$ is exactly estimated) and $\hat{r} = (Q+1)M$ (when $r$ is not estimated). In **figures 7.2 and 7.3**, we compare the computational complexity of different subspace block processing methods. In the left figures, we compare the complexity of the direct and indirect versions of method BM1 with constraint C2 and method BM3 with constraint C3. From these figures it is clear that the indirect versions are the least demanding. In the right figures, we then compare the complexity of these indirect versions with the complexity of some other subspace block processing methods. From

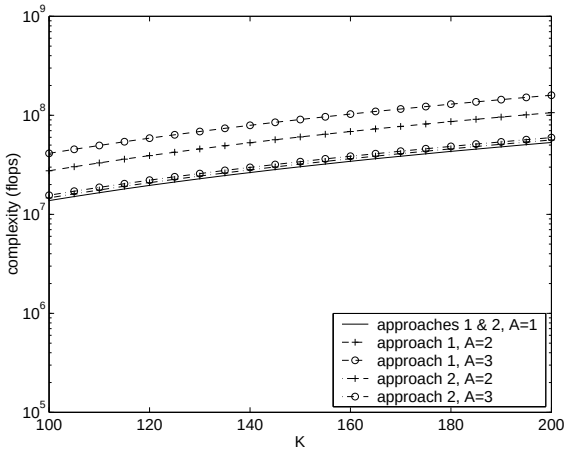



Figure 7.1: Complexity of the BP subspace preprocessing step as a function of $K$ ($J = 2$, $M = 4$, $L_1 = 2$, $L_2 = 4$, $N = 2$, $Q = 4$).

these figures it is clear that the indirect versions of method BM1 with constraint C2 and method BM3 with constraint C3 are cheaper than all the other methods shown in this figure.

In **figure 7.4**, we finally compare the computational complexity of the indirect versions, including the second BP subspace preprocessing approach ($n_{it} = 1$), with the computational complexity of the methods BM2 and BM5 with constraint C1 and method BM4 with constraint C3, which are all non-subspace block processing methods. First of all, we observe that the complexity of the non-subspace block processing methods (especially method BM4 with constraint C3) is more sensitive to $A$ than the complexity of the indirect versions of method BM1 with constraint C2 and method BM3 with constraint C3, including the second BP subspace preprocessing approach. This is due to the fact that the latter is mainly determined by the complexity of this second BP subspace preprocessing approach, which is almost independent of $A$. Second, we notice that for a small $A$ it would be favorable to use method BM4 with constraint C3. However, as discussed in remark 7.4.4, this method does not have such a good performance.

## 7.7 Real Constellation

The block processing methods presented in sections 7.3 and 7.4 are applicable for any modulation format. However, when the data symbols belong to a *real*

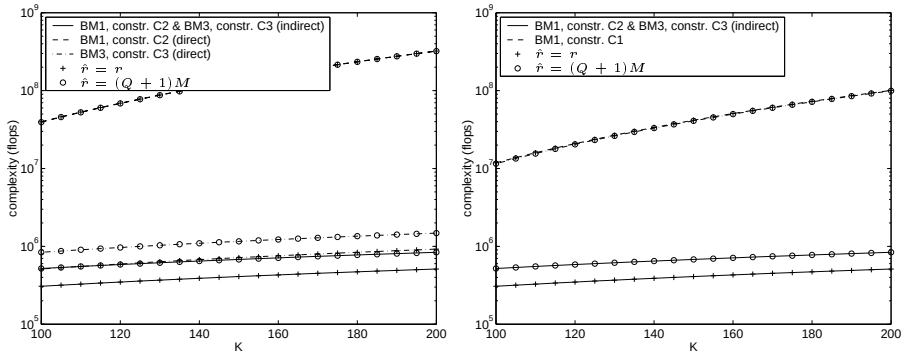



Figure 7.2: Complexity of different subspace block processing methods as a function of $K$ for $A = 1$ ($J = 2$, $M = 4$, $L_1 = 2$, $L_2 = 4$, $N = 2$, $Q = 4$).

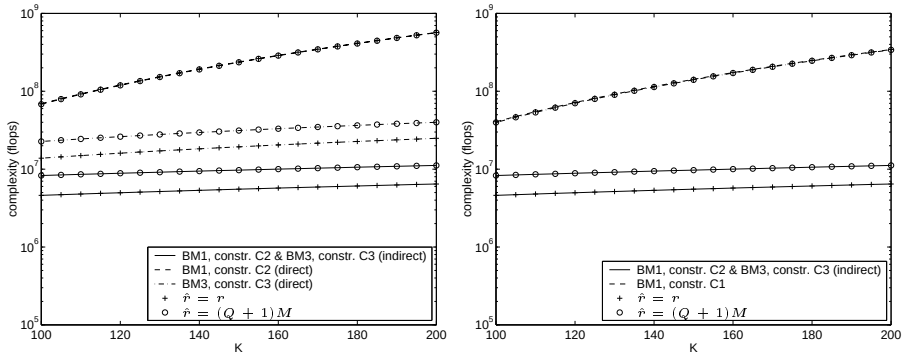



Figure 7.3: Complexity of different subspace block processing methods as a function of $K$ for $A = 3$ ($J = 2$, $M = 4$, $L_1 = 2$, $L_2 = 4$, $N = 2$, $Q = 4$).

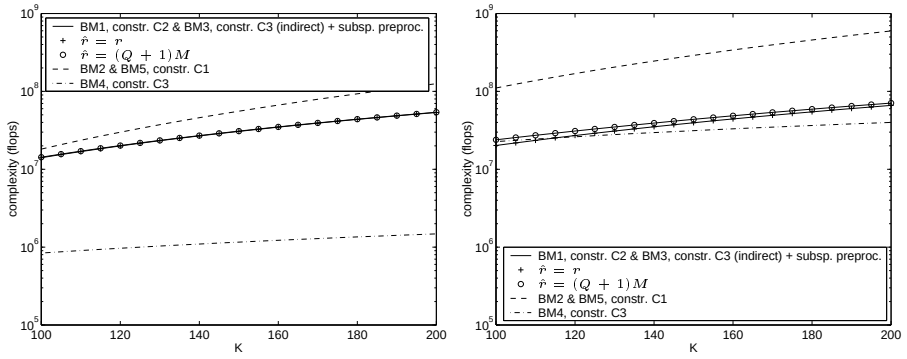



Figure 7.4: Complexity comparison of the indirect versions of method BM1 with constraint C2 and method BM3 with constraint C3, including the second BP subspace preprocessing approach ($n_{it} = 1$), with different non-subspace block processing methods as a function of $K$ for $A = 1$, in the left figure, and $A = 3$, in the right figure ($J = 2$, $M = 4$, $L_1 = 2$, $L_2 = 4$, $N = 2$, $Q = 4$).

*constellation*, we can adapt these methods to exploit the real character of the data symbols, which is discussed in this section. When no coding is used, the knowledge that the constellation is real is usually exploited by splitting the received sequence in its real and imaginary part, hence doubling the number of observations prior to any other operation (see [KOS96, vdVTP97, vdV97, TP97]). Here, we follow a somewhat different approach.

If the data symbols belong to a real constellation, we can rewrite (7.14) as

$$\begin{bmatrix} \Re\{f_1^s\} & \Im\{f_1^s\} \end{bmatrix} \begin{bmatrix} \Re\{\mathcal{V}^{sH}\} & \Im\{\mathcal{V}^{sH}\} \\ -\Im\{\mathcal{V}^{sH}\} & \Re\{\mathcal{V}^{sH}\} \end{bmatrix} - \mathbf{s}_1 \begin{bmatrix} \Re\{\mathcal{C}_1\} & \Im\{\mathcal{C}_1\} \end{bmatrix} = \mathbf{0}. \quad (7.50)$$

It is easy to show that the identifiability result presented in theorem 7.2.4 also holds for (7.50), if we consider condition 7.2.3 with $\Gamma \subseteq \mathbb{R}^{1\times K}$. Under assumption 7.2.2, it is easy to show that condition 7.2.3 with $\Gamma \subseteq \mathbb{R}^{1\times K}$ is equivalent with

$$\begin{bmatrix} \Re\{\mathcal{X}\} & \Im\{\mathcal{X}\} \\ -\Im\{\mathcal{X}\} & \Re\{\mathcal{X}\} \\ \Re\{\mathcal{C}_1\} & \Im\{\mathcal{C}_1\} \end{bmatrix}$$

having a one-dimensional left null space or equivalently having rank $2Ar+K-1$.

1. For *complex code sequences*, it is clear that this can only be satisfied if

$$2A(KN - r) \geq K - 1. \quad (7.51)$$

2. For *real code sequences*, it is clear that this can only be satisfied if

$$A(KN - r) \geq K - 1. \tag{7.52}$$

This is due to the fact that, under assumption 7.2.2,

$$\begin{bmatrix} \Re\{\mathcal{X}\} & \Im\{\mathcal{X}\} \\ -\Im\{\mathcal{X}\} & \Re\{\mathcal{X}\} \\ \Re\{\mathcal{C}_1\} & \Im\{\mathcal{C}_1\} \end{bmatrix} = \begin{bmatrix} \mathcal{X} & \mathbf{O} \\ \mathbf{O} & \mathcal{X} \\ \mathcal{C}_1 & \mathbf{O} \end{bmatrix},$$

can only have a one-dimensional left null space if

$$\begin{bmatrix} \mathcal{X} \\ \mathcal{C}_1 \end{bmatrix}$$

has a one-dimensional left null space.

**Remark 7.7.1** Following a similar reasoning as in remark 7.2.5, we can show that the design of $K$ and $N$ should better not be related to $A$. Instead of (7.51), we therefore take

$$K(2N - 1) \geq 2r - 1 \tag{7.53}$$

as a design criterion for $K$ and $N$, if the code sequences are complex, and instead of (7.52), we therefore take

$$K(N - 1) \geq r - 1 \tag{7.54}$$

as a design criterion for $K$ and $N$, if the code sequences are real. □

**Remark 7.7.2** All the block processing methods proposed in sections 7.3 and 7.4 can easily be adapted to the situation considered in this section. These methods are then referred to as **RC (real constellation) methods**. Mark that constraint C3 is then written as

$$\| \begin{bmatrix} \Re\{\hat{f}_1^s\} & \Im\{\hat{f}_1^s\} \end{bmatrix} \|^2 = 1 \quad \text{or} \quad \| \begin{bmatrix} \Re\{f_1\} & \Im\{f_1\} \end{bmatrix} \|^2 = 1.$$

□

## 7.8 Transmit Diversity

In this section, we describe how the block processing methods presented in sections 7.3 and 7.4 can be adapted when *transmit diversity* is used (multiple transmit antennas per user). We then spread the same data symbol sequence

with different code sequences and transmit the resulting chip sequences through different antennas. Lately, transmit diversity has become a very popular research topic, since it has been shown that for a point-to-point link between a transmitter and a receiver with a fixed number of receive antennas, the optimal capacity linearly increases with the number of transmit antennas, as long as the number of transmit antennas is smaller than or equal to the number of receive antennas (Foschini and Gans [FG98]). A lot of so-called *space-time coding techniques* have been developed to achieve the optimal capacity, such as the space-time coding technique of Foschini [Fos96] (also known as V-BLAST) and the space-time coding technique of Naguib et al [NTSC97]. As mentioned above, we focus on a very simple space-time coding technique that consists of spreading the same data symbol sequence with different code sequences and transmitting the resulting chip sequences through different antennas.

We can still make use of the data model presented in (7.1). Therefore, we have to give a new interpretation to the notations. First of all, the parameter $J$ now denotes the product of the number of users with the number of transmit antennas per user (let's say $M_t > 1$). Hence, we have $J/M_t$ users. Further, $x_{M_t(j_t-1)+m_t}[n]$ ($m_t = 1, 2, \ldots, M_t$ and $j_t = 1, 2, \ldots, J/M_t$) now represents the chip sequence for the $m_t$th transmit antenna of the $j_t$th user. This means that the data symbol sequences $\{s_{M_t(j_t-1)+m_t}[k]\}_{m_t=1}^{M_t}$ are all the same and equal to the data symbol sequence for the $j_t$th user. Hence, we have that $s_{M_t(j_t-1)+1}[k] = s_{M_t(j_t-1)+2}[k] = \cdots = s_{M_t j_t}[k]$, for $j_t = 1, 2, \ldots, J/M_t$. Finally, $\mathbf{g}^{st}_{M_t(j_t-1)+m_t}[n]$ ($m_t = 1, 2, \ldots, M_t$ and $j_t = 1, 2, \ldots, J/M_t$) now represents the discrete-time $M \times 1$ vector channel for the $m_t$th transmit antenna of the $j_t$th user.

To simplify the description of the proposed methods we assume that all the vector channels from the set $\{\mathbf{g}^{st}_{M_t(j_t-1)+m_t}[n]\}_{m_t=1}^{M_t}$ have the same order and delay index. Hence, we assume that $L_{M_t(j_t-1)+1} = L_{M_t(j_t-1)+2} = \cdots = L_{M_t j_t}$ and $\delta_{M_t(j_t-1)+1} = \delta_{M_t(j_t-1)+2} = \cdots = \delta_{M_t j_t}$, for $j_t = 1, 2, \ldots, J/M_t$.

We assume that the first user ($j_t = 1$) is the user of interest and that $\delta_1$ is known at the receiver. We further assume w.l.o.g. that $\delta_1 = 0$. Taking $-Q \leq A_1 \leq A_2 \leq L_1$ ($A = A_2 - A_1 + 1$), we can then, similar to (7.14), write

$$f^s_{m_t} \mathcal{V}^{sH} - \mathbf{s}_1 \mathcal{C}_{m_t} = \mathbf{0}, \quad \text{for } m_t = 1, 2, \ldots, M_t.$$

All these results can be combined, leading to

$$\begin{bmatrix} f^s_1 & \cdots & f^s_{M_t} \end{bmatrix} \begin{bmatrix} \mathcal{V}^{sH} & & \\ & \ddots & \\ & & \mathcal{V}^{sH} \end{bmatrix} - \mathbf{s}_1 \begin{bmatrix} \mathcal{C}_1 & \cdots & \mathcal{C}_{M_t} \end{bmatrix} = \mathbf{0}. \tag{7.55}$$

For a space $\Gamma$ that contains the vector $\mathbf{s}_1$ (e.g., $\Omega^{1\times K}$, $\mathbb{C}^{1\times K}$ or $\mathbb{R}^{1\times K}$), we now introduce the following condition (compare with condition 7.2.3):

**Condition 7.8.1** For any vector $\mathbf{s}_1'$ in $\Gamma$, linearly independent of $\mathbf{s}_1$, there exists an input matrix $\mathbf{X}_a$, with $A_1 \leq a \leq A_2$, and a row $\mathbf{x}'_{m_t,0} = \mathbf{s}_1'\mathbf{C}_{m_t}$, with $1 \leq m_t \leq M_t$, such that $\mathbf{x}'_{m_t,0}$ does not lie in the row space of $\mathbf{X}_a$. □

Using this assumption, we have the following identifiability result (compare with theorem 7.2.4):

**Theorem 7.8.2** *Under assumptions 7.2.1 and 7.2.2,* (7.55) *has a unique (up to a scaling factor admitted by* $\Gamma$*) solution* $\mathbf{s}_1$ *in* $\Gamma$ *for the data symbols, if and only if condition 7.8.1 is satisfied.*

**Proof:** Under assumptions 7.2.1 and 7.2.2, we know that $\mathbf{s}_1$ is a solution of (7.55) in $\Gamma$ for the data symbols. We now prove that condition 7.8.1 is a necessary and sufficient condition for $\mathbf{s}_1$ to be a unique (up to a scaling factor admitted by $\Gamma$) solution in $\Gamma$ for the data symbols.

We first prove that condition 7.8.1 is a necessary condition. Suppose that there exists a vector $\mathbf{s}_1'$ in $\Gamma$, linearly independent of $\mathbf{s}_1$, such that $\mathbf{x}'_{m_t,0} = \mathbf{s}_1'\mathbf{C}_{m_t}$ lies in the row space of $\mathbf{X}_a$, for $a = A_1, A_1+1, \ldots, A_2$ and $m_t = 1, 2, \ldots, M_t$. Because the rows of $\mathbf{V}_a^{sH}$ form an orthonormal basis for the row space of $\mathbf{X}_a$, for $a = A_1, A_1+1, \ldots, A_2$ (this is due to assumptions 7.2.1 and 7.2.2), it is then clear that there exists a set of linear chip equalizers $\{\{\mathbf{f}'^s_{m_t,a}\}_{a=A_1}^{A_2}\}_{m_t=1}^{M_t}$, such that

$$\mathbf{f}'^s_{m_t,a}\mathbf{V}_a^{sH} - \mathbf{x}'_{m_t,0} = \mathbf{f}'^s_{m_t,a}\mathbf{V}_a^{sH} - \mathbf{s}_1'\mathbf{C}_{m_t} = \mathbf{0},$$
$$\text{for } a = A_1, A_1+1, \ldots, A_2 \text{ and } m_t = 1, 2, \ldots, M_t.$$

This means that $\mathbf{s}_1'$ is also a solution of (7.55) in $\Gamma$ for the data symbols.

We then prove that condition 7.8.1 is a sufficient condition. Suppose that there exists a vector $\mathbf{s}_1'$ in $\Gamma$, linearly independent of $\mathbf{s}_1$, such that $\mathbf{s}_1'$ is also a solution of (7.55) in $\Gamma$ for the data symbols. This means that there exists a set of linear chip equalizers $\{\{\mathbf{f}'^s_{m_t,a}\}_{a=A_1}^{A_2}\}_{m_t=1}^{M_t}$, such that

$$\mathbf{f}'^s_{m_t,a}\mathbf{V}_a^{sH} - \mathbf{s}_1'\mathbf{C}_{m_t} = \mathbf{f}'^s_{m_t,a}\mathbf{V}_a^{sH} - \mathbf{x}'_{m_t,0} = \mathbf{0},$$
$$\text{for } a = A_1, A_1+1, \ldots, A_2 \text{ and } m_t = 1, 2, \ldots, M_t.$$

Because the rows of $\mathbf{V}_a^{sH}$ form an orthonormal basis for the row space of $\mathbf{X}_a$, for $a = A_1, A_1+1, \ldots, A_2$ (this is due to assumptions 7.2.1 and 7.2.2), it is then clear that $\mathbf{x}'_{m_t,0} = \mathbf{s}_1'\mathbf{C}_{m_t}$ lies in the row space of $\mathbf{X}_a$, for $a = A_1, A_1+1, \ldots, A_2$ and $m_t = 1, 2, \ldots, M_t$. This concludes the proof. □

Under assumption 7.2.2, it is easy to show that condition 7.8.1 is equivalent

with

$$\begin{bmatrix} \mathcal{X} & & \\ & \ddots & \\ & & \mathcal{X} \\ \mathcal{C}_1 & \cdots & \mathcal{C}_{M_t} \end{bmatrix}$$

having a one-dimensional left null space or equivalently having rank $M_t Ar + K - 1$. It is clear that this can only be satisfied if

$$M_t A(KN - r) \geq K - 1. \tag{7.56}$$

**Remark 7.8.3** Assume that $L_1 \leq L_{M_t(j_t-1)+1}$, for $j_t = 2, 3, \ldots, J/M_t$. Further, assume that there exists a vector $\mathbf{s}_1'$ in $\Gamma$, linearly independent of $\mathbf{s}_1$, such that $\mathbf{x}'_{m_t,0} = \mathbf{s}_1' \mathbf{C}_{m_t}$ is linearly dependent of $\{\mathbf{x}_{j,\delta_j}\}_{j=1}^{J}$, for $m_t = 1, 2, \ldots, M_t$. Then, it is clear that, because the vectors $\{\mathbf{x}_{j,\delta_j}\}_{j=1}^{J}$ are all rows of every input matrix from the set $\{\mathbf{X}_a\}_{a=-Q}^{L_1}$, condition 7.8.1 will never be satisfied, irrespective of $A_1$ and $A_2$, with $-Q \leq A_1 \leq A_2 \leq L_1$. This means that if condition 7.8.1 is not satisfied for $A = 1$, it could very well not be satisfied for $A > 1$. That is why the design of $K$ and $N$ should better not be related to $A$. Instead of (7.56), we therefore take

$$K(M_t N - 1) \geq M_t r - 1 \tag{7.57}$$

as a design criterion for $K$ and $N$. □

**Remark 7.8.4** All the block processing methods proposed in sections 7.3 and 7.4 can easily be adapted to the situation considered in this section. These methods are then referred to as **TD (transmit diversity) methods**. Mark that constraint C3 is then written as

$$\| \begin{bmatrix} \hat{f}_1^s & \cdots & \hat{f}_{M_t}^s \end{bmatrix} \|^2 = 1 \quad \text{or} \quad \| \begin{bmatrix} f_1 & \cdots & f_{M_t} \end{bmatrix} \|^2 = 1.$$

□

## 7.9 Code Modulation

The smaller the spreading factor $N$, the larger the data symbol rate for a fixed bandwidth or the smaller the bandwidth for a fixed data symbol rate. So, it is interesting to investigate the identifiability for $N = 1$. This case is referred to as *code modulation*, since a data symbol sequence is then modulated (and not spread) with a code sequence. The idea of modulating a data symbol sequence

with a code sequence is not new. In [CL97] and [SG98], it is used to get rid of the identifiability conditions for SOS blind single-user channel estimation in a TDMA system with a single user per time channel. We use it, on the other hand, to solve a source separation problem. Moreover, the methods discussed in [CL97] and [SG98] are stochastic methods, while we focus on the development of deterministic methods. Note that in [XBM98], correlative coding is used to solve the source separation problem. Like code modulation, correlative coding does not spread the data symbol sequence. However, this method is rather complex and, like the methods of [CL97] and [SG98], it is a stochastic method.

Under assumption 7.2.2, condition 7.2.3 with $\Gamma = \Omega^{1\times K}$ can very well be satisfied for $N = 1$. Hence, the methods with a finite alphabet constraint on the data symbols can be applied for $N = 1$. However, these methods are rather complex. Under assumption 7.2.2, condition 7.2.3 with $\Gamma = \mathbb{C}^{1\times K}$ is, on the other hand, most likely not satisfied (certainly not satisfied if $A = 1$) for $N = 1$. This is already indicated by (7.17), but can also be explained in more detail by the following remark:

**Remark 7.9.1** First, consider $N = 1$. We then assume w.l.o.g. that $\delta_j = 0$, for $j = 1, 2, \ldots, J$. Further, assume that $L_1 \leq L_j$, with $j \neq 1$. Then, we know that $\mathbf{x}_{j,0} = \mathbf{s}_j\mathbf{C}_j$ is a row of every input matrix from the set $\{\mathbf{X}_a\}_{a=-Q}^{L_1}$, which means that if we take $\mathbf{s}_1' = \mathbf{s}_j\mathbf{C}_j\mathbf{C}_1^{-1}$, $\mathbf{s}_1'\mathbf{C}_1$ is also a row of every input matrix from the set $\{\mathbf{X}_a\}_{a=-Q}^{L_1}$ (because $\mathbf{s}_1'\mathbf{C}_1 = \mathbf{s}_j\mathbf{C}_j = \mathbf{x}_{j,0}$). Since $\mathbf{x}_{j,0}$ is independent from $\mathbf{x}_{1,0}$ (this is due to assumption 7.2.2), we further know that $\mathbf{s}_1'$ is independent from $\mathbf{s}_1$. Hence, condition 7.2.3 with $\Gamma = \mathbb{C}^{1\times K}$ is in this case not satisfied, irrespective of $A_1$ and $A_2$, with $-Q \leq A_1 \leq A_2 \leq L_1$. Only when $L_1 > L_j$, for $j = 2, 3, \ldots, J$, and we take $A_1 = -Q$ and $L_j < A_2 \leq L_1$, for $j = 2, 3, \ldots, J$, condition 7.2.3 with $\Gamma = \mathbb{C}^{1\times K}$ can be satisfied. □

*Hence, the methods with a polynomial constraint (e.g., constraint C1, C2 or C3) should not be applied for $N = 1$.*

Let us then focus on the situation discussed in section 7.7. Under assumption 7.2.2, condition 7.2.3 with $\Gamma = \mathbb{R}^{1\times K}$ can very well be satisfied for $N = 1$, if the code sequences are complex. This is indicated by (7.53). *Hence, the RC methods with a polynomial constraint can be applied for $N = 1$, if the code sequences are complex.* Under assumption 7.2.2, condition 7.2.3 with $\Gamma = \mathbb{R}^{1\times K}$ is, on the other hand, most likely not satisfied (certainly not satisfied if $A = 1$) for $N = 1$, if the code sequences are real. This is already indicated by (7.54), but can also be explained in more detail by a remark similar to remark 7.9.1. *Hence, the RC methods with a polynomial constraint should not be applied for $N = 1$, if the code sequences are real.*

Let us finally focus on the situation discussed in section 7.8 (remember that $M_t > 1$), Under assumption 7.2.2, condition 7.8.1 with $\Gamma = \mathbb{C}^{1\times K}$ can very well

be satisfied for $N = 1$. This is indicated by (7.57). *Hence, the TD methods with a polynomial constraint can be applied for* $N = 1$.

## 7.10 Simulation Results

In this section we will perform some simulations on five different DS-CDMA systems:

1. The first system has 4 receive antennas ($M = 4$) and 2 users ($J = 2$). The spreading factor is $N = 2$. The channel orders are given by $L_1 = 2$, for the first user, and $L_2 = 4$, for the second user. All the delay indices are zero ($\delta_1 = \delta_2 = 0$). The modulation format is QPSK ($s_j[k] = \pm 1/\sqrt{2} \pm 1/\sqrt{2}i$) and we use random complex constant modulus code sequences.
2. The second system has 4 receive antennas ($M = 4$) and 2 users ($J = 2$). The spreading factor is $N = 1$ (code modulation). The channel orders are given by $L_1 = 2$, for the first user, and $L_2 = 4$, for the second user. All the delay indices are zero ($\delta_1 = \delta_2 = 0$). The modulation format is BPSK ($s_j[k] = \pm 1$) and we use random complex constant modulus code sequences.
3. The third system has 8 receive antennas ($M = 8$), 2 users and 2 transmit antennas per user ($M_t = 2$, $J = 4$). The spreading factor is $N = 1$ (code modulation). The channel orders are given by $L_1 = L_2 = 2$, for the first user, and $L_3 = L_4 = 4$, for the second user. All the delay indices are zero ($\delta_1 = \delta_2 = \delta_3 = \delta_4 = 0$). The modulation format is QPSK ($s_j[k] = \pm 1/\sqrt{2} \pm 1/\sqrt{2}i$) and we use random complex constant modulus code sequences.
4. The fourth system has 6 receive antennas ($M = 6$) and 4 users ($J = 4$). The spreading factor is $N = 1$ (code modulation). All the channel orders are zero ($L_1 = L_2 = L_3 = L_4 = 0$), hence, there is no ICI. All the delay indices are zero ($\delta_1 = \delta_2 = \delta_3 = \delta_4 = 0$). The modulation format is BPSK ($s_j[k] = \pm 1$) and we use random complex constant modulus code sequences.
5. The fifth system has 12 receive antennas ($M = 12$), 4 users system and 2 transmit antennas per user ($M_t = 2$, $J = 8$). The spreading factor is $N = 1$ (code modulation). All the channel orders are zero ($L_1 = \cdots = L_8 = 0$), hence, there is no ICI. All the delay indices are zero ($\delta_1 = \cdots = \delta_8 = 0$). The modulation format is QPSK ($s_j[k] = \pm 1/\sqrt{2} \pm 1/\sqrt{2}i$) and we use random complex constant modulus code sequences.

For the case no transmit diversity is used (systems 1, 2 and 4), we assume that the data symbol sequences $\{s_j[k]\}_{j=1}^{J}$ are mutually independent and zero-mean

white. For the case transmit diversity is used (systems 3 and 5), we assume that the data symbol sequences $\{s_{M_t(j_t-1)+1}[k]\}_{j_t=1}^{J/M_t}$ are mutually independent and zero-mean white. We further assume that the additive noises $\{e^{(m)}[n]\}_{m=1}^{M}$ are mutually independent and zero-mean white Gaussian with variance $\sigma_e^2$. Finally, we assume that all users have the same received power. For the case no transmit diversity is used, the SNR at the input of the receiver is defined as

$$SNR = \frac{\sum_{j=1}^{J} \sum_{n=-\infty}^{+\infty} \|\mathbf{g}_j^{st}[n]\|^2}{JM\sigma_e^2}. \tag{7.58}$$

For the case transmit diversity is used, the SNR at the input of the receiver is defined as

$$SNR = \frac{\sum_{j=1}^{J} \sum_{n=-\infty}^{+\infty} \|\mathbf{g}_j^{st}[n]\|^2}{(J/M_t)M\sigma_e^2}.$$

If no ICI is present in the system (this is the case for systems 4 and 5), we always take $Q = 0$, since increasing $Q$ does not lead to a better source separation. Since all the channel orders are zero, we can then only apply $A_1 = A_2 = 0$ ($A = 1$) for every user. If ICI is present in the system (this is the case for systems 1, 2 and 3), we take an appropriate value of $Q$ and consider one of the following scenarios:

1. $A_1 = A_2 = 0$ is applied for every user. No channel order information is required.

2. $A_1 = -Q$ and $A_2 = 0$ is applied for every user. No channel order information is required.

3. For the case no transmit diversity is used, $A_1 = -Q + L_j$ and $A_2 = 0$ (if $Q \geq L_j$) or $A_1 = 0$ and $A_2 = L_j - Q$ (if $Q < L_j$) is applied for the $j$th user ($j = 1, 2, \ldots, J$). This corresponds to the situation where we only take those output matrices, for which the part related to $\mathbf{x}_{j,0}$ has the highest possible expected energy, if the channels are unknown. For the case transmit diversity is used, $A_1 = -Q + L_{M_t(j_t-1)+1}$ and $A_2 = 0$ (if $Q \geq L_{M_t(j_t-1)+1}$) or $A_1 = 0$ and $A_2 = L_{M_t(j_t-1)+1} - Q$ (if $Q < L_{M_t(j_t-1)+1}$) is applied for the $j_t$th user ($j_t = 1, 2, \ldots, J/M_t$). This corresponds to the situation where we only take those output matrices, for which the part related to $\{\mathbf{x}_{M_t(j_t-1)+m_t,0}\}_{m_t=1}^{M_t}$ has the highest possible expected energy, if the channels are unknown. Note that channel order information is required for this scenario.

4. For the case no transmit diversity is used, $A_1 = -Q$ and $A_2 = L_j$ is applied for the $j$th user ($j = 1, 2, \ldots, J$). For the case transmit diversity is used, $A_1 = -Q$ and $A_2 = L_{M_t(j_t-1)+1}$ is applied for the $j_t$th user ($j_t = 1, 2, \ldots, J/M_t$). Note again that channel order information is required for this scenario.

Although many other scenarios can be used, we restrict ourselves to these four scenarios.

In the subspace framework, we consider two values of $\hat{r}$: $\hat{r} = r$ (when $r$ is exactly estimated) and $\hat{r} = (Q+1)M$ (when $r$ is not estimated). We further consider a burst length of $K = 100$.

*Simulation 1:* In this simulation, system 1 is considered. First, we discuss the performance of the different block processing methods discussed in sections 7.3 and 7.4. We consider scenario 1 and take $Q = 4$. In **figure 7.5**, we compare the performance of different subspace block processing methods. Clearly, method BM1 with constraint C2 and method BM3 with constraint C3 outperform all other subspace block processing methods. From this figure it is also clear that the subspace block processing methods are fairly robust against system order overestimation. In **figure 7.6**, we then compare the performance of method BM1 with constraint C2 and method BM3 with constraint C3 with the performance of some non-subspace block processing methods. Note that in this figure the performance of method BM5 with constraint C1 is not plotted, since this performance is almost identical to the performance of method BM2 with constraint C1. From this figure we can conclude that no non-subspace block processing method has a better performance than method BM1 with constraint C2 and method BM3 with constraint C3. As expected (see remark 7.4.4), method BM4 with constraint C3 does not have such a good performance.

In **figure 7.7**, we then show the influence of $Q$ on the performance of method BM1 with constraint C2 and method BM3 with constraint C3. We consider scenario 1 and take $\hat{r} = r$. Clearly, as $Q$ increases the performance improves. However, at a certain point the performance will saturate (not shown).

Finally, we compare the performance of method BM1 with constraint C2 and method BM3 with constraint C3 for different scenarios. We take $Q = 4$ and $\hat{r} = r$. The results are shown in **figure 7.8**. First of all, we see that scenario 2 exhibits a large performance loss w.r.t. scenario 1. On the other hand, there is a performance improvement for scenario 3 w.r.t. scenario 1 and for scenario 4 w.r.t. scenario 3. Note that the performance measure we use is average BER per user. Hence, the BER for a specific user does not necessarily behaves the same way the average BER per user does, e.g., for the system under consideration, the best scenario for the first user is scenario 3, while the best scenario for the second user is scenario 4 (not shown). As a conclusion, we can state that, although channel order information is not really necessary, it can lead to an

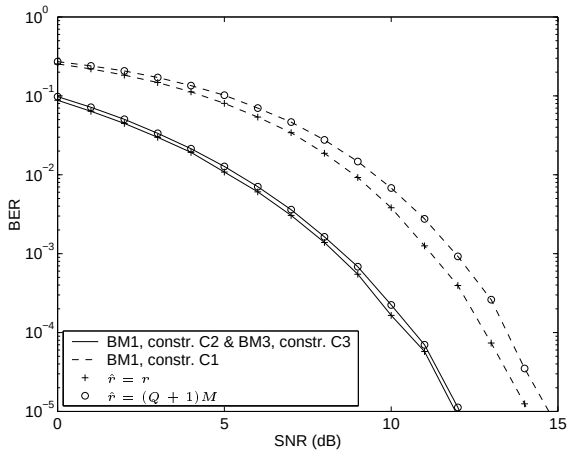



Figure 7.5: Average BER per user as a function of SNR for different subspace block processing methods (system 1, scenario 1, $Q = 4$, $K = 100$).

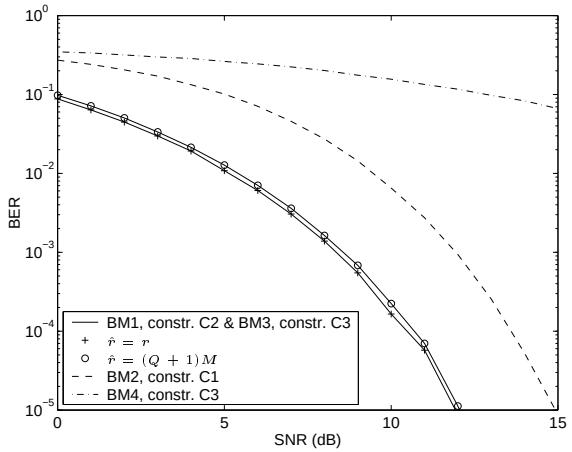



Figure 7.6: Average BER per user as a function of SNR for subspace and non-subspace block processing methods (system 1, scenario 1, $Q = 4$, $K = 100$).

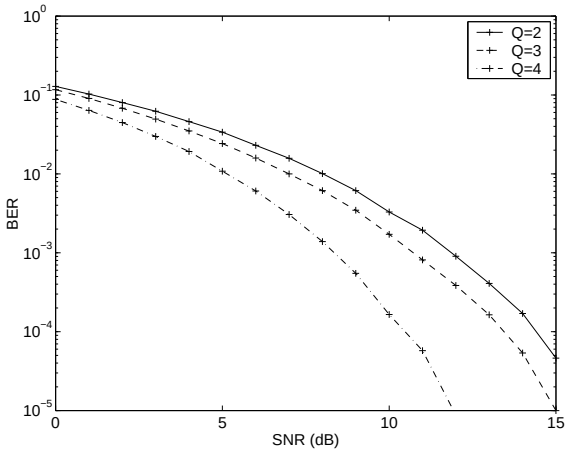



Figure 7.7: Average BER per user as a function of SNR for method BM1 with constraint C2 and method BM3 with constraint C3 for different values of $Q$ (system 1, scenario 1, $\hat{r} = r$, $K = 100$).

improved performance.

*Simulation 2:* In this simulation, we consider code modulation with ICI (systems 2 and 3). First, we consider system 2 and take $Q = 4$. In **figure 7.9**, we show the performance of RC method BM1 with constraint C2 and RC method BM3 with constraint C3 for two scenarios: scenario 2 and scenario 4. Next, we consider system 3 and take $Q = 4$. In **figure 7.10**, we show the performance of TD method BM1 with constraint C2 and TD method BM3 with constraint C3 for the same two scenarios. Like in simulation 1, we observe a robustness against system order overestimation. Also like in simulation 1, we see that although channel order information is not really necessary, it can lead to an improved performance.

*Simulation 3:* In this simulation, we consider code modulation without ICI (systems 4 and 5). As already mentioned, when no ICI is present in the system, we always take $Q = 0$ and we apply $A_1 = A_2 = 0$ ($A = 1$) for every user. First, we consider system 4. In **figure 7.11**, we compare the performance of RC method BM1 with constraint C2 and RC method BM3 with constraint C3 with the performance of the ILSP algorithm of [TVP96] and the RACMA algorithm of [vdV97], applied to a system that is similar to system 4, except that we now take $x_j[k] = s_j[k]$, for $j = 1, 2, 3, 4$ (hence, we use no coding). For the ILSP algorithm we show results for different numbers of random initializations (1, 2 and 3 random initializations). Next, we consider system 5. In **figure 7.12**, we compare the performance of TD method BM1 with constraint C2 and TD method BM3 with constraint C3 with the performance of the ILSP algorithm

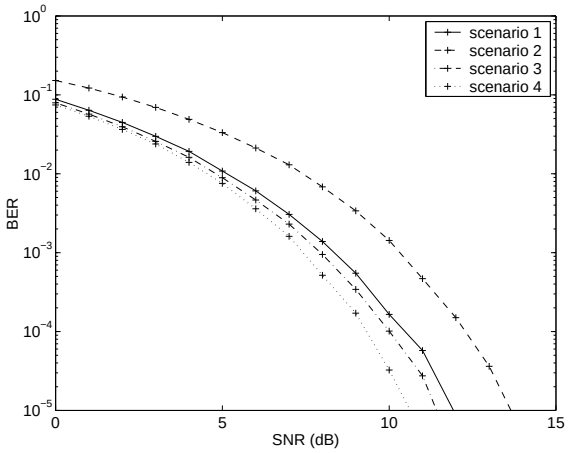



Figure 7.8: Average BER per user as a function of SNR for method BM1 with constraint C2 and method BM3 with constraint C3 for different scenarios (system 1, $Q = 4$, $\hat{r} = r$, $K = 100$).

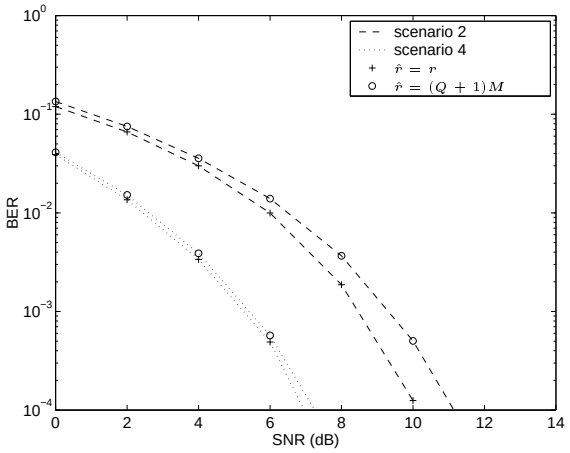



Figure 7.9: Average BER per user as a function of SNR for RC method BM1 with constraint C2 and RC method BM3 with constraint C3 for two scenarios (system 2, $Q = 4$, $K = 100$).

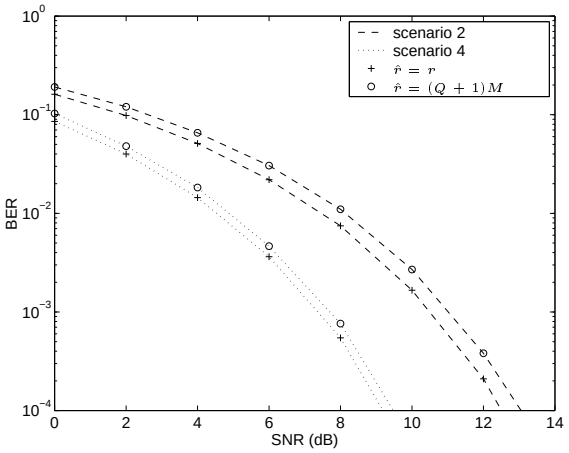



Figure 7.10: Average BER per user as a function of SNR for TD method BM1 with constraint C2 and TD method BM3 with constraint C3 for two scenarios (system 3, $Q = 4$, $K = 100$).

of [TVP96] and the RACMA algorithm of [vdV97], applied to a system that is similar to system 5, except that we now take $x_{2(j_t-1)+1}[k] = \text{sign}\{\Re\{s_{j_t}[k]\}\}$ and $x_{2j_t}[k] = \text{sign}\{\Im\{s_{j_t}[k]\}\}$, for $j_t = 1, 2, 3, 4$ (hence, we use no coding). For the ILSP algorithm we again show results for different numbers of random initializations (1, 4 and 7 random initializations).

Note that for ILSP and RACMA, we split the received sequence in its real and imaginary part, hence doubling the number of observations. We observe that the performance of the ILSP algorithm strongly depends on the number of random initializations. We also see that for a small number of random initializations and a high SNR, the ILSP algorithm may not find the global minimum. The good performance of the RACMA algorithm and the ILSP algorithm (for a large number of random initializations and a low SNR) can be explained by the fact that these two algorithms jointly detect all the data symbol sequences and that they exploit the finite alphabet property of the data symbols.

## 7.11 Conclusions

In this chapter, deterministic blind ST-RAKE receivers based on block processing have been developed. We have considered a subspace framework as well as a non-subspace framework. Direct blind symbol estimation methods

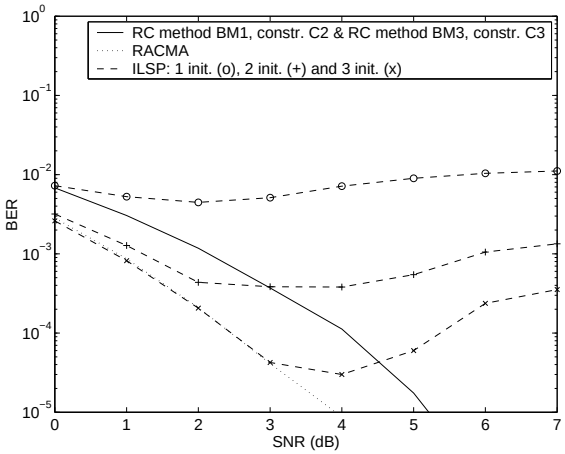



Figure 7.11: Average BER per user as a function of SNR for RC method BM1 with constraint C2 and RC method BM3 with constraint C3, ILSP and RACMA. (system 4, $K = 100$).

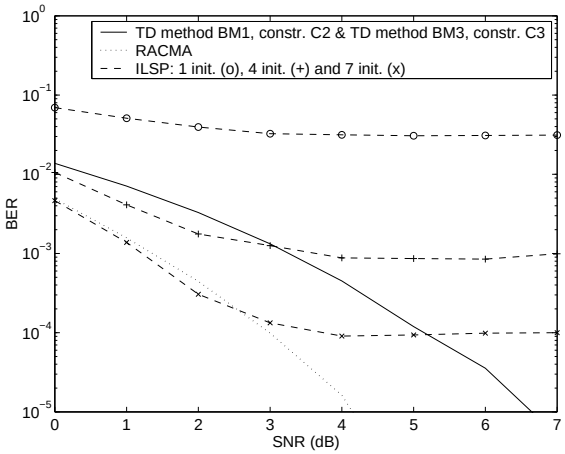



Figure 7.12: Average BER per user as a function of SNR for TD method BM1 with constraint C2 and TD method BM3 with constraint C3, ILSP and RACMA. (system 5, $K = 100$).

as well as direct blind equalizer estimation methods have been described, with eye for computational complexity. Furthermore, we have revealed some interesting relations between the proposed methods, on one side, and the SSI methods of [LX97, vdVTP97] and the MRE method of [GDM97], which are all developed for a TDMA system (no coding), on the other side.

The developed block processing methods are applicable for any modulation format. However, when the data symbols belong to a real constellation, we can adapt these methods to exploit the real character of the data symbols. The methods obtained in this way are referred to as RC methods.

We have also described how the developed block processing methods can be adapted when transmit diversity is used (multiple transmit antennas per user). We then spread the same data symbol sequence with different code sequences and transmit the resulting chip sequences through different antennas. The methods obtained in this way are referred to as TD methods.

Next, we have discussed the case for which the spreading factor $N$ is equal to 1 (code modulation). We have showed that *the methods with a polynomial constraint should not be applied for* $N = 1$. *The RC methods with a polynomial constraint, on the other hand, should also not be applied for* $N = 1$*, if the code sequences are real, but they can be applied for* $N = 1$*, if the code sequences are complex.* Finally, *the TD methods with a polynomial constraint can always be applied for* $N = 1$.

Simulation results have shown that the proposed methods render *robustness against system order overestimation.* Moreover, we have seen that, *although channel order information is not really necessary, it can lead to an improved performance.* Finally, we have illustrated that, when no ICI is present, *the proposed RC or TD methods based on code modulation constitute cheap and reasonably well performing alternatives to existing blind source separation methods that are not based on coding*, such as ILSP [TVP96] and RACMA [vdV97].

# Chapter 8

# Deterministic Blind Adaptive Processing

In this chapter, we develop deterministic blind ST-RAKE receivers based on adaptive processing. In section 8.1, the basic idea behind windowed adaptive processing is explained. Like in the previous chapter, we consider a subspace framework as well as a non-subspace framework. In section 8.2, we then discuss two windowed adaptive processing methods for direct blind symbol estimation (we do not consider direct blind equalizer estimation). These methods are further discussed in section 8.3. A computational complexity analysis is given in section 8.4. Next, in section 8.5, the basic idea behind exponentially weighted adaptive processing is explained. This kind of processing only makes sense in a non-subspace framework. In section 8.6, we then discuss one exponentially weighted adaptive processing method for direct blind symbol estimation (like before, we do not consider direct blind equalizer estimation). This method is further discussed in section 8.7. The computational complexity is examined in section 8.8. Finally, simulations results and conclusions are presented in sections 8.9 and 8.10, respectively.

## 8.1 Windowed Adaptive Processing

For the windowed adaptive processing approach, we consider the data model presented in section 7.1, with $n = kN + a$, where $k$ is some variable time index and $a$ again represents the processing delay, and $\beta = w$, where $w$ represents the window length we will apply ($w \ll K$). Let us then define

$$\mathbf{s}_{1,k} = \begin{bmatrix} s_1[k] & s_1[k+1] & \cdots & s_1[k+w-1] \end{bmatrix},$$

Using (2.1) and (7.4), it is clear that this vector is 'contained' in $\mathbf{x}_{1,kN}$:

$$\mathbf{x}_{1,kN} = \begin{bmatrix} s_1[k]\mathbf{c}_1[k] & \cdots & s_1[k+w-1]\mathbf{c}_1[k+w-1] \end{bmatrix} = \mathbf{s}_{1,k}\mathbf{C}_{1,k}, \qquad (8.1)$$

where $\mathbf{c}_1[k]$ is defined in (7.10) and $\mathbf{C}_{1,k}$ is the $w \times wN$ code matrix, defined as

$$\mathbf{C}_{1,k} = \begin{bmatrix} \mathbf{c}_1[k] & & \\ & \ddots & \\ & & \mathbf{c}_1[k+w-1] \end{bmatrix}. \qquad (8.2)$$

The vector $\mathbf{x}_{1,kN}$ is a row of every input matrix from the set $\{\mathbf{X}_{kN+a}\}_{a=-Q}^{L_1}$ and is therefore 'contained' in every output matrix from the set $\{\mathbf{Y}_{kN+a}\}_{a=-Q}^{L_1}$. Introducing the notation

$$\mathbf{s}_{1,k}^{(l)} = \begin{bmatrix} s_1[k] & s_1[k+1] & \cdots & s_1[k+l-1] \end{bmatrix}, \qquad (8.3)$$

where the superscript indicates the number of data symbols in the vector, the windowed adaptive processing problem addressed here is the following. At time index $k$ ($k = 0, 1, \ldots, K-w$), compute $\mathbf{s}_{1,0}^{(k+w)}$, which is defined using the notation of (8.3) and consists of the length-$w$ segments $\mathbf{s}_{1,0}, \mathbf{s}_{1,1}, \ldots, \mathbf{s}_{1,k}$, from $\{\mathbf{Y}_a\}_{a=A_1}^{A_2}, \{\mathbf{Y}_{N+a}\}_{a=A_1}^{A_2}, \ldots, \{\mathbf{Y}_{kN+a}\}_{a=A_1}^{A_2}$, with $-Q \leq A_1 \leq A_2 \leq L_1$ ($A = A_2 - A_1 + 1$), based only on the knowledge of the code sequence $c_1[n]$.

**Remark 8.1.1** We can also exploit the information present in $\{\mathbf{Y}_{kN+a}\}_{a=A_1}^{A_2}$, for $k = -w+1, -w+2, \ldots, -1$ and $k = K-w+1, K-w+2, \ldots, K-1$. However, since notation then becomes rather cumbersome, we will not use this information in the theoretical derivations. To generate the simulation results presented in section 8.9, on the other hand, we will use this information. □

**Remark 8.1.2** Note that in case of time varying channels, $w$ should not be chosen larger than the coherence time of the channels. □

To solve the above windowed adaptive processing problem, we again consider a subspace framework as well as a non-subspace framework.

### 8.1.1 Subspace Framework

Let us assume that additive noise is present in $\mathbf{Y}_{kN+a}$ ($k = 0, 1, \ldots, K-w$ and $a = A_1, A_1+1, \ldots, A_2$). Calculating the SVD of $\mathbf{Y}_{kN+a}$ then leads to

$$\mathbf{Y}_{kN+a} = \hat{\mathbf{U}}_{kN+a}\hat{\mathbf{\Sigma}}_{kN+a}\hat{\mathbf{V}}_{kN+a}^H,$$

where $\hat{\mathbf{\Sigma}}_{kN+a}$ is a diagonal matrix (diagonal elements in descending order) of the same size as $\mathbf{Y}_{kN+a}$ and $\hat{\mathbf{U}}_{kN+a}$ and $\hat{\mathbf{V}}_{kN+a}$ are square unitary matrices. For an estimate $\hat{r}$ of the system order $r$, let us then define the $wN \times \hat{r}$ matrix $\hat{\mathbf{V}}^s_{kN+a}$ as the collection of the first $\hat{r}$ columns of $\hat{\mathbf{V}}_{kN+a}$ (we define the collection of the last $wN - \hat{r}$ columns of $\hat{\mathbf{V}}_{kN+a}$ as $\hat{\mathbf{V}}^n_{kN+a}$). At time index $k$ ($k = 0, 1, \ldots, K-w$), we then solve the following minimization problem:

$$\boxed{\begin{array}{c}\min\limits_{\mathbf{s}_{1,0}^{(k+w)}, \hat{f}^s_{1,0:k}} \{\|\hat{f}^s_{1,0:k}\hat{\mathcal{V}}^{sH}_{0:k} - \mathbf{s}_{1,0}^{(k+w)}\mathcal{C}_{1,0:k}\|^2\}, \\ \text{s.t. a non-triviality constraint imposed on } \mathbf{s}_{1,0}^{(k+w)} \text{ and/or } \hat{f}^s_{1,0:k},\end{array}} \tag{8.4}$$

where $\hat{f}^s_{1,0:k}$ represents a collection of $k+1$ linear super chip equalizers of size $1 \times A\hat{r}$, $\hat{\mathcal{V}}^s_{0:k}$ is the $(k+1)AwN \times (k+1)A\hat{r}$ matrix, given by

$$\hat{\mathcal{V}}^s_{0:k} = \begin{bmatrix} \hat{\mathcal{V}}^s_0 & & & \\ & \hat{\mathcal{V}}^s_1 & & \\ & & \ddots & \\ & & & \hat{\mathcal{V}}^s_k \end{bmatrix},$$

and $\mathcal{C}_{1,0:k}$ is the $(k+w) \times (k+1)AwN$ matrix, given by

$$\mathcal{C}_{1,0:k} = \begin{bmatrix} \boxed{\mathcal{C}_{1,0}} & \mathbf{0} & & \mathbf{0} \\ \mathbf{0} & \boxed{\mathcal{C}_{1,1}} & \ddots & \vdots \\ \vdots & \vdots & & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & & \boxed{\mathcal{C}_{1,k}} \end{bmatrix},$$

with $\hat{\mathcal{V}}^s_k$ the $AwN \times A\hat{r}$ matrix, given by

$$\hat{\mathcal{V}}^s_k = \begin{bmatrix} \hat{\mathbf{V}}^s_{kN+A_1} & & \\ & \ddots & \\ & & \hat{\mathbf{V}}^s_{kN+A_2} \end{bmatrix},$$

and $\mathcal{C}_{1,k}$ the $AwN \times A\hat{r}$ matrix, given by

$$\mathcal{C}_{1,k} = \begin{bmatrix} \mathbf{C}_{1,k} & \cdots & \mathbf{C}_{1,k} \end{bmatrix}.$$

### 8.1.2 Non-Subspace Framework

Let us assume that additive noise is present in $\mathbf{Y}_{kN+a}$ ($k = 0, 1, \ldots, K-w$ and $a = A_1, A_1+1, \ldots, A_2$). At time index $k$ ($k = 0, 1, \ldots, K-w$), we then solve

the following minimization problem:

$$\boxed{\begin{array}{c}\min\limits_{\mathbf{s}_{1,0}^{(k+w)},f_{1,0:k}} \{\|f_{1,0:k}\mathcal{Y}_{0:k} - \mathbf{s}_{1,0}^{(k+w)}\mathcal{C}_{1,0:k}\|^2\}, \\ \text{s.t. a non-triviality constraint imposed on } \mathbf{s}_{1,0}^{(k+w)} \text{ and/or } f_{1,0:k},\end{array}} \tag{8.5}$$

where $f_{1,0:k}$ represents a collection of $k+1$ linear super chip equalizers of size $1 \times A(Q+1)M$ and $\mathcal{Y}_{0:k}$ is the $(k+1)A(Q+1)M \times (k+1)AwN$ matrix, given by

$$\mathcal{Y}_{0:k} = \begin{bmatrix} \mathcal{Y}_0 & & & \\ & \mathcal{Y}_1 & & \\ & & \ddots & \\ & & & \mathcal{Y}_k \end{bmatrix},$$

with $\mathcal{Y}_k$ the $A(Q+1)M \times AwN$ matrix, given by

$$\mathcal{Y}_k = \begin{bmatrix} \mathbf{Y}_{kN+A_1} & & \\ & \ddots & \\ & & \mathbf{Y}_{kN+A_2} \end{bmatrix}.$$

### 8.1.3 Constraint

In the following section, we discuss different windowed adaptive processing methods based on (8.4) and (8.5). We only focus on direct blind symbol estimation and make use of only one constraint:

C4 Linear constraint on the data symbols:

$$\mathbf{s}_{1,0}^{(k+w)}(1:k-l) = \mathbf{s}_{1,0}^{(k-l)} = \underline{\hat{\mathbf{s}}}_{1,0}^{(k-l)} \quad (\text{assume } k > l \text{ and } l > -w),$$

where $\underline{\hat{\mathbf{s}}}_{1,0}^{(k-l)}$ is defined as

$$\underline{\hat{\mathbf{s}}}_{1,0}^{(k-l)} = \left[\underline{\hat{s}}_1[0] \quad \underline{\hat{s}}_1[1] \quad \cdots \quad \underline{\hat{s}}_1[k-l-1]\right],$$

with $\underline{\hat{s}}_1[k-l]$ representing the hard estimate of $s_1[k-l]$ made at time index $k$.

Applying constraint C4 will result in an RLS algorithm with a fixed computational complexity per data symbol period.

**Remark 8.1.3** Note that constraint C4 imposes some kind of decision feedback. The parameter $l$ determines the degree of decision feedback. It is a design parameter to change the number of unknown data symbols or memory size $w+l$, once $w$ is fixed. □

## 8.2 Direct Blind Symbol Estimation

In this section, we discuss two windowed adaptive processing methods for direct blind symbol estimation.

### 8.2.1 Windowed Adaptive Method WAM1

This method is based on the elimination of $\hat{f}^s_{1,0:k}$ from (8.4). Hence, we do not impose a non-triviality constraint on $\hat{f}^s_{1,0:k}$. Solving (8.4) for $\hat{f}^s_{1,0:k}$ leads to the following relation:

$$\hat{f}^s_{1,k} = \mathbf{s}^{(k+w)}_{1,0} \mathcal{C}_{1,0:k} \hat{\mathcal{V}}^s_{0:k}.$$

Substituting the above result in (8.4), we obtain the following minimization problem:

$$\min_{\mathbf{s}^{(k+w)}_{1,0}} \{\|\mathbf{s}^{(k+w)}_{1,0} \mathcal{C}_{1,0:k} (\mathbf{I} - \hat{\mathcal{V}}^s_{0:k} \hat{\mathcal{V}}^{sH}_{0:k})\|^2\},$$
$$\text{s.t. a non-triviality constraint imposed on } \mathbf{s}^{(k+w)}_{1,0},$$

which is equivalent to

$$\boxed{\begin{array}{c} \min_{\mathbf{s}^{(k+w)}_{1,0}} \{\|\mathbf{s}^{(k+w)}_{1,0} \mathcal{C}_{1,0:k} \hat{\mathcal{V}}^n_{0:k}\|^2\}, \\ \text{s.t. a non-triviality constraint imposed on } \mathbf{s}^{(k+w)}_{1,0}, \end{array}} \tag{8.6}$$

where $\hat{\mathcal{V}}^n_{0:k}$ is the $(k+1)AwN \times (k+1)A(wN-\hat{r})$ matrix, given by

$$\hat{\mathcal{V}}^n_{0:k} = \begin{bmatrix} \hat{\mathcal{V}}^n_0 & & & \\ & \hat{\mathcal{V}}^n_1 & & \\ & & \ddots & \\ & & & \hat{\mathcal{V}}^n_k \end{bmatrix},$$

with $\hat{\mathcal{V}}^n_k$ the $AwN \times A(wN-\hat{r})$ matrix, given by

$$\hat{\mathcal{V}}^n_k = \begin{bmatrix} \hat{\mathbf{V}}^n_{kN+A_1} & & \\ & \ddots & \\ & & \hat{\mathbf{V}}^n_{kN+A_2} \end{bmatrix}.$$

We focus on constraint C4. Note that the size of the minimization problem (8.6) grows with $k$. This problem will be dealt with.

**Example 8.2.1 (constraint C4)** If we use constraint C4 as a non-triviality constraint, (8.6) becomes the following minimization problem at time index $k$:

$$\boxed{\min_{\mathbf{s}_{1,k-l}^{(w+l)}} \{\|\mathbf{s}_{1,k-l}^{(w+l)} \bar{\mathcal{C}}_{1,0:k} \hat{\mathcal{V}}_{0:k}^{n} - \mathbf{b}_{1,0:k} \hat{\mathcal{V}}_{0:k}^{n}\|^2\},} \tag{8.7}$$

where $\mathbf{s}_{1,k-l}^{(w+l)}$ is defined using the notation of (8.3), $\bar{\mathcal{C}}_{1,0:k}$ is the $(w+l) \times (k+1)wN$ matrix, given by

$$\bar{\mathcal{C}}_{1,0:k} = \mathcal{C}_{1,0:k}(k-l+1 : k+w, :),$$

and $\mathbf{b}_{1,0:k}$ is the $1 \times (k+1)AwN$ vector, defined as

$$\mathbf{b}_{1,0:k} = \underline{\hat{\mathbf{s}}}_{1,0}^{(k-l)} \mathcal{C}_{1,0:k}(1 : k-l, :). \tag{8.8}$$

This LS problem can be solved recursively by an RLS algorithm based on QR updating with an R-matrix of *fixed size* $(w+l) \times (w+l)$, as we will show next.

First, rewrite (8.7) as

$$\min_{\mathbf{s}_{1,k-l}^{(w+l)}} \{\|\hat{\mathcal{V}}_{0:k}^{nT} \mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^{T}\} \mathrm{fr}\{\mathbf{s}_{1,k-l}^{(w+l)T}\} - \hat{\mathcal{V}}_{0:k}^{nT} \mathbf{b}_{1,0:k}^{T}\|^2\}, \tag{8.9}$$

where $\mathrm{fc}\{\cdot\}$ represents a flipping of the columns and $\mathrm{fr}\{\cdot\}$ represents a flipping of the rows. The reason for this modified representation will be explained later on. To solve (8.9), we can make use of the QRD [GVL89]. The QRD of the matrix

$$\left[\begin{array}{c|c} \hat{\mathcal{V}}_{0:k}^{nT} \mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^{T}\} & \hat{\mathcal{V}}_{0:k}^{nT} \mathbf{b}_{1,0:k}^{T} \end{array}\right],$$

focusing on all columns but the last, is given by

$$\left[\begin{array}{c|c} \hat{\mathcal{V}}_{0:k}^{nT} \mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^{T}\} & \hat{\mathcal{V}}_{0:k}^{nT} \mathbf{b}_{1,0:k}^{T} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{Q}_k & \star \end{array}\right] \left[\begin{array}{c|c} \mathbf{R}_k & \mathbf{z}_k \\ \hline \mathbf{O} & \star \end{array}\right], \tag{8.10}$$

where $\mathbf{Q}_k \mathbf{R}_k$ is the QRD of $\hat{\mathcal{V}}_{0:k}^{nT} \mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^{T}\}$ and $\left[\begin{array}{c|c} \mathbf{Q}_k & \star \end{array}\right]$ is unitary. Note that $\mathbf{R}_k$ has size $(w+l) \times (w+l)$. We assume $\hat{\mathcal{V}}_{0:k}^{nT} \mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^{T}\}$ has full column rank. The solution of (8.9) then satisfies

$$\mathbf{R}_k \mathrm{fr}\{\mathbf{s}_{1,k-l}^{(w+l)T}\} = \mathbf{z}_k, \tag{8.11}$$

which can be solved through backsubstitution.

Assume now that $\mathbf{R}_k$ and $\mathbf{z}_k$ are known. The aim is then to find an efficient updating rule to update $\mathbf{R}_k$ and $\mathbf{z}_k$ into $\mathbf{R}_{k+1}$ and $\mathbf{z}_{k+1}$. For this, we first have to find $\underline{\hat{s}}_1[k-l]$ (to update $\underline{\hat{\mathbf{s}}}_{1,0}^{(k-l)}$ into $\underline{\hat{\mathbf{s}}}_{1,0}^{(k+1-l)}$). As mentioned, $\underline{\hat{s}}_1[k-l]$

represents the hard estimate of $s_1[k-l]$ made at time index $k$, i.e., made by solving (8.11) for $s_1[k-l]$ and mapping the solution on the closest (in Euclidean distance) element of $\Omega$. Since $s_1[k-l]$ is the last element of $\mathrm{fr}\{\mathbf{s}_{1,k-l}^{(w+l)T}\}$, only the first step of the backsubstitution scheme to solve (8.11) has to be executed in order to find $\underline{\hat{s}}_1[k-l]$ (parallel implementation possible). This is the reason why we used the representation of (8.9) instead of (8.7). So, $\underline{\hat{s}}_1[k-l]$ is obtained as

$$\underline{\hat{s}}_1[k-l] = \underset{\Omega}{\mathrm{proj}}\Big\{\frac{\mathbf{z}_k(w+l)}{\mathbf{R}_k(w+l,w+l)}\Big\}.$$

Once we have found $\underline{\hat{s}}_1[k-l]$ we are ready to update $\mathbf{R}_k$ and $\mathbf{z}_k$ in a second step.

First, we consider $\boxed{l \geq 0}$ $(w+l \geq w)$. From (8.10) we can then derive

$$\left[\begin{array}{c|c} \hat{\mathcal{V}}_{0:k+1}^{nT}\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k+1}^{T}\} & \hat{\mathcal{V}}_{0:k+1}^{nT}\mathbf{b}_{1,0:k+1}^{T} \end{array}\right]$$

$$= \left[\begin{array}{c|c|c} \mathbf{0} & \hat{\mathcal{V}}_{0:k}^{nT}\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^{T}(:,2:l+w)\} & \mathbf{b}_{1,0:k}^{T} - \hat{\mathcal{V}}_{0:k}^{nT}\bar{\mathcal{C}}_{1,0:k}^{T}(:,1)\underline{\hat{s}}_1[k-l] \\ \hline \hat{\mathcal{V}}_{k+1}^{nT}\mathrm{fc}\{\mathcal{C}_{1,k+1}^{T}\} & \mathbf{O} & \mathbf{0} \end{array}\right]$$

$$= \left[\begin{array}{c|c|c} \mathbf{Q}_k & \star & \mathbf{O} \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{I} \end{array}\right] \underbrace{\left[\begin{array}{c|c|c} \mathbf{0} & \mathbf{R}_k(:,1:l+w-1) & \mathbf{z}_k - \mathbf{R}_k(:,l+w)\underline{\hat{s}}_1[k-l] \\ \hline \multicolumn{2}{c|}{\mathbf{O}} & \star \\ \hline \hat{\mathcal{V}}_{k+1}^{nT}\mathrm{fc}\{\mathcal{C}_{1,k+1}^{T}\} & \mathbf{O} & \mathbf{0} \end{array}\right]}_{\tilde{\mathbf{R}}}$$

$$= \left[\begin{array}{c|c|c} \mathbf{Q}_k & \star & \mathbf{O} \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{I} \end{array}\right] \tilde{\mathbf{P}} \left[\begin{array}{c|c} \mathbf{R} & \mathbf{z} \\ \hline \mathbf{O} & \star \\ \hline \mathbf{O} & \star \end{array}\right] = \left[\begin{array}{c||c|c} \mathbf{Q} & \star & \star \end{array}\right] \left[\begin{array}{c|c} \mathbf{R} & \mathbf{z} \\ \hline\hline \mathbf{O} & \star \\ \hline \mathbf{O} & \star \end{array}\right], \qquad (8.12)$$

where $\tilde{\mathbf{P}}^H$ is a unitary transformation matrix that triangularizes $\tilde{\mathbf{R}}$, except for the last column. Note that this triangularization can be performed by transforming only the first and third blockrow of $\tilde{\mathbf{R}}$. If we now compare (8.12) with

$$\left[\begin{array}{c|c} \hat{\mathcal{V}}_{0:k+1}^{nT}\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k+1}^{T}\} & \hat{\mathcal{V}}_{0:k+1}^{nT}\mathbf{b}_{1,0:k+1}^{T} \end{array}\right] = \left[\begin{array}{c||c} \mathbf{Q}_{k+1} & \star \end{array}\right] \left[\begin{array}{c|c} \mathbf{R}_{k+1} & \mathbf{z}_{k+1} \\ \hline\hline \mathbf{O} & \star \end{array}\right],$$

we see that the matrices $\mathbf{R}$ and $\mathbf{z}$ in (8.12) are the matrices $\mathbf{R}_{k+1}$ and $\mathbf{z}_{k+1}$ sought for. Keeping in mind that $\tilde{\mathbf{P}}^H$ only transforms the first and third blockrow of $\tilde{\mathbf{R}}$, the updating formula for $\mathbf{R}_k$ and $\mathbf{z}_k$ is given by

$$\left[\begin{array}{c|c} \mathbf{R}_{k+1} & \mathbf{z}_{k+1} \\ \hline \mathbf{O} & \star \end{array}\right] \leftarrow$$

$$\mathbf{P}^H \left[\begin{array}{c|c|c} \mathbf{0} & \mathbf{R}_k(:,1:w+l-1) & \mathbf{z}_k - \mathbf{R}_k(:,w+l)\underline{\hat{s}}_1[k-l] \\ \hline \hat{\mathcal{V}}_{k+1}^{nT}\mathrm{fc}\{\mathcal{C}_{1,k+1}^T\} & \mathbf{O} & \mathbf{0} \end{array}\right],$$

where $\mathbf{P}$ is obtained by removing the part of $\tilde{\mathbf{P}}$ corresponding to the second blockrow of $\tilde{\mathbf{R}}$.

When we consider $\boxed{-w < l < 0}$ $(0 < w+l < w)$, the updating formula for $\mathbf{R}_k$ and $\mathbf{z}_k$ is derived in a similar fashion and is given by

$$\left[\begin{array}{c|c} \mathbf{R}_{k+1} & \mathbf{z}_{k+1} \\ \hline \mathbf{O} & \star \end{array}\right] \leftarrow$$

$$\mathbf{P}^H \left[\begin{array}{c|c|c} \mathbf{0} & \mathbf{R}_k(:,1:w+l-1) & \mathbf{z}_k - \mathbf{R}_k(:,w+l)\underline{\hat{s}}_1[k-l] \\ \hline \hat{\mathcal{V}}_{k+1}^{nT}\mathrm{fc}\{\mathcal{C}_{1,k+1}^T(:,-l+1:w)\} & & -\hat{\mathcal{V}}_{k+1}^{nT}\mathcal{C}_{1,k+1}^T(:,1:-l)\underline{\hat{\mathbf{s}}}_{1,k+1}^{(-l)T} \end{array}\right].$$

Applying Householder QR updating [GVL89], the computational complexity per data symbol period is

$$8A(wN-\hat{r})(w+l)^2 \text{ flops.} \tag{8.13}$$

It is clear that, while the size of the minimization problem (8.6) grows with $k$, the computational complexity per data symbol period is independent of $k$. For $l \geq 0$, method WAM1 with constraint C4 is summarized in **table 8.1** (see section 8.3.1 for initialization and termination details). The modifications for $-w < l < 0$ are straightforward. □

### 8.2.2 Windowed Adaptive Method WAM2

We consider (8.5) as a minimization problem in $\mathbf{s}_{1,0}^{(k+w)}$. We focus on constraint C4. Note that the size of the minimization problem (8.5) grows with $k$. This problem will be dealt with.

**Example 8.2.2 (constraint C4)** If we use constraint C4 as a non-triviality constraint, (8.5) becomes the following minimization problem at time index $k$:

$$\boxed{\min_{\mathbf{s}_{1,k-l}^{(w+l)}, f_{1,0:k}} \{\| \left[\begin{array}{c|c} f_{1,0:k} & \mathbf{s}_{1,k-l}^{(w+l)} \end{array}\right] \left[\begin{array}{c} \mathcal{Y}_{0:k} \\ \hline -\vec{\mathcal{C}}_{1,0:k} \end{array}\right] - \mathbf{b}_{1,0:k}\|^2\}.} \tag{8.14}$$

This LS problem can be solved recursively by an RLS algorithm based on QR updating with an R-matrix of *fixed size* $(A(Q+1)M+w+l)\times(A(Q+1)M+w+l)$, as we will show next.

1. initialization:

$$\left[\begin{array}{c|c} \mathbf{R} & \mathbf{z} \\ \hline \mathbf{O} & \star \end{array}\right] \leftarrow \mathbf{P}^H \left[\begin{array}{c|c} \hat{\mathcal{V}}_0^{nT}\mathrm{fc}\{\mathcal{C}_{1,0}^T(:,2:w)\} & \hat{\mathcal{V}}_0^{nT}\mathcal{C}_{1,0}^T(:,1)\underline{\hat{s}}_1[0] \end{array}\right]$$

for $k = 0, \ldots, l$

- update $\mathbf{R}$ and $\mathbf{z}$

$$\left[\begin{array}{c|c} \mathbf{R} & \mathbf{z} \\ \hline \mathbf{O} & \star \end{array}\right] \leftarrow \mathbf{P}^H \left[\begin{array}{c|c|c} \mathbf{0} & \mathbf{R} & \mathbf{z} \\ \hline 0 & \mathbf{0} & 0 \\ \hline -\mathrm{fc}\{\mathcal{C}_{1,k+1}^T\} & \mathbf{O} & \mathbf{0} \end{array}\right]$$

2. propagation:
for $k = l+1, \ldots, K-w-1$

- make decision on data symbol

$$\underline{\hat{s}}_1[k-l] = \underset{\Omega}{\mathrm{proj}}\left\{\frac{\mathbf{z}(w+l)}{\mathbf{R}(w+l,w+l)}\right\}$$

- update $\mathbf{R}$ and $\mathbf{z}$

$$\left[\begin{array}{c|c} \mathbf{R} & \mathbf{z} \\ \hline \mathbf{O} & \star \end{array}\right] \leftarrow$$

$$\mathbf{P}^H \left[\begin{array}{c|c|c} \mathbf{0} & \mathbf{R}(:,1:w+l-1) & \mathbf{z} - \mathbf{R}(:,w+l)\underline{\hat{s}}_1[k-l] \\ \hline \hat{\mathcal{V}}_{k+1}^{nT}\mathrm{fc}\{\mathcal{C}_{1,k+1}^T\} & \mathbf{O} & \mathbf{0} \end{array}\right]$$

3. termination:
for $k = K-w, \ldots, K+l-2$ (we define $\zeta = k - K + w$)

- make decision on data symbol

$$\underline{\hat{s}}_1[k-l] = \underset{\Omega}{\mathrm{proj}}\left\{\frac{\mathbf{z}(w+l-\zeta)}{\mathbf{R}(w+l-\zeta,w+l-\zeta)}\right\}$$

- update $\mathbf{R}$ and $\mathbf{z}$

$$\left[\begin{array}{c|c} \mathbf{R} & \mathbf{z} \\ \hline \mathbf{0} & \star \end{array}\right] \leftarrow \left[\begin{array}{c|c} \mathbf{R}(:,1:w+l-\zeta-1) & \mathbf{z} - \mathbf{R}(:,w+l-\zeta)\underline{\hat{s}}_1[k-l] \end{array}\right]$$

make decision on data symbol

$$\underline{\hat{s}}_1[K-1] = \underset{\Omega}{\mathrm{proj}}\left\{\frac{\mathbf{z}(1)}{\mathbf{R}(1,1)}\right\}$$

Table 8.1: Method WAM1 with constraint C4 ($l \geq 0$).

First, rewrite (8.14) as

$$\min_{\mathbf{s}_{1,k-l}^{(w+l)},f_{1,0:k}} \{\| \left[ \begin{array}{c|c} \mathcal{Y}_{0:k}^T & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^T\} \end{array} \right] \left[ \begin{array}{c} f_{1,0:k}^T \\ \hline \mathrm{fr}\{\mathbf{s}_{1,k-l}^{(w+l)T}\} \end{array} \right] - \mathbf{b}_{1,0:k}^T \|^2\}, \qquad (8.15)$$

where $\mathrm{fc}\{\cdot\}$ represents a flipping of the columns and $\mathrm{fr}\{\cdot\}$ represents a flipping of the rows. The reason for this modified representation will be explained later on. To solve (8.15), we can make use of the QRD [GVL89]. The QRD of the matrix

$$\left[ \begin{array}{c|c|c} \mathcal{Y}_{0:k}^T & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^T\} & \mathbf{b}_{1,0:k}^T \end{array} \right],$$

focusing on all columns but the last, is given by

$$\begin{aligned} \left[ \begin{array}{c|c||c} \mathcal{Y}_{0:k}^T & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^T\} & \mathbf{b}_{1,0:k}^T \end{array} \right] &= \left[ \begin{array}{c|c||c} \mathbf{Q}_k^{(1)} & \mathbf{Q}_k^{(2)} & \star \end{array} \right] \left[ \begin{array}{c|c||c} \mathbf{R}_k^{(1,1)} & \mathbf{R}_k^{(1,2)} & \mathbf{z}_k^{(1)} \\ \hline \mathbf{O} & \mathbf{R}_k^{(2,2)} & \mathbf{z}_k^{(2)} \\ \hline \mathbf{O} & \mathbf{O} & \star \end{array} \right] \\ &= \left[ \begin{array}{c||c} \mathbf{Q}_k & \star \end{array} \right] \left[ \begin{array}{c||c} \mathbf{R}_k & \mathbf{z}_k \\ \hline \mathbf{O} & \star \end{array} \right], \end{aligned} \qquad (8.16)$$

where $\mathbf{Q}_k\mathbf{R}_k$ is the QRD of $\left[ \begin{array}{c|c} \mathcal{Y}_{0:k}^T & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^T\} \end{array} \right]$ and $\left[ \begin{array}{c|c} \mathbf{Q}_k & \star \end{array} \right]$ is unitary. Note that $\mathbf{R}_k^{(1,1)}$ has size $(k+1)A(Q+1)M \times (k+1)A(Q+1)M$ and $\mathbf{R}_k^{(2,2)}$ has size $(w+l) \times (w+l)$. We assume $\left[ \begin{array}{c|c} \mathcal{Y}_{0:k}^T & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^T\} \end{array} \right]$ has full column rank. The solution for $\mathbf{s}_{1,k-l}^{(w+l)}$ of (8.15) then satisfies

$$\mathbf{R}_k^{(2,2)} \mathrm{fr}\{\mathbf{s}_{1,k-l}^{(w+l)T}\} = \mathbf{z}_k^{(2)}, \qquad (8.17)$$

which can be solved through backsubstitution.

Assume now that $\mathbf{R}_k^{(2,2)}$ and $\mathbf{z}_k^{(2)}$ are known. The aim is then to find an efficient updating rule to update $\mathbf{R}_k^{(2,2)}$ and $\mathbf{z}_k^{(2)}$ into $\mathbf{R}_{k+1}^{(2,2)}$ and $\mathbf{z}_{k+1}^{(2)}$. Note that this updating can be performed without the knowledge of $\mathbf{R}_k^{(1,1)}$, $\mathbf{R}_k^{(1,2)}$ and $\mathbf{z}_k^{(1)}$. For this, we first have to find $\underline{\hat{s}}_1[k-l]$ (to update $\hat{\mathbf{s}}_{1,0}^{(k-l)}$ into $\hat{\mathbf{s}}_{1,0}^{(k+1-l)}$). As mentioned, $\underline{\hat{s}}_1[k-l]$ represents the hard estimate of $s_1[k-l]$ made at time index $k$, i.e., made by solving (8.17) for $s_1[k-l]$ and mapping the solution on the closest (in Euclidean distance) element of $\Omega$. Since $s_1[k-l]$ is the last element of $\mathrm{fr}\{\mathbf{s}_{1,k-l}^{(w+l)T}\}$, only the first step of the backsubstitution scheme to solve (8.17) has to be executed in order to find $\underline{\hat{s}}_1[k-l]$ (parallel implementation possible). This is the reason why we used the representation of (8.15) instead of (8.14). So, $\underline{\hat{s}}_1[k-l]$ is obtained as

$$\underline{\hat{s}}_1[k-l] = \underset{\Omega}{\mathrm{proj}}\Big\{ \frac{\mathbf{z}_k^{(2)}(w+l)}{\mathbf{R}_k^{(2,2)}(w+l,w+l)} \Big\}.$$

Once we have found $\hat{\underline{s}}_1[k-l]$ we are ready to update $\mathbf{R}_k^{(2,2)}$ and $\mathbf{z}_k^{(2)}$ in a second step.

First, we consider $\boxed{l \geq 0}$ $(w + l \geq w)$. From (8.16) we can then derive ($d = A(Q+1)M$)

$$\left[\begin{array}{c|c||c|c|c} \mathcal{Y}_{0:k+1}^T & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k+1}^T\} & \mathbf{b}_{1,0:k+1}^T \end{array}\right]$$

$$= \left[\begin{array}{c|c||c|c|c} \mathcal{Y}_{0:k}^T & \mathbf{O} & \mathbf{0} & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^T(:, 2:l+w)\} & \mathbf{b}_{1,0:k}^T + \bar{\mathcal{C}}_{1,0:k}^T(:,1)\hat{\underline{s}}_1[k-l] \\ \hline \mathbf{O} & \mathcal{Y}_{k+1}^T & -\mathrm{fc}\{\mathcal{C}_{1,k+1}^T\} & \mathbf{O} & \mathbf{0} \end{array}\right]$$

$$= \left[\begin{array}{c|c|c|c|c} \mathbf{Q}_k^{(1)} & \star & \mathbf{Q}_k^{(2)} & \star & \mathbf{O} \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{I} \end{array}\right] \cdot$$

$$\underbrace{\left[\begin{array}{c|c||c|c|c} \mathbf{R}_k^{(1,1)} & \mathbf{O} & \mathbf{0} & \mathbf{R}_k^{(1,2)}(:, 1:l+w-1) & \mathbf{z}_k^{(1)} - \mathbf{R}_k^{(1,2)}(:, l+w)\hat{\underline{s}}_1[k-l] \\ \hline \mathbf{O} & \mathbf{O}_d & \multicolumn{2}{c|}{\mathbf{O}} & \mathbf{0} \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{0} & \mathbf{R}_k^{(2,2)}(:, 1:l+w-1) & \mathbf{z}_k^{(2)} - \mathbf{R}_k^{(2,2)}(:, l+w)\hat{\underline{s}}_1[k-l] \\ \hline \mathbf{O} & \mathbf{O} & \multicolumn{2}{c|}{\mathbf{O}} & \star \\ \hline \mathbf{O} & \mathcal{Y}_{k+1}^T & -\mathrm{fc}\{\mathcal{C}_{1,k+1}^T\} & \mathbf{O} & \mathbf{0} \end{array}\right]}_{\tilde{\mathbf{R}}}$$

$$= \left[\begin{array}{c|c|c|c|c} \mathbf{Q}_k^{(1)} & \star & \mathbf{Q}_k^{(2)} & \star & \mathbf{O} \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{I} \end{array}\right] \tilde{\mathbf{P}} \left[\begin{array}{c|c||c|c} \multicolumn{4}{c}{\tilde{\mathbf{R}}(1:(k+1)d, :)} \\ \hline \mathbf{O} & \diagdown^{\star} & \star & \star \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{R}^{(2,2)} & \mathbf{z}^{(2)} \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{O} & \star \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{O} & \star \end{array}\right]$$

$$= \left[\begin{array}{c|c|c|c} \mathbf{Q}^{(1)} & \mathbf{Q}^{(2)} & \star & \star \end{array}\right] \left[\begin{array}{c|c||c|c} \multicolumn{4}{c}{\tilde{\mathbf{R}}(1:(k+1)d, :)} \\ \hline \mathbf{O} & \diagdown^{\star} & \star & \star \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{R}^{(2,2)} & \mathbf{z}^{(2)} \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{O} & \star \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{O} & \star \end{array}\right], \qquad (8.18)$$

where $\tilde{\mathbf{P}}^H$ is a unitary transformation matrix that triangularizes $\tilde{\mathbf{R}}$, except for the last column. Note that this triangularization can be performed by transforming only the second, third and fifth blockrow of $\tilde{\mathbf{R}}$. If we now compare (8.18) with

$$\left[\begin{array}{c|c||c|c} \mathcal{Y}_{0:k+1}^T & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k+1}^T\} & \mathbf{b}_{1,0:k+1}^T \end{array}\right]$$

$$= \left[\begin{array}{c|c|c} \mathbf{Q}_{k+1}^{(1)} & \mathbf{Q}_{k+1}^{(2)} & \star \end{array}\right] \left[\begin{array}{c|c|c} \mathbf{R}_{k+1}^{(1,1)} & \mathbf{R}_{k+1}^{(1,2)} & \mathbf{z}_{k+1}^{(1)} \\ \hline \mathbf{O} & \mathbf{R}_{k+1}^{(2,2)} & \mathbf{z}_{k+1}^{(2)} \\ \hline \mathbf{O} & \mathbf{O} & \star \end{array}\right],$$

we see that the matrices $\mathbf{R}^{(2,2)}$ and $\mathbf{z}^{(2)}$ in (8.18) are the matrices $\mathbf{R}_{k+1}^{(2,2)}$ and $\mathbf{z}_{k+1}^{(2)}$ sought for. Keeping in mind that $\tilde{\mathbf{P}}^H$ only transforms the second, third and fifth blockrow of $\tilde{\mathbf{R}}$, the updating formula for $\mathbf{R}_k^{(2,2)}$ and $\mathbf{z}_k^{(2)}$ is given by $(d = A(Q+1)M)$

$$\left[\begin{array}{c|c|c} \diagdown^{\star} & \star & \star \\ \hline \mathbf{O} & \mathbf{R}_{k+1}^{(2,2)} & \mathbf{z}_{k+1}^{(2)} \\ \hline \mathbf{O} & \mathbf{O} & \star \end{array}\right] \leftarrow$$

$$\mathbf{P}^H \left[\begin{array}{c|c|c} \mathbf{O}_d & \mathbf{O} & \mathbf{0} \\ \hline \mathbf{O} & \mathbf{0} \;\big|\; \mathbf{R}_k^{(2,2)}(:, 1:w+l-1) & \mathbf{z}_k^{(2)} - \mathbf{R}_k^{(2,2)}(:, w+l)\underline{\hat{s}}_1[k-l] \\ \hline \mathcal{Y}_{k+1}^T & -\mathrm{fc}\{\mathcal{C}_{1,k+1}^T\} & \mathbf{O} \quad\; \mathbf{0} \end{array}\right],$$

where $\mathbf{P}$ is obtained by removing the part of $\tilde{\mathbf{P}}$ corresponding to the first and fourth blockrow of $\tilde{\mathbf{R}}$.

When we consider $\boxed{-w < l < 0}$ $(0 < w + l < w)$, the updating formula for $\mathbf{R}_k^{(2,2)}$ and $\mathbf{z}_k^{(2)}$ is derived in a similar fashion and is given by $(d = A(Q+1)M)$

$$\left[\begin{array}{c|c|c} \diagdown^{\star} & \star & \star \\ \hline \mathbf{O} & \mathbf{R}_{k+1}^{(2,2)} & \mathbf{z}_{k+1}^{(2)} \\ \hline \mathbf{O} & \mathbf{O} & \star \end{array}\right] \leftarrow$$

$$\mathbf{P}^H \left[\begin{array}{c|c|c} \mathbf{O}_d & \mathbf{O} & \mathbf{0} \\ \hline \mathbf{O} & \mathbf{0} \;\big|\; \mathbf{R}_k^{(2,2)}(:, 1:w+l-1) & \mathbf{z}_k^{(2)} - \mathbf{R}_k^{(2,2)}(:, w+l)\underline{\hat{s}}_1[k-l] \\ \hline \mathcal{Y}_{k+1}^T & -\mathrm{fc}\{\mathcal{C}_{1,k+1}^T(:, -l+1:w)\} & \mathcal{C}_{1,k+1}^T(:, 1:-l)\underline{\hat{\mathbf{s}}}_{1,k+1}^{(-l)T} \end{array}\right].$$

Applying Householder QR updating [GVL89], the computational complexity per data symbol period is

$$8AwN(A(Q+1)M + w + l)^2 \text{ flops.} \tag{8.19}$$

It is clear that, while the size of the minimization problem (8.5) grows with $k$, the computational complexity per data symbol period is independent of $k$. For $l \geq 0$, method WAM2 with constraint C4 is summarized in **table 8.2** (see section 8.3.1 for initialization and termination details). The modifications for $-w < l < 0$ are straightforward. □

1. initialization:

$$\left[\begin{array}{c|c|c} \diagdown^{\star} & \star & \star \\ \hline \mathbf{O} & \mathbf{R} & \mathbf{z} \\ \hline \mathbf{O} & \mathbf{O} & \star \end{array}\right] \leftarrow \mathbf{P}^H \left[\begin{array}{c|c|c} \mathbf{O}_d & \mathbf{0} & \mathbf{0} \\ \hline \mathcal{Y}_0^T & -\mathrm{fc}\{\mathcal{C}_{1,0}^T(:,2:w)\} & \mathcal{C}_{1,0}^T(:,1)\underline{\hat{s}}_1[0] \end{array}\right]$$

for $k = 0, \ldots, l$

- update $\mathbf{R}$ and $\mathbf{z}$

$$\left[\begin{array}{c|c|c} \diagdown^{\star} & \star & \star \\ \hline \mathbf{O} & \mathbf{R} & \mathbf{z} \\ \hline \mathbf{O} & \mathbf{O} & \star \end{array}\right] \leftarrow \mathbf{P}^H \left[\begin{array}{c|cc|c} \mathbf{O}_d & \multicolumn{2}{c|}{\mathbf{O}} & \mathbf{0} \\ \hline \mathbf{O} & \mathbf{0} & \mathbf{R} & \mathbf{z} \\ \hline \mathbf{0} & 0 & \mathbf{0} & 0 \\ \hline \mathcal{Y}_{k+1}^T & \multicolumn{2}{c|}{-\mathrm{fc}\{\mathcal{C}_{1,k+1}^T\}} & \mathbf{0} \end{array}\right]$$

2. propagation:
for $k = l+1, \ldots, K-w-1$

- make decision on data symbol

$$\underline{\hat{s}}_1[k-l] = \operatorname*{proj}_{\Omega}\left\{\frac{\boldsymbol{z}(w+l)}{\mathbf{R}(w+l,w+l)}\right\}$$

- update $\mathbf{R}$ and $\mathbf{z}$

$$\left[\begin{array}{c|c|c} \diagdown^{\star} & \star & \star \\ \hline \mathbf{O} & \mathbf{R} & \mathbf{z} \\ \hline \mathbf{O} & \mathbf{O} & \star \end{array}\right] \leftarrow$$

$$\mathbf{P}^H \left[\begin{array}{c|cc|c} \mathbf{O}_d & \multicolumn{2}{c|}{\mathbf{O}} & \mathbf{0} \\ \hline \mathbf{O} & \mathbf{0} & \mathbf{R}(:,1:w+l-1) & \mathbf{z}-\mathbf{R}(:,w+l)\underline{\hat{s}}_1[k-l] \\ \hline \mathcal{Y}_{k+1}^T & -\mathrm{fc}\{\mathcal{C}_{1,k+1}^T\} & \mathbf{O} & \mathbf{0} \end{array}\right]$$

3. termination:
for $k = K-w, \ldots, K+l-2$ (we define $\zeta = k-K+w$)

- make decision on data symbol

$$\underline{\hat{s}}_1[k-l] = \operatorname*{proj}_{\Omega}\left\{\frac{\mathbf{z}(w+l-\zeta)}{\mathbf{R}(w+l-\zeta,w+l-\zeta)}\right\}$$

- update $\mathbf{R}$ and $\mathbf{z}$

$$\left[\begin{array}{c|c} \mathbf{R} & \mathbf{z} \\ \hline \mathbf{0} & \star \end{array}\right] \leftarrow \left[\begin{array}{c|c} \mathbf{R}(:,1:w+l-\zeta-1) & \mathbf{z}-\mathbf{R}(:,w+l-\zeta)\underline{\hat{s}}_1[k-l] \end{array}\right]$$

make decision on data symbol

$$\underline{\hat{s}}_1[K-1] = \operatorname*{proj}_{\Omega}\left\{\frac{\mathbf{z}(1)}{\mathbf{R}(1,1)}\right\}$$

Table 8.2: Method WAM2 with constraint C4 ($l \geq 0$).

## 8.3 Further Discussion

### 8.3.1 Initialization and Termination

Applying constraint C4, we need some kind of initialization to start the adaptive procedure. Therefore, we assume the knowledge of the first transmitted data symbol $s_1[0]$ (hence, $\underline{\hat{s}}_1[0] = s_1[0]$). The termination, on the other hand, solves $\mathbf{s}_{1,K-w-l}^{(w+l)}$ (the unknown data symbols for the time index $k = K - w$) through backsubstitution. This initialization and termination procedure is illustrated in the full description of methods WAM1 and WAM2 with constraint C4 (see tables 8.1 and 8.2).

### 8.3.2 Relations with Existing Algorithms

As for the block processing methods based on direct symbol estimation, the proposed windowed adaptive processing methods based on direct blind symbol estimation have to the best of our knowledge not yet been studied in the literature.

For a TDMA system with multiple users per time channel (no coding), a method similar to method WAM1 with constraint C4 for $l = -w + 1$ is discussed in [GS96]. This method relies on decision feedback to suppress the ISI as well as the MUI. Therefore, this method heavily depends on a good initialization step, e.g., a rather complex initialization based on block processing [LX97, vdVTP97] (see section 7.5.2). If a wrong starting point is chosen, the method will not be able to converge. *In a DS-CDMA system, the use of coding prevents this problem.*

### 8.3.3 Other Constraints

If the data symbols belong to a constant modulus constellation (assume the modulus of all the elements in $\Omega$ is $\phi$), we can also combine constraint C4 with a norm constraint on the data symbols:

$$\mathbf{s}_{1,0}^{(k-l)} = \underline{\hat{\mathbf{s}}}_{1,0}^{(k-l)} \quad \text{and} \quad \|\mathbf{s}_{1,k-l}^{(w+l)}\|^2 = (w+l)\phi^2 \quad (\text{assume } k > l \text{ and } l > -w).$$

Applying this constraint for one of the methods discussed in section 8.2 will result in an RTLS algorithm with a fixed computational complexity per data symbol period. In the subspace framework and for a TDMA system with a single user per time channel (no coding), such an RTLS algorithm has been discussed in [VM98a] (see also [Van99]). This algorithm can easily be extended for the DS-CDMA system under consideration.

| WAM | constr. | complexity (flops) |
|---|---|---|
| WAM1 | C4 | $8A(wN-\hat{r})(w+l)^2$ |
| WAM2 | C4 | $8AwN(A(Q+1)M+w+l)^2$ |

Table 8.3: Computational complexity of the windowed adaptive processing methods.

We can also make use of a finite alphabet constraint on the data symbols:

$$\mathbf{s}_{1,0}^{(k+w)} \in \Omega^{1\times(k+w)}.$$

Applying this constraint for one of the methods discussed in section 8.2 will result in a Viterbi algorithm with a fixed computational complexity per data symbol period. In the subspace framework and for a TDMA system with a single user per time channel (no coding), such a Viterbi algorithm has been discussed in [VM96] (see also [Van99]). This algorithm can easily be extended for the DS-CDMA system under consideration. Note that another multi-user extension of the Viterbi algorithm of [VM96] can be found in [VM98b]. This algorithm is based on the use of linear block coding.

## 8.4 Computational Complexity

The computational complexities derived in section 8.2 are summarized in **table 8.3**. Note that for the subspace windowed adaptive processing method (method WAM1), this complexity does not yet include the computational complexity of the subspace preprocessing step. A subspace windowed adaptive processing method requires the set of matrices $\{\hat{\mathbf{V}}_{kN+a}\}_{a=A_1}^{A_2}$ at time index $k$. Like in the previous chapter, two subspace preprocessing approaches can be used (we will refer to them as WAP subspace preprocessing approaches since they are used for windowed adaptive processing). The first approach computes the SVD of $\mathbf{Y}_n$ for every $n$ satisfying $A_1 \leq n-N\lfloor n/N\rfloor \leq A_2$ or $A_1 \leq n-N\lceil n/N\rceil \leq A_2$ from scratch, e.g., using the algorithm of [DBMS79]. At time index $k$, we then consider the decompositions corresponding to $\{n=kN+a\}_{a=A_1}^{A_2}$. This approach has the following computational complexity per data symbol period:

$$\min\{A,N\}(16w^2N^2(Q+1)M+12wN(Q+1)^2M^2) \text{ flops.} \tag{8.20}$$

Note that for $A \geq N$, this complexity is independent of $A$. The second approach computes the SVD of $\mathbf{Y}_n$ for every $n$ in a recursive fashion, e.g., using a modified version of the algorithm of [MDPM95] that computes the SVD based on a QR up- and downdating step followed by $n_{it}$ Jacobi iterations. At time index $k$, we then again consider the decompositions corresponding to $\{n=kN+a\}_{a=A_1}^{A_2}$.

This approach has the following computational complexity per data symbol period:

$$N(18w^2N^2+(29+36n_{it})(Q+1)^2M^2+(24+18n_{it})wN(Q+1)M) \text{ flops.} \quad (8.21)$$

Note that this complexity is independent of $A$.

**Remark 8.4.1** Note that if $\hat{r} = (Q+1)M$ the complexity of the two WAP subspace preprocessing approaches can be decreased because $\hat{\mathbf{V}}_{kN+a}$ can then be computed from the QRD of $\mathbf{Y}_{kN+a}$. However, this will not be considered in this work. □

**Remark 8.4.2** Like in the previous chapter, to obtain the simulation results presented in section 8.9, we will always make use of the first approach. The results for the second approach are approximately the same, even for $n_{it} = 1$. □

*Numerical example:* We consider the same system parameters as in the numerical example of section 7.6: $J = 2$, $M = 4$, $L_1 = 2$, $L_2 = 4$ and $N = 2$. Remember that these parameters correspond to system 1 of section 7.10. We further take $Q = 4$.

We first compare the computational complexity per data symbol period of the two WAP subspace preprocessing approaches discussed above (for the second approach, we take $n_{it} = 1$, which usually is sufficient). The results are shown in **figure 8.1**. From this figure it is clear that the second approach is always cheaper than the first approach.

Next, we compare the computational complexity per data symbol period of the different methods discussed in section 8.2. We only consider $A = 1$ and $A = 3$ and we investigate two values of $l$: $l = 0$ and $l = -w + 1$. For the subspace windowed adaptive processing method (method WAM1) we examine two values of $\hat{r}$: $\hat{r} = r$ (when $r$ is exactly estimated) and $\hat{r} = (Q+1)M$ (when $r$ is not estimated). In **figure 8.2**, we compare the computational complexity per data symbol period of method WAM1 with constraint C4, including the second WAP subspace preprocessing approach ($n_{it} = 1$), with the computational complexity per data symbol period of method WAM2 with constraint C4. We observe that for a small value of $A$ and $l$ method WAM2 with constraint C4 can become cheaper than method WAM1 with constraint C4, including the second WAP subspace preprocessing approach. However, for larger values of $A$ and/or $l$, this is no longer the case.

To show the potential computational advantage of adaptive processing over block processing, we finally compare the computational complexity of method WAM1 with constraint C4 ($w = 20$), including the second WAP subspace preprocessing approach ($n_{it} = 1$), with the computational complexity of method

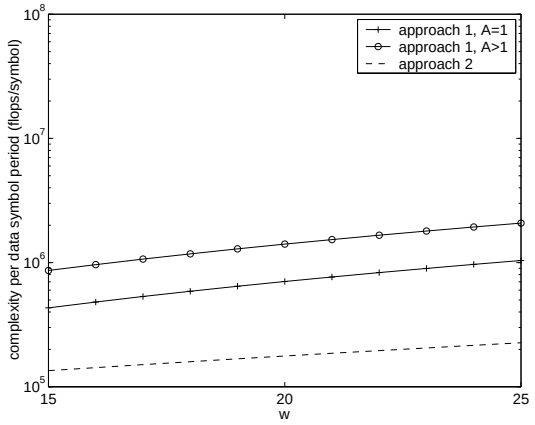



Figure 8.1: Complexity of the WAP subspace preprocessing step as a function of $w$ ($J = 2$, $M = 4$, $L_1 = 2$, $L_2 = 4$, $N = 2$, $Q = 4$).

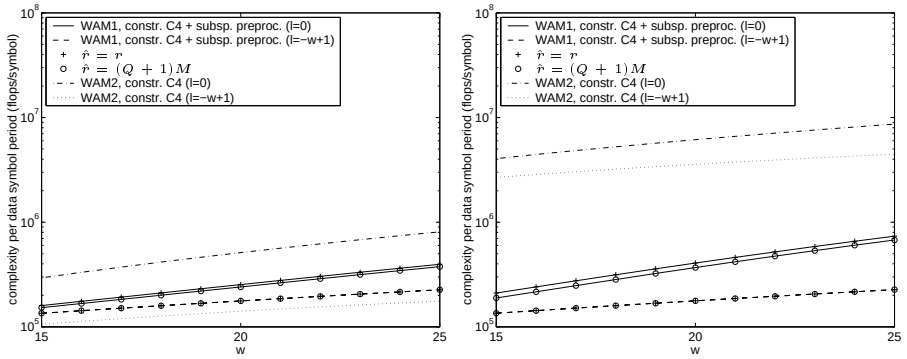



Figure 8.2: Complexity comparison of method WAM1 with constraint C4, including the second subspace preprocessing approach ($n_{it} = 1$), with method WAM2 with constraint C4 as a function of $w$ for $A = 1$, in the left figure, and $A = 3$, in the right figure ($J = 2$, $M = 4$, $L_1 = 2$, $L_2 = 4$, $N = 2$, $Q = 4$).

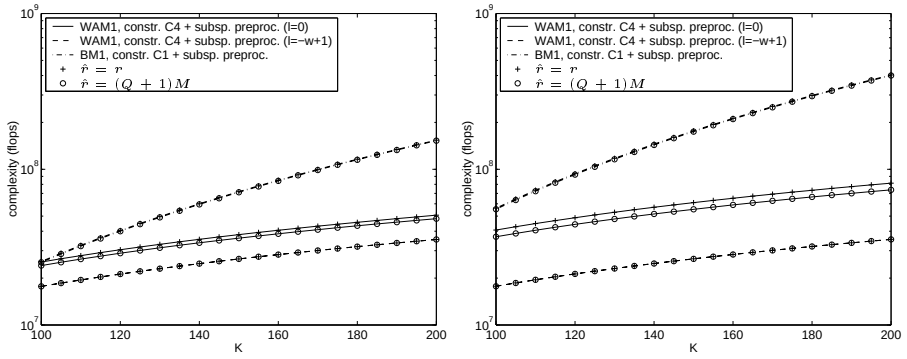



Figure 8.3: Complexity comparison of method WAM1 with constraint C4 ($w = 20$), including the second WAP subspace preprocessing approach ($n_{it} = 1$), with method BM1 with constraint C1, including the second BP subspace preprocessing approach ($n_{it} = 1$), as a function of $K$ for $A = 1$, in the left figure, and $A = 3$, in the right figure ($J = 2$, $M = 4$, $L_1 = 2$, $L_2 = 4$, $N = 2$, $Q = 4$).

BM1 with constraint C1, including the second BP subspace preprocessing approach ($n_{it} = 1$). Note that method BM1 with constraint C1 is the block processing method that is the most related to method WAM1 with constraint C4. The results are shown in **figure 8.3**. We notice that over the range of burst lengths that is shown, method BM1 with constraint C1, including the second BP subspace preprocessing approach, requires the most computations. Moreover, its complexity increases more rapidly with the burst length $K$ than the complexity of method WAM1 with constraint C4, including the second WAP subspace preprocessing approach.

## 8.5 Exponentially Weighted Adaptive Processing

For the exponentially weighted adaptive processing approach, we consider the same data model as for the windowed adaptive processing approach, taking $w = 1$. We further make use of a *forget factor* $\lambda \leq 1$.

**Remark 8.5.1** Note that in case of time varying channels, $\lambda$ should be chosen in correspondence with the coherence time of the channels. □

In contrast with windowed adaptive processing, exponentially weighted adaptive processing only makes sense in a non-subspace framework.

### 8.5.1 Non-Subspace Framework

At time index $k$ ($k = 0, 1, \ldots, K-1$), we solve the following minimization problem (compare with (8.5)):

$$\boxed{\begin{array}{l} \min\limits_{\mathbf{s}_{1,0}^{(k+1)}, f_{1,k}} \{\|f_{1,k}\mathcal{Y}_{0:k}^{exp} - \mathbf{s}_{1,0}^{(k+1)}\mathcal{C}_{1,0:k}^{exp}\|^2\}, \\ \text{s.t. a non-triviality constraint imposed on } \mathbf{s}_{1,0}^{(k+1)} \text{ and/or } f_{1,k}, \end{array}} \tag{8.22}$$

where $\mathcal{Y}_{0:k}^{exp}$ is the $A(Q+1)M \times (k+1)AN$ matrix, given by

$$\mathcal{Y}_{0:k}^{exp} = \begin{bmatrix} \lambda^k \mathcal{Y}_0 & \cdots & \lambda \mathcal{Y}_{k-1} & \mathcal{Y}_k \end{bmatrix},$$

and $\mathcal{C}_{1,0:k}^{exp}$ is the $(k+1) \times (k+1)AN$ matrix, given by

$$\mathcal{C}_{1,0:k}^{exp} = \begin{bmatrix} \lambda^k \mathcal{C}_{1,0} & & & \\ & \ddots & & \\ & & \lambda \mathcal{C}_{1,k-1} & \\ & & & \mathcal{C}_{1,k} \end{bmatrix}.$$

Here, $\mathcal{Y}_k$ and $\mathcal{C}_{1,k}$ are defined like before, taking $w = 1$. Note that here only one linear super chip equalizer $f_{1,k}$ is used, common to all subproblems (with $w = 1$), while in (8.5) all subproblems (with arbitrary $w$) use different linear super chip equalizers (combined in $f_{1,0:k}$).

### 8.5.2 Constraint

In the following section, we discuss an exponentially weighted adaptive processing method based on (8.22). Like before, we only focus on direct blind symbol estimation and make use of only one constraint:

C5 Linear constraint on the data symbols:

$$\mathbf{s}_{1,0}^{(k+1)}(1 : k-l) = \mathbf{s}_{1,0}^{(k-l)} = \underline{\hat{\mathbf{s}}}_{1,0}^{(k-l)} \quad \text{(assume } k > l \text{ and } l \geq 0),$$

where $\underline{\hat{\mathbf{s}}}_{1,0}^{(k-l)}$ is defined as

$$\underline{\hat{\mathbf{s}}}_{1,0}^{(k-l)} = \begin{bmatrix} \underline{\hat{s}}_1[0] & \underline{\hat{s}}_1[1] & \cdots & \underline{\hat{s}}_1[k-l-1] \end{bmatrix},$$

with $\underline{\hat{s}}_1[k-l]$ representing the hard estimate of $s_1[k-l]$ made at time index $k$.

Applying constraint C5 will result in an RLS algorithm with a fixed computational complexity per data symbol period.

**Remark 8.5.2** Note that constraint C5 imposes some kind of decision feedback. The parameter $l$ determines the degree of decision feedback. It is a design parameter to change the number of unknown data symbols or memory size $l+1$. □

## 8.6 Direct Blind Symbol Estimation

In this section, we discuss one exponentially weighted adaptive processing method for direct blind symbol estimation.

### 8.6.1 Exponentially Weighted Adaptive Method EWAM

We consider (8.22) as a minimization problem in $\mathbf{s}_{1,0}^{(k+1)}$. We focus on constraint C5. Note that the size of the minimization problem (8.22) grows with $k$. This problem will be dealt with.

**Example 8.6.1 (constraint C5)** If we use constraint C5 as a non-triviality constraint, (8.22) becomes the following minimization problem at time index $k$:

$$\boxed{\min_{\mathbf{s}_{1,k-l}^{(l+1)}, f_{1,k}} \{\| \left[ \begin{array}{c|c} f_{1,k} & \mathbf{s}_{1,k-l}^{(l+1)} \end{array} \right] \left[ \begin{array}{c} \mathcal{Y}_{0:k}^{exp} \\ \hline -\bar{\mathcal{C}}_{1,0:k}^{exp} \end{array} \right] - \mathbf{b}_{1,0:k}^{exp} \|^2\},} \tag{8.23}$$

where $\mathbf{s}_{1,k-l}^{(l+1)}$ is defined using the notation of (8.3), $\bar{\mathcal{C}}_{1,0:k}^{exp}$ is the $(l+1)\times(k+1)N$ matrix, given by

$$\bar{\mathcal{C}}_{1,0:k}^{exp} = \mathcal{C}_{1,0:k}^{exp}(k-l+1:k+1,:),$$

and $\mathbf{b}_{1,0:k}^{exp}$ is the $1\times(k+1)AwN$ vector, defined as

$$\mathbf{b}_{1,0:k}^{exp} = \underline{\hat{\mathbf{s}}}_{1,0}^{(k-l)} \mathcal{C}_{1,0:k}^{exp}(1:k-l,:).$$

This LS problem can be solved recursively by an RLS algorithm based on QR updating with an R-matrix of *fixed size* $(A(Q+1)M+l+1)\times(A(Q+1)M+l+1)$, as we will show next.

First, rewrite (8.23) as

$$\min_{\mathbf{s}_{1,k-l}^{(l+1)},f_{1,k}} \{\| \left[\begin{array}{c|c} \mathcal{Y}_{0:k}^{expT} & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^{expT}\} \end{array}\right] \left[\begin{array}{c} f_{1,k}^T \\ \hline \mathrm{fr}\{\mathbf{s}_{1,k-l}^{(l+1)T}\} \end{array}\right] - \mathbf{b}_{1,0:k}^{expT}\|^2\}. \tag{8.24}$$

To solve (8.24), we can make use of the QRD [GVL89]. The QRD of the matrix

$$\left[\begin{array}{c|c|c} \mathcal{Y}_{0:k}^{expT} & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^{expT}\} & \mathbf{b}_{1,0:k}^{expT} \end{array}\right],$$

focusing on all columns but the last, is given by

$$\left[\begin{array}{c|c|c} \mathcal{Y}_{0:k}^{expT} & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^{expT}\} & \mathbf{b}_{1,0:k}^{expT} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{Q}_k & \star \end{array}\right] \left[\begin{array}{c|c} \mathbf{R}_k & \mathbf{z}_k \\ \hline \mathbf{O} & \star \end{array}\right], \tag{8.25}$$

where $\mathbf{Q}_k\mathbf{R}_k$ is the QRD of $\left[\begin{array}{c|c} \mathcal{Y}_{0:k}^{expT} & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^{expT}\} \end{array}\right]$ and $\left[\begin{array}{c|c} \mathbf{Q}_k & \star \end{array}\right]$ is unitary. Note that $\mathbf{R}_k$ has size $(A(Q+1)M+l+1) \times (A(Q+1)M+l+1)$. We assume $\left[\begin{array}{c|c} \mathcal{Y}_{0:k}^{expT} & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:k}^{expT}\} \end{array}\right]$ has full column rank.

Using a reasoning similar as before, $\underline{\hat{s}}_1[k-l]$ is obtained as $(d = A(Q+1)M)$

$$\underline{\hat{s}}_1[k-l] = \underset{\Omega}{\mathrm{proj}}\left\{ \frac{\mathbf{z}_k(d+l+1)}{\mathbf{R}_k(d+l+1, d+l+1)} \right\}$$

and we can derive the following updating formula for $\mathbf{R}_k$ and $\mathbf{z}_k$ $(d = A(Q+1)M)$:

$$\left[\begin{array}{c|c} \mathbf{R}_{k+1} & \mathbf{z}_{k+1} \\ \hline \mathbf{O} & \star \end{array}\right] \leftarrow$$

$$\mathbf{P}^H \left[\begin{array}{c|c|c|c} \lambda\mathbf{R}_k(:,1:d) & \mathbf{0} & \lambda\mathbf{R}_k(:,d+1:d+l) & \lambda\mathbf{u}_k \\ \hline \mathcal{Y}_{k+1}^T & -\mathcal{C}_{1,k+1}^T & \mathbf{O} & \mathbf{0} \end{array}\right],$$

where

$$\mathbf{u}_k = \mathbf{z}_k - \mathbf{R}_k(:, d+l+1)\underline{\hat{s}}_1[k-l].$$

Applying Householder QR updating [GVL89], the computational complexity per data symbol period is

$$8AN(A(Q+1)M+l+1)^2 \text{ flops.} \tag{8.26}$$

It is clear that, while the size of the minimization problem (8.22) grows with $k$, the computational complexity per data symbol period is independent of $k$. Method EWAM with constraint C5 is summarized in **table 8.4** (see section 8.7.1 for initialization and termination details). □

1. initialization:

$$\left[\begin{array}{c|c} \mathbf{R} & \mathbf{z} \\ \hline \mathbf{O} & \star \end{array}\right] \leftarrow \mathbf{P}^H \left[\begin{array}{c|c|c} \mathcal{Y}_{0:l+1}^{expT} & -\mathrm{fc}\{\bar{\mathcal{C}}_{1,0:l+1}^{expT}\} & \mathbf{b}_{1,0:l+1}^{expT} \end{array}\right]$$

2. propagation:
for $k = l+1, \ldots, K-2$

- make decision on data symbol

$$\underline{\hat{s}}_1[k-l] = \underset{\Omega}{\mathrm{proj}}\Big\{\frac{\mathbf{z}(d+l+1)}{\mathbf{R}(d+l+1, d+l+1)}\Big\}$$

- update $\mathbf{R}$ and $\mathbf{z}$

$$\left[\begin{array}{c|c} \mathbf{R} & \mathbf{z} \\ \hline \mathbf{O} & \star \end{array}\right] \leftarrow$$

$$\mathbf{P}^H \left[\begin{array}{c|c|c|c} \lambda\mathbf{R}(:, 1:d) & \mathbf{0} & \lambda\mathbf{R}(:, d+1:d+l) & \lambda(\mathbf{z} - \mathbf{R}(:, d+l+1)\underline{\hat{s}}_1[k-l]) \\ \hline \mathcal{Y}_{k+1}^T & -\mathcal{C}_{1,k+1}^T & \mathbf{O} & \mathbf{0} \end{array}\right]$$

3. termination:
for $k = K-1, \ldots, K+l-2$ (we define $\zeta = k - K + 1$)

- make decision on data symbol

$$\underline{\hat{s}}_1[k-l] = \underset{\Omega}{\mathrm{proj}}\Big\{\frac{\mathbf{z}(d+l+1-\zeta)}{\mathbf{R}(d+l+1-\zeta, d+l+1-\zeta)}\Big\}$$

- update $\mathbf{R}$ and $\mathbf{z}$

$$\left[\begin{array}{c|c} \mathbf{R} & \mathbf{z} \\ \hline \mathbf{O} & \star \end{array}\right] \leftarrow \left[\begin{array}{c|c} \lambda\mathbf{R}(:, 1:d+l-\zeta) & \lambda(\mathbf{z} - \mathbf{R}(:, d+l+1-\zeta)\underline{\hat{s}}_1[k-l]) \end{array}\right]$$

make decision on data symbol

$$\underline{\hat{s}}_1[K-1] = \underset{\Omega}{\mathrm{proj}}\Big\{\frac{\mathbf{z}(d+1)}{\mathbf{R}(d+1, d+1)}\Big\}$$

Table 8.4: Method EWAM with constraint C5.

## 8.7 Further Discussion

### 8.7.1 Initialization and Termination

Applying constraint C5, we need some kind of initialization to start the adaptive procedure. Therefore, we again assume the knowledge of the first transmitted data symbol $s_1[0]$ (hence, $\underline{\hat{s}}_1[0] = s_1[0]$). The termination, on the other hand, solves $\mathbf{s}_{1,K-l-1}^{(l+1)}$ (the unknown data symbols for the time index $k = K-1$) through backsubstitution. This initialization and termination procedure is illustrated in the full description of method EWAM with constraint C5 (see table 8.4).

### 8.7.2 Relations with Existing Algorithms

The proposed exponentially-weighted adaptive processing method based on direct blind symbol estimation has to the best of our knowledge not yet been studied in the literature.

## 8.8 Computational Complexity

The computational complexity of method EWAM with constraint C5 is $w$ times smaller than the computational complexity of method WAM2 with constraint C4, if both methods have the same number of unknown data symbols or memory size, i.e., if the value of $l+1$ for method EWAM with constraint C5 is equal to the value of $w+l$ for method WAM2 with constraint C4.

## 8.9 Simulation Results

In this section, we will perform some simulations on system 1 of section 7.10 ($M = 4$, $J = 2$, $N = 2$, $L_1 = 2$, $L_2 = 4$ and $\delta_1 = \delta_2 = 0$). We assume that the data symbol sequences $\{s_j[k]\}_{j=1}^{J}$ are mutually independent and zero-mean white. We further assume that the additive noises $\{e^{(m)}[n]\}_{m=1}^{M}$ are mutually independent and zero-mean white Gaussian with variance $\sigma_e^2$. Finally, we assume that the system order is exactly estimated ($\hat{r} = r$). We only focus on the performance of the first user and consider two scenarios: scenario 1 and scenario 3 (see section 7.10). Remember from simulation 1 in section 7.10 that scenario 3 resulted in the best performance for the first user. We consider a burst length of $K = 151$ and take $Q = 4$. For the windowed adaptive processing

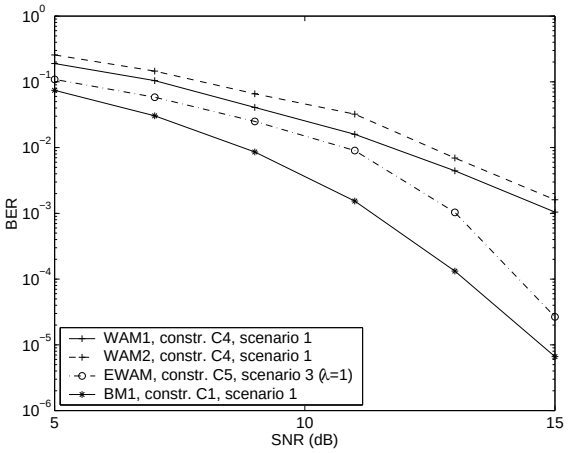



Figure 8.4: BER of user 1 as a function of SNR for different methods in a time invariant environment (system 1, $\hat{r} = r$, $Q = 4$, $K = 151$).

methods we take $w = 20$. For the exponentially weighted adaptive processing method we take $l = 19$.

*Simulation 1:* We first consider a set of time invariant channels (the same set as used in section 7.10). We assume that all users have the same received power. The SNR at the input of the receiver is defined as in (7.58). In **figure 8.4**, we show the performance of different adaptive processing methods and compare it with the performance of method BM1 with constraint C1. As could be expected, the block processing method has a better performance than the adaptive processing methods, because it exploits the fact that the channels are time invariant. However, we have seen in section 8.4, that this better performance goes at the expense of a larger computational complexity.

*Simulation 2:* In this second simulation, we consider a set of time varying channels (see remarks 8.1.2 and 8.5.1). We assume a Doppler frequency of 100 Hz (speed of 120 km/h for a carrier frequency of 900 MHz) and a data symbol rate of 50 ksymbols/s. **Figure 8.5** shows the typical time variation of a channel tap. We assume that all users have the same *average* (over time) received power. The SNR at the input of the receiver is defined as the *average* (over time) of the SNR defined in (7.58). In **figure 8.6**, we show the same results as in figure 8.4, but now for time varying channels. We see that the block processing method can no longer be used, while the adaptive processing methods still show a reasonable performance. Note that it would also be possible to split the burst into different parts and apply the block processing method on all these parts separately, reducing the time variation of the channels over one part. However, this will not be investigated in this work.

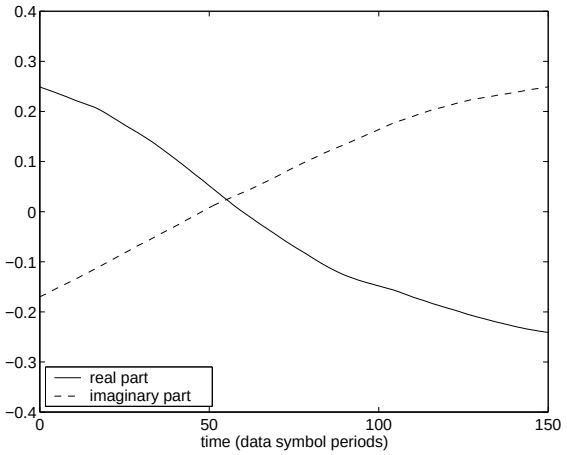



Figure 8.5: Typical time variation of a channel tap.

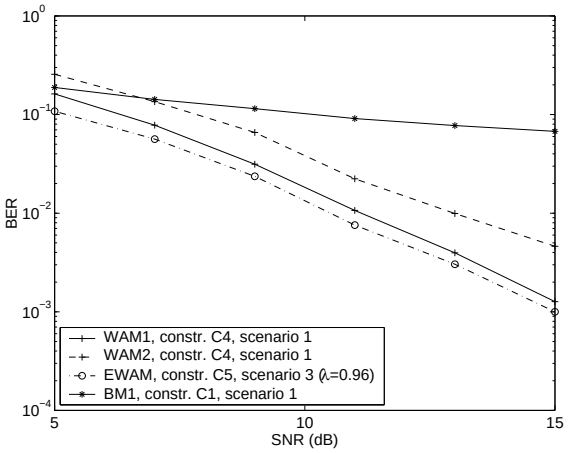



Figure 8.6: BER of user 1 as a function of SNR for different methods in a time varying environment (system 1, $\hat{r} = r$, $Q = 4$, $K = 151$).

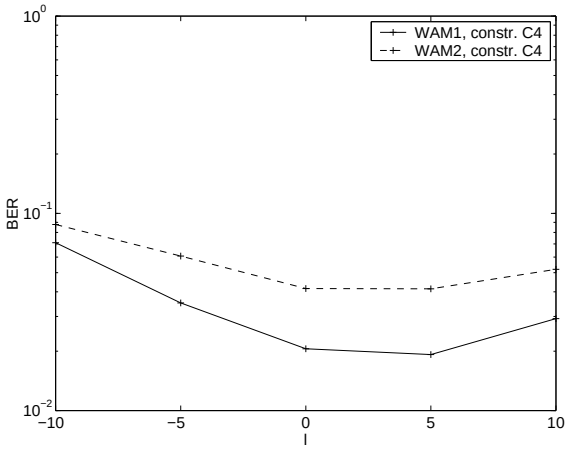



Figure 8.7: BER of user 1 for different windowed adaptive processing methods with constraint C4 in a time varying environment as a function of $l$ (system 1, scenario 1, $\hat{r} = r$, $Q = 4$, $K = 151$, $SNR$=10 dB).

Next, we consider the influence of $l$ on the performance of the windowed adaptive processing methods with constraint C4. Remember that this parameter $l$ can be used to change the number of unknown data symbols or memory size $w + l$, once the window length $w$ is fixed. We only focus on scenario 1 and consider an SNR of 10 dB. The results are shown in **figure 8.7**. Clearly, there is an optimal value of $l$. This can be explained as follows. If $l$ decreases, the number of unknown data symbols decreases and there is more chance to make an error, because we make a decision on the data symbols too soon. On the other hand, if $l$ increases, the number of zeros in $\mathbf{b}_{1,0:k}$ (see (8.8)) increases (because of the structure of $\mathcal{C}_{1,0:k}$) and therefore the solution for $\mathbf{s}_{1,k-l}^{(w+l)}$ tends more and more to zero, leading to an increased BER.

## 8.10 Conclusions

In this chapter, deterministic blind ST-RAKE receivers based on adaptive processing have been developed. First, windowed adaptive processing has been discussed. We have considered a subspace framework as well as a non-subspace framework. We have only focused on direct blind symbol estimation. Second, exponentially weighted adaptive processing has been discussed. This type of adaptive processing only makes sense in a non-subspace framework. Like before, we have only focused on direct blind symbol estimation.

We have shown that *there is a potential computational advantage of adaptive processing over block processing.* Moreover, simulation results have illustrated that *the proposed adaptive processing methods perform well both in a time invariant and a time varying environment.* Note that in this time varying environment the performance of block processing on a large block of received samples is not satisfactory anymore.

# Chapter 9

# Extensions for a DMT-DS-CDMA System

In this chapter, we show how the deterministic blind ST-RAKE receivers based on block processing, presented in chapter 7, can also be applied on a DMT-DS-CDMA system (similar reasonings are possible for the deterministic blind ST-RAKE receivers based on adaptive processing, presented in chapter 8). The data model is introduced in section 9.1.

We show that the use of a cyclic prefix reduces the ICI in a DMT-DS-CDMA system. For instance, if the system is synchronous, it is possible to look at this system in such a way that it seems as if each channel order is reduced with the length of the cyclic prefix (if it is larger than the length of the cyclic prefix) or is reduced to zero (if it is smaller than or equal to the length of the cyclic prefix). As a result of this reduction of ICI, the computational complexity of the block processing methods, presented in chapter 7, can be reduced when they are carried out for a tone of a DMT-DS-CDMA system w.r.t. the case they are carried out for the related DS-CDMA system. This will be described in section 9.2. Moreover, the simulation results presented in section 9.3, show that this reduction in computational complexity even comes with a performance improvement. Conclusions are given in section 9.4.

**Remark 9.0.1** The approaches we present in this chapter can be viewed as blind and multi-user extensions of the (training-based and single-user) per tone equalization approach presented in [VALM+99]. Since we do not require that the system is synchronous (although this can be advantageous) and since each channel order does not have to be smaller than or equal to the length of the cyclic prefix, they also extend the results of the training-based combined

OFDM/SDMA approach of [VTEDM99]. □

## 9.1 Data Model

We consider a DMT-DS-CDMA system with $M$ receive antennas. The data model that will be used is given by (2.17), which is repeated here for convenience:

$$\mathbf{y}^{st}[n] = \sum_{j=1}^{J} \sum_{n'=-\infty}^{+\infty} \mathbf{g}_j^{st}[n'] x_j[n-n'] + \mathbf{e}^{st}[n]. \tag{9.1}$$

At the $m$th receive antenna, we then block the received sequence $y^{(m)}[n]$ (block size $P+\mu$):

$$\mathbf{y}^{dmt(m)}[\nu;n] = \left[ y^{(m)}[\nu(P+\mu)+n] \quad \cdots \quad y^{(m)}[(\nu+1)(P+\mu)+n-1] \right]^T.$$

We further remove the cyclic prefix of length $\mu$ and perform a discrete Fourier transformation (DFT):

$$\tilde{\mathbf{y}}^{dmt(m)}[\nu;n] = 1/\sqrt{P}\,\mathcal{F}_P \mathbf{P}_r \mathbf{y}^{dmt(m)}[\nu;n], \tag{9.2}$$

where $\mathcal{F}_P$ represents the $P \times P$ DFT matrix and $\mathbf{P}_r$ is the $P \times (P+\mu)$ matrix, given by

$$\mathbf{P}_r = \left[ \begin{array}{c|c} \mathbf{O} & \mathbf{I}_P \end{array} \right].$$

Introducing the notation

$$\tilde{\mathbf{y}}^{dmt(m)}[\nu;n] = \left[ \tilde{y}_1^{(m)}[\nu;n] \quad \tilde{y}_2^{(m)}[\nu;n] \quad \cdots \quad \tilde{y}_P^{(m)}[\nu;n] \right],$$

and stacking the results for the $p$th tone obtained from the $M$ receive antennas:

$$\tilde{\mathbf{y}}_p^{st}[\nu;n] = \left[ \tilde{y}_p^{(1)}[\nu;n] \quad \tilde{y}_p^{(2)}[\nu;n] \quad \cdots \tilde{y}_p^{(M)}[\nu;n] \right]^T,$$

we can write

$$\tilde{\mathbf{y}}_p^{st}[\nu;n] = \sum_{j=1}^{J} \sum_{n'=-\infty}^{+\infty} \mathbf{g}_j^{st}[n'] \tilde{x}_{j,p}[\nu;n-n'] + \tilde{\mathbf{e}}_p^{st}[\nu;n], \tag{9.3}$$

where $\tilde{\mathbf{e}}_p^{st}[\nu;n]$ is similarly defined as $\tilde{\mathbf{y}}_p^{st}[\nu;n]$ and

$$\tilde{x}_{j,p}[\nu;n] = 1/\sqrt{P} \sum_{n'=0}^{P-1} \mathcal{F}_P(p,n'+1) x_j[\nu(P+\mu)+\mu+n'+n].$$

Note that because we use a cyclic prefix of length $\mu$, we have

$$\tilde{x}_{j,p}[\nu;-l] = e^{-i2\pi pl/P}\tilde{x}_{j,p}[\nu], \text{ for } l = 0,1,\ldots,\mu. \tag{9.4}$$

For $l < 0$ and $l > \mu$, $\tilde{x}_{j,p}[\nu;-l]$ incorporates contributions from other tones than tone $p$ at time index $\nu$ as well as from tones at other time indices than time index $\nu$. Further note that the DFT in (9.2) can be viewed as a filtering operation and that the presentation in (9.3) is obtained by swapping this filtering operation with the vector channel $\mathbf{g}_j^{st}[n]$.

Let us then introduce the following $(Q+1)M \times \beta N$ output matrix (not a Hankel structure) for the $p$th tone:

$$\tilde{\mathbf{Y}}_{p,\nu;n} = \begin{bmatrix} \tilde{\mathbf{y}}_p^{st}[\nu;n] & \tilde{\mathbf{y}}_p^{st}[\nu+1;n] & \cdots & \tilde{\mathbf{y}}_p^{st}[\nu+\beta N-1;n] \\ \vdots & \vdots & & \vdots \\ \tilde{\mathbf{y}}_p^{st}[\nu;n+Q] & \tilde{\mathbf{y}}_p^{st}[\nu+1;n+Q] & \cdots & \tilde{\mathbf{y}}_p^{st}[\nu+\beta N-1;n+Q] \end{bmatrix},$$

where $\beta > 0$ is the amount of DMT subsymbols 'under consideration' (this will be specified later) and $Q \geq 0$ determines the amount of *temporal smoothing*. This output matrix for the $p$th tone can be written as

$$\tilde{\mathbf{Y}}_{p,\nu;n} = \sum_{j=1}^{J} \mathcal{G}_j \tilde{\mathbf{X}}_{j,p,\nu;n} + \tilde{\mathbf{E}}_{p,\nu;n}, \tag{9.5}$$

where $\tilde{\mathbf{E}}_{p,\nu;n}$ is similarly defined as $\tilde{\mathbf{Y}}_{p,\nu;n}$, $\mathcal{G}_j$ is the $(Q+1)M \times r_j$ ($r_j = Q+1+L_j$) channel matrix (with Toeplitz structure) for the $j$th user, given by (7.3), and $\tilde{\mathbf{X}}_{j,p,\nu;n}$ is the following $r_j \times \beta N$ input matrix (not a Hankel structure) for the $p$th tone of the $j$th user:

$$\tilde{\mathbf{X}}_{j,p,\nu;n} = \begin{bmatrix} \tilde{\mathbf{x}}_{j,p,\nu;n-\delta_j-L_j}^T & \cdots & \tilde{\mathbf{x}}_{j,p,\nu;n-\delta_j+Q}^T \end{bmatrix}^T,$$

with

$$\tilde{\mathbf{x}}_{j,p,\nu;n} = \begin{bmatrix} \tilde{x}_{j,p}[\nu;n] & \tilde{x}_{j,p}[\nu+1;n] & \cdots & \tilde{x}_{j,p}[\nu+\beta N-1;n] \end{bmatrix}.$$

Note that (9.5) can also be written as

$$\tilde{\mathbf{Y}}_{p,\nu;n} = \mathcal{G}\tilde{\mathbf{X}}_{p,\nu;n} + \tilde{\mathbf{E}}_{p,\nu;n},$$

where $\mathcal{G}$ is the $(Q+1)M \times r$ ($r = \sum_{j=1}^{J} r_j$) channel matrix, given by (7.6), and $\tilde{\mathbf{X}}_{p,\nu;n}$ is the following $r \times \beta N$ input matrix:

$$\tilde{\mathbf{X}}_{p,\nu;n} = \begin{bmatrix} \tilde{\mathbf{X}}_{1,p,\nu;n}^T & \cdots & \tilde{\mathbf{X}}_{J,p,\nu;n}^T \end{bmatrix}^T.$$

Because of (9.4), which is equivalent to

$$\tilde{\mathbf{x}}_{j,p,\nu;-l} = e^{-i2\pi pl/P}\tilde{\mathbf{x}}_{j,p,\nu}, \quad \text{for } l = 0, 1, \ldots, \mu, \tag{9.6}$$

with

$$\tilde{\mathbf{x}}_{j,p,\nu} = \begin{bmatrix} \tilde{x}_{j,p}[\nu] & \tilde{x}_{j,p}[\nu+1] & \cdots & \tilde{x}_{j,p}[\nu+\beta N-1] \end{bmatrix}, \tag{9.7}$$

we can rewrite model (9.5) as

$$\tilde{\mathbf{Y}}_{p,\nu;n} = \sum_{j=1}^{J} \mathcal{G}^{red}_{j,p;n}\tilde{\mathbf{X}}^{red}_{j,p,\nu;n} + \tilde{\mathbf{E}}_{p,\nu;n}, \tag{9.8}$$

where $\tilde{\mathbf{X}}^{red}_{j,p,\nu;n}$ is the $r^{red}_{j;n} \times \beta N$ input matrix obtained from $\tilde{\mathbf{X}}_{j,p,\nu;n}$ as follows. If $\tilde{\mathbf{X}}_{j,p,\nu;n}$ has no vectors from the set $\{\tilde{\mathbf{x}}_{j,p,\nu;-l}\}_{l=0}^{\mu}$ as a row or only the vector $\tilde{\mathbf{x}}_{j,p,\nu;0} = \tilde{\mathbf{x}}_{j,p,\nu}$ as a row, then $\tilde{\mathbf{X}}^{red}_{j,p,\nu;n} = \tilde{\mathbf{X}}_{j,p,\nu;n}$. Else, we remove from $\tilde{\mathbf{X}}_{j,p,\nu;n}$ all the rows that correspond to vectors from the set $\{\tilde{\mathbf{x}}_{j,p,\nu;-l}\}_{l=0}^{\mu}$ and insert the vector $\tilde{\mathbf{x}}_{j,p,\nu}$ at the place where we have removed them. The corresponding $(Q+1)M \times r^{red}_{j;n}$ channel matrix $\mathcal{G}^{red}_{j,p;n}$ is then easily obtained using (9.6). If $\tilde{\mathbf{X}}_{j,p,\nu;n}$ has no vectors from the set $\{\tilde{\mathbf{x}}_{j,p,\nu;-l}\}_{l=0}^{\mu}$ as a row or only the vector $\tilde{\mathbf{x}}_{j,p,\nu;0} = \tilde{\mathbf{x}}_{j,p,\nu}$ as a row, then $\mathcal{G}^{red}_{j,p;n} = \mathcal{G}_j$. Else, every channel matrix from the set $\{\mathcal{G}^{red}_{j,p;n}\}_{p=1}^{P}$ has $r^{red}_{j;n} - 1$ columns in common with $\mathcal{G}_j$. The position of these columns in $\mathcal{G}_j$ will depend on $n$. The remaining column will depend on $n$ and $p$. Note that (9.8) can also be written as

$$\tilde{\mathbf{Y}}_{p,\nu;n} = \mathcal{G}^{red}_{p;n}\tilde{\mathbf{X}}^{red}_{p,\nu;n} + \tilde{\mathbf{E}}_{p,\nu;n}, \tag{9.9}$$

where $\mathcal{G}^{red}_{p;n}$ is the $(Q+1)M \times r^{red}_{n}$ ($r^{red}_{n} = \sum_{j=1}^{J} r^{red}_{j;n}$) channel matrix, given by

$$\mathcal{G}^{red}_{p;n} = \begin{bmatrix} \mathcal{G}^{red}_{1,p;n} & \cdots & \mathcal{G}^{red}_{J,p;n} \end{bmatrix},$$

and $\tilde{\mathbf{X}}^{red}_{p,\nu;n}$ is the following $r^{red}_{n} \times \beta N$ input matrix:

$$\tilde{\mathbf{X}}^{red}_{p,\nu;n} = \begin{bmatrix} \tilde{\mathbf{X}}^{redT}_{1,p,\nu;n} & \tilde{\mathbf{X}}^{redT}_{2,p,\nu;n} & \cdots & \tilde{\mathbf{X}}^{redT}_{J,p,\nu;n} \end{bmatrix}^{T}.$$

Note that we have $r_j - \mu \leq r^{red}_{j;n} \leq r_j$ and $r - J\mu \leq r^{red}_{n} \leq r$.

We consider a per tone burst length of $K$ DMT subsymbols ($\tilde{r}_{j,p}[\kappa] \neq 0$, for $\kappa = 0, 1, \ldots, K-1$, and $\tilde{r}_{j,p}[\kappa] = 0$, for $\kappa < 0$ and $\kappa \geq K$) and constant modulus code symbols:

$$|c_{j,p}[\nu]| = \frac{1}{\sqrt{N}}, \quad \text{for } \nu = 0, 1, \ldots, \rho N - 1. \tag{9.10}$$

Note that we do not restrict ourselves to the use of short code sequences. We assume that the $p$th tone of the first user ($j = 1$) is desired and that $\delta_1$ is known at the receiver. We further assume w.l.o.g. that $\delta_1 = 0$.

## 9.2 Block Processing

For the block processing approach, we consider the data model presented in (9.8) or (9.9), with $\nu = 0$, $n = a$, where $a$ represents the *processing delay*, and $\beta = K$, where $K$ is the per tone burst length. Let us then define

$$\tilde{\mathbf{r}}_{1,p} = \begin{bmatrix} \tilde{r}_{1,p}[0] & \tilde{r}_{1,p}[1] & \cdots & \tilde{r}_{1,p}[K-1] \end{bmatrix}. \tag{9.11}$$

Using (2.2) and (9.7), it is clear that this vector is 'contained' in $\tilde{\mathbf{x}}_{1,p,0}$:

$$\tilde{\mathbf{x}}_{1,p,0} = \begin{bmatrix} \tilde{r}_{1,p}[0]\mathbf{c}_{1,p}[0] & \cdots & \tilde{r}_{1,p}[K-1]\mathbf{c}_{1,p}[K-1] \end{bmatrix} = \tilde{\mathbf{r}}_{1,p}\mathbf{C}_{1,p}, \tag{9.12}$$

where $\mathbf{c}_{1,p}[\kappa]$ is the code vector used to spread the DMT subsymbol $\tilde{r}_{1,p}[\kappa]$ (spreading factor $N$):

$$\mathbf{c}_{1,p}[\kappa] =$$
$$\begin{bmatrix} c_{1,p}[(\kappa \bmod \rho)N] & c_{1,p}[(\kappa \bmod \rho)N+1] & \cdots & c_{1,p}[(\kappa \bmod \rho)N+N-1] \end{bmatrix},$$

and $\mathbf{C}_{1,p}$ is the $K \times KN$ code matrix, defined as

$$\mathbf{C}_{1,p} = \begin{bmatrix} \mathbf{c}_{1,p}[0] & & \\ & \ddots & \\ & & \mathbf{c}_{1,p}[K-1] \end{bmatrix}. \tag{9.13}$$

The vector $\tilde{\mathbf{x}}_{1,p,0}$ is a row of every input matrix from the set $\{\tilde{\mathbf{X}}^{red}_{p,0;a}\}_{a=-Q-\mu}^{L_1}$ and is therefore 'contained' in every output matrix from the set $\{\tilde{\mathbf{Y}}_{p,0;a}\}_{a=-Q-\mu}^{L_1}$. The problem addressed here is to compute $\tilde{\mathbf{r}}_{1,p}$ from $\{\tilde{\mathbf{Y}}_{p,0;a}\}_{a=A_1}^{A_2}$, with $-Q-\mu \leq A_1 \leq A_2 \leq L_1$ ($A = A_2 - A_1 + 1$), based only on the knowledge of the code sequence $c_{1,p}[\nu]$. The solutions we will describe for this problem are based on the following assumptions:

**Assumption 9.2.1** The channel matrix $\mathcal{G}^{red}_{p;a}$ has full column rank $r^{red}_a$ ($r^{red}_a$ is then called the delay-$a$ system order), for $a = A_1, A_1+1, \ldots, A_2$. □

**Assumption 9.2.2** The input matrix $\tilde{\mathbf{X}}^{red}_{p,0;a}$ has full row rank $r^{red}_a$, for $a = A_1, A_1+1, \ldots, A_2$. □

The first assumption requires that

$$(Q+1)M \geq r^{red}_a, \quad \text{for } a = A_1, A_1+1, \ldots, A_2. \tag{9.14}$$

The second assumption, on the other hand, requires that

$$KN \geq r^{red}_a, \quad \text{for } a = A_1, A_1+1, \ldots, A_2.$$

**Example 9.2.3** Assume that $\delta_j = 0$, for $j = 1, 2, \ldots, J$ (synchronous users). Further, assume that $L_j \leq \mu$, for $j = 1, 2, \ldots, J'$, and $L_j > \mu$, for $j = J'+1, J'+2, \ldots, J$. For $j = 1, 2, \ldots, J'$, we then know that the vectors $\{\tilde{\mathbf{x}}_{j,p,\nu;-l}\}_{l=0}^{L_j}$ are all rows of the $r_j \times KN$ matrix $\tilde{\mathbf{X}}_{j,p,\nu;0}$ and hence $\tilde{\mathbf{X}}_{j,p,\nu;0}^{red}$ will contain $r_j - L_j$ rows: $r_{j;0}^{red} = r_j - L_j$. For $j = J'+1, J'+2, \ldots, J$, on the other hand, we then know that the vectors $\{\tilde{\mathbf{x}}_{j,p,\nu;-l}\}_{l=0}^{\mu}$ are all rows of the $r_j \times KN$ matrix $\tilde{\mathbf{X}}_{j,p,\nu;0}$ and hence $\tilde{\mathbf{X}}_{j,p,\nu;0}^{red}$ will contain $r_j - \mu$ rows: $r_{j;0}^{red} = r_j - \mu$. It is therefore clear that $r_0^{red} = r - (J - J')\mu - \sum_{j=1}^{J'} L_j$. Hence, for $a = 0$, it seems as if the channel order of the $j$th user is reduced to 0 (if $L_j \leq \mu$) or reduced with $\mu$ (if $L_j > \mu$). Taking $A_1 = A_2 = 0$ ($A = 1$), (9.14) can be written as

$$(Q+1)M \geq r - (J - J')\mu - \sum_{j=1}^{J'} L_j.$$

Since $r = \sum_{j=1}^{J} r_j = (Q+1)J + \sum_{j=1}^{J} L_j$, this is equivalent with

$$(Q+1)(M - J) \geq \sum_{j=J'+1}^{J} (L_j - \mu). \tag{9.15}$$

□

The above block processing problem is similar to the block processing problem presented in section 7.2. Hence, to solve this problem, we can follow the same ideas as presented in sections 7.2.1 and 7.2.2 and use the same block processing methods as discussed in sections 7.3 and 7.4.

### 9.2.1 Comparison with DS-CDMA

In this section, we examine how the complexity of the block processing methods, presented in chapter 7, behave when they are carried out for a tone of a DMT-DS-CDMA system w.r.t. the case they are carried out for the related DS-CDMA system.

First of all, note that, for the part of a subspace block processing method corresponding to processing delay $a$, with $A_1 \leq a \leq A_2$, and carried out for the $p$th tone of the DMT-DS-CDMA system, we make use of a possibly $a$-dependent estimate $\hat{r}_a^{red}$ of the delay-$a$ system order $r_a^{red}$, while, for the part of a subspace block processing method corresponding to the same processing delay $a$ and carried out for the related DS-CDMA system, we make use of an $a$-independent estimate $\hat{r}$ of the system order $r$. In the complexity formulas of section 7.6 we should therefore replace $A\hat{r}$ by $\sum_{a=A_1}^{A_2} \hat{r}_a^{red}$.

Assume now that for the DMT-DS-CDMA system as well as for the related DS-CDMA system we use the same values for $K$ and $A$ (remember that for the DMT-DS-CDMA system $K$ represents the per tone burst length, while for the related DS-CDMA system $K$ represents the burst length) and that $KN$ is very large. Further, assume that we always take the minimal required value for $Q$ and that we take $\hat{r}_a^{red} = r_a^{red}$, for $a = A_1, A_1 + 1, \ldots, A_2$, for the DMT-DS-CDMA system, and $\hat{r} = r$, for the related DS-CDMA system. Finally, assume that there is a reduction of the ICI ($\sum_{a=A_1}^{A_2} r_a^{red} < Ar$), such that the minimal required value for $Q$ for the DMT-DS-CDMA system is smaller than the one for the related DS-CDMA system. To illustrate that this is possible, compare for instance (9.15) with (7.12).

It is then clear that the BP subspace preprocessing step for the $p$th tone of the DMT-DS-CDMA system has a smaller complexity than the one for the related DS-CDMA system (this is true for both approaches; see (7.48) and (7.49)). It is then also clear that the indirect versions of method BM1 with constraint C2 and method BM3 with constraint C3 for the $p$th tone of the DMT-DS-CDMA system have a smaller complexity than the ones for the related DS-CDMA system (see (7.29)). The same holds for the complexity of the direct version of method BM3 with constraint C3 (see (7.35)). The complexity of the direct version of method BM1 with constraint C2, on the other hand, stays more or less the same (see (7.27)). The same holds for the complexity of method BM1 with constraint C1 (see (7.25)).

Focusing on the non-subspace block processing methods, it is then clear that method BM4 with constraint C3 for the $p$th tone of the DMT-DS-CDMA system has a smaller complexity than the one for the related DS-CDMA system (see (7.41)). The same holds for the methods BM2 and BM5 with constraint C1 (see (7.31)).

In conclusion, considering the scenario mentioned above, all the discussed block processing methods for the $p$th tone of the DMT-DS-CDMA system are cheaper than the ones for the related DS-CDMA system.

**Remark 9.2.4** In all the above complexity comparisons, we have not taken into account the complexity of the DFT and IDFT operations, that have to be performed in a DMT-DS-CDMA system. However, note in this context that a DFT respectively IDFT can be implemented easily by means of a fast Fourier transformation (FFT) respectively inverse fast Fourier transformation (IFFT). Further, note that when $\tilde{\mathbf{y}}^{dmt(m)}[\nu; n]$ (see (9.2)) is known, $\tilde{\mathbf{y}}^{dmt(m)}[\nu; n+1]$ can easily be computed using a sliding FFT (see [VALM+99]). □

**Remark 9.2.5** The ideas and methods discussed in sections 7.7, 7.8 and 7.9 can also be applied on a DMT-DS-CDMA system. □

## 9.3 Simulation Results

In this section, we will perform some simulations on the DMT-DS-CDMA system related to system 1 of section 7.10 ($M = 4$, $J = 2$, $N = 2$, $L_1 = 2$, $L_2 = 4$ and $\delta_1 = \delta_2 = 0$). We assume that the data symbol sequences $\{s_j[k]\}_{j=1}^{J}$ are mutually independent and zero-mean white. We further assume that the additive noises $\{e^{(m)}[n]\}_{m=1}^{M}$ are mutually independent and zero-mean white Gaussian with variance $\sigma_e^2$. Finally, we assume that all users have the same received power. The SNR at the input of the receiver is defined as in (7.58). We apply scenario 1 (see section 7.10) and assume that the delay-0 system order is exactly estimated ($\hat{r}_0^{red} = r_0^{red}$). We take $P = 20$ and consider a per tone burst length of $K = 100$. We will consider three values of $\mu$: $\mu = 0$ (no cyclic prefix), $\mu = 2$ and $\mu = 4$. From example 9.2.3, we can then calculate for each of these values of $\mu$ the minimal required value for $Q$. When $\mu = 0$ (we then have $J' = 0$), the minimal required value for $Q$ is 2 (same minimal required value for $Q$ as for system 1 of section 7.10). When $\mu = 2$ (we then have $J' = 1$) or $\mu = 4$ (we then have $J' = 2$), the minimal required value for $Q$ is 0. For $\mu = 0$, we take two values of $Q$: $Q = 2$ (we then have $r_0^{red} = (Q+1)M = 12$) and $Q = 4$ (we then have $r_0^{red} = 12 < (Q+1)M = 20$). For $\mu = 2$ or $\mu = 4$, we take $Q = 0$ (for $\mu = 2$ we then have $r_0^{red} = (Q+1)M = 4$ and for $\mu = 4$ we then have $r_0^{red} = 2 < (Q+1)M = 4$). We only consider method BM1 with constraint C2 and method BM3 with constraint C3. The results are shown in **figure 9.1**. In this figure, we also repeat some results, presented in the figures 7.7 and 7.8, generated for system 1 of section 7.10. Remember that these results are also obtained by method BM1 with constraint C2 and method BM3 with constraint C3, assuming that the system order is exactly estimated ($\hat{r} = r$) and considering a burst length of $K = 100$. First of all, we observe that increasing $Q$ does not have a positive effect on the performance for the DMT-DS-CDMA system taking $\mu = 0$, while increasing $Q$ does improve the performance for the related DS-CDMA system. Increasing $\mu$ and taking the minimal required value for $Q$, on the other hand, does improve the performance for the DMT-DS-CDMA system quite a bit. The performance for the DMT-DS-CDMA system taking $\mu = 4$ and $Q = 0$ is even better than the performance for the related DS-CDMA system taking $Q = 4$ and considering scenario 4.

## 9.4 Conclusions

In this chapter, we have shown how the deterministic blind ST-RAKE receivers based on block processing, presented in chapter 7, can also be applied on a DMT-DS-CDMA system (similar reasonings are possible for the deterministic blind ST-RAKE receivers based on adaptive processing, presented in chapter 8). We have shown that the use of a cyclic prefix reduces the ICI in

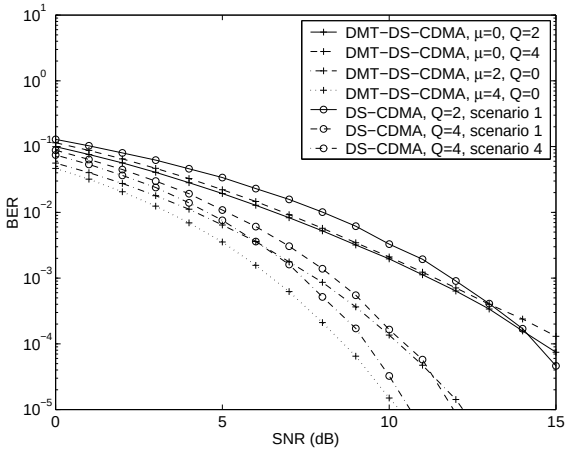



Figure 9.1: Average BER per user as a function of SNR for method BM1 with constraint C2 and method BM3 with constraint C3 applied on a DMT-DS-CDMA system and the related DS-CDMA system (DMT-DS-CDMA: system related to system 1, scenario 1, $\hat{r}_0^{red} = r_0^{red}$, $P = 20$, $K = 100$; DS-CDMA: system 1, $\hat{r} = r$, $K = 100$).

a DMT-DS-CDMA system and that, as a result of this reduction of ICI, *the computational complexity of the block processing methods, presented in chapter 7, can be reduced when they are carried out for a tone of a DMT-DS-CDMA system w.r.t. the case they are carried out for the related DS-CDMA system.* Moreover, simulation results have shown that *this reduction in computational complexity even comes with a performance improvement.*

# Chapter 10

# Conclusions and Further Research

This chapter consists of two sections. In section 10.1, we summarize the main conclusions of this thesis. Section 10.2 gives some suggestions for further research.

## 10.1 Conclusions

For a DS-CDMA and DMT-CDMA system, multi-user receivers achieve high performances. However, they are rather complex. In **part I**, we have therefore developed new alternatives to multi-user receivers, yielding a lower computational complexity and a comparable performance. These alternatives, which are based on the concept of block spreading, have been derived in the context of a single receive antenna and an ICI-limited quasi-synchronous propagation model.

For a DS-CDMA system based on block spreading (DS-CDMA-BS system), we have introduced the MUI-free receiver, which completely removes the MUI, without using any channel information. We have then compared this receiver with the linear multi-user equalizer developed for a DS-CDMA system. It has been shown that the complexity of the MUI-free receiver is much smaller than the complexity of the linear multi-user equalizer, if the receiver design is based on the appropriate subspace deterministic blind channel estimation algorithm. Moreover, the complexity of a complete bank of MUI-free receivers is also much smaller than the complexity of a complete bank of linear multi-user equalizers, although the latter is of the same order as the complexity of

only one linear multi-user equalizer. Next, simulation results have shown that the performance of the MUI-free receiver with ZF respectively MMSE linear block combining is comparable with the performance of the ZF respectively MMSE linear multi-user equalizer. This for the case the channels and the noise variance (necessary for both MMSE receivers) are known as well as for the case the channels are estimated using the appropriate subspace deterministic blind channel estimation algorithm and the noise variance is estimated as a byproduct from the appropriate blind channel gain estimation algorithm. We have also compared the MUI-free receiver with the corresponding FFT-based VL-AMOUR transceiver [WSGB99]. We have shown that the latter uses a transmitter scheme that is more complex in operation than the spreading operation used in a DS-CDMA-BS system and uses a receiver scheme that is more complex in operation than the MUI-free receiver. For the case the channels are known, simulation results illustrate that the performance of the MUI-free receiver with ZF linear block combining is comparable with the performance of the corresponding FFT-based VL-AMOUR transceiver with ZF linear block combining.

For a DMT-CDMA system based on block spreading (DMT-CDMA-BS system), we have introduced the MUI-free receiver, which completely removes the MUI, without using any channel information, and the MUI/ITI-free receiver, which completely removes the MUI and ITI, without using any channel information. We have then compared these receivers with the linear multi-user equalizer developed for a DMT-CDMA system. The complexity of the MUI-free receiver is much smaller than the complexity of the linear multi-user equalizer, if the receiver design is based on the appropriate subspace deterministic blind channel estimation algorithm. Further, note that the MUI/ITI-free receiver is cheaper than the MUI-free receiver, because the use of the linear block combiner is avoided. Next, simulation results have shown that the performance of the MUI-free receiver and the MUI/ITI-free receiver, considering a scenario where the power of every DMT subsymbol sequence is adjusted to the power of the channel at the corresponding frequency (best scenario for these receivers), is respectively *better than* and *comparable to* the performance of the linear multi-user equalizer, considering a scenario where the power of every DMT subsymbol sequence is the same (best scenario for this receiver).

In **part II**, we have given a general description of deterministic blind ST-RAKE receivers for a DS-CDMA system with multiple receive antennas. First, we have developed deterministic blind ST-RAKE receivers based on block processing. We have considered a subspace framework as well as a non-subspace framework. Direct blind symbol estimation methods as well as direct blind equalizer estimation methods have been described, with eye for computational complexity. Furthermore, we have revealed some interesting relations between the proposed methods, on one side, and the SSI methods of [LX97, vdVTP97] and the MRE method of [GDM97], which are all developed for a TDMA system

(no coding), on the other side.

The developed block processing methods are applicable for any modulation format. However, when the data symbols belong to a real constellation, we can adapt these methods to exploit the real character of the data symbols. The methods obtained in this way are referred to as RC methods.

We have also described how the developed block processing methods can be adapted when transmit diversity is used (multiple transmit antennas per user). We then spread the same data symbol sequence with different code sequences and transmit the resulting chip sequences through different antennas. The methods obtained in this way are referred to as TD methods.

Next, we have discussed the case for which the spreading factor $N$ is equal to 1 (code modulation). We have showed that the methods with a polynomial constraint should not be applied for $N = 1$. The RC methods with a polynomial constraint, on the other hand, should also not be applied for $N = 1$, if the code sequences are real, but they can be applied for $N = 1$, if the code sequences are complex. Finally, the TD methods with a polynomial constraint can always be applied for $N = 1$.

Simulation results have shown that the proposed methods render robustness against system order overestimation. Moreover, we have seen that, although channel order information is not really necessary, it can lead to an improved performance. Finally, we have illustrated that, when no ICI is present, the proposed RC or TD methods based on code modulation constitute cheap and reasonably well performing alternatives to existing blind source separation methods that are not based on coding, such as ILSP [TVP96] and RACMA [vdV97].

Then, we have developed some deterministic blind ST-RAKE receivers based on adaptive processing. First, windowed adaptive processing has been discussed. We have considered a subspace framework as well as a non-subspace framework. We have only focused on direct blind symbol estimation. Second, exponentially weighted adaptive processing has been discussed. This type of adaptive processing only makes sense in a non-subspace framework. Like before, we have only focused on direct blind symbol estimation.

We have shown that there is a potential computational advantage of adaptive processing over block processing. Moreover, simulation results have illustrated that the presented adaptive processing methods perform well both in a time invariant and a time varying environment. Note that in this time varying environment the performance of block processing on a large block of received samples is not satisfactory anymore.

Finally, we have shown how the deterministic blind ST-RAKE receivers based on block processing can also be applied on a DMT-DS-CDMA system (similar reasonings are possible for the deterministic blind ST-RAKE receivers based on

adaptive processing). We have shown that the use of a cyclic prefix reduces the ICI in a DMT-DS-CDMA system and that, as a result of this reduction of ICI, the computational complexity of the block processing methods can be reduced when they are carried out for a tone of a DMT-DS-CDMA system w.r.t. the case they are carried out for the related DS-CDMA system. Moreover, simulation results have shown that this reduction in computational complexity even comes with a performance improvement.

## 10.2 Further Research

The new receivers for block spreading systems, developed in part I, are based on the use of a single receive antenna. It would be interesting to investigate the extension to multiple receive antennas, aiming at an increase of the user capacity of the system. Note that this extension is not that straightforward since the number of shift-orthogonal codes we can use in a block spreading system is only determined by the spreading factor and not by the number of receive antennas.

Remember from part I that the MUI-free operation of an MUI-free receiver is obtained by the use of a shift-orthogonal set of code sequences. From such a set of code sequences we can easily derive an orthogonal set of code sequences. At first sight the orthogonal code sequences belonging to such a set allow a more uniform spreading than, for instance, Walsh-Hadamard code sequences, which are also orthogonal. Hence, it can be expected that they are good candidates for asynchronous DS-CDMA communication. The question now is if they are better candidates than, for instance, Gold code sequences.

W-CDMA proposals [OP98] foresee the use of training sequences to detect a user. In this perspective, it would be meaningful to investigate deterministic *semi-blind* ST-RAKE receivers, which would combine elements of the deterministic blind ST-RAKE receivers, developed in part II, with elements of training-based approaches.

Another interesting research topic is the extension of the deterministic blind ST-RAKE receivers based on transmit diversity, which we have discussed in part II, to blind receivers for systems based on other (more optimal) space-time coding techniques, such as the space-time coding technique of Foschini [Fos96] (also known as V-BLAST) and the space-time coding technique of Naguib et al [NTSC97].

# Bibliography


[ADC96] N. Al-Dahir and J. M. Cioffi. Optimum Finite-Length Equalization for Multicarrier Transceivers. *IEEE Transactions on Communications*, 44(1):56–64, January 1996.

[AMCG+97] K. Abed-Meraim, J.-F. Cardoso, A. Y. Gorokhov, P. Loubaton, and E. Moulines. On Subspace Methods for Blind Identification of Single-Input Multiple-Output FIR Systems. *IEEE Transactions on Signal Processing*, 45(1):42–55, January 1997.

[AMML97] K. Abed-Meraim, E. Moulines, and P. Loubaton. Prediction Error Method for Second-Order Blind Identification. *IEEE Transactions on Signal Processing*, 45(3):694–705, March 1997.

[AMR95] K. Anand, G. Mathew, and V. U. Reddy. Blind Separation of Multiple Co-Channel BPSK Signals Arriving at an Antenna Array. *IEEE Signal Processing Letters*, 2(9):176–178, September 1995.

[BA96] S. E. Bensley and B. Aazhang. Subspace-Based Channel Estimation for Code Division Multiple Access Communication Systems. *IEEE Transactions on Communications*, 44(8):1009–1020, August 1996.

[Bin90] J. A. C. Bingham. Multicarrier Modulation for Data Transmission: An Idea Whose Time Has Come. *IEEE Communications Magazine*, 28(5), May 1990.

[Cha82] T. Chan. An Improved Algorithm for Computing the Singular Value Decomposition. *ACM Transactions on Mathematical Software*, 8:72–83, 1982.

[CL97] A. Chevreuil and P. Loubaton. Blind Second-Order Identification of FIR Channels: Forced Cyclostationarity and Structured Subspace Method. *IEEE Signal Processing Letters*, 4(7):204–206, July 1997.

[CT97] H. A. Cirpan and M. K. Tsatsanis. Chip Interleaving in Direct Sequence CDMA Systems. In *Proc. ICASSP*, pages 3877–3880, Munich, Germany, April 1997.

[CZ98] Y.-F. Chen and M. D. Zoltowski. Blind 2-D RAKE Receivers Based on RLS-Type Space-Time Adaptive Filtering for DS-CDMA System. In *Proc. ICASSP*, Seattle, Washington, May 1998.

[DBMS79] J. J. Dongarra, J. R. Bunch, C. B. Moler, and G. W. Stewart. *LINPACK Users' Guide*. SIAM, 1979.

[DH95] A. Duel-Hallen. A Family of Multiuser Decision-Feedback Detectors for Asynchronous Code-Division Multiple-Access Channels. *IEEE Transactions on Communications*, 43(2/3/4):421–434, February/March/April 1995.

[Din97] Z. Ding. Matrix Outer-Product Decomposition Method for Blind Multiple Channel Identification. *IEEE Transactions on Signal Processing*, 45(12):3053–3061, December 1997.

[DS94] V. M. DaSilva and E. S. Sousa. Multicarrier Orthogonal CDMA Signals for Quasi-Synchronous Communication Systems. *IEEE Transactions on Communications*, 12(5):842–852, June 1994.

[FG98] G. J. Foschini and M. J. Gans. On Limits of Wireless Communications in a Fading Environment When Using Multiple Antennas. *Wireless Personal Communications*, 6:311–335, 1998.

[For92] G. D. Forney, Jr. Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference. *IEEE Transactions on Information Theory*, 18(3):363–378, May 1992.

[Fos96] G. J. Foschini. Layered Space-Time Architecture for Wireless Communication in a Fading Environment When Using Multi-Element Antennas. *Bell Labs Technical Journal*, 1(2), Autumn 1996.

[FP93] K. Fazel and L. Papke. On the Performance of Convolutionally Coded CDMA-OFDM for Mobile Communication Systems. In *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 468–472, Yokohama, Japan, September 1993.

[GD97] D. Gesbert and P. Duhamel. Robust Blind Channel Identification and Equalization Based on Multi-Step Predictors. In *Proc. ICASSP*, pages 3621–3624, Munich, Germany, April 1997.

[GDM97] D. Gesbert, P. Duhamel, and S. Mayrargue. On-Line Blind Multichannel Equalization Based on Mutually Referenced Filters. *IEEE Transactions on Signal Processing*, 45(9):2307–2317, September 1997.

[GH97] G. B. Giannakis and S. D. Halford. Asymptotically Optimal Fractionally Spaced Channel Estimation and Performance Analysis. *IEEE Transactions on Signal Processing*, 45(7):1815–1830, July 1997.

[GS96] J. Gunther and A. Swindlehurst. Blind Sequential Symbol Estimation of Co-Channel Finite Alphabet Signals. In *Proc. Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, California, November 1996.

[GSP98] D. Gesbert, J. Sorelius, and A. Paulraj. Blind Multi-User MMSE Detection of CDMA Channels. In *Proc. ICASSP*, Seattle, Washington, May 1998.

[GT99] G. B. Giannakis and C. Tepedelenlioğlu. Direct Blind Equalizers of Multiple FIR Channels: A Deterministic Approach. *IEEE Transactions on Signal Processing*, 47(1):62–74, January 1999.

[GvdVP99] D. Gesbert, A.-J. van der Veen, and A. Paulraj. On the Equivalence of Blind Equalizers Based on MRE and Subspace Intersections. *IEEE Transactions on Signal Processing*, 47(3):856–859, March 1999.

[GVL89] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 1989.

[GWSB99] G. B. Giannakis, Z. Wang, A. Scaglione, and S. Barbarossa. Mutually Orthogonal Transceivers for Blind Uplink CDMA Irrespective of Multipath Channel Nulls. In *Proc. ICASSP*, pages 2431–2434, Phoenix, Arizona, March 1999.

[HAMW97] Y. Hua, K. Abed-Meraim, and M. Wax. Blind System Identification Using Minimum Noise Subspace. *IEEE Transactions on Signal Processing*, 45(3):770–773, March 1997.

[HX97] L. K. Hansen and G. Xu. A Fast Sequential Source Separation Algorithm for Digital Cochannel Signals. *IEEE Signal Processing Letters*, 4(2):58–61, February 1997.

[KB93] A. Klein and P. W. Baier. Linear Unbiased Data Estimation in Mobile Radio Systems Applying CDMA. *IEEE Journal on Selected Areas in Communications*, 11(7):1058–1066, September 1993.

[KKKB96] A. Klein, G. Kawas Kaleh, and P. W. Baier. Zero Forcing and Minimum Mean-Square-Error Equalization for Multiple-Access Channels. *IEEE Transactions on Vehicular Technology*, 45(2):276–287, May 1996.

[KM95] T. Kawahara and T. Matsumoto. Joint Decorrelating Multiuser Detection and Channel Estimation in Asynchronous CDMA Mobile Communications Channels. *IEEE Transactions on Vehicular Technology*, 44(3):506–515, August 1995.

[KOS96] M. Kristensson, B. Ottersten, and D. Slock. Blind Subspace Identification of a BPSK Communication Channel. In *Proc. Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, California, November 1996.

[LCC95] I. Lee, J. S. Chow, and J. M. Cioffi. Performance Evaluation of a Fast Computation Algorithm for the DMT in High-Speed Subscriber Loop. *IEEE Journal on Selected Areas in Communications*, 13(9):1564–1570, December 1995.

[LG99] G. Leus and D. Gesbert. Recursive Blind Source Separation for BPSK Signals. In *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 267–270, Annapolis, Maryand, May 1999.

[LLV93] F. Li, H. Liu, and R. J. Vaccaro. Performance Analysis for DOA Estimation Algorithms: Unification, Simplification, and Observations. *IEEE Transactions on Aerospace and Electronic Systems*, 29(4):1170–1184, October 1993.

[LM97a] G. Leus and M. Moonen. Adaptive Blind Equalization for Synchronous DS-CDMA Systems. In *Proc. COST 254 Workshop on Emerging Techniques for Communication Terminals*, pages 373–377, Toulouse, France, July 1997.

[LM97b] G. Leus and M. Moonen. An Adaptive Blind Receiver for Asynchronous DS-CDMA Based on Recursive SVD and RLS. In *Proc. ProRISC/IEEE Workshop on Circuits, Systems and Signal Processing*, pages 615–620, Mierlo, The Netherlands, November 1997.

[LM97c] G. Leus and M. Moonen. An Adaptive Blind Receiver for Asynchronous DS-CDMA Based on Recursive SVD and Viterbi Decoding. In *Proc. IEEE Benelux Symposium on Communications and Vehicular Technology*, pages 40–47, Enschede, The Netherlands, October 1997.

[LM98a] G. Leus and M. Moonen. Adaptive Blind Equalization for Asynchronous DS-CDMA Systems Based on RLS. In *Proc. EUSIPCO*, pages 765–768, Rhodes, Greece, September 1998.

[LM98b] G. Leus and M. Moonen. Fully Decorrelating Blind Equalization for Synchronous DS-CDMA Systems. In *Proc. IEEE Digital Signal Processing Workshop*, Bryce Canyon, Utah, August 1998.

[LM99a] G. Leus and M. Moonen. Block Spreading for Discrete Multi-Tone CDMA Systems in the Presence of Frequency-Selective Fading. In *Proc. Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, California, October 1999.

[LM99b] G. Leus and M. Moonen. MUI-Free Receiver for a Synchronous DS-CDMA System Based on Block Spreading in the Presence of Frequency-Selective Fading. *IEEE Transactions on Signal Processing*, 1999. Accepted for publication.

[LM00a] G. Leus and M. Moonen. MUI-Free Receiver for a Shift-Orthogonal Quasi-Synchronous DS-CDMA System Based on Block Spreading in Frequency-Selective Fading. In *Proc. ICASSP*, Istanbul, Turkey, June 2000.

[LM00b] G. Leus and M. Moonen. Viterbi and RLS Decoding for Deterministic Blind Symbol Estimation in DS-CDMA Wireless Communication. *Signal Processing*, 8(5):745–771, May 2000.

[LV89] R. Lupas and S. Verdú. Linear Multiuser Detectors for Synchronous Code-Division Multiple-Access Channels. *IEEE Transactions on Information Theory*, 35(1):123–136, January 1989.

[LV90] R. Lupas and S. Verdú. Near-Far Resistance of Multiuser Detectors in Asynchronous Channels. *IEEE Transactions on Communications*, 38(4):496–508, April 1990.

[LVM99] G. Leus, P. Vandaele, and M. Moonen. Deterministic Blind Modulation-Induced Source Separation for Digital Wireless Communications. *IEEE Transactions on Signal Processing*, 1999. Submitted for publication.

[LVM00] G. Leus, P. Vandaele, and M. Moonen. Per Tone Blind Signal Separation for a DMT-DS-CDMA System. In *Proc. EUSIPCO*, Tampere, Finland, September 2000. Accepted for publication.

[LX96a] H. Liu and G. Xu. Blind Equalization for CDMA Systems with Aperiodic Spreading Sequences. In *Proc. ICASSP*, Atlanta, Georgia, May 1996.

[LX96b] H. Liu and G. Xu. A Subspace Method for Signature Waveform Estimation in Synchronous CDMA Systems. *IEEE Transactions on Communications*, 44(10):1346–1354, October 1996.

[LX97] H. Liu and G. Xu. Smart Antennas in Wireless Systems: Uplink Multiuser Blind Channel and Sequence Detection. *IEEE Transactions on Communications*, 45(2):187–199, February 1997.

[LZ97] H. Liu and M. D. Zoltowski. Blind Equalization in Antenna Array CDMA Systems. *IEEE Transactions on Signal Processing*, 45(1):161–172, January 1997.

[MDCM95] E. Moulines, P. Duhamel, J.-F. Cardoso, and S. Mayrargue. Subspace Methods for the Blind Identification of Multichannel FIR Filters. *IEEE Transactions on Signal Processing*, 43(2):516–525, February 1995.

[MDPM95] M. Moonen, E. Deprettere, I. K. Proudler, and J. G. McWhirter. On The Derivation of Parallel Filter Structures for Adaptive Eigenvalue and Singular Value Decompositions. In *Proc. ICASSP*, pages 3247–3250, Detroit, Michigan, May 1995.

[MH94] U. Madhow and M. L. Honig. MMSE Interference Suppression for Direct-Sequence Spread-Spectrum CDMA. *IEEE Transactions on Communications*, 42(12):3178–3188, December 1994.

[Mil95] S. L. Miller. An Adaptive Direct-Sequence Code-Division Multiple-Access Receiver for Multiuser Interference Rejection. *IEEE Transactions on Communications*, 43(2/3/4):1746–1755, February/March/April 1995.

[MP92] M. Mouly and M.-B. Pautet. *The GSM System for Mobile Communications*. Palaiseau, 1992.

[MYR96] P. J. W. Melsa, R. C. Younce, and C. E. Rohrs. Impulse Response Shortening for Discrete Multitone Transceivers. *IEEE Transactions on Communications*, 44(12):1662–1672, December 1996.

[NPK94] A. F. Naguib, A. Paulraj, and T. Kailath. Capacity Improvement with Base-Station Antenna Arrays in Cellular CDMA. *IEEE Transactions on Vehicular Technology*, 43(3):691–698, August 1994.

[NTSC97] A. F. Naguib, V. Tarokh, N. Seshadri, and A. R. Calderbank. Space-Time Coded Modulation for High Data Rate Wireless Communications. In *Proc. GLOBECOM*, pages 102–109, Phoenix, Arizona, November 1997.

[OP98] T. Ojanperä and R. Prasad. *Wideband CDMA for Third Generation Mobile Communications*. Artech House, 1998.

[PG58] R. Price and P. E. Green, Jr. A Communication Technique for Multipath Channels. *Proc. IRE*, 46:555–570, March 1958.

[PH94] P. Patel and J. Holtzman. Analysis of a Simple Successive Interference Cancellation Scheme in a DS-CDMA System. *IEEE Journal on Selected Areas in Communications*, 12(5):796–807, June 1994.

[Pro89] J. G. Proakis. *Digital Communications*. McGraw-Hill, second edition, 1989.

[RZ97] J. Ramos and M. D. Zoltowski. Blind Space-Time Processor for SINR Maximization in CDMA Cellular Systems. In *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 273–276, Paris, France, April 1997.

[SBG99] A. Scaglione, S. Barbarossa, and G. B. Giannakis. Fading-Resistant and MUI-Free Codes for CDMA Systems. In *Proc. ICASSP*, pages 2290–2293, Phoenix, Arizona, March 1999.

[SD99] J. Shen and Z. Ding. Blind MMSE CDMA Detection in Multipath Channels. In *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Annapolis, Maryland, May 1999.

[SG98] E. Serpedin and G. B. Giannakis. Blind Channel Identification and Equalization with Modulation-Induced Cyclostationarity. *IEEE Transactions on Signal Processing*, 46(7):1930–1944, July 1998.

[SG99] A. Scaglione and G. B. Giannakis. Design of User Codes in QS-CDMA Systems for MUI Elimination in Unknown Multipath. *IEEE Communications Letters*, 3(2):25–27, February 1999.

[SNXP93] B Suard, A. F. Naguib, G. Xu, and A. Paulraj. Performance Analysis of CDMA Mobile Communication Systems Using Antenna Arrays. In *Proc. ICASSP*, volume IV, pages 153–156, Minneapolis, Minnesota, April 1993.

[TG96] M. K. Tsatsanis and G. B. Giannakis. Optimal Decorrelating Receivers for DS-CDMA Systems: A Signal Processing Framework. *IEEE Transactions on Signal Processing*, 44(12):3044–3055, December 1996.

[TG97] M. K. Tsatsanis and G. B. Giannakis. Blind Estimation of Direct Sequence Spread Spectrum Signals in Multipath. *IEEE Transactions on Signal Processing*, 45(5):1241–1252, May 1997.

[TP97] S. Talwar and A. Paulraj. Blind Separation of Synchronous Co-Channel Digital Signals Using an Antenna Array - Part II: Performance Analysis. *IEEE Transactions on Signal Processing*, 45(3):706–718, March 1997.

[Tsa97] M. K. Tsatsanis. Inverse Filtering for CDMA Systems. *IEEE Transactions on Signal Processing*, 45(1):102–112, January 1997.

[TVP96] S. Talwar, M. Viberg, and A. Paulraj. Blind Separation of Synchronous Co-Channel Digital Signals Using an Antenna Array - Part I: Algorithms. *IEEE Transactions on Signal Processing*, 44(5):1184–1197, May 1996.

[TX97] M. Torlak and G. Xu. Blind Multiuser Channel Estimation in Asynchronous CDMA Systems. *IEEE Transactions on Signal Processing*, 45(1):137–147, January 1997.

[TX98] M. K. Tsatsanis and Z. Xu. Performance Analysis of Minimum Variance Receivers. *IEEE Transactions on Signal Processing*, 46(11):3014–3022, November 1998.

[TXK94] L. Tong, G. Xu, and T. Kailath. Blind Identification and Equalization Based on Second-Order Statistics: A Time Domain Approach. *IEEE Transactions on Information Theory*, 40(2):340–349, March 1994.

[Ung76] G. Ungerboeck. Fractional Tap-Spacing Equalizer and Consequences for Clock Recovery in Data Modems. *IEEE Transactions on Communications*, 24(8):856–864, August 1976.

[VA90] M. K. Varanasi and B. Aazhang. Multistage Detection in Asynchronous Code-Division Multiple-Access Communications. *IEEE Transactions on Communications*, 38(4):509–519, April 1990.

[VALM+99] K. Van Acker, G. Leus, M. Moonen, O. van de Wiel, and T. Pollet. Per Tone Equalization for DMT Receivers. In *Proc. GLOBECOM*, pages 2311–2315, Rio de Janeiro, Brazil, December 1999.

[VALM+00] K. Van Acker, G. Leus, M. Moonen, O. van de Wiel, and T. Pollet. DMT-Based Systems with Per Tone Equalization. *IEEE Transactions on Communications*, 2000. Accepted for publication.

[VALMP99] K. Van Acker, G. Leus, M. Moonen, and T. Pollet. Frequency Domain Equalization with Tone Grouping in DMT/ADSL-Receivers. In *Proc. Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, California, October 1999.

[VALMP00] K. Van Acker, G. Leus, M. Moonen, and T. Pollet. RLS-Based Initialization for Per Tone Equalizers in DMT-Receivers. In *Proc. EUSIPCO*, Tampere, Finland, September 2000. Accepted for publication.

[Van95] L. Vandendorpe. Multitone Spread Spectrum Multiple Access Communications System in a Multipath Rician Fading Channel. *IEEE Transactions on Vehicular Technology*, 44(2):327–337, May 1995.

[Van99] P. Vandaele. *Space-Time Processing Algorithms for Smart Antennas in Wireless Communication Networks*. PhD thesis, Faculty of Applied Sciences, K.U.Leuven (Leuven, Belgium), November 1999.

[VAPLM99] K. Van Acker, T. Pollet, G. Leus, and M. Moonen. Combination of Per Tone Equalization and Windowing in DMT-Receivers. In *Proc. Bayona Workshop on Emerging Technologies in Telecommunications*, pages 49–53, Bayona, Spain, September 1999.

[VBM95] M. Van Bladel and M. Moeneclaey. Time-Domain Equalization for Multicarrier Communication. In *Proc. GLOBECOM*, pages 167–171, 1995.

[vdV97] A.-J. van der Veen. Analytical Method for Blind Binary Signal Separation. *IEEE Transactions on Signal Processing*, 45(4):1078–1082, April 1997.

[vdVP96] A.-J. van der Veen and A. Paulraj. An Analytical Constant Modulus Algorithm. *IEEE Transactions on Signal Processing*, 44(5):1136–1155, May 1996.

[vdVTP97] A.-J. van der Veen, S. Talwar, and A. Paulraj. A Subspace Approach to Blind Space-Time Signal Processing for Wireless Communication Systems. *IEEE Transactions on Signal Processing*, 45(1):173–190, January 1997.

[Ver86] S. Verdú. Minimum Probability of Error for Asynchronous Gaussian Multiple-Access Channels. *IEEE Transactions on Information Theory*, 32(1):85–96, January 1986.

[VHV91] S. Van Huffel and J. Vandewalle. *The Total Least Squares Problem, Computational Aspects and Analysis*. SIAM, 1991.

[Vit95] A. J. Viterbi. *CDMA: Principles of Spread Spectrum Communication*. Addison-Wesley, 1995.

[VKS96] J. F. Van Kerckhove and P. Spruyt. Adapted Optimization Criterion for FDM-Based DMT-ADSL Equalization. In *Proc. ICC*, pages 1328–1334, 1996.

[VLM99a] P. Vandaele, G. Leus, and M. Moonen. A Non-Iterative Blind Binary Signal Separation Algorithm Based on Linear Coding. In *Proc. IEEE Workshop on Signal Processing Advances in Wireless*

*Communications (SPAWC)*, pages 98–101, Annapolis, Maryand, May 1999.

[VLM99b] P. Vandaele, G. Leus, and M. Moonen. A Non-Iterative Blind Signal Separation Algorithm Based on Transmit Diversity and Coding. In *Proc. Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, California, October 1999.

[VM96] P. Vandaele and M. Moonen. An 'SVD+Viterbi' Algorithm for Adaptive Blind Equalization of Mobile Radio Channel. In *Proc. Bayona Workshop on Intelligent Methods for Signal Processing and Communications*, pages 159–163, Bayona, Spain, June 1996.

[VM98a] P. Vandaele and M. Moonen. Adaptive Algorithms for Single User Blind Channel Equalization. *Mathematics in Signal Processing IV (J. G. McWhirter and I. K. Proudler, eds.)*, pages 79–94, 1998.

[VM98b] P. Vandaele and M. Moonen. An 'SVD+Viterbi' Algorithm for Multi-User Adaptive Blind Equalization of Mobile Radio Channel. *IEEE Signal Processing Letters*, pages 260–264, October 1998.

[VTEDM99] P. Vandenameele, S. Thoen, M. Engels, and H. De Man. A Combined OFDM/SDMA Approach for WLAN. In *Proc. IEEE Vehicular Technology Conference (VTC)*, pages 1712–1716, Houston, Texas, May 1999.

[WLLZ98] T. F. Wong, T. M. Lok, J. S. Lehnert, and M. D. Zoltowski. A Linear Receiver for Direct-Sequence Spread-Spectrum Multiple-Access Systems with Antenna Arrays and Blind Adaptation. *IEEE Transactions on Information Theory*, 44(2):659–676, March 1998.

[WP98] X. Wang and H. V. Poor. Blind Equalization and Multiuser Detection in Dispersive CDMA Channels. *IEEE Transactions on Communications*, 46(1):91–103, January 1998.

[WSGB99] Z. Wang, A. Scaglione, G. B. Giannakis, and S. Barbarossa. Vandermonde-Lagrange Mutually Orthogonal Flexible Transceivers for Blind CDMA in Unknown Multipath. In *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 42–45, Annapolis, Maryland, May 1999.

[XBM98] J. Xavier, V. Barroso, and J. M. F. Moura. Closed-Form Blind Identification of MIMO Channels. In *Proc. ICASSP*, Seattle, Washington, May 1998.

[XSR90] Z. Xie, R. T. Short, and C. K. Rushforth. A Family of Suboptimum Detectors for Coherent Multiuser Communications. *IEEE Journal on Selected Areas in Communications*, 8(4):683–690, May 1990.

[XT98] Z. Xu and M. K. Tsatsanis. Adaptive Minimum Variance Methods for Direct Blind Multichannel Equalization. In *Proc. ICASSP*, Seattle, Washington, May 1998.

[XT99] Z. Xu and M. K. Tsatsanis. On Correlation Matching Approaches to Multipath Parameter Estimation in Multiuser CDMA Systems. In *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Annapolis, Maryland, May 1999.

[YLF93] N. Yee, J.-P. Linnartz, and G. Fettweis. Multi-Carrier CDMA in Indoor Wireless Radio Networks. In *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 109–113, Yokohama, Japan, September 1993.

[ZB95] Z. Zvonar and D. Brady. Suboptimal Multiuser Detector for Frequency-Selective Rayleigh Fading Synchronous CDMA Channels. *IEEE Transactions on Communications*, 43(2/3/4):154–157, February/March/April 1995.

[ZB96] Z. Zvonar and D. Brady. Linear Multipath-Decorrelating Receivers for CDMA Frequency-Selective Fading Channels. *IEEE Transactions on Communications*, 44(6):650–653, June 1996.

[ZT97a] H. H. Zeng and L. Tong. Blind Channel Estimation Using the Second-Order Statistics: Algorithms. *IEEE Transactions on Signal Processing*, 45(8):1919–1930, August 1997.

[ZT97b] H. H. Zeng and L. Tong. Blind Channel Estimation Using the Second-Order Statistics: Asymptotic Performance and Limitations. *IEEE Transactions on Signal Processing*, 45(8):2061–2071, August 1997.

[Zvo96] Z. Zvonar. Combined Multiuser Detection and Diversity Reception for Wireless CDMA Systems. *IEEE Transactions on Vehicular Technology*, 45(1):205–211, February 1996.