A COMPRESSED SENSING APPROACH TO BLOCK-ITERATIVE EQUALIZATION: CONNECTIONS AND APPLICATIONS TO RADAR IMAGING RECONSTRUCTION

Rafael Gustavo da Cunha Pereira Pinto

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Elétrica.

Orientador: Ricardo Merched

Rio de Janeiro
Março de 2017

A COMPRESSED SENSING APPROACH TO BLOCK-ITERATIVE EQUALIZATION: CONNECTIONS AND APPLICATIONS TO RADAR IMAGING RECONSTRUCTION

Rafael Gustavo da Cunha Pereira Pinto

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

Prof. Ricardo Merched, Ph.D.

Prof. Fernando Gil Vianna Resende Junior, Ph.D.

Prof. Mariane Rembold Petraglia, Ph.D.

Prof. Vítor Heloiz Nascimento, Ph.D.

Prof. Antônio Petraglia, Ph.D.

RIO DE JANEIRO, RJ – BRASIL
MARÇO DE 2017

Pinto, Rafael Gustavo da Cunha Pereira

A Compressed Sensing Approach to Block-Iterative Equalization: Connections and Applications to Radar Imaging Reconstruction/Rafael Gustavo da Cunha Pereira Pinto. – Rio de Janeiro: UFRJ/COPPE, 2017.

XIV, 149 p.: il.; 29, 7cm.

Orientador: Ricardo Merched

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2017.

Referências Bibliográficas: p. 138 – 147.

1. Compressed sensing. 2. Equalização. I. Merched, Ricardo. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

To Paty, my wife, who gave me support throughout this journey.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

UMA ABORDAGEM PARA EQUALIZAÇÃO ITERATIVA EM BLOCOS VIA SENSORIAMENTO COMPRESSIVO: CONEXÕES E APLICAÇÕES EM RECONSTRUÇÃO DE IMAGENS DE RADAR

Rafael Gustavo da Cunha Pereira Pinto

Março/2017

Orientador: Ricardo Merched

Programa: Engenharia Elétrica

A proliferação de sistemas sub-determinados trouxe a tona uma gama de novas soluções algorítmicas, baseadas no sensoriamento compressivo (CS) de dados esparsos. As recursões do tipo *greedy* e de limitação iterativa para CS se apresentam comumente como um filtro adaptativo seguido de um operador proximal, não muito diferente dos equalizadores de realimentação de decisão iterativos em blocos (BI-DFE), em que um decisor explora a estrutura do sinal de constelação.

A partir da esparsidade intrínseca presente na modulação de sinais no contexto de comunicações, a interferência entre blocos (IBI) pode ser abordada utilizando-se o conceito de CS, onde a realimentação ótima de símbolos detectados é realizada de forma adaptativa. O novo DFE se apresenta como um esquema mais eficiente de reestimação, baseado na atualização por mínimos quadrados recursivos (RLS). Sempre que possível, estas recursões são propostas via formulação linear no sentido amplo, o que reduz ainda mais o erro médio quadrático mínimo (MMSE) em comparação com abordagens tradicionais. Além de maximizar a taxa de transferência de informação, o novo algoritmo exibe um desempenho significativamente superior quando comparado aos métodos existentes.

Também mostraremos que um equalizador BI-DFE formulado adequadamente se torna um poderoso algoritmo de CS. O novo algoritmo CS-BDFE apresenta convergência e detecção aprimoradas, quando comparado a métodos de primeira ordem, superando as recursões de Passagem de Mensagem Aproximada para Complexos (CAMP). Os méritos das novas recursões são ilustrados através de um modelo tridimensional para radares MIMO recentemente proposto, onde o algoritmo CAMP falha em aspectos importantes de medidas de desempenho.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

A COMPRESSED SENSING APPROACH TO BLOCK-ITERATIVE EQUALIZATION: CONNECTIONS AND APPLICATIONS TO RADAR IMAGING RECONSTRUCTION

Rafael Gustavo da Cunha Pereira Pinto

March/2017

Advisor: Ricardo Merched

Department: Electrical Engineering

The widespread of underdetermined systems has brought forth a variety of new algorithmic solutions, which capitalize on the Compressed Sensing (CS) of sparse data. While well known greedy or iterative threshold type of CS recursions take the form of an adaptive filter followed by a proximal operator, this is no different in spirit from the role of block iterative decision-feedback equalizers (BI-DFE), where structure is roughly exploited by the signal constellation slicer.

By taking advantage of the intrinsic sparsity of signal modulations in a communications scenario, the concept of interblock interference (IBI) can be approached more cunningly in light of CS concepts, whereby the optimal feedback of detected symbols is devised adaptively. The new DFE takes the form of a more efficient re-estimation scheme, proposed under recursive-least-squares based adaptations. Whenever suitable, these recursions are derived under a reduced-complexity, widely-linear formulation, which further reduces the minimum-mean-square-error (MMSE) in comparison with traditional strictly-linear approaches. Besides maximizing system throughput, the new algorithms exhibit significantly higher performance when compared to existing methods.

Our reasoning will also show that a properly formulated BI-DFE turns out to be a powerful CS algorithm itself. A new algorithm, referred to as CS-Block DFE (CS-BDFE) exhibits improved convergence and detection when compared to first order methods, thus outperforming the state-of-the-art Complex Approximate Message Passing (CAMP) recursions. The merits of the new recursions are illustrated under a novel 3D MIMO Radar formulation, where the CAMP algorithm is shown to fail with respect to important performance measures.

Contents

List of Figures

List of Tables

Notation

Throughout the text, we are going to use a few typographic notations, listed below:

x	lower-case letter denotes a scalar
z^*	complex conjugation of a complex number z
$\Re(z)$	the real part of a complex number z
$\Im(z)$	the imaginary part of a complex number z
$\lvert x\rvert$	absolute value of x
$\boldsymbol{x}, \mathbf{x}, \mathfrak{x}$	lower-case boldface letter denotes a vector
$x(i)$	i-th element of a vector
$\widehat{\boldsymbol{x}}$	an unit vector in the same direction as $\boldsymbol{x}$
$\hat{\boldsymbol{x}}$	an estimate of $\boldsymbol{x}$
$\boldsymbol{A}$, $\boldsymbol{\mathcal{A}}, \boldsymbol{\mathscr{A}}$	upper-case boldface and calligraphic letters denote matrices
$\boldsymbol{I}_N$	$N \times N$ identity matrix
$\mathbf{1}_{N\times M}$	$N \times M$ all-ones matrix
$\mathbf{0}_{N\times M}$	$N \times M$ null matrix
$\boldsymbol{A}^T$	matrix transposition
$\boldsymbol{A}^*$	matrix Hermitian transposition
$[\boldsymbol{A}]_i$	i-th column of a matrix
$[\boldsymbol{A}]_{i,j}$	the element at row i, column j
$[\boldsymbol{A}]_{i,:}$	i-th row of a matrix
$\lVert\boldsymbol{x}\rVert_p$	p-norm of vector $\boldsymbol{x}$
$\lVert\boldsymbol{x}\rVert$	Euclidean norm of vector $\boldsymbol{x}$
$\lVert\boldsymbol{A}\rVert_F$	Fröbenius norm of a matrix
$\lVert\boldsymbol{A}\rVert_p$	induced p-norm of a matrix
$\lVert\boldsymbol{A}\rVert, \lVert\boldsymbol{A}\rVert_2$	induced 2-norm of a matrix
$\lambda_{\max}(\boldsymbol{A})$	largest magnitude eingenvalue of a matrix
$\otimes$	Kronecker product
$\odot$	Hadamard product
$\mathrm{vec}(\cdot)$	column stacking operator
$\mathrm{null}(\cdot)$	null space of a matrix

$\circledast$	convolution operator
$\frac{\partial}{\partial x}$	partial derivative with respect to variable x
$\frac{\partial^2}{\partial^2 x}$	second-order partial derivative with respect to variable x
$f'(x)$	first-order derivative of a function of one variable
$f''(x)$	second-order derivative of a function of one variable

In general, upper-case letters denote integer quantities. Dimensions for the identity matrix $\boldsymbol{I}$, the all-ones matrix $\mathbf{1}$ and the null matrix $\mathbf{0}$ can be omitted whenever possible and inferred from the implied dimensions.

Chapter 1

Introduction

Compressed sensing (CS) has been the key to resolving the underdeterminacy deep-rooted in a variety of practical systems featuring few available measurements [1–3]. Generally speaking, the solution to these systems when restricted to the smallest cardinality, represented as the ℓ_0-norm, is NP-complex. In case this restriction is relaxed to the smallest ℓ_1-norm, even though obtaining the solution can be easily expressed as a linear programming, the ongoing search for iterative algorithms that avoid the computational burden of interior point methods is paramount[4–10]. When tackled iteratively, the resulting CS recursions take the form of a block adaptive algorithm equipped with a generic (soft/hard) thresholding function, thus reflecting some *a priori* knowledge of sparsity in the unknown parameter. This is by no means different in spirit from the role of a decision device commonly deployed by *block iterative decision-feedback equalizers* (BI-DFE) [11, 12], where structure is translated to a projection, e.g., of a *least-squares* (LS) or *minimum-mean square-error* (MMSE) estimate, onto the transmitted signal constellation. Such type of iterative DFE has been proposed, for example, in [11], under *signal-to-interference-plus-noise ratio* (SINR) maximization, as a re-estimation procedure for the transmitted block given prior decisions. A similar approach has been further considered in [12] under MMSE or a *zero-forcing* (ZF) based cost function, in the so-called reduced-redundancy block DFE transceivers.

Despite the similarities between CS and BI-DFE based recursions, these theories have evolved rather independently, as different recipes working in distinct scenarios. In this respect, several subtleties arise when comparing the DFE functioning in a communications setting, and a CS algorithm generally designed for sparsity recovery. For instance, the block DFE algorithms considered in [11, 12] are similar in nature to the widely known *greedy* and *iterative soft/hard thresholding* (IST and IHT) type recursions (to be reviewed shortly) commonly referenced in the CS literature, albeit ones with a few remarkable differences:

(**1**) First, in a BI-DFE, the signal slicer acts on each entry of the MMSE estimated vector as a projection operator onto the closest point of the signal constellation, while in a CS algorithm, each entry is projected onto the origin according to a soft or hard thresholding rule. Second, in block transmission transceivers, the role of the sensing matrix is frequently assumed by a 'fat' matrix possessing a Toeplitz structure, as a result of the convolutional model inherent to *linear time-invariant* (LTI) channels. In order to cope with the intrinsic underdeterminacy of this problem, the effective transmission matrix can be modified by introducing some form of redundancy that accounts for *interblock interference* (IBI), and/or by the use of past decisions so as to remove IBI — be it in a linear equalization, or in a block memoryless transmission fashion.

(**2**) In a general use CS setting, on the other hand, while the sensing matrix still represents an underdetermined problem, it may not be restricted to Toeplitz structures, and sparsity allows for data recovery up to a certain extent. The accuracy in the estimation under additive noise is commonly dictated by the so-called *restricted isometry property* (RIP) of the sensing matrix [13], which is quantified by the noise level, and the presumed sparsity in the target vector. This is further related to a *coherence* measure [1] of the sensing matrix, which is simpler to calculate, and yields useful bounds on different norms of the estimation error.

The above nuances raise several intuitive questions and suggest that a closer look into the connections between these two approaches can lead us to improved performances in each setting individually. For example, what is the exact relation between the classes of IST/IHT types of compressed sensing algorithms, which aim sparse recovery from a CS perspective, and the long studied structures of iterative DFE schemes towards constellation detection in digital communications? Is there a common line in the derivation of both techniques which can enlighten us with more efficient and accurate schemes beneficial for each scenario?

The kickoff of this presentation is to highlight that while these concepts have developed apparently unrelated, both originate from the same exact problem formulation and can be developed into more efficient techniques to recover structured signals in both communications and generic sparse settings. We capitalize on the interplays between the CS and the BI-DFE formulations in order to derive new algorithms applied to equalization and compressed sensing problems.

From these connections, we approach the CS problem from a LS perspective by employing a suitable regularization function, in addition to the well known regularized *recursive least-squares* (RLS) problem formulation [14]. That is, in a communications setting, for a complex vector $\mathbf{x}$ with entries belonging to a signal

constellation, and given its a priori estimate, say, $\mathbf{x}_{i-1}$, any centered vector $(\mathbf{x}-\mathbf{x}_{i-1})$ is potentially a sparse vector. This suggests an ℓ_0 or ℓ_1 type norm regularizer on $(\mathbf{x}-\mathbf{x}_{i-1})$, as a means to reflecting sparsity adaptively into the BI-DFE formulation. The iterative solution to such problem will naturally yield a BI-DFE without the explicit optimization of a feedback matrix as it has been the case in DFE derivations. The new algorithm can be obtained exactly as Iterative-Shrinking recursions, and because these are essentially RLS type algorithms, they will generally include an inverse Hessian matrix term at each algorithm step. The proposed CS algorithms are specially useful in the case of long transmitted blocks of data, where complexity becomes higher and sparsity exploitation assumes a crucial role.

A significant implication of this observation for block transmissions is that, while traditional techniques commonly deploy some form of redundancy to account for IBI cancellation [12, 15, 16], a CS technique applied to the transmitted vector, if successful, would require no redundancy whatsoever, and is capable of detecting both the symbol and its interference altogether. This is to be contrasted with systems that perform block-by-block detection and employ conventional DFEs (with memory), or block memoryless DFEs employing minimum, or even zero-redundancy, in order to eliminate IBI. In fact, for a channel of length L, the concept of *"minimum redundancy"* transmissions [17], which have been fairly mentioned in the communications literature in the context of ZF/MMSE equalizers, makes use of $\delta = \lceil (L-1)/2 \rceil$ zeros appended to the input vector, and will turn out to be not really minimum, according to our receiver design. That is, our interpretation of sparsity embedded into the transmitted vector will allow us to recover it exactly, and with higher throughput than the one assumed by a zero-redundancy scheme. As a fallout, a procedure that attempts to retrieve $\mathbf{x}$, if successful, will imply that not only redundancy in transmission is unnecessary, but also that the received signal can be sampled at a lower rate than what is predicted by Nyquist theory. Although this is the very essence of CS, here we show how this is particularly applied to signal constellations in digital communications. Moreover, a compressed sensing approach devised to detect constellation points will show further improvement in performance when compared to conventional standard MMSE/ZF-based equalizers that employs the same level of (low) redundancy, and inevitably suffers from ill-conditioning.

Conversely, we make use of the rationale behind the derivation of BI-DFE algorithms in order to tackle difficult sensing problems, not commonly solved by known CS techniques. To this end, we derive a new *Compressed Sensing Block DFE* (CS-BDFE) based algorithm, intended (but not limited) to generally ill-posed scenarios. Unlike in the DFE setup, where the target vector exhibits a constellation signal nature, in a pure CS application, structure in the data is solely characterized by vector sparsity, so that current CS algorithms may still not reach the desired

performance for highly coherent dictionaries. For example, in RADAR applications, the underlying sensing matrix exhibits a block Toeplitz structure, which can become very ill-conditioned [18–22]. The unifying approach pursued here will suggest that a block equalization algorithm, when properly adjusted to detect 'zeros' rather than a constellation signal, can outperform the state-of-the-art of CS algorithms found in the literature for the same purpose.

Overall, while most CS iterative algorithms consist of least-mean-squares (LMS) type updates [1–3], we pursue instead a more computationally demanding approach towards a solution based on second-order methods — albeit one provided with accelerated convergence, as well as improved accuracy in terms of final estimate and target support. Despite the additional complexity, the proposed method outperforms state-of-the-art of CS algorithms based on the concept of message passing, which is used in this work as a basis for comparison, under meaningful performance measures. The famed *Complex Approximate Message Passing* (CAMP) algorithm, which derives from *belief propagation* concepts and studied in [10], constitutes an important class of CS, *iterative shrinking* recursions, which is well suited for sparse scenarios when neither the noise variance nor the sparsity levels of the target are known *a priori* [23].

Considering the above, the specific contributions of this work are the following:

1. We bring attention to the fact that a properly formulated BI-DFE can be seen as a CS algorithm obtained in its own right, and not as a pre-imposed structure optimized via some criterion. While traditional formulations of (block) DFEs commonly begin with the optimization of feedforward and feedback matrices in a stochastic sense [11, 12, 24], such architecture in fact arises naturally from a suitably regularized LS problem, and is therefore optimal in this sense. This interpretation brings us an important advantage w.r.t. the most common forms of CS (LMS-based) algorithms, from a BI-DFE perspective: The Ricatti variable propagated through the Kalman filter exempts us from most concerns that arise in a general CS scenario, where accuracy is highly dependent on the sensing matrix coherence. This is analogous to the role of a RLS adaptive algorithm in the estimation of correlated regressors. Moreover, unlike in the original BI-DFE[11], where uncertainty throughout the iterations can be a pre-computed complex phase, a CS based formulation allows for an adaptive calculation of uncertainty in detection, which is available from the error covariance before slicing. We pursue the new algorithms under different policies of assigning uncertainty to the detected entries. Each one exploits a different strategy for symbol feedback, exhibiting their own complexity and performance advantages;

2. Whenever suited, we further derive a *reduced-complexity* (RC) *widely-linear* (WL) [25] version of the BI-DFE, which does not require the traditional circularity assumption on the involved target and noise vectors. The advantage of this approach is two-fold: To improve the performance of current iterative DFE equalizers in a communications setting for non-circular models or improper signals, and to provide alternative efficient CS algorithms that are complex in nature. For example, in the case of *binary phase-shifting keying* (BPSK) signals, the performance of zero redundancy equalizers can be further improved compared to their *strictly-linear* (SL) counterparts, and becomes even more prominent for redundant transmissions. The RC approach has been recently shown to be equivalent to the original WL formulation [26], with order of complexity approximately equal to the SL one, and for this reason becomes highly motivated;

3. For a greedy approach, we show that a special case of the *vertical Bell Labs Layered space time* (V-BLAST) algorithm [27] employing sequential detection of entries can make use of a *fast transversal filter* (FTF) to compute the DFE matrices, henceforth providing a new alternative to the complexity issue implied by the Cholesky factorization inherent to its solution. Moreover, for batch estimation, we shall propose superfast solutions in connection to the proposed WL algorithms, so that further gains in performance are achieved for improper signals;

4. We investigate the state of the art of superfast realizations of RR transceivers w.r.t. the so-called linear *reduced redundancy* (RR) systems, which motivates the architectures considered in this work. We verify that unlike what has been claimed in the literature, MR schemes offer no advantage over standard schemes in neither MC or SC configurations, regardless of its use under coding or via *discrete Hartley transform* (DHT) implementations more recently proposed in [28]. This will be illustrated for both DFT and what the referred papers name as DHT based transceivers. We demonstrate that the latter corresponds simply to a DFT based expression, written in terms of DHT transforms, and that a negligible BER gain is obtained in comparison to standard schemes. Our simulation results contradict the conclusions in [28–32], and verify on the other hand, that simple reduced redundancy block decision feedback equalizers (DFEs), which are widely known, outperform the MC and SC-FD linear MR counterparts significantly, under much lower redundancy transmissions, with equal superfast complexity. We consider a simple *Extended Pedestrian A* (EPA) model of the *Long Term Evolution* (LTE) standard [33], and verify that redundancy cannot be reduced towards its optimal value in ZF

or MMSE equalization, a condition that can only be achieved by the proposed block DFE receivers;

5. In view of the discussion in the previous item, and given a primary goal of minimizing redundancy in block transmissions, we propose a new CS-based DFE algorithm to deal with signals belonging to signal constellations, which are capable of retrieving the information vector from an underdetermined transmission system (that is, below the zero redundancy level);

6. We derive a general purpose CS-BDFE algorithm intended to more strict, structured sensing problems. Instead of LMS type updates used by most iterative algorithms in CS, our approach relies on second-order methods, which although more computationally demanding, exhibits improved accuracy in terms of final estimate and target support;

7. In order to validate the previous result, we develop a full joint range/cross-range convolution model for *multi-input-multi-output* (MIMO) Radars, and obtain conditions for which CS techniques can be employed when reconstructing a volumetric image. After constructing the corresponding discrete model suitable for CS, we take a step further by decoupling it into two separate sparse problems, albeit ones that exhibit more structured separated models, convenient for efficient implementations. The recovery ability in this case is assessed by means of their coherence measure. Using synthetic simulations, and a *finite differences in time domain* (FDTD) [34] simulation of Maxwell's equations, we gauge both the model and the CS-BDFE algorithm performances, with excellent results.

This work is organized as follows. We provide a quick review on the basics of CS concepts, recovery guarantees, and CS algorithms, as well as introduce the CAMP algorithm in Sec. 1.1. The well known IHT and IST algorithms are obtained from a more general formulation, which shall motivate the CS-DFE based approach in the sequel.

In Chapter 2 the BI-DFE based recursion is shown to rise naturally from a RLS cost function, without a priori assumptions on the DFE feedback structure. We show how the well known V-BLAST algorithm follows from our derivation as a special case, and propose a fast computation of its constituting DFE matrices based on the FTF recursions. From the main CS problem that originates the IHT/IST algorithms, we show how to extend it in order to obtain a CS-based DFE recursion. The new algorithm makes use of a statistical hypothesis test that selects the correctly detected entries to be fed back in order to remove the 'columns interference' from

the linear model. The performance of the novel recursions is evaluated considering several distinct and useful scenarios.

In Chapter 3 we develop a CS algorithm starting from a stochastic formulation, yielding a new CS-BDFE algorithm intended to generic compressed sensing problems. We then proceed to evaluate the performance of the proposed CS-BDFE and the CAMP algorithms, by means of three different performance indicators, based on phase transition diagrams. The algorithms efficacy are verified for general purpose sensing matrices, structured block Toeplitz, as well as matrices designed exactly from a real MIMO radar setup. Our goal is to demonstrate that, despite being more complex, the superiority of the proposed CS-BDFE algorithm emerges as an alternative solution, when fast convergence and accuracy in detection are mandatory. This becomes specially relevant in difficult, ill-posed scenarios where the involved sensing matrices are not user-designed.

Chapter 4 applies our previous solutions to the important scenario of 3D radar imaging reconstruction recently developed by the author in [18, 19], considering a full joint range/cross-range convolution model for MIMO radars. We further relate the array geometry and transmitted pulses directly to the radar's recovery ability, assessed via the underlying sensing matrix coherence. After constructing such model, suitable for CS, we take a step further by decoupling it into two separate sparse problems, albeit ones that exhibit more structured linear relations, for efficient implementations. This model is then used by a CS-BDFE algorithm applied to a real-world radar system, considering three performance indicators. We consider a FDTD simulation of a real radar in order to show that the proposed CS-BDFE can even disregard secondary scattering, a non-linear effect that is not taken into account in our model in the first place, which appears as interference.

The publications that resulted from the research present in this dissertation are the following:

1. A **conference** paper "**Compressed sensing joint range and cross-range MIMO Radar imaging**" [18] presented in the *2015 International Conference on Acoustics Speech and Signal Processing* (ICASSP 2015), where the author develops the full 3D model for range and cross-range convolution in the MIMO radar setup, and its corresponding conditions for image recovery via compressed sensing — see Chapter 4, corresponding to item **7**;

2. A **conference** paper "**An efficient two-step procedure for compressed sensing 3D MIMO radar**" [19] presented in the *2015 23rd European Signal Processing Conference* (EUSIPCO 2015), proposing an efficient two-step procedure for CS — see Chapter 4 corresponding to item **7**;

3. A submitted **journal paper** "**Compressed Sensing Block Decision Feedback Equalization in Radar Imaging: Interplays and Connections for Sparse Recovery**" [35] to the *EURASIP Journal on Advances in Signal Processing*, describing the CD-BDFE algorithm, which is evaluated against the model presented in the previous publications, corresponding to contributions **6** and **7**;

4. A **journal paper (in preparation)** on the unification of CS algorithms for sparse recovery of signal constellation vectors in communications.

1.1 Compressed sensing

Compressed sensing is a technique for signal reconstruction, where the underlying linear model that relates measures and the searched parameter consists of an underdetermined system of equations. The main assumption in CS reckons on an a priori information that the signal vector to be recovered is sparse, i.e., most of its entries are null. More specifically, a vector is said to be k-sparse, if at most k of its entries are non-zero.

The motivation behind its application in signal processing stems from the fact that although a vast amount of real signals are not sparse, most can be represented by a sparse combination of bases, notably termed as *compressible signals*. One can see CS as way to tackle the ill-conditioning of a certain model, or to reduce the received signal sampling rate to more tractable levels.

Suppose we have a linear system $\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x}$, where $\boldsymbol{H}$ is underdetermined. To find the sparsest solution, one can resort to the minimization problem:

$$\begin{aligned} \boldsymbol{x}^o \;=\; & \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{x}\|_0 \\ & \text{s.t. } \boldsymbol{H}\boldsymbol{x} = \boldsymbol{y}, \end{aligned} \tag{1.1}$$

where $\|\cdot\|_0$ denotes the cardinality of a vector, often treated as a pseudo-norm. There are a few sufficient conditions to ensure that the minimizer $\boldsymbol{x}^o$ is unique.

The first sufficient condition is the so-called *null-space property*(NSP), which states that it suffices to recover any k-sparse vector such that no $2k$-sparse vectors are projected onto the null-space of the sensing matrix. The NSP can be characterized through the notion of *spark* of a matrix [1], which is given by the smallest number of its linearly dependent columns. A fundamental theorem by Donoho (Thm. 3 and Cor. 1 in [36]) shows that if

$$\|\boldsymbol{x}\|_0 < \frac{1}{2}\mathsf{Spark}(\boldsymbol{H}) \tag{1.2}$$

is a solution of (1.1), then it is the sparsest possible solution, and necessarily its unique minimizer. In general, this property is not of practical use, as the spark of a matrix is difficult to compute, requiring 2^P operations for an $M \times P$ matrix, except for a handful of structured matrices. The notion of spark, and in particular, the so-called *full spark* matrices, will be significant to our development, and will be elaborated in Sec. 2.10. Reference [37] presents some classes of matrices that are full spark, i.e., rectangular $M \times P$ matrices $\boldsymbol{H}$, $P \geq M$, where $\mathsf{Spark}(\boldsymbol{H}) = M + 1$.

A second condition for the existence of a unique minimizer $\boldsymbol{x}^o$ is based on another measure known as *restricted isometry property* (RIP), introduced by Candès and Tao[13]. The RIP is defined in terms of the isometry constant of a matrix $\boldsymbol{H}$, which

is the smallest number $\delta_k \in (0,1)$ such that

$$(1-\delta_k)\|\boldsymbol{x}\|_2^2 \leq \|\boldsymbol{H}\boldsymbol{x}\|_2^2 \leq (1+\delta_k)\|\boldsymbol{x}\|_2^2 \tag{1.3}$$

Loosely speaking, a matrix $\boldsymbol{H}$ is said to obey the RIP of order k, if δ_k is not too close to one, implying that it preserves the Euclidean distances between all k-sparse vectors[3].

To verify how RIP is related to the uniqueness of the minimizer of (1.1), take the difference between two k-sparse vectors, for which, in general, we obtain a $2k$-sparse vector. If $\boldsymbol{H}$ satisfies RIP of order $2k$, that is, $0 < \delta_{2k} < 1$, then it will preserve the distance between those two vectors. Hence, for any k-sparse minimizer $\boldsymbol{x}^o$ of a system whose measurement matrix obeys RIP of order $2k$, we have that

$$(1-\delta_{2k})\|\boldsymbol{x}^o - \boldsymbol{x}\|_2^2 \;\leq\; \|\boldsymbol{H}(\boldsymbol{x}^o - \boldsymbol{x})\|_2^2 \;\leq\; (1+\delta_{2k})\|\boldsymbol{x}^o - \boldsymbol{x}\|_2^2. \tag{1.4}$$

where $\boldsymbol{x}$ is any other k-sparse vector, $\boldsymbol{x} \neq \boldsymbol{x}^o$. Now, supposing $\delta_{2k} = 1$, the right hand side of (1.4) becomes zero, meaning that there may exist a $\boldsymbol{x}$ such that $\boldsymbol{H}(\boldsymbol{x}^o - \boldsymbol{x}) = \boldsymbol{0}$, thus violating the NSP. In this situation, it is not possible to recover all k-sparse vectors after projection by $\boldsymbol{H}$. However, just as difficult as calculating the spark, determining the isometry constant of a matrix is not practical and seldom used, except for a few classes of matrices where it can be determined by their structure.

A more tractable way to address the uniqueness of the minimizer in (1.1) makes use of the concept of *coherence* of a matrix, defined as:

$$\mu(\boldsymbol{H}) \triangleq \max_{i \neq j} \frac{|\boldsymbol{h}_i^* \boldsymbol{h}_j|}{\|\boldsymbol{h}_i\|_2 \|\boldsymbol{h}_j\|_2}, \tag{1.5}$$

where $\boldsymbol{h}_i$ denotes the i-th column of $\boldsymbol{H}$. As shown in Lemma 1.4 of [1], it is possible to use the Geršgorin circle theorem to relate the coherence of a matrix to its spark.

The relation between the spark and the mutual coherence, on the other hand, can be derived by contradiction. Without loss of generality, assume that the columns of $\boldsymbol{H}$ are normalized. Then, all diagonal elements of the $(P \times P)$ Gram matrix $\boldsymbol{\mathcal{G}}(\boldsymbol{H}) = \boldsymbol{H}^*\boldsymbol{H}$ will be equal to one, and the mutual coherence becomes the maximum absolute value of any off-diagonal element of $\boldsymbol{\mathcal{G}}(\boldsymbol{H})$. Let $\mathsf{Spark}(\boldsymbol{H}) = p$,

and denote by $\boldsymbol{\mathcal{B}}_p$ any $p \times p$ principal submatrix[1] of $\boldsymbol{\mathcal{G}}(\boldsymbol{H})$. Then, we must have

$$|[\boldsymbol{\mathcal{B}}_p]_{ii}| = 1\,, \tag{1.6}$$

$$\sum_{\substack{j=0 \\ j\neq i}}^{p-1} |[\boldsymbol{\mathcal{B}}_p]_{ij}| \leq (p-1)\mu(\boldsymbol{H})\,. \tag{1.7}$$

Now, suppose $p < 1+\mu(\boldsymbol{H})^{-1}$, so that $1 > (p-1)\mu(\boldsymbol{H})$. This implies that $|[\boldsymbol{\mathcal{B}}_p]_{ii}| > \sum_{j\neq i} |[\boldsymbol{\mathcal{B}}_p]_{ij}|$ and by the Geršgorin circles theorem, we must have $\boldsymbol{\mathcal{B}}_p > \mathbf{0}$. This is however a contradiction, since having positive definite submatrices $\boldsymbol{\mathcal{B}}_p$ means that $\boldsymbol{H}$ should have at least p linearly independent columns, i.e., $\mathsf{Spark}(\boldsymbol{H}) > p$. This establishes the following lowerbound:

$$\mathsf{Spark}(\boldsymbol{H}) \geq 1 + \frac{1}{\mu(\boldsymbol{H})}\,.$$

Although simple to compute, the spark obtained from the coherence measure can be considered a worst-case lowerbound[38], since often, the spark is much larger then coherence. As a consequence, using (1.2), any $k-$sparse solution of (1.1) is unique, given that

$$k < \frac{1}{2}\left(1 + \frac{1}{\mu(\boldsymbol{H})}\right). \tag{1.8}$$

1.1.1 Compressed sensing algorithms

Solving (1.1) directly is NP-hard, as it is a combinatorial optimization problem. Therefore, one of the greatest efforts within the CS community is to find efficient algorithms for solving this task. The existing CS algorithms found in the literature of CS can be roughly classified into three groups [2, 3]:

- Exact ℓ_1 regularized convex optimization,
- Greedy algorithms,
- Iterative Shrinkage algorithms.

[1] A $n \times n$ principal submatrix of a $P \times P$ matrix is obtained by removing from it $P-n$ similarly indexed, and not necessarily contiguous, columns and rows. As a consequence, the diagonal of a principal submatrix is composed by elements of diagonal of the full matrix.

ℓ_1 norm regularized least squares (LS)

Replacing the cardinality measure in (1.1) by an ℓ_1-norm, it collapses to the formulation known as *Basis Pursuit*(BP):

$$\begin{aligned} \boldsymbol{x}^o \;&=\; \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{x}\|_1 \\ &\text{s.t. } \boldsymbol{H}\boldsymbol{x} = \boldsymbol{y}. \end{aligned} \tag{1.9}$$

A result due to Donoho (Thm. 7 in [36]) shows that, if (1.8) holds, then the unique minimizer of (1.1) is also the unique minimizer of (1.9). The algorithms for solving ℓ_1-norm regularized least squares (LS) problems have a long history of development, leading to very efficient implementations [39–41]. Another result by Candès (Thm. 1.1 in [42]) shows that while the isometry constant $\delta_{2k} < 1$ ensures the uniqueness of the minimizer of (1.1), if $\delta_{2k} < \sqrt{2} - 1$, the minimizer of (1.9) is also a unique minimizer of (1.1).

In a noisy environment, the model is described as $\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{v}$, where $\boldsymbol{v}$ is an uncorrelated noise; for such models, we can relax the constraint of the BP in (1.9) and state the problem as

$$\begin{aligned} \boldsymbol{x}^o \;&=\; \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{x}\|_1 \\ &\text{s.t. } \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2 \leq \sigma. \end{aligned} \tag{1.10}$$

This is known as *Basis Pursuit De-Noising*(BPDN) formulation. Note that the BP is essentially the BPDN with $\sigma = 0$. Both algorithms were originally proposed for decomposing signals in terms of atoms in an overcomplete expansion [43]. A mathematically equivalent formulation, which also arises within the statistics community is given by

$$\begin{aligned} \boldsymbol{x}^o \;&=\; \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2 \\ &\text{s.t. } \|\boldsymbol{x}\|_1 < \rho. \end{aligned} \tag{1.11}$$

For appropriate choices of σ and ρ, this is frequently referred to as the *Least Absolute Shrinkage and Selection Operator*(LASSO) (named after an iterative algorithm for solving this objective function[39]). Moreover, both problems can be formulated as an unconstrained (regularized) minimization, i.e.,

$$\boldsymbol{x}^o \;=\; \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2^2 \;+\; \epsilon\|\boldsymbol{x}\|_1, \tag{1.12}$$

and are equivalent to (1.9) and (1.10) under a proper choice of the lagrange multiplier ϵ (see [44]). Theorem 1.2 in [42] shows that, for a small enough isometry constant

δ_{2k}, the minimizers of (1.10), (1.11) and (1.12) are the unique minimizers of (1.1).

Generally speaking, these problem can be solved by a *quadratic constrained quadratic program* (QCQP), for which many solvers are readily available. In CS applications the number of variables can be quite large, and QCQP becomes prohibitively slow, as the underlying *Karush-Khun-Tucker* (KKT) system of equations [44] becomes too complex. To overcome this limitation, gradient descent algorithms based on subgradient operators, like LASSO [39], ℓ_1-magic [40] and SPGL1 [41] have been extensively studied and motivated through different perspectives.

Greedy algorithms

Greedy algorithms mainly seek the active columns of the sensing matrix, i.e., the support of $\boldsymbol{x}$, and then estimate those parameters. In this direction, one of the most popular strategies relies on the *Orthogonal Matching Pursuit* (OMP) algorithm [4], also developed for decomposing signals via overcomplete bases. In OMP, the inner product between the measurement vector and each column of the measurement matrix is used to detect the support. After a new column is added to the support of the parameter vector, the parameter is estimated via LS. In the next iteration, the estimation error is used as measurement vector, and the algorithm continues until the error becomes smaller than a user-defined level. Since those vectors are orthogonal to each other[14], OMP will never reselect a column of the measurement matrix.

Many algorithms extend the idea behind OMP. Some examples include: the *Stagewise* OMP (StOMP) [8], the *Compressed Sampling Matched Pursuit* (CoSaMP) and *Subspace Pursuit* (SP)[1]. While the first differs in the way the active columns are selected, the last two algorithms extend the idea of OMP by obtaining the entire support in each iteration, and then employing a gradient descent method for subsequent LS estimations.

Iterative shrinkage algorithms

Iterative algorithms have become an attractive alternative for sparsity recovery, and can be shown to minimize either (1.1) or (1.9) without the burden of performing direct matrix inversions with respect to the sensing matrix in the approaches aforementioned, or of adding constraint variables to the problem. Thus, consider the following quadratically regularized weighted LS cost function:

$$J_S(\boldsymbol{x};\boldsymbol{x}_{i-1}) \;=\; \|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{x}\|^2_{\boldsymbol{\mathcal{W}}} \;+\; \|\boldsymbol{x}-\boldsymbol{x}_{i-1}\|^2_{\boldsymbol{\mathcal{R}}} \;+\; \varrho(\boldsymbol{x}) \tag{1.13}$$

where $\boldsymbol{x}_{i-1}$ is a prior estimate for the vector parameter, and $\|\boldsymbol{z}\|^2_{\boldsymbol{\mathcal{W}}} \triangleq \boldsymbol{z}^*\boldsymbol{\mathcal{W}}\boldsymbol{z}$ is the weighted squared Euclidean norm. The resulting recursions build upon the notion of *Proximal Mappings* [2, 3, 45], commonly employed in CS, leading to different algorithms depending on the choice of $\varrho(\boldsymbol{x})$. An important class of CS recursions known as the *iterative hard thresholding* (IHT) [7] algorithm, can be obtained via minimization of (1.13), by choosing $\varrho(\boldsymbol{x}) = \epsilon\|\boldsymbol{x}\|_0$, and setting

$$\begin{aligned} \boldsymbol{\mathcal{W}} &= \boldsymbol{I} \\ \boldsymbol{\mathcal{R}} &= \boldsymbol{I} - \boldsymbol{H}^*\boldsymbol{H}. \end{aligned}$$

In the CS community, regularization is referred to as a *surrogate cost* [7] with respect to the pure LS problem, and represents nothing but the incorporation of uncertainty into the problem with respect to $\boldsymbol{x}_{i-1}$. These choices are the key to achieving a simple recursion; when $\boldsymbol{\mathcal{R}} > 0$, we have $J_S(\boldsymbol{x};\boldsymbol{x}_{i-1}) \geq J(\boldsymbol{x}) = J_S(\boldsymbol{x};\boldsymbol{x})$, so that the surrogate cost becomes an upper bound for the original ℓ_0 problem. Therefore, by iteratively minimizing $J_S(\boldsymbol{x};\boldsymbol{x}_{i-1})$, we are ultimately using a majorization-minimization mechanism in order to find the minimum of $J(\boldsymbol{x})$.

When $\|\boldsymbol{H}\|_2 < 1$, we have that $\boldsymbol{\mathcal{R}} = \boldsymbol{I} - \boldsymbol{H}^*\boldsymbol{H}$ is positive definite, and with the selection $\boldsymbol{\mathcal{W}} = \boldsymbol{I}$, it can be verified that the optimization of $J_S(\boldsymbol{x};\boldsymbol{x}_{i-1})$ in (1.13) can be carried ***entrywise*** [7, 23]. Thus, for $\varrho(\boldsymbol{x}) = \epsilon\|\boldsymbol{x}\|_0$, by expanding the squares in (1.13), we have

$$\begin{aligned} J_S(\boldsymbol{x};\boldsymbol{x}_{i-1}) = \boldsymbol{y}^*\boldsymbol{y} &+ \boldsymbol{x}^*\boldsymbol{H}^*\boldsymbol{H}\boldsymbol{x} - \boldsymbol{x}^*\boldsymbol{H}^*\boldsymbol{y} - \boldsymbol{y}^*\boldsymbol{H}\boldsymbol{x} \\ &+ \boldsymbol{x}^*\boldsymbol{x} - \boldsymbol{x}^*\boldsymbol{H}^*\boldsymbol{H}\boldsymbol{x} + \boldsymbol{x}_{i-1}^*\boldsymbol{x}_{i-1} - \boldsymbol{x}_{i-1}^*\boldsymbol{H}^*\boldsymbol{H}\boldsymbol{x}_{i-1} \\ &- \boldsymbol{x}^*\boldsymbol{x}_{i-1} + \boldsymbol{x}^*\boldsymbol{H}^*\boldsymbol{H}\boldsymbol{x}_{i-1} - \boldsymbol{x}_{i-1}^*\boldsymbol{x} + \boldsymbol{x}_{i-1}^*\boldsymbol{H}^*\boldsymbol{H}\boldsymbol{x} \\ &+ \epsilon\|\boldsymbol{x}\|_0 \end{aligned} \tag{1.14}$$

so that grouping this expression in terms of $\boldsymbol{x}$ and $\boldsymbol{x}^*$, we have

$$\begin{aligned} J_S(\boldsymbol{x};\boldsymbol{x}_{i-1}) = \|\boldsymbol{y}\|^2 &+ \|\boldsymbol{x}_{i-1}\|^2 - \|\boldsymbol{H}\boldsymbol{x}_{i-1}\|^2 \\ &+ \boldsymbol{x}^*\boldsymbol{x} \\ &+ \boldsymbol{x}^*\left[\boldsymbol{H}^*\boldsymbol{H}\boldsymbol{x}_{i-1} - \boldsymbol{x}_{i-1} - \boldsymbol{H}^*\boldsymbol{y}\right] \\ &+ \left[\boldsymbol{x}_{i-1}^*\boldsymbol{H}^*\boldsymbol{H} - \boldsymbol{x}_{i-1}^* - \boldsymbol{y}^*\boldsymbol{H}\right]\boldsymbol{x} \\ &+ \epsilon\|\boldsymbol{x}\|_0 \end{aligned} \tag{1.15}$$

Now, by introducing

$$\hat{\boldsymbol{x}}_i = \boldsymbol{x}_{i-1} + \boldsymbol{H}^*\left(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}_{i-1}\right),$$

this cost can be written as

$$\begin{aligned} J_S(\boldsymbol{x};\boldsymbol{x}_{i-1}) = \|\boldsymbol{y}\|^2 + \|\boldsymbol{x}_{i-1}\|^2 - \|\boldsymbol{H}\boldsymbol{x}_{i-1}\|^2 \\ + \boldsymbol{x}^*\boldsymbol{x} - \boldsymbol{x}^*\hat{\boldsymbol{x}}_i - \hat{\boldsymbol{x}}_i^*\boldsymbol{x} + \epsilon\|\boldsymbol{x}\|_0 \end{aligned} \tag{1.16}$$

Note that the first three terms are constants, while the last four terms can be written as a summation:

$$\begin{aligned} J_S(\boldsymbol{x};\boldsymbol{x}_{i-1}) = \|\boldsymbol{y}\|^2 + \|\boldsymbol{x}_{i-1}\|^2 - \|\boldsymbol{H}\boldsymbol{x}_{i-1}\|^2 \\ + \sum_{k=0}^{P-1} x^*(k)x(k) - x^*(k)\hat{x}_i(k) - \hat{x}_i^*(k)x(k) + \epsilon|x(k)|_0, \end{aligned} \tag{1.17}$$

where $|x(k)|_0$ is defined as the indicator function that returns 0 whenever $x(k) = 0$, and 1 otherwise. As a consequence, we can minimize $J_S(\boldsymbol{x};\boldsymbol{x}_{i-1})$ for $\boldsymbol{x}_i$ entrywise, by introducing the scalar function

$$D(x(k)) = x^*(k)x(k) + x^*(k)\hat{x}_i(k) + \hat{x}_i^*(k)x(k) + \epsilon|x(k)|_0, \tag{1.18}$$

which evaluates to

$$D(x(k)) = \begin{cases} 0, & \text{if } x(k) = 0 \\ x^*(k)x(k) - x^*(k)\hat{x}_i(k) - \hat{x}_i^*(k)x(k) + \epsilon, & \text{otherwise} \end{cases} \tag{1.19}$$

In the second line of (1.19), the minimum is achieved when

$$\frac{\partial D(x(k))}{\partial x^*(k)} = 0, \text{ or} \tag{1.20}$$

$$x(k) = \hat{x}_i(k), \tag{1.21}$$

and, in this case, $D(\hat{x}_i(k)) = \epsilon - |\hat{x}_i(k)|^2$, which becomes negative if $|\hat{x}_i(k)| \geq \sqrt{\epsilon}$. Thus, the k-th element of $\boldsymbol{x}^o$ is given by

$$x^o(k) = \begin{cases} \hat{x}_i(k), & \text{if } |\hat{x}_i(k)| \geq \sqrt{\epsilon} \\ 0, & \text{otherwise} \end{cases} \tag{1.22}$$

The resulting IHT algorithm assumes the following form:

$$\boldsymbol{x}_i = \mathcal{T}_{\sqrt{\epsilon}}\left[\boldsymbol{x}_{i-1} + \boldsymbol{H}^*\left(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}_{i-1}\right)\right], \tag{1.23}$$

where $\mathcal{T}_{\sqrt{\epsilon}}(\cdot)$ is the hard threshold operator defined as

$$\mathcal{T}_{\sqrt{\epsilon}}(z) = \begin{cases} z, & \text{if } |z| > \sqrt{\epsilon} \\ 0, & \text{otherwise} \end{cases}$$

On the other hand, by choosing $\varrho(\boldsymbol{x}) = \epsilon\|\boldsymbol{x}\|_1$ in the surrogate cost (1.13), and by following similar same steps that led to the IHT, we obtain what is known as the *iterative soft threshold* (IST) [5] recursion:

$$\boldsymbol{x}_i = \mathcal{S}_\epsilon \left[\boldsymbol{x}_{i-1} + \boldsymbol{H}^* \left(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}_{i-1}\right)\right], \tag{1.24}$$

where $\mathcal{S}_\epsilon(\cdot)$ refers to the soft-threshold operator, defined as

$$\mathcal{S}_\epsilon(z) = \begin{cases} z\left(1 - \frac{\epsilon}{|z|}\right), & \text{if } |z| > \epsilon \\ 0, & \text{otherwise} \end{cases} \tag{1.25}$$

Because of their simplicity, both IST and IHT are widely used in CS, despite some shortcomings that impair their recovery ability in some scenarios [23]. The main one is related to selecting the threshold parameter ϵ. While there is no direct relation between ϵ and the signal-to-noise ratio (SNR) or, to the sparsity of the vector parameter, those two factors are of major influence in the ability of recovery, given a threshold value. Hence, for the IHT algorithm, there is no guarantee that it will reach a fixed point, so that the solution obtained may not be a sparse one; for the IST algorithm, on the other hand, for a given threshold, the SNR heavily impairs detection. Thus, either an adaptive procedure or a selection heuristic must be employed, in order for these algorithms to properly work. Usually, the IHT threshold is determined indirectly, by first selecting a sparsity level, and then setting the corresponding smallest entries of the estimate to zero, at each iteration. For the IST algorithm such heuristic does not work, and a more sophisticated adaptive scheme must be pursued.

One of the advantages of thresholding algorithms is that, besides avoiding matrix inversions in exact LS problems, compared to Greedy iterations, the solution support is entirely estimated at each new iteration, which further allows for recursive corrections until convergence.

Approximate Message Passing (AMP) algorithm

Message passing algorithms constitute one of the most recent advances emerging in the field of CS, and it is strongly based on the theory of graphs, and Bayesian networks [1, 9, 10, 46]. They are designed to be as fast as IST, but less sensitive to noise variations and sparsity. The idea behind those algorithms is to split the unconstrained ℓ_1 regularized cost from (1.12) in two parts, corresponding to the neighborhoods of the variables x_i and the measurements, also called *factors*. That

is, we write the target cost (assuming the real case scenario) as

$$J(\boldsymbol{x}) = \sum_{j \in F} (y_j - [\boldsymbol{H}]_{j,:}\boldsymbol{x}) + \lambda \sum_{x \in V} |x_i|, \tag{1.26}$$

where F and V are the sets of factors and variables, respectively, and $[\boldsymbol{H}]_{j,:}$ is the j-th row of $\boldsymbol{H}$. Under this philosophy, (1.26) can be approached from a graph perspective, where the well known min-sum algorithm can be applied [1, 9, 10, 23]. The resulting recursions are very similar to the IST algorithm and are given by

$$\boldsymbol{x}_i = \mathcal{S}_{\theta_i}(\boldsymbol{x}_{i-1} + \boldsymbol{H}^T \boldsymbol{e}_{i-1}) \tag{1.27}$$

$$\boldsymbol{e}_i = \boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}_{i-1} + \frac{1}{P}\|\boldsymbol{x}_{i-1}\|_0 \boldsymbol{e}_{i-1} \tag{1.28}$$

where P is the length of $\boldsymbol{x}$. The algorithm starts with $\boldsymbol{e}_0 = \boldsymbol{y}$ and $\boldsymbol{x}_0 = \boldsymbol{0}$, and iterates until $\|\boldsymbol{x}_i - \boldsymbol{x}_{i-1}\|$ reaches a user determined level. The last term on the second equation on (1.27), is known as *Onsager* term, and relates to the sparsity of the solution. The threshold value θ_i is determined by the Lagrange multiplier λ of the LASSO in (1.26), resulting

$$\theta_i = \frac{\lambda}{\left(1 - \frac{1}{P}\|\boldsymbol{x}_{i-1}\|_0\right)} \tag{1.29}$$

In practice, λ is chosen empirically and can be adapted throughout the iterations.

Equation (1.27) accounts only for real valued vectors and matrices. For complex valued problems, the *complex* AMP(CAMP)[10] was proposed, with recursions still similar to the AMP:

$$\hat{\boldsymbol{x}}_i = \boldsymbol{x}_{i-1} + \boldsymbol{H}^* \boldsymbol{e}_{i-1} \tag{1.30}$$

$$\boldsymbol{e}_i = \boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}_{i-1} + \boldsymbol{e}_{i-1}\frac{1}{2P}\sum_{j=0}^{m-1}\left[\mathcal{S}'_{\theta_i;\Re}\left(\hat{x}_i(j)\right) + \mathcal{S}'_{\theta_i;\Im}\left(\hat{x}_i(j)\right)\right] \tag{1.31}$$

$$\boldsymbol{x}_i = \mathcal{S}_{\theta_i}\left(\hat{\boldsymbol{x}}_i\right) \tag{1.32}$$

where

$$\mathcal{S}'_{\theta_t;\Re}(x) = \Re\left(\frac{\partial \mathcal{S}_{\theta_t}(x)}{\partial \Re(x)}\right) \quad \text{and} \quad \mathcal{S}'_{\theta_t;\Im}(x) = \Im\left(\frac{\partial \mathcal{S}_{\theta_t}(x)}{\partial \Im(x)}\right).$$

Unlike AMP, in CAMP recursions, the threshold value cannot be obtained directly from the Lagrange multiplier, so that the authors suggest using the following heuristics,

$$\theta_i = \kappa\sqrt{\frac{1}{\ln 2}}\,\text{median}\left(|\hat{\boldsymbol{x}}_{i-1}|\right), \tag{1.33}$$

where κ is a user-defined parameter that controls the regularization. As in AMP, the iterations begin with $\boldsymbol{e}_0 = \boldsymbol{y}$ and $\boldsymbol{x}_0 = \boldsymbol{0}$, and stop when $\|\boldsymbol{x}_i - \boldsymbol{x}_{i-1}\|$ reaches a

lower limit.

The main advantage of the AMP and the CAMP algorithms over the pure thresholding methods aforementioned is that they do not require a complex heuristics for avoiding divergence for typical sensing matrices [46] (although reference [47] points out that it might fail to converge for arbitrary ones). Message passing algorithms can be shown to converge to the LASSO solution in the limit, and usually show superior performance when compared to the IST [1, 10, 46]. In this sense, CAMP is usually the best choice when little information about noise or sparsity of the input is available [23].

Chapter 2

A Unified Approach to Compressed Sensing and Block-Iterative Decision Feedback Equalizers

Ideally speaking, when dealing with a Finite Impulse Response (FIR) channel, its equalizer as an inversion operation should be an all-pole Infinite Impulse Response (IIR) filter[48]. In practice, this poses some difficulties in equalization, as FIR channels often appear with non-minimum phase zeros, leading to unstable equalizers. Moreover, the presence of poles near the unity circle would amplify significantly the channel noise. Decision Feedback Equalizers were originally proposed to ensure that all signals in the equalizer are bounded by feeding back detected symbols, unlike an all-pole IIR filter would do. Such equalizers are known to greatly outperform their linear counterparts[48].

While in the literature, (block) DFE equalizers are traditionally derived with the optimization of feedforward and feedback matrices in a stochastic sense [11, 12, 24], which in turn are derived from a pre-determined structure, in this chapter we show that a properly formulated BI-DFE can be seen as a CS algorithm obtained in its own right, and that such architecture in fact arises naturally from a suitably regularized LS problem.

Moreover, unlike the original BI-DFE formulation, where uncertainty throughout the iterations can be a pre-computed, computationally demanding phase [11], we shall show that a CS-based formulation allows for an adaptive calculation of uncertainty in detection, which is available from the error covariance before slicing. We pursue the new algorithms under different policies of assigning uncertainty to the detected entries. Each one exploits a different strategy for symbol feedback,

exhibiting their own complexity and performance advantages.

Specifically, the contributions of this chapter are the following:

1. We show that a properly formulated BI-DFE can be seen as a CS algorithm obtained in its own right, and not as a pre-imposed structure optimized according to some criterion. While traditional formulations of (block) DFEs commonly begin with the optimization of feedforward and feedback matrices in a stochastic sense, such architecture in fact arises naturally from a suitably regularized LS problem, and is therefore optimal in this sense.

2. As a Greedy approach to recursive estimation, we show that the *Vertical Bell Labs Layered Space Time*(V-BLAST) algorithm [27] employing sequential detection of entries can make use of a *fast transversal filter* (FTF) in order to compute the DFE matrices, henceforth providing a new alternative to the complexity issue implied by the Cholesky factorization inherent to its solution.

3. We derive a *Reduced-Complexity* (RC) *Widely-Linear* (WL) [25] version of BI-DFE, which does not require the traditional circularity assumption on the involved target and noise vectors.

4. We motivate the new architectures of this work by investigating the so-called linear *reduced redundancy* (RR) systems which have been recently published by the authors in [28–32], and clarify on the actual value of these implementations. As verified in our experiments, we first conclude that superfast MR schemes offer no advantage over standard multicarrier and single carrier schemes, regardless of its use under coding or via 'new' discrete Hartley transform (DHT) implementations as claimed in [28]. This will be illustrated for both DFT and what the referred papers name as DHT based transceivers; We demonstrate that the latter corresponds to a DFT based expression, written in terms of DHT transforms, and that a negligible BER gain is obtained in comparison to standard schemes. Our simulation results contradict the conclusions in [28–32], and verify on the other hand, that simple reduced redundancy block decision feedback equalizers (DFEs), which are widely known, outperform the MC and SC-FD linear MR counterparts significantly, under much lower redundancy transmissions, and equal superfast complexity. We consider a simple *Extended Pedestrian A* (EPA) model of the *Long Term Evolution* (LTE) standard [33], and verify that redundancy cannot be reduced towards its optimal value in ZF or MMSE equalization, a condition that can only be achieved by the proposed block DFE receivers.

5. Motivated by the ever increasing demands of throughput in digital transmission systems, and the desire to achieve low bit-error-rates (BER) simultaneously with a scenario of minimized redundancy transmission, we introduce a novel CS-based algorithm that relies on RLS iterations, which outperforms the traditional BI-DFE approaches. The CS interpretation of the original BI-DFE allows us to go beyond minimum redundancy transmissions, and even below the zero redundancy level, by exploiting sparsity in typical signal modulations. The crucial point we elaborate on is also motivated by the discussion in Sec. 1.1.1 regarding the difficulty in selecting regularization parameters in CS algorithms. We shall propose an adaptive calculation of the uncertainty in detection, which will be available from the error covariance computed before slicing at every iteration of the algorithm.

The chapter is organized as follows. In Section 2.1 we show that the RLS algorithm (or Kalman Filter from a stochastic point of view) can be interpreted as a block iterative DFE, without direct optimization of feedforward and feedback matrices. Section 2.2 presents the DFE algorithm from a stochastic approach, and discusses its optimality in the presence of the slicer. We then extend the deterministic derivation of the DFE in Sec. 2.3 by showing that a single step of a RLS iteration collapses to the well-known DFE expressions. In Sec. 2.4, we extend the Strictly Linear model commonly used in the literature to a Widely-Linear one. In Sec. 2.5 we show that a successive cancelation DFE algorithm based on the DFE formulation of Sec. 2.3 collapses to the V-BLAST algorithm. We then propose a particular form of the V-BLAST where detection is performed sequentially, in a decreasing index order of entries, and whose DFE feedforward matrices can be obtained from an FTF recursion. In Sec. 2.7 we review the reduced redundancy methods of IBI canceling, and superfast structures for DFT based receivers, which are used as MMSE estimators for initializing the DFE algorithms. In Sec. 2.9, we discuss the claims from [28–32] on minimum and reduced redundancy schemes, which motivate the new iterative solutions proposed in this work. We show that the performance of such schemes depends on the channel itself, such that for some channels there is no gain in terms of BER and throughput that justify a more complex, superfast realization. Finally in Sec. 2.10 we incorporate an additional regularization function to the RLS cost, which will lead us to a new CS-based BI-DFE structure. The performance of the new algorithm is evaluated in Sec. 2.12.

2.1 Iterative Estimation in Block-Based Equalizers — Motivation

Figure 2.1 illustrates a generic precoded transmission scheme in terms of a synthesis filter bank followed by a length-L discrete LTI single-input-single-output (SISO) channel $H(z)$. This system corresponds to the trasmitting end of a so-called digital transmultiplexer [48]. We assume that the reader is acquainted with the theory of multirate systems, and their descriptions as modern digital transmultiplexers. Their interpretation as block-based transmissions, are widely used nowadays.

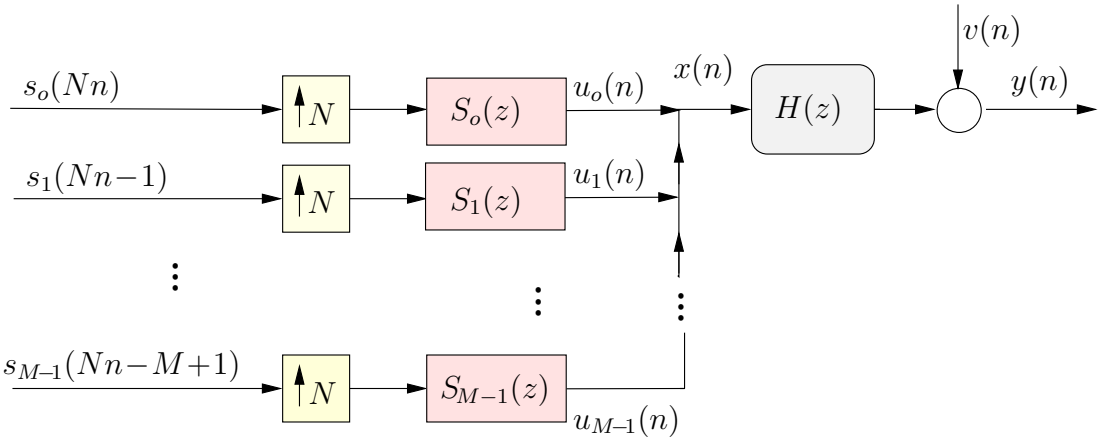


Figure 2.1: Block Transmission Scheme.

In a digital transmultiplexer, the sampled symbols $s_k(n)$, $0 \leq k \leq M-1$ are passed through a set of interpolation filters, which produce $u_k(n)$, and are combined to form a unique stream $x(n)$ that is transmitted over the channel, described by the transfer function $H(z)$. In the receiving end, decimating filters separate and restore the original rate of the transmitted signals. With the proper value of N, by using well known multirate identities and polyphase representations for the involved quantities, the channel $H(z)$ can be described via a $P \times M$ matrix-valued transfer function $(M \leq P)$

$$\boldsymbol{\mathcal{H}}(z) \triangleq \sum_{\ell=0}^{L_B-1} \boldsymbol{\mathcal{H}}_\ell \, z^{-k} \,, \tag{2.1}$$

where $L_B = \lfloor P+L-1 \rfloor / M + 1$ is the corresponding number of block coefficients $\{\boldsymbol{\mathcal{H}}_\ell\}$ defining $\boldsymbol{\mathcal{H}}(z)$. Let $\mathbf{x}_n = [\, x(n) \quad x(n-1) \quad \cdots \quad x(n-P-L+2) \,]^\mathsf{T} = [\, \boldsymbol{x}_n^\mathsf{T} \quad \boldsymbol{x}_{n-1}^\mathsf{T} \quad \cdots \quad \boldsymbol{x}_{n-L_B+1}^\mathsf{T}(0:r-1) \,]^\mathsf{T}$, with $r = (P+L-1) \bmod M$, be the full $(P+L-1) \times 1$ transmitted sequence at time i. The received block can be

equivalently written as

$$\mathbf{y}_n = \sum_{\ell=0}^{L_B-1} \boldsymbol{\mathcal{H}}_\ell \, \boldsymbol{x}_{n-\ell} \tag{2.2}$$

$$= \boldsymbol{H}\mathbf{x}_n + \boldsymbol{v}_n, \tag{2.3}$$

where we form the $P \times (P + L - 1)$ channel matrix

$$\boldsymbol{H} = [\, \boldsymbol{\mathcal{H}}_0 \;\; \boldsymbol{\mathcal{H}}_1 \;\; \boldsymbol{\mathcal{H}}_2 \;\cdots\; \boldsymbol{\mathcal{H}}_{L_B-1}(:, 0\!:\!r-1) \,]$$

For the sake of generality, we consider a generic block affine precoding scheme for the transmitted sequence, i.e., $\mathbf{x}_n = \mathbf{T}_n\mathbf{s}_n + \mathbf{t}_n$, where $\mathbf{s}_n = [\, \boldsymbol{s}_n^{\mathsf{T}} \;\; \boldsymbol{s}_{n-1}^{\mathsf{T}} \;\cdots\; \boldsymbol{s}_{n-L_B+1}^{\mathsf{T}}(0\!:\!r-1) \,]^{\mathsf{T}}$ denotes the information vector, and $\mathbf{t}_n$ is a possibly superimposed training sequence used for estimating the channel within the n-th transmitted block.

Now, consider the linear model (2.3), and assume for simplicity that $\mathbf{t}_n = \mathbf{0}$, and $\mathbf{T}_n = \boldsymbol{I}$, so that $\mathbf{s}_n = \mathbf{x}_n$. For compactness of notation, we shall drop the time index n in $\{\mathbf{x}_n, \mathbf{y}_n, \boldsymbol{v}_n\}$. Our goal is to demonstrate how an iterative DFE receiver naturally arises as a Kalman adaptive algorithm without using the common assumptions used in its conventional derivation. The problem of recursive estimation of a transmitted vector $\mathbf{x}$ can be seen as one that originates from a *constant* state-space model, i.e.,

$$\mathbf{x}_{k+1} = \boldsymbol{F}\mathbf{x}_k \tag{2.4}$$

$$\mathbf{y}_k = \boldsymbol{H}\mathbf{x}_k + \mathbf{v} \tag{2.5}$$

where $\boldsymbol{F} = \boldsymbol{I}$, and k is now the iteration index for the transmitted block estimate $\mathbf{x}$. Define the block-column quantities

$$\mathbf{v}_k = \mathsf{col}\{\mathbf{v}, \mathbf{v}, \ldots, \mathbf{v}\} \tag{2.6}$$

$$\mathbf{y}_k = \mathsf{col}\{\mathbf{y}, \mathbf{y}, \ldots, \mathbf{y}\} \tag{2.7}$$

$$\mathbf{H}_k = \mathsf{col}\{\boldsymbol{H}, \boldsymbol{H}, \ldots, \boldsymbol{H}\} \tag{2.8}$$

comprising $k + 1$ measurements of $\{\mathbf{y}, \mathbf{v}, \boldsymbol{H}\}$, as well as the corresponding block-diagonal matrix

$$\mathbf{W}_k^{-1} \triangleq (\boldsymbol{R}_0 \oplus \boldsymbol{R}_1 \oplus \ldots \oplus \boldsymbol{R}_k) \tag{2.9}$$

with constant blocks $\boldsymbol{R}_j = \boldsymbol{R}$. We pose the problem of estimating $\mathbf{x}_0 = \mathbf{x}$, from the sequence of observations $\mathbf{y}$ in a regularized fashion as

$$\min_{\mathbf{x}} \quad \|\mathbf{y}_k - \mathbf{H}_k\mathbf{x}\|^2_{\mathbf{W}_k} \;+\; \|\mathbf{x}\|^2_{\boldsymbol{\Pi}_0^{-1}} \tag{2.10}$$

Thus, fix a time instant i and assume that a filtered MMSE estimate $\hat{\mathbf{x}}_{i-1}$ has been forcefully projected onto $\mathbf{x}_{i-1} = f(\hat{\mathbf{x}}_{i-1})$. This is commonly the case, for example, when $\mathbf{x}$ belongs to a signal constellation, and $f(\cdot)$ is some approximate threshold function to ensure that $\hat{\mathbf{x}}_i$ belongs to the support of $\mathbf{x}$. Denoting the residue of this projection as

$$\bar{\mathbf{x}}_{i-1} \triangleq \mathbf{x} - \mathbf{x}_{i-1}, \tag{2.11}$$

with corresponding error variance

$$\boldsymbol{P}_{i-1} \;=\; \mathsf{E}\bar{\mathbf{x}}_{i-1}\bar{\mathbf{x}}_{i-1}^{*}, \tag{2.12}$$

then, given the channel output measurement $\mathbf{y}$, (2.10) can be equivalently posed as

$$\min_{\mathbf{x}} \;\; \|\mathbf{y} - \boldsymbol{H}\mathbf{x}\|^2_{\boldsymbol{R}_v^{-1}} \;+\; \|\mathbf{x} - \mathbf{x}_{i-1}\|^2_{\boldsymbol{P}_{i-1}^{-1}} \tag{2.13}$$

The solution $\hat{\mathbf{x}}_i$ of any such regularized problem can be recursively computed as [14]

$$\begin{aligned}
\boldsymbol{R}_{\bar{e},i} &= \boldsymbol{R}_v + \boldsymbol{H}\boldsymbol{P}_{i-1}\boldsymbol{H}^{*} && && (2.14)\\
\boldsymbol{K}_{p,i} &= \boldsymbol{P}_{i-1}\boldsymbol{H}^{*}\boldsymbol{R}_{\bar{e},i}^{-1} && && (2.15)\\
\bar{\mathbf{e}}_i &= \mathbf{y} - \boldsymbol{H}\mathbf{x}_{i-1} && && (2.16)\\
\hat{\mathbf{x}}_i &= \mathbf{x}_{i-1} + \boldsymbol{K}_{p,i}\bar{\mathbf{e}}_i\,, && \hat{\mathbf{x}}_0 = \mathbf{0} && (2.17)\\
\widehat{\boldsymbol{P}}_i &= \boldsymbol{P}_{i-1} - \boldsymbol{K}_{p,i}\boldsymbol{R}_{\bar{e},i}\boldsymbol{K}_{p,i}^{*} && && (2.18)
\end{aligned}$$

and a new estimate for $\mathbf{x}$ taken as

$$\mathbf{x}_i = f(\hat{\mathbf{x}}_i) \tag{2.19}$$

Observe that in order to compute the solution, we should be able to relate the LS variance $\widehat{\boldsymbol{P}}_i$ and the corresponding error variance $\boldsymbol{P}_i$ after $f(\cdot)$ is applied. In particular, using a linear function $f(\hat{\mathbf{x}}_i) = \hat{\mathbf{x}}_i$, we have $\mathbf{x}_i = \hat{\mathbf{x}}_i$, and $\widehat{\boldsymbol{P}}_i = \boldsymbol{P}_i$, so that in this case, the recursions simply amount to the well known RLS algorithm, or Kalman recursions from a stochastic point of view (The latter can be assured for jointly Gaussian variables, when the optimal estimator in the MMSE sense becomes an affine one. For other signal distributions, such as the one considered in this work, this approximation is good enough[14, 49]).

Combining (2.17) and (2.19), $\mathbf{x}_i$ can be written alternatively as

$$\mathbf{x}_i \;=\; f(\boldsymbol{F}_{p,i}\mathbf{x}_{i-1} + \boldsymbol{K}_{p,i}\mathbf{y}) \tag{2.20}$$

with

$$\boldsymbol{F}_{p,i} = \boldsymbol{I} - \boldsymbol{K}_{p,i}\boldsymbol{H} \tag{2.21}$$

Hence, referring to Fig. 2.2, one may readily observe its resemblance to a DFE that estimates $\boldsymbol{x}$ through feedforward and feedback matrices $\{\boldsymbol{K}_{p,i}, \boldsymbol{F}_{p,i}\}$.

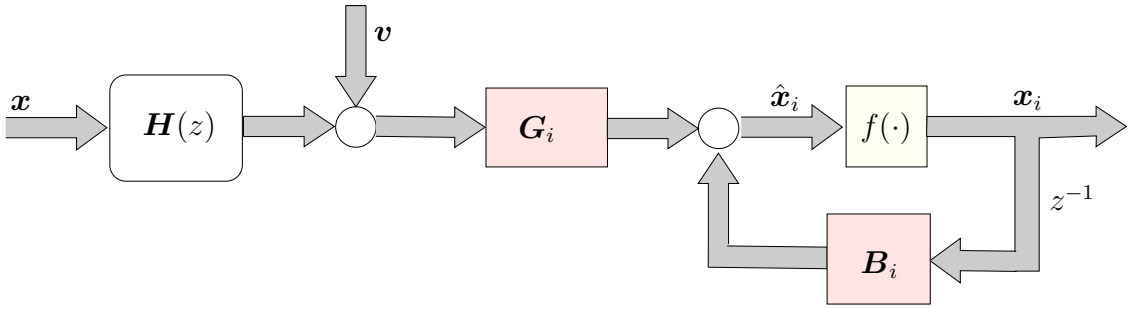


Figure 2.2: Kalman-like DFE estimation.

Now, the channel matrix in question represents in general an undertermined system, and for arbitrary signals and threshold functions $f(\cdot)$, an iterative attempt to estimate the transmitted sequence might not succeed. This is because such recursive estimation employs a constant fat matrix $\boldsymbol{H}$, i.e., assumed full row-rank, in that $\widehat{\boldsymbol{P}}_i$ is ill-conditioned. It is also the case when $\mathbf{x}$ belongs to a signal constellation, so that selecting $f(\cdot)$ as a commonly used constellation 'slicer' would still suffer from the same impairment. Moreover, as we have mentioned, given (2.19), we would still need to find a recursive relation between the variances $\{\boldsymbol{P}_i, \widehat{\boldsymbol{P}}_i\}$ so that these quantities are duly propagated. In order to further shed light into the deterministic formulation of (2.13) and its shortcomings, we examine its analogous stochastic formulation.

2.2 Stochastic Problem and the Relation to Iterative DFE

We now resort to the equivalence between the deterministic cost of (2.10) and its stochastic (smoothing) counterpart,

$$\min_{\boldsymbol{K}} \mathsf{E}\, \|\mathbf{x} - \boldsymbol{K}\mathbf{y}_i\|^2 \ , \tag{2.22}$$

where $\mathbf{y}_i$ is defined in (2.7) and

$$\mathsf{E}\mathbf{x}\mathbf{x}^* = \boldsymbol{\Pi}\,, \quad \mathsf{E}\mathbf{x} = \mathbf{x}_{i-1} = f_i(\hat{\mathbf{x}}_{i-1})\,,$$

with $f_i(\cdot)$ allowed to be time-varying. Let the innovations vector of estimation errors be defined as

$$\mathbf{e}_i = \mathsf{col}\{\bar{\mathbf{e}}_0, \bar{\mathbf{e}}_1, \ldots, \bar{\mathbf{e}}_i\} \tag{2.23}$$

Given the model in (2.3), the solution to this problem is equivalent to the one that replaces $\mathbf{y}_i$ with $\mathbf{e}_i$, and is given by

$$\hat{\mathbf{x}}_i = \boldsymbol{R}_{\mathbf{x}\mathbf{e}_i}\boldsymbol{R}_{\mathbf{e}_i}^{-1}\mathbf{e}_i \tag{2.24}$$

$$= \sum_{k=0}^{i} \boldsymbol{R}_{\mathbf{x}\mathbf{e}_k}\boldsymbol{R}_{\bar{e},k}^{-1}\bar{\mathbf{e}}_k \tag{2.25}$$

where $\mathbf{x}_i$ is easily seen to be recursively computed as an approximation, by projecting the estimate $\hat{\mathbf{x}}_i$ onto the constellation, as

$$\bar{\mathbf{e}}_i = \mathbf{y} - \boldsymbol{H}\mathbf{x}_{i-1} \tag{2.26}$$

$$\hat{\mathbf{x}}_i = \mathbf{x}_{i-1} + \boldsymbol{R}_{\mathbf{x}\mathbf{e}_i}\boldsymbol{R}_{\bar{e},i}^{-1}\bar{\mathbf{e}}_i \tag{2.27}$$

$$\mathbf{x}_i = f_i(\hat{\mathbf{x}}_i) \tag{2.28}$$

with

$$\boldsymbol{R}_{\mathbf{x}\mathbf{e}_i} = \mathsf{E}\,\mathbf{x}\bar{\mathbf{e}}_i^* = \mathsf{E}\,\mathbf{x}(\boldsymbol{H}\bar{\mathbf{x}}_{i-1} + \mathbf{v})^* = (\mathsf{E}\mathbf{x}\bar{\mathbf{x}}_{i-1}^*)\boldsymbol{H}^* \tag{2.29}$$

$$\boldsymbol{R}_{\bar{e},i} = \mathsf{E}\,\bar{\mathbf{e}}_i\bar{\mathbf{e}}_i^* = \mathsf{E}\,(\boldsymbol{H}\bar{\mathbf{x}}_{i-1} + \mathbf{v})(\boldsymbol{H}\bar{\mathbf{x}}_{i-1} + \mathbf{v})^* = \boldsymbol{H}\boldsymbol{P}_{i-1}\boldsymbol{H}^* + \boldsymbol{R}_{\mathbf{v}} \tag{2.30}$$

We may note that the quantitiy $(\mathsf{E}\mathbf{x}\bar{\mathbf{x}}_{i-1}^*)$ in (2.29) can be expressed as

$$\mathsf{E}\mathbf{x}\bar{\mathbf{x}}_{i-1}^* = \mathsf{E}\mathbf{x}(\mathbf{x} - \mathbf{x}_{i-1})^* \tag{2.31}$$

$$= \boldsymbol{\Pi} - \boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} \tag{2.32}$$

where

$$\boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} \triangleq \mathsf{E}\mathbf{x}\mathbf{x}_{i-1}^*\,. \tag{2.33}$$

We shall also define

$$\boldsymbol{R}_{\mathbf{x}_i} \triangleq \mathsf{E}\mathbf{x}_i\mathbf{x}_i^* \tag{2.34}$$

for later reference. Hence,

$$\mathsf{E}\mathbf{x}\bar{\mathbf{x}}_{i-1}^* = \mathsf{E}(\bar{\mathbf{x}}_{i-1} + \mathbf{x}_{i-1})\bar{\mathbf{x}}_{i-1}^* \tag{2.35}$$

$$= \mathsf{E}\bar{\mathbf{x}}_{i-1}\bar{\mathbf{x}}_{i-1}^* + \mathsf{E}\mathbf{x}_{i-1}\bar{\mathbf{x}}_{i-1}^* \tag{2.36}$$

$$= \boldsymbol{P}_{i-1} + \mathsf{E}\mathbf{x}_{i-1}\bar{\mathbf{x}}_{i-1}^* \tag{2.37}$$

Recall that when $\mathbf{x}_i = f_i(\hat{\mathbf{x}}_i) = \hat{\mathbf{x}}_i$, the orthogonality principle naturally yields $\mathsf{E}\mathbf{x}_{i-1}\bar{\mathbf{x}}_{i-1}^* = 0$, in which case the recursions collapse to the Kalman filter. In general, however, this relation does not hold, i.e., given $f_i(\cdot)$, we have

$$\mathsf{E}\, f_i(\hat{\mathbf{x}}_i)[\mathbf{x} - f_i(\hat{\mathbf{x}}_i)] \neq 0 \tag{2.38}$$

and a true stochastic algorithm would require that both $\boldsymbol{P}_{i-1}$ and $\left(\mathsf{E}\mathbf{x}_{i-1}\bar{\mathbf{x}}_{i-1}^*\right)$ be properly propagated. Nevertheless, if $f_i(\cdot)$ is chosen as the minimizer of

$$\min_{f_i(\cdot)} \mathsf{E}\bar{\mathbf{x}}_i\bar{\mathbf{x}}_i^* \tag{2.39}$$

then,

$$\mathbf{x}_i = \mathsf{E}(\mathbf{x}|\hat{\mathbf{x}}_i) = \int_{\mathcal{S}_\mathbf{x}} \mathbf{x} p(\mathbf{x}|\hat{\mathbf{x}}_i)\mathsf{d}\mathbf{x} \tag{2.40}$$

in terms of the conditional probability $p(\cdot)$, with $\mathcal{S}_\mathbf{x}$ denoting the support of $\mathbf{x}$, so that

$$\mathsf{E}\,\mathbf{x}_i\bar{\mathbf{x}}_i^* = 0$$

and Eqs. (2.14)–(2.19) will follow. That is, the solution based on the innovations (2.25) becomes

$$\hat{\mathbf{x}}_i = \sum_{k=0}^{i} \boldsymbol{P}_{k-1}\boldsymbol{H}^*(\boldsymbol{H}\boldsymbol{P}_{k-1}\boldsymbol{H}^* + \boldsymbol{R}_\mathbf{v})^{-1}\bar{\mathbf{e}}_k \tag{2.41}$$

$$= \sum_{k=0}^{i} (\boldsymbol{P}_{k-1}^{-1} + \boldsymbol{H}^*\boldsymbol{R}_\mathbf{v}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^*\boldsymbol{R}_\mathbf{v}^{-1}\bar{\mathbf{e}}_k \tag{2.42}$$

$$= \mathbf{x}_{i-1} + (\boldsymbol{P}_{i-1}^{-1} + \boldsymbol{H}^*\boldsymbol{R}_\mathbf{v}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^*\boldsymbol{R}_\mathbf{v}^{-1}\bar{\mathbf{e}}_i \tag{2.43}$$

In other words, we obtain $(\mathsf{E}\mathbf{x}\bar{\mathbf{x}}_{i-1}^*) = \boldsymbol{P}_{i-1}$ in (2.29), and the recursions will assume the form of a Kalman filter followed by the optimal $f_i(\cdot)$ that minimizes (2.39).

From what we have discussed, the optimality on the choice of $f_i(\cdot)$ is a consequence of the orthogonality condition in (2.38), which can be achieved by using the probabilistic model that generates $\hat{\mathbf{x}}_i$ from $\mathbf{x}$, and upon minimization of (2.39). The exact optimal estimator of $\mathbf{x}$ given $\mathbf{y}$ can be extremely complex for general vector sizes, and therefore, the exact $f_i(\cdot)$ that minimizes (2.39) is intractable. Still, when the noise p.d.f. is Gaussian, a maximum a posteriori (MAP) criterion optimally designed for symbol estimation justifies the use of a slicer in the ℓ_2-norm sense, case of the majority of practical applications involving DFEs. The success in detection is, however, still highly dependent on the structure of $\boldsymbol{H}$, which in

case of underdetermined systems, turns detection into an impossible task (specially if approached with conventional receivers and/or slicers). This is further aggravated for "bad" channels, so that the inherent Toeplitz structure of $\boldsymbol{H}$ makes this matrix highly ill-conditioned.

In the following, our goal is to pave the way to a broader concept of IBI estimation and cancellation, by first showing how conventional block DFEs build up exactly from LS estimation problem, and that, except for a fixed slicer, the DFE structure naturally arises without imposing its structure a priori. Moreover, the approach will suggest an extension of the conventional DFE formulas employing memory — see e.g., [50] — to a general block scenario also with memory, carrying 2 important features:

1. Transmission of block sizes with arbitrary length M, possibly smaller than the channel span. More generally, for $M < L$, block equalization aims the recovery of $\boldsymbol{s}_{n-\delta}$, where δ is referred to as the decision-delay associated to the subblock $\boldsymbol{\mathcal{H}}$ of the matrix $\boldsymbol{H}$ exhibiting the best conditioning, while $\boldsymbol{s}_{n-k}$ for $k \neq \delta$ is considered IBI.

2. We show that the structure in Fig. 2.2 can be cast as a special case of a block DFE with memory equipped with an iterative procedure, similar in spirit to the ones employed by iterative DFEs. Moreover, the conventional DFE turns out to represent a single iteration of such iterative scheme, which assumes perfect detection of past IBI delayed blocks of $\boldsymbol{s}_{n-\delta}$, when $M = P$.

2.3 Block Linear Equalization Revisited

The conventional approach to deal with the underdeterminacy in a given channel model that deals with a 'fat' matrix $\boldsymbol{H}$, is an attempt to design feedfoward and feedback matrices in the MMSE sense, by first removing IBI through subtraction of past detected (block) symbols (easily extended to the MIMO case). As we have shown, this is similar to the role of the Kalman variables $\{\boldsymbol{K}_{p,i}, \boldsymbol{F}_{p,i}\}$ obtained exactly from a deterministic cost, whenever a quantizer (slicer) that makes perfect decisions over the signal constellation is employed — see Fig. 2.2.

The fundamental difference with respect to the previous discussion relates to the target portion of the vector $\mathbf{x}$ we wish to estimate, which can be reduced by selecting the corresponding set of columns of $\boldsymbol{H}$ and considering the remaining entries as nuisance. Next, we readdress the DFE problem under the light of the iterative block estimation scenario aforementioned, and more importantly, without assuming its structure a priori.

Let

$$\boldsymbol{x}_\delta = [\, x(n-\delta) \quad x(n-\delta-1) \;\cdots\; x(n-\delta-M+1)\,]^{\mathsf{T}} \tag{2.44}$$

be the transmitted block of interest within $\mathbf{x}$, where δ is the decision-delay to be suitably chosen, and consider the following partitionings:

$$\boldsymbol{H} \;=\; \left[\begin{array}{ccc} \boldsymbol{H}_\delta & \boldsymbol{\mathcal{H}} & \overline{\boldsymbol{H}} \end{array}\right] \;=\; \left[\begin{array}{cc} \boldsymbol{H}' & \overline{\boldsymbol{H}} \end{array}\right] \tag{2.45}$$

$$\mathbf{x} \;=\; \left[\begin{array}{ccc} \mathbf{x}_f^{\mathsf{T}} & \boldsymbol{x}_\delta^{\mathsf{T}} & \mathbf{x}_b^{\mathsf{T}} \end{array}\right]^{\mathsf{T}} \;=\; \left[\begin{array}{cc} \mathbf{x}'^{\,\mathsf{T}} & \mathbf{x}_b^{\mathsf{T}} \end{array}\right]^{\mathsf{T}} \tag{2.46}$$

with block dimensions defined as follows:

$$\boldsymbol{H}_\delta \;:\; P \times \delta \tag{2.47}$$

$$\boldsymbol{\mathcal{H}} \;:\; P \times M \tag{2.48}$$

$$\overline{\boldsymbol{H}} \;:\; P \times (P+L-M-\delta-1) \tag{2.49}$$

and where we defined $\mathbf{x}' = [\mathbf{x}_f^{\mathsf{T}} \quad \boldsymbol{x}_\delta^{\mathsf{T}}\,]^{\mathsf{T}}$. The trailing vector $\mathbf{x}_b$ is $(P+L-M-\delta-1)\times 1$, and consists of past transmitted symbols with the general form

$$\mathbf{x}_b = [x(n-\delta-M) \;\cdots\; x(P+L-M-\delta-2)]^{\mathsf{T}}$$

Note that the delay δ is not necessarily a multiple of the transmitted block size M (and hence the notation $\boldsymbol{\mathcal{H}}$ instead of $\boldsymbol{\mathcal{H}}_\delta$ in general). The received block in (2.3) can thus be equivalently written as

$$\mathbf{y} \;=\; \boldsymbol{\mathcal{H}}\boldsymbol{x}_\delta \;+\; \boldsymbol{H}_\delta \mathbf{x}_f \;+\; \overline{\boldsymbol{H}}\mathbf{x}_b \;+\; \boldsymbol{v} \tag{2.50}$$

$$\;=\; \boldsymbol{\mathcal{H}}\boldsymbol{x}_\delta \;+\; \boldsymbol{v}_c \tag{2.51}$$

Now, define the estimation errors

$$\tilde{\boldsymbol{x}}_{\delta,i} \;\triangleq\; \boldsymbol{x}_\delta - \hat{\boldsymbol{x}}_{\delta,i} \quad \text{with variance} \quad \widehat{\boldsymbol{P}}_i = \mathsf{E}\,\tilde{\boldsymbol{x}}_{\delta,i}\tilde{\boldsymbol{x}}_{\delta,i}^* \tag{2.52}$$

$$\bar{\boldsymbol{x}}_{\delta,i} \;\triangleq\; \boldsymbol{x}_\delta - \boldsymbol{x}_{\delta,i} \quad \text{with variance} \quad \boldsymbol{P}_{\delta,i} = \mathsf{E}\,\bar{\boldsymbol{x}}_{\delta,i}\bar{\boldsymbol{x}}_{\delta,i}^* \tag{2.53}$$

corresponding to the LS estimate and detected symbols (after the slicer) respectively. Also, let

$$\check{\mathbf{x}}_b \;=\; [x_m(n-\delta-M) \;\cdots\; x_m(n-P-L+M+\delta+2)]^{\mathsf{T}} \tag{2.54}$$

$$\bar{\mathbf{x}}_b \;=\; [\bar{x}_m(n-\delta-M) \;\cdots\; \bar{x}_m(n-P-L+M+\delta+2)]^{\mathsf{T}} \tag{2.55}$$

denote the previously detected vector and final estimation errors, say, after $i = m$

iterations, so that

$$\boldsymbol{P}_{\mathbf{x}_b} = \mathsf{E}\bar{\mathbf{x}}_b\bar{\mathbf{x}}_b^* \tag{2.56}$$

The above model can thus be centralized, by removing the contribution of $\check{\mathbf{x}}_b$ from $\mathbf{y}$, as

$$\begin{aligned}
\mathbf{y}' &= \mathbf{y} - \overline{\boldsymbol{H}}\check{\mathbf{x}}_b && (2.57)\\
&= \boldsymbol{\mathcal{H}}\boldsymbol{x}_\delta \;+\; \boldsymbol{H}_\delta\mathbf{x}_f \;+\; \overline{\boldsymbol{H}}\bar{\mathbf{x}}_b \;+\; \boldsymbol{v} && (2.58)\\
&= \boldsymbol{\mathcal{H}}\boldsymbol{x}_\delta \;+\; \boldsymbol{v}_c && (2.59)
\end{aligned}$$

where $\boldsymbol{v}_c \triangleq \boldsymbol{H}_\delta\mathbf{x}_f \;+\; \overline{\boldsymbol{H}}\bar{\mathbf{x}}_b + \boldsymbol{v}$ is regarded as noise, with variance given by

$$\begin{aligned}
\boldsymbol{R}_{\boldsymbol{v}_c} = \mathsf{E}\boldsymbol{v}_c\boldsymbol{v}_c^* &= \mathsf{E}(\boldsymbol{H}_\delta\mathbf{x}_f \;+\; \overline{\boldsymbol{H}}\bar{\mathbf{x}}_b + \boldsymbol{v})(\boldsymbol{H}_\delta\mathbf{x}_f \;+\; \overline{\boldsymbol{H}}\bar{\mathbf{x}}_b + \boldsymbol{v})^* && (2.60)\\
&= \boldsymbol{H}_\delta\mathsf{E}\mathbf{x}_f\mathbf{x}_f^*\boldsymbol{H}_\delta^* \;+\; \overline{\boldsymbol{H}}\mathsf{E}\bar{\mathbf{x}}_b\bar{\mathbf{x}}_b^*\overline{\boldsymbol{H}}^* \;+\; \boldsymbol{R}_{\boldsymbol{v}} && (2.61)\\
&= \boldsymbol{H}_\delta\boldsymbol{R}_{\mathbf{x}_f}\boldsymbol{H}_\delta^* \;+\; \overline{\boldsymbol{H}}\,\boldsymbol{P}_{\mathbf{x}_b}\overline{\boldsymbol{H}}^* \;+\; \boldsymbol{R}_{\boldsymbol{v}} && (2.62)
\end{aligned}$$

We are now ready to apply a recursive procedure for symbol estimation which employs the Kalman recursions. Thus, let $f(\cdot)$ be a slicer that quantizes $\hat{\boldsymbol{x}}_{\delta,i}$. By making the following identifications,

$$\begin{aligned}
\boldsymbol{H} &\longleftarrow \boldsymbol{\mathcal{H}}\,, \qquad \boldsymbol{v} \longleftarrow \boldsymbol{v}_c\,, \qquad \boldsymbol{R}_{\boldsymbol{v}} \longleftarrow \boldsymbol{R}_{\boldsymbol{v}_c} && (2.63)\\
\mathbf{x}_i &\longleftarrow \boldsymbol{x}_{\delta,i}\,, \qquad \mathbf{y} \longleftarrow \mathbf{y}' \qquad \boldsymbol{P}_i \longleftarrow \boldsymbol{P}_{\delta,i} && (2.64)
\end{aligned}$$

we obtain

$$\begin{aligned}
\boldsymbol{R}_{\bar{e},i} &= \boldsymbol{R}_{\boldsymbol{v}} + \boldsymbol{H}_\delta\boldsymbol{R}_{\mathbf{x}_f}\boldsymbol{H}_\delta^* + \overline{\boldsymbol{H}}\boldsymbol{P}_{\mathbf{x}_b}\overline{\boldsymbol{H}}^* + \boldsymbol{\mathcal{H}}\boldsymbol{P}_{\delta,i-1}\boldsymbol{\mathcal{H}}^* && (2.65)\\
\boldsymbol{K}_{p,i} &= \boldsymbol{P}_{\delta,i-1}\boldsymbol{\mathcal{H}}^*\boldsymbol{R}_{\bar{e},i}^{-1} && (2.66)\\
\bar{\mathbf{e}}_i &= \mathbf{y}' - \boldsymbol{\mathcal{H}}\boldsymbol{x}_{\delta,i-1} && (2.67)\\
\hat{\boldsymbol{x}}_{\delta,i} &= \boldsymbol{x}_{\delta,i-1} + \boldsymbol{K}_{p,i}\bar{\mathbf{e}}_i\,, \qquad \boldsymbol{x}_{\delta,0} = \mathbf{0} && (2.68)\\
\boldsymbol{x}_{\delta,i} &= f(\hat{\boldsymbol{x}}_{\delta,i}) && (2.69)\\
\widehat{\boldsymbol{P}}_{\delta,i} &= \boldsymbol{P}_{\delta,i-1} - \boldsymbol{K}_{p,i}\boldsymbol{R}_{\bar{e},i}\boldsymbol{K}_{p,i}^*\,, \qquad \text{with } \boldsymbol{P}_{\delta,0} = \boldsymbol{R}_{\boldsymbol{x}_\delta} && (2.70)
\end{aligned}$$

Observe that these recursions tell us how to update the quantized estimate $\boldsymbol{x}_{\delta,i}$ by using the variance $\boldsymbol{P}_i$, so we would still need to know how to perform the update $\widehat{\boldsymbol{P}}_i \longrightarrow \boldsymbol{P}_i$. Figure 2.3 illustrates the resulting block diagram.

In order to show that these equations can be written in a familiar DFE form, we simply express $\boldsymbol{x}_{\delta,i}$ as in Eq. (2.20). That is, combining (2.67) and (2.68), and using

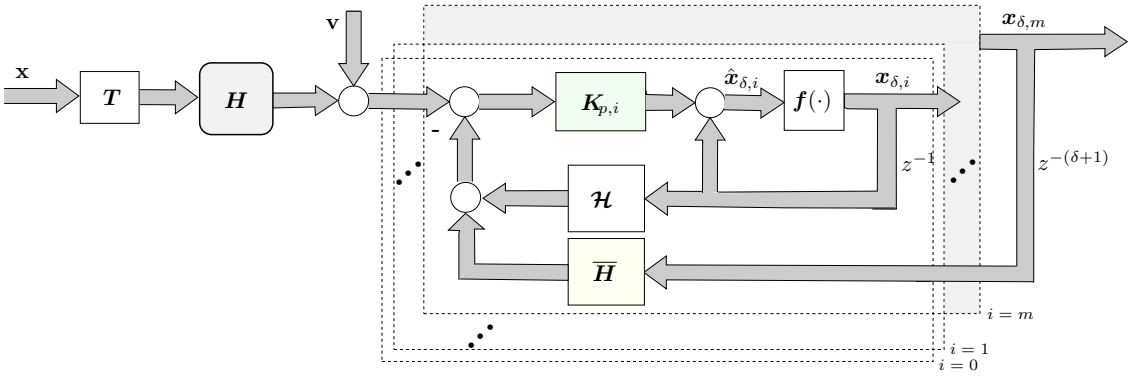


Figure 2.3: Generalized Kalman DFE estimation.

(2.57), $\boldsymbol{x}_{\delta,i}$ can be alternatively written as

$$\begin{aligned} \boldsymbol{x}_{\delta,i} &= f\left(\boldsymbol{K}_{p,i}\mathbf{y} - \boldsymbol{K}_{p,i}\overline{\boldsymbol{H}}\check{\mathbf{x}}_b - (\boldsymbol{K}_{p,i}\boldsymbol{\mathcal{H}} - \boldsymbol{I})\boldsymbol{x}_{\delta,i-1}\right) && (2.71)\\ &= f(\boldsymbol{G}_i\mathbf{y} - \boldsymbol{B}_i\check{\mathbf{x}}_b - \boldsymbol{B}_i'\boldsymbol{x}_{\delta,i-1}) && (2.72) \end{aligned}$$

where we have defined

$$\boldsymbol{G}_i \triangleq \boldsymbol{K}_{p,i}, \quad \boldsymbol{B}_i \triangleq \boldsymbol{G}_i\overline{\boldsymbol{H}}, \quad \text{and} \quad \boldsymbol{B}_i' \triangleq \boldsymbol{G}_i\boldsymbol{\mathcal{H}} - \boldsymbol{I}$$

Assume that δ is a multiple of the block size M. Note that for $M < P$, the matrix inner product $\boldsymbol{B}_i\check{\mathbf{x}}_b$ can be written as $\sum_{\ell=\delta+1}^{L_B-1}\boldsymbol{\mathfrak{B}}_{\ell,i}\,\boldsymbol{x}_{n-\ell,m}$, so that we can associate a (time-varying) transfer function $\boldsymbol{B}_i(z)$ to the matrix coefficients $\{\boldsymbol{\mathfrak{B}}_{\ell,i}\}$ in $\boldsymbol{B}_i$. Similarly, the output of $\mathbf{y}$ can be divided into blocks of size, say, Q, as $\mathbf{y} = [\,\boldsymbol{y}_n^{\mathsf{T}} \quad \boldsymbol{y}_{n-1}^{\mathsf{T}} \quad \cdots \quad \boldsymbol{y}_{n-P/Q}^{\mathsf{T}}\,]^{\mathsf{T}}$, and its inner product with $\boldsymbol{G}_i$ written as $\sum_{\ell=0}^{P/Q}\boldsymbol{\mathfrak{G}}_{\ell,i}\,\boldsymbol{y}_{n-\ell}$. In this manner, we associate a matrix transfer function $\boldsymbol{G}_i(z)$ to the matrix coefficients in $\{\boldsymbol{\mathfrak{G}}_{\ell,i}\}$ in $\boldsymbol{G}_i$ as well. The equivalent of Fig. 2.3 is shown in Fig. 2.4.

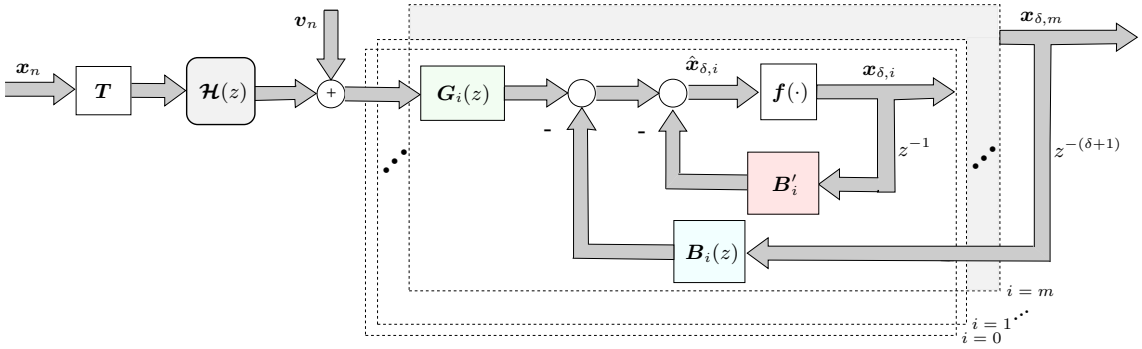


Figure 2.4: Equivalent DFE architecture.

To appreciate the connection of this recursions with a conventional DFE, assume

that past symbols have been correctly detected. This implies $\boldsymbol{P}_{\mathbf{x}_b} = \mathbf{0}$, so that

$$\boldsymbol{K}_{p,i} = \boldsymbol{R}_{\boldsymbol{x}}\boldsymbol{\mathcal{H}}^*(\boldsymbol{\mathcal{H}}\boldsymbol{P}_{\delta,i-1}\boldsymbol{\mathcal{H}}^* + \boldsymbol{H}_\delta\boldsymbol{R}_{\mathbf{x}_f}\boldsymbol{H}_\delta^* + \boldsymbol{R}_{\boldsymbol{v}})^{-1}$$

If we initialize this recursions with $\boldsymbol{x}_{\delta,0} = \mathbf{0}$ and $\boldsymbol{P}_{\delta,0} = \boldsymbol{R}_{\boldsymbol{x}_\delta} = \boldsymbol{R}_{\boldsymbol{x}}$, and apply a constellation slicer after a single iteration, then (2.71) collapses to

$$\boldsymbol{x}_{\delta,1} \;=\; f(\boldsymbol{K}_{p,1}\mathbf{y} - \boldsymbol{K}_{p,1}\overline{\boldsymbol{H}}\check{\mathbf{x}}_b) \tag{2.73}$$

Hence, by defining

$$\boldsymbol{R}_{\mathbf{x}'} = \mathsf{E}\mathbf{x}'\mathbf{x}'^* = \mathsf{Diag}(\boldsymbol{R}_{\mathbf{x}_f}, \boldsymbol{R}_{\mathbf{x}_\delta}) \tag{2.74}$$

we get

$$\begin{cases} \boldsymbol{G}_1 = \boldsymbol{R}_{\boldsymbol{x}}\boldsymbol{\mathcal{H}}^*(\boldsymbol{H}'\boldsymbol{R}_{\mathbf{x}'}\boldsymbol{H}'^* + \boldsymbol{R}_{\boldsymbol{v}})^{-1} \\ \boldsymbol{B}_1 = \boldsymbol{G}\overline{\boldsymbol{H}} \end{cases} \tag{2.75}$$

which are *the exact well known expressions for the feedforward and feedback DFE coefficient matrices obtained in the literature* [12, 24]. Here, however, they were seen as a special case of a single iteration of an algorithm obtained in more general grounds. In other words, the DFE represents an algorithm in its own right, and without assuming its structure a priori. This procedure may continue until a more accurate estimate $\boldsymbol{x}_{\delta,i}$ is obtained, with corresponding LS error variance propagated by (2.70).

2.4 Reduced Complexity Widely-Linear BI-DFE

From the basics of complex random processes, it is well established that in order to completely characterize the second-order statistics of a signal, one must specify both its related covariance and pseudo-covariance functions. The particular case where only its covariance information suffices occurs when the underlying process is already *circular*, since in this situation, its pseudo-covariance becomes naturally zero [14, 25, 26]. In this respect, *Widely Linear* (WL) formulations of parameter estimation have been highly motivated, and offer significant gains in performance by estimating an augmented vector comprising both the original data, and its conjugate. In contrast to the more common form of *Strictly Linear* (SL) estimation, the WL approach enables us to capture both covariance and pseudo-covariance information, minimizing the MSE when the traditional circularity or *properness* assumption in the data no longer holds. For example, the complex envelope found in some modulation schemes such as M-ary amplitude shift keying (ASK), binary phase shift

keying (BPSK), offset quadrature phase shift keying (OQPSK) and minimum-shift keying (MSK), constitute some important examples of improper signals.

It is noteworthy that improperness arises in several other instances with regard to signaling, as well as noise and/or interference. This is the case of DS-CDMA systems with improper complex constellations or due to the use of iterative multiuser receivers. Beamformers for the extraction of an unknown signal from non-circular interferences is another instance investigated in [25]. Transmitters and receivers with in-phase and quadrature (IQ) imbalances can also benefit from WL formulations. Some space-time block codes may also result in improper complex signals. In multicarrier systems, improper narrowband signals in the form of overlay networks or due to crosstalk or radio frequency interferences may also appear. In the latter, the output of a baseband OFDM system may also contain an improper signal, as a result of colored noise at the channel output.

Despite the gains in performance with respect to the SL approach in these scenarios, the original WL approach has a potential drawback, in that the complex WL estimate is double the size of the traditional SL vector. This implies not only an increase in computational complexity and excess MSE involved, but also a reduction in convergence speed, in case LMS-like iterative procedures are employed. Such caveats have hindered the use of the original WL formulation in a number of applications that require at least the same complexity used by SL solutions.

Recently, as been pointed out by [25], an equivalent formulation of the WL approach with reduced complexity can be achieved, by replacing the parameter vector estimate with a real vector, comprising the real and the imaginary parts of the target complex data. With this simple modification, the redundant second-order information is eliminated from the vector autocorrelation matrix, and the computational complexity of the WL filters becomes similar to the one of SL filters. That is, the modified filters are equivalent to their standard WL counterparts, in addition to providing reduced complexity.

Motivated by these benefits, in this section we extend the iterative estimation architectures commonly obtained in a SL scenario to a WL formulation. One particular advantage of these formulations is that in the case of real constellations, the BER performance and/or throughput can be considerably improved, specially with regard to transceivers that employ zero redundancy, as we shall see.

Widely Linear BI-DFE formulas can be obtained by redefining the channel model in (2.3) in terms of extended quantities, starting by expressing the received signal as an augmented vector $\mathbf{y}_e$, comprising the real and imaginary parts of its complex

counterpart $\mathbf{y}$:

$$\mathbf{y}_e = \begin{bmatrix} \Re(\mathbf{y}) \\ \Im(\mathbf{y}) \end{bmatrix} \tag{2.76}$$

In this way, the channel model now reads

$$\begin{aligned} \mathbf{y}_e &= \begin{bmatrix} \boldsymbol{H}_R & -\boldsymbol{H}_I \\ \boldsymbol{H}_I & \boldsymbol{H}_R \end{bmatrix} \begin{bmatrix} \mathbf{x}_R \\ \mathbf{x}_I \end{bmatrix} + \begin{bmatrix} \mathbf{v}_R \\ \mathbf{v}_I \end{bmatrix} \\ &= \boldsymbol{H}_e \mathbf{x}_e + \mathbf{v}_e, \end{aligned} \tag{2.77}$$

where, accordingly, we have now defined extended input and noise vectors $\mathbf{x}_e$ and $\mathbf{v}_e$, as well as the extended channel matrix $\boldsymbol{H}_e$ implied by (2.77), with

$$\mathbf{x}_R \triangleq \Re(\mathbf{x}), \;\; \mathbf{x}_I \triangleq \Im(\mathbf{x}), \;\; \mathbf{v}_R \triangleq \Re(\mathbf{v}), \;\; \mathbf{v}_I \triangleq \Im(\mathbf{v}), \;\; \boldsymbol{H}_R \triangleq \Re(\boldsymbol{H}), \;\; \boldsymbol{H}_I \triangleq \Im(\boldsymbol{H}) \tag{2.78}$$

Note that in the case of real constellations such as BPSK or PAM, (2.77) simplifies to

$$\mathbf{y}_e = \begin{bmatrix} \boldsymbol{H}_R \\ \boldsymbol{H}_I \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{v}_R \\ \mathbf{v}_I \end{bmatrix} \tag{2.79}$$

which suggests better conditioning in any estimation mechanism, as a result of the now tall matrix transmission matrix $\boldsymbol{H}_e$.

Moreover, because of the block (2×2) pseudocirculant structure of $\boldsymbol{H}_e$, it is easy to verify that it admits the following factorization:

$$\boldsymbol{H}_e = (\mathbf{D}_{-1} \otimes \boldsymbol{I}_P)^* (\boldsymbol{F} \otimes \boldsymbol{I}_P)^* \boldsymbol{\Lambda} (\boldsymbol{F} \otimes \boldsymbol{I}_Q)(\mathbf{D}_{-1} \otimes \boldsymbol{I}_Q) \tag{2.80}$$

$$= (\mathbf{D}_{-1}\boldsymbol{F} \otimes \boldsymbol{I}_P)^* \boldsymbol{\Lambda} (\boldsymbol{F}\mathbf{D}_{-1} \otimes \boldsymbol{I}_Q) \tag{2.81}$$

where $\mathbf{D}_{-1} = (1 \oplus j)$ (to be defined more generally in (2.110), for $M = 2$ and $\phi = -1$), $\boldsymbol{F}$ is a 2×2 DFT matrix, and

$$\boldsymbol{\Lambda} = \begin{bmatrix} \boldsymbol{H} & \\ & (\boldsymbol{H}^*)^{\mathsf{T}} \end{bmatrix} \tag{2.82}$$

defined in terms of the complex-valued $\boldsymbol{H}$. Note that we can write (2.81) more compactly as

$$\boldsymbol{H}_e = \mathbf{U}_P^* \boldsymbol{\Lambda} \mathbf{U}_Q \tag{2.83}$$

where

$$\mathbf{U}_Q \;=\; \boldsymbol{F}\mathbf{D}_{-1} \otimes \boldsymbol{I}_Q \;=\; \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{I}_Q & j\boldsymbol{I}_Q \\ \boldsymbol{I}_Q & -j\boldsymbol{I}_Q \end{bmatrix} \tag{2.84}$$

and with $\mathbf{U}_P$ defined accordingly. Also, multiplying $\mathbf{y}_e$ by $\mathbf{U}_P$, we get a linear model description in terms of the original WL formulation (i.e., in terms of augmented vectors comprising the data and its conjugate):

$$\mathbf{y}' \;=\; \boldsymbol{\Lambda}\mathbf{x}' + \mathbf{v}' \tag{2.85}$$

where $\mathbf{y}' = \mathbf{U}_P\mathbf{y}_e$, $\mathbf{x}' = \mathbf{U}_Q\mathbf{x}_e$, and $\mathbf{v}' = \mathbf{U}_P\mathbf{v}$.

2.5 Optimal Decision-Delay

At this point, an important question concerning the optimality of the above scheme is how to select the optimal decision-delay δ so as to maximize detection performance. To this end, intuition tells us that since our optimal estimator is based on the linear model (2.59), we must select δ in a way that $\boldsymbol{R}_{\boldsymbol{v}_c}$ contains a small contribution through the blocks $\boldsymbol{H}_\delta$ and $\overline{\boldsymbol{H}}$ — and at the same time, allowing $\boldsymbol{\mathcal{H}}$ to capture the best conditioned block within $\boldsymbol{H}$. Of course, if $\check{\mathbf{x}}_b$ has been accurately estimated, its contribution to $\boldsymbol{R}_{\boldsymbol{v}_c}$ through $\overline{\boldsymbol{H}}$ will be negligible; For instance, when dealing with minimum-phase channels, it makes sense to set $\delta = 0$. Still, in order for it to be accurately estimated in the previous block, the same reasoning must apply to the choice of δ over that block. Several criteria can be considered when selecting the optimal δ. One solution can be achieved by choosing the delay that minimizes the norm of the gain matrix

$$\min_{\delta} \|\boldsymbol{K}_{p,i}\|, \tag{2.86}$$

thus reducing the noise and interference amplification. From the viewpoint of signal processing, and considering the Toeplitz structured of $\boldsymbol{H}$, the solution to this problem is due to a famous paper by Scaglione et al. [16], where it has been shown (according to their particular equalization context) to be approximately given by

$$\delta_{\text{opt}} \;=\; \#\text{ maximum-phase zeros of } H(z) \tag{2.87}$$

building on a reasoning that yielded the scalar counterpart solution (see the references therein). We shall see further ahead how this results follow as a special case, when the columns of $\boldsymbol{H}$ are selected arbitrarily, so that the corresponding $\|\boldsymbol{K}_{p,i}\|$ is minimized.

The block DFE recursions along with the above criterion for selecting the optimal

subblock $\boldsymbol{\mathcal{H}}$, generalizes the concept of ordered successive cancellation, commonly encountered in communications, whenever the entries of the received block (which is IBI-free) are detected serially. In the latter, instead of attempting to estimate an entire optimal block $\boldsymbol{x}_\delta$ of size M in a single shot, the entries of $\boldsymbol{x}_\delta$ can be estimated successively, in exactly M outer loops, until the entire vector is recovered. The order of detection within the vector $\boldsymbol{x}_\delta$ can in turn follow the same optimality criterion considered when selecting the optimal subblock $\boldsymbol{\mathcal{H}}$, e.g., (2.86). Successive cancellation is a safer choice when it comes to feeding back past detected symbols, although resulting in a more complex implementation.

The interpretation of this procedure in light of the above iterative scheme is the following. Assume that a single iteration for each symbol is sufficient for reliable detection. As a consequence, note that not only the corresponding column of each symbol can be completely extracted from the received signal, but IBI can also be considered perfectly removed. Suppose otherwise that IBI is not removed, and that $\mathbf{h}_\ell = \boldsymbol{h}_{j_\ell}$, $\ell = 0, 1, \ldots, M-1$ denote the j-th column of $\boldsymbol{H}$, ordered according to some criterion. This means that $\bar{\boldsymbol{x}}_{i-1}$ will have zero entries at the corresponding perfect decisions (the same reasoning can be applied if we replace $\boldsymbol{H}$ with $\boldsymbol{\mathcal{H}}$, and eliminate IBI first, with $\boldsymbol{P}_{\mathbf{x}_b} = 0$), and $\boldsymbol{P}_{\delta,i-1}$ will also have null rows and columns at the same indexes. Denote by $\boldsymbol{H}_\ell$ the matrix that contains the remaining columns of $\boldsymbol{H}$, for every $\mathbf{h}_\ell$ removed. Then, using (2.74) with $\boldsymbol{R}_{\mathbf{x}'} = \sigma_x^2 \boldsymbol{I}$, and $\boldsymbol{R}_{\boldsymbol{v}} = \sigma_v^2 \boldsymbol{I}$, recursions (2.65)-(2.70) simplify to

$$\begin{aligned}
\boldsymbol{R}_{\bar{e},\ell} &= \sigma_v^2 \boldsymbol{I} + \sigma_x^2 \boldsymbol{H}_\ell \boldsymbol{H}_\ell^* \,, \qquad \boldsymbol{H}_0 = \boldsymbol{H} && (2.88)\\
\boldsymbol{K}_{p,\ell} &= \sigma_x^2 \mathbf{h}_\ell^* \boldsymbol{R}_{\bar{e},\ell}^{-1} && (2.89)\\
\bar{\mathbf{e}}_\ell &= \bar{\mathbf{e}}_{\ell-1} - \mathbf{h}_{\ell-1} x_{\ell-1} \,, \qquad \bar{\mathbf{e}}_0 = \mathbf{y}' \;\;,\;\; x_{-1} = 0 && (2.90)\\
\hat{x}_\ell &= \boldsymbol{K}_{p,\ell} \bar{\mathbf{e}}_\ell \,, && (2.91)\\
x_\ell &= f(\hat{x}_\ell) && (2.92)
\end{aligned}$$

for $\ell = 0, 1, \ldots, M-1$. For a more familiar notation, let $\boldsymbol{K}_{p,\ell} = \boldsymbol{g}_\ell^*$, and select the order of detection from the most powerful signal to the least, by picking j_ℓ that minimizes (2.86), or, $\|\boldsymbol{g}_\ell\|^2$. If we further denote the scalar $\sigma^2 \triangleq \sigma_v^2/\sigma_x^2 = 1/\mathsf{SNR}$, (2.88)–(2.92) becomes equivalent to the following recursions:

$$\begin{aligned}
\boldsymbol{g}_\ell &= (\sigma^2 \boldsymbol{I} + \boldsymbol{H}_\ell \boldsymbol{H}_\ell^*)^{-1} \mathbf{h}_\ell \,, \quad \text{where } \mathbf{h}_\ell = \arg\,\min \|\boldsymbol{g}_\ell\|^2 \,, \quad \boldsymbol{H}_0 = \boldsymbol{H} && (2.93)\\
\bar{\mathbf{e}}_\ell &= \bar{\mathbf{e}}_{\ell-1} - \mathbf{h}_{\ell-1} x_{\ell-1} \,, \qquad \bar{\mathbf{e}}_0 = \mathbf{y}' \;\;,\;\; x_{-1} = 0 && (2.94)\\
x_\ell &= f(\boldsymbol{g}_\ell^* \bar{\mathbf{e}}_\ell) && (2.95)
\end{aligned}$$

The above algorithm is well known as the *Vertical Bell Labs Layered Space Time* (V-BLAST), which has been long proposed in [27] as a possible receiver

architecture for MIMO systems. Note however, that here we are not restricting the transmission matrix $\boldsymbol{H}$ to be a tall, or square, well-conditioned one. In the next section, we shall propose a fast procedure for obtaining the V-BLAST vectors in the context of zero-padding based transceivers.

2.6 Fast Computation of the Widely-Linear V-BLAST for Increasing Index Ordered Detection

The V-BLAST algorithm that relies on optimal ordering is computationally demanding, due to the optimization of the norms in (2.93). We now argue that this burden can be substantially reduced if detection is performed sequentially, in a decreasing order of entry indexes. This is specially useful in the context of a ZP transceiver, in case IBI is not removed by the previously detected block, but rather optimally estimated within the current transmission. A significant consequence of this fact is that, as far as symbol redundancy is concerned, throughput can be maximized, since the effective redundancy in transmission can be lowered down to zero. Moreover, the WL formulation can be accounted for, by replacing $\boldsymbol{H}$ with its extended definition $\boldsymbol{H}_e$. In order to see this, consider our original model in (2.3), which we reproduce here for convenience:

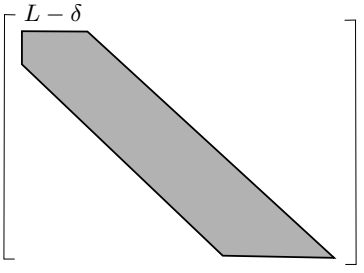
$$\mathbf{y}_n \;=\; \boldsymbol{H}\mathbf{T}_n\mathbf{s}_n + \boldsymbol{v}_n, \tag{2.96}$$

and assume that the precoder $\mathbf{T}_n$ is chosen, similarly to (2.100), as

$$\boldsymbol{\mathcal{I}}_\delta \;=\; \left[\begin{array}{c} \mathbf{0}_{\delta\times P+L-1} \\ \left[\begin{array}{cc} \boldsymbol{I}_{P+L-\delta-1} & \mathbf{0}_{P+L-\delta-1\times\delta} \end{array}\right] \end{array}\right] \tag{2.97}$$

This means that we are padding δ zeros to the transmitted block, which has size $P+L-\delta-1$. Hence, if $0 \leq \delta \leq L-1$, transmission is accomplished via zero-redundancy, and by denoting $\mathbf{H}_\delta \triangleq \boldsymbol{H}\mathbf{T}_n$, i.e.,

$$\mathbf{H}_\delta = \begin{array}{c} \\ \delta+1 \\ \\ \end{array}\!\!\left[\begin{array}{c} \overbrace{\qquad\qquad}^{L-\delta} \\ \\ \\ \end{array}\right.$$



our goal is to recover $\boldsymbol{s}_n = \mathbf{s}_n(0 : P + L - \delta - 2)$ efficiently from

$$\mathbf{y}_n \;\; = \;\; \mathbf{H}_\delta \boldsymbol{s}_n + \boldsymbol{v}_n. \tag{2.98}$$

Now, we recognize that the forward filter (2.93) *is the exact expression of the vector-valued Kalman gain found in a standard regularized LS problem*, which admits a fast computation, whenever the data matrix $\mathbf{H}_\delta$ is formed sequentially in its natural time index ordering (i.e., without its optimization). This can be done by associating to $\mathbf{h}_\ell$ the columns of $\mathbf{H}_\delta$ starting from its right-most one. As a result, we can elegantly borrow the Fast Transversal Filter (FTF) recursions used for efficient LS algorithms to achieve a fast computation of all $\{\boldsymbol{g}_\ell\}$ comprising the feedforward DFE matrix. We remark that no fast computation for the V-BLAST in this scenarion is available in the literature. Observe that this amounts to $P + L - 1$ operations per sample, which can be contrasted with its scalar DFE counterpart, requiring $P(P + L - 1)$ operations per sample. Table 1 lists the FTF algorithm considering a pre-windowed data matrix $\mathbf{H}_\delta$ arising in the (non-optimized ordering) V-BLAST algorithm[1].

The initialization step begins by assigning the inverse SNR, σ^2, to the minimum costs $\zeta^f(-2) = \zeta^b(-1)$, the likelihood variable, $\gamma_M(0) = 1$, and the vectors $\boldsymbol{w}^f_{-1} = \boldsymbol{w}^b_0 = \boldsymbol{k}_{M-1,0} = \mathbf{0}$. The regressor $\boldsymbol{u}_n$ corresponds to the n-th row of $\mathbf{H}_\delta$, and we iterate recursions 1)–13) in order to compute all $\boldsymbol{g}_\ell$.

Now, a further improvement is obtained when $\delta \geq L - 1$. In this case, $\mathbf{H}_\delta$ becomes lower triangular, which means that only half of the recursions in Table 2.1 are necessary. This is because the backward prediction filters used by the FTF algorithm are always null before $P + L - \delta$ iterations of the recursions, leaving the solution solely to the forward prediction part.

Finally, due to the LS form of $\boldsymbol{g}_\ell$ in the V-BLAST algorithm, when the complex channel is replaced by its extended version $\boldsymbol{H}_e$, we can decouple our model into two with complex channels $\left\{\boldsymbol{H}, (\boldsymbol{H}^*)^{\mathsf{T}}\right\}$, and proceed by running the above FTF recursions twice. For the case of real constellations, instead, we can equivalently reformulate the channel model in (2.79) by regrouping the real and imaginary entries in pairs, so that the resulting data matrix becomes block Toeplitz with 2×1 block entries. In this way, the FTF algorithm is still run in the exact same form, with double the complexity.

[1]The FTF algorithm is known to suffer from numerical stability. However, instability is normally a concern when dealing with long streams of data, when numerical errors inevitably accumulate. Here, the input data to the algorithm is a finite length impulse response, not exceeding a few hundreds of taps in most typical cases, and during this time divergence is unlikely to occur.

Initialization:

$\zeta^f(-2) = \zeta^b(-1) = \sigma^2, \quad \gamma_M(0) = 1, \quad \boldsymbol{w}_{-1}^f = \boldsymbol{w}_0^b = \boldsymbol{k}_{M-1,0} = \mathbf{0}, \quad \mathbf{h}_{-1} = \mathbf{0}$

For $\ell = 0$ to $P + L - \delta - 1$, repeat (1)–(14):

1) $\alpha_M(\ell-1) = \hat{h}(\ell) - \mathbf{h}_{\ell-1}\boldsymbol{w}_{\ell-2}^f$
2) $f(\ell-1) = \gamma_M(\ell-1)\alpha(\ell-1)$
3) $\boldsymbol{k}_{M,\ell-1} = \begin{bmatrix} 0 \\ \boldsymbol{k}_{M-1,\ell-1} \end{bmatrix} + \frac{\alpha^*(\ell-1)}{\zeta^f(\ell-2)} \begin{bmatrix} 1 \\ -\boldsymbol{w}_{\ell-2}^f \end{bmatrix}$
4) $\zeta^f(\ell-1) = \zeta^f(\ell-2) + \alpha^*(\ell-1)f(\ell-1)$
5) $\boldsymbol{w}_{\ell-1}^f = \boldsymbol{w}_{\ell-2}^f + \boldsymbol{k}_{M-1,\ell-1}f(\ell-1)$
6) $\gamma_M(\ell) = \gamma_{M-1}(\ell-1)\frac{\zeta^f(\ell-2)}{\zeta^f(\ell-1)}$
7) $\nu(\ell) = (\text{last entry of } \boldsymbol{k}_{M,\ell-1})$
8) $\boldsymbol{k}_{M-1,\ell} = \boldsymbol{k}_{M,\ell-1}(1:M-1) + \nu(\ell)\boldsymbol{w}_{\ell-1}^b$
9) $\beta(\ell) = \zeta^b(\ell-1)\nu^*(\ell)$
10) $\gamma_{M-1}(\ell) = \gamma_M(\ell)/(1 - \gamma_M(\ell)\beta(\ell)\nu(\ell))$
11) $b(\ell) = \gamma_{M-1}(\ell)\beta(\ell)$
12) $\zeta^b(\ell) = \zeta^b(\ell-1) + \beta^*(\ell)b(\ell)$
13) $\boldsymbol{w}_\ell^b = \boldsymbol{w}_{\ell-1}^b + \boldsymbol{k}_{M-1,\ell}b(\ell)$
14) Set: $\boldsymbol{g}_\ell = \boldsymbol{k}_{M,\ell-1}\gamma_M(\ell)$

Table 2.1: Fast transversal computation of the V-BLAST filters $\boldsymbol{g}_\ell$.

Initialization:

$\zeta^f(-2) = \sigma^2, \quad \gamma_M(0) = 1, \quad \boldsymbol{w}_{-1}^f = \boldsymbol{k}_{M-1,0} = \mathbf{0}, \quad \mathbf{h}_{-1} = \mathbf{0}$

For $\ell = 0$ to $P + L - \delta - 1$, repeat (1)–(7):

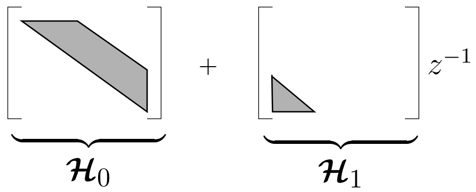
1) $\alpha_M(\ell-1) = \hat{h}(\ell) - \mathbf{h}_{\ell-1}\boldsymbol{w}_{\ell-2}^f$
2) $f(\ell-1) = \gamma_M(\ell-1)\alpha(\ell-1)$
3) $\boldsymbol{k}_{M,\ell-1} = \begin{bmatrix} 0 \\ \boldsymbol{k}_{M-1,\ell-1} \end{bmatrix} + \frac{\alpha^*(\ell-1)}{\zeta^f(\ell-2)} \begin{bmatrix} 1 \\ -\boldsymbol{w}_{\ell-2}^f \end{bmatrix}$
4) $\zeta^f(\ell-1) = \zeta^f(\ell-2) + \alpha^*(\ell-1)f(\ell-1)$
5) $\boldsymbol{w}_{\ell-1}^f = \boldsymbol{w}_{\ell-2}^f + \boldsymbol{k}_{M-1,\ell-1}f(\ell-1)$
6) $\gamma_M(\ell) = \gamma_{M-1}(\ell-1)\frac{\zeta^f(\ell-2)}{\zeta^f(\ell-1)}$
7) Set: $\boldsymbol{g}_\ell = \boldsymbol{k}_{M,\ell-1}\gamma_M(\ell)$

Table 2.2: Fast transversal computation of the V-BLAST filters $\boldsymbol{g}_\ell$ when $\delta > L - 1$.

2.7 Block Memoryless Equalization with Redundancy

A special case of (2.2) arises when $M = P \geq L$ (see e.g., [15]), so that the block impulse response comprises 2 square matrix coefficients, say, $\boldsymbol{\mathcal{H}}_0$, with first row given by the channel samples $[\,h(0) \quad h(1) \quad \cdots \quad h(L-1) \quad 0 \cdots \ 0\,]$, representing *inter-symbol-interference* (ISI), and $\boldsymbol{\mathcal{H}}_1$, the remaining IBI, i.e.,

$\boldsymbol{\mathcal{H}}(z) \quad =$



(2.99)

so that $\mathbf{x}_n = [\,\boldsymbol{x}_n^{\mathsf{T}} \quad \boldsymbol{x}_{n-1}^{\mathsf{T}}(0\!:\!L-2)\,]^{\mathsf{T}}$, and $\boldsymbol{H} = [\,\boldsymbol{\mathcal{H}}_0 \quad \boldsymbol{\mathcal{H}}_1(\,:\,,0\!:\!L-2)\,]$. Considering $\boldsymbol{\mathcal{H}}(z)$ in (2.99), in block memoryless equalization, $\mathbf{T}_n = (\boldsymbol{I} \otimes \mathbf{T})$, so that $\boldsymbol{x}_n = \mathbf{T}\boldsymbol{s}_n + \boldsymbol{t}_n$, where $\boldsymbol{s}_n = [\,s(Mn) \quad s(Mn-1) \quad \cdots \quad s(Mn-M+1)\,]^{\mathsf{T}}$ is the information vector and $\boldsymbol{t} = [\,t(Mn) \quad t(Mn-1) \quad \cdots \quad t(Mn-M+1)\,]^{\mathsf{T}}$ is the corresponding training vector for that particular block. In this case, redundancy eliminates IBI via zero-padding (ZP) or zero-jamming (ZJ) with or without cyclic prefixing, or in the more general case, through a hybrid form of these schemes (ZP-ZJ) [17]. That is, assume that the channel state information (CSI) is available, and define the restriction matrices

$$\boldsymbol{\mathcal{I}}_\delta \quad = \quad \begin{bmatrix} \mathbf{0}_{\delta \times M} \\ \boldsymbol{I}_M \end{bmatrix} \tag{2.100}$$

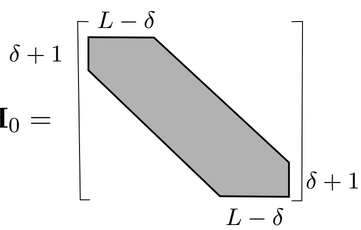
$$\bar{\boldsymbol{\mathcal{I}}}_\delta^{\mathsf{T}} \quad = \quad \begin{bmatrix} \boldsymbol{I}_{P-L+1+\delta} & \mathbf{0}_{(P-L+1+\delta)\times(L-1-\delta)} \end{bmatrix}, \tag{2.101}$$

where $\delta \in \{0, L-1\}$. The matrix (2.100) is multiplied by the transmitted vector so as to perform ZP, while (2.101) is multiplied by the output block for the purpose of ZJ. In this way, defining the precoder as $\boldsymbol{T} = \boldsymbol{\mathcal{I}}_\delta \mathbf{T}$, and assuming an additive noise vector $\boldsymbol{v}_n$ with power $\boldsymbol{R}_{\boldsymbol{v}} = \sigma_v^2 \boldsymbol{I}$, the received block after IBI removal is given by

$$\mathbf{y}_n^o \quad = \quad (\bar{\boldsymbol{\mathcal{I}}}_\delta^{\mathsf{T}}\, \boldsymbol{\mathcal{H}}_0 \boldsymbol{\mathcal{I}}_\delta)\mathbf{T}\boldsymbol{s}_n \; + \; \bar{\boldsymbol{\mathcal{I}}}_\delta^{\mathsf{T}}\, \boldsymbol{v}_n = \mathbf{H}_0 \boldsymbol{T}\boldsymbol{s}_n \; + \; \boldsymbol{v}_n^o \tag{2.102}$$

where $\mathbf{H}_0 \triangleq \bar{\boldsymbol{\mathcal{I}}}_\delta^{\mathsf{T}}\, \boldsymbol{\mathcal{H}}_0 \boldsymbol{\mathcal{I}}_\delta$ is $(M + 2\delta - L + 1) \times M$. It assumes the following general banded Toeplitz structure,

$\mathbf{H}_0 =$



(2.103)

with first row given by $[\,h(\delta) \;\cdots\; h(L-1) \quad 0 \;\cdots\; 0\,]$, and first column $[\,h(\delta) \;\cdots\; h(0) \quad 0 \;\cdots\; 0\,]^{\mathsf{T}}$. The extreme cases of $\delta = 0$ and $\delta = L-1$ correspond to the full ZJ and ZP schemes respectively. Choices between these values are said to be *of reduced redundancy* [17], since typically in block transceivers the redundancy is chosen to be of the same order of the channel length. The case when $\delta = \lceil (L-1)/2 \rceil$ zeros are padded and discarded at the receiver has been referred to as a *minimum-redundancy* system, where the reminiscent ISI is expressed by a square Toeplitz matrix. Hence, in principle, from the perspective of bandwidth efficiency, such minimum-redundancy transmission obtained with $\delta = \lfloor (L-1)/2 \rfloor$ is appealing.

In case IBI of symbols is removed by the discarding of output samples, a minimum-redundancy receiver assumes a priori that the transmitted vector is estimated in a single shot by inverting a submatrix of $\mathbf{H}_0$ — or, accordingly, its corresponding covariance $(\alpha \boldsymbol{I} + \mathbf{H}_0^* \mathbf{H}_0)$, in the case of a regularized LS (MMSE) estimator of the form $(\alpha \boldsymbol{I} + \mathbf{H}_0^* \mathbf{H}_0)^{-1} \mathbf{H}_0^*$. This is however, a naive assumption, since it fixes a single formula for recovering the transmitted vector, and ignores the fact that $\boldsymbol{x}$ exhibits a constellation structure. Moreover, from our discussion on the optimality of the decision delay δ in the block-DFE derivation, we see the optimal-redundancy in a reduced-redundancy scenario assumes the same role of the optimal value given by Eq. (2.87). That is, the optimal redundancy level should be chosen in a way that it minimizes the norm of the estimator, and any non-optimal choice would increase it. As a result, we conclude that for arbitrary channels, and therefore arbitrary maximum-phase zeros, their number may not match the optimal redundancy level, such that square transmission matrices would imply the highest probability of noise amplification upon inversion, or even linear MMSE estimation. We shall return to this issue on Sec. 2.9 when analyzing the value of RR systems more carefully.

Alternatively, as argued in [12], redundancy can be lowered down to zero, by feeding back past detected blocks $\boldsymbol{x}_{n-1}(0\!:\!L-2)$ in a DFE mode and still outperform a receiver that simply discards samples. Of course, if $P \gg L$, the contribution of $\boldsymbol{\mathcal{H}}_1$ is small, so that usually detecting $\boldsymbol{x}_n$ becomes more reliable than attempting to obtain $\boldsymbol{x}_{n-1}$ first. If $P = L$, on the other hand, for impulse responses behaving toward maximum-phase systems, the contribution of $\boldsymbol{\mathcal{H}}_0$ will be negligible. In other words, differently from the ZP-ZJ approach, inter-block-interference can be removed via decision-directed receivers, and more importantly, without introducing any form of redundancy, through a simple one-tap block DFE [51]. This concept has been exploited in [24], and more recently in [12] in a way that a DFE receiver that eliminates $\boldsymbol{\mathcal{H}}_1$ through feedback outperforms a system that simply discards relevant received samples on which IBI exists, in terms of symbol error rate and mutual

information. That is, once the channel is estimated, IBI is removed and the role of the receiver is to deal with the remaining ISI represented by $\boldsymbol{\mathcal{H}}_0\boldsymbol{T}$. For example, in [24], after IBI removal, the remaining ISI is removed by another DFE as illustrated in Fig. 2.5.

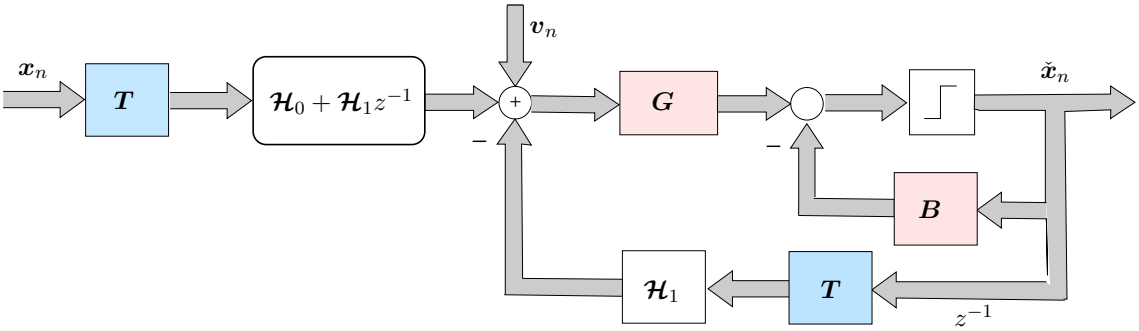


Figure 2.5: One-tap block DFE.

The two forms of IBI cancelation seen in the above block DFE and in (linear) reduced redundancy schemes, therefore suggest a more powerful combination of these schemes into a single one, exhibiting enhanced detection performance. That is, note that after padding with δ zeros at transmission, instead of discarding $L-1-\delta$ samples at reception, we may opt to cancel these remaining IBI samples by decision feedback. Specifically, write

$$\mathbf{y}_n^o = \boldsymbol{\mathcal{H}}_0\boldsymbol{\mathcal{I}}_\delta\boldsymbol{T}\boldsymbol{s}_n + \boldsymbol{\mathcal{H}}_1\boldsymbol{\mathcal{I}}_\delta\boldsymbol{T}\boldsymbol{s}_{n-1} + \boldsymbol{v}_n, \tag{2.104}$$

$$= \boldsymbol{H}_0\boldsymbol{T}\boldsymbol{s}_n + \boldsymbol{H}_1\boldsymbol{T}\boldsymbol{s}_{n-1} + \boldsymbol{v}_n \tag{2.105}$$

with $(M+\delta)\times M$ blocks $\boldsymbol{H}_0$ and $\boldsymbol{H}_1$, so that

$$\mathbf{y}_n' = \mathbf{y}_n^o - \boldsymbol{\mathcal{H}}_1\boldsymbol{\mathcal{I}}_\delta\boldsymbol{T}\check{\boldsymbol{s}}_{n-1} = \boldsymbol{H}_0\boldsymbol{T}\boldsymbol{s}_n + \boldsymbol{v}_n. \tag{2.106}$$

Therefore, compared to linear MR schemes, a block DFE employing $\delta_o < \lceil (L-1)/2 \rceil$ is still expected to result in superior BER performance, as long as redundancy is not so small that ill-conditioning and error propagation become important; in this way, one can seek a balance between a minimum, zero-redundancy scheme, and one that employs more redundant samples, with superior performance against the linear MR scheme, considering the computational complexity involved. Next, we shall pursue 2 forms of efficient implementations of transmultiplexers, focusing on zero redundancy transceivers with reduced complexity and WL formulations.

2.7.1 Displacement Structure in Signal Processing

In this section, we briefly review the concept of displacement structure in signal processing, and how it is connected to the efficient implementation of the algorithmic solutions pursued in this work. We formally introduce the displacement of an arbitrarily structured matrix, and refer the reader to [12] for further mathematical details.

Definition 1. *A matrix $\boldsymbol{M}$ is said to have a displacement structure with respect to the operator matrices $\{\boldsymbol{\Phi}, \boldsymbol{\Xi}\}$, if it satisfies the Stein and/or Sylvester displacement equations*

$$\nabla_{\{\boldsymbol{\Phi},\boldsymbol{\Xi}^*\}}(\boldsymbol{M}) \triangleq \boldsymbol{M} - \boldsymbol{\Phi}\boldsymbol{M}\boldsymbol{\Xi}^* = \boldsymbol{\mathcal{L}}\boldsymbol{\mathcal{Q}}^*, \qquad \nabla_{\{\boldsymbol{\Phi},\boldsymbol{\Xi}^*\}}(\boldsymbol{M}) \triangleq \boldsymbol{\Phi}\boldsymbol{M} - \boldsymbol{M}\boldsymbol{\Xi}^* = \boldsymbol{\mathcal{L}}'\boldsymbol{\mathcal{Q}}'^* \tag{2.107}$$

where $\{\boldsymbol{\mathcal{L}}, \boldsymbol{\mathcal{Q}}\}$ are $M \times r$ matrices whose columns are referred to as the generators of $\boldsymbol{M}$. The cardinal r is called the displacement rank of $\boldsymbol{M}$, where $r \ll M$.

In (2.107), the structure of the operators $\{\boldsymbol{\Phi}, \boldsymbol{\Xi}\}$ are properly chosen in accordance to the structure of $\boldsymbol{M}$ such that it yields a low rank factorization. For instance, Toeplitz and Hankel matrices have displacements ranks with respect to factor circulant operators $\{\boldsymbol{\Phi} = \boldsymbol{Z}_\phi, \boldsymbol{\Xi} = \boldsymbol{Z}_\varphi\}$, which does not exceed 2 [see, e.g., (2.112) further ahead]; *Cauchy* and the so-called *polynomial Vandermonde matrices* have displacement ranks with respect to diagonals $\{\boldsymbol{\Phi} = \boldsymbol{D}_s, \boldsymbol{\Xi} = \boldsymbol{D}_t\}$ and diagonal/Hessenberg matrices $\{\boldsymbol{\Phi} = \boldsymbol{D}_t, \boldsymbol{\Xi} = \boldsymbol{\Psi}\}$ which does not exceed 1 [52]. While these results can be proven for such types of matrices, defining displacement operators for arbitrary structures can be a more involved task.

The displacement structure of matrices has been exploited *implicitly* and *explicitly*, in a number of scenarios in the past, through proper choices of operators that can produce a low rank representation of a covariance, say, when $\boldsymbol{M} = \boldsymbol{P}_i$. The *Extended Generalized Sliding-Window Fast Transversal Filter* (EGSWFTF) algorithm of [53] is an example where the displacement generators are implicitly used to update the solution of a LS problem, by replacing the direct operations with the coefficient matrix $\boldsymbol{P}_i$, with the ones involving *its generators* instead. This is the core of every fast sequential RLS adaptive filter.

A second way to exploit structure, is to solve the displacement equations of (2.107) for $\boldsymbol{M}$ (either in its Stein or Sylvester forms). Depending on the operators choice, the solution may be represented efficiently, and used explicitly, for example, in the realization of a LS or a MMSE formula for a certain signal processing application. In this sense, given a linear model defined via a channel matrix $\mathbf{H}$, *Single-Carrier Frequency-Domain* (SC-FD) and multicarrier (MC) type schemes can be promptly envisioned via efficient factorizations of either the inverse of a square

submatrix of $\mathbf{H}$, in the case of a zero-forcing (ZF) receiver, or its corresponding covariance $(\alpha \boldsymbol{I} + \mathbf{H}^*\mathbf{H})^{-1}$, in the case of a regularized estimator of the form $(\alpha \boldsymbol{I} + \mathbf{H}^*\mathbf{H})^{-1}\mathbf{H}^*$. This is an application of displacement structure in a non-adaptive, explicit realization of $\boldsymbol{P}_i$.

Except when $\mathbf{H}$ exhibits an upper and/or lower triangular structure, it can be shown from [53], that in general, the following displacement equation for $\boldsymbol{P}_i$ holds, in connection to its defining fast *Kalman* recursion variables[2]:

$$\begin{aligned} \nabla_{\{\boldsymbol{\Phi}_{\boldsymbol{\theta}}, \boldsymbol{\Phi}_{\boldsymbol{\varsigma}}^*\}}(\boldsymbol{P}_i) \triangleq \boldsymbol{P}_i - \boldsymbol{\Phi}_{\boldsymbol{\theta}} \boldsymbol{P}_i \boldsymbol{\Phi}_{\boldsymbol{\varsigma}}^* \;\; &= \;\; \boldsymbol{\Phi}_{\boldsymbol{\theta}} \bar{\breve{\boldsymbol{k}}}_{M,N}^{d_o} \bar{\breve{\boldsymbol{k}}}_{M,N}^{d_o *} \boldsymbol{\Phi}_{\boldsymbol{\varsigma}}^* \; + \; \bar{\boldsymbol{w}}_{M-1,N}^{b} \bar{\boldsymbol{w}}_{M-1,N}^{b*} \\ &\quad - \boldsymbol{Z}_{\boldsymbol{\theta}}^{-1} \bar{\boldsymbol{w}}_{M-1,N-1}^{f} \bar{\boldsymbol{w}}_{M-1,N-1}^{f*} \boldsymbol{Z}_{\boldsymbol{\varsigma}}^{-*} \; - \; \tilde{\boldsymbol{k}}_{M-1,N} \tilde{\boldsymbol{k}}_{M-1,N}^{*} \end{aligned} \tag{2.108}$$

where $\{\bar{\boldsymbol{w}}_{M-1,N}^{b}, \bar{\boldsymbol{w}}_{M-1,N-1}^{f}, \bar{\breve{\boldsymbol{k}}}_{M,N}^{d_o}, \tilde{\boldsymbol{k}}_{M-1,N}\}$ correspond to normalized backward and forward prediction vectors, and the Kalman gains associated to data breakpoints at the first and last row of $\mathbf{H}$. The matrices $\{\boldsymbol{Z}_{\boldsymbol{\theta}}, \boldsymbol{Z}_{\boldsymbol{\varsigma}}\}$ have companion forms, with last columns given by the vectors $\{\boldsymbol{\theta} = [\,\theta_0 \;\cdots\; \theta_{M-1}\,]^T, \boldsymbol{\varsigma} = [\,\varsigma_0 \;\cdots\; \varsigma_{M-1}\,]^T\}$:

$$\boldsymbol{Z}_{\boldsymbol{\theta}} = \begin{bmatrix} 0 & 0 & \cdots & & \theta_0 \\ 1 & 0 & \cdots & & \theta_1 \\ \vdots & \vdots & \ddots & & \vdots \\ 0 & 0 & \cdots & 1 & \theta_{M-1} \end{bmatrix}, \quad \boldsymbol{Z}_{\boldsymbol{\varsigma}} = \begin{bmatrix} 0 & 0 & \cdots & & \varsigma_0 \\ 1 & 0 & \cdots & & \varsigma_1 \\ \vdots & \vdots & \ddots & & \vdots \\ 0 & 0 & \cdots & 1 & \varsigma_{M-1} \end{bmatrix}. \tag{2.109}$$

A key result of [54] is that a low displacement rank (in the above example, of 4), can always be satisfied as long as the operators $\{\boldsymbol{\Phi}_{\boldsymbol{\theta}}, \boldsymbol{\Phi}_{\boldsymbol{\varsigma}}\}$ are chosen in connection to the basis functions that generate the data in $\mathbf{H}$ as $\{\boldsymbol{\Phi}_{\boldsymbol{\theta}} = \boldsymbol{Z}_{\boldsymbol{\theta}}^{-1}, \boldsymbol{\Phi}_{\boldsymbol{\varsigma}} = \boldsymbol{Z}_{\boldsymbol{\varsigma}}^{-1}\boldsymbol{\Psi}\}$. Hence, by solving (2.108), we are able to find a general representation for $\boldsymbol{P}_i$ in terms of the eigenvectors of the constructed operators $\{\boldsymbol{\Phi}_{\boldsymbol{\theta}}, \boldsymbol{\Phi}_{\boldsymbol{\varsigma}}\}$.

2.7.2 DFT-based Superfast Receivers

Superfast representation of matrices refers to the $\mathcal{O}(M \log_2^p M)$ operations $(p \leq 3)$ that arise when multiplying a structured matrix by a vector, after solving the displacement equation (2.108) for some specific operators. One particular example arises when $\mathbf{H}$ is induced by tapped-delay-line models, which is equivalent to setting $\boldsymbol{\Psi} = \boldsymbol{I}$, thereby reflecting its Toeplitz-like structure. The well known DFT-representation is thus a special case of the above formula, when $\{\theta_i\} = \{\varsigma_i\} =$

[2]Moreover, since the parameters of this decomposition have an exact interpretation as *normalized* Kalman and prediction vectors, the computation of the generators can be accomplished by an EGSWFTF algorithm as well

0, for $i \neq 0$, so that the matrices $\{\boldsymbol{Z}_{\boldsymbol{\theta}}, \boldsymbol{Z}_{\boldsymbol{\varsigma}}\}$ collapse to what is known as θ_0- and ς_0 $-$*factor circulant* operators. Its eigenvalues correspond to the zeros of the so-called master polynomials $\bar{\Omega}_{\boldsymbol{\theta}}(z) = \phi_0 + z^{-M}$, and $\bar{\Omega}'_{\boldsymbol{\theta}}(z) = \varrho_0 + z^{-M}$. When these are associated to the entries of the vectors $\{\boldsymbol{z}_1, 1/\boldsymbol{z}_2^*\}$, then $z_1(m) = \phi e^{j\frac{2\pi m}{M}}$, where $\phi = |\phi_0|^{-1/M} e^{j\frac{\angle -\phi_0^{-1}}{M}}$, and $z_2(m) = \varrho e^{j\frac{2\pi m}{M}}$, with $\varrho = |\varrho_0|^{1/M} e^{j\frac{\angle -\varrho_0}{M}}$. Define

$$\boldsymbol{D}_{\phi} \triangleq \text{diag}(\{\phi^{-m}\}_{m=0}^{M-1}), \tag{2.110}$$

and the *DFT filterbanks* $\boldsymbol{V}_{\mathcal{P}}(\boldsymbol{z}_1) = \sqrt{M}\boldsymbol{F}\boldsymbol{D}_{\phi}$, $\boldsymbol{V}_{\mathcal{P}}(\boldsymbol{z}_2) = \sqrt{M}\boldsymbol{F}\boldsymbol{D}_{\varrho}, \boldsymbol{V}_{\mathcal{P}}(1/\boldsymbol{z}_2^*) = \sqrt{M}\boldsymbol{F}\boldsymbol{D}_{1/\varrho^*}$, where $\boldsymbol{F}$ is the DFT matrix. Then, it can be shown that the corresponding representation of $\boldsymbol{P}_i$ is given by

$$\boldsymbol{P}_i = \frac{1}{(\phi_0 - \varrho_0^*)} \boldsymbol{D}_{1/\phi} \boldsymbol{F}^* \sum_{k=1}^{4} \iota_k \boldsymbol{\Lambda}_{\boldsymbol{V}_1, \boldsymbol{b}_{k,\boldsymbol{\theta}}} \boldsymbol{F} \boldsymbol{D}_{\phi\varrho^*} \boldsymbol{F}^* \tilde{\boldsymbol{\Lambda}}^*_{\boldsymbol{V}_2, \boldsymbol{b}_{k,\boldsymbol{\varsigma}}} \boldsymbol{F} \boldsymbol{D}_{1/\varrho^*} \tag{2.111}$$

where $\boldsymbol{\Lambda}_{V_1, \boldsymbol{b}_{k,\boldsymbol{\theta}}} = \mathsf{Diag}(\boldsymbol{V}_{\mathcal{P}}(\boldsymbol{z}_1)\boldsymbol{b}_{k,\boldsymbol{\theta}})$, $\tilde{\boldsymbol{\Lambda}}_{V_2, \boldsymbol{b}_{k,\boldsymbol{\varsigma}}} = \varrho^{(M-1)} \boldsymbol{\Lambda}_{V_2, \boldsymbol{b}_{k,\boldsymbol{\varsigma}}} \mathsf{Diag}\left(e^{-j\frac{2\pi m}{M}}|_{m=0}^{M-1}\right)$, are diagonal matrices with $\boldsymbol{\Lambda}_{V_2, \boldsymbol{b}_{k,\boldsymbol{\varsigma}}} = \mathsf{Diag}(\boldsymbol{V}_{\mathcal{P}}(\boldsymbol{z}_2)\boldsymbol{b}_{k,\boldsymbol{\varsigma}})$, in which, for compactness of notation we denote $\boldsymbol{b}_{1,\boldsymbol{\theta}} \triangleq \bar{\bar{\boldsymbol{k}}}^{d_o}_{M,N}$, $\boldsymbol{b}_{2,\boldsymbol{\theta}} \triangleq \bar{\boldsymbol{w}}^{f}_{M-1,N-1}$, $\boldsymbol{b}_{3,\boldsymbol{\theta}} \triangleq \boldsymbol{Z}_{\boldsymbol{\theta}} \bar{\boldsymbol{w}}^{b}_{M-1,N}$, $\boldsymbol{b}_{4,\boldsymbol{\theta}} \triangleq \boldsymbol{Z}_{\boldsymbol{\theta}} \tilde{\boldsymbol{k}}_{M-1,N}$, $\boldsymbol{b}_{1,\boldsymbol{\varsigma}} \triangleq \boldsymbol{Z}_{\boldsymbol{\varsigma}} \bar{\bar{\boldsymbol{k}}}^{d_o}_{M,N}$, $\boldsymbol{b}_{2,\boldsymbol{\varsigma}} \triangleq \boldsymbol{Z}_{\boldsymbol{\varsigma}}^{-1} \bar{\boldsymbol{w}}^{f}_{M-1,N-1}$, $\boldsymbol{b}_{3,\boldsymbol{\varsigma}} \triangleq \bar{\boldsymbol{w}}^{b}_{M-1,N}$, and $\boldsymbol{b}_{4,\boldsymbol{\varsigma}} \triangleq \tilde{\boldsymbol{k}}_{M-1,N}$, with $\iota_1 = \iota_3 = -1, \iota_2 = \iota_4 = 1$. The resulting receiver is illustrated in Fig. 2.6.

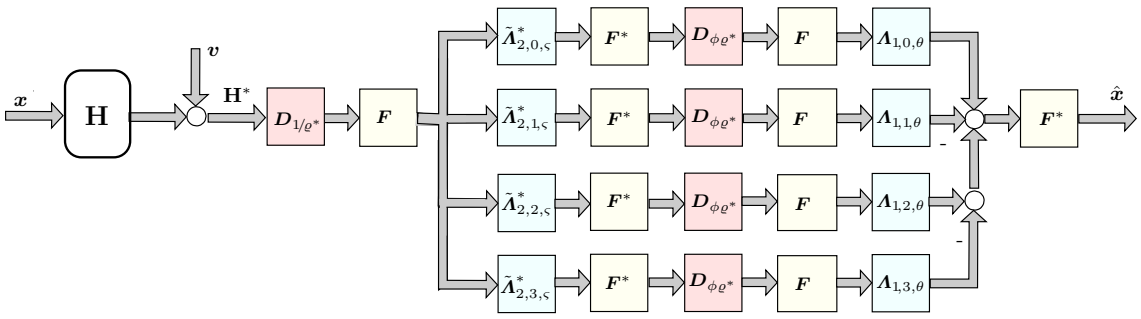


Figure 2.6: *SC-FD DFT Decomposition.*

Remark: Two particular transceivers that rely on the inversion of Toeplitz matrices are of special interest:

(i) ***ZF receiver***: In this case, a portion of size M of the received vector $\boldsymbol{y}'$ is captured, so that the resulting linear model relies on a simple inversion of a square Toeplitz matrix. Since Toeplitz inverses have a displacement rank of 2 with respect to circulant factors, any square subblock inverse of $\boldsymbol{\mathcal{H}}$ of within

$\boldsymbol{H}$ can be represented as follows:

$$\boldsymbol{\mathcal{H}}^{-1} = \frac{1}{1-\phi_0/\varrho_0^*}\boldsymbol{D}_{1/\phi}\boldsymbol{F}^*\Big[\boldsymbol{\Lambda}_{\boldsymbol{\theta},\bar{\boldsymbol{w}}_1}\boldsymbol{F}\boldsymbol{D}_{\phi\varrho^*}\boldsymbol{F}^*\boldsymbol{\Lambda}^*_{\boldsymbol{\varsigma},\bar{\boldsymbol{w}}_2} - \boldsymbol{\Lambda}_{\boldsymbol{\theta},\boldsymbol{Z}_{\boldsymbol{\theta}}\bar{\boldsymbol{w}}_2^{\#}}\boldsymbol{F}\boldsymbol{D}_{\phi\varrho^*}\boldsymbol{F}^*\boldsymbol{\Lambda}^*_{\boldsymbol{\varsigma},\boldsymbol{Z}_{\boldsymbol{\varsigma}}\bar{\boldsymbol{w}}_1^{\#}}\Big]\boldsymbol{F}\boldsymbol{D}_{1/\varrho^*} \quad (2.112)$$

Unlike the rank-4 case, the diagonal matrices $\boldsymbol{\Lambda}_{(\cdot,\cdot)}$ depend on only two prediction (generating) vectors $\{\bar{\boldsymbol{w}}_1, \bar{\boldsymbol{w}}_2\}$ (see details in [55],[54]).

(ii) ***Full*** $(L-1)$ ***redundancy ZP receiver***: It is well known that higher redundancy results in better BER performance. Moreover, besides superiority in detection, ZP schemes also allow for less complex representations, since when $\delta = L-1$, $\mathbf{H}$ exhibits a doubly-windowed structure, and $\mathbf{H}^*\mathbf{H}$ in this case becomes symmetric Toeplitz, with displacement rank 2. Hence, its inverse is represented via 2 branches only, except that here symmetry implies computation of a single generating vector. This fact was already used in [56],[55] for channel estimation in a high Doppler OFDM setup.

2.8 Reduced-Complexity-Widely-Linear Superfast BI-DFE

The initial estimate obtained before any iterative detection scheme requires efficient implementation of an MMSE estimate. The DFT-based expressions of (2.111) and (2.112) can be extended to the WL scenario by replacing the channel matrix $\boldsymbol{H}$ with its extended WL version, i.e., $\boldsymbol{H}_e$. We can envision two main forms of estimation, namely, ZF and MMSE, and all we need is to write the corresponding decompositions for a subblock of $\boldsymbol{H}_e$, or its corresponding covariance $(\boldsymbol{\mathcal{D}} + \boldsymbol{H}_e^*\boldsymbol{H}_e)^{-1}$ respectively, for some block diagonal matrix $\boldsymbol{\mathcal{D}}$. Since from (2.83), $\boldsymbol{H}_e = \mathbf{U}_P^*\boldsymbol{\Lambda}\mathbf{U}_Q$, these solutions are easily implemented via (2.111) and (2.112) by replacing the channel definition with $\{\boldsymbol{H}, (\boldsymbol{H}^*)^{\mathsf{T}}\}$. This means that we can simply use these formulas to express $\boldsymbol{P}_e = \mathbf{U}_Q^*(\boldsymbol{\mathcal{D}} + \boldsymbol{\Lambda}^*\boldsymbol{\Lambda})^{-1}\mathbf{U}_Q$ in terms of the decoupled covariances $\{\boldsymbol{P}, \boldsymbol{P}^{\mathsf{T}}\}$, since $(\boldsymbol{\mathcal{D}} + \boldsymbol{\Lambda}^*\boldsymbol{\Lambda})^{-1} = (\boldsymbol{P} \oplus \boldsymbol{P}^{\mathsf{T}})$.

On the other hand, when $\boldsymbol{H}_e$ has the tall structure defined in (2.79), we may expect further improvement in comparison to its SL counterpart. This is the case when real modulation schemes such as BPSK or PAM are employed. Observe that if $\boldsymbol{H}$ is $(P \times P + L - 1)$, $\boldsymbol{H}_e$ is $(2P \times P + L - 1)$, and all we need in general for well-conditioned estimation is that $P \geq L-1$, a requirement easily satisfied. Note that such configuration not only implies zero redundancy, but a higher throuput since we are mapping $P + L - 1$ samples of real data to P samples of complex data.

2.9 On Reduced-Redundancy Efficient Superfast Transceivers

As reported recently in [57, 58], the superfast formula of (2.111) has been claimed novel by the authors of [28–32, 59], in the context of block *reduced-redundancy* (RR) scenarios. In this section, we point out that the performance of those RR schemes depends on the channel, in such way that, for most channels this gain is negligible. Moreover, we argue against the choice of system *throughput* as the sole figure of merit, since this is considered under coded transmission. The purpose of this section is to complement the results of [12] regarding the use of such transceivers, in a three-fold sense: First, to clarify that contrary to the claims of [29], there is no advantage in considering superfast DFT-based linear *minimum-redundancy* (MR) transceivers against standard *orthogonal frequency division multiplexing* (OFDM) and *single-carrier frequency division* (SC-FD) schemes even for coded transmissions; Second, to show that what the authors refer to as a DHT-based receiver also offers no gain compared to either the analogous DFT or standard schemes as well. Third, to use these conclusions in order to motivate the importance of the new iterative solutions proposed in this work, which instead can benefit from superfast realizations, given an actual, justified gain in terms of BER and throughput.

Thus, recall from our previous discussion on superfast receivers, that their MMSE or LS structure relies mainly on the decomposition of the covariance $(\alpha \boldsymbol{I} + \mathbf{H}^{*}\mathbf{H})^{-1}$. Because in general its displacement rank is 4, it yields a 4-branch receiver design (a ZF receiver would lead to a simpler 2-branch receiver). In either case, what is relevant to our discussion here is that the receiver is composed solely by FFTs matrices, denoted by $\boldsymbol{F}$, fixed diagonal matrices denoted by $\boldsymbol{D}_{(.)}$, and the diagonal matrices $\{\boldsymbol{\Lambda}_{(.)}, \tilde{\boldsymbol{\Lambda}}_{(.)}\}$, which represent the equalizer parameters, while multiplication by $\mathbf{H}^{*}$ is done efficiently via circulant embedding. A multi-carrier (MC) version of this scheme is simply obtained by moving the far end FFT (or any derived transform, for that matter) to the transmitter [12].

In [29] and all their subsequent papers, the authors argue that these superfast transceivers show competitive performance against standard OFDM/SC-FD schemes in some cases, and that these can only be noticed under coded transmissions, when the best figure of merit is the data throughput. Now it is known that when characterizing a receiver, the uncoded bit-error-rate (BER) is normally the first performance measure observed, and it is much more sensitive than block error rate (BLER) or coded BER alone. We bring attention to the fact that when proposing a new receiver, it is paramount that its BER be examined independently of the coding scheme employed. This is because it is also widely known that depending on the coding strategy, the BER can experience substantial improvements, which

can potentially mask the true benefit delivered by the proposed transceiver.

We do not believe that a comparison based on coded transmissions is the only option. One can equally argue that if we are to consider a real mobile environment, one should provide a more general experiment setup that can reflect the real difficulties encountered when inverting structured square matrices, as for instance: *i*) The effect of channel estimation; *ii*) The effect of ubiquitous intercarrier interference (ICI); *iii*) Most importantly: the real impact of ill-conditioned square matrices inherent to MR schemes considering *i*) and *ii*) — which is the reason why such square matrix inversions have always been avoided; *iv*) The combination of *i*), *ii*) and *iii*). Note that MR receivers are also more computationally demanding than conventional OFDM/SC-FD schemes, being roughly 10 times more complex (Fig. 2.6).

We invite the reader to revisit Eq. (2.87), in that the optimal redundancy of a block-by-block based scheme is given by the number of maximum-phase zeros of the underlying channel. This well established result, and which has an analytical proof (due to Scaglione et al.'s famous paper [15]), already tells us which channels can be equalized depending on the desired redundancy (which translates to decision delay). This fact answers why an MR scheme will not work in most cases. The explanation is as follows: The minimum $\lceil (L-1)/2 \rceil$ redundancy is optimal (in a minimum-norm-ZF sense) if it equals the number of stable zeros of the channel impulse response. The longer the block size compared to the channel, the taller is its full convolution matrix, and the smaller the probability that its optimal subblock corresponds to the one of an MR system. The FIR channels in, e.g., [28] and other publications are frequently of order 4 at most, chosen with 1 or 2 maximum-phase zeros, for which the minimum is optimal. In [29], it is stated that "If ill-conditioned channels are not taken into account, by including only the channels whose condition number of the Toeplitz matrix is 6 times larger than that of the circulant matrix (comparing to standard OFDM/SC-FD schemes), the MR scheme can achieve better results in terms of BER". Since the author simulates a random channel, this constitutes a modification of the actual scenario. These channels should not be taken out of the simulations, since they correspond to those where minimum redundancy is different from its optimal value, a fact that is reflected in the perceived ill-conditioning.

In [12], it can be verified that by choosing the channel zeros at any configuration different from having channels exhibiting average half stable zeros (= MR), including other power delay profiles (where the path delays do not vary randomly in general), and maximum-phase channels, the MR equalizer by itself fails by far.

Moreover, in an uncoded scheme that presents no BER gain at all, that is, one that exhibits a 'flat' BER curve (see, e.g., Fig. 2.9) — case of most MR simulations,

any gains seen come from the coding scheme itself and there is no advantage in considering those over the simplicity of standard OFDM/SF-FC schemes. And it is understood that in a more sophisticated coding scenario (specially when relying on feedback of soft-decisions), one can observe huge BER improvements — see, e.g. [55] in the case of superfast schemes in high mobility environments.

That being said, should the ultimate goal be to reduce redundancy in transmission, one may simply make use of a memoryless, zero-redundancy transceiver, as the one long proposed in [51] (and never cited by those authors), so that equipped with proper coding it can easily outperform the linear MR ones.

As we shall see next, simulations within a *Long Term Evolution* (LTE) setup show that a simple block DFE achieves superior performance, with the same complexity, and smaller redundancy than the lower bound allowed in the linear case.

2.9.1 Preliminary Simulations

In this section, we compare the performance of standard MC and SC-FD schemes with the ones of the corresponding DFT and DHT-based minimum and optimal linear reduced redundancy transceivers. We also contrast the performance of the latter with simple DFE receivers that eliminate IBI via decision feedback, which just like the linear case, are shown to be implemented in superfast complexity. For the sake of comparison with the experimental results of [28–32], we assume exact channel state information (CSI) and remark that in these references, the BER curves shown by the authors are displayed in the unrealistic range up to 50 dB SNR.

♦ ***Experiment 1 (Coded Reduced-Redundancy in SL Transceivers)*** : We illustrate the performance of RR systems under coded transmissions, by considering the same *throughput* figure of merit used in [29]:

$$\mathsf{Throughput} \;=\; b \cdot r_c \frac{M}{M+\delta}(1-\mathsf{BLER}) f_s \;\; \mathsf{bps} \tag{2.113}$$

with b denoting the number of bits required to represent a constellation symbol, r_c the code rate considering the protection of channel coding, f_s the sampling frequency, and with BLER standing for block-error rate, assuming that a data block is discarded whenever at least one of its original bits is incorrectly decoded at the receiver.

Figure 2.7 shows the throughput when blocks of $M = 32$, 4-QAM symbols are transmitted. These channel models are sampled at $f_s = 150\,\mathrm{MHz}$, and a convolutional code with constraint length 7, $r_c = 1/2$, and generators $g_0 = [133]\,(octal)$ and $g_1 = [165]\,(octal)$ are used. The channels have length $L = 15$,

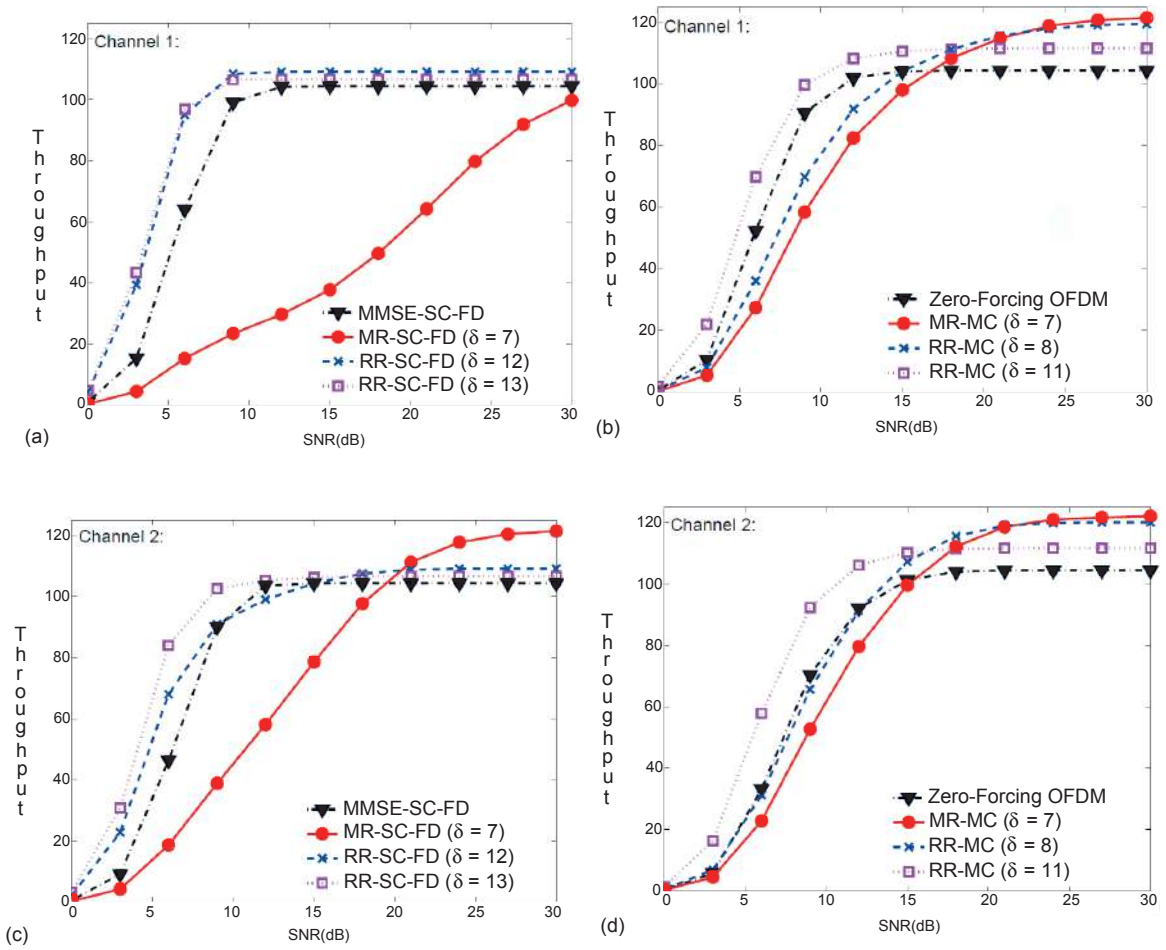


Figure 2.7: SC-FD [(a) and (c)] and MC [(b) and (d)] schemes for channels 1 and 2, for RR, MR, and standard systems — Throughput (Mbps) × SNR (dB).

which are borrowed from [12] (used in the uncoded experiments 1 and 3 therein):

$$H_1(z) = (0.77 + 0.38j) + 0.58jz^{-8} - 0.58z^{-9} - 0.567z^{-10} + 2.7z^{-13} + 0.4z^{-14},$$
$$H_2(z) = (0.77 + 2.38j) + 1.58jz^{-8} - 0.358z^{-9} - 0.567jz^{-10} + 0.5z^{-13} + 0.1z^{-14}.$$

The amount of redundancy ranges from the minimum $\delta = (L-1)/2 = 7$ to full redundancy $\delta = 14$. All receivers are MMSE-based [in fact these are least-squares (LS)-based, and approximately regularized by the SNR], except for the standard OFDM which is designed as ZF, for comparison. Note that it is known that for standard OFDM, both ZF and MMSE perform the same.

The results are more than clear. First, all SC-FD schemes clearly fail [see Figs. 2.7(a), 2.7(c)]. Note that the MR-SC-FD of Fig. 1(a) is of no use, while all RR-SC schemes require almost full redundancy (i.e., $\delta = 12$, $\delta = 13$) to outperform the standard ones in a range of low SNR [Fig. 2.7(c)], say up to 10 dB. Above 10 dB, the gains are negligible. In a meaningful SNR range, which is usually below ≈17-20 dB, even after coding, both ZF-OFDM/MMSE-SC-FD

schemes outperform the MR-MMSE all the way. Above that, the coded MR-MC curves [Figs. 2.7(b), 2.7(d)] only achieve a small throughput gain of at most 10% over standard schemes at 20 dB, and at 22 dB for Fig. 2.7(b), approximately. Actually, within this range, one can easily see that the area in the gap between all OFDM/SC-FD schemes and the MR curves is considerably larger than the corresponding one above 17-20 dB (i.e., in the limited range the MR system is claimed to work). Observe that these are *static* channels, and yet in such idealistic case (i.e., exact CSI, absence of ICI, reduced numerical errors due to high finite precision and SNR regularization in the simulations, etc), the small gains in throughput given by MR systems are only seen at **high SNR** figures. Moreover, we readily verify that even after coding, the meaningful SNR range for throughput improvement is rather reduced.

Remark 1: Note the high complexity employed by such receivers which, even under coding and MMSE regularization, cannot reach the performance of a simple standard *Zero-Forcing*-OFDM. The corresponding ZF-MR-MC schemes show no throughput gains for any the above channels.

Remark 2: The simulations shown in [29],[28] display the BER/Throughput curves in the SNR range up to 30-45 dB, which is unrealistic in practical scenarios, and any gain is only noticeable at very high SNR levels. As an example, the LTE specifications [33] consider any SNR value above 20 dB as excellent, and the recommended range for equipment testing is 13-20 dB.

♦ ***Experiment 2 (Comparison between DFT and DHT-based SL schemes)***: In [60], the authors claim that they have proposed a new DHT counterpart of the scheme in Fig. 2.6 that works for any channel impulse response, regardless of its symmetry. This is because, as we have previously mentioned, while the DFT-based formula originates from the so-called factor circulants $\{\boldsymbol{Z}_{-\phi_0}, \boldsymbol{Z}_{-\varrho_0}\}$, these can be alternatively expressed through DHTs, just by noticing that

$$\boldsymbol{Z}_1 = \boldsymbol{D}_{1/\phi}\boldsymbol{F}^*\boldsymbol{\Lambda}_{\bar{z}_1}\boldsymbol{F}\boldsymbol{D}_{\phi} = \mathbb{H}\mathcal{X}_{b_{k,\theta}}\mathbb{H} \tag{2.114}$$

$$\boldsymbol{Z}_{-1} = \boldsymbol{D}_{1/\rho}\boldsymbol{F}^*\boldsymbol{\Lambda}_{\bar{z}_2}\boldsymbol{F}\boldsymbol{D}_{\rho} = \mathbb{H}_{II}^T\mathcal{X}_{b_{k,\varsigma}}\mathbb{H}_{II} \tag{2.115}$$

where

$$\mathbb{H} \triangleq \Re\{\boldsymbol{F}\} + \Im\{\boldsymbol{F}\} \quad \text{and} \quad \mathbb{H}_{II} \triangleq \Re\{\boldsymbol{F}\boldsymbol{D}_{\phi}\} + \Im\{\boldsymbol{F}\boldsymbol{D}_{\phi}\}$$

are the Hartley transforms corresponding to the DFT, and a modified DFT, usually referred to as DFT-*II*, i.e., $\boldsymbol{F}_{II} \triangleq \boldsymbol{F}\boldsymbol{D}_{\phi}$, where again, $\boldsymbol{D}_{(\cdot)}$ are certain fixed diagonal matrices. While $\{\boldsymbol{\Lambda}_{\bar{z}_i}\}$ are diagonal, the matrices $\{\mathcal{X}_{b_{k,\theta}}, \mathcal{X}_{b_{k,\varsigma}}\}$ possess a nonzero diagonal and anti-diagonal only, except that their elements are dependent through

a mirror symmetry. Obtaining their DHT scheme is straightforward: Replacing (2.114) and (2.115), into the solution of a rank-n displacement equation for a generically structured covariance $\boldsymbol{P} = (\mathbf{H}^*\mathbf{H})^{-1}$ yields

$$\boldsymbol{P} = \frac{1}{2}\sum_{k=0}^{n} \mathbb{H}_{II}^{T}\mathcal{X}_{k,\varsigma}\mathbb{H}_{II}\mathbb{H}\mathcal{X}_{k,\theta}\mathbb{H}\,, \tag{2.116}$$

where n refers to the number of receive branches, given by the displacement rank of the channel matrix (in the case of ZF) or of its covariance (in the LS case). The matrices $\{\mathcal{X}_{k,\theta}, \mathcal{X}_{k,\varsigma}\}$ are of the same form of $\{\mathcal{X}_{b_{k,\theta}}, \mathcal{X}_{b_{k,\varsigma}}\}$. This amounts to replacing each receiver branch in Fig. 2.7 by matrix products of the form inside the summation of (2.116). That is, essentially, the DFT and the DHT formulas stem from the same displacement operator, and the latter is not a new invention.

Figure 2.8 illustrates the uncoded BER curves for the case when 9 delay paths are randomly located within an ensemble of 50000 $L = 41$-tap channels for MC and SC-FD schemes, with $M = 64$. This is more reasonable than considering trivial fourth order channels as random variables, given known wireless communication standards. It is also the best case scenario for the MR systems, since averaging the power delay profiles results on average in half stable zeros (=MR), which is not the case in practice. Still, even the optimally chosen redundancy DFT-based RR-MC systems reach worse performance than standard schemes. We further include full $(L-1)$ redundancy transmissions for comparison. The DHT-based schemes are illustrated with dashed curves. We can clearly see that, on average, there is also no BER performance gain of the DHT, over DFT-based schemes, except for a negligible improvement in the case of an MMSE receiver at high SNR, and for the optimal redundancy scenario. The minimum redundancy systems performs poorly, and much worse than standard multicarrier schemes, even for lengths and block sizes $\{L,M\}$ of the same order.

♦ ***Experiment 3 (Performance for an LTE channel and comparison with SL ZJ-DFE)***: Here we reinforce the simulations of [12] by considering an important case in which, in theory, no linear reduced-redundancy can outperform the standard MC/SC-FD schemes. Thus, consider an *Extended Pedestrian A (EPA)* model, in a LTE environment. We compare in Fig. 2.9, for $L = 21$, $M = 64$, QAM-4, the following (DFT-based) schemes, referring to the figure label, from the top: (a) A linear MR-MMSE receiver using zero-jamming (ZJ); (b) A linear full $(L-1)$-redundancy MC-MMSE receiver; (c) A linear MR-MC-MMSE receiver using ZJ, with one symbol re-estimation as in [12]; (d) The linear full $(L-1)$-redundancy MC-MMSE receiver with ZJ, and one symbol re-estimation; (e) The ZF counterpart of (a); (f) The ZF counterpart of (b); (g) Standard

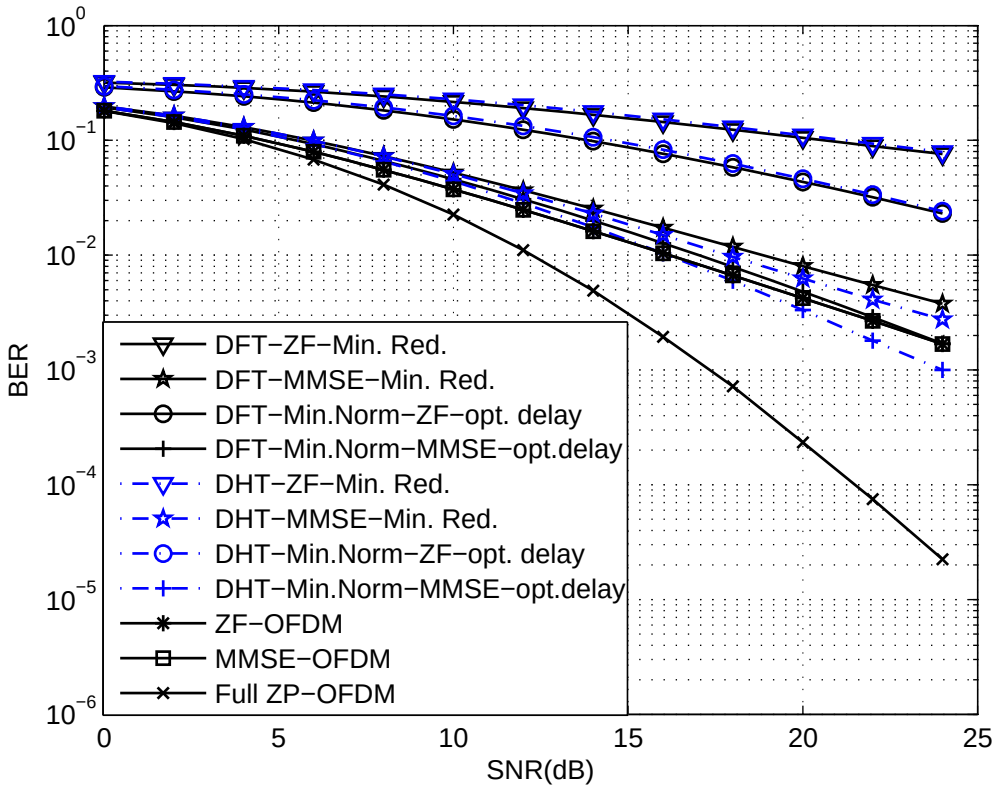


Figure 2.8: MC transceivers, $M = 64, L = 41$ (9 path gains randomly located).

ZF-SC-FD, and (h) Standard ZF-OFDM.

This scenario shows that the situation is different for a practical power delay profile. It is possible to verify that the LTE channel has $\approx$ 5-6 maximum-phase zeros, implying that $\delta = 5$ or 6 redundancies are optimal, according to [12],[15]. Observe that this level of redundancy cannot be attained by any linear or DFE receiver via ZP-ZJ as the one proposed by the authors in [61], even employing symbol re-estimations (as in [12]). They can only reduce redundancy down to the minimum of $\delta = (L-1)/2 = 10$. As can be seen from Fig. 2.9 (dark, flat curves), the ZF schemes exhibit high, constant BERs = 0.4 for a ZF receiver, and around 0.02 for the ZJ-DFEs, for SNR > 15 dB. Moreover, as in the previous example, the channel and the block size are of the same order, and unlike suggested in [29, 30], it gives no advantage to the MR systems.

For completeness of the argument, we include in Fig. 2.10, for the same setup, the memoryless superfast BI-GDFE that eliminates IBI via decision feedback (instead of ZJ), as proposed in [12].

This example employs *half of the minimum-redundancy* used by the author of [29], i.e., $\delta = 5$ (and roughly the same computational complexity). Compared to the standard ZF-MC/SC transceivers, the reader can verify how these curves stand out

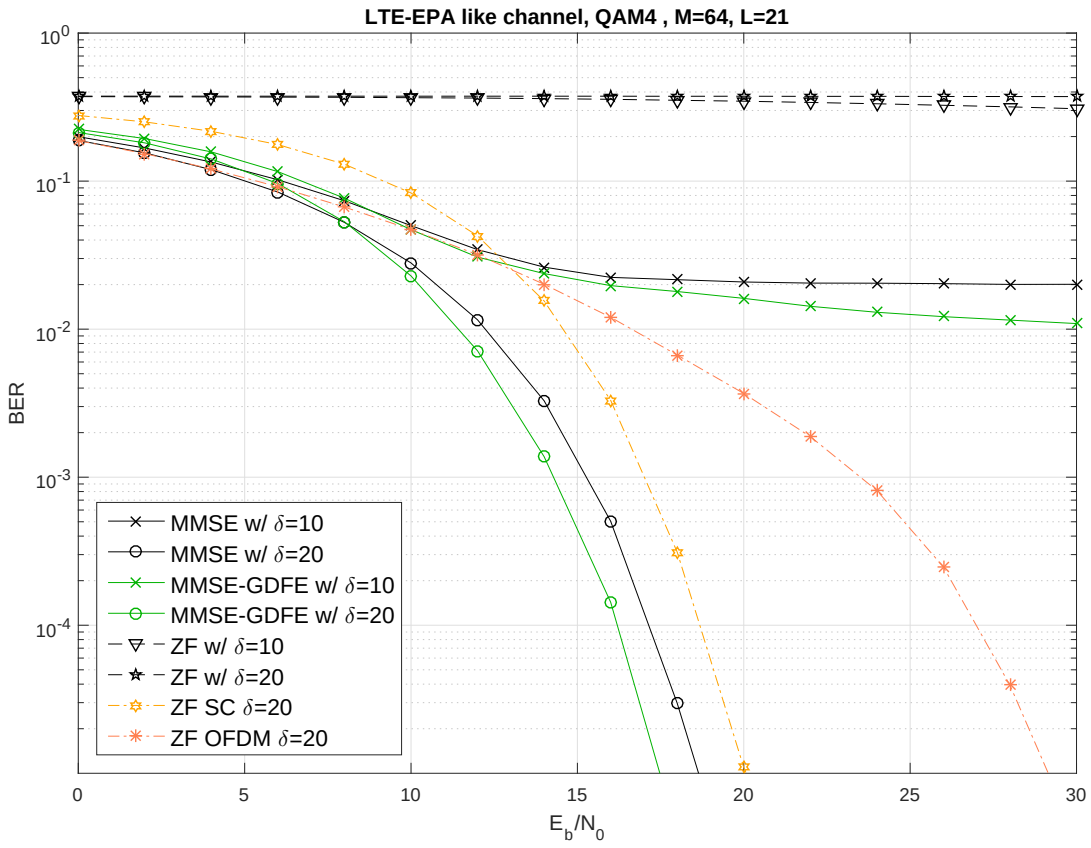


Figure 2.9: LTE EPA channel model — Comparison with the ZJ scheme.

significantly, and how close these are to the one of full redundancy-DFE ($\delta = 20$). The latter in turn performs just as well as the one employing $\delta = 10$, which is the same amount of redundancy used by the linear MR scheme proposed by the authors in [29]. Moreover, note from Fig. 2.9, that the $\delta = 10$ redundancy scheme, where IBI is removed by ZJ, shows a performance that is significantly inferior to the one of Fig. 2.10, which instead removes IBI via decision feedback.

In summary, these are the conditions in which MR is claimed to work. Gains are only seen:

1) Under coded transmissions, in terms of Throughput;

2) For channels with exactly half stable zeros;

3) When M and L are of the same order;

4) Only for MMSE receiver;

5) Within all the above, in a reduced range, at high SNR.

Add to all the above, the complexity of the transceiver which is approximately 10 times the complexity of the standard OFDM (which still performs better, even under coding), the referred superfast RR transceivers are clearly of no use.

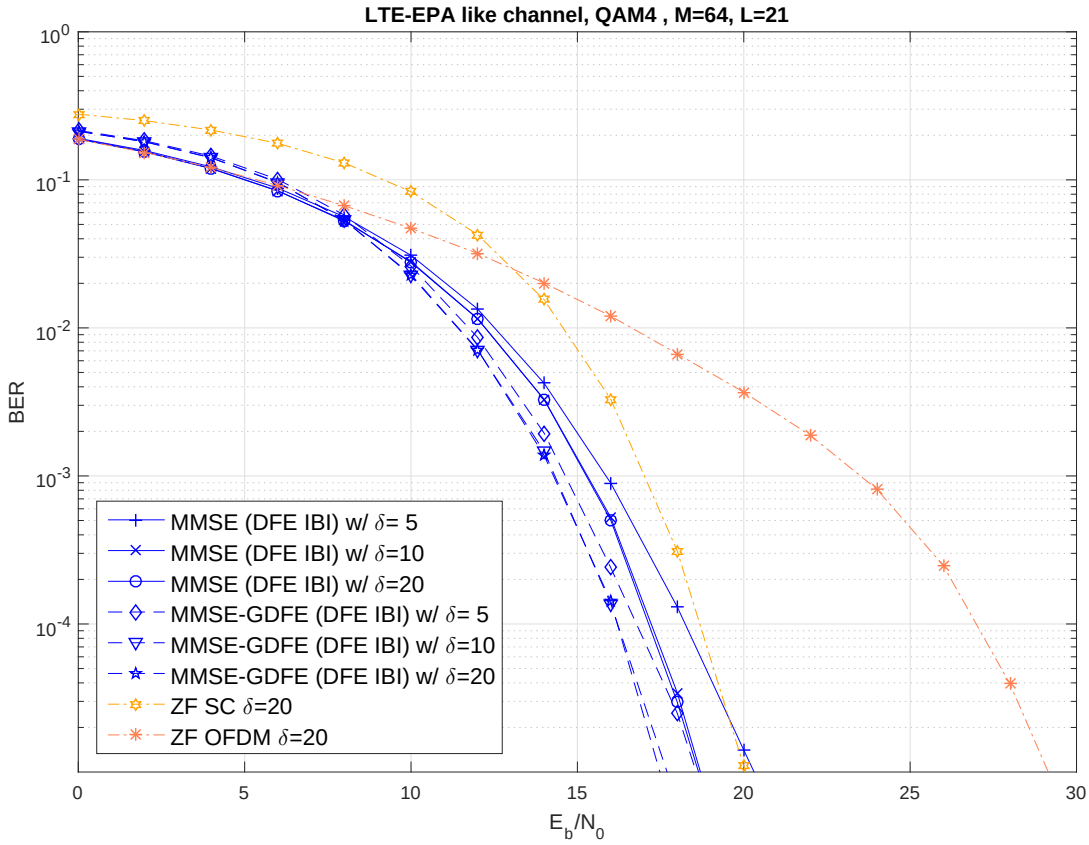


Figure 2.10: LTE-EPA: Block GDFE (DF-IBI) $\times$ ZJ-DFE $\times$ standard schemes.

2.10 Optimal Iterative Estimation of Digitally Modulated Signal Constellations

From the above conclusions and the discussion on the need for a reduced redundancy δ, the optimality of the equivalent decision delay δ plays an important role when choosing the best set of *contiguous entries* to be estimated within the transmitted vector $\mathbf{x}$. This assumes that we are looking into the best submatrix of $\boldsymbol{H}$ constituted by *contiguous columns*, such that, besides being invertible, it also exhibits the best conditioning for inversion or vector estimation, given the equivalent noise model defined through the remaining subblocks. The selected columns of $\boldsymbol{H}$ should then act as a good 'sensing matrix', so that the former properties can be more easily achieved.

These concepts are strongly related to the building blocks of the theory of compressed sensing, with regards to the properties of general *frames* (not necessarily Toeplitz) and which play the role of sensing matrices for sparse signals. It is well known that good sensing matrices, say, of size $M \times P$, $P \geq M$, are given by the so-called *full spark* frames. An $M \times P$ matrix $\mathbf{H}$ is said to be *full spark* if its spark is as large as possible, i.e., $\mathsf{Spark}(\mathbf{H}) = M + 1$. This means that they are capable of

supporting signals with the smallest possible sparsity level, that is, signals with the largest cardinality. Equivalently, full spark matrices have the property that every $M \times M$ submatrix of $\mathbf{H}$ is invertible, and as such, are necessarily full rank (hence constituting a frame). While full spark matrices have invertible submatrices, they may not be well-conditioned, which is an important feature in compressed sensing. Sensing matrices with well-conditioned submatrices satisfying the RIP allows for stable and efficient recovery of sparse signals. Unfortunately, as we pointed out in Sec. 1.1, determining the isometry constant that defines the RIP is generally an intractable problem — the same conclusion holding for finding its spark — so in practice we replace it by a lower bound on the spark, based on the mutual coherence.

The mutual coherence also plays a role in establishing a lowerbound on the non-zero eigenvalues of the Gramian matrix of full spark frames. First, note that all eigenvalues of a Gramian matrix are non-negative. In this case, assuming that $\mathsf{Spark}(\mathbf{H}) = M + 1$, there is at least one principal submatrix $\boldsymbol{\mathcal{B}}_{M+1}$ for which $\lambda_{\min}(\boldsymbol{\mathcal{B}}_{M+1}) = 0$. From the discussion about the relation between mutual coherence and spark in Sec. 1.1, we have that all principal $M \times M$ submatrices $\boldsymbol{\mathcal{B}}_M$ are positive definite. By the Geršgorin circles theorem, there exists an eigenvalue in the circle defined by

$$\left|\lambda - [\boldsymbol{\mathcal{B}}_M]_{ii}\right| \leq \sum_{\substack{j=0 \\ j\neq i}}^{p-1} [\boldsymbol{\mathcal{B}}_M]_{ij} \; . \tag{2.117}$$

Accordingly, in view of (1.6) and (1.7) we have

$$|\lambda - 1| \leq (M-1)\mu(\mathbf{H}) \tag{2.118}$$

Since $\boldsymbol{\mathcal{B}}_M > \mathbf{0}$, a lowerbound on the smallest eigenvalue of any principal submatrix $\boldsymbol{\mathcal{B}}_M$ is

$$\lambda_1(\boldsymbol{\mathcal{B}}_M) \geq 1 - (M-1)\mu(\mathbf{H}), \tag{2.119}$$

where $\lambda_1(\boldsymbol{\mathcal{B}}_M) \leq \lambda_2(\boldsymbol{\mathcal{B}}_M) \leq \cdots \leq \lambda_M(\boldsymbol{\mathcal{B}}_M)$. Applying the interlacing property on the Gramian itself, we have

$$\begin{aligned}
&\lambda_1(\boldsymbol{\mathcal{G}}(\mathbf{H})) \leq \lambda_1(\boldsymbol{\mathcal{B}}_{M+1}) \leq \lambda_{P-M}(\boldsymbol{\mathcal{G}}(\mathbf{H})) \;\; \Rightarrow \;\; \lambda_{P-M}(\boldsymbol{\mathcal{G}}(\mathbf{H})) \geq 0\,, \\
&\lambda_1(\boldsymbol{\mathcal{G}}(\mathbf{H})) \leq \lambda_1(\boldsymbol{\mathcal{B}}_{M}) \leq \lambda_{P-M+1}(\boldsymbol{\mathcal{G}}(\mathbf{H})) \;\; \Rightarrow \;\; \lambda_{P-M+1}(\boldsymbol{\mathcal{G}}(\mathbf{H})) \geq 1 - (M-1)\mu(\mathbf{H})\,.
\end{aligned}$$

Since $P \geq M$, $\boldsymbol{\mathcal{G}}(\mathbf{H})$ will have $P - M$ zero eigenvalues, so that

$$0 = \lambda_1(\boldsymbol{\mathcal{G}}(\mathbf{H})) = \cdots = \lambda_{P-M}(\boldsymbol{\mathcal{G}}(\mathbf{H})) \leq 1 - (M-1)\mu(\mathbf{H}) \\ \leq \lambda_{P-M+1}(\boldsymbol{\mathcal{G}}(\mathbf{H})) \leq \cdots \leq \lambda_P(\boldsymbol{\mathcal{G}}(\mathbf{H})).$$

That is, the smallest non-zero eigenvalue of the Gramian matrix of a full spark frame is lower bounded by $1 - (M-1)\mu(\mathbf{H})$. Even though these results were derived with $\mathbf{H}$ having normalized columns, for general matrices the conclusion that coherence bounds the smallest non-zero eigenvalue of its Gramian matrix still holds (with a different lowerbound, considering the proper scalings). That is, from the above reasoning, the smaller the coherence of a matrix, the larger the smallest non-zero eigenvalue of its Gramian becomes.

Note that in [16] the optimal delay for equalization is obtained by selecting a square submatrix of $\mathbf{H}$, formed by a contiguous set of columns, and whose inverse has minimum norm. Since the norm of the inverse is bounded by the reciprocal of the smallest singular value, this corresponds to selecting the submatrix of $\mathbf{H}$ with the largest smallest singular value. Thus, selecting the optimal delay also corresponds to selecting the submatrix (within a now limited subset) with the smallest coherence and consequently, the largest lowerbound for the spark (given that the non-zero eigenvalues of the Gramian of this submatrix are the square of its singular values).

Now, in the CS scenario, vector estimation is not restricted to contiguous entries. This is equivalent to saying that the best submatrix within a sensing matrix is not restricted to a set of contiguous columns, so the choice of the optimum decision delay δ within $\mathbf{x}$ is no longer relevant. In the communications setup, on the other hand, the iterative algorithm should be able to detect the correct entries that belong to a given constellation. These two phases can be cast into a unified framework that aims to adaptively (and therefore implicitly) select the best set of columns to act as a sensing matrix to the received data. In our context, the class of transmitting vectors consisting of modulated signals suggest that these are sparse with respect to any vector belonging to the constellation, and in particular, to any vector that attempts to approximate this signal, e.g., a projected estimate, say, $\mathbf{x}_{i-1}$. That is, the difference between a vector belonging to the constellation and its projected estimate is expected to be sparse. In this respect, we have that

$$\mathsf{Spark}(\mathbf{H}) = \min\left\{\|\mathbf{x} - \mathbf{x}_{i-1}\|_0 \;\; : \;\; \mathbf{H}(\mathbf{x} - \mathbf{x}_{i-1}) = \mathbf{0}\,, \quad \mathbf{x} \neq \mathbf{x}_{i-1}\right\} \tag{2.120}$$

As a consequence, enforcing this condition in any of the LS type costs aforementioned may result in further improvement in estimation (since the conditioning of a Toeplitz subblock of $\boldsymbol{H}$ may be compromised in general, we may think of precoding as one

way to design the overall equivalent sensing matrix towards a full spark one[3]).

At this point, it should be highlighted that the terms "IBI removal" and "output estimate removal", which are commonly employed as independent procedures, are in fact closely related. That is, the a priori error fed back in a re-estimation process, as e.g., in the BI-DFE aforementioned, is nothing but an attempt to remove IBI, which, if accomplished exactly through detection, inherits the name "IBI cancelation". This paves the way to a wider notion of decision feedback, which we shall pursue exactly with a CS formulation in mind.

In contrast to the block DFE discussed in the previous section, a CS-based DFE does not need to assume that the data within $\mathbf{x}$ to be estimated is a contiguous block, as defined by $\boldsymbol{x}_\delta$ in (2.44). The information of interest can be any subset of $\mathbf{x}$, while the remaining entries can be cast, either as noise, or as previously detected entries. The latter, in turn, if assumed correct, can be further used to remove the corresponding columns from the channel matrix. This can be seen as a generalization of the notion of IBI removal, prior to any re-estimation procedure. In other words, these steps can be further merged through an enhanced algorithm that detects the entire data in $\mathbf{x}$ without the need for past IBI cancellation (or combined with a fixed a priori IBI cancelation as in the conventional DFE). This may further include hybrid architectures that make use of transmitted redundancy. As a fallout, a procedure that attempts to retrieve $\mathbf{x}$, if successful, will imply that not only redundancy in transmission is unecessary, but also that the received signal can be sampled at a lower rate than what is predicted by Nyquist theory.

2.10.1 Adaptive CS-based Formulation

One way towards solving the underlying underdetermined problem is to account for a priori knowledge on the parameter $\mathbf{x}$ itself, by adding a suitable constraint on the vector parameter $\mathbf{x}$. This is equivalent to introducing a regularization function $\varrho(\mathbf{x})$ into the cost (2.13):

$$\min_{\mathbf{x}} \ \ \|\mathbf{y} - \boldsymbol{H}\mathbf{x}\|^2_{\boldsymbol{R}_v^{-1}} \ + \ \|\mathbf{x} - \mathbf{x}_{i-1}\|^2_{\mathbf{P}_{i-1}^{-1}} \ + \ \varrho(\mathbf{x}) \tag{2.121}$$

where $\varrho(\cdot)$ is some regularization or penalty function (not neccessarily smooth). Note that this problem corresponds to (1.13), by setting $\boldsymbol{\mathcal{W}} \longleftarrow \boldsymbol{R}_{\boldsymbol{v}}^{-1}$ and $\boldsymbol{\mathcal{R}} \longleftarrow \mathbf{P}_{i-1}^{-1}$. By realizing that $\mathbf{x}$ is sparse with respect to any vector $\mathbf{z}$ that belongs to the same constellation, $\varrho(\cdot)$ can be chosen to be a sparsity inducing regularization function in terms of $\mathbf{x} - \mathbf{z}$. Note that in particular, by setting $\mathbf{z} = \mathbf{x}_{i-1}$, we can express our

[3]This may further shift the load of performing Hessian inversions in decision directed iterations to the transmitter, so that in some cases, a simpler (block) LMS type algorithm can be a simpler choice.

uncertainty about some previous estimate of $\mathbf{x}$ in the ℓ_0- or ℓ_1-norm sense, commonly employed in CS problems, in addition to the ℓ_2-norm regularization. In this case, (2.121) can be explicitly stated as

$$\min_{\mathbf{x}} \ \|\boldsymbol{y}-\boldsymbol{H}\mathbf{x}\|^2_{\boldsymbol{R}_v^{-1}} \ + \ \|\mathbf{x}-\mathbf{x}_{i-1}\|^2_{\mathbf{P}_{i-1}^{-1}} \ + \ \varrho(\mathbf{x}-\mathbf{x}_{i-1}) \tag{2.122}$$

By rewriting the first two terms of (2.122) in a standard completion-of-squares form, and after ignoring the remaining constant $\|\boldsymbol{y}\|^2_{(\boldsymbol{R}_v+\boldsymbol{H}\mathbf{P}_{i-1}\boldsymbol{H}^*)^{-1}}$, we can express it equivalently as

$$\min_{\mathbf{x}} \ \|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\left(\mathbf{P}_{i-1}^{-1}+\boldsymbol{H}^*\boldsymbol{R}_v^{-1}\boldsymbol{H}\right)} \quad + \quad \varrho(\mathbf{x}-\mathbf{x}_{i-1}) \tag{2.123}$$

$$\equiv \ \min_{\mathbf{x}} \ \|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\widehat{\mathbf{P}}_i^{-1}} \ + \ \varrho(\mathbf{x}-\mathbf{x}_{i-1}) \tag{2.124}$$

where

$$\widehat{\mathbf{P}}_i^{-1} = \mathbf{P}_{i-1}^{-1} + \boldsymbol{H}^*\boldsymbol{R}_v^{-1}\boldsymbol{H}$$

and with $\hat{\mathbf{x}}_i$ computed via (2.14)–(2.17).

Now, exact minimization of the above iterative problem is only possible in case $\widehat{\mathbf{P}}_i$ is diagonal, since in this case, minimization can be carried out entrywise, yielding a solution in closed form (see Appendix A, [23]). The presence of the non-diagonal weighting matrix, in general, hinders the direct extension of this algorithm to a true recursive LS update, which calls for an alternative path.

One way to circumvent this problem is by taking advantage of the *norm equivalence property*, which ensures that we can bound $\|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\widehat{\mathbf{P}}_i^{-1}}$ in the following manner:

$$r_{1,i}\,\|\mathbf{x}-\hat{\mathbf{x}}_i\|^2 \le \|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\widehat{\mathbf{P}}_i^{-1}} \le r_{2,i}\,\|\mathbf{x}-\hat{\mathbf{x}}_i\|^2$$

for some non-negative constants $\{r_{1,i}, r_{2,i}\}$. Since $\widehat{\mathbf{P}}_i^{-1}$ is a Hermitian positive-definite matrix, this property can be made more explicit, by using the Rayleigh-Ritz characterization of eigenvalues (λ_k) as

$$\lambda_{\min}\left(\widehat{\mathbf{P}}_i^{-1}\right)\|\mathbf{x}-\hat{\mathbf{x}}_i\|^2 \le \|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\widehat{\mathbf{P}}_i^{-1}} \le \lambda_{\max}\left(\widehat{\mathbf{P}}_i^{-1}\right)\|\mathbf{x}-\hat{\mathbf{x}}_i\|^2 .$$

As a consequence, we can approximate the weighted norm in (2.124) by

$$\|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\mathbf{P}_i^{-1}} \ \approx \ c_i\,\|\mathbf{x}-\hat{\mathbf{x}}_i\|^2 \ = \ c_i\sum_{k=0}^{M-1}|x(k)-\hat{x}_i(k)|^2$$

where c_i is an iteration dependent constant. Because of the norm equivalence property, we can take a step further and assign different weights for each term

of the summation, thus leading to an even better approximation

$$\|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\mathbf{P}_i^{-1}} \;\approx\; \|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\mathbf{C}_i} \tag{2.125}$$

where $\mathbf{C}_i = \mathsf{Diag}\,(c_{0,i},\cdots,c_{M-1,i})$ is a diagonal matrix, where all diagonal entries are bounded by $\lambda_{\min}\left(\mathbf{P}_i^{-1}\right) \leq c_{k,i} \leq \lambda_{\max}\left(\mathbf{P}_i^{-1}\right)$. Moreover, hereafter, we shall consider a weighted ℓ_0-norm regularization denoted by

$$\varrho(\mathbf{x}-\mathbf{x}_{i-1}) = \|\mathbf{x}-\mathbf{x}_{i-1}\|_{0,\boldsymbol{\Lambda}_i}$$

with $\boldsymbol{\Lambda}_i = \mathsf{Diag}\left(\epsilon'_{i,0},\cdots,\epsilon'_{i,M-1}\right)$. In this case, we have

$$\|\mathbf{x}-\mathbf{x}_{i-1}\|_{0,\boldsymbol{\Lambda}_i} = \sum_{k=0}^{M-1} \epsilon'_{i,k}|x(k)-x_{i-1}(k)|_0, \tag{2.126}$$

with $|x(k)-x_{i-1}(k)|_0$ denoting an indicator function that returns 0 whenever we have $x(k)-x_{i-1}(k)=0$, and 1 otherwise, and with $\epsilon'_{i,k}$ the corresponding weighting scalars. As a consequence, the solution to the problem in (2.124) can be obtained by solving

$$\min_{\mathbf{x}} \quad \|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\mathbf{C}_i} \;+\; \|\mathbf{x}-\mathbf{x}_{i-1}\|_{0,\boldsymbol{\Lambda}_i}. \tag{2.127}$$

Since $\mathbf{C}_i$ is diagonal, the solution $\dot{\mathbf{x}}_i$, as a proximal mapping, can be computed exactly as (see App. A) :

$$\begin{aligned}\dot{\mathbf{x}}_i &= \mathbf{x}_{i-1} + q_i(\hat{\mathbf{x}}_i - \mathbf{x}_{i-1}) &(2.128)\\ &= \mathbf{x}_{i-1} + q_i(\boldsymbol{K}_{p,i}\bar{\mathbf{e}}_i). &(2.129)\end{aligned}$$

where $q_i(\cdot)$ is now the entrywise hard threshold operator

$$q_i(\theta_k) = \begin{cases}\theta_k, & |\theta_k| > \sqrt{\epsilon_{i,k}}\\ 0, & \text{otherwise,}\end{cases} \tag{2.130}$$

and where we write $\epsilon_{i,k} = \epsilon'_{i,k}/c_{i,k}$ to denote a threshold scalar that incorporates the weights from $\mathbf{C}_i$ and the regularization constants $\epsilon'_{i,k}$.

At this point, exact knowledge of the weights $\epsilon'_{i,k}$ that will lead to the scalars $\epsilon_{i,k}$ is not important. This is because, as we shall see, the underlying thresholds will be calculated adaptively, so that in this sense, the approximation in (2.125), could be replaced by an equality, without the need to establish $\epsilon'_{i,k}$ or $c_{i,k}$ in first place. Those values will be obtained implicitly through a thresholding strategy. We shall return to this issue further ahead, in light of the new recursions introduced next.

Observe that $q_i(\cdot)$ is iteration dependent, and can be represented by a diagonal matrix $\boldsymbol{\Delta}_i$ that selects the surviving entries of its vector argument, while zeroing out the remaining ones. That is, defining $\boldsymbol{\Delta}_{i,k}$ its k-th diagonal entry,

$$\boldsymbol{\Delta}_{i,k} = \begin{cases} 1, & |\theta_k| > \sqrt{\epsilon_{i,k}} \\ 0, & \text{otherwise.} \end{cases} \tag{2.131}$$

so that from (2.129) we have

$$\dot{\mathbf{x}}_i \;=\; \mathbf{x}_{i-1} + \boldsymbol{\Delta}_i \boldsymbol{K}_{p,i} \bar{\mathbf{e}}_i \,. \tag{2.132}$$

In words, with the given thresholds, when a symbol is considered incorrect, we simply set $\boldsymbol{\Delta}_{i,k} = 1$, so that $\dot{x}_i(k) = \hat{x}_i(k)$. On the other hand, when the symbol is considered correct, no further update is required for that entry, and $\dot{x}_i(k) = x_{i-1}(k)$. This implies that we should set to zero its corresponding error covariance elements, while updating the remaining ones. That is, from (2.132), the error covariance of the new estimate before the slicer, denoted as $\dot{\mathbf{P}}_i$, reads

$$\dot{\mathbf{P}}_i \;=\; \mathbf{P}_{i-1} - \boldsymbol{\Delta}_i \mathbf{K}_{p,i} \boldsymbol{R}_{\bar{e},i} \mathbf{K}_{p,i}^* \boldsymbol{\Delta}_i^* \,. \tag{2.133}$$

Finally, the projection onto the constellation symbol yields

$$\mathbf{x}_i \;=\; f_i(\dot{\mathbf{x}}_i) \tag{2.134}$$

Note that once the k-th entry of $\mathbf{x}_i$ is assumed correct, its corresponding variance becomes zero. On the other hand, the incorrect entries are projected onto constellation points. With this in mind, it remains to compute the error covariance of the signal obtained after the slicer. To this end, we can adopt a first-order model for $f(\cdot)$ as follows:

$$\mathbf{x}_i = \dot{\mathbf{x}}_i + \boldsymbol{\Delta}_i \mathbf{q}_i \tag{2.135}$$

where $\mathbf{q}_i$ is a zero-mean random perturbation vector, with i.i.d. entries whose distribution depends on the input signal modulation. For example, if a M-ary QAM is used, then $\mathbf{q}_i(k)$ is uniformly distributed inside of a square on the complex plane, with sides equal to the minimum distance between symbols. The product with the diagonal matrix $\boldsymbol{\Delta}_i$ reflects the fact that the slicer operates solely on the incorrect entries, leaving the remaining ones undisturbed. Hence, denoting the covariance

matrix of the perturbation vector as $\mathbf{Q}_i$, we get

$$\begin{aligned}
\mathbf{P}_i &= \boldsymbol{\Delta}_i \dot{\mathbf{P}}_i \boldsymbol{\Delta}_i^* + \boldsymbol{\Delta}_i \mathbf{Q}_i \boldsymbol{\Delta}_i^* && (2.136)\\
&= \boldsymbol{\Delta}_i \mathbf{P}_{i-1} \boldsymbol{\Delta}_i^* - \boldsymbol{\Delta}_i \mathbf{K}_{p,i} \boldsymbol{R}_{\bar{e},i} \mathbf{K}_{p,i}^* \boldsymbol{\Delta}_i^* + \boldsymbol{\Delta}_i \mathbf{Q}_i \boldsymbol{\Delta}_i^* && (2.137)\\
&= \boldsymbol{\Delta}_i \left(\mathbf{P}_{i-1} - \mathbf{K}_{p,i} \boldsymbol{R}_{\bar{e},i} \mathbf{K}_{p,i}^* + \mathbf{Q}_i \right) \boldsymbol{\Delta}_i^* && (2.138)
\end{aligned}$$

Eqs. (2.14)-(2.19) should then be replaced by

$$\begin{aligned}
\boldsymbol{R}_{\bar{e},i} &= \boldsymbol{R}_{\boldsymbol{v}} + \boldsymbol{H} \mathbf{P}_{i-1} \boldsymbol{H}^* , && (2.139)\\
\mathbf{K}_{p,i} &= \mathbf{P}_{i-1} \boldsymbol{H}^* \boldsymbol{R}_{\bar{e},i}^{-1} , && (2.140)\\
\bar{\mathbf{e}}_i &= \mathbf{y} - \boldsymbol{H} \mathbf{x}_{i-1} , && (2.141)\\
\dot{\mathbf{x}}_i &= \mathbf{x}_{i-1} + \boldsymbol{\Delta}_i \boldsymbol{K}_{p,i} \bar{\mathbf{e}}_i , \qquad \mathbf{x}_0 = \mathbf{0} && (2.142)\\
\mathbf{x}_i &= f_i(\dot{\mathbf{x}}_i) && (2.143)\\
\mathbf{P}_i &= \boldsymbol{\Delta}_i \left(\mathbf{P}_{i-1} - \mathbf{K}_{p,i} \boldsymbol{R}_{\bar{e},i} \mathbf{K}_{p,i}^* + \mathbf{Q}_i \right) \boldsymbol{\Delta}_i^* , \qquad \dot{\mathbf{P}}_0 = \mathbf{R}_{\mathbf{x}} && (2.144)
\end{aligned}$$

As a result, given $\epsilon_{i,k}$, we can gradually reduce the size of our channel model as follows. Let $\boldsymbol{x}_i^c$ denote the entries of $\mathbf{x}_i$ which correspond to zero error, i.e., $\boldsymbol{x}_i^c = \dot{\mathbf{x}}_i(I_{c_i}) = \mathbf{x}_{i-1}(I_{c_i})$, and where we define the set of indexes of correct decisions as $I_{c_i} = \{k \in \mathbb{Z} \,|\, |\dot{x}_i(k) - \dot{x}_{i-1}(k)| < \epsilon_{i,k}\}$. Likewise, let $\boldsymbol{x}_i$ be the vector comprising the incorrect entries, having error covariance $\boldsymbol{P}_i$. Note that the effect of $\boldsymbol{x}_{i-1}^c$ in the a priori error $\bar{\mathbf{e}}_i$ is to succesively remove IBI from the original model.

Assume that at the i-th iteration we have removed the columns of $\boldsymbol{H}$ that correspond to correct decisions, and define $\boldsymbol{H}_i^c$ the matrix that collects such columns at time i. Likewise, let $\boldsymbol{H}_i$ be the matrix comprising the surviving columns. Then, observe that $\bar{\mathbf{e}}_i$ can be expressed as

$$\begin{aligned}
\bar{\mathbf{e}}_i &= \mathbf{y} - \boldsymbol{H}_i^c \boldsymbol{x}_{i-1}^c - \boldsymbol{H}_i \boldsymbol{x}_{i-1} && (2.145)\\
&= \mathbf{y}_i - \boldsymbol{H}_i \boldsymbol{x}_{i-1} && (2.146)
\end{aligned}$$

where we have further defined

$$\mathbf{y}_i \triangleq \mathbf{y} - \boldsymbol{H}_i^c \boldsymbol{x}_{i-1}^c$$

Because the entries in $\boldsymbol{x}_{i-1}^c$ contain past assumed correct decisions that did not change from the previous iterations, we can still write

$$\begin{aligned}
\mathbf{y}_i &= \mathbf{y} - \boldsymbol{H}_i^c \boldsymbol{x}_{i-1}^c && (2.147)\\
&= \mathbf{y} - \boldsymbol{H}_{i-1}^c \boldsymbol{x}_{i-2}^c - \mathring{\boldsymbol{H}}_i \mathring{\boldsymbol{x}}_{i-1} && (2.148)\\
&= \mathbf{y}_{i-1} - \mathring{\boldsymbol{H}}_i \mathring{\boldsymbol{x}}_{i-1} && (2.149)
\end{aligned}$$

where $\mathring{\boldsymbol{x}}_{i-1}$ corresponds to new correct decisions added to the set I_{c_i}. At each iteration, a matrix $\boldsymbol{J}_i$ is formed such that it selects the incorrectly detected entries of the current LS estimate $\boldsymbol{x}_i$. Similarly, the correct entries are collected into $\mathring{\boldsymbol{x}}_i$ using a matrix $\boldsymbol{J}_i^c$, complementary to $\boldsymbol{J}_i$. From these findings, Eqs. (2.139)-(2.144) collapse to the CS-based DFE algorithm in Table 2.3.

Initialization:

$\boldsymbol{H}_0 = \boldsymbol{H},\ \boldsymbol{P}_0 = \boldsymbol{R}_{\mathbf{x}},\ \boldsymbol{x}_0 = \mathbf{x}_0 = \mathbf{0},\ \mathbf{y}_0 = \mathbf{y}$

$$\boldsymbol{R}_{\bar{e},i} = \boldsymbol{R}_{\boldsymbol{v}} + \boldsymbol{H}_i \boldsymbol{P}_{i-1} \boldsymbol{H}_i^* \tag{2.150}$$

$$\boldsymbol{K}_{p,i} = \boldsymbol{P}_{i-1} \boldsymbol{H}_i^* \boldsymbol{R}_{\bar{e},i}^{-1} \tag{2.151}$$

$$\mathbf{y}_i = \mathbf{y}_{i-1} - \mathring{\boldsymbol{H}}_i \mathring{\boldsymbol{x}}_{i-1}\ , \tag{2.152}$$

$$\bar{\mathbf{e}}_i = \mathbf{y}_i - \boldsymbol{H}_i \boldsymbol{x}_{i-1} \tag{2.153}$$

$$\boldsymbol{x}_i = f_i\left(\boldsymbol{J}_i \boldsymbol{x}_{i-1} + \boldsymbol{J}_i \boldsymbol{K}_{p,i} \bar{\mathbf{e}}_i\right) \tag{2.154}$$

$$\mathring{\boldsymbol{x}}_i = \boldsymbol{J}_i^c \boldsymbol{x}_{i-1} \tag{2.155}$$

$$\boldsymbol{P}_i = \boldsymbol{J}_i \left(\boldsymbol{P}_{i-1} - \boldsymbol{K}_{p,i} \boldsymbol{R}_{\bar{e},i} \boldsymbol{K}_{p,i}^* + \mathbf{Q}_i\right) \boldsymbol{J}_i^* \tag{2.156}$$

Table 2.3: CS-based Kalman algorithm.

Note that the size of the data matrix $\boldsymbol{H}_i$, as well as $\boldsymbol{x}_i$, recursively shrinks; accordingly, the channel model at time i yields an error covariance $\boldsymbol{P}_i$ with reduced size as well.

2.10.2 Computation of the Threshold

A critical factor in deciding which of the detected entries should be taken as correct, relies heavily on the adaptive choice for the threshold scalars $\epsilon_{i,k}$, which are used by $q_i(\hat{\mathbf{x}}_i - \mathbf{x}_{i-1}) = q_i(\boldsymbol{K}_{p,i}\bar{\mathbf{e}}_i)$ in (2.130). One way towards its derivation, which was roughly pursued in [62] (according to their own residual error definition), consists in the following. Let the centralized estimate at time i be defined as

$$\Delta\hat{\mathbf{x}}_i = \hat{\mathbf{x}}_i - \mathbf{x}_{i-1} \tag{2.157}$$

$$= \mathbf{x} - \tilde{\mathbf{x}}_i - \mathbf{x}_{i-1} \tag{2.158}$$

$$= \bar{\mathbf{x}}_{i-1} - \tilde{\mathbf{x}}_i = \boldsymbol{K}_{p,i}\bar{\mathbf{e}}_i \tag{2.159}$$

We readily notice that the error after detection, $\bar{\mathbf{x}}_{i-1}$, is a sparse vector containing a few spikes that correspond to the incorrect entries, while $\tilde{\mathbf{x}}_i$ is simply the MMSE before detection.

The approach followed in [62] (albeit one lacking the level of generality introduced here), is to consider $\tilde{\mathbf{x}}_i$ as a Gaussian vector noise, in which the spikes of $\bar{\mathbf{x}}_{i-1}$ are

embedded. Note that because this variable indeed defines an MMSE estimation error, if $\mathbf{x}$ and $\mathbf{y}$ are jointly Gaussian, this assumption becomes exact. Regardless of this fact, the idea is based on a well known result, which states that the mean of the largest magnitude sample of a Gaussian random vector $\mathbf{z}_i$ of length M, say, $z_i(k) \sim \mathcal{CN}(0, \sigma_z^2)$, $k = 0, 1, \ldots, M-1$, is bounded by [63]

$$\mathsf{E}\|\mathbf{z}_i\|_\infty < \sqrt{2\sigma_z^2 \ln M} \tag{2.160}$$

By making the association $\mathbf{z}_i \longleftarrow \tilde{\mathbf{x}}_i$, and considering this variable as background noise, this suggest a possible way to distinguish the error spikes in $(\hat{\mathbf{x}}_i - \mathbf{x}_{i-1})$ from $\tilde{\mathbf{x}}_i$ such that any entry of $\tilde{\mathbf{x}}_i$ with magnitude falling below $\sqrt{2\sigma_z^2 \ln M}$ can be considered a fluctuation around a correct decision, and therefore are not in error. Such reasoning was pursued in [62], assuming that $\mathbf{z}_i$ has Gaussian entries with the same variance σ_z^2. Although the Gaussianity assumption can be approximated, in general, the variances along the error vector are not identical, so that the application of this result becomes very restricted. That is, since each element of $\mathbf{z}_i$ has a different variance, the corresponding upper bound for the maximum absolute value must be replaced by

$$\mathsf{E}\|\tilde{\mathbf{x}}_i\|_\infty < \sqrt{2\sigma_{i,\max}^2 \ln M}, \tag{2.161}$$

where we defined

$$\sigma_{i,\max}^2 \;=\; \max_{0 \le k \le M-1} \sigma_{i,k}^2$$

Having such a large upper bound for the entries $\tilde{x}_i(k)$ with small variances could force correct symbols to be rejected. Having different thresholds for each symbol is therefore a more plausible assertion.

Specifically, from (2.159), we have that

$$\tilde{\mathbf{x}}_i \;=\; \bar{\mathbf{x}}_{i-1} \;-\; \boldsymbol{K}_{p,i}\bar{\mathbf{e}}_i \tag{2.162}$$

and we already know that it yields the covariance recursion (2.18), i.e.,

$$\widehat{\boldsymbol{P}}_i \;=\; \boldsymbol{P}_{i-1} - \boldsymbol{K}_{p,i}\boldsymbol{R}_{\bar{e},i}\boldsymbol{K}_{p,i}^* \tag{2.163}$$

Since the diagonal elements of $\widehat{\boldsymbol{P}}_i$ contains the variances $\sigma_{i,k}^2$ of each entry of $\tilde{\mathbf{x}}_i$, this suggests that they can be used more precisely towards the computation of a threshold.

Let the minimum distance between two constellation points be denoted by $d_{\min}$, as depicted in Fig. 2.11, and observe that either $\bar{x}_{i-1}(k) = 0$ or $|\bar{x}_{i-1}(k)| \ge d_{\min}$, meaning correct or incorrect symbol detection, respectively. We remark that in [62], it is further assumed that most non-zero entries of $\bar{\mathbf{x}}_{i-1}$ are such that

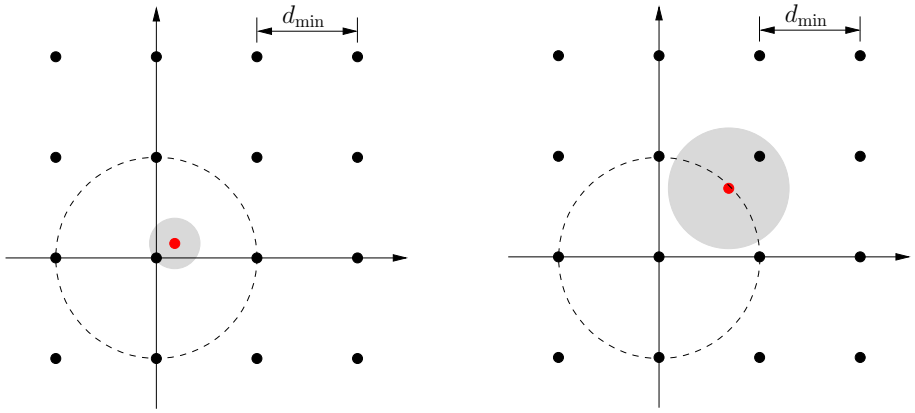


Figure 2.11: A (a) correctly detected and a (b) incorrectly detected symbol are shown. Black dots show the possible values of $\bar{x}_{i-1}(k)$, given an estimate $x_{i-1}(k)$. The red dot is the current update $\Delta\hat{x}_i(k)$ and the gray circle corresponds to the possible values of $\Delta\hat{x}_i(k) + \tilde{x}_i(k)$ given the variance $\sigma^2_{i,k}$. The dashed circle has radius $d_{\min}$.

$\max[\bar{x}_{i-1}(k)] = d_{\min}$, an assumption needed to estimate the sparsity of $\bar{\mathbf{x}}_{i-1}$ in their scenario. Rearranging (2.159), we have that

$$\bar{x}_{i-1}(k) \;=\; \tilde{x}_i(k) \;+\; \Delta\hat{x}_i(k), \tag{2.164}$$

so that applying the triangular inequality to this relation, we get

$$|\tilde{x}_i(k)| \;+\; |\Delta\hat{x}_i(k)| \;\geq\; |\bar{x}_{i-1}(k)|. \tag{2.165}$$

Since incorrect detection implies $|\bar{x}_{i-1}(k)| \geq d_{\min}$, using this fact along with (2.165) gives

$$|\tilde{x}_i(k)| \;+\; |\Delta\hat{x}_i(k)| \;\geq\; d_{\min} \tag{2.166}$$

or, equivalently,

$$|\tilde{x}_i(k)| \;\;\geq\;\; d_{\min} - |\Delta\hat{x}_i(k)| \tag{2.167}$$

For example, in Fig 2.11(a), the grey circle around the red dot corresponding to $\Delta\hat{x}_i$ would not cross the dashed circle with radius $d_{\min}$, corresponding to a correctly detected symbol. In Fig 2.11(b), the situation for an incorrectly detected symbol is depicted.

In the following, we shall assume that $\tilde{x}_i(k)$ is circular complex Gaussian, for which it is well known that $|\tilde{x}_i(k)|$ is Rayleigh distributed [64, 65]. Then, using its corresponding cumulative probability function, the inequality in (2.167) will be true

with probability

$$\Pr\left(\,|\tilde{x}_i(k)| \geq d_{\min} - |\Delta\hat{x}_i(k)|\right) = e^{-\frac{(d_{\min}-|\Delta\hat{x}_i(k)|)^2}{2\sigma^2_{i,k}}}. \tag{2.168}$$

Let the maximum probability of incorrect symbol detection be denoted by p_k. Then, (2.168) gives

$$e^{-\frac{(d_{\min}-|\Delta\hat{x}_i(k)|)^2}{2\sigma^2_{i,k}}} \quad \leq \quad p_k \tag{2.169}$$

which yields

$$|\Delta\hat{x}_i(k)| \quad \leq \quad d_{\min} - \sqrt{2\sigma^2_{i,k}\ln\frac{1}{p_k}} \tag{2.170}$$

Since p_k quantifies the probability of false detection, the smaller p_k, the more restrictive will be the assertion of correct decision. For instance, selecting $p_k \to 0$ would imply $|\Delta\hat{x}_i(k)| \to -\infty$ in (2.170), meaning that all symbols are considered incorrect. On the other hand, having a large p_k would set the threshold too large, so that any symbol could be taken as correct. One reasonable choice is $p_k = {}^1\!/_M$, which corresponds to allowing at most one symbol in error within M symbols, considering they are equally likely to be correct. Observe that applying the upper bound $|\tilde{x}_i(k)| \leq \mathsf{E}\|\tilde{\mathbf{x}}_i\|_\infty$ to (2.167), would imply correct detection whenever

$$|\Delta\hat{x}_i(k)| \quad \leq \quad d_{\min} - \sqrt{2\sigma^2_{i,\max}\ln M}\,, \tag{2.171}$$

which follows the same reasoning of the threshold in (2.170) when $p_k = {}^1\!/_M$.

An alternative threshold-based strategy (as opposed to the one based on the incorrect detection alone) can be devised considering a likelihood test. From the basics of hypothesis testing, the Neyman-Pearson lemma establishes that the likelihood ratio constitutes the most powerful test for a given significance level [66, 67], defined as the probability of rejecting a correct symbol. In that case, a symbol is considered correct if

$$\frac{\Pr(\bar{x}_{i-1}(k)=0)}{\Pr(|\bar{x}_{i-1}(k)| \geq d_{\min})} \gg 1. \tag{2.172}$$

Since $|\bar{x}_{i-1}(k)| \geq d_{\min}$ implies $|\tilde{x}_i(k)| \geq d_{\min} - |\Delta\hat{x}_i(k)|$ (although the converse is not true), we have that

$$\Pr(|\bar{x}_{i-1}(k)| \geq d_{\min}) \quad < \quad \Pr(|\tilde{x}_i(k)| \geq d_{\min} - |\Delta\hat{x}_i(k)|), \tag{2.173}$$

and therefore,

$$\frac{\Pr(\bar{x}_{i-1}(k)=0)}{\Pr(|\bar{x}_{i-1}(k)|\geq d_{\min})} > \frac{\Pr(\bar{x}_{i-1}(k)=0)}{\Pr(|\tilde{x}_i(k)|\geq d_{\min}-|\Delta\hat{x}_i(k)|)}. \tag{2.174}$$

Thus if

$$\frac{\Pr(\bar{x}_{i-1}(k)=0)}{\Pr(|\tilde{x}_i(k)|\geq d_{\min}-|\Delta\hat{x}_i(k)|)} > \alpha \tag{2.175}$$

for some likelihood ratio $\alpha \gg 1$, the condition in eq. (2.172) is also established.

From (2.164), when a symbol is detected correctly, $\bar{x}_{i-1}(k)=0$ and

$$\tilde{x}_i(k) \;=\; -\Delta\hat{x}_i(k). \tag{2.176}$$

The probability of correct decision then reads

$$\Pr(\tilde{x}_i(k)=-\Delta\hat{x}_i(k)) = \frac{1}{2\pi\sigma_{i,k}^2} e^{-\frac{|\Delta\hat{x}_i(k)|^2}{2\sigma_{i,k}^2}} \tag{2.177}$$

so that plugging (2.168) and (2.177) into (2.175), we get

$$\frac{1}{2\pi\sigma_{i,k}^2} e^{-\frac{|\Delta\hat{x}_i(k)|^2}{2\sigma_{i,k}^2}} \;>\; \alpha e^{-\frac{(d_{\min}-|\Delta\hat{x}_i(k)|)^2}{2\sigma_{i,k}^2}}. \tag{2.178}$$

Finally, after some algebra, we have that for a likelihood ratio α, under the condition that

$$|\Delta\hat{x}_i(k)| \;<\; \frac{d_{\min}}{2} - \frac{\sigma_{i,k}^2}{d_{\min}} \ln(2\alpha\pi\sigma_{i,k}^2) \tag{2.179}$$

we can assume that the symbol has been correctly detected.

Although the Neyman-Pearson lemma also establishes a way to derive α directly from the significance level [67], this is not possible, since the probability distribution of $|\Delta\hat{x}_i(k)|$ is not known *a priori*. We verified that the above criterion is robust with respect to the choice of α, which allowed us to choose it experimentally with good results.

We remark that the proposed algorithm can be readily extended to use a WL formulation, by replacing $\boldsymbol{H}$ with its extended definition $\boldsymbol{H}_e$. Although in that case $\tilde{x}(k)$ is a real quantity and its magnitude $|\tilde{x}(k)|$ follows a half-normal distribution, the Rayleigh c.d.f can still be used as an upper bound for its probability. That is, the threshold strategy based on the likelihood test still applies in that case, considering a proper scaling of α.

2.11 Related Works

An algorithm that shares some resemblance with the recursions presented here has been proposed in [62], in the context of a DFE applied to OFDM related models, in the specific case of square unstructured channel matrices. Those however, exhibit several fundamental differences from what is proposed in this work:

(**A**) First, the DFE in [62] appears as a pre-defined structure, to which CS is applied to the estimation error in an roughly ad-hoc manner. Here, in contrast, we show that a DFE is itself the result of a CS formulation, from which, aside from the fixed threshold function, the algorihm arises naturally as a solution of a regularized problem; Both classes of recursions considered in this work provide improved performance compared to the one in [62];

(**B**) The algorithm in [62] is not adaptive in the strict sense. Note that while we update the optimal solution from the previous estimate, their approach follows a re-estimation of $\mathbf{x}_i$ afresh at every iteration; this is in part due to the fact that the interpretation of their variables as actual MMSE quantities is missing, yet, commonly found in an MMSE adaptive scenario;

(**C**) Their threshold $q_i(\cdot)$ is applied to a centralized estimate of $\mathbf{x}$, and in contrast to our approach, assumes the form $q_i(\hat{\mathbf{x}}_i - \mathbf{x}_{i-1}) = q_i(\boldsymbol{H}_i^* \bar{\mathbf{e}}_i)$ instead of relying on its true MMSE-based estimate $q_i(\boldsymbol{K}_{p,i}\bar{\mathbf{e}}_i)$ defined in (2.129);

(**D**) The assumptions used when defining a threshold for $q_i(\cdot)$ differ considerably from ours. In particular, we arrive at an expression that makes use of independent thresholds for every entry of the estimated vector, instead of a single one applied to all entries. This will further impact the performance of the algorithm in comparison to what is proposed in [62];

(**E**) The algorithm in [62] was devised to deal with square or tall transmission matrices; Here, we have focused mainly on the more difficult problem of generally underdetermined systems, and in particular, on Toeplitz-like sensing matrices. This is a much more difficult scenario, and the new algorithms still outperform the ones in [62];

(**F**) The recursions herein are further proposed under a WL formulation, for which additional gains can be achieved, in terms of throughput and/or BER. For example, in the case of real constellations, we shall see that conditioning is further improved compared to SL solutions;

We remark that the algorithm in [62] was inspired by the StOMP [8] recursions for CS. Specifically, StOMP also relies on the matched filter output $\boldsymbol{H}_i^* \bar{\mathbf{e}}_i$ in order

to select the entries to be included in the support of the solution. In StOMP, one threshold value is used for all entries, under the assumption that the estimation errors are Gaussian with the same variance.

In [25] the *Complex Adaptive Reweighting Homotopy* (C-ARH) algorithm was introduced to reduce the complexity of *Robust Capon Beamformers* (RCB), where, traditionally, an eigenvalue decomposition is used to determine the regularization needed to overcome the ill-conditioning of the input signal covariance matrix. It is then further extended to an iterative version which operates on the updates of the beamformer coefficients, namely the *Iterative C-ARH* (It-C-ARH). Since the interfering sources in that context are supposed to vary slowly compared to the snapshot sampling frequency, such updating vectors exhibit some level of sparsity, which benefit from a CS-based approach. The It-C-ARH algorithm differs from ours in the following points:

(**A**) Instead of a ℓ_0-norm regularization, the It-C-ARH solves a weighted ℓ_1-norm regularized LS problem, with distinct regularizers for each entry of the update vector;

(**B**) The LS cost in [25] is not quadratically regularized, and the ℓ_2-norm term is not weighted;

(**C**) It relies on homotopies, where the penalty in the coefficients are initially chosen such that it yields a zero solution, and then reduced gradually until convergence; at each step, a line search finds the optimally penalized coefficients that includes or removes a single entry in the support of the solution. That is, a solution is obtained by gradually reducing the weights in the ℓ_1-norm term. This is in contrast to our approach, where the relation between the adaptive penalties and the thresholds is accounted for implicitly, and does not need to be defined explicitly in the cost function.

2.12 Simulations

In order to assess the performance of the proposed CS-based Kalman iterations, we provide a number of experiments comparing the new algorithm against similar existing transceivers, considering that there are several parameters that need to be properly tunned.

Experiment 1 - Choice of the threshold parameter α

The threshold strategy used in the CS-based Kalman algorithm depends on a parameter α used by the likelihood test. As previously stated, finding its optimal value depends on the prior knowledge of the true probability density of the updated estimate, which is not available. Nevertheless, we verify that the likelihood strategy is robust with respect to the choice of α.

In Figs. 2.12-2.16 we verify the effect of varying α in the CS-based Kalman iterations, for different channel lenghts and decision-delays. All experiments employ $M = 64$ symbols from QAM-4 constellations, and were run for an ensemble of 900 random channels, with 30 block transmissions for each channel realization, whose taps were drawn from an i.i.d. complex Gaussian distribution. For comparison, we consider the resulting BER of the 'Sparsity Enhanced DFE' proposed in [62], since it follows similar philosophy for the probabilistic threshold strategy, and the BI-GDFE from [11] (which is the original BI-DFE algorithm). Because here we are dealing with the more challenging case of 'fat' and square Toeplitz matrices, it is expected that the Sparsity Enhanced DFE will not perform well, as it has been designed for tall, and circulant matrices. In these experiments, we used a ZP-ZJ approach in order to remove the IBI.

Figure 2.12 depicts the case of a full zero-jamming ($\delta = 0$) scheme, for random channels of length $L = 15$. The Kalman CS-based iterations outperform the BI-GDFE, even though the respective BER is high for all values E_b/N_0. It is interesting to note that any value of $\alpha > 2$ has little effect in performance, although $\alpha = 10$ gives the best overall response. Performance degrades when the channel length is increased, as shown in Fig. 2.13 for $L = 31$, where the most accurate detections occur for $\alpha = 2$ and $\alpha = 5$. In both cases, the high BER is caused by the noise amplification due the ill-conditioning of the channel matrix, which further worsens for longer channels. The likelihood test becomes less effective as the magnitude of the errors $|\tilde{x}(k)|$ can become fairly large even for correctly detected entries.

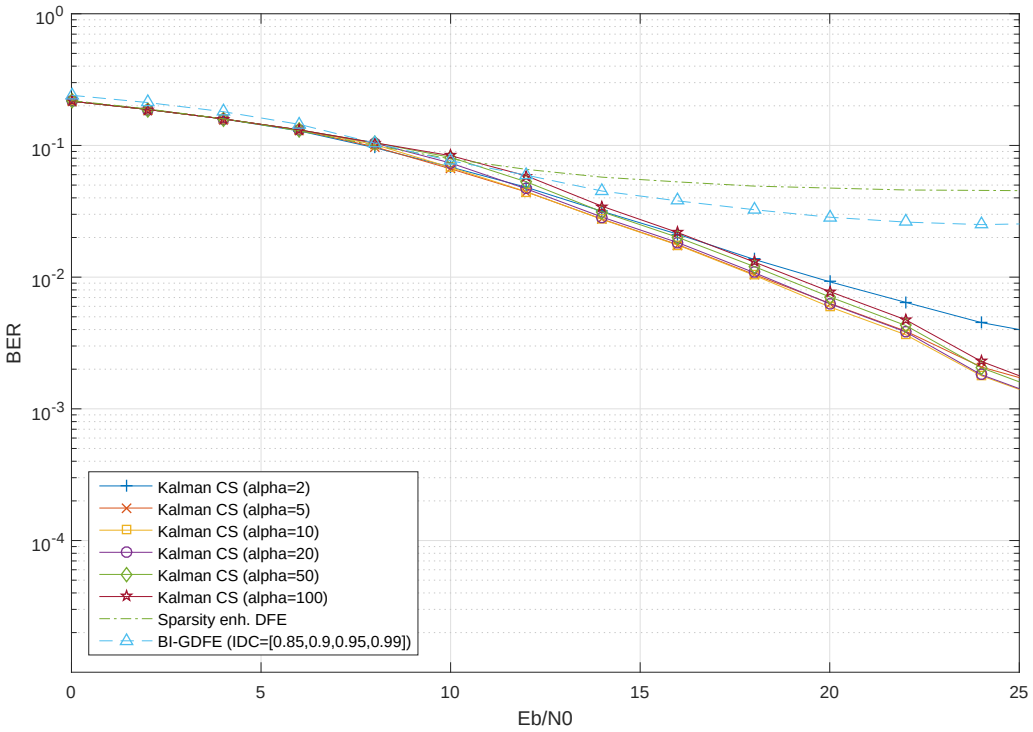


Figure 2.12: Varying α, $M = 64$, QAM-4 symbols, $L = 15$, $\delta = 0$, 900 random channels.

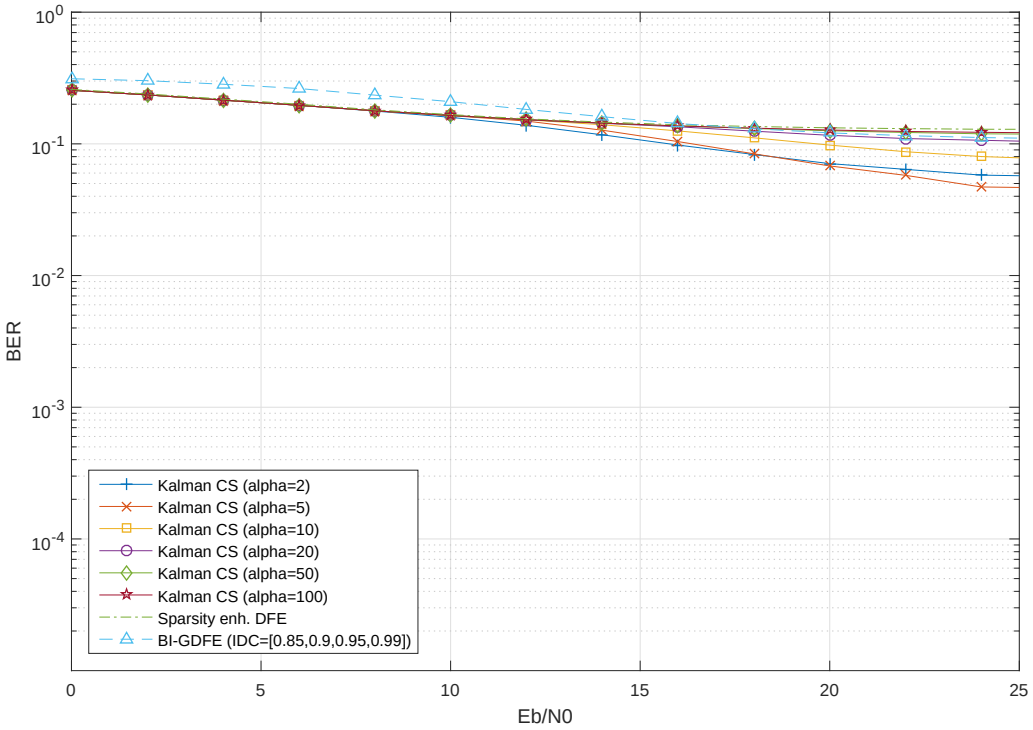


Figure 2.13: Varying α, $M = 64$, QAM-4 symbols, $L = 31$, $\delta = 0$, 900 random channels

In a minimum redundancy (MR) scenario, picking $\alpha \geq 50$ yields the best BER for $E_b/N_0 \geq 14$ dB, as shown in Figs. 2.14–2.16. Note that around $E_b/N_0 = 10$ dB the curves exhibit an inflexion, whose slope depends on the α, and which does not change significantly when $\alpha > 50$. The magnitude of the errors $|\tilde{x}(k)|$ under higher

noise figures is also responsible for false rejections in the likelihood test. Thus, we can reduce α at low SNR in order to reduce this effect. In Fig. 2.14, for a channel length $L = 15$, the curve for $\alpha = 20$ exhibits the lowest BER for $E_b/N_0 < 14$ dB. The same value of α applies for longer channels, say, $L = 31$ (Fig. 2.15), and $L = 43$ (Fig. 2.16).

We see that the thresholding strategy is very robust with choice of α, and allows us to implement simple heuristics that further improve symbol detection.

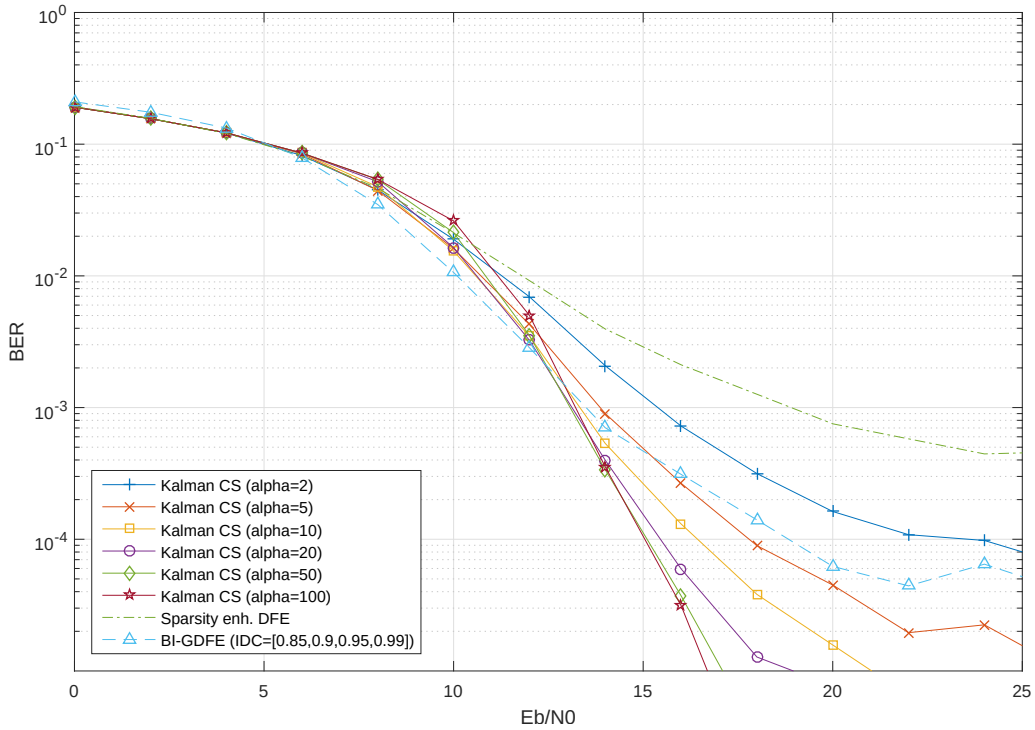


Figure 2.14: Varying α, $M = 64$, QAM-4 symbols, $L = 15$, $\delta = 7$ (MR), 900 random channels.

Experiment 2 - Performance Comparison among Transceivers

Figures 2.17-2.19 compare different transceiver algorithms and the Kalman CS-based iterations, for three different channel lengths in the case of minimum redundancy, ZP-ZJ setting. We contrast the new algorithm with: a single shot linear MMSE estimation, the V-BLAST, the sparsity enhanced DFE, and the BI-GDFE, the latter by varying the number of iterations from 1–4. The experiments are run for an ensemble of 500 channels with 40 block transmissions through each one, and considering QAM-4 constellations.

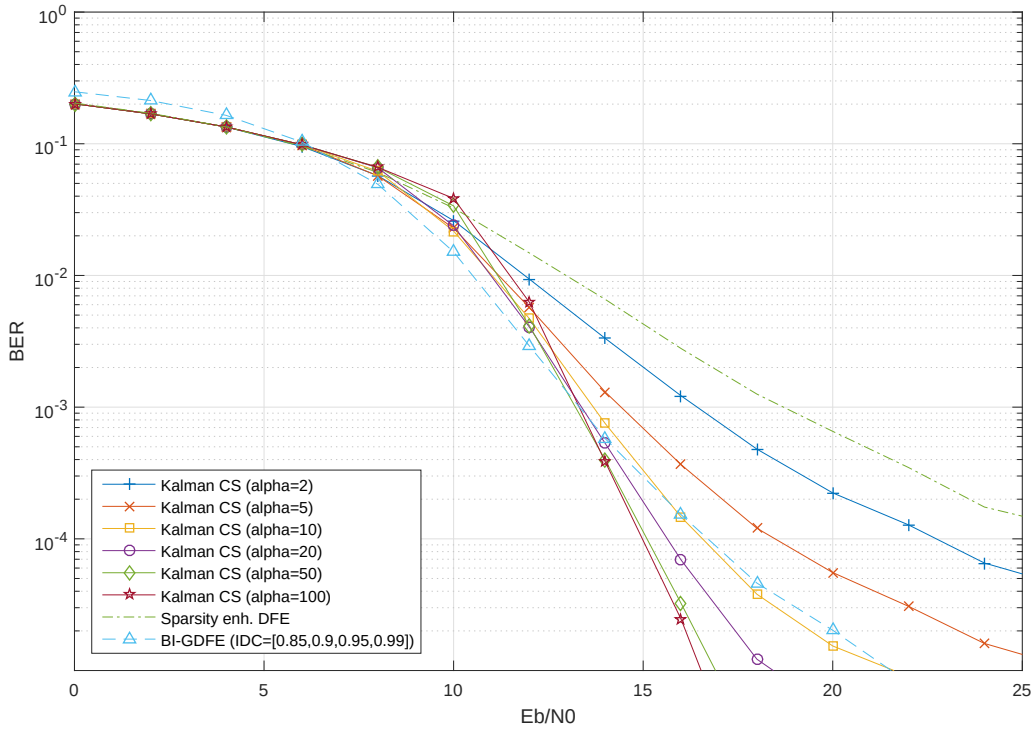


Figure 2.15: Varying α, $M = 64$, QAM-4 symbols, $L = 31$, $\delta = 15$ (MR), 900 random channels.

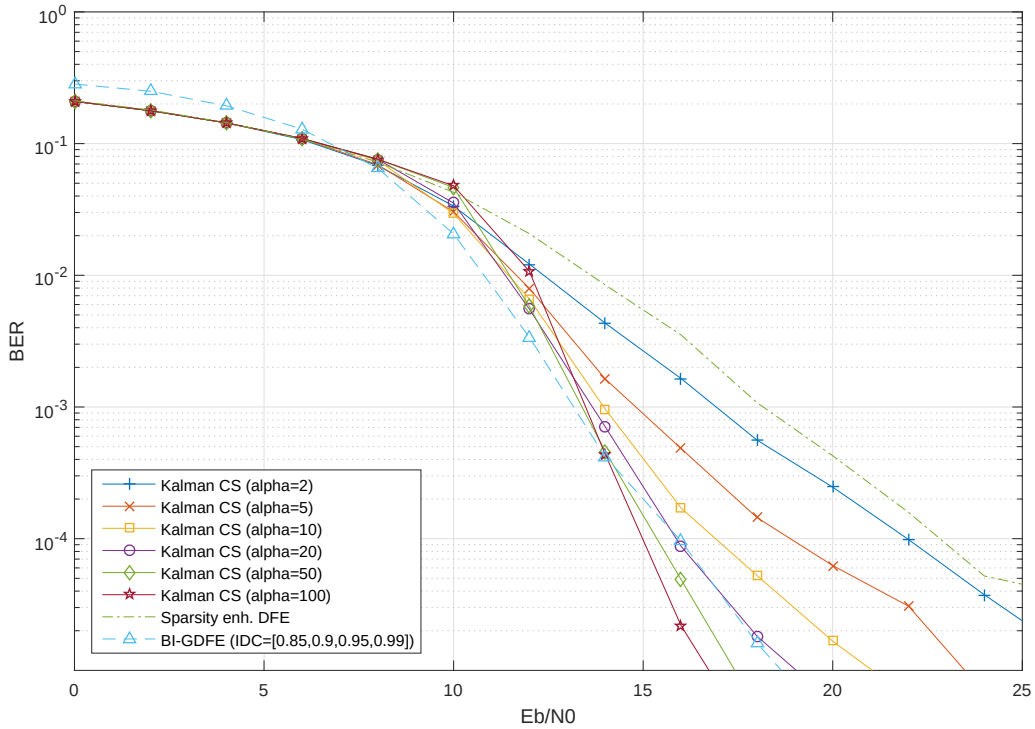


Figure 2.16: Varying α, $M = 64$, QAM-4 symbols, $L = 47$, $\delta = 23$ (MR), 900 random channels.

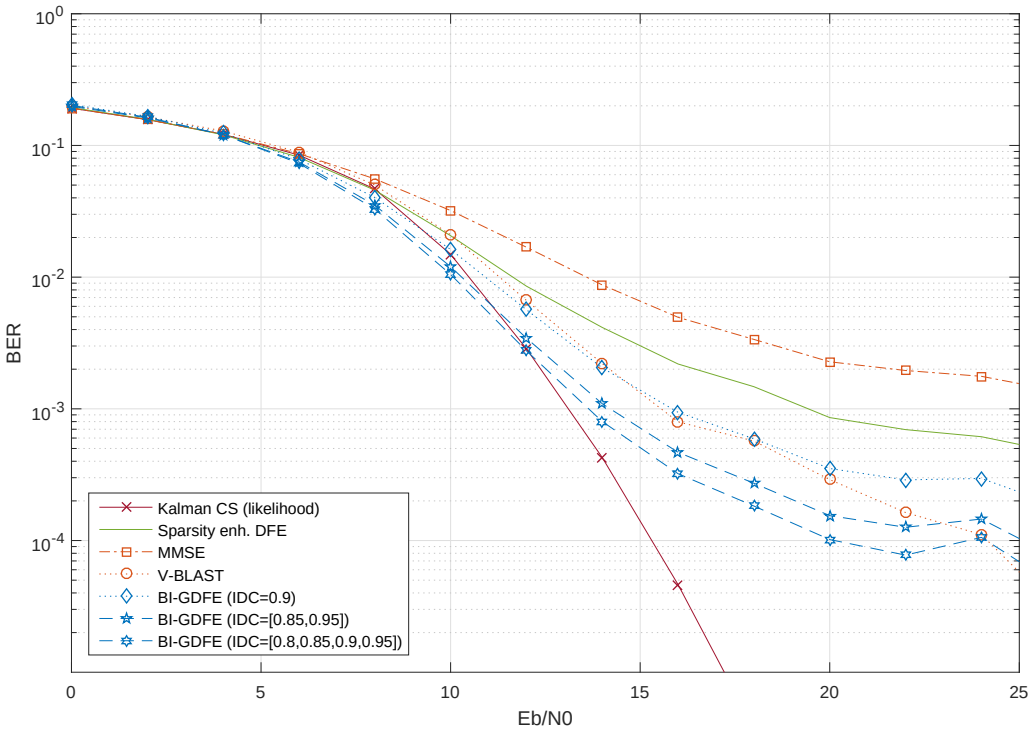


Figure 2.17: $M = 64$, QAM-4 symbols, $L = 15$, $\delta = 7$ (MR), 500 random channels.

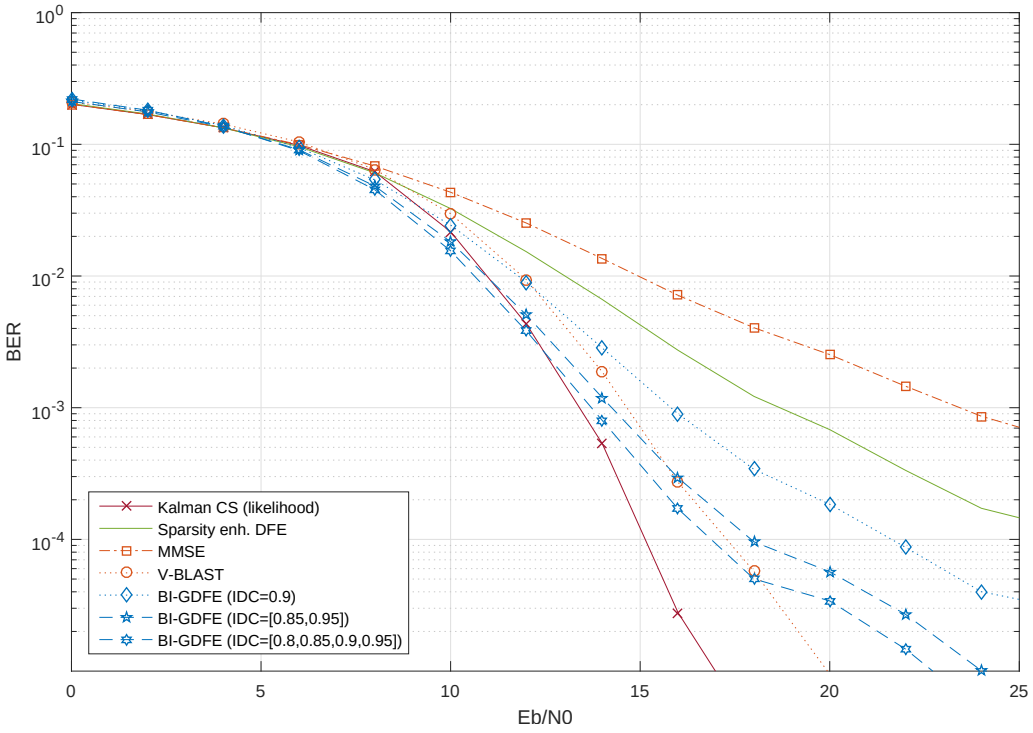


Figure 2.18: $M = 64$, QAM-4 symbols, $L = 31$, $\delta = 15$ (MR), 500 random channels.

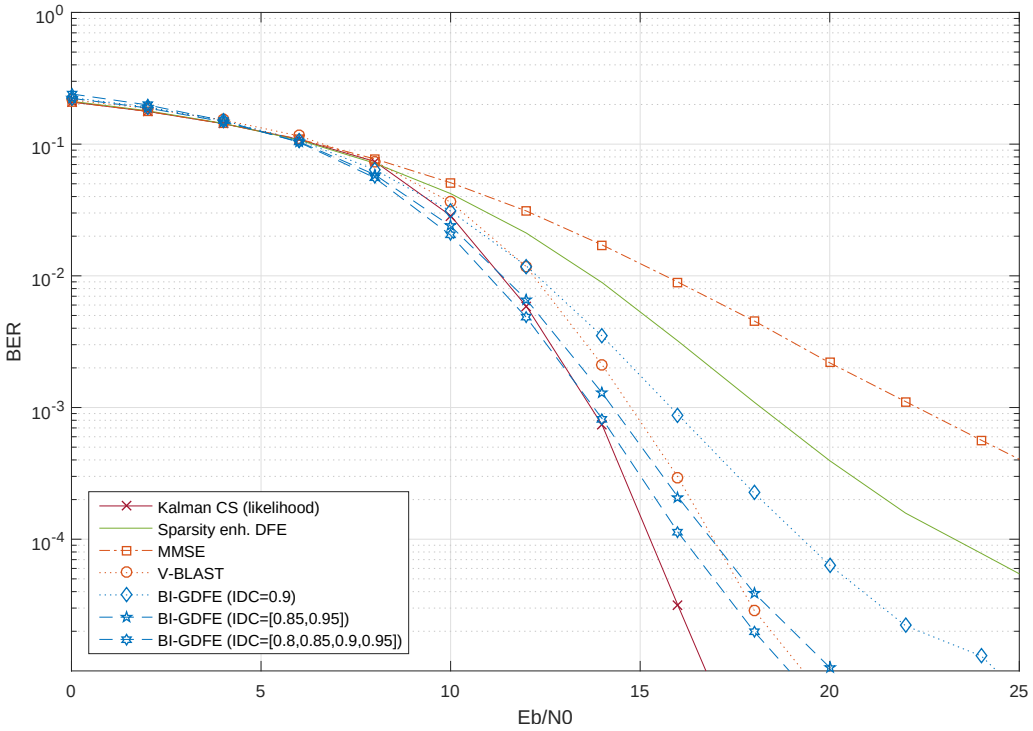


Figure 2.19: $M = 64$, QAM-4 symbols, $L = 47$, $\delta = 23$ (MR), 500 random channels.

Note that, at lower noise levels, the CS-based iterations show superior performance in most cases. At higher noise levels, the CS-based algorithm followed closely the BI-GDFE algorithm, but the IDC values in the latter have to be tuned offline, according to the procedure described in [11]. While this procedure is not very different from how we proceeded in the selection of α, the CS-based algorithm is less sensitive to its variation, and a single heuristic for α is deployed in all iterations. In these three experiments, we have selected $\alpha = 20$ for $E_b/N_0 < 14\,\text{dB}$ and $\alpha = 100$ otherwise.

Using smaller values of δ, which corresponds to transmitting more information samples, the CS-based iterations yield better performance than the competing ones, as seen in Figs. 2.20 ($L = 15$, $\delta = 3$) and 2.21 ($L = 31$, $\delta = 7$), although the performance for longer channels degrades quickly as result of ill-conditioning of the channel matrix. In Fig. 2.21 the change in α becomes evident from the jagged line appearance around 14 dB SNR. Although the heuristics of changing the value of α was applied to all experiments, this has been the only case which exhibited this artifact.

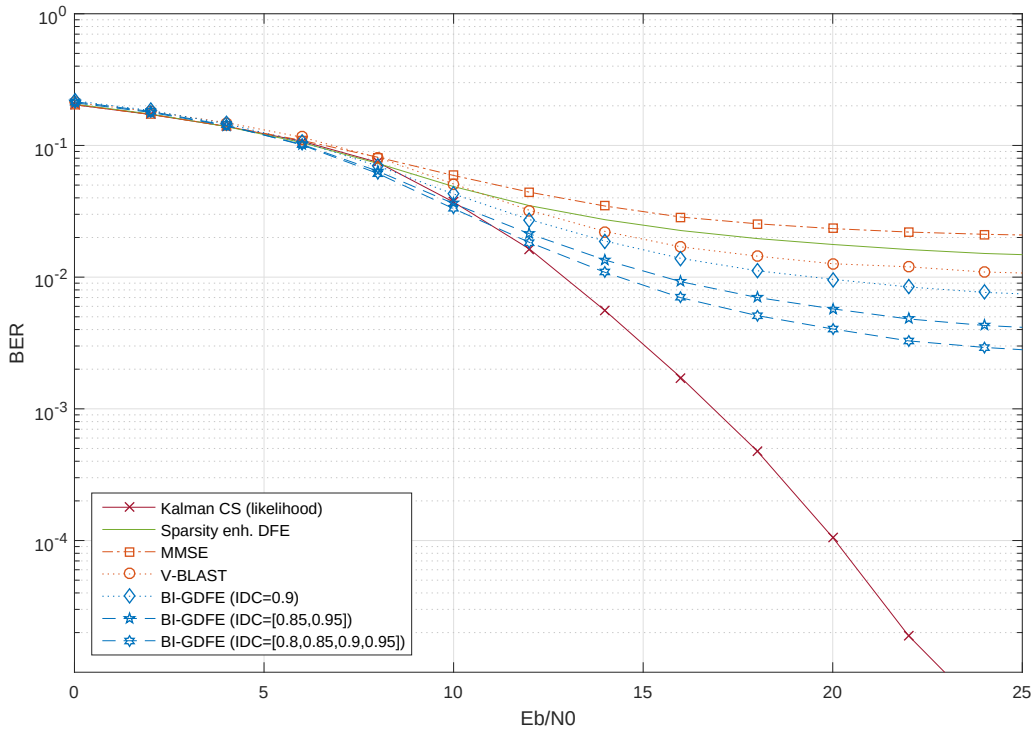


Figure 2.20: $M = 64$ QAM-4 symbols, $L = 15$, $\delta = 3$, 500 random channels.

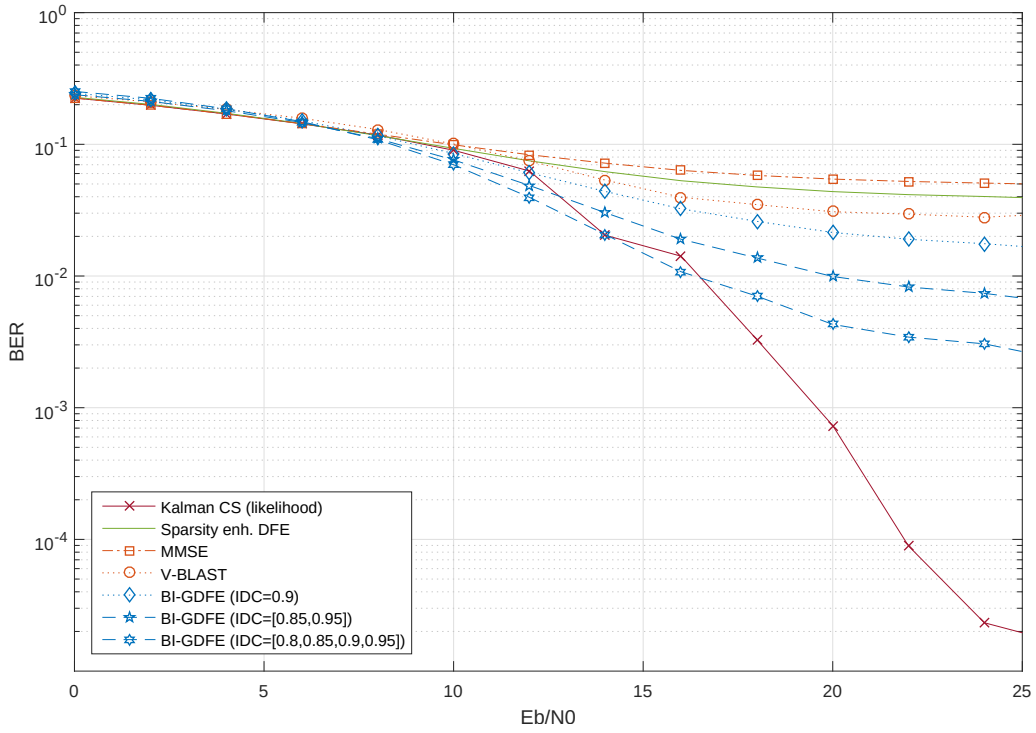


Figure 2.21: $M = 64$ QAM-4 symbols, $L = 31$, $\delta = 7$, 500 random channels.

Experiment 3 - Optimal decision delay

Figures 2.22-2.24 show the effect of the decision delay in the symbol detection. We consider the following channel impulse response

$$H(z) = (0.77 + 0.38j) + 0.57z^{-8} - 0.58z^{-9} - 0.567z^{-10} + 2.7z^{-13} + 0.4z^{-14}$$

through which 10000 QAM-4 blocks of symbols of size $M = 64$ were transmitted.

Figure 2.22 considers a minimum-redundancy scenario. Note that, apart from the CS-based algorithm, all other equalizers exhibited high BER performance. This is expected, since the optimal decision delay for this channel is $\delta_{\text{opt}} = 13$. With $\delta = 9$ (Fig. 2.23), the performance of the V-BLAST becomes comparable to the CS-based algorithm. Note that using the optimal decision delay for all schemes, as seen in Fig. 2.24, even a linear MMSE shows exceptional performance in recovering symbols. The good results observed in the case of the CS-based and the V-BLAST schemes are due to the optimality in selecting the entries that are fed back in the DFE loop. Since in the former case we are feeding back multiple symbols per iteration, the conditioning of the channel matrix improves faster, which accounts for the outcomes seen in the MR setting.

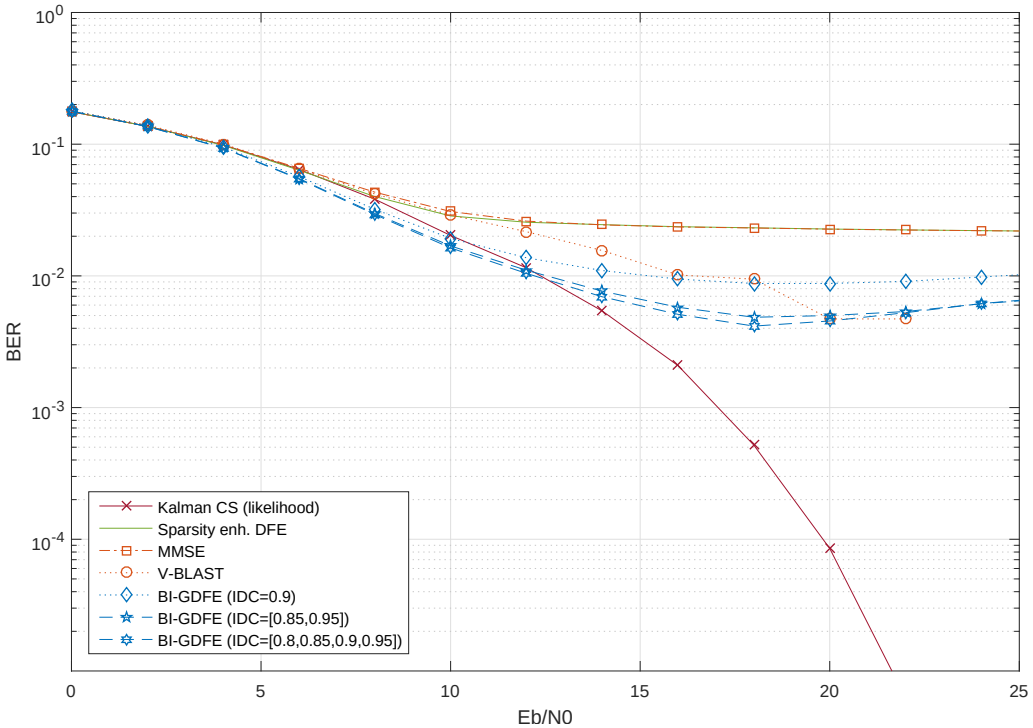


Figure 2.22: $M = 64$ QAM-4 symbols, $L = 15$, $\delta = 7$ (MR), single channel.

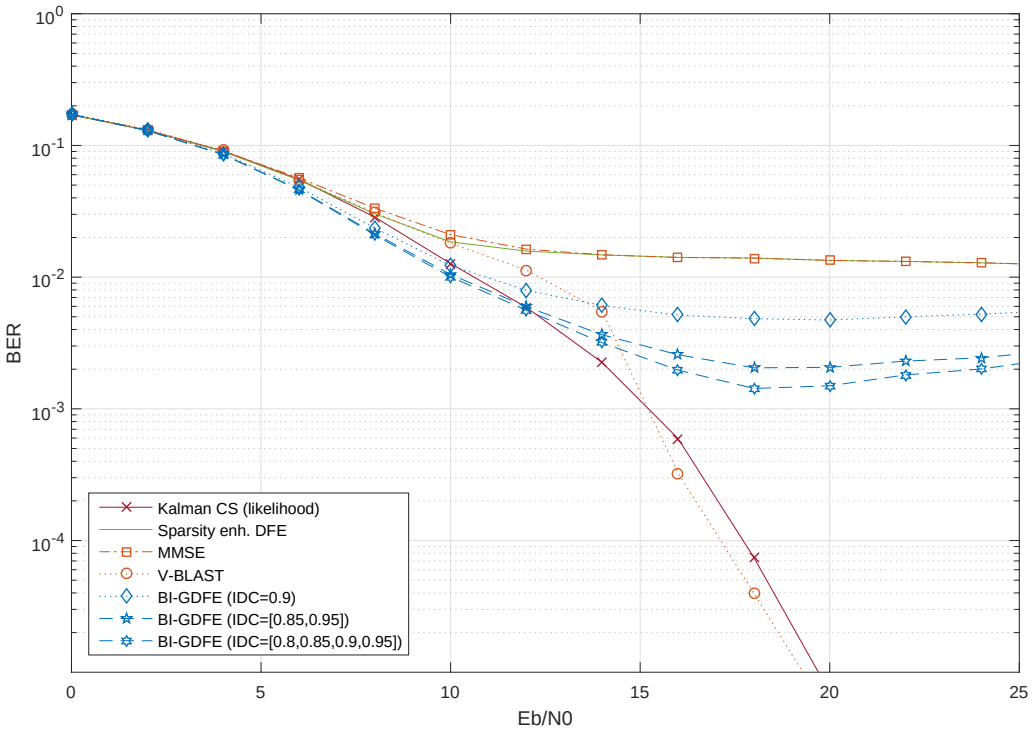


Figure 2.23: $M = 64$ QAM-4 symbols, $L = 15$, $\delta = 9$, single channel.

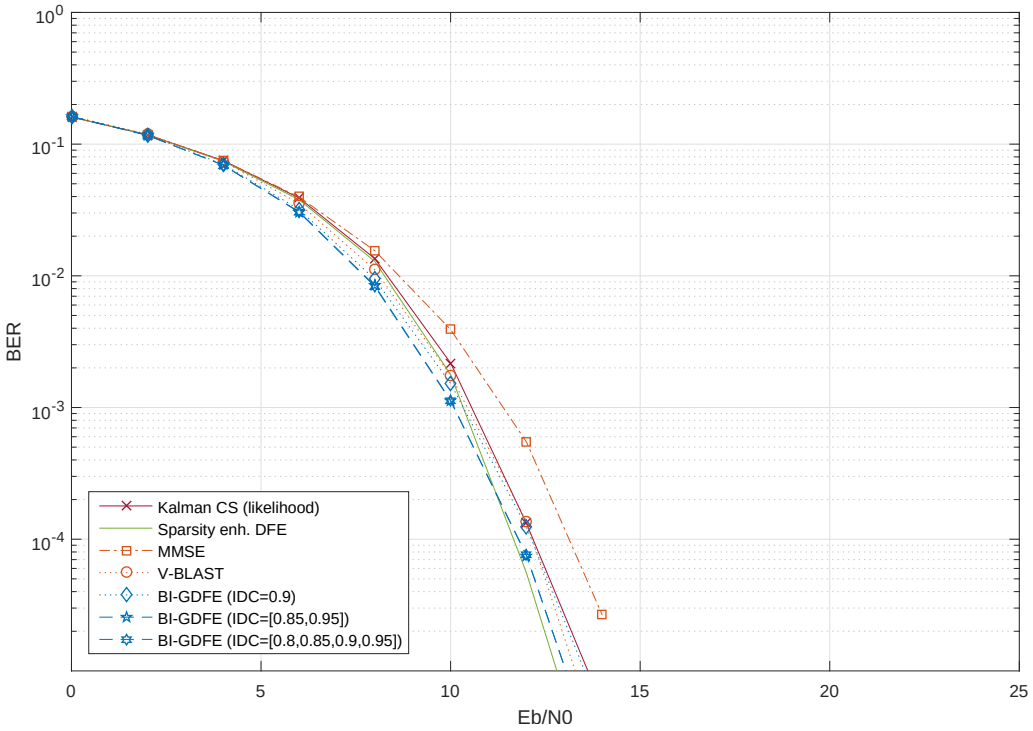


Figure 2.24: $M = 64$ QAM-4 symbols, $L = 15$, $\delta = 13$, single channel.

Experiment 4 - Performance without discarding samples

In the conventional ZP-ZJ approach considered in the previous experiments, the channel output samples are discarded prior to equalization. Following instead the model in (2.98), Figs. 2.25–2.27 illustrate data recovery for the case where only

zero-padding is applied, while no IBI removal is attempted.

Figure 2.25 shows the impressive case where from $M = 64$ received samples, the CS-based algorithm is capable to recover 75 symbols within two transmitted blocks at once, with fairly reasonable BER, by padding only 3 zeros to trailing data. That is, effectively, the received signal can be understood as being sampled at a lower rate than what is expected from the Nyquist spacing (see also Fig. 2.26). For a longer channel, shown in Fig. 2.27, the noise amplification still impairs the BER score, regardless of having a larger number of zeros padded in transmission. Nevertheless, we see that even in such harsh scenario, the CS-based recursions are still able to recover more symbols than what is received, except for an increased BER level.

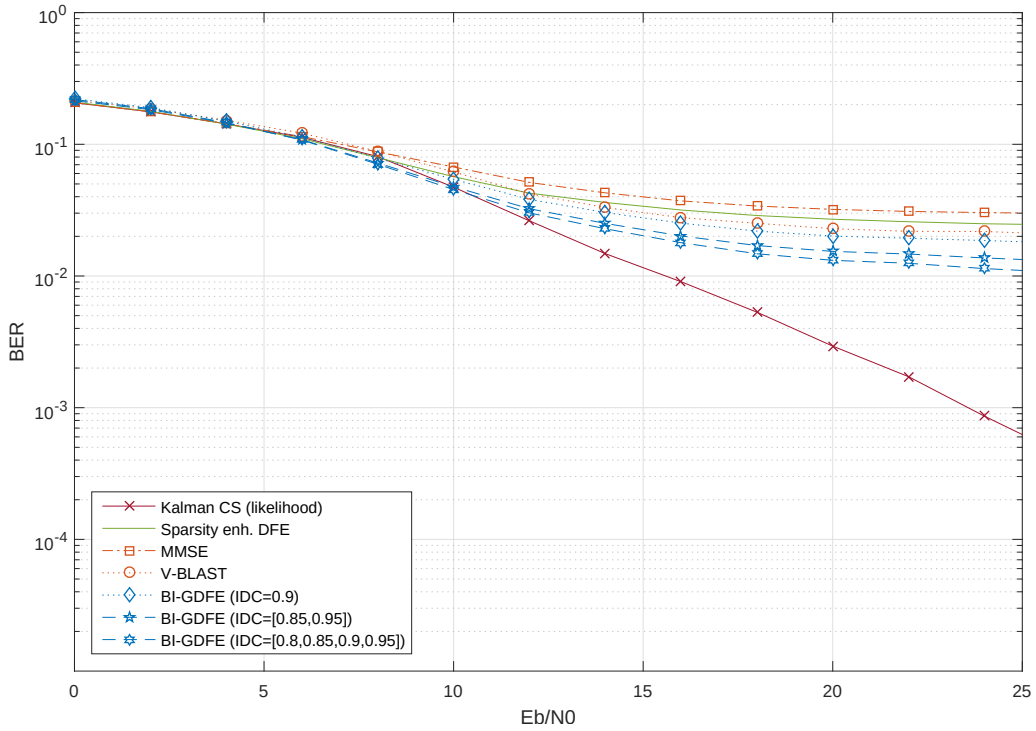


Figure 2.25: 75 QAM-4 symbols recovered from 64 received samples, with no discarded inputs and 3 padded zeros (500 random channels, $L = 15$, $\delta = 3$).

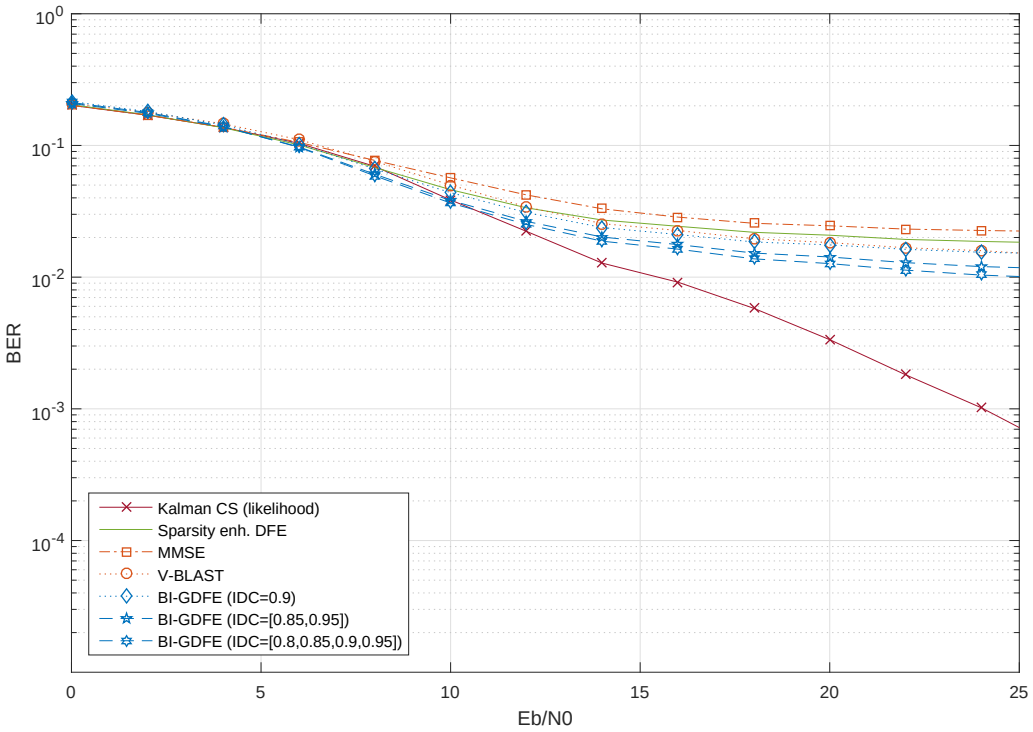


Figure 2.26: 71 symbols recovered from 64 received samples, with no discarded inputs and 7 padded zeros (500 random channels, $L = 15$, $\delta = 7$).

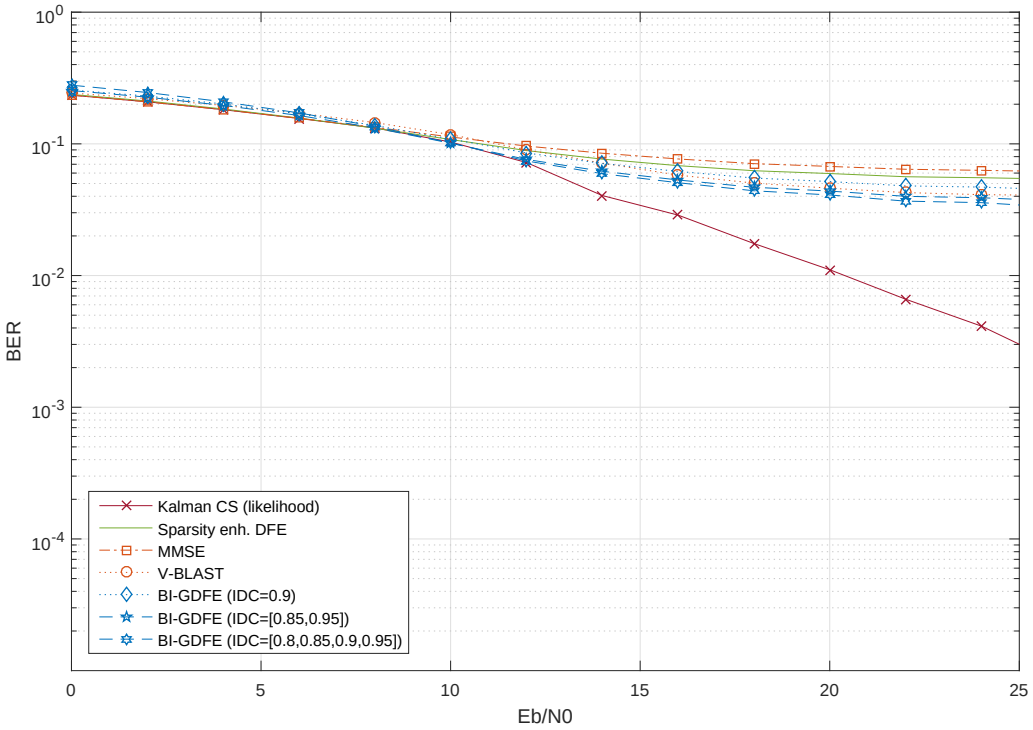


Figure 2.27: 87 QAM-4 symbols recovered from 64 received samples, with no discarded inputs and 23 padded zeros (500 random channels, $L = 47$, $\delta = 23$).

We consider now the following channel impulse response

$$H(z) = (0.77 + 0.38j) - 0.58z^{-3} + 0.28z^{-8} - 0.567z^{-10} + 0.7z^{-13} + 0.4z^{-15}$$

through which we recovered 73 symbols from $M = 64$ received samples. In this case, 5 zeros were padded to the trailing data, coinciding to the optimal decision delay of

$\delta_{\text{opt}} = 5$. Figure 2.28 shows that, for this specific channel, the sequential V-BLAST algorithm shows better performance than the CS-Kalman iteration, except for very high SNR values. That is, although on average the CS-Kalman algorithm exhibits the best performance, we can expect the sequential V-BLAST to be an alternative choice for specific channels. Here, from the knowledge of the optimal decision delay, the latter can make use of the FTF recursions to compute the DFE matrices, leading to a an efficient implementation compared to the CS-Kalman algorithm.

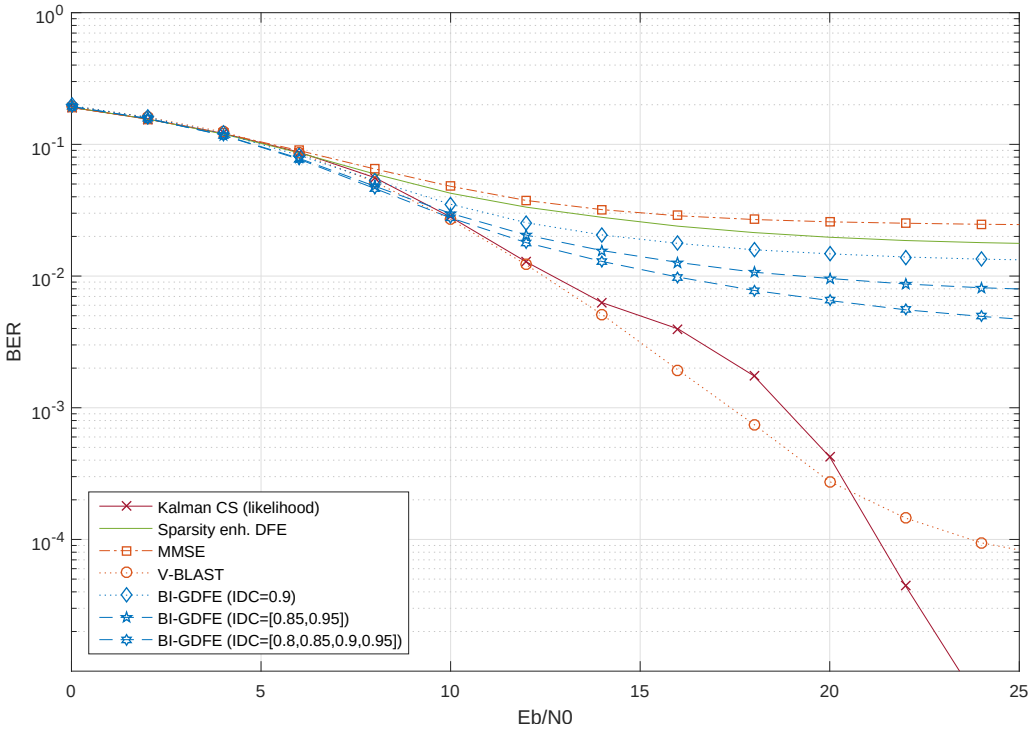


Figure 2.28: 73 symbols recovered from 64 received samples, with no discarded inputs and 5 padded zeros (single, $L = 15$, $\delta_{\text{opt}} = 5$).

Experiment 5 - Reduced-Complexity Widely-Linear model

Here, we compare a Reduced-Complexity, Widely-Linear version of the proposed algorithms, which considers the model representation of (2.79), to their Strictly-Linear counterparts, when recovering PAM-4 transmissions. In this experiment, we considered 500 random channels of length $L = 15$, over which 100 blocks of size $M = 64$ symbols were sent. Figure 2.29 shows that the WL model results in improved recovery of the symbols. We highlight that under a WL formulation, the V-BLAST equalizer achieved the same performance of the CS-Kalman algorithm, since the channel matrix in this case becomes taller, and thus, better conditioned.

The RC-WL approach also improves the BER when no IBI removal is considered in PAM-4 transmissions. Figure 2.30 illustrates the recovery of 78 symbols from 64 input samples, without zero-padding or zero-jamming, while Fig. 2.31 depicts the

case of 3 padded zeros. Note that, again, the V-BLAST algorithm performs very well.

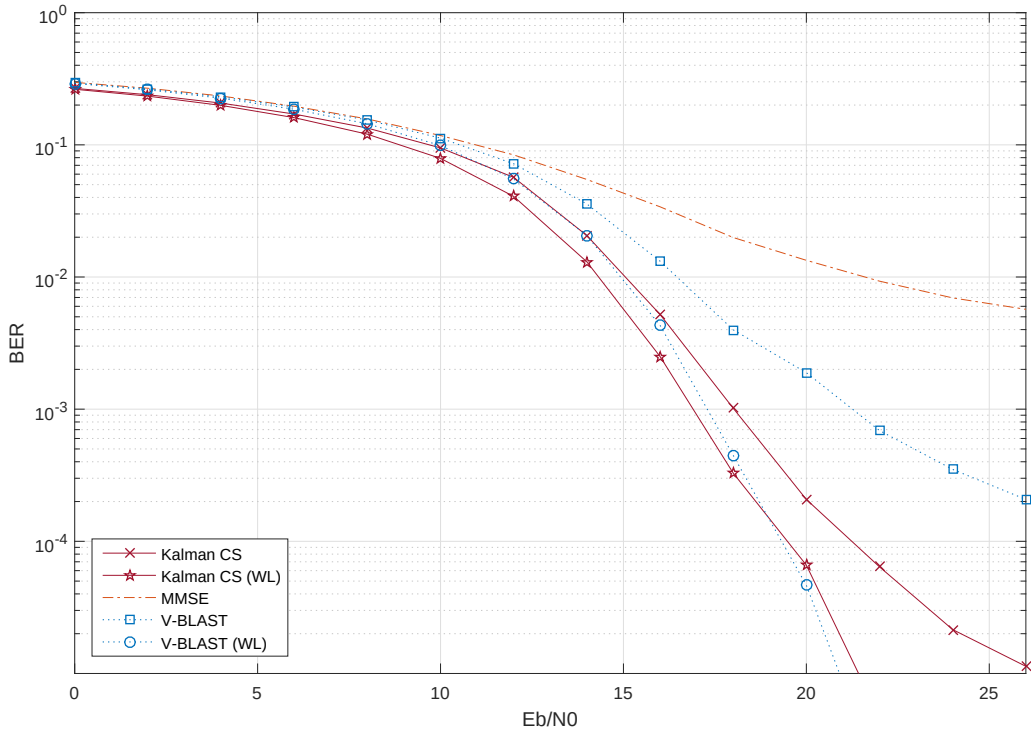


Figure 2.29: $M = 64$ PAM-4 symbols, $L = 15$, $\delta = 7$ (MR), 500 random channels.

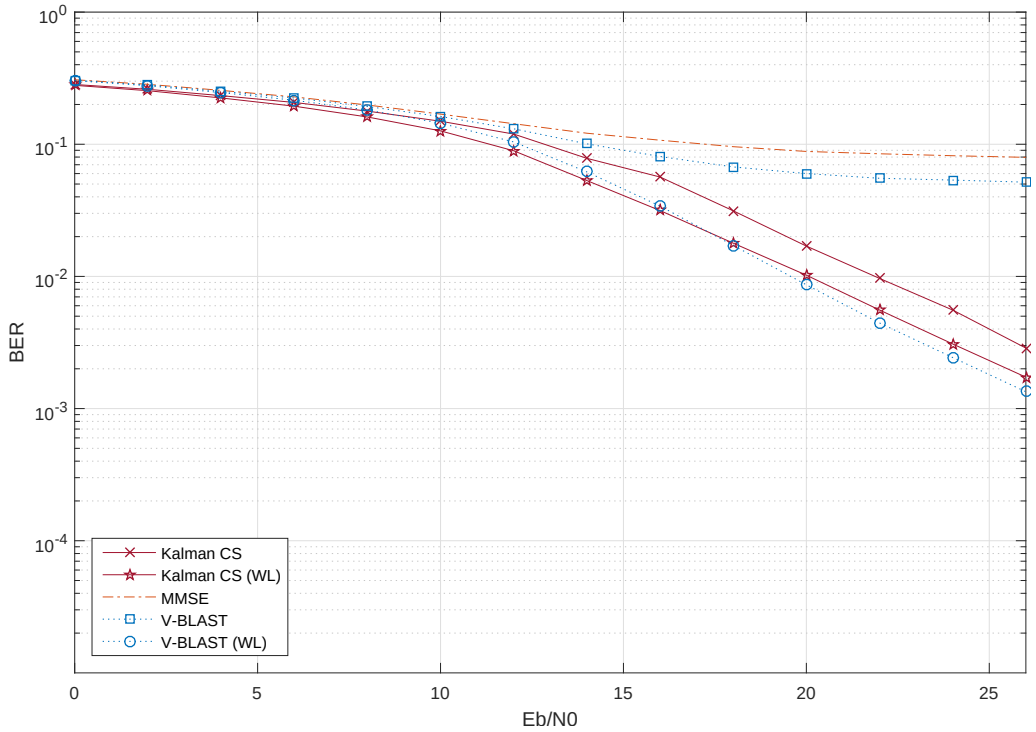


Figure 2.30: 78 PAM-4 symbols recovered from 64 received samples, with no discarded inputs and no padded zeros (500 random channels, $L = 15$, $\delta = 0$).

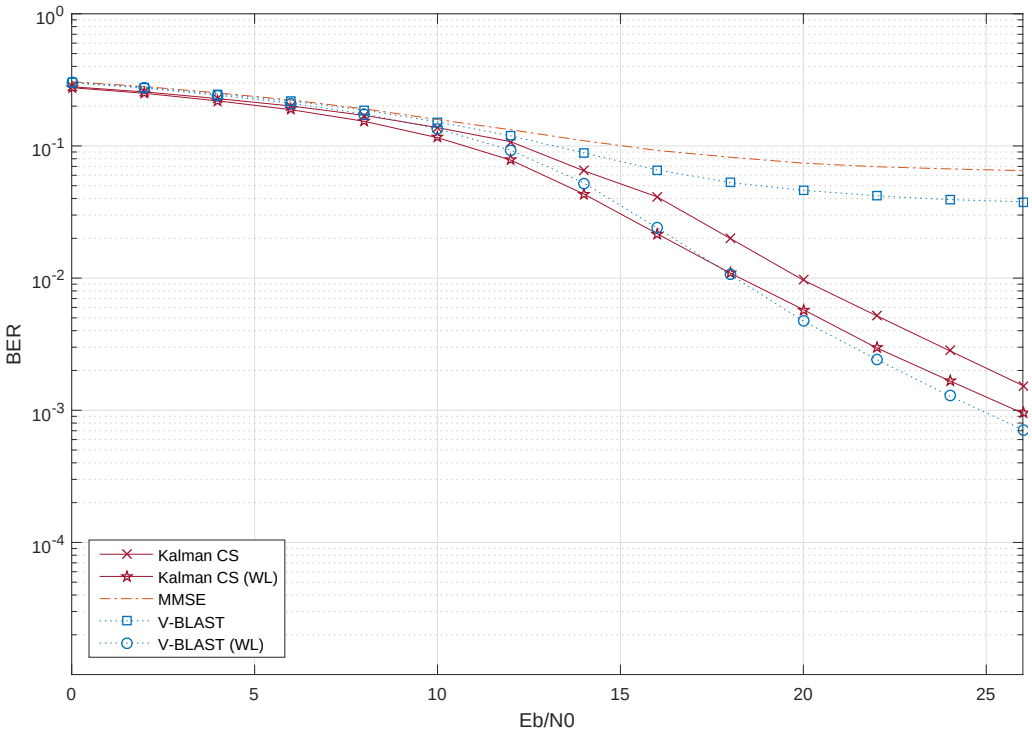


Figure 2.31: 75 PAM-4 symbols recovered from 64 received samples, with no discarded inputs and 3 padded zeros (500 random channels, $L = 15$, $\delta = 3$).

Experiment 6 - LTE channel and block IBI removal

We reinstate the scenario of experiment 3 in Sec. 2.9.1, however, this time including the CS-Kalman iterations in the plot. Just as before, Fig. 2.32 shows the case where the optimal redundancy level cannot be achieved with ZP-ZJ, either via linear or a DFE receiver. We clearly see that the CS-Kalman algorithm outperforms all remaining schemes. Actually, for very high SNR levels, above 25 dB, its performance becomes at least an order of magnitude superior compared to the iterative DFE that employs a single step of re-estimation proposed in [12]. Note that it is still worse than a ZF-OFDM approach, which is much simpler, despite employing double the redundancy.

In the simulations of Fig. 2.33 we have removed IBI via decision feedback as a first step. We clearly verify that the superiority of the CS-based iterations stands out. With the optimal redundancy, say, $\delta = 5$, and for $E_b/N_0 > 12$ dB, it achieves at least half of the BER obtained for the DFE employing a single re-estimation, except near $E_b/N_0 = 14$ dB, where the heuristics changes the value of α; with minimum redundancy, $\delta = 10$, the CS-Kalman exhibits the best performance, with BER of almost one order of magnitude smaller than the one of a DFE employing a single estimation, at SNR$>$ 16 dB.

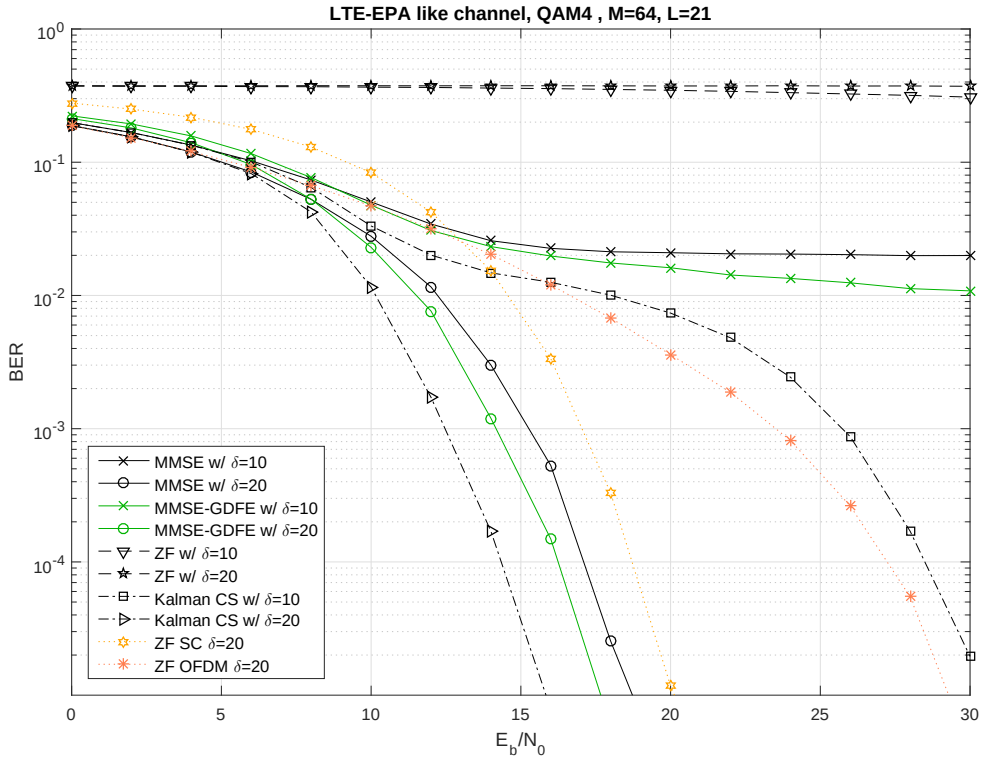


Figure 2.32: LTE EPA channel model — Comparison with the ZJ scheme.

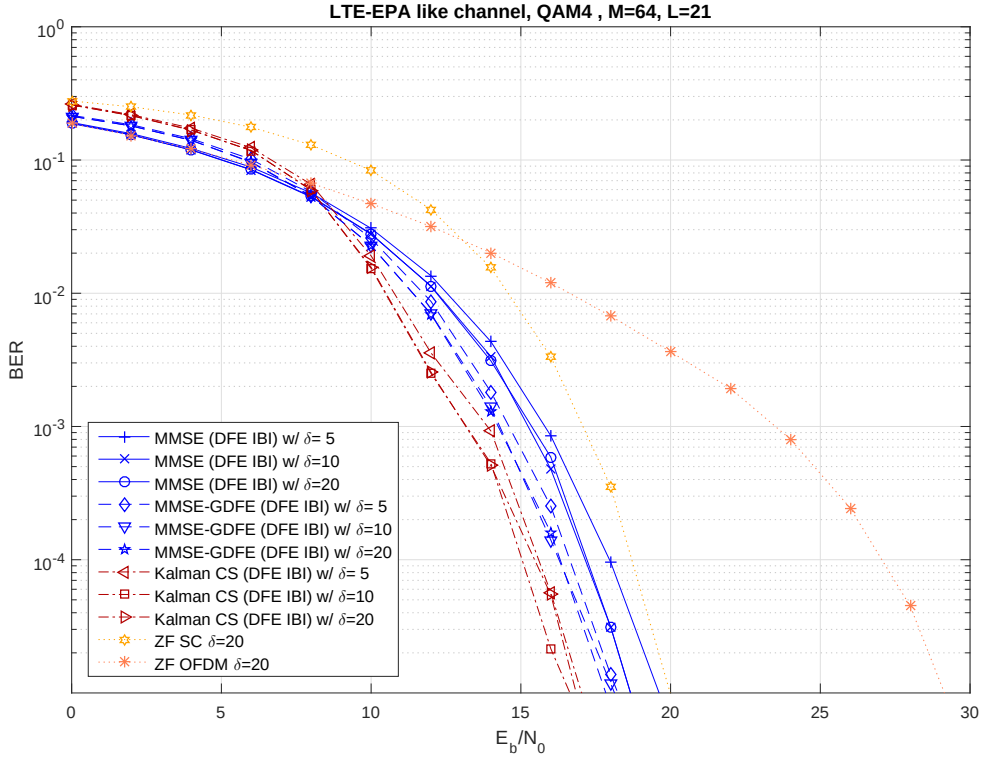


Figure 2.33: LTE-EPA: Block GDFE (DF-IBI) × ZJ-DFE × standard schemes.

Chapter 3

Stochastic Compressed Sensing Block-Iterative Decision Feedback Equalization (CS-BDFE)

Although message passing algorithms show faster convergence compared to the IST and IHT recursions [46], those are still first-order methods. That is, from an adaptive filtering standpoint, AMP/CAMP can be cast into the well known class of LMS-based adaptive algorithms, which are notorious for their slow convergence, when compared, e.g., to algorithms that follow a Newton-descend direction. Moreover, when dealing with large sensing matrices, specially ill-conditioned ones as found in a radar imaging setup (developed in the next chapter), the performance of these recursions can become rather poor, so that convergence and accuracy in detection become at stake. In this chapter, we pursue a novel approach toward a CS solution based on second-order methods, with the intent to accelerate convergence, as well as minimizing excess MSE. To this end, we shall readdress the CS formulation from a stochastic block equalization perspective.

We derive a CS-BDFE algorithm intended to more strict compressed sensing problems with regard to the structure of the sensing matrix. Unlike LMS type updates, which are most commonly used for CS, our approach relies on second-order methods, which, although more demanding, exhibits improved accuracy in terms of final estimate and target support.

The chapter is organized as follows. In Sec. 3.1 we motivate a second-order algorithm for CS, following similar steps of Sec. 2.1. In Sec. 3.2 we derive the new CS-BDFE algorithm from a stochastic DFE formulation, which differs from the CS-Kalman of the previous chapter, in the way uncertainty is assigned to the detected entries.

3.1 Kalman Filter based Compressed Sensing

In Sec. 2.1, the compressed sensing algorithm introduced for constellation detection was obtained by exploiting the inherent sparsity in the centralized estimates $\mathbf{x}-\mathbf{x}_{i-1}$, where $\mathbf{x}_{i-1}$ refers to the most recent estimate of $\mathbf{x}$. This can be approached in the ℓ_0- or ℓ_1-norm sense, assuming that $\mathbf{x}$ is sparse with respect to any vector that belongs to the same constellation. By enforcing regularization onto the vector $\mathbf{x}$ itself, we express our prior knowledge about its own sparsity, where ℓ_0- or ℓ_1-norm regularizers can be employed, just like in the iterative shriking algorithms of Sec. 1.1. While in the latter these common algorithms consist of LMS-based recursions, here, on the other hand, we shall consider a non-diagonal covariance method for CS, obtained from a stochastic formulation.

Thus, given a LS estimate $\hat{\mathbf{x}}_i$ and its uncertainty $\widehat{\boldsymbol{P}}_i$, by setting $\varrho(\mathbf{x}) = \|\mathbf{x}\|_{0,\boldsymbol{\Lambda}_i}$ in (2.124), the minimization problem reads

$$\min_{\mathbf{x}} \ \|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\widehat{\boldsymbol{P}}_i^{-1}} \ + \ \ \|\mathbf{x}\|_{0,\boldsymbol{\Lambda}_i} \tag{3.1}$$

in terms of the weighted ℓ_0-norm defined in (2.126). Again, resorting to the norm equivalence property, we can proceed by substituting

$$\|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\widehat{\boldsymbol{P}}_i^{-1}} \approx \|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\boldsymbol{C}_i}\,, \tag{3.2}$$

so that (3.1) is replaced by

$$\min_{\mathbf{x}} \quad \|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\boldsymbol{C}_i} \ + \ \ \|\mathbf{x}\|_{0,\boldsymbol{\Lambda}_i}. \tag{3.3}$$

The corresponding proximal mapping is computed exactly as

$$\mathbf{x}_i \ \ = \ \ q_i(\hat{\mathbf{x}}_i) \tag{3.4}$$

where $q_i(\cdot)$ is the entrywise hard threshold operator defined in (2.130), and reproduced here for convenience:

$$q_i(\theta_k) = \begin{cases} \theta_k, & |\theta_k| > \sqrt{\epsilon_{i,k}} \\ 0, & \text{otherwise,} \end{cases} \tag{3.5}$$

That is, given $\epsilon_{i,k}$, $\mathbf{x}_i$ is computed by projecting some of the entries of $\hat{\mathbf{x}}$ onto the origin. As we shall see, however, differently from the CS-Kalman algorithm of Sec. 2.1, adaptation of a detected entry that is considered correct is not terminated based on the threshold. Moreover, the underlying thresholds will be calculated adaptively, considering an approach similar to the one used in the previous chapter.

3.2 Block-Iterative DFE

Considering the DFE scheme of Fig. 2.2, it is noteworthy that while the Kalman recursions attempt to re-estimate a full vector per iteration, for an underdetermined, or even ill-conditioned problem, this may impair its ability to detect its correct support. A procedure that detects one symbol at a time, on the other hand, may offer a safer detection mechanism over a batch processing, despite requiring more computationally demanding implementations. The above scheme thus reveals a useful approach to CS, by predefining a detection strategy along with an explicit optimization of the matrices $\{\boldsymbol{G}_i, \boldsymbol{B}_i\}$ instead. Among several detection schemes, one stands out in the context of block equalization, where the nature of the vector parameter is not a sparse one, but belongs to a given constellation from a digital modulation.

In the so-called BI-GDFE, under a SL formulation, the authors in [11] formulate the problem in much the same way as in the conventional derivation, however, instead of relying on a slicer that makes perfect decisions through the traditional assumption $\mathsf{E}\mathbf{x}\mathbf{x}_i^* = \boldsymbol{R}_{\mathbf{x}\mathbf{x}_i} = \sigma_x^2\boldsymbol{I}$, the BI-GDFE relies on "soft" decisions in the sense that $\mathbf{x}_i$ admits uncertainty. This has been accomplished by optimizing the receiver for feedforward and feedback matrices $\{\boldsymbol{G}, \boldsymbol{B}\}$ with respect to the signal-to-interference-plus-noise ratio (SINR), where uncertainty in the decision defined through a correlation coefficient between the transmitted vector and its detected version, i.e., as $\boldsymbol{R}_{\mathbf{x}\mathbf{x}_i} = \rho_i\sigma_x^2\boldsymbol{I}$. That is, ρ_i is referred to as the input-decision-correlation (IDC) coefficient that reflects the reliability on the decisions taken at the i-th iteration of a re-estimation process. Although the optimallity in these recursions can be achieved without explicitly optimizing for the feedback matrix $\boldsymbol{B}$ (i.e., similarly to Sec. 2.2, with proper constraints), in order to gain further insight into the DFE recursions, we can resort to the stochastic dual of (2.13) , by rewriting the minimization problem explicitly in terms of DFE matrices $\{\boldsymbol{G}, \boldsymbol{B}\}$ as

$$\min_{\boldsymbol{W}} \ \mathsf{E}\left\| \mathbf{x} - \underbrace{[\boldsymbol{G} \ \ \boldsymbol{B}]}_{\boldsymbol{W}} \underbrace{\begin{bmatrix} \mathbf{y} \\ -\mathbf{x}_{i-1} \end{bmatrix}}_{\boldsymbol{u}} \right\|^2 \quad \text{s.t. structured zero-pattern } \boldsymbol{B} \tag{3.6}$$

The solution $\boldsymbol{W}_i = [\boldsymbol{G}_i \ \ \boldsymbol{B}_i]$ can be found similarly to the Wiener-Hopf technique [49], depending on the availability of a initial estimate, and/or how many entries of $\mathbf{x}$ we wish to detect per algorithm step. Recall that Wiener's solution is motivated by a causal, or, more specifically, lower triangular structure for $\boldsymbol{B}$. More general cases can be accounted for by restricting $\boldsymbol{B}$ to have a particular structure,

so that the solution to (3.6) must satisfy the normal equation

$$\boldsymbol{W}_i\boldsymbol{R}_{\boldsymbol{u}} - \boldsymbol{R}_{\boldsymbol{xu}} = [\,\boldsymbol{0} \quad \boldsymbol{A}_i\,]. \tag{3.7}$$

That is, the matrix $\boldsymbol{A}_i$ is the result of the a priori zero-pattern pre-imposed in $\boldsymbol{B}$, not captured by the normal equations, and which implicitly reflects alternative detection schemes. For example, if the entire estimate $\mathbf{x}_{i-1}$ is available, we can opt to feedback all the entries at once, so that $\boldsymbol{A}_i = \boldsymbol{0}$, and (3.7) colapses to the regular normal equations. The case of an ordered detection corresponds to the Wiener (causal) solution, where the entries of $\mathbf{x}_{i-1}$ are fed back as they become available. This is equivalent to assuming that $\boldsymbol{B}$ is lower triangular, so that $\boldsymbol{A}_i$ becomes an upper triangular matrix. The order of the detection can be changed by replacing $\mathbf{x}_{i-1}$ in (3.6) with $\boldsymbol{J}\mathbf{x}_{i-1}$, where $\boldsymbol{J}$ is a permutation matrix. One procedure that will prove successful in our context is to have each entry $x_i(k)$ estimated individually, after centralization of the model via the remaining entries of $\mathbf{x}_{i-1}$. This corresponds to selecting $\boldsymbol{B}_i$ as a full matrix with null diagonal, which implies that $\boldsymbol{A}_i$ is a diagonal matrix.

Using a SL model (2.3) or the WL description of (2.77), along with the definitions (2.33) and (2.34) (also extended accordingly for a WL derivation), we have

$$\begin{bmatrix}\boldsymbol{G}_i & \boldsymbol{B}_i\end{bmatrix}\begin{bmatrix}\boldsymbol{R}_{\boldsymbol{v}} + \sigma_x^2\boldsymbol{H}\boldsymbol{H}^* & -\boldsymbol{H}\boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} \\ -\boldsymbol{R}^*_{\mathbf{x}\mathbf{x}_{i-1}}\boldsymbol{H}^* & \boldsymbol{R}_{\mathbf{x}_{i-1}}\end{bmatrix} - [\,\sigma_x^2\boldsymbol{H}^* \quad -\boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}}\,] = [\,\boldsymbol{0} \;\; \boldsymbol{A}_i\,] \tag{3.8}$$

where either $\boldsymbol{R}_x = \sigma_x^2\boldsymbol{I}$ in the SL case, or $\boldsymbol{R}_{x_e} = (\sigma_{x_R}^2\boldsymbol{I} \oplus \sigma_{x_I}^2\boldsymbol{I})$ in the WL scenario (assuming uncorrelated real and imaginary parts). For simplicity, we shall continue with a SL notation, and highlight that the WL formulas can be easily obtained by properly associating the corresponding quantities. Hence, performing a standard block triangular factorization of $\boldsymbol{R}_{\boldsymbol{u}}$, that is,

$$\boldsymbol{R}_{\boldsymbol{u}} = \begin{bmatrix}\boldsymbol{I} & -\boldsymbol{H}\boldsymbol{K}_{i-1} \\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{R}_{\boldsymbol{v}} + \boldsymbol{H}\breve{\boldsymbol{P}}_{i-1}\boldsymbol{H}^* & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R}_{\mathbf{x}_{i-1}}\end{bmatrix}\begin{bmatrix}\boldsymbol{I} & \boldsymbol{0} \\ -\boldsymbol{K}^*_{i-1}\boldsymbol{H}^* & \boldsymbol{I}\end{bmatrix} \tag{3.9}$$

and the solution to (3.8) gives

$$\begin{cases}\boldsymbol{G}_i &= (\,\breve{\boldsymbol{P}}_{i-1} + \boldsymbol{A}_i\boldsymbol{K}^*_{i-1}\,)\boldsymbol{H}^*(\boldsymbol{R}_{\boldsymbol{v}} + \boldsymbol{H}\breve{\boldsymbol{P}}_{i-1}\boldsymbol{H}^*)^{-1} \\ \boldsymbol{B}_i &= \boldsymbol{G}_i\boldsymbol{H}\boldsymbol{K}_{i-1} - \boldsymbol{K}_{i-1} + \boldsymbol{A}_i\boldsymbol{R}^{-1}_{\mathbf{x}_{i-1}}\end{cases} \tag{3.10}$$

where we defined

$$\boldsymbol{K}_i \triangleq \boldsymbol{R}_{\mathbf{x}\mathbf{x}_i}\boldsymbol{R}^{-1}_{\mathbf{x}_i} \tag{3.11}$$

and the Schur complement in $\boldsymbol{R}_{\boldsymbol{u}}$ by

$$\breve{\boldsymbol{P}}_{i-1} \triangleq \sigma_x^2 \boldsymbol{I} - \boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} \boldsymbol{R}_{\mathbf{x}_{i-1}}^{-1} \boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}}^{*} \tag{3.12}$$

$$= \sigma_x^2 \boldsymbol{I} - \boldsymbol{K}_{i-1} \boldsymbol{R}_{\mathbf{x}_{i-1}} \boldsymbol{K}_{i-1}^{*} \tag{3.13}$$

We remark that $\breve{\boldsymbol{P}}_{i-1}$ is simply the minimum cost that results when estimating $\mathbf{x}$ from $\mathbf{x}_{i-1}$, via $\boldsymbol{K}_{i-1}$. This estimate is denoted

$$\breve{\mathbf{x}}_{i-1} = \boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} \boldsymbol{R}_{\mathbf{x}_{i-1}}^{-1} \mathbf{x}_{i-1} = \boldsymbol{K}_{i-1} \mathbf{x}_{i-1} \tag{3.14}$$

Let $\hat{\mathbf{x}}_i$ be the optimal solution to (3.6). Using (3.10), we obtain

$$\hat{\mathbf{x}}_i = \breve{\mathbf{x}}_{i-1} - \boldsymbol{A}_i \boldsymbol{R}_{\mathbf{x}_{i-1}}^{-1} \mathbf{x}_{i-1} + \boldsymbol{G}_i \mathbf{e}_i \tag{3.15}$$

$$= (\boldsymbol{I} - \boldsymbol{A}_i \boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}}^{-1}) \breve{\mathbf{x}}_{i-1} + \boldsymbol{G}_i \mathbf{e}_i \tag{3.16}$$

where we define $\mathbf{e}_i \triangleq (\mathbf{y} - \boldsymbol{H}\breve{\mathbf{x}}_{i-1})$.

To obtain $\boldsymbol{A}_i$, note that its entries can be determined by extracting the diagonal of $\boldsymbol{B}_i$ in (3.10) as

$$\mathsf{Diag}\left(\boldsymbol{G}_i \boldsymbol{H} \boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} \boldsymbol{R}_{\mathbf{x}_{i-1}}^{-1}\right) - \mathsf{Diag}\left(\boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} \boldsymbol{R}_{\mathbf{x}_{i-1}}^{-1}\right) + \boldsymbol{A}_i \mathsf{Diag}\left(\boldsymbol{R}_{\mathbf{x}_{i-1}}^{-1}\right) = \mathbf{0}. \tag{3.17}$$

Furthermore, if it is assumed that $\{\boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}}, \boldsymbol{R}_{\mathbf{x}_i}\}$ are diagonal matrices, we obtain

$$\boldsymbol{A}_i = \left[\boldsymbol{I} - \mathsf{Diag}(\boldsymbol{G}_i \boldsymbol{H})\right] \boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} \tag{3.18}$$

This is a reasonable assumption, considering that detection resolves the dependency among entries, and is consistent with the assumption in [11]. In this case, replacing (3.18) into (3.15), we obtain

$$\hat{\mathbf{x}}_i = \mathsf{Diag}(\boldsymbol{G}_i \boldsymbol{H}) \breve{\mathbf{x}}_{i-1} + \boldsymbol{G}_i \mathbf{e}_i \tag{3.19}$$

Figure 3.1 illustrates the corresponding block diagram.

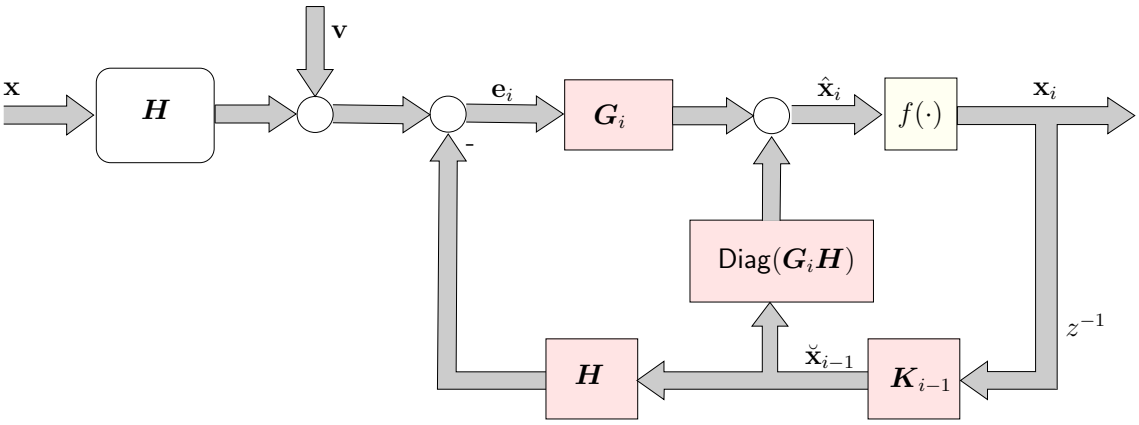


Figure 3.1: Resulting Block DFE structure.

Now, define

$$\boldsymbol{R}_{\mathbf{e},i} = \boldsymbol{R}_{\boldsymbol{v}} + \boldsymbol{H}\breve{\boldsymbol{P}}_{i-1}\boldsymbol{H}^*. \tag{3.20}$$

Replacing $\boldsymbol{G}_i$ from (3.10) into (3.18) yields

$$\begin{aligned} \boldsymbol{A}_i &= \boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} - (\,\breve{\boldsymbol{P}}_{i-1} + \boldsymbol{A}_i\boldsymbol{K}_{i-1}^*\,)\underbrace{\mathsf{Diag}\left(\boldsymbol{H}^*\boldsymbol{R}_{\mathbf{e},i}^{-1}\boldsymbol{H}\right)}_{\boldsymbol{D}_i}\boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} && (3.21)\\ &= \boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} - \breve{\boldsymbol{P}}_{i-1}\boldsymbol{D}_i\boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} - \boldsymbol{A}_i\boldsymbol{K}_{i-1}^*\boldsymbol{D}_i\boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}} && (3.22)\end{aligned}$$

where we denote $\boldsymbol{D}_i$ as indicated. Therefore, solving for $\boldsymbol{A}_i$, we get

$$\boldsymbol{A}_i = (\boldsymbol{I} - \breve{\boldsymbol{P}}_{i-1}\boldsymbol{D}_i)\boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}}(\boldsymbol{I} + \boldsymbol{K}_{i-1}^*\boldsymbol{D}_i\boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}})^{-1} \tag{3.23}$$

Note that introducing

$$\boldsymbol{\Gamma}_i = \breve{\boldsymbol{P}}_{i-1} + \boldsymbol{A}_i\boldsymbol{K}_{i-1}^* \tag{3.24}$$

we can write $\boldsymbol{G}_i = \boldsymbol{\Gamma}_i\boldsymbol{H}^*\boldsymbol{R}_{\mathbf{e},i}^{-1}$, so that $\mathsf{Diag}(\boldsymbol{G}_i\boldsymbol{H})$ can be expressed using only pre-computed diagonal matrices. That is

$$\mathsf{Diag}(\boldsymbol{G}_i\boldsymbol{H}) = \left(\breve{\boldsymbol{P}}_{i-1} + \boldsymbol{A}_i\boldsymbol{K}_{i-1}^*\right)\mathsf{Diag}(\boldsymbol{H}^*\boldsymbol{R}_{\mathbf{e},i}^{-1}\boldsymbol{H}) = \boldsymbol{\Gamma}_i\boldsymbol{D}_i. \tag{3.25}$$

Thus (3.19) simplifies to

$$\hat{\mathbf{x}}_i = \boldsymbol{\Gamma}_i\left(\boldsymbol{D}_i\breve{\mathbf{x}}_{i-1} + \boldsymbol{H}^*\boldsymbol{R}_{\mathbf{e},i}^{-1}\mathbf{e}_i\right) \tag{3.26}$$

We remark that the initial estimate $\mathbf{x}_0$ can be obtained as a pure MMSE estimator, where we consider that no entry has been annihilated by the threshold operation. Since we are also assuming diagonal covariances $\{\boldsymbol{R}_{\mathbf{x}\mathbf{x}_{i-1}}, \boldsymbol{R}_{\mathbf{x}_{i-1}}\}$, then

$\breve{\boldsymbol{P}}_0$ is defined by the diagonal entries of the MMSE estimation error. That is

$$\begin{aligned}
\widehat{\boldsymbol{P}}_0 &= \left(\sigma_x^{-2}\boldsymbol{I} + \boldsymbol{H}^*\boldsymbol{R}_{\boldsymbol{v}}^{-1}\boldsymbol{H}\right)^{-1} \\
\mathbf{x}_0 &= \widehat{\boldsymbol{P}}_0\boldsymbol{H}^*\boldsymbol{R}_{\boldsymbol{v}}^{-1}\boldsymbol{y} \\
\boldsymbol{K}_0 &= \boldsymbol{I} \\
\breve{\mathbf{x}}_0 &= \mathbf{x}_0 \\
\breve{\boldsymbol{P}}_0 &= \mathsf{Diag}\left(\widehat{\boldsymbol{P}}_0\right) \\
\boldsymbol{R}_{\mathbf{x}\mathbf{x}_0} &= \sigma_x^2\boldsymbol{I} - \breve{\boldsymbol{P}}_0,
\end{aligned}$$

which in turn can be calculated by superfast receivers as discussed in Sec. 2.7.1.

In order to compute $\{\boldsymbol{R}_{\mathbf{x}\mathbf{x}_i}, \boldsymbol{R}_{\mathbf{x}_i}\}$, however, we need to adopt a model for the threshold $q_i(\cdot)$.

3.2.1 Threshold Model

We adopt a first order random walk model for the threshold function $q_i(\cdot)$, where $\hat{\mathbf{x}}_i$ is disturbed by a zero-mean uncorrelated random vector $\mathbf{q}_i$, with variance $\boldsymbol{Q}_i$, i.e.,

$$\mathbf{x}_i = \hat{\mathbf{x}}_i + \mathbf{q}_i \ , \tag{3.27}$$

$$\boldsymbol{Q}_i \triangleq \mathsf{E}\mathbf{q}_i\mathbf{q}_i^*. \tag{3.28}$$

In this case, we obtain

$$\begin{aligned}
\boldsymbol{R}_{\mathbf{x}\mathbf{x}_i} &= \mathsf{E}\mathbf{x}\mathbf{x}_i^* && (3.29)\\
&= \mathsf{E}(\hat{\mathbf{x}}_i + \tilde{\mathbf{x}}_i)(\hat{\mathbf{x}}_i + \mathbf{q}_i)^* \\
&= \boldsymbol{R}_{\hat{\mathbf{x}}_i} = \sigma_x^2\boldsymbol{I} - \widehat{\boldsymbol{P}}_i && (3.30)
\end{aligned}$$

where we defined the estimation error $\tilde{\mathbf{x}}_i = \mathbf{x} - \hat{\mathbf{x}}_i$, with $\widehat{\boldsymbol{P}}_i = \mathsf{E}\tilde{\mathbf{x}}_i\tilde{\mathbf{x}}_i^*$, and assumed that $\tilde{\mathbf{x}}_i$ and $\mathbf{q}_i$ are uncorrelated. As a consequence, we have that

$$\widehat{\boldsymbol{P}}_i = \sigma_x^2\boldsymbol{I} - \boldsymbol{\Gamma}_i\boldsymbol{D}_i\left(\sigma_x^2\boldsymbol{I} - \breve{\boldsymbol{P}}_{i-1}\right)\boldsymbol{D}_i^*\boldsymbol{\Gamma}_i^* - \boldsymbol{G}_i\boldsymbol{R}_{\mathbf{e},i}\boldsymbol{G}_i^*. \tag{3.31}$$

While the covariance matrix for $\mathsf{E}\hat{\mathbf{x}}_i\hat{\mathbf{x}}_i^*$ is no longer diagonal, in practice we can use the approximation

$$\boldsymbol{R}_{\hat{\mathbf{x}}_i} \approx \sigma_x^2\boldsymbol{I} - \mathsf{Diag}(\widehat{\boldsymbol{P}}_i). \tag{3.32}$$

These diagonal entries correspond to a marginalization with respect to the covariance of $\hat{\mathbf{x}}_i$, as if the entries of $\mathbf{x}_i$ were independently drawn. Note that by taking the

diagonal of (3.31), the last term of this recursion simplifies to

$$\begin{aligned} \mathsf{Diag}\left(\boldsymbol{G}_i \boldsymbol{R}_{\mathbf{e},i} \boldsymbol{G}_i^*\right) &= \boldsymbol{\Gamma}_i \mathsf{Diag}\left(\boldsymbol{H}^* \boldsymbol{R}_{\mathbf{e},i}^{-1} \boldsymbol{H}\right) \boldsymbol{\Gamma}_i^* && (3.33) \\ &= \boldsymbol{\Gamma}_i \boldsymbol{D}_i \boldsymbol{\Gamma}_i^* && (3.34) \end{aligned}$$

Hence, we obtain

$$\mathsf{Diag}(\widehat{\boldsymbol{P}}_i) \triangleq \boldsymbol{\Upsilon}_i = \sigma_x^2 \boldsymbol{I} - \boldsymbol{\Gamma}_i \boldsymbol{D}_i \left(\sigma_x^2 \boldsymbol{I} - \breve{\boldsymbol{P}}_{i-1}\right) \boldsymbol{D}_i^* \boldsymbol{\Gamma}_i^* - \boldsymbol{\Gamma}_i \boldsymbol{D}_i \boldsymbol{\Gamma}_i^* \tag{3.35}$$

Next, for $\boldsymbol{R}_{\mathbf{x}_i}$, we have

$$\begin{aligned} \boldsymbol{R}_{\mathbf{x}_i} &= (\hat{\mathbf{x}}_i + \mathbf{q}_i)(\hat{\mathbf{x}}_i + \mathbf{q}_i)^* && (3.36) \\ &= \boldsymbol{R}_{\hat{\mathbf{x}}_i} + \boldsymbol{Q}_i. && (3.37) \end{aligned}$$

so that by virtue of (3.32), we can also take it as a diagonal matrix. At this point, we are left to specifying the covariance $\boldsymbol{Q}_i$.

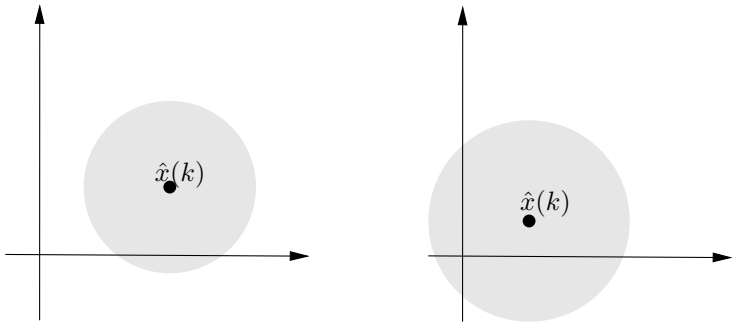
Recall from (3.4), and according to (3.27), that by setting $x_i(k) = \hat{x}(k)$ we are equivalently saying that the k-th entry of the LS estimate has not been disturbed by the threshold. This corresponds to selecting $\mathrm{q}_i(k) = 0$, and since $\mathbf{q}_i$ is zero mean, we can set $[\boldsymbol{Q}_i]_{kk} = 0$. When otherwise, $x_i(k) = 0$, we assume that the corresponding entry $\mathrm{q}_i(k)$ is drawn from a two dimensional uniform distribution in the complex plane, enclosed by a disc with radius $|\hat{x}_i(k)|$. This implies setting its variance to $[\boldsymbol{Q}_i]_{kk} = |\hat{x}_i(k)|^2/8$. Such uniform distribution yields maximum entropy in a closed interval, and allows for any value inside the disc to be picked with equal probability. Thus, we can accomodate both cases by equivalently writing

$$\boldsymbol{Q}_i = \frac{1}{8} \mathsf{diag}\left(|\hat{\mathbf{x}}_i - \mathbf{x}_i|^2\right). \tag{3.38}$$

Now, in block equalization, the estimate $\mathbf{x}_i$ is obtained from a projection of $\hat{\mathbf{x}}_i$ onto the closest point of a signal constellation, via a function $f(\cdot)$ that acts as a slicer in the ℓ_2 sense. In the CS context, and following (3.3)–(3.5) on the other hand, we are interested in projecting the entries of $\hat{\mathbf{x}}_i$ onto the origin, while leaving the remaining ones unaffected. Given the deterministic-stochastic duality of regularized LS and MMSE problems, it turns out that for any problem of the form (3.1), once we have properly propagated the uncertainty $\widehat{\boldsymbol{P}}_i$ via its respective defining recursions, we shall have a procedure to solve (3.3) in the context of a weighted ℓ_0 norm regularization.

The adaptive choice for the thresholds $\epsilon_{i,k}$ in (3.5) will follow a similar path used on the previous chapter for the CS-based estimation of constellation signals. Again, each entry $\tilde{x}_i(k)$ has variance denoted by $\sigma_{i,k}^2$, which is extracted from the

corresponding diagonal entry of $\widehat{\boldsymbol{P}}_i$. Due to the Gaussian circularity assumption, the true value of $x(k) = \hat{x}(k) + \tilde{x}(k)$ will be at any point inside a disc centered in $\hat{x}_i(k)$ with radius $|\tilde{x}_i(k)|$. Now, note that this disc will not contain the origin if $|\tilde{x}_i(k)| < |\hat{x}_i(k)|$, as shown in Fig. 3.2a. In this case there is no possibility of having $x(k) = 0$. On the other hand, if $|\tilde{x}_i(k)| \geq |\hat{x}_i(k)|$, as in Fig. 3.2b, then we should be able to set $x(k) = 0$. Our goal in the remaining of this section, is thus to provide an robust mechanism to obtain the thresholds for (2.130), based on the pre-computed statistics of the estimation error $\tilde{\mathbf{x}}_i$, under a certain hypothesis test.



(a) $|\tilde{x}(k)| < |\hat{x}(k)|$ (b) $|\tilde{x}(k)| > |\hat{x}(k)|$

Figure 3.2: Two discs in the complex plane, with different values of $\hat{x}$ and $\tilde{x}$. The shaded area denotes the possible values for $x(k) = \hat{x}(k) + \tilde{x}(k)$.

Based on this observation, we establish our hypothesis test on the likelihoods. That is, we consider that $x(k) \neq 0$ if

$$\frac{\Pr\big(x(k) \neq 0\big)}{\Pr\big(x(k) = 0\big)} > \alpha \tag{3.39}$$

for a predefined level $\alpha \gg 1$. Since $|x(k)|$ cannot be zero when $|\tilde{x}_i(k)| < |\hat{x}_i(k)|$ (but not the converse), we have that

$$\Pr\big(x(k) \neq 0\big) > \Pr\big(|\tilde{x}_i(k)| < |\hat{x}_i(k)|\big). \tag{3.40}$$

On the other hand, if $|\tilde{x}_i(k)| \geq |\hat{x}_i(k)|$, $x(k)$ can assume a zero value, so that

$$\Pr\big(x(k) = 0\big) < \Pr\big(|\tilde{x}_i(k)| \geq |\hat{x}_i(k)|\big). \tag{3.41}$$

Then, in order for (3.39) to hold, it suffices that

$$\frac{\Pr\big(|\tilde{x}_i(k)| < |\hat{x}_i(k)|\big)}{\Pr\big(|\tilde{x}_i(k)| \geq |\hat{x}_i(k)|\big)} > \alpha, \tag{3.42}$$

since

$$\frac{\Pr\left(x(k)\neq 0\right)}{\Pr\left(x(k)=0\right)} > \frac{\Pr\left(|\tilde{x}_i(k)|<|\hat{x}_i(k)|\right)}{\Pr\left(|\tilde{x}_i(k)|\geq|\hat{x}_i(k)|\right)}. \tag{3.43}$$

Using the fact that the magnitude $|\tilde{x}_i(k)|$ follows a Rayleigh distribution, the likelihood test (3.42) can be expressed analytically as

$$\frac{1-e^{-\frac{|\hat{x}(k)|^2}{2\sigma_{i,k}^2}}}{e^{-\frac{|\hat{x}(k)|^2}{2\sigma_{i,k}^2}}} > \alpha\ , \tag{3.44}$$

so that after some simple algebraic manipulation, we conclude that we should set $x_i(k)=\hat{x}_i(k)$ whenever

$$|\hat{x}_i(k)| > \sigma_{i,k}\sqrt{2\ln\left(1+\alpha\right)}\ , \tag{3.45}$$

otherwise, $x_i(k)=0$ [but with an increase in uncertainty corresponding to this entry for the next iteration, by virtue of (3.27)]. We remark that α is a user defined scalar for the relevance of accepting one entry in $\mathbf{x}_i$ as correct. The same scalar must be used for all entries, as we need a common base of comparison in $\mathbf{x}_i$.

Again it is not possible to derive α directly from the significance level since the probability distribution of $\hat{x}_i(k)$ is not known a priori, but again we verified that the above criterion is robust with respect to the choice of α (this allowed us to choose it experimentally with good results). The scalar α plays a role similar to κ in (1.33), for the CAMP algorithm, which has to be experimentally chosen as well. Finally, the complete algorithm is summarized in Table 3.1.

3.2.2 Numerical evaluation

We assess the performance of the new CS-BDFE algorithm via a commonly used phase transition diagram [68], computed for a large number of sensing matrices, spanning a range of underdeterminacy and sparsity levels. The latter is normalized by the number of system variables. That is, given the channel model $\boldsymbol{y}=\boldsymbol{H}\boldsymbol{x}+\boldsymbol{v}$, where $\boldsymbol{H}$ is $M\times N$, and $\boldsymbol{x}$ is a k-sparse vector, the horizontal axis in a phase diagram represents the ratio M/N, while in the vertical axis we read the ratio k/M. Each point in the resulting plane represents an average value of a performance measure, here adopted as the *Average Fractional Error*, i.e.,

$$\text{AFE} \triangleq \mathsf{E}\,\frac{\|\boldsymbol{x}-\boldsymbol{x}_{i_{\max}}\|}{\|\boldsymbol{x}\|},$$

Initialization:

$$\widehat{\boldsymbol{P}}_0 = \left(\sigma_x^{-2}\boldsymbol{I} + \boldsymbol{H}^*\boldsymbol{R}_{\boldsymbol{v}}^{-1}\boldsymbol{H}\right)^{-1},\ \mathbf{x}_0 = \widehat{\boldsymbol{P}}_0\boldsymbol{H}^*\boldsymbol{R}_{\boldsymbol{v}}^{-1}\boldsymbol{y}, \breve{\boldsymbol{P}}_0 = \mathsf{Diag}\left(\widehat{\boldsymbol{P}}_0\right),$$
$$\boldsymbol{R}_{\mathbf{xx}_0} = \sigma_x^2\boldsymbol{I} - \breve{\boldsymbol{P}}_0,\ \boldsymbol{K}_0 = \boldsymbol{I}$$

$$\boldsymbol{R}_{\mathbf{e},i} = \boldsymbol{R}_{\boldsymbol{v}} + \boldsymbol{H}\breve{\boldsymbol{P}}_{i-1}\boldsymbol{H}^* \tag{3.46}$$

$$\boldsymbol{D}_i = \mathsf{Diag}\left(\boldsymbol{H}^*\boldsymbol{R}_{\mathbf{e},i}^{-1}\boldsymbol{H}\right) \tag{3.47}$$

$$\boldsymbol{A}_i = (\boldsymbol{I} - \breve{\boldsymbol{P}}_{i-1}\boldsymbol{D}_i)\boldsymbol{R}_{\mathbf{xx}_{i-1}}(\boldsymbol{I} + \boldsymbol{K}_{i-1}^*\boldsymbol{D}_i\boldsymbol{R}_{\mathbf{xx}_{i-1}})^{-1} \tag{3.48}$$

$$\boldsymbol{\Gamma}_i = (\,\breve{\boldsymbol{P}}_{i-1} + \boldsymbol{A}_i\boldsymbol{K}_{i-1}^*\,) \tag{3.49}$$

$$\boldsymbol{G}_i = \boldsymbol{\Gamma}_i\boldsymbol{H}^*\boldsymbol{R}_{\mathbf{e},i}^{-1} \tag{3.50}$$

$$\mathbf{e}_i = \boldsymbol{y} - \boldsymbol{H}\breve{\mathbf{x}}_{i-1} \tag{3.51}$$

$$\hat{\mathbf{x}}_i = \boldsymbol{\Gamma}_i\boldsymbol{D}_i\breve{\mathbf{x}}_{i-1} + \boldsymbol{G}_i\mathbf{e}_i \tag{3.52}$$

$$\boldsymbol{\Upsilon}_i = \sigma_x^2\boldsymbol{I} - \boldsymbol{\Gamma}_i\boldsymbol{D}_i\left(\sigma_x^2\boldsymbol{I} - \breve{\boldsymbol{P}}_{i-1}\right)\boldsymbol{D}_i^*\boldsymbol{\Gamma}_i^* - \boldsymbol{\Gamma}_i\boldsymbol{D}_i\boldsymbol{\Gamma}_i^* \tag{3.53}$$

$$\boldsymbol{R}_{\mathbf{xx}_i} = \sigma_x^2\boldsymbol{I} - \boldsymbol{\Upsilon}_i \tag{3.54}$$

$$\boldsymbol{R}_{\mathbf{x}_i} = \boldsymbol{R}_{\mathbf{xx}_i} + \mathsf{diag}\left(|\hat{\mathbf{x}}_i - \mathbf{x}_i|^2\right)/8 \tag{3.55}$$

$$\boldsymbol{K}_i = \boldsymbol{R}_{\mathbf{xx}_i}\boldsymbol{R}_{\mathbf{x}_i}^{-1} \tag{3.56}$$

$$\breve{\mathbf{x}}_i = \boldsymbol{K}_i\mathbf{x}_i \tag{3.57}$$

$$\breve{\boldsymbol{P}}_i = \sigma_x^2\boldsymbol{I} - \boldsymbol{K}_i\boldsymbol{R}_{\mathbf{x}_i}\boldsymbol{K}_i^* \tag{3.58}$$

$$x_i(k) = \begin{cases} \hat{x}_i(k), & |\hat{x}_i(k)| > \sigma_{i,k}\sqrt{2\ln(1+\alpha)} \\ 0, & \text{otherwise} \end{cases} \tag{3.59}$$

Table 3.1: CS-based Generalized Block Iterative DFE.

where $\boldsymbol{x}_{i_{\max}}$ is the final estimated vector. The rate of success is translated by a color shading over the plane, where the sharper the transition from red to blue, the higher the detection ability of the corresponding algorithm, blue being the highest scores for a given algorithm. While Gaussian matrices are normally used in constructing the phase transition plot (since good sensing matrices can be designed for certain applications), in this work, because we deal with the more difficult case of structured matrices (more specifically, from a MIMO radar scenario in the next chapter), we shall focus not only on the performance of general purpose Gaussian-based matrices, but also on block-Toeplitz structures, arising in the MIMO Radar scenario. Moreover, the superiority of the AMP algorithm observed in [23], motivates us to adopt the CAMP recursions (1.30)–(1.32) as a basis for comparison.

Figure 3.3 shows the AFE performance of the CAMP and CS-BDFE algorithms, for complex-valued Gaussian matrices, considering a 15 dB signal-to-noise-ratio (SNR) environment. We have generated diagrams for 30 ensembles with $N = 100$, and varying both M and k in the range $[1, 100]$. We have set the parameter $\alpha = 10$ in (3.45). Similarly, we set the CAMP parameter $\kappa = 1$ in (1.33), optimized offline. Moreover, the CS-BDFE algorithm requires choosing values for the variances of signal and noise. Since $\boldsymbol{x}$ has k in N entries drawn

from a Gaussian distribution with unitary variance, then we have $\sigma_x^2 = k/N$, and $\sigma_v^2 = 10^{-1.5}\sigma_x^2$. We see that while CAMP exhibits good performance for as little as 10 observations, it does not perform as well for less sparse vectors, even when the number of observations is larger. The CS-BDFE yields small AFE only for systems with more than 50 observations, exhibiting a performance similar to the CAMP for such matrices. On the other hand, CS-BDFE yields lower AFE values than CAMP for denser vectors, and its performance degrades less abruptly as sparsity diminishes.

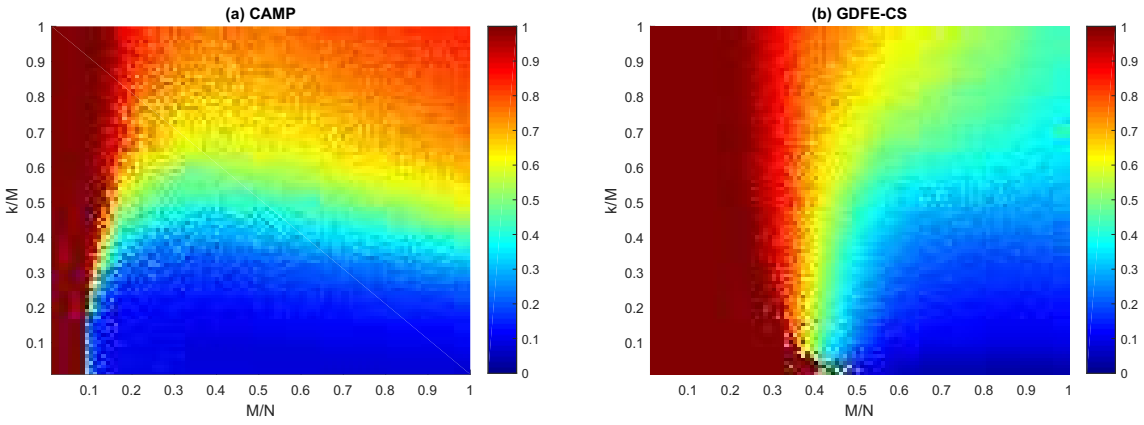


Figure 3.3: Average Fractional Error for CAMP and CS-BDFE algorithms, with 15 dB SNR.

The CAMP algorithm severely deteriorates for block-Toeplitz structured matrices, while the CS-BDFE shows improved performance, as seen in Fig. 3.4. Note that there is a much more distinct transition in the detection ability in the latter case. This is significant to our model, as the MIMO Radar to be presented in the next chapter is highly structured. In this experiment, we have set the CAMP parameter to $\kappa = 1.4$ and the CS-BDFE parameter to $\alpha = 50$.

Each matrix used in this experiment has been generated by shifting down a block column with 5 columns and $M - 19$ non-zero rows drawn from i.i.d. Gaussian sequences. The Toeplitz block was constructed by shifting down this block column by one row, 20 times. Hence, the smallest number of rows M for the sensing matrices used in the experiment is 20, so that the lowest value in the horizontal axis is 0.2.

Another useful figure of merit for radar systems is the *False Discovery Ratio*, defined as

$$\text{FDR} \triangleq \frac{\#\text{ of entries } x_i(k) \neq 0 \text{ for which } x(k) = 0}{\#\text{ of entries } x_i(k) \neq 0}.$$

That, is, non-zero entries in $\boldsymbol{x}_i$ that correspond to null entries in the true vector $\boldsymbol{x}$. These are usually referred to as *Type I* errors in detection theory. Similarly, the

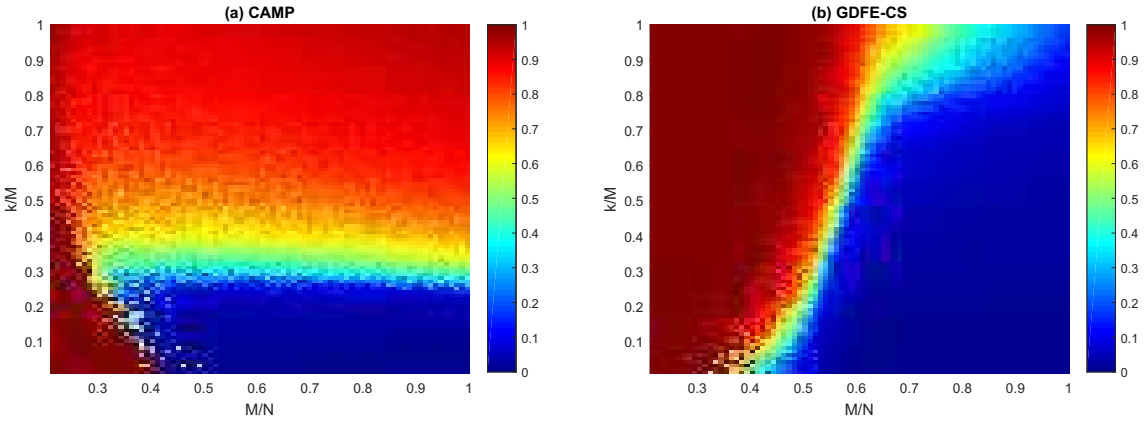


Figure 3.4: AFE for CAMP and CS-BDFE algorithms, with 15 dB SNR, for block Toeplitz sensing matrices.

False Rejection Ratio, defined as the ratio

$$\text{FRR} \triangleq \frac{\#\text{ of entries } x_i(k) = 0 \text{ for which } x(k) \neq 0}{\#\text{ of entries } x_i(k) = 0},$$

corresponds to evaluating the so-called *Type II* error, that is, null entries in $\boldsymbol{x}_i$ that correspond to non-zero entries in the true vector $\boldsymbol{x}$. Those two measurements are rather pertinent, since often in radar imaging the support is more relevant then the target reflectivity itself. Figures 3.5 (a,b) show that CS-BDFE yields better FDR scores than CAMP for less sparse vectors, even in situations of less than 50 observations. In terms of FRR, in Figs. 3.5 (c,d), CS-BDFE exhibits a better score on denser vectors, although not as pronounced as in the case of the FDR.

The FDR and FRR in the case of block-Toeplitz sensing matrices for both algorithms are depicted in Figs. 3.6 (a-d). They show that the FDR for the CAMP algorithm has improved, and yet CS-BDFE outperforms it, specially for sparser vectors. Matrix structure impairs the FRR value of the CAMP algorithm for denser vectors. For the CS-BDFE, the best FRR figures follow closely the best AFE ones. That is, for vector sparsities where the CS-BDFE best identifies the solution, it also excels in finding the right support. When the number of observations falls below transition in the AFE diagram, the CS-BDFE falsely rejects some entries. During this experiment, we also noticed that reducing the noise level increases the number of false detections observed in the case of the CS-BDFE algorithm for sparser vectors, while keeping the other scores unchanged. This was caused by a drastic reduction of the uncertainty used for determining the threshold levels, and can be compensated by an increase in the parameter α.

In the next chapter, we complete the evaluation of the CS-BDFE algorithm in light of a fully developed MIMO Radar model, where the sensing matrices become much more structured.

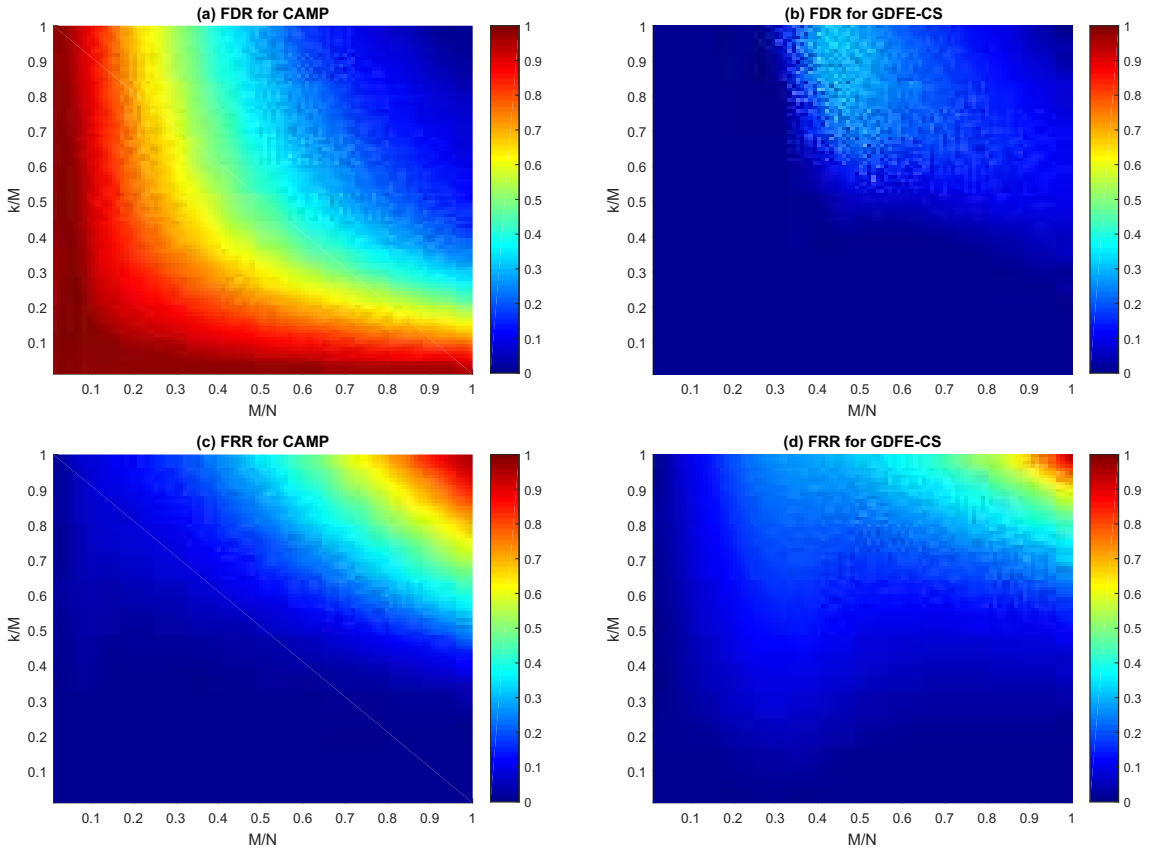


Figure 3.5: False Discovery Ratio for (a) CAMP and (b) CS-BDFE algorithms, and False Rejection Ratio for (c) CAMP and (d) CS-BDFE algorithms, with 15 dB SNR.

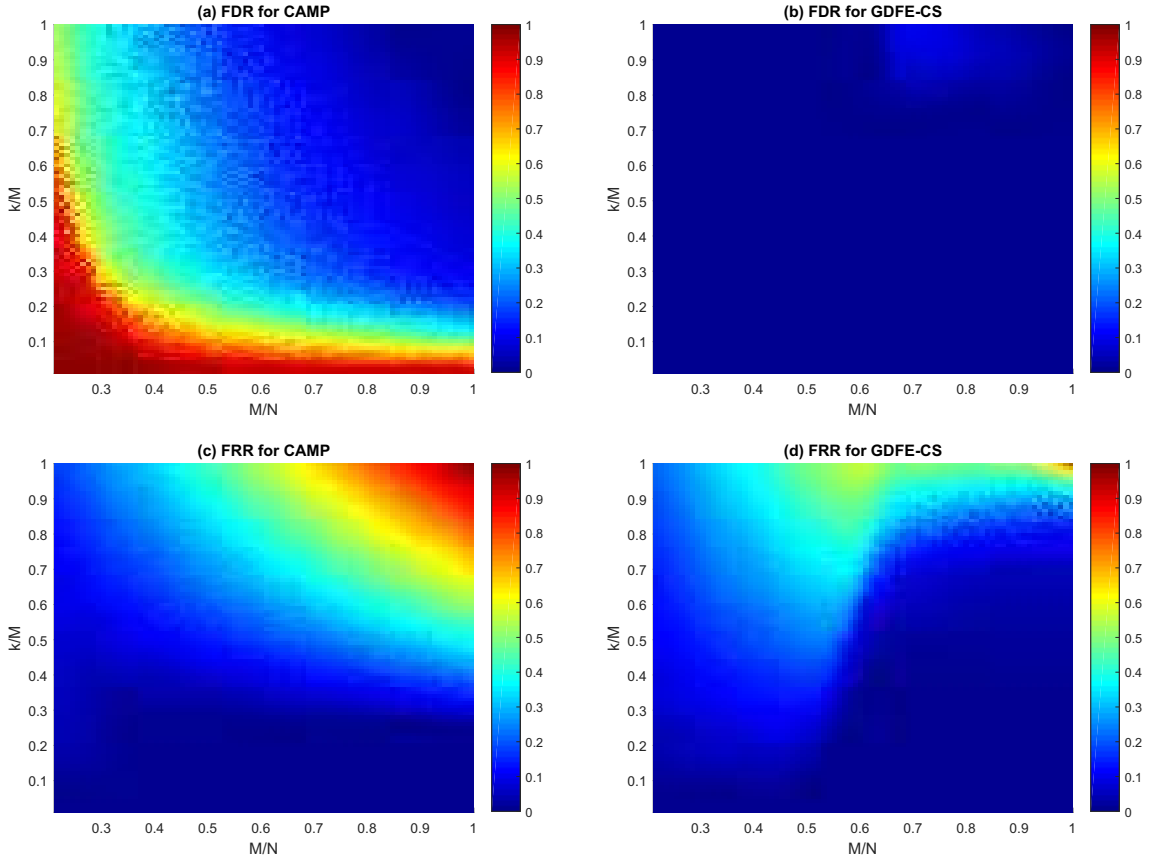


Figure 3.6: FDR for (a) CAMP and (b) CS-BDFE algorithms, and FRR for (c) CAMP and (d) CS-BDFE algorithms, with 15 dB SNR, for block-Toeplitz structured systems.

Chapter 4

3D Compressed Sensing Radar Imaging and Application of the New Algorithms

Although the theory of Compressed Sensing finds applicability in numerous situations, important advances have been achieved in fields that exploit spatially sparse scenarios. This includes Multi-Input-Multi-Output (MIMO) radar systems, which have proved successful in enhancing resolution, parameter identifiability, and robustness, specially when compared to its single-antenna and phased-array analogues [69, 70]. The use of compressed sensing for MIMO radar imaging is proposed, e.g., in [71], and further exploited in [22], where it is shown to improve cross-range resolution with randomly positioned antennas. In the latter, the well known assumptions on Nyquist spatial spacing between the antennas are no longer relevant. While in [22] the range information is captured by a bank of matched filters, CS imaging over a full 3D model has been proposed in [20], considering that each transmitting antenna sends a long Gaussian sequence as a probing pulse. MIMO radars have also found motivation in the biomedical imaging of sparsely located tumors, where detection of their support is crucial for an early diagnosis [72, 73].

Regardless of the application, the physical nature of the overall point spread function that gives rise to an approximate linear model for the radar, turns the sensing problem into a formidable task. Some figures of merit as false-rejection and false-detection rates as seen in the previous chapter become very relevant if considered in a radar scenario. For example, even with the significant progress in improving current X-ray mammography techniques, the low sensitivity and the lack of specificity of these methods, result in high false-rejection ($4\% - 34\%$), and false-detection rates, where approximately 70% of all lesions identified by an X-ray are benign tumors [74, 75]. These are useful figures of merit that extend to general

MIMO settings, and are considered in this work in order to assess the performance of the methods proposed herein.

More specifically, the major hindrance in these applications rises from the fact that, just like in the DFE setup, the inherent sensing matrix in a MIMO radar model is not fully controlled by the designer, so that not only the problem itself is naturally ill-posed, but can become highly structured (due to its defining manifold vectors). This further aggravates ill-conditioning, and leads to adverse results depending on the iterative approach envisaged.

The specific contributions of this chapter are the following:

1. We develop a full joint range/cross-range convolution model for MIMO radars and obtain conditions for which CS techniques can be employed for reconstructing the volume image.

2. After constructing such model, suitable for CS, we take a step further by decoupling it into two separate sparse problems, albeit ones that exhibit more structured independent models, suitable for efficient implementations. The recovery ability of each one is assessed by means of their coherence measure (Eq. (1.5)).

3. We validate the CS-BDFE algorithm of Chapter 3 simulating a real-world radar system, using the proposed radar model. Then using a *Finite Differences in Time Domain* (FDTD) simulation of wave propagation, we assess both the model and the algorithm performance, with favorable results.

The chapter is organized as follows. In Sec. 4.1 we review the basics of wave propagation, that will be used in Sec. 4.2 where we construct a full 3D model for volume imaging. In Sec. 4.4 we assess the recovery ability by means of its coherence, by proposing two different approaches for image reconstruction. In Sec. 4.5 we use the MIMO radar proposed in this chapter to simulate real-world radar systems, that are used to validate the CS-BDFE algorithm, considering three performance indicators, and comparing it to the CAMP. Finally, we consider a FDTD simulation of a real radar system in order to show that the proposed CS-BDFE can disregard non-linearities not taken into account in our model.

4.1 Wave Propagation

The correct model for electromagnetic waves propagation (also extended to acoustic analogues) uses the Maxwell's equations. However, when modeling a radar system,

the scalar wave propagation equation is commonly used:

$$\left(\nabla^2 - \frac{1}{c^2(\mathfrak{x})}\frac{\partial^2}{\partial^2 t}\right)\mathcal{U}(\mathfrak{x},t) = 0 \tag{4.1}$$

where $\mathcal{U}(\mathfrak{x},t)$ is either the electric or magnetic field, and $c(\mathfrak{x})$ is the speed of propagation as a function of position $\mathfrak{x}$. This equation is valid for the wave propagation in dry air, but it does not take into account polarization changes on the wave. The wave equation has shown to be adequate in some applications including ultrasound and some geophysical imaging scenarios [76].

The field in free space, generated by an impulse applied at the origin is the solution to

$$\left(\nabla^2 - \frac{1}{c^2(\mathfrak{x})}\frac{\partial^2}{\partial^2 t}\right)\mathcal{G}(\mathfrak{x},t) = -\delta(\mathfrak{x})\delta(t), \tag{4.2}$$

when $c(\mathfrak{x}) = c_0$ is constant. The solution of this equation (known as the Green's function) is given by

$$\mathcal{G}(\mathfrak{x},t) = \frac{\delta(t - \|\mathfrak{x}\|/c_0)}{4\pi\|\mathfrak{x}\|}, \tag{4.3}$$

which is time and translation invariant.

Equation (4.1) assumes a point-like radiator at the origin in a Cartesian space. If we were to consider the antenna geometry and the transmitted pulse shape, the resulting field would be given by the convolution [76, 77]

$$\mathcal{U}(\mathfrak{x},t) = \mathcal{G}(\mathfrak{x},t) \circledast [p(t)\mathcal{J}_s(\mathfrak{x})], \tag{4.4}$$

where $\mathcal{J}_s(\mathfrak{x})$ is the time-derivative of the antenna current distribution and $p(t)$ the is pulse function.

At this point, we shall consider ideal omnidirectional radiators, commonly used in MIMO radar analysis. Note that, given a source located at $\boldsymbol{\zeta}$, in free space, the incident wave at a position $\mathfrak{x}$ is

$$\mathcal{U}_{\boldsymbol{\zeta}}^{in}(\mathfrak{x},t) = \mathcal{G}(\boldsymbol{\zeta} - \mathfrak{x},t) \circledast p(t) = \frac{p(t - \|\boldsymbol{\zeta} - \mathfrak{x}\|/c_0)}{4\pi\|\boldsymbol{\zeta} - \mathfrak{x}\|} \tag{4.5}$$

since, from (4.3)

$$\mathcal{G}(\boldsymbol{\zeta} - \mathfrak{x},t) = \frac{\delta(t - \|\boldsymbol{\zeta} - \mathfrak{x}\|/c_0)}{4\pi\|\boldsymbol{\zeta} - \mathfrak{x}\|}.$$

In general, the total field received at a point-like receiver at position $\tilde{\boldsymbol{\zeta}}$ is given by the scattering solution for the wave equation (4.1) [76]

$$\mathcal{U}_{\boldsymbol{\zeta}}(\tilde{\boldsymbol{\zeta}},t) = \mathcal{U}_{\boldsymbol{\zeta}}^{in}(\tilde{\boldsymbol{\zeta}},t) + \mathcal{U}_{\boldsymbol{\zeta}}^{sc}(\tilde{\boldsymbol{\zeta}},t) \tag{4.6}$$

where $\mathcal{U}_{\boldsymbol{\zeta}}^{sc}$ is the scattered field, given by

$$\mathcal{U}_{\boldsymbol{\zeta}}^{sc}(\tilde{\boldsymbol{\zeta}},t)=\int \mathcal{G}(\tilde{\boldsymbol{\zeta}}-\mathfrak{z},t-\tau)\mathcal{V}(\mathfrak{z})\frac{\partial^2\mathcal{U}_{\boldsymbol{\zeta}}(\mathfrak{z},\tau)}{\partial^2\tau}d\tau d^3\mathfrak{z}, \tag{4.7}$$

and $\mathcal{V}(\mathfrak{x}) \triangleq c_0^{-2}-c^{-2}(\mathfrak{x})$ is a reflectivity function, containing all information about how the medium differs from free space. One common approximation, used in this presentation, is known as the *Born* or the *single scattering approximation*, where the total field $\mathcal{U}_{\mathfrak{x}}(\mathfrak{z},t)$ on the right hand side of (4.7) is replaced by the incident field $\mathcal{U}_{\mathfrak{x}}^{in}(\mathfrak{z},t)$:

$$\mathcal{U}_{\boldsymbol{\zeta}}^{sc}(\tilde{\boldsymbol{\zeta}},t)=\int \mathcal{G}(\tilde{\boldsymbol{\zeta}}-\mathfrak{z},t-\tau)\mathcal{V}(\mathfrak{z})\frac{\partial^2\mathcal{U}_{\boldsymbol{\zeta}}^{in}(\mathfrak{z},\tau)}{\partial^2\tau}d\tau d^3\mathfrak{z} \tag{4.8}$$

As pointed out in Appendix A.1 of [76], the Born approximation allows the field from a point scatterer to be well defined and nonzero. In this case the scattered field due to a target at $\bar{\boldsymbol{r}}$ can be obtained just by setting $\mathcal{V}(\mathfrak{x})=\bar{s}\delta(\bar{\boldsymbol{r}}-\mathfrak{x})$:

$$\mathcal{U}_{\boldsymbol{\zeta}}^{sc}(\tilde{\boldsymbol{\zeta}},t)=\mathcal{G}(\bar{\boldsymbol{r}}-\tilde{\boldsymbol{\zeta}},t)\circledast\left(\bar{s}\frac{\partial^2\mathcal{U}_{\boldsymbol{\zeta}}^{in}(\bar{\boldsymbol{r}},t)}{\partial^2 t}\right), \tag{4.9}$$

where $\bar{s}$ is the scalar denoting the reflectivity coefficient of the target.

4.2 MIMO Radar Modeling

Consider an array of M_T isotropic transmitters each positioned at $\boldsymbol{\zeta}_i$, $i=0,1,\ldots,M_T$. The signal is scattered by K point targets located at the same range D, at positions $\bar{\boldsymbol{r}}_k$, $k=0,1,\ldots,K$, and received by an array of M_R receivers positioned at $\tilde{\boldsymbol{\zeta}}_j$, $j=0,1,\ldots,M_R$, as shown in Fig. 4.1. Also, we shall denote by $\{\boldsymbol{\zeta}_o,\tilde{\boldsymbol{\zeta}}_o\}$ the geometric centers of the arrays.

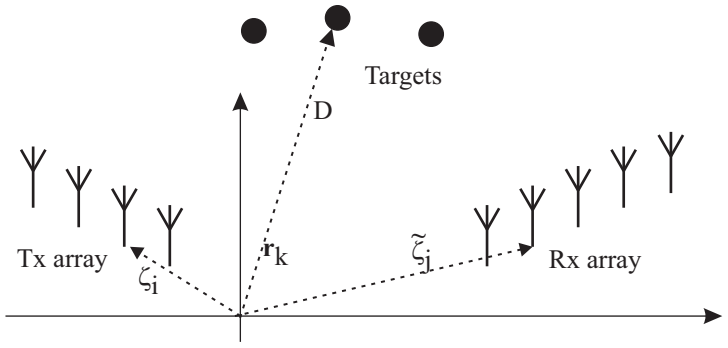


Figure 4.1: Simplified MIMO radar model

Using (4.9), if each antenna transmits a pulse $p_i(t)$, the signals received by the

sensors can be collected into a $M_R \times 1$ size vector given by the convolution sum

$$\boldsymbol{y}_{\bar{\boldsymbol{r}}}(t) = \boldsymbol{\mathcal{H}}_{\bar{\boldsymbol{r}}}(t) \circledast \underbrace{\begin{bmatrix} \bar{s}_0 & & & \\ & \bar{s}_1 & & \\ & & \ddots & \\ & & & \bar{s}_{K-1} \end{bmatrix}}_{\boldsymbol{S}_{\bar{\boldsymbol{r}}}} \frac{\partial^2}{\partial^2 t} \underbrace{\left(\boldsymbol{\mathcal{G}}_{\bar{\boldsymbol{r}}}(t) \circledast \begin{bmatrix} p_0(t) \\ p_1(t) \\ \vdots \\ p_{M_T-1}(t) \end{bmatrix} \right)}_{\boldsymbol{u}_{\bar{\boldsymbol{r}}}(t)} \tag{4.10}$$

Here, each $\bar{s}_k$ denotes the reflectance of the target at $\bar{\boldsymbol{r}}_k$, which we collect into a diagonal matrix $\boldsymbol{S}_{\bar{\boldsymbol{r}}} = \mathrm{diag}[\bar{s}_0 \dots \bar{s}_{K-1}]$, and $\boldsymbol{\mathcal{G}}_{\bar{\boldsymbol{r}}}(t)$ is a matrix containing the corresponding Green's functions of the underlying medium, from point $\boldsymbol{\zeta}_i$ to $\bar{\boldsymbol{r}}_k$:

$$\boldsymbol{\mathcal{G}}_{\bar{\boldsymbol{r}}}(t) = \begin{bmatrix} \mathcal{g}_0(\boldsymbol{\zeta}_0, t) & \mathcal{g}_0(\boldsymbol{\zeta}_1, t) & \cdots & \mathcal{g}_0(\boldsymbol{\zeta}_{M_T-1}, t) \\ \mathcal{g}_1(\boldsymbol{\zeta}_0, t) & \mathcal{g}_1(\boldsymbol{\zeta}_1, t) & \cdots & \mathcal{g}_1(\boldsymbol{\zeta}_{M_T-1}, t) \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{g}_{K-1}(\boldsymbol{\zeta}_0, t) & \mathcal{g}_{K-1}(\boldsymbol{\zeta}_1, t) & \cdots & \mathcal{g}_{K-1}(\boldsymbol{\zeta}_{M_T-1}, t) \end{bmatrix},$$

where $\mathcal{g}_i(\boldsymbol{\zeta}_j, t) = \mathcal{G}(\bar{\boldsymbol{r}}_i - \boldsymbol{\zeta}_j, t)$. Similarly, $\boldsymbol{\mathcal{H}}_{\bar{\boldsymbol{r}}}(t)$ is a matrix whose elements correspond to the Green's functions that define the path from the targets back to the receivers.

We also define the incident signal in (4.10) as:

$$\boldsymbol{u}_{\bar{\boldsymbol{r}}}(t) \triangleq \boldsymbol{\mathcal{G}}_{\bar{\boldsymbol{r}}}(t) \circledast \boldsymbol{p}(t) = \begin{bmatrix} \vdots \\ \sum_{i=0}^{M_T-1} \mathcal{g}_k(\boldsymbol{\zeta}_i, t) \circledast p_i(t) \\ \vdots \end{bmatrix} \tag{4.11}$$

where $\boldsymbol{p}(t) \triangleq \begin{bmatrix} p_0(t) & p_1(t) & \dots & p_{M_T-1}(t) \end{bmatrix}^{\mathsf{T}}$.

Now, as we are considering a free space scenario, applying (4.5) to (4.11), the k-th element of $\boldsymbol{u}_{\bar{\boldsymbol{r}}}(t)$ is expressed by:

$$\boldsymbol{u}_{\bar{\boldsymbol{r}}_k}(t) = \sum_{i=0}^{M_T-1} \mathcal{g}_k(\boldsymbol{\zeta}_i, t) \circledast p_i(t) = \sum_{i=0}^{M_T-1} \frac{p_i(t - \|\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_i\|/c_0)}{4\pi \|\bar{\boldsymbol{r}} - \boldsymbol{\zeta}_i\|} \tag{4.12}$$

In order to arrive at a MIMO radar linear model, we shall further consider two common approximations, namely, a narrowband and a far-field approximation.

Narrowband approximation

If we assume $p_i(t)$ to be narrowband, we can model it as a slow varying envelope $\tilde{p}_i(t)$ modulating a complex exponential at a frequency ω_0, say, $p_i(t) = \tilde{p}_i(t)e^{j\omega_0 t}$. Hence, the k-th second-order derivative in (4.10) is given by

$$\begin{aligned}\frac{\partial^2}{\partial t^2}\boldsymbol{u}_{\bar{\boldsymbol{r}}_k}(t) &= \frac{1}{4\pi}\sum_{i=0}^{M_T-1}\frac{p_i''(t-\|\bar{\boldsymbol{r}}_k-\boldsymbol{\zeta}_i\|/c_0)}{\|\bar{\boldsymbol{r}}_k-\boldsymbol{\zeta}_i\|} \\ &\simeq -\frac{\omega_0^2}{4\pi}\sum_{i=0}^{M_T-1}\frac{e^{j\omega_0(t-\tau_i(\bar{\boldsymbol{r}}_k))}}{D_{ki}}\tilde{p}_i(t-\tau_i(\bar{\boldsymbol{r}}_k)) \end{aligned} \tag{4.13}$$

where $p''(\cdot)$ denotes the second derivative with respect to time, $D_{ki} \triangleq \|\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_i\|$, and $\tau_i(\bar{\boldsymbol{r}}_k) \triangleq D_{ki}c_0^{-1}$.

Using (4.12), let us define

$$\bar{\boldsymbol{a}}(\bar{\boldsymbol{r}}_k) \triangleq \begin{bmatrix} \frac{e^{j\omega_0(t-\tau_0(\bar{\boldsymbol{r}}_k))}}{D_{k0}} \\ \frac{e^{j\omega_0(t-\tau_1(\bar{\boldsymbol{r}}_k))}}{D_{k1}} \\ \vdots \\ \frac{e^{j\omega_0\left(t-\tau_{M_T-1}(\bar{\boldsymbol{r}}_k)\right)}}{D_{k(M_T-1)}} \end{bmatrix}. \tag{4.14}$$

Equation (4.13) can thus be written as:

$$\frac{\partial^2}{\partial t^2}\boldsymbol{u}_{\bar{\boldsymbol{r}}_k}(t) = -\frac{\omega_0^2}{4\pi}\bar{\boldsymbol{a}}^{\intercal}(\bar{\boldsymbol{r}}_k)\begin{bmatrix} \tilde{p}_0\left(t-\tau_0(\bar{\boldsymbol{r}}_k)\right) \\ \tilde{p}_1\left(t-\tau_1(\bar{\boldsymbol{r}}_k)\right) \\ \vdots \\ \tilde{p}_{M_T-1}\left(t-\tau_{M_T-1}(\bar{\boldsymbol{r}}_k)\right) \end{bmatrix} \tag{4.15}$$

Up to this point, we can conclude that each target is illuminated by delayed replicas of the source pulses, attenuated by the free-space path loss. There is one more consequence of the narrowband approximation. First, consider the geometric center of the transmitting array, which we denote by $\boldsymbol{\zeta}_o$, and define the maximum travel time within the array range as

$$\Delta T_{max} \triangleq \max_{i=0\ldots M_T-1}\frac{\|\boldsymbol{\zeta}_i - \boldsymbol{\zeta}_o\|}{c_0}$$

Given that $p_i(t)$ is a slow-varying envelope with bandwidth B_p, and assuming that $B_p\Delta T_{max} \ll 1$, then [78]

$$\tilde{p}_i(t-\tau_i(\bar{\boldsymbol{r}}_k)) \simeq \tilde{p}_i(t-\tau(\bar{\boldsymbol{r}}_k)) \tag{4.16}$$

where $\tau(\boldsymbol{r}_k)$ is the time delay from the center of the array to the targets' common range

$$\tau(\bar{\boldsymbol{r}}_k) \triangleq \frac{\|\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_o\|}{c_0}.$$

Also, note that for targets along the same range, we can define

$$D_{ki} \simeq D \triangleq \|\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_o\|, \tag{4.17}$$

and

$$\tau_r \triangleq \tau(\bar{\boldsymbol{r}}_k),$$

for all targets. In this way, the narrowband approximation becomes

$$\frac{\partial^2}{\partial t^2}\boldsymbol{u}_{\bar{\boldsymbol{r}}_k}(t) = -\frac{\omega_0^2}{4\pi}\bar{\boldsymbol{a}}^{\mathsf{T}}(\bar{\boldsymbol{r}}_k)\tilde{\boldsymbol{p}}\left(t-\tau_r\right), \tag{4.18}$$

where $\tilde{\boldsymbol{p}}(t) \triangleq [\tilde{p}_0(t)\ldots\tilde{p}_{M_T-1}(t)]^{\mathsf{T}}$.

Far-field approximation

If the targets are in the far-field, the Fraunhoffer approximation is commonly used:

$$\|\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_i\| \simeq \|\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_o\| - \widehat{\left(\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_o\right)}^{\mathsf{T}} \boldsymbol{q}_i,$$

where $\boldsymbol{q}_i \triangleq \boldsymbol{\zeta}_i - \boldsymbol{\zeta}_o$ denotes the vector pointing from the center of the array $\boldsymbol{\zeta}_o$ to the antenna at $\boldsymbol{\zeta}_i$, and $\widehat{(\cdot)}$ denotes a unit vector in the same direction as $(\cdot)$. Then the delays $\tau_i(\bar{\boldsymbol{r}}_k)$ become

$$\begin{aligned}\tau_i(\bar{\boldsymbol{r}}_k) \triangleq \frac{\|\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_i\|}{c_0} &\simeq \frac{\|\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_o\|}{c_0} - \frac{\widehat{\left(\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_o\right)}^{\mathsf{T}} \boldsymbol{q}_i}{c_0} \\ &= \tau_r - \frac{\widehat{\left(\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_o\right)}^{\mathsf{T}} \boldsymbol{q}_i}{c_0},\end{aligned}$$

so that from (4.15) we obtain

$$\tilde{p}_i(t-\tau_i(\bar{\boldsymbol{r}}_k))e^{j\omega_0(t-\tau_i(\bar{\boldsymbol{r}}_k))} = \tilde{p}_i(t-\tau_r)e^{j\omega_0(t-\tau_r)}e^{j\frac{\omega_0}{c_0}\widehat{\left(\bar{\boldsymbol{r}}_k-\boldsymbol{\zeta}_o\right)}^{\mathsf{T}}\boldsymbol{q}_i}. \tag{4.19}$$

Considering $\|\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_i\| \gg \widehat{\left(\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_o\right)}^{\mathsf{T}} \boldsymbol{q}_i$, and defining

$$\boldsymbol{d}_k \triangleq \widehat{\left(\bar{\boldsymbol{r}}_k - \boldsymbol{\zeta}_o\right)}\frac{\omega_0}{c_0}$$

using (4.17), we have that $\bar{\boldsymbol{a}}(\bar{\boldsymbol{r}}_k)$ in (4.14) can be written as

$$\bar{\boldsymbol{a}}(\bar{\boldsymbol{r}}_k) = -\frac{e^{j\omega_0(t-\tau_r)}}{D}\boldsymbol{a}(\bar{\boldsymbol{r}}_k), \tag{4.20}$$

where $\boldsymbol{a}(\bar{\boldsymbol{r}}_k)$ is the so called *array manifold* vector associated to the transmitting antennas:

$$\boldsymbol{a}(\bar{\boldsymbol{r}}_k) = \begin{bmatrix} e^{j\boldsymbol{d}_k^{\mathsf{T}}\boldsymbol{q}_0} \\ e^{j\boldsymbol{d}_k^{\mathsf{T}}\boldsymbol{q}_1} \\ \vdots \\ e^{j\boldsymbol{d}_k^{\mathsf{T}}\boldsymbol{q}_{M_T-1}} \end{bmatrix}. \tag{4.21}$$

Substituting (4.20) into (4.15), we arrive at

$$\begin{aligned} \frac{\partial^2}{\partial t^2}u_{\bar{\boldsymbol{r}}_k}(t) &= -\frac{\omega_0^2}{4\pi D}e^{j\omega_0(t-\tau_r)}\boldsymbol{a}^{\mathsf{T}}(\bar{\boldsymbol{r}}_k)\tilde{\boldsymbol{p}}(t-\tau_r) \\ &= -\frac{\omega_0^2}{4\pi D}\boldsymbol{a}^{\mathsf{T}}(\bar{\boldsymbol{r}}_k)\boldsymbol{p}(t-\tau_r). \end{aligned} \tag{4.22}$$

Now, returning to Eq. (4.10), and using (4.22), we can write

$$\boldsymbol{y}_{\bar{\boldsymbol{r}}}(t) = \boldsymbol{\mathcal{H}}_{\bar{\boldsymbol{r}}}(t) \circledast \boldsymbol{S}_{\bar{\boldsymbol{r}}}\frac{-\omega_0^2}{4\pi D}\boldsymbol{\mathcal{A}}^{\mathsf{T}}(\bar{\boldsymbol{r}})\boldsymbol{p}(t-\tau_r) \tag{4.23}$$

where $\boldsymbol{A}(\bar{\boldsymbol{r}})$ is a matrix containing the manifold vectors corresponding to all targets at the same range:

$$\boldsymbol{\mathcal{A}}(\bar{\boldsymbol{r}}) = \begin{bmatrix} \vdots & \vdots & & \vdots \\ \boldsymbol{a}(\bar{\boldsymbol{r}}_0) & \boldsymbol{a}(\bar{\boldsymbol{r}}_1) & \cdots & \boldsymbol{a}(\bar{\boldsymbol{r}}_{K-1}) \\ \vdots & \vdots & & \vdots \end{bmatrix}. \tag{4.24}$$

Analogously, using the narrowband and far-field approximations for the return path (i.e., from the targets to the receiving antennas), we write equivalently

$$\boldsymbol{y}_{\bar{\boldsymbol{r}}}(t) = -\frac{\omega_0^2}{16\pi^2 D^2}\boldsymbol{\mathcal{B}}(\bar{\boldsymbol{r}})\boldsymbol{S}_{\bar{\boldsymbol{r}}}\boldsymbol{\mathcal{A}}^{\mathsf{T}}(\bar{\boldsymbol{r}})\boldsymbol{p}(t-2\tau_r), \tag{4.25}$$

where $\boldsymbol{\mathcal{B}}(\bar{\boldsymbol{r}})$ follows the same structure of $\boldsymbol{\mathcal{A}}(\bar{\boldsymbol{r}})$,

$$\boldsymbol{\mathcal{B}}(\bar{\boldsymbol{r}}) = \begin{bmatrix} \vdots & \vdots & & \vdots \\ \boldsymbol{b}(\bar{\boldsymbol{r}}_0) & \boldsymbol{b}(\bar{\boldsymbol{r}}_1) & \cdots & \boldsymbol{b}(\bar{\boldsymbol{r}}_{K-1}) \\ \vdots & \vdots & & \vdots \end{bmatrix}.$$

Each $\boldsymbol{b}(\bar{\boldsymbol{r}}_k)$ is the manifold vector of the receiving array, relatively to $\bar{\boldsymbol{r}}_k$, and to the geometric center of the receiving array, $\tilde{\boldsymbol{\zeta}}_o$. Here, we consider that τ_r is approximately

the same for the trasmitting and receiving paths, since, from our initial arrangement, the transmitting and receiving arrays are colocated, and we can choose $\{\boldsymbol{\zeta}_o, \tilde{\boldsymbol{\zeta}}_o\}$ to coincide. The spatial invariance of the Green's functions also makes it possible to set the centers at the origin of the cartesian space $\boldsymbol{\zeta}_o = \tilde{\boldsymbol{\zeta}}_o = \mathbf{0}$, which simplifies the analysis of array geometries.

Let

$$\bar{x}_k \triangleq -\frac{\omega_0^2}{16\pi^2 D^2}\bar{s}_k$$

be the free space corrected reflectance. Equation (4.25) can be written as:

$$\begin{aligned} \boldsymbol{y}_{\boldsymbol{r}}(t) &= \sum_{k=0}^{K-1} \bar{x}_k \boldsymbol{b}(\boldsymbol{r}_k)\boldsymbol{p}^{\mathsf{T}}(t-2\tau_r)\boldsymbol{a}(\boldsymbol{r}_k) && (4.26)\\ &= \sum_{k=0}^{K-1} \bar{x}_k \left[\boldsymbol{p}^{\mathsf{T}}(t-2\tau_r)\boldsymbol{a}(\bar{\boldsymbol{r}}_k)\right] \otimes \boldsymbol{b}(\bar{\boldsymbol{r}}_k). && (4.27) \end{aligned}$$

Note that this summation can be represented in matrix form as

$$\begin{aligned} \boldsymbol{y}_{\bar{\boldsymbol{r}}}(t) &= \begin{bmatrix} \boldsymbol{f}_{\bar{\boldsymbol{r}}_0}(t-2\tau_r) & \cdots & \boldsymbol{f}_{\bar{\boldsymbol{r}}_{K-1}}(t-2\tau_r) \end{bmatrix} \begin{bmatrix} \bar{x}_0 \\ \vdots \\ \bar{x}_{K-1} \end{bmatrix} && (4.28)\\ &\triangleq \boldsymbol{\mathcal{F}}_{\bar{\boldsymbol{r}}}(t-2\tau_r)\bar{\boldsymbol{x}}_{\boldsymbol{r}} && (4.29) \end{aligned}$$

where

$$\begin{aligned} \boldsymbol{f}_{\bar{\boldsymbol{r}}_k}(t) &= \left[\boldsymbol{p}^{\mathsf{T}}(t)\boldsymbol{a}(\bar{\boldsymbol{r}}_k)\right] \otimes \boldsymbol{b}(\bar{\boldsymbol{r}}_k) && (4.30)\\ &= \left[\boldsymbol{p}^{\mathsf{T}}(t) \otimes \boldsymbol{I}_{M_R}\right]\left[\boldsymbol{a}(\bar{\boldsymbol{r}}_k) \otimes \boldsymbol{b}(\bar{\boldsymbol{r}}_k)\right], && (4.31) \end{aligned}$$

and $\bar{\boldsymbol{x}}_{\boldsymbol{r}} = \begin{bmatrix} \bar{x}_0 & \cdots & \bar{x}_{K-1} \end{bmatrix}^{\mathsf{T}}$.

A closer look at (4.28) and (4.31) reveals that the received signal is equivalent to the sum of M_R replicas of each transmitted signal, after being apodized by the combined steering vector $\boldsymbol{a}(\bar{\boldsymbol{r}}_k) \otimes \boldsymbol{b}(\bar{\boldsymbol{r}}_k)$, and scaled by the reflectivity of each target.

If we sample the received signals in $\boldsymbol{y}_{\bar{\boldsymbol{r}}}(t)$ at $t = 2\tau_r + nt_s$, $n = 0, \ldots, N-1$, then (4.28) can be written as

$$\boldsymbol{y}_{\bar{\boldsymbol{r}}} = \begin{bmatrix} \boldsymbol{f}_{\bar{\boldsymbol{r}}_0} & \cdots & \boldsymbol{f}_{\bar{\boldsymbol{r}}_{K-1}} \end{bmatrix} \bar{\boldsymbol{x}}_{\boldsymbol{r}} \triangleq \boldsymbol{\mathcal{F}}_{\bar{\boldsymbol{r}}}\bar{\boldsymbol{x}}_{\boldsymbol{r}}, \qquad (4.32)$$

where

$$\boldsymbol{y}_{\bar{\boldsymbol{r}}} = \begin{bmatrix} \boldsymbol{y}_{\bar{\boldsymbol{r}}}(N-1) \\ \boldsymbol{y}_{\bar{\boldsymbol{r}}}(N-2) \\ \vdots \\ \boldsymbol{y}_{\bar{\boldsymbol{r}}}(0) \end{bmatrix}, \tag{4.33}$$

and

$$\begin{aligned} \boldsymbol{f}_{\bar{\boldsymbol{r}}_k} &= \begin{bmatrix} \boldsymbol{f}_{\bar{\boldsymbol{r}}_k}(N-1) \\ \boldsymbol{f}_{\bar{\boldsymbol{r}}_k}(N-2) \\ \vdots \\ \boldsymbol{f}_{\bar{\boldsymbol{r}}_k}(0) \end{bmatrix} = \begin{bmatrix} [\boldsymbol{p}^{\mathsf{T}}(N-1)\boldsymbol{a}(\bar{\boldsymbol{r}}_k)] \otimes \boldsymbol{b}(\bar{\boldsymbol{r}}_k) \\ [\boldsymbol{p}^{\mathsf{T}}(N-2)\boldsymbol{a}(\bar{\boldsymbol{r}}_k)] \otimes \boldsymbol{b}(\bar{\boldsymbol{r}}_k) \\ \vdots \\ [\boldsymbol{p}^{\mathsf{T}}(0)\boldsymbol{a}(\bar{\boldsymbol{r}}_k)] \otimes \boldsymbol{b}(\bar{\boldsymbol{r}}_k) \end{bmatrix} \qquad (4.34) \\ &= \begin{bmatrix} \boldsymbol{p}^{\mathsf{T}}(N-1)\boldsymbol{a}(\bar{\boldsymbol{r}}_k) \\ \boldsymbol{p}^{\mathsf{T}}(N-2)\boldsymbol{a}(\bar{\boldsymbol{r}}_k) \\ \vdots \\ \boldsymbol{p}^{\mathsf{T}}(0)\boldsymbol{a}(\bar{\boldsymbol{r}}_k) \end{bmatrix} \otimes \boldsymbol{b}(\bar{\boldsymbol{r}}_k) \\ &= [\boldsymbol{\mathcal{P}}\boldsymbol{a}(\bar{\boldsymbol{r}}_k)] \otimes \boldsymbol{b}(\bar{\boldsymbol{r}}_k) \qquad (4.35) \\ &= [\boldsymbol{\mathcal{P}} \otimes \boldsymbol{I}_{M_R}]\,[\boldsymbol{a}(\bar{\boldsymbol{r}}_k) \otimes \boldsymbol{b}(\bar{\boldsymbol{r}}_k)]\,, \qquad (4.36) \end{aligned}$$

where in the sampled version of $\{\boldsymbol{y}_{\bar{\boldsymbol{r}}}(t), \boldsymbol{\mathcal{F}}_{\bar{\boldsymbol{r}}}(t)\}$ we have removed the time indexes from (4.29). The resulting sampled MIMO pulse is N samples long, which we define the by a vector pulse $\boldsymbol{\mathcal{P}}$ as

$$\boldsymbol{\mathcal{P}} \triangleq \begin{bmatrix} \boldsymbol{p}^{\mathsf{T}}(N-1) \\ \boldsymbol{p}^{\mathsf{T}}(N-2) \\ \vdots \\ \boldsymbol{p}^{\mathsf{T}}(0) \end{bmatrix} = \begin{bmatrix} p_0(N-1) & p_1(N-1) & \cdots & p_{M_T-1}(N-1) \\ p_0(N-2) & p_1(N-2) & \cdots & p_{M_T-1}(N-2) \\ \vdots & \vdots & \vdots & \vdots \\ p_0(0) & p_1(0) & \cdots & p_{M_T-1}(0) \end{bmatrix}. \tag{4.37}$$

We can expect matrix $\boldsymbol{\mathcal{F}}_{\bar{\boldsymbol{r}}}$ to be ill-conditioned. That is, observe that its column structure determines an upper bound for its rank. Although the product $\boldsymbol{\mathcal{P}}\boldsymbol{a}(\bar{\boldsymbol{r}}_k)$ is a $N \times 1$ vector, it has only M_T degrees of freedom (the inner dimension in the product); As a result, each column $\boldsymbol{f}_{\boldsymbol{r}_k}$ will have at most $M_T M_R$ degrees of freedom, and hence, $\text{rank}(\boldsymbol{\mathcal{F}}_{\bar{\boldsymbol{r}}}) \leq M_T M_R$. In general, we expect to probe $K \gg M_T M_R$ directions, which turns $\boldsymbol{\mathcal{F}}_{\bar{\boldsymbol{r}}}$ into a very ill-conditioned matrix.

4.3 Beamforming

A typical application of a radar system is to create an image from the measured data. This means recovering not only the $\bar{x}_k$ (or $\bar{s}_k$), but also the target positions

$\bar{\boldsymbol{r}}_k$. For the sake of simplicity, most systems separate the problem in two steps, solved individually at range and cross-range. In order to improve range resolution, a typical system also makes use of a compressible pulse, i.e., one that generates a very short peak when passed through a matched filter[76]. Cross-range imaging processes the signal received from a fixed range, generally assuming that the targets are well resolved along a longitudinal line in the far-field.

4.3.1 Phased-array systems

A special case of the MIMO radar described in the previous section is the so called *phased-array system*. In this kind of radar, a single pulse $p(t)$ is sent by all transmitters. That is, let $\boldsymbol{p}(t) = \mathbf{1}_{M_T \times 1} p(t)$, where

$$\mathbf{1}_{M_T \times 1} = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}^{\mathsf{T}}. \tag{4.38}$$

Eq. (4.26) then yields

$$\boldsymbol{y}_{\bar{\boldsymbol{r}}}(t) = p(t - 2\tau_r) \sum_{k=0}^{K-1} \bar{x}_k \boldsymbol{b}(\bar{\boldsymbol{r}}_k) \mathbf{1}_{1 \times M_T} \boldsymbol{a}(\bar{\boldsymbol{r}}_k) \tag{4.39}$$

As a means to increase diversity, one considers a delayed transmission in each sensor, by replacing the $\mathbf{1}_{M_T \times 1}$ vector in (4.39) with an apodizing vector $\boldsymbol{\beta}_T(\boldsymbol{r}_s)$:

$$\boldsymbol{y}_{\bar{\boldsymbol{r}}}(t) = p(t - 2\tau_r) \sum_{k=0}^{K-1} \bar{x}_k \boldsymbol{b}(\bar{\boldsymbol{r}}_k) \boldsymbol{\beta}_T^{\mathsf{T}}(\boldsymbol{r}_s) \boldsymbol{a}(\bar{\boldsymbol{r}}_k) \tag{4.40}$$

The vector $\boldsymbol{\beta}_T(\boldsymbol{r}_s)$ corresponds to a steering vector pointing to an arbitrary position $\boldsymbol{r}_s$ and whose structure is similar to one of the manifold vector $\boldsymbol{a}(\bar{\boldsymbol{r}}_k)$ in (4.21). Considering that these quantities are formed by complex exponentials, by the Cauchy-Schwarz inequality, the magnitude of the inner product $\boldsymbol{\beta}_T^{\mathsf{T}}(\boldsymbol{r}_s)\boldsymbol{a}(\bar{\boldsymbol{r}}_k)$ will attain its maximum when $\boldsymbol{\beta}_T(\boldsymbol{r}_s)$ is the conjugate of $\boldsymbol{a}(\bar{\boldsymbol{r}}_k)$.

Usually, a phased-array system operates as a *delay-and-sum* array, that is, each received signal is weighted by another apodizing vector. In this case, the received vector becomes a scalar quantity $y_{r_s}(t) = \boldsymbol{\beta}_R^{\mathsf{T}}(\boldsymbol{r}_s)\boldsymbol{y}_{\bar{\boldsymbol{r}}}(t)$, i.e.,

$$y_{r_s}(t) = p(t - 2\tau_r) \sum_{k=0}^{K-1} \bar{x}_k \boldsymbol{\beta}_R^{\mathsf{T}}(\boldsymbol{r}_s) \boldsymbol{b}(\bar{\boldsymbol{r}}_k) \boldsymbol{a}^{\mathsf{T}}(\bar{\boldsymbol{r}}_k) \boldsymbol{\beta}_T(\boldsymbol{r}_s). \tag{4.41}$$

Notice again that the vector $\boldsymbol{\beta}_R(\boldsymbol{r}_s)$ must be the conjugate of $\boldsymbol{b}(\bar{\boldsymbol{r}}_k)$, so as to maximize the signal energy in $y_{r_s}(t)$.

As mentioned in the introduction of this section, for recovering an image from

the received signal, we would have to know each $\bar{\boldsymbol{r}}_k$ *a priori*. The simplest way to create an image without this knowledge is known as *beam scanning*: a radar system would vary the steering vectors $\boldsymbol{\beta_T}(\boldsymbol{r}_s)$ and $\boldsymbol{\beta_R}(\boldsymbol{r}_s)$ so as to scan the entire space, looking for the maxima of $y_{r_s}(t)$.

To model a beam scanning system, one can replace $\bar{\boldsymbol{x}}_{\boldsymbol{r}}$ in (4.29) by a larger vector $\boldsymbol{x_r} \triangleq \begin{bmatrix} x_0 & \cdots & x_G \end{bmatrix}^\intercal$ containing the reflectivity seen from each direction in a fine grid containing G points at $\boldsymbol{r}_k$. We assume that all targets are positioned exactly in the grid, meaning that $\{\boldsymbol{r}_k | 0 \leq k \leq G-1\} \supseteq \{\bar{\boldsymbol{r}}_k | 0 \leq k \leq K-1\}$. In this way, grid points that do not contain a true target will have zero reflectance. We cannot probe the entire space with a single transmission, as some target directions may correspond to the nulls of $\boldsymbol{\beta}_T^\intercal(\boldsymbol{r}_s)\boldsymbol{a}(\bar{\boldsymbol{r}}_k)$ or $\boldsymbol{\beta}_R^\intercal(\boldsymbol{r}_s)\boldsymbol{b}(\bar{\boldsymbol{r}}_k)$. On the other hand, we can probe several times, steering in different directions (This of course will only work if the target reflectivity varies very slowly). Therefore, after probing G directions, we can write

$$\begin{aligned} \boldsymbol{y_r}(t) = \begin{bmatrix} y_0(t) \\ y_1(t) \\ \vdots \\ y_G(t) \end{bmatrix} =& p(t-2\tau_r) \left(\begin{bmatrix} \cdots \boldsymbol{\beta}_R^\intercal(\boldsymbol{r}_0) \cdots \\ \vdots \\ \cdots \boldsymbol{\beta}_R^\intercal(\boldsymbol{r}_G) \cdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots \\ \boldsymbol{b}(\bar{\boldsymbol{r}}_0) \cdots \boldsymbol{b}(\bar{\boldsymbol{r}}_K) \\ \vdots & \vdots \end{bmatrix} \right) \odot \\ & \odot \left(\begin{bmatrix} \cdots \boldsymbol{\beta}_T^\intercal(\boldsymbol{r}_0) \cdots \\ \vdots \\ \cdots \boldsymbol{\beta}_T^\intercal(\boldsymbol{r}_G) \cdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots \\ \boldsymbol{a}(\bar{\boldsymbol{r}}_0) \cdots \boldsymbol{a}(\bar{\boldsymbol{r}}_K) \\ \vdots & \vdots \end{bmatrix} \right) \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_G \end{bmatrix} \end{aligned} \tag{4.42}$$

or

$$\boldsymbol{y}_{\tilde{\boldsymbol{r}}}(t) = p(t-2\tau_r)\boldsymbol{H_R}(\boldsymbol{r}) \odot \boldsymbol{H_T}(\boldsymbol{r})\boldsymbol{x_r} \tag{4.43}$$

where $\odot$ denotes the Haddamard product operator. Even though this model is linear, we cannot recover x_k exactly from $\boldsymbol{y_r}(t)$, as both matrices $\boldsymbol{H_R}(\boldsymbol{r})$ and $\boldsymbol{H_T}(\boldsymbol{r})$ are rank deficient, with $\text{rank}(\boldsymbol{H_R}(\boldsymbol{r})) = M_R$ and $\text{rank}(\boldsymbol{H_T}(\boldsymbol{r})) = M_T$. This represents a very ill-posed system, since it is a well-known fact that $\text{rank}(\boldsymbol{H_R}(\boldsymbol{r}) \odot \boldsymbol{H_T}(\boldsymbol{r})) \leq M_R M_T$ [79], and G is usually greater than $M_T M_R$.

One useful notion which arises from the manifold vectors combinations is the beampattern function [78], which, for the transmitting antenna, is given by

$$\Upsilon_T(\boldsymbol{r}_i, \boldsymbol{r}_j) = \boldsymbol{a}^*(\boldsymbol{r}_i)\boldsymbol{a}(\boldsymbol{r}_j), \tag{4.44}$$

with analogous definition for the receiving beampattern $\Upsilon_R(\boldsymbol{r}_i, \boldsymbol{r}_j)$. It is interesting to note that, for an optimal choice of apodizing vectors, i.e., $\boldsymbol{\beta}_T^*(\boldsymbol{r}_i) = \boldsymbol{a}^\intercal(\boldsymbol{r}_i)$ and $\boldsymbol{\beta}_R^*(\boldsymbol{r}_i) = \boldsymbol{b}^\intercal(\boldsymbol{r}_i)$, each element of $\boldsymbol{H_R}$ and $\boldsymbol{H_T}$ in (4.42) corresponds to the

beampattern value between the grid direction and the apodizing direction, that is $[\boldsymbol{H_T}]_{i,j} = \Upsilon_T(\boldsymbol{r}_i, \bar{\boldsymbol{r}}_j)$ and $[\boldsymbol{H_R}]_{i,j} = \Upsilon_R(\boldsymbol{r}_i, \bar{\boldsymbol{r}}_j)$.

We remark that, although we assumed that the grid $\{\boldsymbol{r}_k | 0 \leq k \leq G\}$ is a superset of the target grid, this assumption can be relaxed, at the expense of a gridding error due to any target not belonging to the grid.

Another insteresting fact is that the an optimal selection of steerings can be seen as a spatial matched filter. That is, rearranging (4.41), and considering an optimal choice of $\boldsymbol{\beta}_T(\boldsymbol{r}_s)$ and $\boldsymbol{\beta}_R(\boldsymbol{r}_s)$, it is easy to see that the best beamforming is given by the Hermitian transpose of the model matrix columns:

$$\begin{aligned} y_{r_s}(t) &= p(t - 2\tau_r) \sum_{k=0}^{K-1} \bar{x}_k \boldsymbol{\beta}_R^{\mathsf{T}}(\boldsymbol{r}_s) \boldsymbol{b}(\bar{\boldsymbol{r}}_k) \boldsymbol{\beta}_T^{\mathsf{T}}(\boldsymbol{r}_s) \boldsymbol{a}(\bar{\boldsymbol{r}}_k) \\ &= p(t - 2\tau_r) \sum_{k=0}^{K-1} \bar{x}_k \left[\boldsymbol{\beta}_T(\boldsymbol{r}_s) \otimes \boldsymbol{\beta}_R(\boldsymbol{r}_s)\right]^{\mathsf{T}} \left[\boldsymbol{a}(\bar{\boldsymbol{r}}_k) \otimes \boldsymbol{b}(\bar{\boldsymbol{r}}_k)\right] \\ &= p(t - 2\tau_r) \sum_{k=0}^{K-1} \bar{x}_k \Upsilon_C(\boldsymbol{r}_s, \bar{\boldsymbol{r}}_k) \end{aligned} \tag{4.45}$$

where $\Upsilon_C(\boldsymbol{r}_s, \bar{\boldsymbol{r}}_k)$ is the combined beampattern of the receiving and transmitting arrays, known as the system's point spread function, which maximizes the received signal energy whenever $\boldsymbol{r}_s$ matches a target direction $\bar{\boldsymbol{r}}_k$. Still, there is no guarantee that a peak in $y_{r_s}(t)$ will correspond to a specific target, since their relative positions can contribute constructively to the received signal energy. That is, unless each individual target beampattern has its peak located at possible nulls of the beampattern corresponding to the steered direction $\boldsymbol{r_s}$, any sidelobe can add up to a higher peak in the overall point spread function.

4.3.2 Array geometry considerations

All of the above approximations make no consideration on the array geometry. Still, some choices of the sensor positions can considerably simplify the radar design. First, note that it is useful to represent the direction vector $\boldsymbol{d}_k$ in (4.21) in spherical coordinates:

$$\boldsymbol{d}_k = \frac{\omega_0}{c_0} \begin{bmatrix} \sin\theta_k \cos\phi_k \\ \cos\theta_k \cos\phi_k \\ \sin\phi_k \end{bmatrix} = \frac{2\pi}{\lambda_0} \begin{bmatrix} \sin\theta_k \cos\phi_k \\ \cos\theta_k \cos\phi_k \\ \sin\phi_k \end{bmatrix}. \tag{4.46}$$

This representation allows us to formulate the most common geometry seen in the literature, namely, the Uniform Linear Array (ULA), composed by a 1D arrangement of, say, M_T equally spaced elements. For the sake of simplicity, one usually aligns the elements coordinates with the z-axis, so that all sensors are expressed from a

single reference point q_o:

$$\boldsymbol{q}_i = \begin{bmatrix} 0 \\ 0 \\ id + q_o \end{bmatrix} \tag{4.47}$$

where d is a fixed spacing between the array elements.

Using (4.46) and (4.47), and defining $\psi_k \triangleq \frac{2\pi}{\lambda_0}\sin(\phi_k)$, the manifold vectors as in (4.21) read:

$$\boldsymbol{a}(\boldsymbol{r}_k) = e^{j\psi_k q_o} \begin{bmatrix} 1 \\ e^{j\psi_k d} \\ e^{j\psi_k 2d} \\ \vdots \\ e^{j\psi_k (M_T-1)d} \end{bmatrix} \tag{4.48}$$

The structure of $\boldsymbol{a}(\boldsymbol{r}_k)$ suggests that the manifold matrix in (4.24) can be written as the product of a Fourier and a diagonal matrix:

$$\boldsymbol{\mathcal{A}}(\boldsymbol{r}) = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ e^{j\psi_0 d} & e^{j\psi_1 d} & \cdots & e^{j\psi_{(G-1)} d} \\ \vdots & \vdots & \vdots & \vdots \\ e^{j\psi_0 (M_T-1)d} & e^{j\psi_1 (M_T-1)d} & \cdots & e^{j\psi_{(G-1)} (M_T-1)d} \end{bmatrix} \begin{bmatrix} e^{jq_o\psi_0} & & & \\ & e^{jq_o\psi_1} & & \\ & & \ddots & \\ & & & e^{jq_o\psi_{(G-1)}} \end{bmatrix}. \tag{4.49}$$

We can use this fact to induce a desired structure in $\boldsymbol{\mathcal{A}}(\boldsymbol{r})$. For instance, by choosing probing directions in a way that $\sin\phi_k$ is uniformly spaced, the manifold matrix becomes essentially a $M_T \times G$ partial Discrete Fourier Transform (DFT) matrix. That is, let $\psi_k = k\delta + \psi_0$. Then (4.49) can be written as

$$\boldsymbol{\mathcal{A}}(\boldsymbol{r}) = e^{jq_o\psi_0} \begin{bmatrix} 1 & & & \\ & e^{jd\psi_0} & & \\ & & \ddots & \\ & & & e^{j(M-1)d\psi_0} \end{bmatrix} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & e^{j\delta d} & \cdots & e^{j(G-1)\delta d} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & e^{j\delta(M-1)d} & \cdots & e^{j\delta(G-1)(M-1)d} \end{bmatrix} \cdot \begin{bmatrix} 1 & & & \\ & e^{jq_o\delta} & & \\ & & \ddots & \\ & & & e^{j(G-1)q_o\delta} \end{bmatrix} \tag{4.50}$$

Using (4.50) the beampattern function for the ULA is given by

$$\begin{aligned}\Upsilon(\phi_i,\phi_j) = \boldsymbol{a}^*(\boldsymbol{r}_i)\boldsymbol{a}(\boldsymbol{r}_j) &= e^{-j\psi_{ij}q_o}\sum_{k=0}^{M-1} e^{j\psi_{ij}kd} \\ &= e^{-j\psi_{ij}\left(q_o+\frac{(M-1)d}{2}\right)}\frac{\sin\left(\psi_{ij}\frac{Md}{2}\right)}{\sin\left(\psi_{ij}\frac{d}{2}\right)}\end{aligned} \tag{4.51}$$

where $\psi_{ij} = \frac{2\pi}{\lambda_0}(\sin\phi_j - \sin\phi_i)$. The beampattern for a ULA becomes a real quantity whenever $q_o = -\frac{(M-1)d}{2}$, which can be achieved by setting the reference point at the center of the array. Figure 4.2 shows an example of a beampattern for a 5 element array.

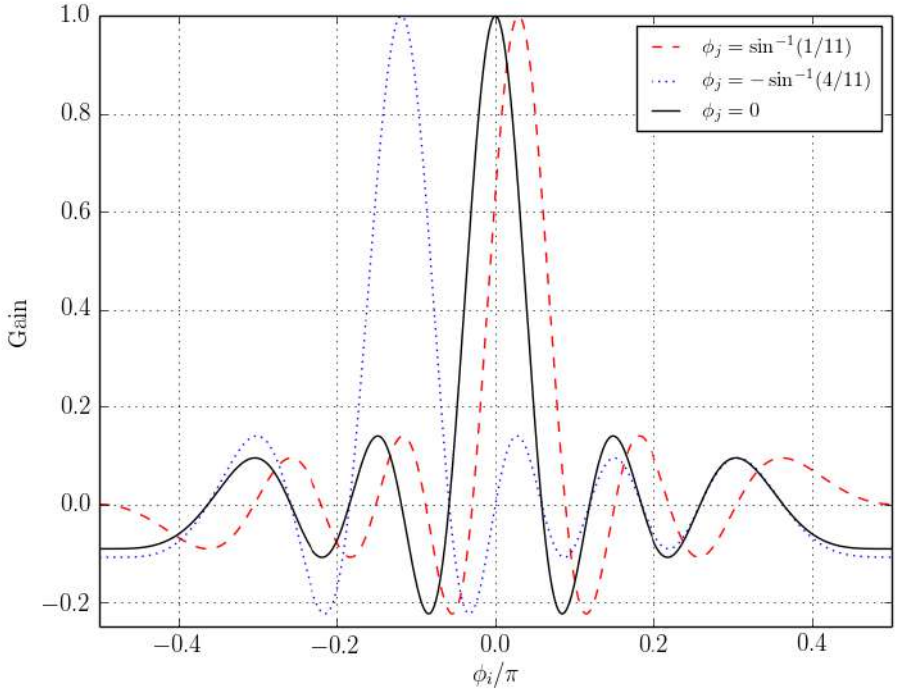


Figure 4.2: Beampattern of an array with 11 elements separated by $d = \lambda_0/2$

One common arrangement used in radar systems is the so-called *virtual ULA* [70]. In this arrangement, the transmitting and receiving array are designed in a way that the combined manifold vector $\boldsymbol{a}(\boldsymbol{r}_k) \otimes \boldsymbol{b}(\boldsymbol{r}_k)$ keeps its components in a geometric progression, behaving as a larger array. Usually, this is achieved by separating the transmitting elements by $d_T = M_R d_R$ apart. Alternatively one can multiply the spacing between receivers by the number of transmitters $d_R = M_T d_T$. If the center of the transmiting and receiving arrays are, respectively, at q_o and $\tilde{q}_o$, the resulting manifold vector is equivalent to a single array with $M_T M_R$ sensors centered at $z_o = q_o + \tilde{q}_o$. Referring to our model in (4.32), the virtual ULA is one of the few arrangements where $\operatorname{rank}(\boldsymbol{\mathcal{F}}_{\boldsymbol{r}}) = M_T M_R$ [70].

Figure 4.3 shows the combined beampattern for a virtual ULA, where both arrays are centered at the origin, $d = \lambda_0/2$, $M_T = 5$ and $M_R = 11$. In this figure we can see that the choice $d_T = M_R d_R$ aligns the transmiter's beampattern peaks with the receiver's beampattern zeros, except for the peak at the origin.

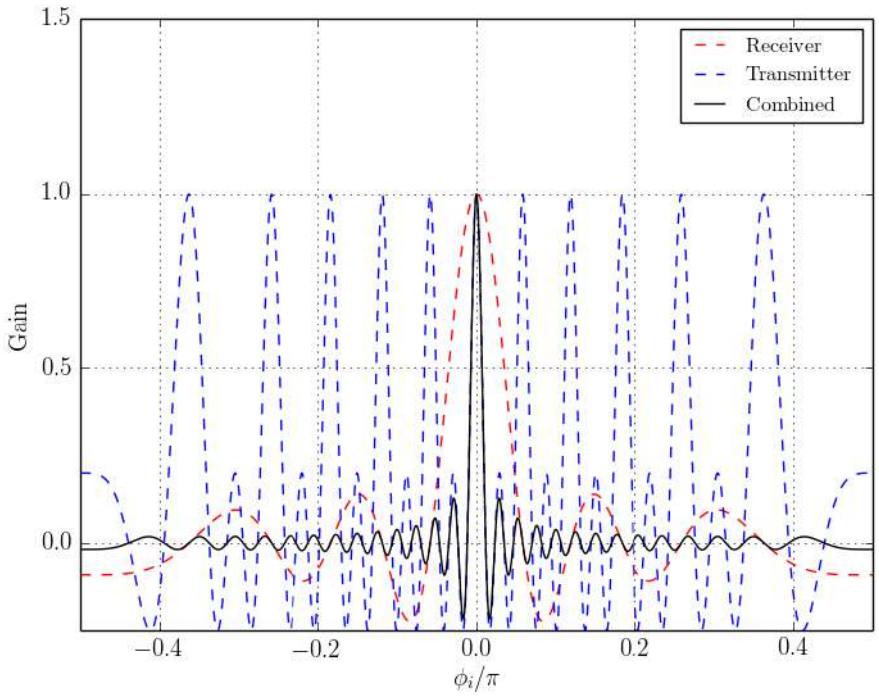


Figure 4.3: Combined beampattern of a virtual array with $M_T = 5$, $M_R = 11$, $d_T = {}^{11\lambda_0}/_2$ and $d_R = {}^{\lambda_0}/_2$

4.3.3 Multistatic and MIMO radars

The phased array system has the disadvantage of being unable to probe multiple directions simultaneously, as the transmitter has to be steered to a specific direction. Hence, loosely speaking, if a target is positioned at a null of its beampattern, it cannot be recovered. To overcome this limitation, systems that take advantage of the spatial diversity of the transmitter have been vastly studied.

One of these systems is the multistatic radar, where the transmitters take turns in sending probing pulses. We can model this operation by using a block-diagonal

pulse matrix in (4.32) and (4.35), such as

$$\boldsymbol{\mathcal{P}} = \begin{bmatrix} \boldsymbol{p}_0 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{p}_1 & \mathbf{0} & \cdots & \vdots \\ \vdots & \mathbf{0} & \boldsymbol{p}_2 & \cdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \cdots & \boldsymbol{p}_{M_T-1} \end{bmatrix},$$

where $\boldsymbol{p}_i = [p_i(N-1) \dots p_i(0)]^\intercal$, is the pulse transmitted by the i-th antenna. Note that each pulse is delayed with respect to the previous one. Again, since we have multiple pulses being transmitted, the reflectivity of the targets must vary slowly during an entire probing.

The MIMO radar operates in a similar fashion, but instead of transmitting multiple pulses sequentially, all transmitters send different pulses at once [69]. In [69–71, 80], these probing pulses are designed as orthogonal codes, and the received signal $\boldsymbol{y_r}(t)$ is passed through a bank of matched filters, sampled at $2\tau_r$. Defining the combined manifold vector as $\boldsymbol{c}_i \triangleq \boldsymbol{a}(\boldsymbol{r}_i) \otimes \boldsymbol{b}(\boldsymbol{r}_i)$, by virtue of orthonormality of the codes, the signal model after the matched filtering and sampling becomes:

$$\bar{\boldsymbol{y}}_{\boldsymbol{r}}(2\tau_r) = \begin{bmatrix} \boldsymbol{c}_0 & \boldsymbol{c}_1 & \cdots & \boldsymbol{c}_{K-1} \end{bmatrix} \boldsymbol{x_r} \triangleq \boldsymbol{\mathscr{C}} \boldsymbol{x_r}. \tag{4.52}$$

Reference [70] shows, based on results from [81], that for such configuration a radar can detect

$$K \in \left(\frac{M_T + M_R - 2}{2}, \frac{M_T M_R + 1}{2} \right] \tag{4.53}$$

targets, given that the columns of $\boldsymbol{\mathscr{C}}$ are linearly independent.

Simultaneous range and cross-range detection in MIMO radars

As we have mentioned, range and cross-range detections are usually treated separately. The main reason behind this is that range detection demands a high sampling rate (at least in electromagnetic applications), which is normally hindered by comercial Analog-to-Digital Converters (ADC). For instance, at the speed of the light, sub-millimeter detection would require a sampling rate of the order of 6ps, corresponding to approximately 167 GHz. Nevertheless, assuming that we are able to sample as fast as necessary, we can now propose a model that treats range and cross-range simultaneously.

The model we began with in Sec. 4.2 considers all targets as belonging to the same range. For multiple ranges, $\boldsymbol{x_r}$ becomes a function of the range delay τ_r, so

that the received signal is more generally replaced by a convolution integral:

$$\boldsymbol{y}_r(t) \;=\; \int \sum_{k=0}^{K-1} x_k(\tau_r)\boldsymbol{b}(\boldsymbol{r}_k)\,[\boldsymbol{p}^{\mathsf{T}}(t-2\tau_r)\boldsymbol{a}(\boldsymbol{r}_k)]\mathrm{d}\tau_r \;+\; \boldsymbol{v}(t) \tag{4.54}$$

Hereafter, we shall refer to x_k as the target reflectances, as we will only deal with the radar grid. We shall also add a signal vector $\boldsymbol{v}(t)$ to account for any noise as an interference in the system.

If we discretize this model at a sampling rate t_s, such that multiple ranges can be written as $\tau_{r_n} = \tau_{r_0} - nt_s/2$, and considering that all targets are confined to Q range bins, we can stack the signal vectors received from multiple ranges into a vector as

$$\begin{aligned}
\boldsymbol{y} &= \begin{bmatrix} \boldsymbol{y_r}(2\tau_{r_0}+(N-1)t_s) \\ \boldsymbol{y_r}(2\tau_{r_0}+(N-2)t_s) \\ \vdots \\ \boldsymbol{y_r}\left(2\tau_{r_0}\right) \end{bmatrix} = \\
&= \underbrace{\begin{bmatrix} \boldsymbol{\mathcal{F}_r}(N-1) & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \vdots & \boldsymbol{\mathcal{F}_r}(N-1) & \ddots & \vdots \\ \vdots & \ddots & \ddots & \boldsymbol{0} \\ \boldsymbol{\mathcal{F}_r}(N-Q) & \ddots & \ddots & \boldsymbol{\mathcal{F}_r}(N-1) \\ \vdots & \ddots & \ddots & \vdots \\ \boldsymbol{\mathcal{F}_r}(0) & \ddots & \ddots & \boldsymbol{\mathcal{F}_r}(Q-1) \\ \boldsymbol{0} & \boldsymbol{\mathcal{F}_r}(0) & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{\mathcal{F}_r}(0) \end{bmatrix}}_{\boldsymbol{\mathcal{F}}} \underbrace{\begin{bmatrix} \boldsymbol{x}_{Q-1} \\ \boldsymbol{x}_{Q-2} \\ \vdots \\ \boldsymbol{x}_0 \end{bmatrix}}_{\boldsymbol{x}} + \underbrace{\begin{bmatrix} \boldsymbol{v}(2\tau_{r_0}+(N-1)t_s) \\ \boldsymbol{v}(2\tau_{r_0}+(N-2)t_s) \\ \vdots \\ \boldsymbol{v}\left(2\tau_{r_0}\right) \end{bmatrix}}_{\boldsymbol{v}}
\end{aligned} \tag{4.55}$$

where the vector $\boldsymbol{x}$ is composed by stacking the reflectance vectors $\boldsymbol{x}_r$ for all ranges, and $\boldsymbol{v}$ is defined accordingly as above.

While it is tempting to find a least-squares solution for this model, this system turns out to be very ill-conditioned, as it is rank-deficient by construction. To see this, note that we can write $\boldsymbol{\mathcal{F}}$ in (4.55) in terms of block columns as

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{\mathcal{F}}_0 & \boldsymbol{\mathcal{F}}_1 & \cdots & \boldsymbol{\mathcal{F}}_{Q-1} \end{bmatrix} \boldsymbol{x} + \boldsymbol{v}. \tag{4.56}$$

Let the augmented pulse matrix be defined as

$$\boldsymbol{\mathcal{P}}_\delta = \begin{bmatrix} \mathbf{0}_{\delta \times M_T} \\ \boldsymbol{\mathcal{P}} \\ \mathbf{0}_{(Q-1)-\delta \times M_T} \end{bmatrix}. \tag{4.57}$$

Then, each $\boldsymbol{\mathcal{F}}_\delta$ is a block-column defined by

$$\boldsymbol{\mathcal{F}}_\delta \triangleq \Big[[\boldsymbol{\mathcal{P}}_\delta \boldsymbol{a}(\boldsymbol{r}_0)] \otimes \boldsymbol{b}(\boldsymbol{r}_0) \quad \cdots \quad [\boldsymbol{\mathcal{P}}_\delta \boldsymbol{a}(\boldsymbol{r}_{G-1})] \otimes \boldsymbol{b}(\boldsymbol{r}_{G-1}) \Big] \tag{4.58}$$

$$= (\boldsymbol{\mathcal{P}}_\delta \otimes \boldsymbol{I}_{M_R}) \Big[\boldsymbol{a}(\boldsymbol{r}_0) \otimes \boldsymbol{b}(\boldsymbol{r}_0) \quad \cdots \quad \boldsymbol{a}(\boldsymbol{r}_{G-1}) \otimes \boldsymbol{b}(\boldsymbol{r}_{G-1}) \Big] \tag{4.59}$$

Resorting to the same arguments that led to the conclusions of rank-deficiency in the single range model, one can verify that $\text{rank}(\boldsymbol{\mathcal{F}}_\delta) \leq M_T M_R$. Thus, we can expect $\text{rank}(\boldsymbol{\mathcal{F}}) \leq Q M_T M_R$. Since we are now probing a 3-dimensional grid with QG points, and generally $QG > Q M_T M_R$, $\boldsymbol{\mathcal{F}}$ is rank deficient and therefore, ill-posed for LS estimation.

The limitation implied by (4.53) with regard to target detection, as well as the ill-conditioning imposed by the MIMO radar model in (4.55), call for new solutions and algorithmic forms which allow one to surpass the resolution limits and computational complexity involved in the existing formulations. In the next section, we approach these issues in light of the theory of compressed sensing, and present preliminary results targeting robustness and improved imaging performance in the case of MIMO transmissions.

4.4 Coherence conditions for $\boldsymbol{\mathcal{F}}$

The model presented in the previous sections assumes that targets are populated in a dense grid. In many realistic scenarios, however, most of the grid will be empty, so that the vector $\boldsymbol{x}$ will be truly sparse. In this scenario, the theory of Compressed Sensing (CS) enlightens possible ways to recover the targets information, overcoming the ill-conditioning of $\boldsymbol{\mathcal{F}}$ imposed implicitly by (spatial) Nyquist limits.

The use of compressed sensing for MIMO radar imaging is proposed, e.g., in [71], and further exploited in [22], where it is shown to improve cross-range resolution in MIMO radars with randomly positioned antennas. While in [22] the range information is addressed by a bank of matched filters, CS imaging over a full 3D model has been proposed in [20], considering that each transmitting antenna sends a long Gaussian sequence as probing pulse.

In this section, we aim to derive sufficient recovery conditions for radar imaging for the full 3D model presented in (4.56), which results in optimal choices of

transmitted pulses and grid spacing.

As discussed in Sec. 1.1, accurate recovery of $\boldsymbol{x}$ depends heavily on the coherence of the columns of $\boldsymbol{\mathcal{F}}$

$$\mu(\boldsymbol{\mathcal{F}}) = \max_{i \neq j} \frac{|[\boldsymbol{\mathcal{F}}]_i^* [\boldsymbol{\mathcal{F}}]_j|}{\|[\boldsymbol{\mathcal{F}}]_i\| \|[\boldsymbol{\mathcal{F}}]_j\|} \tag{4.60}$$

In order to improve the resolution and recovery ability of this system, our goal is to minimize the coherence measure in (4.60). Both numerator and denominator can be retrieved from the Gram matrix $\boldsymbol{\mathcal{F}}^*\boldsymbol{\mathcal{F}}$: the energy of the columns appear in its diagonal, while their inner products correspond to the off-diagonal elements. Our goal is to use the fact that $\boldsymbol{\mathcal{F}}$ is highly structured in order to derive conditions on the transmitted pulse, such that $\mu(\boldsymbol{\mathcal{F}})$ is minimized.

Thus, by forming the Gram matrix from the block columns of $\boldsymbol{\mathcal{F}}$

$$\boldsymbol{\mathcal{F}}^*\boldsymbol{\mathcal{F}} = \begin{bmatrix} \boldsymbol{\mathcal{F}}_0^*\boldsymbol{\mathcal{F}}_0 & \boldsymbol{\mathcal{F}}_0^*\boldsymbol{\mathcal{F}}_1 & \cdots & \boldsymbol{\mathcal{F}}_0^*\boldsymbol{\mathcal{F}}_{Q-1} \\ \boldsymbol{\mathcal{F}}_1^*\boldsymbol{\mathcal{F}}_0 & \boldsymbol{\mathcal{F}}_1^*\boldsymbol{\mathcal{F}}_1 & \cdots & \boldsymbol{\mathcal{F}}_1^*\boldsymbol{\mathcal{F}}_{Q-1} \\ \vdots & \vdots & & \vdots \\ \boldsymbol{\mathcal{F}}_{Q-1}^*\boldsymbol{\mathcal{F}}_0 & \boldsymbol{\mathcal{F}}_{Q-1}^*\boldsymbol{\mathcal{F}}_{Q-1} & \cdots & \boldsymbol{\mathcal{F}}_{Q-1}^*\boldsymbol{\mathcal{F}}_{Q-1} \end{bmatrix}, \tag{4.61}$$

each element within its block-elements can be written as

$$\begin{aligned} [\boldsymbol{\mathcal{F}}_l^*\boldsymbol{\mathcal{F}}_m]_{i,j} &= [[\boldsymbol{\mathcal{P}}_l \boldsymbol{a}(\boldsymbol{r}_i)] \otimes \boldsymbol{b}(\boldsymbol{r}_i)]^* [[\boldsymbol{\mathcal{P}}_m \boldsymbol{a}(\boldsymbol{r}_j)] \otimes \boldsymbol{b}(\boldsymbol{r}_j)] \\ &= [[\boldsymbol{a}^*(\boldsymbol{r}_i)\boldsymbol{\mathcal{P}}_l^*] \otimes \boldsymbol{b}^*(\boldsymbol{r}_i)] [[\boldsymbol{\mathcal{P}}_m \boldsymbol{a}(\boldsymbol{r}_j)] \otimes \boldsymbol{b}(\boldsymbol{r}_j)] \\ &= [\boldsymbol{a}^*(\boldsymbol{r}_i)\boldsymbol{\mathcal{P}}_l^*\boldsymbol{\mathcal{P}}_m \boldsymbol{a}(\boldsymbol{r}_j)] \otimes [\boldsymbol{b}^*(\boldsymbol{r}_i)\boldsymbol{b}(\boldsymbol{r}_j)] \\ &= [\boldsymbol{a}^*(\boldsymbol{r}_i)\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(l-m)\boldsymbol{a}(\boldsymbol{r}_j)] [\boldsymbol{b}^*(\boldsymbol{r}_i)\boldsymbol{b}(\boldsymbol{r}_j)] \\ &\triangleq [\boldsymbol{\mathcal{G}}(l-m)]_{i,j} \end{aligned} \tag{4.62}$$

where $\boldsymbol{\mathcal{P}}_\delta$ is the augmented pulse matrix defined in (4.57), also reproduced here for convenience:

$$\boldsymbol{\mathcal{P}}_\delta \;=\; \begin{bmatrix} \mathbf{0}_{\delta \times M_T} \\ \boldsymbol{\mathcal{P}} \\ \mathbf{0}_{(Q-1)-\delta \times M_T} \end{bmatrix}, \tag{4.63}$$

and $\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(l-m)$ denotes the pulse vector autocorrelation, which can be expressed as

$$\begin{aligned} \boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(l-m) \;&=\; \boldsymbol{\mathcal{P}}_l^*\boldsymbol{\mathcal{P}}_m = \\ &= \begin{bmatrix} r_{00}(l-m) & r_{01}(l-m) & \cdots & r_{0(M_T-1)}(l-m) \\ r_{10}(l-m) & r_{11}(l-m) & \cdots & r_{1(M_T-1)}(l-m) \\ \vdots & \vdots & \cdots & \vdots \\ r_{(M_T-1)0}(l-m) & r_{(M_T-1)1}(l-m) & \cdots & r_{(M_T-1)(M_T-1)}(l-m) \end{bmatrix}. \end{aligned} \tag{4.64}$$

The elements $r_{ij}(k)$ denote the aperiodic cross-correlations of $p_i(t)$ and $p_j(t)$ at lag $k = l - m$

$$r_{ij}(k) = \sum_{i=0}^{N+k-1} p_i^*(i) p_j(i-k) \tag{4.65}$$

Alternatively, defining the combined manifold vector as

$$\boldsymbol{c}_i \triangleq \boldsymbol{a}(\boldsymbol{r}_i) \otimes \boldsymbol{b}(\boldsymbol{r}_i)$$

with $\boldsymbol{\mathcal{Q}}(k) \triangleq \boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(l-m) \otimes \boldsymbol{I}_{M_R}$, and using (4.31), it is possible to write $[\boldsymbol{\mathcal{G}}(l-m)]_{i,j}$ in (4.62) as

$$\begin{aligned}
[\boldsymbol{\mathcal{G}}(l-m)]_{i,j} &= \{[\boldsymbol{\mathcal{P}}_l \otimes \boldsymbol{I}_{M_R}]\,[\boldsymbol{a}(\boldsymbol{r}_i) \otimes \boldsymbol{b}(\boldsymbol{r}_i)]\}^* \{[\boldsymbol{\mathcal{P}}_m \otimes \boldsymbol{I}_{M_R}]\,[\boldsymbol{a}(\boldsymbol{r}_j) \otimes \boldsymbol{b}(\boldsymbol{r}_j)]\} \\
&= \{[\boldsymbol{\mathcal{P}}_l \otimes \boldsymbol{I}_{M_R}]\,\boldsymbol{c}_i\}^* \{[\boldsymbol{\mathcal{P}}_m \otimes \boldsymbol{I}_{M_R}]\,\boldsymbol{c}_j\} \\
&= \boldsymbol{c}_i^* [\boldsymbol{\mathcal{P}}_l^* \otimes \boldsymbol{I}_{M_R}]\,[\boldsymbol{\mathcal{P}}_m \otimes \boldsymbol{I}_{M_R}]\,\boldsymbol{c}_j \\
&= \boldsymbol{c}_i^* [\boldsymbol{\mathcal{P}}_l^* \boldsymbol{\mathcal{P}}_m] \otimes [\boldsymbol{I}_{M_R} \boldsymbol{I}_{M_R}]\,\boldsymbol{c}_j \\
&= \boldsymbol{c}_i^* [\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(l-m) \otimes \boldsymbol{I}_{M_R}]\,\boldsymbol{c}_j \\
&= \boldsymbol{c}_i^* \boldsymbol{\mathcal{Q}}(l-m) \boldsymbol{c}_j
\end{aligned} \tag{4.66}$$

With the explicit elements of the Gram matrix as in (4.66), the coherence of $\boldsymbol{\mathcal{F}}$ can be rewritten as

$$\mu(\boldsymbol{\mathcal{F}}) = \max_{i \neq j \vee k \neq 0} \frac{\boldsymbol{c}_i^* \boldsymbol{\mathcal{Q}}(k) \boldsymbol{c}_j}{\left[\boldsymbol{c}_i^* \boldsymbol{\mathcal{Q}}(0) \boldsymbol{c}_i\right]^{1/2} \left[\boldsymbol{c}_j^* \boldsymbol{\mathcal{Q}}(0) \boldsymbol{c}_j\right]^{1/2}} \,. \tag{4.67}$$

Observe that in minimizing the above quotient, we have freedom to select both the MIMO correlation function and the manifold directions. Because exact (weighted) orthogonality of the manifold vectors can only be attained at one particular lag k, we can adopt the following procedure.

For the lag $k = 0$, we pick a set of directions such that $\boldsymbol{c}_i$ and $\boldsymbol{c}_j$ annihilates (4.67) for a given $i \neq j$. For $k \neq 0$, we would like to design $\boldsymbol{\mathcal{Q}}(k)$ as close to the null matrix as possible. That is, for $k = 0$ the numerator in (4.67) is given by (4.62):

$$[\boldsymbol{\mathcal{G}}(0)]_{i,j} = [\boldsymbol{a}^*(\boldsymbol{r}_i) \boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(0) \boldsymbol{a}(\boldsymbol{r}_j)]\,[\boldsymbol{b}^*(\boldsymbol{r}_i) \boldsymbol{b}(\boldsymbol{r}_j)] \tag{4.68}$$

which represents the product between the receiver beampattern, i.e., $\Upsilon_{RX}(\boldsymbol{r}_i, \boldsymbol{r}_j) \triangleq \boldsymbol{b}^*(\boldsymbol{r}_i) \boldsymbol{b}(\boldsymbol{r}_j)$, and the weighted beampattern of the transmitter, defined as $\Upsilon_{TX;W}(\boldsymbol{r}_i, \boldsymbol{r}_j) \triangleq \boldsymbol{a}^*(\boldsymbol{r}_i) \boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(0) \boldsymbol{a}(\boldsymbol{r}_j)$. The fact is that for some array geometries, it is possible to choose directions where either beampatterns are zero; the number of selected directions defines the dimension G of the grid. The simplest choice is to enforce $\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(0) = N\boldsymbol{I}$, which ensures that all the pulses have the same power and

are orthogonal at the zero lag. As a result, this choice also simplifies the design of the transmitting array.

One of such geometries that offers a simple selection of directions is the virtual ULA, presented in Sec. 4.3.2. As stated in that section, the combined manifold vectors $\boldsymbol{c}_i$ for a virtual ULA can be viewed as an equivalent manifold vector for an array having $M \triangleq M_T M_R$ entries. With the appropriate parameters in (4.51), the combined patterns will generate $M_T M_R - 1$ possible angles ϕ_j for each selected angle ϕ_i. For such arrangement, we can pick a grid of $G = M_T M_R$ angles, namely, $\phi_i = \arcsin\left({}^1\!/_{2\pi}\left(\sin(\phi_0) + i\delta\right)\right)$, where $\delta \triangleq ({}^2\!/_{M_R M_T})$, and $i \in \mathbb{Z}$, $i \in [0, G)$.

Now, introduce the Cholesky factorization $\boldsymbol{\mathcal{Q}}(0) = \boldsymbol{L}\boldsymbol{L}^*$, and define $\bar{\boldsymbol{c}}_i \triangleq \boldsymbol{L}^* \boldsymbol{c}_i$. For a correlation lag $k \neq 0$, the ratio in (4.67) can be written as

$$\mu(\boldsymbol{\mathcal{F}}) = \max_{i \neq j \vee k \neq 0} \frac{\bar{\boldsymbol{c}}_i^* \boldsymbol{L}^{-1} \boldsymbol{\mathcal{Q}}(k) \boldsymbol{L}^{-*} \bar{\boldsymbol{c}}_j}{\|\bar{\boldsymbol{c}}_i\| \|\bar{\boldsymbol{c}}_j\|} \tag{4.69}$$

which assumes the form of the well known *Rayleigh quotient* (see, e.g., 9.8.36 in [79]), however, one for every $\boldsymbol{\mathcal{Q}}(k)$. A simple upper bound for (4.69) is

$$\mu(\boldsymbol{\mathcal{F}}) \leq \lambda_{\max}(\boldsymbol{L}^{-1} \boldsymbol{\mathcal{Q}}(k) \boldsymbol{L}^{-*}) \tag{4.70}$$

in terms of the maximum eigenvalue $\lambda_{\max}(\cdot)$. Since we have that

$$\begin{aligned} \lambda_{\max}(\boldsymbol{L}^{-1} \boldsymbol{\mathcal{G}}(k) \boldsymbol{L}^{-*}) \leq\ & \|\boldsymbol{L}^{-1} \boldsymbol{\mathcal{Q}}(k) \boldsymbol{L}^{-*}\|_F \\ =\ & \|\boldsymbol{\mathcal{Q}}^{-1}(0) \boldsymbol{\mathcal{Q}}(k)\|_F \\ =\ & M_R^{1/2} \|\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}^{-1}(0) \boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(k)\|, \end{aligned} \tag{4.71}$$

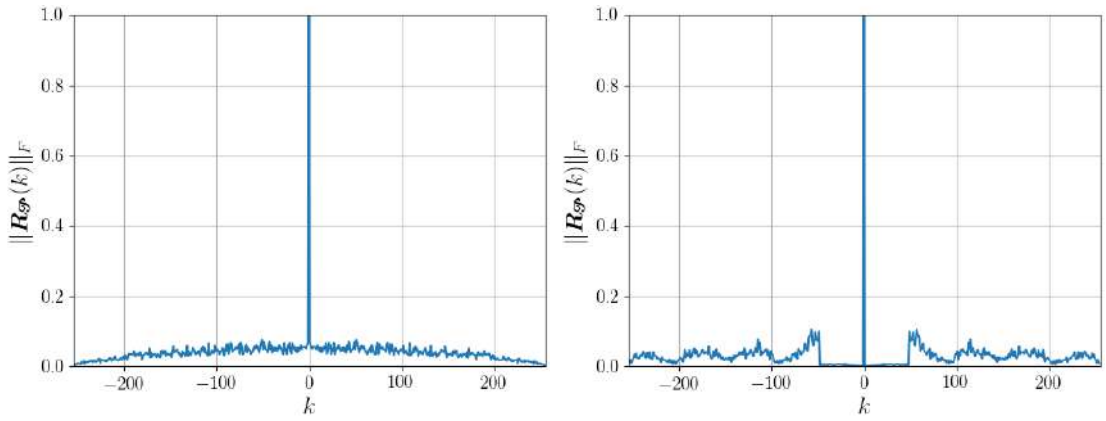
and $\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(0)$ is finite, this requires ideally, $\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(k)$ as a null matrix for all $k \in [1, Q)$.

Independent Gaussian sequence sets, as considered in [82], allows us to approximate the above requirements, i.e., $\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(0) = N\boldsymbol{I}$ and $\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(k) = 0$, in a stochastic sense. However, these may demand a high level of synchronization. As an alternative, we make use of the so-called *complementary sequence sets*, which can be generated by optimizing the following block LS criterion (see Eq. (11) in [83]):

$$\min_{\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(k)} \ \|\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(0) - N\boldsymbol{I}_{M_T}\|_F^2 + 2\sum_{k=1}^{Q-1} \|\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(k)\|_F^2. \tag{4.72}$$

One advantage of working with complementary sequences is that we can produce zero correlations in a range of only $Q - 1$ samples, yielding lower cross-correlation within the same range, when compared to its Gaussian sequences counterpart. Figure 4.4 shows the Frobenius norm of the cross-correlation matrix in different lags, for a Gaussian sequence set (a) and a complementary sequence set generated using the WeCAN [83] algorithm (b), both with $N = 256$ samples in $M_T = 5$ sequences.

The complementary sequence set exhibits a very low norm when $|k| < 48$, unlike the Gaussian sequence set.



(a) Gaussian sequences (b) Complementary sequences

Figure 4.4: Comparison of $\|\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(k)\|_F$ in different lags, for two different sequence sets.

Note that, by adding constraints to (4.72), it is also possible to restrict the pulse samples to specific modulations, such as QAM or BPSK, or to constrain the peak-to-average ratio (PAR) of the sequences (see, e.g., [84]).

Two-step procedure

Note that substituting $\boldsymbol{c}_i \triangleq \boldsymbol{a}(\boldsymbol{r}_i) \otimes \boldsymbol{b}(\boldsymbol{r}_i)$ in (4.59), each block-column of $\boldsymbol{\mathcal{F}}$ can be written as:

$$\begin{aligned}\boldsymbol{\mathcal{F}}_\delta &= (\boldsymbol{\mathcal{P}}_\delta \otimes \boldsymbol{I}_{M_R}) \begin{bmatrix} \boldsymbol{a}(\boldsymbol{r}_0) \otimes \boldsymbol{b}(\boldsymbol{r}_0) & \cdots & \boldsymbol{a}(\boldsymbol{r}_{G-1}) \otimes \boldsymbol{b}(\boldsymbol{r}_{G-1}) \end{bmatrix} \\ &= (\boldsymbol{\mathcal{P}}_\delta \otimes \boldsymbol{I}_{M_R}) \begin{bmatrix} \boldsymbol{c}_0 & \cdots & \boldsymbol{c}_{G-1} \end{bmatrix} \\ &\triangleq (\boldsymbol{\mathcal{P}}_\delta \otimes \boldsymbol{I}_{M_R}) \boldsymbol{\mathscr{C}}. \end{aligned} \tag{4.73}$$

Thus, (4.56) can be written as

$$\begin{aligned} y &= \begin{bmatrix} \boldsymbol{\mathcal{F}}_0 & \boldsymbol{\mathcal{F}}_1 & \cdots & \boldsymbol{\mathcal{F}}_Q \end{bmatrix} \boldsymbol{x} + \boldsymbol{v} \\ &= \begin{bmatrix} (\boldsymbol{\mathcal{P}}_0 \otimes \boldsymbol{I}_{M_R}) \boldsymbol{\mathscr{C}} & (\boldsymbol{\mathcal{P}}_1 \otimes \boldsymbol{I}_{M_R}) \boldsymbol{\mathscr{C}} & \cdots & (\boldsymbol{\mathcal{P}}_{Q-1} \otimes \boldsymbol{I}_{M_R}) \boldsymbol{\mathscr{C}} \end{bmatrix} \boldsymbol{x} + \boldsymbol{v} \\ &= \begin{bmatrix} \boldsymbol{\mathcal{P}}_0 \otimes \boldsymbol{I}_{M_R} & \boldsymbol{\mathcal{P}}_1 \otimes \boldsymbol{I}_{M_R} & \cdots & \boldsymbol{\mathcal{P}}_{Q-1} \otimes \boldsymbol{I}_{M_R} \end{bmatrix} \begin{bmatrix} \boldsymbol{\mathscr{C}} & & & \\ & \boldsymbol{\mathscr{C}} & & \\ & & \ddots & \\ & & & \boldsymbol{\mathscr{C}} \end{bmatrix} \boldsymbol{x} + \boldsymbol{v} \\ &= \left(\begin{bmatrix} \boldsymbol{\mathcal{P}}_0 & \boldsymbol{\mathcal{P}}_1 & \cdots & \boldsymbol{\mathcal{P}}_{Q-1} \end{bmatrix} \otimes \boldsymbol{I}_{M_R} \right) (\boldsymbol{I}_Q \otimes \boldsymbol{\mathscr{C}}) \boldsymbol{x} + \boldsymbol{v}. \end{aligned} \tag{4.74}$$

Then, introducing $\boldsymbol{T} \triangleq \begin{bmatrix} \boldsymbol{\mathcal{P}}_0 & \boldsymbol{\mathcal{P}}_1 & \cdots & \boldsymbol{\mathcal{P}}_{Q-1} \end{bmatrix} \otimes \boldsymbol{I}_{M_R}$ and $\boldsymbol{\mathcal{C}} \triangleq \boldsymbol{I}_Q \otimes \boldsymbol{\mathscr{C}}$, (4.74) can be written as

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{T}\boldsymbol{\mathcal{C}}\boldsymbol{x} + \boldsymbol{v} \\ &= \boldsymbol{T}\,\mathrm{vec}(\boldsymbol{\mathscr{C}}\boldsymbol{X}) + \boldsymbol{v}, \end{aligned} \tag{4.75}$$

where $\boldsymbol{X}$ is the reshaped vector $\boldsymbol{x}$ into a size $G \times Q$ matrix. Note that $\boldsymbol{x}$ shows two levels of sparsity: it is a block-sparse vector, with each block, in turn, sparse itself, as shown in (4.76). That is, each column of $\boldsymbol{X}$ is a sparse vector:

$$\boldsymbol{x} = \begin{bmatrix} \begin{bmatrix} \bullet \\ 0 \\ 0 \\ \vdots \\ \bullet \end{bmatrix} \\ \begin{bmatrix} \bullet \\ \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix} \\ \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \\ \vdots \\ \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ \bullet \end{bmatrix} \end{bmatrix} \qquad \Longleftrightarrow \qquad \boldsymbol{X} = \begin{bmatrix} \begin{bmatrix} \bullet \\ 0 \\ 0 \\ \vdots \\ \bullet \end{bmatrix} & \begin{bmatrix} \bullet \\ \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} & \cdots & \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ \bullet \end{bmatrix} \end{bmatrix} \tag{4.76}$$

A more exact approach should treat $\boldsymbol{x}$ as a sparse tensor, and, in that case, block sparsity along all directions could be exploited. However, for such problems, one would have to resort to multi-linear algebra, or to a *Matrix Completion* approach.

Now, we know from (4.49), that in the case of a virtual ULA arrangement, $\boldsymbol{\mathscr{C}}$ can be written as $\boldsymbol{\mathscr{C}} \triangleq \boldsymbol{\mathscr{C}}_V \boldsymbol{\mathscr{D}}$, where $\boldsymbol{\mathscr{D}} = \mathrm{diag}\left\{ e^{jz_0\psi_0} \quad e^{jz_0\psi_1} \quad \cdots \quad e^{jz_0\psi_{(G-1)}} \right\}$, and $\boldsymbol{\mathscr{C}}_V$ is an $M \times G$ matrix of discrete Fourier bases defined by the node vector $\begin{bmatrix} e^{jd\psi_0} & e^{jd\psi_1} & \cdots & e^{jd\psi_{G-1}} \end{bmatrix}$, with $\psi_i = \sin(\phi_i)$, and ϕ_i a probing angle, as previously defined.

Now, from (4.75), let

$$\boldsymbol{z} \triangleq \mathrm{vec}(\boldsymbol{\mathscr{C}}\boldsymbol{X}) \tag{4.77}$$

so that

$$\boldsymbol{y} = \boldsymbol{T}\,\boldsymbol{z} + \boldsymbol{v}\,. \tag{4.78}$$

Note that vector $\boldsymbol{z}$ maintains the block-sparsity property of $\boldsymbol{x}$.

It is known that $M\times G$ Fourier matrices composed by distinct bases are full spark, with equal norm columns, given that $M \leq G$ (see [37]). Moreover, it is also known that if $\boldsymbol{\mathscr{C}}$ is full spark, this condition allows us to recover the columns of $\boldsymbol{X}$ that are up to $G/2$-sparse, since we can recover the block-sparse $\boldsymbol{z}$ from $\boldsymbol{y}$. Fourier matrices with equally spaced bases on the unit circle can be shown to satisfy the RIP, and exhibit the smallest worst-case coherence when $G \geq 2M$ [37]. We can induce this structure, using the same probing directions derived previously for optimizing the coherence of $\boldsymbol{\mathcal{F}}$. That way, $\boldsymbol{\mathscr{C}} = e^{jz_0\psi_0}\boldsymbol{\mathscr{D}}_1\boldsymbol{\mathscr{C}}_F\boldsymbol{\mathscr{D}}_2$, where

$$\boldsymbol{\mathscr{D}}_1 = \begin{bmatrix} 1 & & & \\ & e^{jd\psi_0} & & \\ & & \ddots & \\ & & & e^{j(M-1)d\psi_0} \end{bmatrix}, \quad \boldsymbol{\mathscr{D}}_2 = \begin{bmatrix} 1 & & & \\ & e^{jz_0\delta} & & \\ & & \ddots & \\ & & & e^{jz_0(G-1)\delta} \end{bmatrix}, \tag{4.79}$$

and where $\boldsymbol{\mathscr{C}}_F$ is an $M\times G$ partial DFT matrix with $e^{jd\delta}$ as basis.

In most practical scenarios, when $\boldsymbol{\mathscr{C}}$ is essentially a partial DFT matrix, $\boldsymbol{z}$ will be quasi-sparse: if a column of the target image $\boldsymbol{X}$ contains a wide target, its corresponding column in $\boldsymbol{Z} \triangleq \boldsymbol{\mathscr{C}}\boldsymbol{X}$ will exhibit localized elements (Fourier bandwidth trade-off). Thus, the only situation where we would have a dense matrix $\boldsymbol{Z}$ is when $\boldsymbol{X}$ represents a very narrow (cross-range) and long (range) target.

Henceforth, assuming that $\boldsymbol{z}$ is sparse, it can be fully recovered using compressed sensing techniques from $\boldsymbol{y}$ by designing the columnns of $\boldsymbol{T}$ to have low mutual coherence. Now, to compute the coherence, one can take advantage of the structure in $\boldsymbol{T}$, whose Gram matrix is block-Toeplitz. In this case, each such block is given by

$$\boldsymbol{\mathcal{G}}_{\boldsymbol{T}}(l-m) = [\boldsymbol{\mathcal{P}}_l \otimes \boldsymbol{I}_{M_R}]^* [\boldsymbol{\mathcal{P}}_m \otimes \boldsymbol{I}_{M_R}] \tag{4.80}$$

$$= \boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(l-m) \otimes \boldsymbol{I}_{M_R}. \tag{4.81}$$

Just like we argued in the case of a one step CS procedure, assuming that all pulses have the same power, the diagonal elements of $\boldsymbol{\mathcal{G}}_{\boldsymbol{T}}(0)$ become constant, and we can optimize $\mu(\boldsymbol{T})$ by minimizing all off-diagonal elements of the Gram matrix. This can be achieved by approximating $\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(0) = N\boldsymbol{I}_{M_T}$ and $\boldsymbol{R}_{\boldsymbol{\mathcal{P}}}(k) = \boldsymbol{0}, 0<k<Q$, which is the same approximation used for optimizing the coherence of the full model matrix $\boldsymbol{\mathcal{F}}$.

The advantage of recovering range and cross-range sequentially as in (4.77)-(4.78)

is the freedom in controlling the regularization parameters in the CS problems for each direction individually, allowing for different levels of sparsity in $\boldsymbol{z}$ and $\boldsymbol{x}$.

Efficient implementation

Exact ℓ_1 regularized convex optimization turns out to be inefficient in our case, since for each grid point to be recovered, a significant number of constraints is added, resulting in $3QG$ variables. Furthermore, e.g., interior point solvers do not take full advantage of the problem structure, as they often require the measurement matrix to be formed explicitly, and demand a QR decomposition of the KKT system at each step [44].

On the other hand, as we have seen in Sec. 1.1.1, iterative algorithms have their complexity dictated by products with the measurement matrix, which can be implemented very efficiently through (4.77) and (4.78). Here, we are assuming that the complexity of a matrix-vector product is roughly the same for both the direct and transposed matrix operations. The superior performance seen in [23] with regards to the approximate message passing algorithm when the noise variance and sparsity of input are unknown, motivates us to adopt the CAMP algorithm as a basis for comparison with standard techniques, since this is a very likely condition to be found in a radar imaging scenario.

Unlike the direct product $\boldsymbol{\mathcal{C}x}$, which in general requires GM_TM_RQ operations, the block diagonal products require only $Q[G\log_2 G+G+M_TM_R]$ operations if $\boldsymbol{\mathscr{C}}$ is a Fourier matrix. Recognizing the block-Toeplitz like structure in $\boldsymbol{S}$ we can reduce the operations count for $\boldsymbol{Tz}$ from $M_R^2M_TNQ$ to $(N+Q-1)M_T[M_TM_R+(M_R+2)\log_2(N+Q-1)]$, by using FFT based algorithms.

For the sake of comparison, we implement two iterative procedures, using message passing algorithms. In the first one we use the CAMP to recover $\boldsymbol{x}$ directly from $\boldsymbol{y}$. In this version, each algorithm iteration requires the computation of $\boldsymbol{T\mathcal{C}x}$ and $\boldsymbol{\mathcal{C}^*T^*y}$. In the second version, we considered the two-step procedure aforementioned, first recovering $\boldsymbol{z}$ from $\boldsymbol{y}$ (which makes use of products like $\boldsymbol{Tz}$ and $\boldsymbol{T^*y}$). Having recovered $\boldsymbol{z}$, we reshape it and use the CAMP for each column of $\boldsymbol{Z}$, say, $[\boldsymbol{Z}]_i$, so as to retrieve the columns of $\boldsymbol{X}$. In this step, the products used are mainly $\boldsymbol{\mathscr{C}}[\boldsymbol{X}]_i$ and $\boldsymbol{\mathscr{C}}^*[\boldsymbol{Z}]_i$.

Note that the two algorithm versions will have the same worst case complexity, if we consider that the CAMP will always require the same maximum number of iterations to converge. In practice, the second version is marginally faster, as the CAMP converges in a smaller number of iterations in the second step, where the problems are smaller.

4.5 Simulations

We simulate a MIMO radar setup using the exact model from (4.56), considering $M_T = 5$ transmitters and $M_R = 11$ receivers, configured as a Virtual ULA. A complementary sequence comprising 40 pulses is used in each transmission, with a low correlation interval of 8 pulses, generated using the WeCAN algorithm [83]. The grid was set with $G = 55$ probing directions and $Q = 7$ ranges. Since in this case we deal with a fixed sensing matrix $\boldsymbol{\mathcal{F}}$, we could only vary the sparsity of the vectors used as targets. For each possible sparsity ratio, an ensemble of 30 random vectors was generated, with the non-zero entries drawn from a circular complex Gaussian distribution. For this simulation, we have set $\kappa = 1.2$ for CAMP, and $\alpha = 50$ for the CS-BDFE. To assess the performance in this scenario, the AFE, FDR and FRR figures are displayed with the horizontal axis corresponding to the target sparsity, while in the vertical axis we have the corresponding performance score. A 15 dB SNR noise was added to all transmissions in the system. This plot can be understood as a "slice" of the phase transition diagrams presented in Sec. 3.2.2, parallel to the vertical axis.

Figure 4.5(a) illustrates the AFE performance. For comparison purposes, the AFE of a MMSE estimator is also included as an initialization step for the CS-BDFE algorithm. Note that the CS-BDFE improves its initial estimate, except for very dense vectors, with above 70% of non-zero entries. On the other hand, the CAMP algorithm performs worse than the CS-BDFE, as it is impaired for vectors exhibiting over 50% non-zero entries. The FRR for this scenario is shown in Fig. 4.5(b). The FRR is almost the same for sparser vectors, but the CS-BDFE is superior when it comprises more than 50% non-zero entries.

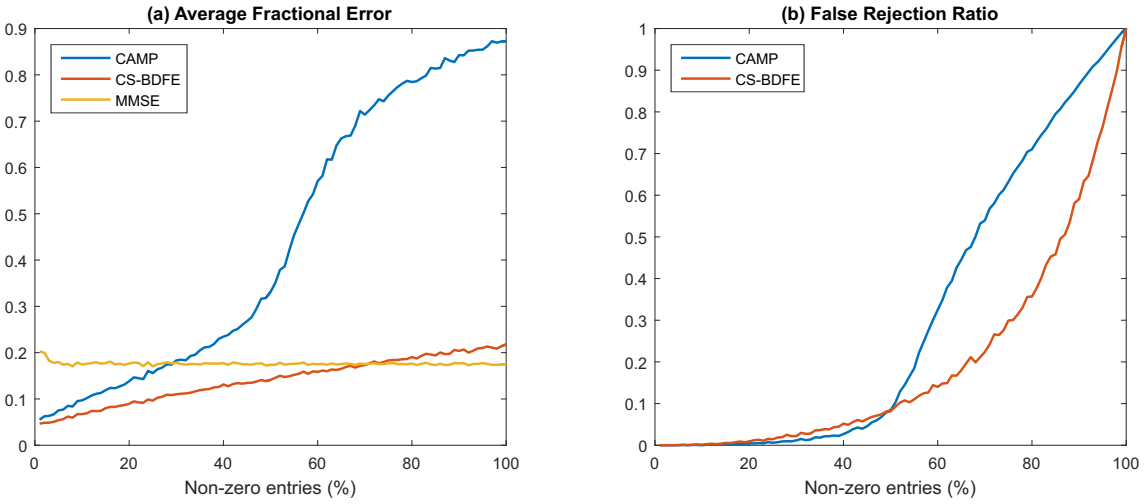


Figure 4.5: (a) AFE and (b) FRR for CAMP and CS-BDFE algorithms for a MIMO radar setup under 15 dB SNR.

Figure 4.6 details the FRR for less than 50% non-zero entries. Although CAMP presents a better performance than CS-BDFE, at this scale the difference corresponds to CS-BDFE rejecting only one more entry than CAMP.

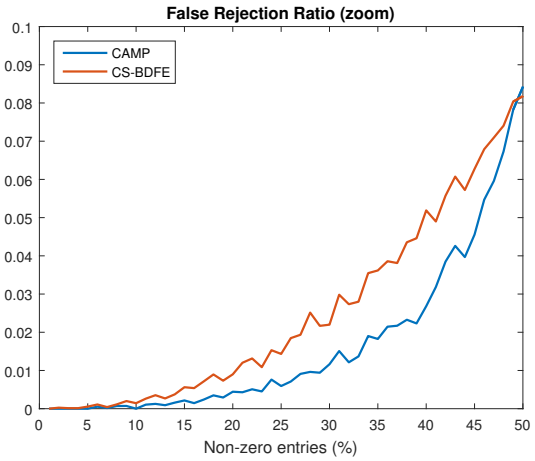


Figure 4.6: Detail of the FRR plot for CAMP and CS-BDFE algorithms for a MIMO radar setup under 15 dB SNR.

From Fig. 4.7(a) we verify that the CS-BDFE algorithm outperforms the CAMP with respect to the FDR, even though CS-BDFE requires prior knowledge of the SNR, which is not the case for the CAMP recursions. On the other hand, CAMP still requires the user to experimentally select the parameter κ in (1.33).

A mismatch between the system SNR and the parameters σ_x^2 and σ_v^2 set in the CS-BDFE will affect the performance, especially for the FDR measure, as shown in Fig. 4.7(b). This is verified by setting the $\sigma_v^2 = 10^{-2}\sigma_x^2$, which would be appropriate for a 20 dB SNR, when the actual SNR is at 15 dB. We verify that both AFE and FRR did not change significantly, but for a larger mismatch, those performance measures degrade as well. We remark that in this experiment, CAMP required an average of 15 iterations to converge, while CS-BDFE required around 4 iterations. These results indicate that, although the CS-BDFE adds extra complexity per iteration compared to other iterative algorithms, it still represents an alternative choice for systems where the FDR is the most significant figure of merit. In the case of MIMO radars, where the sensing matrix is nearly square, and the corresponding target vectors may not be sufficiently sparse, the CS-BDFE presents itself as a viable alternative.

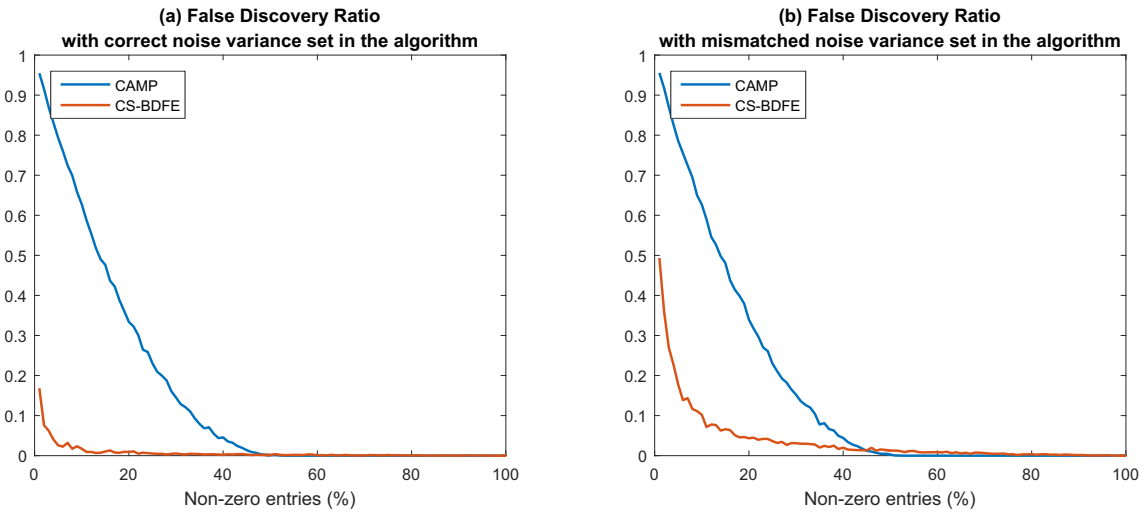


Figure 4.7: FDR for CAMP and CS-BDFE algorithms for a MIMO radar setup. In (a) the noise level is 15 dB SNR, and the algorithm is set up with $\sigma_v^2 = 10^{-1.5}\sigma_x^2$. In (b) the noise level is kept at 15 dB SNR, while the algorithm run with $\sigma_v^2 = 10^{-2}\sigma_x^2$ which is pertinent to a 20 dB SNR.

In a second test, we consider a base image and use it to simulate a real-world scenario. The setup makes use of 7 transmitters and 10 receivers, with transmitting sequences comprising 256 pulses. The SNR is adjusted to 10 dB and the targets are limited to 30 range bins. Figure 4.8 shows the exact target image used in our simulations.

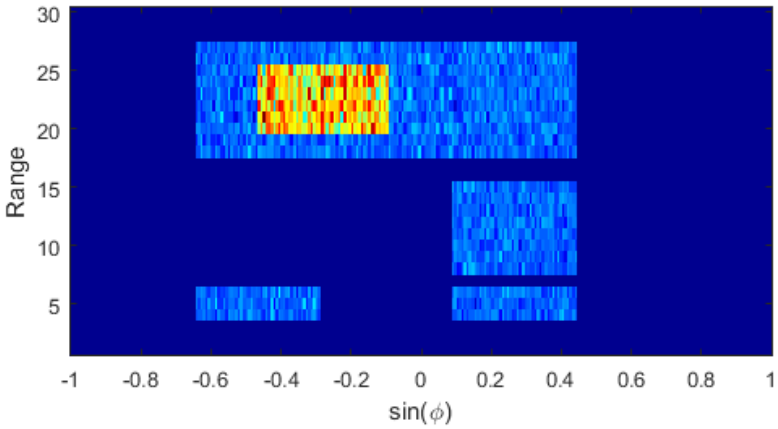


Figure 4.8: Exact target image for radar testing.

The virtual ULA used in this experiment allows us to construct a grid composed of 70 directions, which is less dense than the true target grid. We circumvented this issue by dividing the target grid into four interleaved direction grids of the same number, say, 70. The results for a single step recovery, that is, recovery based on $\boldsymbol{\mathcal{F}}$ matrix as sensing matrix to retrieve $\boldsymbol{x}$ is shown in Figs. 4.9 and 4.10, for the CAMP and CS-BDFE algorithms, respectively.

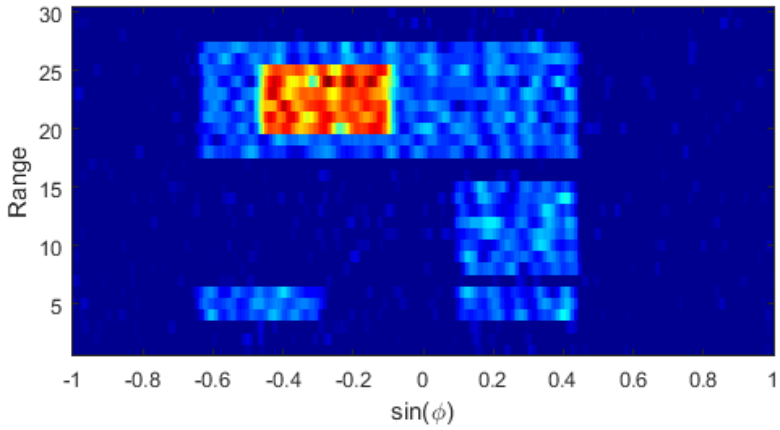


Figure 4.9: Image recovered using CAMP in a single step, with 10dB SNR.

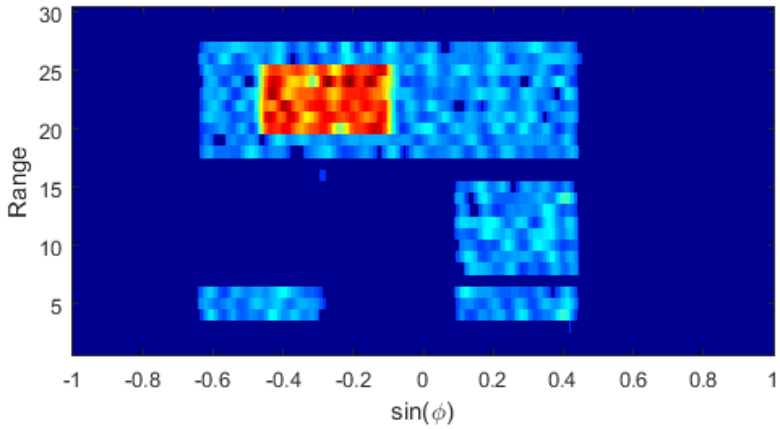


Figure 4.10: Image recovered using GDFE in a single step, with 10dB SNR.

In Fig. 4.11, the CS-BDFE algorithm was used, but the σ_v^2 parameter was set incorrectly to $10^{-1.5}$, which would be adequate for a 15 db SNR.

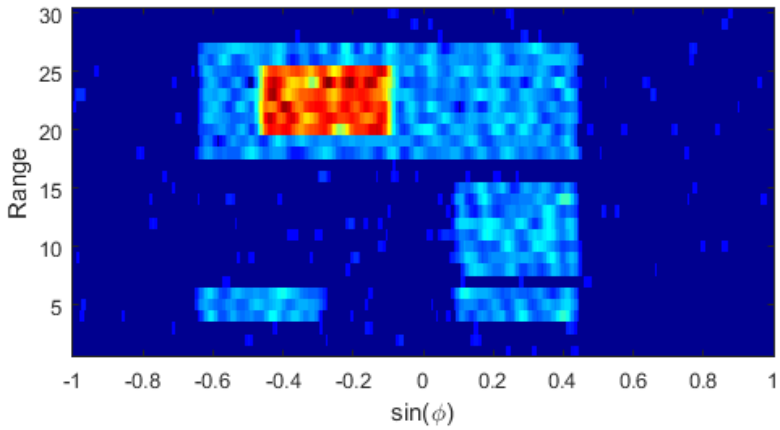


Figure 4.11: Image recovered using GDFE in a single step, with mismatched σ_v^2 set in the algorithm.

The resulting performance measurements for this experiment are shown in Table 4.1. Note that, although more complex, the CS-BDFE algorithm requires a smaller number of iterations.

Table 4.1: Single step recovery performance

	AFE	FDR	FRR	Iterations
CAMP	0.19	35.9%	0.1%	23
CS-BDFE	0.23	0.2%	1.2%	7
CS-BDFE (mismatch SNR)	0.23	9.7%	0.2%	10

Figures 4.12 and 4.13 show the resulting image for a two-step procedure using CAMP and CS-BDFE. That is, instead of using $\boldsymbol{\mathcal{F}}$ as a sensing matrix to recover $\boldsymbol{x}$ directly from $\boldsymbol{y}$, we follow (4.77) and (4.78) and construct a pulse matrix $\boldsymbol{T}$ to recover a sparse vector $\boldsymbol{z}$ from $\boldsymbol{y}$, which is then reshaped into a matrix $\boldsymbol{Z}$. The columns of $\boldsymbol{X}$ are then recovered from $\boldsymbol{Z}$ using $\boldsymbol{\mathcal{C}}$ as sensing matrix. The resulting measures are presented in Table 4.2. The same variance values were used in both steps, even though in the second step the noise level might not correspond to the actual SNR due to the detection in the first step. Also note that, in the second step, the CS-BDFE algorithm requires a very small number of iterations, when compared to the CAMP.

Table 4.2: Single step recovery performance

	AFE	FDR	FRR	Iterations (1^{st} step)	Avg. Iterations (2^{nd} step)
CAMP	0.23	29%	1.1%	17	12
CS-BDFE	0.28	0.1%	1.7%	7	3

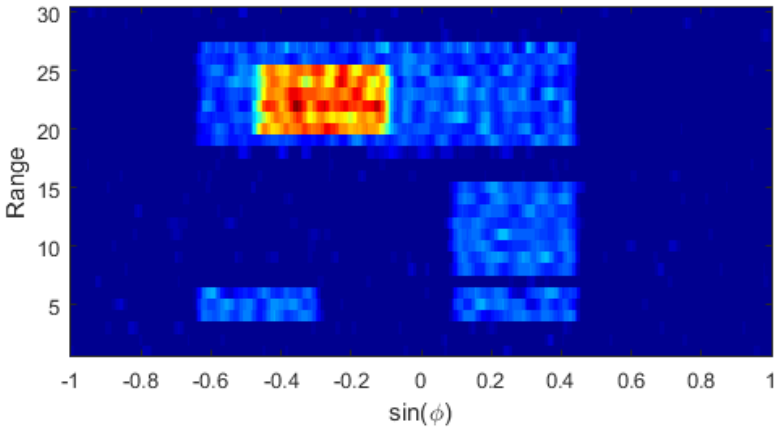


Figure 4.12: Image recovered using CAMP in a two-step procedure, with 10db SNR

The last experiment is an FDTD[34] simulated scene containing two small solid boxes with different uniform dielectric constants and a small electrical conductivity,

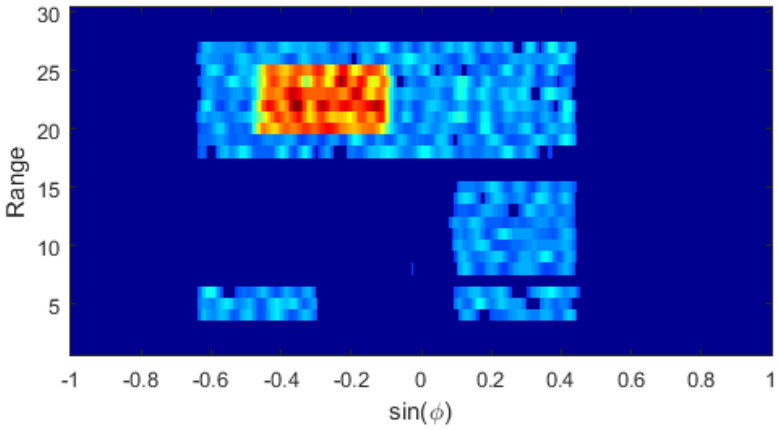


Figure 4.13: Image recovered using GDFE in a two-step procedure, with 10db SNR

and is intended to verify the algorithm behavior under some more realistic conditions such as non-ideal antenna patterns and multiple scattered signals.

For this simulation, we make use of the OpenEMS package [85, 86], with $M_T = 5$ transmitting elements and $M_R = 11$ receiving antennas in a virtual ULA setup, using the complementary sequence set from Experiment 3, which modulates a $f_0 = 8\,\text{GHz}$ sinusoidal signal. The simulated antennas are elementary dipoles, and their positions are quantized to the simulation grid cell size $\lambda_0/20$. The simulated environment is a square sheet of 600 mm each side, three cells thick, surrounded by 8 layers of cells configured as *perfectly matched layers* (PML)[34] in order to simulate an open environment. The target boxes are square with 60 mm sides, relative permeability of $\varepsilon_1 = 3$ and $\varepsilon_2 = 5$, and both have conductivity of $0.1\,\text{S/m}$.

As the OpenEMS system is not able to simulate multiple sources, and FDTD simulations tends to oscillate in the natural frequencies of the model grid, we have to use the linearity of Maxwell's equations to circumvent those limitations. Each pulse transmission was simulated individually in two cycles, one with the targets and the other in free environment. The environment response is subtracted from the target scattering data and the resulting signal is summed into the overall response. OpenEMS automatically adapts its time steps during the simulation, based on the carrier frequency and the grid density, hence the output signals are sampled each 3.6 ps. Finaly, a post-processing script in MATLAB demodulates the 8 GHz carrier and recovers the complex envelope for the imaging algorithms.

Figures 4.14 and 4.15 show the result of the two different recovery algorithms using the one-step procedure. Since electromagnetic waves are reflected solely at medium interfaces, only the borders within the critical angle become visible, as it would happen to a glass cube. The higher reflectances correspond to the box corners, and the apparent angular distortion seen is an effect of the non-cartesian mapping of the axes (the expected recovered shape of the boxes is shown in the insertion at the

bottom of Figs. 4.14 and 4.15). In the CAMP recovery, between the boxes it appears the effect of re-scattered fields, not taken into account by the Born approximation used in first place. Note that the CS-BDFE algorithm correctly rejected this field as interference.

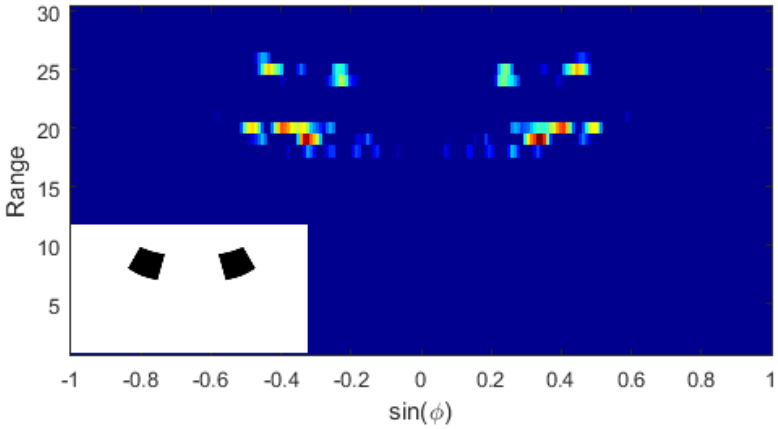


Figure 4.14: FDTD image recovered using CAMP in one step.

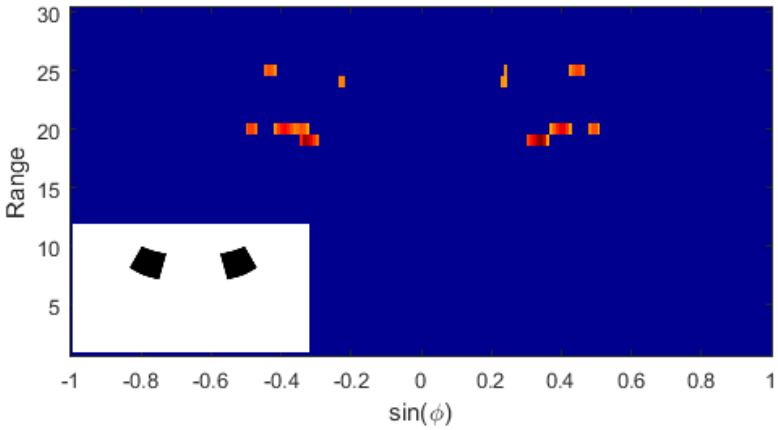


Figure 4.15: FDTD image recovered using CS-BDFE in one step.

Chapter 5

Future Work Proposals

In this chapter we overview the research topics considered as candidates for continuity of this work, which can lead to new algorithms and techniques that can be exploited by radar imaging, as well as in equalization of digital communication channels.

5.1 Compressed sensing for downsampled received signals

As argued in Sec. 4.3.3, simultaneous range and cross-range detection demands a high sample-rate, at least in electromagnetic applications. On the other hand, reducing the sampling rate at the receiver hinders the radar's ability of discriminating details in the range dimension. Compressed sensing techniques allow us to operate in a lower sample rate, as long as the recovery conditions (NSP, RIP or low-coherence) presented in Sec. 1.1 can be achieved.

As a motivation, consider the single antenna model presented in Fig. 5.1

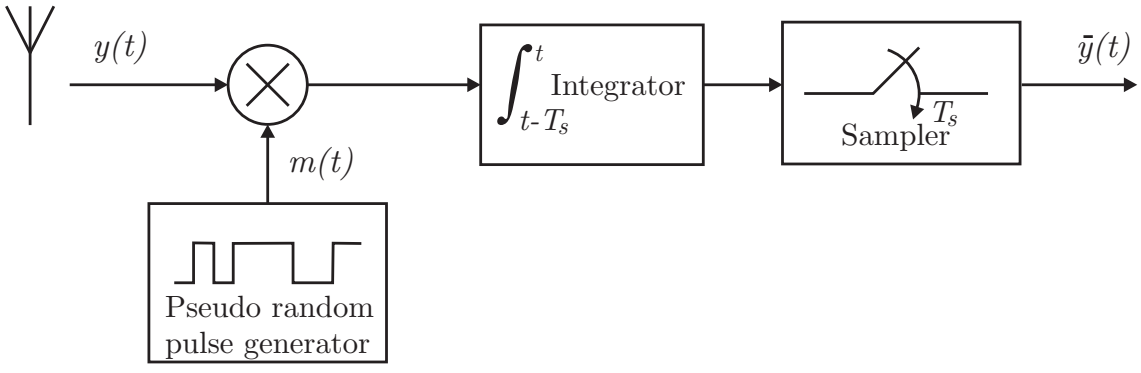


Figure 5.1: Single receiver downsampler

Suppose that the pseudo-random pulses $m(t)$, as well as the received signal $y(t)$ are generated at a T_s/R rate. Then, the received signal $\bar{y}(t)$ can be modeled as

$$\bar{y}(t) = \begin{bmatrix} m(t) & m(t-\frac{T_s}{R}) & \cdots & m(t-\frac{(R-1)T_s}{R})] \end{bmatrix} \begin{bmatrix} y(t) \\ y(t-\frac{T_s}{R}) \\ \cdots \\ y(t-\frac{(R-1)T_s}{R})] \end{bmatrix}. \tag{5.1}$$

If $m(t)$ is constrained, e.g., to $\{-1, +1\}$, this downsampler can be easily implemented in the analog domain.

This can be extended to the MIMO model (4.55), by considering a downsampling matrix $\boldsymbol{M}$, whose rows contain pseudo-random sequences and allows us not only to downsample the received vector, but to combine its different signals with each other. Thus, we can write

$$\begin{aligned} \bar{\boldsymbol{y}} &= \boldsymbol{M}\boldsymbol{y} \\ &= \boldsymbol{M}\boldsymbol{\mathcal{F}}\boldsymbol{x} \\ &= \boldsymbol{M}\boldsymbol{T}\boldsymbol{\mathcal{C}}\boldsymbol{x}. \end{aligned} \tag{5.2}$$

Under the assumption that $\boldsymbol{\mathcal{C}}\boldsymbol{x}$ is sparse, our problem now is to investigate how to optimize the pulse set, such that the coherence $\mu(\boldsymbol{M}\boldsymbol{T})$ is minimized for a fixed matrix $\boldsymbol{M}$ corresponding to a hardware implementation of the downsampler. Following a similar reasoning, the pulse set designed for this downsampler would also be appropriate for CS in the single step procedure.

5.2 CS-BDFE applied to the estimation of constellation signals

As noted in Sec. 3.2, a procedure that detects one symbol at a time may offer a safer detection mechanism for an underdetermined, or even ill-conditioned problem, over a batch processing such as Kalman recursions. In this case, the full vector is re-estimated at once per iteration.

Resorting again to the deterministic-stochastic duality of regularized LS and MMSE problems, it might be possible to extend the CS-BDFE algorithm to a communications setting, when $\mathbf{x}$ belongs on a constellation, and devise a procedure to solve the regularized LS problem in (2.122). This can be approached by using the estimate $\hat{\mathbf{x}}_i$ and its corresponding uncertainty $\widehat{\boldsymbol{P}}_i$ obtained from a stochastic formulation. The threshold strategy of 2.10.2 could be adjusted in order to adapt the uncertainties in this version of the CS-BDFE.

Such implementation, although more computationally demanding, could outperform the CS-Kalman in ill-conditioned scenarios, such as transmissions with no IBI cancellation or employing full ZJ.

Chapter 6

Final Considerations

In this work, we unified and took advantage of the interplays between the CS and the BI-DFE formulations in order to obtain new algorithms both for equalization and compressed sensing purposes. Although these theories have evolved rather independently, they both share a common LS formulation with a suitable regularization function, so that we were able to derive algorithms based on sparsity constraints for both compressed sensing and block equalization problems.

In Chapter 2, we showed that recursions derived from a LS formulation present themself as a BI-DFE structure, which is not a result of some pre-imposed structure, but an algorithm on its own right. We extended the strictly linear BI-DFE problems to Reduced Complexity Widely Linear formulations, by rewriting the underlying linear models as ones based on real and imaginary vector extensions. We discussed the optimality of the decision-delay in the BI-DFE considering the seminal paper of Scaglione et al. [15, 16], applied to a successive cancellation approach, which led to the well known V-BLAST algorithm. A significant contribution arising from our development is that by employing sequential detection, the feedforward matrices can be computed from fast transversal filter recursions. When the channel length is smaller than the block size, the model transmission model comprises 2 block coefficients. This allows the IBI to be removed by using redundancy, either via zero-padding or zero-jamming, or by a combination of both. In case the IBI is completely removed, a linear MMSE estimator can be used, and implemented by superfast structures. As DFE receivers demand an initial (linear) MMSE estimate, superfast receivers become of great importance. We clarified the claims in [28–32] about minimum and reduced redundancy systems and verified in our experiments, that MR schemes offer no advantage over standard schemes in neither MC or SC configurations, regardless of its use under coding or via discrete Hartley transform (DHT) implementations. Simulations on a simple *Extended Pedestrian A* (EPA) model of the *Long Term Evolution* (LTE) standard, verify that redundancy cannot be reduced towards its optimal value in a ZF or MMSE equalization scenario,

which motivates this work on the search of sparsity-based solutions.

A new CS-based DFE algorithm has been derived, based on the LS cost used similarly for iterative thresholding methods, however, one with a sparsity regularizer enforced with respect to the most recent Kalman filtered estimate. This will target constellation based signals, of great interest in digital transmissions. The resulting recursions can be seen as a generalization of the notion of IBI removal, prior to any re-estimation procedure. The algorithm makes use of a likelihood test to determine the thresholds used to assess whether an entry was correctly estimated, which turned out very effective. It was tested within different contexts, either with or without redundancy, and proved itself very efficient in terms of BER. When deployed without IBI removal, the CS-BDFE can retrieve more samples from the input vector than the length of the output window, that is, the received signal could be sampled at a lower rate than what is predicted by the Nyquist theory. It also outperformed other DFE receivers in the minimum redundancy setting, even when the optimal redundancy could not be achieved by such scheme.

In Chapter 3, we have relied on the same LS form for the cost function used by the iterative thresholding algorithms, in which we enforced sparsity regularization on the target vector itself, arriving at Kalman-based recursions for CS. This accelerates convergence, as well as minimizes excess MSE. To this end, we reformulated the CS problem from a communications perspective, by showing how a CS algorithm naturally arises as a solution to well known equalization schemes, vastly exploited in signal processing community. The resulting CS-BDFE algorithm uses a procedure that detects one symbol at a time, since re-estimating a full vector at once per iteration, as done in the Kalman recursions, might impair its ability to detect the correct support for underdetermined or ill-conditioned problems. A procedure that detects one symbol at a time is a safer mechanism, despite requiring more computationally demanding implementations. We employed a DFE structure in which the feedforward and feedback matrices were optimized. When compared against CAMP, the CS-BDFE showed improved performance when the sensing matrix has a block-Toeplitz structure.

In Chapter 4, we developed a full joint range/cross-range convolution model for MIMO radars and obtained conditions in which CS techniques can reconstruct a volumetric image. After constructing such model, we take a step further by decoupling it into two separate sparse problems, albeit ones that exhibit more structured models, suitable for efficient implementations. We were able to relate the array geometry and transmitted pulses directly to the radar's recovery ability, assessed by the coherence of the sensing matrix. This model is then used to attest the CS-BDFE performance in a real-world radar application. The algorithm behaves favorably compared to the CAMP recursions, in terms of recovering the image

support. From an FDTD simulation of a real radar, the CS-BDFE even disregarded secondary scattering as interference.

Bibliography

[1] ELDAR, Y. C., KUTYNIOK, G. *Compressed Sensing: Theory and Applications*. New York, NY, USA, Cambridge University Press, 2012.

[2] THEODORIDIS, S. *Machine Learning: A Bayesian and Optimization Perspective*. 1st ed. Boston, MA, USA, Academic Press, 2015. ISBN: 9780128015223.

[3] THEODORIDIS, S., KOPSINIS, Y., SLAVAKIS, K. "Sparsity-Aware Learning and Compressed Sensing: An Overview", *CoRR*, v. abs/1211.5231, 2012.

[4] PATI, Y., REZAIIFAR, R., KRISHNAPRASAD, P. "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition". In: *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pp. 40–44 vol.1, Nov 1993. doi: 10.1109/ACSSC.1993.342465.

[5] DAUBECHIES, I., DEFRISE, M., DE MOL, C. "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint", *Communications on Pure and Applied Mathematics*, v. 57, n. 11, pp. 1413–1457, 2004. ISSN: 1097-0312. doi: 10.1002/cpa.20042. Disponível em: <http://dx.doi.org/10.1002/cpa.20042>.

[6] DONOHO, D. L., TSAIG, Y. "Fast Solution of ℓ_1-Norm Minimization Problems When the Solution May Be Sparse", *IEEE Transactions on Information Theory*, v. 54, n. 11, pp. 4789–4812, Nov 2008. ISSN: 0018-9448. doi: 10.1109/TIT.2008.929958.

[7] BLUMENSATH, T., DAVIES, M. "Iterative Thresholding for Sparse Approximations", *Journal of Fourier Analysis and Applications*, v. 14, n. 5-6, pp. 629–654, 2008. ISSN: 1069-5869. doi: 10.1007/s00041-008-9035-z. Disponível em: <http://dx.doi.org/10.1007/s00041-008-9035-z>.

[8] DONOHO, D. L., TSAIG, Y., DRORI, I., et al. "Sparse Solution of Underdetermined Systems of Linear Equations by Stagewise Orthogonal

Matching Pursuit", *IEEE Transactions on Information Theory*, v. 58, n. 2, pp. 1094–1121, Feb 2012. ISSN: 0018-9448. doi: 10.1109/TIT.2011.2173241.

[9] DONOHO, D. L., MALEKI, A., MONTANARI, A. "Message-passing algorithms for compressed sensing", *Proc. of the Nat. Acad. of Sci.*, v. 106, n. 45, pp. 18914–18919, 2009. doi: 10.1073/pnas.0909892106. Disponível em: <http://www.pnas.org/content/106/45/18914.abstract>.

[10] MALEKI, A., ANITORI, L., YANG, Z., et al. "Asymptotic Analysis of Complex LASSO via Complex Approximate Message Passing (CAMP)", *IEEE Trans. on Inf. Theory*, v. 59, n. 7, pp. 4290–4308, July 2013. ISSN: 0018-9448. doi: 10.1109/TIT.2013.2252232.

[11] LIANG, Y.-C., SUN, S., HO, C. K. "Block-iterative generalized decision feedback equalizers for large MIMO systems: algorithm design and asymptotic performance analysis", *IEEE Trans. on Sig. Proc.*, v. 54, n. 6, pp. 2035–2048, June 2006. ISSN: 1053-587X. doi: 10.1109/TSP.2006.873485.

[12] MERCHED, R. "A Unified Approach to Reduced-Redundancy Transceivers: Superfast Linear and Block-Iterative Generalized Decision Feedback Equalizers", *IEEE Transactions on Signal Processing*, v. 61, n. 17, pp. 4214–4229, Sept 2013. ISSN: 1053-587X. doi: 10.1109/TSP.2013.2264919.

[13] CANDÈS, E., TAO, T. "Decoding by linear programming", *Information Theory, IEEE Transactions on*, v. 51, n. 12, pp. 4203–4215, Dec 2005. ISSN: 0018-9448. doi: 10.1109/TIT.2005.858979.

[14] SAYED, A. H. *Adaptive Filters*. Hoboken, NJ, USA, John Wiley & Sons, 2008.

[15] SCAGLIONE, A., GIANNAKIS, G. B., BARBAROSSA, S. "Redundant filterbank precoders and equalizers. I. Unification and optimal designs", *IEEE Transactions on Signal Processing*, v. 47, n. 7, pp. 1988–2006, Jul 1999. ISSN: 1053-587X. doi: 10.1109/78.771047.

[16] SCAGLIONE, A., GIANNAKIS, G. B., BARBAROSSA, S. "Redundant filterbank precoders and equalizers. II. Blind channel estimation, synchronization, and direct equalization", *IEEE Transactions on Signal Processing*, v. 47, n. 7, pp. 2007–2022, Jul 1999. ISSN: 1053-587X. doi: 10.1109/78.771048.

[17] LIN, Y.-P., WENG, P.-L., PHOONG, S.-M. "Block based DMT systems with reduced redundancy". In: *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, v. 4, pp. 2357–2360 vol.4, 2001. doi: 10.1109/ICASSP.2001.940473.

[18] PINTO, R., MERCHED, R. "Compressed sensing joint range and cross-range MIMO Radar imaging". In: *2015 IEEE International Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2339–2343, April 2015. doi: 10.1109/ICASSP.2015.7178389.

[19] PINTO, R., MERCHED, R. "An efficient two-step procedure for compressed sensing 3D MIMO radar". In: *2015 23rd European Signal Processing Conf. (EUSIPCO)*, pp. 1771–1775, Aug 2015. doi: 10.1109/EUSIPCO.2015.7362688.

[20] HERMAN, M., STROHMER, T. "High-Resolution Radar via Compressed Sensing", *Signal Processing, IEEE Transactions on*, v. 57, n. 6, pp. 2275–2284, 2009. ISSN: 1053-587X. doi: 10.1109/TSP.2009.2014277.

[21] CARIN, L., LIU, D., GUO, B. "In situ compressive sensing", *Inverse Problems*, v. 24, n. 1, pp. 015023, 2008. Disponível em: <http://stacks.iop.org/0266-5611/24/i=1/a=015023>.

[22] OHLSSON, H., ELDAR, Y. C., YANG, A. Y., et al. "Compressive Shift Retrieval", *CoRR*, v. abs/1303.4996, 2013. Submitted to IEEE Transactions on Signal Processing.

[23] GÖRTZ, N. "Iterative Schemes and Approximate Message Passing for recovery in Compressed Sensing - Lecture notes for (389.178)". 2014. Disponível em: <http://www.nt.tuwien.ac.at/teaching/courses/summer-term/389178/>.

[24] STAMOULIS, A., GIANNAKIS, G. B., SCAGLIONE, A. "Block FIR decision-feedback equalizers for filterbank precoded transmissions with blind channel estimation capabilities", *IEEE Transactions on Communications*, v. 49, n. 1, pp. 69–83, Jan 2001. ISSN: 0090-6778. doi: 10.1109/26.898252.

[25] ALMEIDA NETO, F. G. *Low-Complexity Adaptive Filtering*. Tese de Doutorado, Escola Politécnica, Universidade de São Paulo, 2015. Disponível em: <http://www.teses.usp.br/teses/disponiveis/3/3142/tde-05022016-061143/>.

[26] PICINBONO, B., CHEVALIER, P. "Widely linear estimation with complex data", *IEEE Transactions on Signal Processing*, v. 43, n. 8, pp. 2030–2033, Aug 1995. ISSN: 1053-587X. doi: 10.1109/78.403373.

[27] FOSCHINI, G. J. "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas", *Bell Labs Technical Journal*, v. 1, n. 2, pp. 41–59, Autumn 1996. ISSN: 1089-7089. doi: 10.1002/bltj.2015.

[28] MARTINS, W. A., DINIZ, P. S. R. "Combating noise gains in high-throughput block transceivers using CSI at the transmitter". In: *2010 7th International Symposium on Wireless Communication Systems*, pp. 275–279, Sept 2010. doi: 10.1109/ISWCS.2010.5624295.

[29] MARTINS, W. A., DINIZ, P. S. R. "Block-Based Transceivers With Minimum Redundancy", *IEEE Transactions on Signal Processing*, v. 58, n. 3, pp. 1321–1333, March 2010. ISSN: 1053-587X. doi: 10.1109/TSP.2009.2033000.

[30] MARTINS, W. A., DINIZ, P. S. R. "Minimum redundancy multicarrier and single-carrier systems based on Hartley transforms". In: *2009 17th European Signal Processing Conference*, pp. 661–665, Aug 2009.

[31] MARTINS, W. A., DINIZ, P. S. R. "Suboptimal Linear MMSE Equalizers With Minimum Redundancy", *IEEE Signal Processing Letters*, v. 17, n. 4, pp. 387–390, April 2010. ISSN: 1070-9908. doi: 10.1109/LSP.2010.2042515.

[32] MARTINS, W. A., DINIZ, P. S. R. "Pilot-aided designs of memoryless block equalizers with minimum redundancy". In: *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, pp. 3312–3315, May 2010. doi: 10.1109/ISCAS.2010.5537975.

[33] ETSI 3GPP. *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio transmission and reception (3GPP TS 36.101 version 10.3.0 Release 10).* Technical report, ETSI 3rd Generation Partnership Project (3GPP), 2011.

[34] TAFLOVE, A., HAGNESS, S. C. *Computational Electrodynamics: The Finite-Difference Time-Domain Method, Third Edition.* 3 ed. New York, NY, USA, Artech House, jun. 2005. ISBN: 1580538320. Disponível em: <http://www.worldcat.org/isbn/1580538320>.

[35] PINTO, R., MERCHED, R. "Compressed Sensing Block Decision Feedback Equalization in Radar Imaging : Interplays and Connections for Sparse Recovery", *Submitted to EURASIP Journal on Advances in Signal Processing*, 2017.

[36] DONOHO, D. L., ELAD, M. "Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization", *Proceedings of the National Academy of Sciences*, v. 100, n. 5, pp. 2197–2202, 2003. doi: 10.1073/pnas.0437847100. Disponível em: <http://www.pnas.org/content/100/5/2197.abstract>.

[37] ALEXEEV, B., CAHILL, J., MIXON, D. "Full Spark Frames", *Journal of Fourier Anal. and App.*, v. 18, n. 6, pp. 1167–1194, 2012. ISSN: 1069-5869. doi: 10.1007/s00041-012-9235-4. Disponível em: <http://dx.doi.org/10.1007/s00041-012-9235-4>.

[38] DONOHO, D. L., HUO, X. "Uncertainty principles and ideal atomic decomposition", *IEEE Transactions on Information Theory*, v. 47, n. 7, pp. 2845–2862, Nov 2001. ISSN: 0018-9448. doi: 10.1109/18.959265.

[39] TIBSHIRANI, R. "Regression Shrinkage and Selection Via the Lasso", *Journal of the Royal Statistical Society, Series B*, v. 58, pp. 267–288, 1994.

[40] C, E., ROMBERG, J. "ℓ1-magic: Recovery of Sparse Signals via Convex Programming". 2005. Disponível em: <http://users.ece.gatech.edu/~justin/l1magic/>.

[41] VAN DEN BERG, E., FRIEDLANDER, M. P. "Probing the Pareto frontier for basis pursuit solutions", *SIAM Journal on Scientific Computing*, v. 31, n. 2, pp. 890–912, 2008. doi: 10.1137/080714488. Disponível em: <http://link.aip.org/link/?SCE/31/890>.

[42] CANDÈS, E. J. "The restricted isometry property and its implications for compressed sensing", *Comptes Rendus Mathematique*, v. 346, n. 9?10, pp. 589 – 592, 2008. ISSN: 1631-073X. doi: http://dx.doi.org/10.1016/j.crma.2008.03.014. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1631073X08000964>.

[43] CHEN, S., DONOHO, D., SAUNDERS, M. "Atomic Decomposition by Basis Pursuit", *SIAM Journal on Scientific Computing*, v. 20, n. 1, pp. 33–61, 1998. doi: 10.1137/S1064827596304010. Disponível em: <http://dx.doi.org/10.1137/S1064827596304010>.

[44] BOYD, S., VANDENBERGHE, L. *Convex Optimization*. New York, NY, USA, Cambridge University Press, 2004. ISBN: 0521833787.

[45] PARIKH, N., BOYD, S. "Proximal Algorithms", *Found. Trends Optim.*, v. 1, n. 3, pp. 127–239, jan. 2014. ISSN: 2167-3888. doi: 10.1561/2400000003. Disponível em: <http://dx.doi.org/10.1561/2400000003>.

[46] MALEKI, M. A., DONOHO, D. L., GRAY, R., et al. *Approximate message passing algorithms for compressed sensing*. Tese de Doutorado, Stanford University, 2010. Disponível em: <http://www-ece.rice.edu/%7Emam15/thesis.pdf>.

[47] RANGAN, S., SCHNITER, P., FLETCHER, A. "On the convergence of approximate message passing with arbitrary matrices". In: *2014 IEEE International Symposium on Information Theory*, pp. 236–240, June 2014. doi: 10.1109/ISIT.2014.6874830.

[48] VAIDYANATHAN, P. P., PHOONG, S. M., LIN, Y. *Signal Processing and Optimization for Transceiver Systems*. New York, NY, USA, Cambridge University Press, 2010.

[49] KAILATH, T., SAYED, A. H., HASSIBI, B. *Linear Estimation*. 1 ed. Upper Saddle River, NJ, USA, Prentice Hall, 2000. ISBN: 0130224642,9780130224644.

[50] MERCHED, R., YOUSEF, N. R. "Fast techniques for computing finite-length MIMO MMSE decision feedback equalizers", *IEEE Transactions on Signal Processing*, v. 54, n. 2, pp. 701–711, Feb 2006. ISSN: 1053-587X. doi: 10.1109/TSP.2005.861900.

[51] SUN, Y. "Bandwidth-efficient wireless OFDM", *IEEE Journal on Selected Areas in Communications*, v. 19, n. 11, pp. 2267–2278, Nov 2001. ISSN: 0733-8716. doi: 10.1109/49.963812.

[52] KAILATH, T., OLSHEVSKY, V. "Displacement-structure approach to polynomial Vandermonde and related matrices", *Linear Algebra and its Applications*, v. 261, n. 1, pp. 49 – 90, 1997. ISSN: 0024-3795. doi: http://dx.doi.org/10.1016/S0024-3795(96)00315-1. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0024379596003151>.

[53] MERCHED, R. "Fast generalized sliding window RLS recursions for IIR recurrence related basis functions". In: *2011 17th International*

Conference on Digital Signal Processing (DSP), pp. 1–6, July 2011. doi: 10.1109/ICDSP.2011.6004909.

[54] MERCHED, R. "A unified approach to structured covariances: Polynomial Vandermonde Bezoutian representations". In: *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pp. 1860–1864, Aug 2012.

[55] MERCHED, R. "Fast Algorithms in Slow and High Doppler Mobile Environments", *IEEE Transactions on Wireless Communications*, v. 9, n. 9, pp. 2890–2901, September 2010. ISSN: 1536-1276. doi: 10.1109/TWC.2010.062910.091638.

[56] MERCHED, R. "Turbo equalization in high doppler mobile environments: Channel estimation, fast algorithms and adaptive solutions". In: *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3197–3200, March 2008. doi: 10.1109/ICASSP.2008.4518330.

[57] MERCHED, R. "Exact trigonometric superfast inverse covariance representations". In: *2013 International Conference on Computing, Networking and Communications (ICNC)*, pp. 490–495, Jan 2013. doi: 10.1109/ICCNC.2013.6504134.

[58] MERCHED, R. "Superfast reduced-redundancy block memoryless linear equalizers". In: *2013 IEEE International Conference on Communications (ICC)*, pp. 4597–4602, June 2013. doi: 10.1109/ICC.2013.6655295.

[59] MARTINS, W. A., DINIZ, P. S. R. "LTI Transceivers With Reduced Redundancy", *IEEE Transactions on Signal Processing*, v. 60, n. 2, pp. 766–780, Feb 2012. ISSN: 1053-587X. doi: 10.1109/TSP.2011.2174056.

[60] MARTINS, W. A., DINIZ, P. S. R. "DHT-Based Transceivers With Reduced Redundancy", *IEEE Transactions on Signal Processing*, v. 60, n. 11, pp. 6080–6085, Nov 2012. ISSN: 1053-587X. doi: 10.1109/TSP.2012.2212017.

[61] MARTINS, W. A., DINIZ, P. S. R. "Block-based decision-feedback equalizers with reduced redundancy". In: *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pp. 56–60, Aug 2012.

[62] ILIC, J., STROHMER, T. "Sparsity Enhanced Decision Feedback Equalization", *IEEE Transactions on Signal Processing*, v. 60, n. 5,

pp. 2422–2432, May 2012. ISSN: 1053-587X. doi: 10.1109/TSP.2012.2189387.

[63] LEADBETTER, M. R., LINDGREN, G., ROOTZÉN, H. *Extremes and Related Properties of Random Sequences and Processes*. New York, NY, USA, Springer-Verlag, 1983.

[64] PEEBLES, P. *Probability, random variables, and random signal principles*. McGraw-Hill series in electrical engineering. Communications and signal processing. 2nd ed ed. New York, NY, USA, McGraw-Hill, 1987. ISBN: 0-07-044514-0,0-07-049219-0,0-07-049220-4.

[65] PAPOULIS, A., PILLAI, S. U. *Probability, Random Variables and Stochastic Processes*. 4th ed. New York, NY, USA, McGraw-Hill, 2002. ISBN: 0071226613,9780071226615.

[66] GEORGE CASELLA, R. L. B. *Statistical Inference*. 2 ed. Pacific Grove, CA, USA, Duxbury Press, 2001. ISBN: 0534243126,9780534243128.

[67] LOVRIC, M. *International Encyclopedia of Statistical Science*. 1st ed. New York, NY, USA, Springer, 2010. ISBN: 3642048978,9783642048975.

[68] DONOHO, D., TANNER, J. "Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing", *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, v. 367, n. 1906, pp. 4273–4293, 2009. ISSN: 1364-503X. doi: 10.1098/rsta.2009.0152. Disponível em: <`http://rsta.royalsocietypublishing.org/content/367/1906/4273`>.

[69] FISHLER, E., HAIMOVICH, A., BLUM, R., et al. "MIMO radar: an idea whose time has come". In: *Radar Conference, 2004. Proceedings of the IEEE*, pp. 71–78, 2004. doi: 10.1109/NRC.2004.1316398.

[70] LI, J., STOICA, P., XU, L., et al. "On Parameter Identifiability of MIMO Radar", *IEEE Sig. Proc. Letters*, v. 14, n. 12, pp. 968–971, 2007. ISSN: 1070-9908. doi: 10.1109/LSP.2007.905051.

[71] CARIN, L. "On the Relationship Between Compressive Sensing and Random Sensor Arrays", *Antennas and Propagation Magazine, IEEE*, v. 51, n. 5, pp. 72–81, 2009. ISSN: 1045-9243. doi: 10.1109/MAP.2009.5432044.

[72] XIE, Y., GUO, B., XU, L., et al. "Multi-Static Adaptive Microwave Imaging for Early Breast Cancer Detection". In: *Conference Record of the*

Thirty-Ninth Asilomar Conference onSignals, Systems and Computers, 2005., pp. 285–289, Oct 2005. doi: 10.1109/ACSSC.2005.1599751.

[73] NIKOLOVA, N. K. "Microwave Imaging for Breast Cancer", *IEEE Microwave Magazine*, v. 12, n. 7, pp. 78–94, Dec 2011. ISSN: 1527-3342. doi: 10.1109/MMM.2011.942702.

[74] LEHMAN, C. D., WHITE, E., PEACOCK, S., et al. "Effect of age and breast density on screening mammograms with false-positive findings." *American Journal of Roentgenology*, v. 173, n. 6, pp. 1651–1655, Dec 1999. ISSN: 0361-803X. doi: 10.2214/ajr.173.6.10584815. Disponível em: <http://dx.doi.org/10.2214/ajr.173.6.10584815>.

[75] KERLIKOWSKE, K., GRADY, D., BARCLAY, J., et al. "Effect of age, breast density, and family history on the sensitivity of first screening mammography", *JAMA*, v. 276, n. 1, pp. 33–38, 1996. doi: 10.1001/jama.1996.03540010035027. Disponível em: <http://dx.doi.org/10.1001/jama.1996.03540010035027>.

[76] CHENEY, M. "A Mathematical Tutorial on Synthetic Aperture Radar", *SIAM Rev.*, v. 43, n. 2, pp. 301–312, fev. 2001. ISSN: 0036-1445. doi: 10.1137/S0036144500368859. Disponível em: <http://dx.doi.org/10.1137/S0036144500368859>.

[77] NOLAN, C. J., CHENEY, M. "Synthetic aperture inversion", *Inverse Problems*, v. 18, n. 1, pp. 221, 2002. Disponível em: <http://stacks.iop.org/0266-5611/18/i=1/a=315>.

[78] VAN TREES, H. L. *Optimum Array Processing (Detection, Estimation, and Modulation Theory, Part IV)*. 1 ed. Hoboken, NJ, USA, Wiley-Interscience, mar. 2002. ISBN: 0471093904. Disponível em: <http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0471093904>.

[79] BERNSTEIN, D. S. *Matrix Mathematics: Theory, Facts, and Formulas*. 2nd ed. Princeton, NJ, USA, Princeton University Press, 2009.

[80] ROSSI, M., HAIMOVICH, A., ELDAR, Y. "Spatial Compressive Sensing for MIMO Radar", *IEEE Trans. on Sig. Proc.*, v. 62, n. 2, pp. 419–430, Jan 2014. ISSN: 1053-587X. doi: 10.1109/TSP.2013.2289875.

[81] WAX, M., ZISKIND, I. "On unique localization of multiple sources by passive sensor arrays", *Acoustics, Speech and Signal Processing, IEEE*

Transactions on, v. 37, n. 7, pp. 996–1000, 1989. ISSN: 0096-3518. doi: 10.1109/29.32277.

[82] STROHMER, T., FRIEDLANDER, B. "Analysis of Sparse MIMO Radar", *CoRR*, v. abs/1203.2690, 2012.

[83] HE, H., STOICA, P., LI, J. "Designing Unimodular Sequence Sets With Good Correlations - Including an Application to MIMO Radar", *IEEE Trans. on Sig. Proc.*, v. 57, n. 11, pp. 4391–4405, Nov 2009. ISSN: 1053-587X. doi: 10.1109/TSP.2009.2025108.

[84] SOLTANALIAN, M., NAGHSH, M. M., STOICA, P. "A fast algorithm for designing complementary sets of sequences", *Signal Processing*, v. 93, n. 7, pp. 2096–2102, 2013.

[85] LIEBIG, T., RENNINGS, A., HELD, S., et al. "OpenEMS-a free and open source equivalent-circuit (EC) FDTD simulation platform supporting cylindrical coordinates suitable for the analysis of traveling wave MRI applications", *Intl. Journal of Num. Modelling: Elec. Networks, Dev. and Fields*, v. 26, n. 6, pp. 680–696, 2013. ISSN: 1099-1204. doi: 10.1002/jnm.1875. Disponível em: <`http://dx.doi.org/10.1002/jnm.1875`>.

[86] LIEBIG, T. "OpenEMS - Open Electromagnetic Field Solver". 2011. Disponível em: <`http://openEMS.de`>.

Appendix A

Entrywise solution for the ℓ_0 proximal mapping

Consider the problem in eq. (2.127), where the norm has a diagonal weighting matrix. We want to minimize the cost

$$J(\mathbf{x}) \;=\; \|\mathbf{x}-\hat{\mathbf{x}}_i\|^2_{\mathbf{C}_i} \;+\; \|\mathbf{x}-\mathbf{x}_{i-1}\|_{0,\boldsymbol{\Lambda}_i}. \tag{A.1}$$

where

$$\|\mathbf{x}-\mathbf{x}_{i-1}\|_{0,\boldsymbol{\Lambda}_i} \;=\; \sum_{k=0}^{M-1} \epsilon'_{i,k}|x(k)-x_{i-1}(k)|_0$$

is the weighted ℓ_0-norm,and $|x(k)-x_{i-1}(k)|_0$ is an indicator function that returns 0 whenever $x(k)-x_{i-1}(k)=0$, and 1 otherwise, and $\mathbf{C}_i = \mathsf{Diag}(c_{i,0},\cdots,c_{i,M-1})$.

Defining $\mathbf{w}=\mathbf{x}-\mathbf{x}_{i-1}$ and $\Delta\hat{\mathbf{x}}_i = \hat{\mathbf{x}}_i - \mathbf{x}_{i-1}$, this is equivalent to find $\mathbf{w}$ that minimizes

$$\begin{aligned}
J(\mathbf{w}) &= \|\mathbf{w}-\Delta\hat{\mathbf{x}}_i\|^2_{\mathbf{C}_i} + \|\mathbf{w}\|_{0,\boldsymbol{\Lambda}_i} && \text{(A.2)}\\
&= \|\Delta\hat{\mathbf{x}}_i\|^2_{\mathbf{C}_i} + \sum_{k=0}^{M-1} c_{i,k}\left[w(k)w^*(k) - w(k)\Delta\hat{x}^*_i(k) - \Delta\hat{x}_i(k)w^*(k) + \frac{\epsilon'_{i,k}}{c_{i,k}}|w(k)|_0\right] && \text{(A.3)}\\
&\triangleq \|\Delta\hat{\mathbf{x}}_i\|^2_{\mathbf{C}_i} + \sum_{k=0}^{M-1} c_{i,k} D_k(w(k)). && \text{(A.4)}
\end{aligned}$$

Thus, minimizing $J(\mathbf{w})$ corresponds to minimize each $D_k(w(k))$. Note that $D_k(w(k))$ evaluate as follows

$$D_k(w(k)) = \begin{cases} 0, & \text{if } w(k)=0 \\ w(k)w^*(k) - w(k)\Delta\hat{x}^*_i(k) - \Delta\hat{x}_i(k)w^*(k) + \epsilon_{i,k}, & \text{otherwise,} \end{cases} \tag{A.5}$$

where we introduced $\epsilon_{i,k} = \frac{\epsilon'_{i,k}}{c_{i,k}}$. In the second case, we obtain a minimum when

$$\frac{\partial D_k(w_o(k))}{\partial w^*(k)} = 0 \Rightarrow w_o(k) - \Delta\hat{x}_i(k) = 0 \Rightarrow w_o(k) = \Delta\hat{x}_i(k) \tag{A.6}$$

In that case, the minimum cost $D_k(\Delta\hat{x}_i(k))$ will be

$$D_k(\Delta\hat{x}_i(k)) = \epsilon_{i,k} - |\Delta\hat{x}_i(k)|^2, \tag{A.7}$$

which is negative when $|\Delta\hat{x}_i(k)| \geq \sqrt{\epsilon_{i,k}}$. Hence, the optimal solution is

$$w_o(k) = \begin{cases} 0, & \text{if } |\Delta\hat{x}_i(k)| < \sqrt{\epsilon_{i,k}} \\ \Delta\hat{x}_i(k), & \text{otherwise} \end{cases} \tag{A.8}$$

Thus, replacing $\mathbf{w}$ by its definition, the solution is

$$x_i(k) = \begin{cases} x_{i-1}(k), & \text{if } |\Delta\hat{x}_i(k)| < \sqrt{\epsilon_{i,k}} \\ x_{i-1}(k) + \Delta\hat{x}_i(k), & \text{otherwise} \end{cases} \tag{A.9}$$

A more general problem, when the weighting matrix is not diagonal, is known to be combinatorial, and usually iterative algorithms are used to find a solution [1, 3, 7].