Linköping studies in science and technology. Dissertations,
No. 1570

Efficient
MIMO Detection Methods

Mirsad Čirkić

Division of Communication Systems
Department of Electrical Engineering (ISY)
Linköping University, SE-581 83 Linköping, Sweden
www.commsys.isy.liu.se

Linköping 2014

ISBN: 978-91-7519-413-4
ISSN 0345-7524
Printed in Sweden by LiU-Tryck, Linköping 2014

Sve za moju ljubav i moju familiju.

“Što ne boli—to nije život; što ne prolazi—to nije sreća”
Ivo Andrić

Abstract

For the past decades, the demand in transferring large amounts of data rapidly and reliably has been increasing drastically. One of the more promising techniques that can provide the desired performance is multiple-input multiple-output (MIMO) technology where multiple antennas are placed at both the transmitting and receiving side of the communication link. This performance potential is extremely high when the dimensions of the MIMO system are increased to an extreme (in the number of hundreds or thousands of antennas). One major implementation difficulty of the MIMO technology is the signal separation (detection) problem at the receiving side of the MIMO link, which holds for medium-size MIMO systems and even more so for large-size systems. This is due to the fact that the transmitted signals interfere with each other and that separating them can be very difficult if the MIMO channel conditions are not beneficial, i.e., the channel is not well-conditioned.

The main problem of interest is to develop algorithms for practically feasible MIMO implementations without sacrificing the promising performance potential that such systems bring. These methods involve inevitably different levels of approximation. There are computationally cheap methods that come with low accuracy and there are computationally expensive methods that come with high accuracy. Some methods are more applicable in medium-size MIMO than in large-size MIMO and vice versa. Some simple methods for instance, which are typically inaccurate for medium-sized settings, can achieve optimal accuracy for certain large-sized settings that offer close-to-orthogonal spatial signatures. However, when the dimensions are overly increased, then even these (previously) simple methods become computationally burdensome. In different MIMO setups, the difficulty in detection shifts since methods with optimal accuracy are not the same. Therefore, devising one single algorithm which is well-suited for feasible MIMO implementations in all settings is not easy.

This thesis addresses the general MIMO detection problem in two ways. One part treats a development of new and more efficient detection techniques for the different MIMO settings. The techniques that are proposed in this thesis demonstrate unprecedented performance in many relevant cases. The other part revolves around utilizing already proposed detection algorithms and their advantages versus disadvantages in an adaptive manner. For well-conditioned channels, low-complexity detection methods are often sufficiently accurate. In such cases, performing computationally very expensive optimal detection would be a waste of computational power. This said, for MIMO detection in a coded system, there is always a trade-off between performance and complexity. Intuitively, computational resources should be utilized more efficiently by performing optimal detection only when it is needed, and something simpler when it is not. However, it is not clear whether this is true or not. In trying to answer this, a general framework for adaptive computational-resource allocation to different ("simple" and "difficult") detection problems is proposed. This general framework is applicable to any MIMO detector and scenario of choice, and it is exemplified using one particular detection method for which specific allocation techniques are developed and evaluated.

Acknowledgments

First of all, I would like to thank my supervisor, Prof. Erik G. Larsson, who has given me the opportunity and guidance to my first steps in entering the world's research community. His great knowledge and ability to question, to address, and to solve technical problems is very inspiring. I would also like to thank my colleagues at Communication Systems and Information Coding divisions. Especially, Dr. Daniel Persson and Docent Jan-Åke Larsson, who were never too "busy" to give a helping hand and were always open for fruitful discussions, deserve a big thank you. I also want thank my fellow PhD students: especially Reza Moosavi, TVK Chaitanya, Erik Axell, and Johannes Lindblom for many valuable technical and non-technical discussions. I wish to convey my sincere thanks and appreciation to Dr. Lasse Alfredsson and Dr. Mikael Olofsson for their help in my teaching work.

I thank my parents, Emir & Raza Čirkić, that have supported and encouraged me to pursue my dream of becoming a researcher. Last but most definitely not least, I would like to thank my wife, Therése M. Čirkić, and my daughter, Leila T. Čirkić. Thank you Therese for never doubting in me and for putting up with me working many late after-office hours. Thank you Leila for bringing such joy to my life.

Linköping, 2014
Mirsad Čirkić

Contents

Part I

Introduction

Motivation

During the past decades, the demand in transferring large amounts of data rapidly and reliably [1] has been increasing drastically. This demand is not likely to reduce but rather increase. One way of increasing data speed/throughput is to increase the spectral bandwidth. However, in the frequency bands where communication is possible today, bandwidth is a very limited resource. One could keep the spectral bandwidth fixed and increase the transmission power to get better signal strengths and thus to be able to send more bits per transmission. However, this would cause interference to fellow “communicators” that would also like to send information with the same data speed. Another way, and an even more promising one, is to increase the spatial bandwidth (number of elements along the spatial dimension), which is not utilized fully today and therefore offers many degrees of freedom yet to be exploited. One way of doing that is by placing multiple antennas at both the transmitting and receiving side of the communication link. This technology is called multiple-input multiple-output (MIMO). It has many advantages over the single-antenna technologies. For instance, it increases the information throughput [2] and link reliability by exploiting link diversity [3] in rich scattering environments. However, as many other technologies, MIMO involves certain obstacles that are not trivial and are very difficult to solve.

One such major implementation difficulty of the MIMO technology is the signal separation (detection) problem at the receiving side of the MIMO link. This is due to the fact that the transmitted signals superimpose and the information from the different antennas interfere with each other. Now, if the MIMO channel matrix is not well-conditioned and is close to singular, solving the detection problem becomes very difficult. Optimal detection methods, which have been well-known for a long time, enumerate the transmitted constellation. Unfortunately, the complexity of such enumeration procedures increases exponentially in the number of transmitting antennas, and therefore they are not realizable for large numbers of transmitting antennas. To overcome this difficulty, many smart techniques that have a feasible computational complexity have been proposed, some examples are [4–16]. All of them employ different approximations in order to achieve low complexity at the expense of accuracy. There is an inherent trade-off between accuracy and complexity that decides what total performance a detector will have and how good the algorithm is. The goal is to achieve the accuracy of the optimal methods with as low complexity as possible. When comparing different methods, which is not always a simple task to do fairly, one has to take into account other parts of the communication system. For instance, if a detection method has an accuracy which is tolerable, then using a different algorithm which has better accuracy at the expense of higher complexity is not preferable. The reason is that most communication systems protect the data with outer codes which tolerate a certain degree of inaccuracy. By contrast, in the case where the current detector is not the complexity bottleneck of the system, then employing a different detector with a much lower complexity at the expense of accuracy is not preferred. Therefore, just looking at the raw figures of accuracy and complexity is not always enough to decide which detector is better than the other one.

The research revolving around MIMO detection had its first boom during the code-division multiple-access (CDMA) era, where the “MIMO link” consisted of multiple interferring streams measured by a receiver with multiple correlators intended for the individual streams [17–20]. Since then, a vast literature has been produced that presents different approximate detection methods. The simplest and computationally cheapest algorithms use some type of linear preprocessing in order to decouple the streams before making the decision on which sequence of bits has been transmitted. Other methods use non-linear approximations to overcome the shortcomings of the linear methods. The more advanced methods offer the possibility of trading accuracy for computational complexity via some user parameter: sphere decoding (SD) [4], reduced-dimension maximum a posteriori [21], fixed-complexity SD (FCSD) [5], soft-output via partial marginalization (PM) [6], reduced-dimension maximum-likelihood search [22], and lattice reduction (LR) [7].

For the past five years, there has been a shift in focus in the MIMO detection literature. Namely, the emerging of MIMO systems with very large-sized antenna arrays with sizes in the number of hundreds. In the earlier research literature, MIMO stood most often for multiantenna systems that had tens or less number of antennas. These setups are now usually called as conventional (or as medium-sized) MIMO systems and have a square or close to square structure. This means that the number of transmitting antennas is roughly equal to the number of receiving antennas. The terminology very large MIMO, massive MIMO, large multi-user MIMO, large antenna arrays etc., can refer to MIMO systems that have both a square and a rectangular structure. Due to the overly large dimensions, some of the methods previously developed mainly for medium-sized MIMO and that performed well, were not that well-suited for large-sized

MIMO systems with hundreds of antennas. Other and very simple search methods with very low complexity [23] that were specifically developed for large-sized MIMO showed to have very good accuracy at a very low computational complexity. At the same time, as will be discussed in this thesis, certain linear methods achieve close-to-optimal performance with very low complexity. Even more interestingly for the rectangular MIMO systems, where the ratio between the number of transmitting and receiving antennas is very small, is that methods which previously performed poorly turned out to be close-to-optimal.

The vast majority of the MIMO detection literature presents different detection techniques, but not which technique to use based on the effective channel conditions. For well-conditioned channels, sub-optimal low-complexity methods such as soft zero-forcing (ZF), are often sufficiently accurate. In fact, for wireless environments that yield channel matrices with orthogonal columns, the ZF method is optimal and equivalent to maximum ratio combining. In such a case, performing optimal detection and computationally heavy enumeration would be a waste of computational power. This said, for MIMO detection, there is always a trade-off between performance and complexity. The fundamental question is, can we save computational resources by performing heavy enumeration only when it is needed, and something simpler when it is not? There is not much earlier work that considers such adaptive detection, which is one of the questions that this thesis addresses and tries to answer. The other aspect is to develop smart detection techniques that work very well for all MIMO systems.

Chapter 1

MIMO Detection

The effective MIMO channel model that is used in this thesis, and commonly in literature, is

$$\boldsymbol{y}_{\mathrm{c}} = \boldsymbol{H}_{\mathrm{c}}\boldsymbol{s}_{\mathrm{c}} + \boldsymbol{e}_{\mathrm{c}}, \tag{1}$$

where $\boldsymbol{y}_{\mathrm{c}} \in \mathbb{C}^{{N_{\mathrm{R}}}'}$ is the received (output) vector, $\boldsymbol{H}_{\mathrm{c}} \in \mathbb{C}^{{N_{\mathrm{R}}}' \times {N_{\mathrm{T}}}'}$ is MIMO-channel matrix, $\boldsymbol{s}_{\mathrm{c}} \in \mathcal{S}_{\mathrm{c}}^{{N_{\mathrm{T}}}'}$ is the transmitted (input) vector, and $\boldsymbol{e}_{\mathrm{c}} \in \mathbb{C}^{{N_{\mathrm{R}}}'}$ is the received noise. The noise is typically distributed as a zero-mean cirularly symmetric complex Gaussian stochastic vector. Several different communication settings can be modeled with (1). The most common setting used in this context is the flat-fading scenario with ${N_{\mathrm{T}}}'$ transmit antennas (input) and ${N_{\mathrm{R}}}'$ receive antennas (output), where each input or output actually represents one physical antenna as in Fig. 1. Other examples are for instance a bidirectional single-antenna communication link that spans several frequency bins and time slots, where each frequency-and-time slot is associated with one element in $\boldsymbol{H}_{\mathrm{c}}$. To simplify the exposition of some of the methods addressed in this thesis, we rewrite the complex-valued model (1) into an equivalent real-valued MIMO-channel model of double dimensions ($N_{\mathrm{R}} = 2{N_{\mathrm{R}}}'$ and $N_{\mathrm{T}} = 2{N_{\mathrm{T}}}'$),

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{s} + \boldsymbol{e}, \tag{2}$$

where $\boldsymbol{H} \in \mathbb{R}^{N_{\mathrm{R}} \times N_{\mathrm{T}}}$ is the MIMO channel matrix with full column-rank and $\boldsymbol{s} \in \mathcal{S}^{N_{\mathrm{T}}}$ is the transmitted vector. Further, $\boldsymbol{e} \in \mathbb{R}^{N_{\mathrm{R}}} \sim \mathcal{N}(\boldsymbol{0}, \frac{N_0}{2}\boldsymbol{I})$ denotes the noise vector and $\boldsymbol{y} \in \mathbb{R}^{N_{\mathrm{R}}}$ is the received vector. This can be done for instance if the constellation $\mathcal{S}_{\mathrm{c}}$ is seperable into two real-valued, identical, and independent dimensions. There are other more general conditions when equivalence applies, but they are not necessary to introduce in this thesis. The exact details of this rewrite are presented on page 127. To simplify the exposition of this thesis, the restriction $N_{\mathrm{R}} \geq N_{\mathrm{T}}$ is enforced since some detection methods that are treated here require it—some do not as will be explicitly stated. It is noteworthy that this restriction is typical in practice.

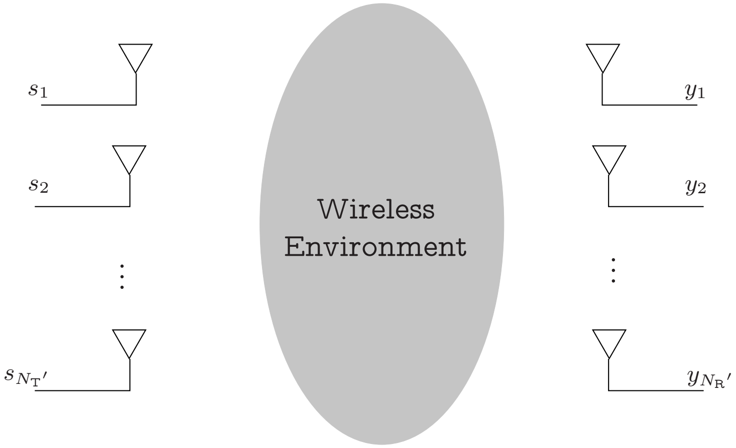

Figure 1: MIMO Setup.

There are two main categories of detectors: hard decision detectors which decide whether a bit is zero or one and soft decision detectors which decide how likely it is that a bit is zero or one. The latter category of detectors in comparison with the previous one produces more information at the output and in a receiving chain with appropriate soft decoders will yield much better performance. Nevertheless, with the so-called max-log approximation, one can make any hard decision detector, such as SD, to produce soft values. This has resulted in much of the literature focusing on finding efficient hard decision methods.

Next, there are two more important categories of MIMO detectors. The first consists of detectors whose complexity (run-time) depends on the particular channel and noise realization. This category includes, in particular, methods that perform a tree-search. Notable examples include the SD method and its many relatives [14,22,24–27]. Unfortunately, these methods have an exponential worst-case complexity unless a suboptimal termination criterion is used. The other category of detectors consists of methods that have a fixed (deterministic) complexity that does not depend on the channel realization. These methods are more desirable from an implementation point of view, as they eliminate the need for data buffers and over-dimensioning (for the worst-case) of the hardware. Examples of such detectors are the reduced dimension maximum a posteriori (RDMAP) method [21] and the fixed-complexity SD (FCSD) method [5]. These fixed-complexity detectors provide a simple and well-defined tradeoff between computational complexity and performance, they have a fixed and fully predictable run-time, and they are highly parallelizable.

1.1 Hard-Decision MIMO Detection

The optimal hard MIMO (symbol-vector) detector is the one that chooses the candidate in the symbol vector constellation $\mathcal{S}^{N_\mathrm{T}}$ that maximizes the a posteriori probability of the symbol vectors. Hence,

$$\hat{\boldsymbol{s}} \triangleq \underset{\boldsymbol{s}\in\mathcal{S}^{N_\mathrm{T}}}{\operatorname{argmax}}\, P(\boldsymbol{s}|\boldsymbol{y}) = \underset{\boldsymbol{s}\in\mathcal{S}^{N_\mathrm{T}}}{\operatorname{argmax}}\, p(\boldsymbol{y}|\boldsymbol{s})P(\boldsymbol{s}). \tag{3}$$

With uniform a priori probabilities and the model in (2), the hard detection problem is equivalent to

$$\hat{\boldsymbol{s}} = \underset{\boldsymbol{s}\in\mathcal{S}^{N_\mathrm{T}}}{\operatorname{argmin}}\, \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{s}\|^2 . \tag{4}$$

This problem, which is often referred to as the maximum likelihood (ML) problem, if solved via brute-force enumeration, has an exponential complexity in N_T ($|\mathcal{S}|^{N_\mathrm{T}}$ possible solution candidates). The high complexity that arises in this problem statement is the main issue in MIMO detection. It has required serious attention and still does. Thus, many approximate methods have been proposed, some of which are explained thoroughly in the sections that follow. A more condensed overview can be found in [25, 28].

1.1.1 Zero Forcing and Matched Filtering

One of the crudest and the simplest approximations of the problem in (4) is the zero forcing (ZF) solution. It consists of two steps: decoupling of the interfering symbols using the pseudo-inverse $(\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T$, i.e., the least squares filter, and then quantizing the decoupled symbols independently per dimension. An even simpler method is the matched filter (MF) method, which applies only a scaled version of $\boldsymbol{H}^T$ and disregards the inverse involved in ZF. MF works well only for scenarios that yield orthogonal (or very close to orthoganl) columns in $\boldsymbol{H}$ making the MF equivalent to ZF since the matrix $\boldsymbol{H}^T\boldsymbol{H}$ becomes diagonal (or very close to diagonal).

The first step involves an approximation that relaxes the ML problem and the constraint $\boldsymbol{s} \in \mathcal{S}^{N_\mathrm{T}}$,

$$\hat{\boldsymbol{s}}_{\mathrm{uq}} \triangleq \underset{\boldsymbol{s}\in\mathbb{R}^{N_\mathrm{T}}}{\operatorname{argmin}}\, \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{s}\|^2 = (\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{y}. \tag{5}$$

The unquantized solution $\hat{\boldsymbol{s}}_{\mathrm{uq}}$ is then elementwise quantized (clipped) to the closest constellation points in $\mathcal{S}$,

$$\hat{\boldsymbol{s}}_{\mathrm{zf}} \triangleq \underset{\boldsymbol{s}\in\mathcal{S}^{N_\mathrm{T}}}{\operatorname{argmin}}\, \|\hat{\boldsymbol{s}}_{\mathrm{r}} - \boldsymbol{s}\| . \tag{6}$$

The complexity of the ZF detector is mainly dictated by the computation of $(\boldsymbol{H}^T\boldsymbol{H})^{-1}$, which is much (exponentially) smaller than that of brute-force enumeration. The main problem that inhibits the performance of the ZF algorithm is the occurrence of ill-conditioned $\boldsymbol{H}$ matrices, i.e., close to linearly dependent columns, which after the inversion in (5) significantly increase the noise variance in $\hat{\boldsymbol{s}}_{\text{uq}}$ compared to that in $\boldsymbol{y}$. Note that if $\boldsymbol{H}$ has not full column-rank, i.e., $\boldsymbol{H}^T\boldsymbol{H}$ is singular, which happens for instance if $\boldsymbol{H}$ is underdetermined, the ZF solution does no exist.

One particularly useful extension of this procedure is the one that uses the linear minimum mean square-error (MMSE) filter instead of the least squares filter. The underlying assumption differs in that MMSE assumes $\boldsymbol{s}$ as random and Gaussian distributed (typically with independent and identically distributed elements). The MMSE inspired solutions improve the performance somewhat, but unfortunately do not avoid the fundamental issue regarding ill-conditioned problems. The procedural difference to MMSE is that, instead of the pseudo-inverse, MMSE uses $(\boldsymbol{H}^T\boldsymbol{H}+\boldsymbol{I}\frac{N_0}{2})^{-1}\boldsymbol{H}^T$. Hence, MMSE requires knowledge of N_0 as opposed to ZF, but it can be used for underdetermined MIMO systems as well. For more details, see Sec. 1.2.3.

1.1.2 Zero Forcing with Decision-Feedback

The ZF with decision-feedback (ZF-DF) [29] method is an iterative extension to the ZF method. It is also sometimes referred to as zero forcing with successive interference cancellation (ZF-SIC). Note that there is a similar variation of the algorithm known as MMSE-SIC, which extends ZF-SIC in the same manner as ZF in Sec. 1.1.1 is extended to MMSE. Instead of performing the elementwise quantization in parallel as in ZF, ZF-DF does it successively. Hence, if we let $\boldsymbol{h}_i \in \mathbb{R}^{N_\text{R}}$ be the columns of $\boldsymbol{H}$ and run the algorithm in natural order, the steps are as follows

$$\begin{aligned}
\hat{\boldsymbol{s}}_1 &= \underset{s_i\in\mathbb{R}, i=1,\ldots,N_\text{T}}{\operatorname{argmin}} \Big\|\boldsymbol{y} - \sum_{i=1}^{N_\text{T}} \boldsymbol{h}_i s_i\Big\|^2, && \hat{s}_{\text{df},1} \triangleq \lfloor \hat{s}_{1,1} \rceil,\\
\hat{\boldsymbol{s}}_2 &= \underset{s_i\in\mathbb{R}, i=2,\ldots,N_\text{T}}{\operatorname{argmin}} \Big\|\boldsymbol{y} - \boldsymbol{h}_1\hat{s}_{\text{df},1} - \sum_{i=1}^{N_\text{T}} \boldsymbol{h}_i s_i\Big\|^2, && \hat{s}_{\text{df},2} \triangleq \lfloor \hat{s}_{2,1} \rceil,\\
&\vdots\\
\hat{\boldsymbol{s}}_{N_\text{T}} &= \underset{s_{N_\text{T}}\in\mathbb{R}}{\operatorname{argmin}} \Big\|\boldsymbol{y} - \sum_{i=1}^{N_\text{T}-1} \boldsymbol{h}_i\hat{s}_{\text{df},i} - \boldsymbol{h}_{N_\text{T}} s_{N_\text{T}}\Big\|^2, && \hat{s}_{\text{df},N_\text{T}} \triangleq \lfloor \hat{s}_{N_\text{T},1} \rceil.
\end{aligned}$$

where $\lfloor x \rceil \triangleq \operatorname{argmin}_{s\in\mathcal{S}} \|s-x\|^2$ and the ZF-DF solution is denoted with $\hat{\boldsymbol{s}}_{\text{df}}$.

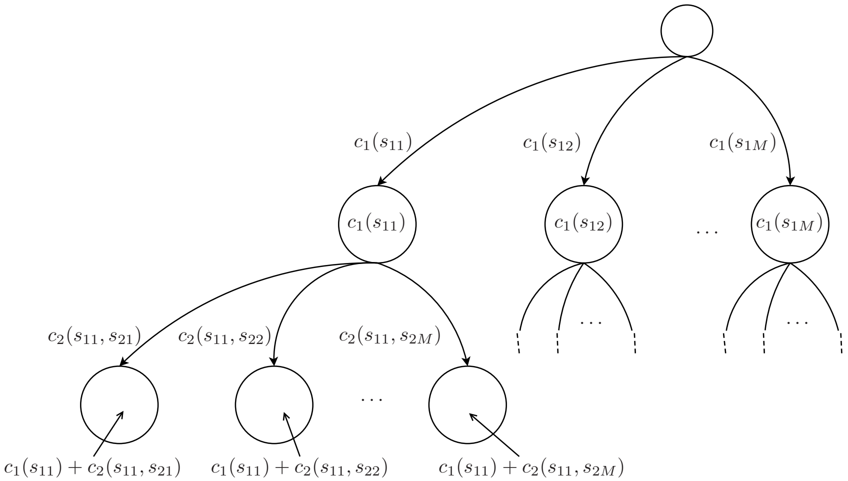

Figure 2: Sphere decoding and hard decision detection as a tree-search problem. In this figure, the first three layers are shown where the branch metric at layer k is denoted as $c_k(s_1, \dots, s_k) \triangleq \left(y'_k - \sum_{j=1}^{k} L_{kj} s_j\right)^2$ and a certain constellation point m at layer k as s_{km}.

Compared to the ZF method, the complexity is of ZF-DF is of the same order if implemented efficiently. One efficient procedure uses the rank-one update formula by Sherman and Morrison [30]. ZF-DF method generally yields better performance. The main issue with it is, due to occurrence of large initial noise variance, the increasing error propagation that occurs when a wrong decision is made in a preceding step. This error is then highly likely to induce an error in the next decision and so on. Error propagation can be partially reduced by ordering the symbols properly, instead of using natural ordering as the algorithm is presented. A good ordering approach is to detect the symbols with the lowest noise variance first. This is not necessarily the optimal approach and there are indeed many possible ways available to order the symbols to improve the performance [31–33]. Unfortunately, even with optimal ordering (the ordering that minimizes $\|\boldsymbol{y} - \boldsymbol{H}\hat{\boldsymbol{s}}_{\mathrm{df}}\|^2$), the error propagation has a significant impact on the performance since the fundamental issue with ill-conditioned channel matrices remains [31]. The worse the conditioning is, the larger is the risk of significant error propagation.

1.1.3 Sphere Decoding

The sphere decoding method, presented in [34] in a different context than MIMO detection, performs a partial enumeration of the entire constellation $\mathcal{S}^{N_\mathrm{T}}$ [4, 35, 36]. It does that via a pre-determined user parameter, the so-called sphere radius R, by considering only candidate vectors $\boldsymbol{s} \in \mathcal{S}^{N_\mathrm{T}}$ that lie inside a certain sphere; hence the name sphere decoding. In order to explain this method, the channel matrix is QL-decomposed according to $\boldsymbol{H} = \boldsymbol{QL}$, where $\boldsymbol{Q} \in \mathbb{R}^{N_\mathrm{R} \times N_\mathrm{T}}$ has orthonormal columns and $\boldsymbol{L} \in \mathbb{R}^{N_\mathrm{T} \times N_\mathrm{T}}$ is a lower-triangular matrix. Hence,

$$\hat{\boldsymbol{s}} = \underset{\boldsymbol{s} \in \mathcal{S}^{N_\mathrm{T}}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{Hs}\|^2 = \underset{\boldsymbol{s} \in \mathcal{S}^{N_\mathrm{T}}}{\operatorname{argmin}} \|\boldsymbol{y}' - \boldsymbol{Ls}\|^2 = \underset{\boldsymbol{s} \in \mathcal{S}^{N_\mathrm{T}}}{\operatorname{argmin}} \sum_{i=1}^{N_\mathrm{T}} \Big(y'_i - \sum_{j=1}^{i} L_{ij} s_j\Big)^2, \quad (7)$$

where $\boldsymbol{y}' = \boldsymbol{Q}^T \boldsymbol{y}$. This problem can be thought of as a top-down tree-search problem (see Fig. 2) with the branch metric $\big(y'_k - \sum_{j=1}^{k} L_{kj} s_j\big)^2$ at tree-level k. For tree-level $k = 1, \ldots, N_\mathrm{T}$, starting at level 1, element s_k is fixed to a value in $\mathcal{S}$, which decides the next branch to be traversed down to level $k+1$. At tree-level k, the accumulated branch metric, which is $\sum_{i=1}^{k} \big(y'_i - \sum_{j=1}^{i} L_{ij} s_j\big)^2$, is validated not to be larger than the SD radius R, i.e., fulfills $\sum_{i=1}^{k}(y'_i - \sum_{j=1}^{i} L_{ij} s_j)^2 \leq R$. If that is the case, the algorithm moves ahead to level $k+1$ along the branch that corresponds to the value s_k was fixed to. If by contrary the accumulated branch metric is larger than R, moving ahead to level $k+1, k+2, \ldots, N_\mathrm{T}$ will only increase the accumulated metric and therefore always be larger than R no matter which branches below level k are chosen. Therefore, the algorithm prunes the sub-tree that is connected to the main tree via the particular branch that failed the radius check. In the end, if the radius R is large enough such that the SD method visits the paths down to layer N_T, i.e., the leaf nodes, the fixed values of $s_1, s_2, \ldots, s_{N_\mathrm{T}}$ that yield the smallest accumulated branch metric $(\sum_{i=1}^{N_\mathrm{T}}(y'_i - \sum_{j=1}^{i} L_{ij} s_j)^2)$ when the SD method finishes give the solution to (4). If R is not large enough, the algorithm will not reach any leaf node and therefore not yield a complete solution. Also note that if R is chosen to be very large, say ∞, then the algorithm will not prune any sub-trees and therefore visit all $|\mathcal{S}|^{N_\mathrm{T}}$ leaf nodes. Thus, choosing R properly is very important in order for SD to solve the problem in (4) efficiently and completely.

The procedure described above assumes that the matrix $\boldsymbol{L}$ is invertible (all diagonal elements are non-zero), which is the case if $\boldsymbol{H}$ has full column-rank. If $\boldsymbol{L}$ is not invertible, the situation becomes more difficult. However, with some extra tricks, it is possible to use SD in such scenarios as well [37].

There are many extensions to this method. One of the simplest extensions is SD with adaptive radius updating. Some more advanced extensions are mentioned in Sec. 1.1.4 and 1.1.5. In the procedure with adaptive radius R, the radius is first set to some very large value. Then, when the algorithm reaches a leaf node (a node in layer N_T) and a candidate solution vector $\hat{s}$ is found, its accumulated branch metric will be smaller than the current radius, i.e., $\|\boldsymbol{y}' - \boldsymbol{L}\hat{\boldsymbol{s}}\|^2 < R$—this is true since we know that SD will only choose a path that passes the radius check. The current radius R is then reduced to the current accumulated metric $\|\boldsymbol{y}' - \boldsymbol{L}\hat{\boldsymbol{s}}\|^2$. If the algorithm reaches another leaf node afterwards and another candidate solution is found, the then current radius is updated again since the accumulated metric found then will be smaller. The procedure ends when no other leaf node is reached due to pruning with the last updated radius. How fast the radius is reduced from one leaf node to another depends highly on the conditioning of the channel matrix, where ill-conditioned matrices make the pruning process (radius reduction) very slow.

The main advantages of the SD is that it is simple to implement and apprehend, and that it guaranties to deliver an optimal solution. One of the main drawbacks of the SD algorithm is the variable complexity depending on the effective channel conditions, and that the worst-case and as well as the expected-case complexity is exponential in the number N_T [38]. Due to the sequential tree-search like nature of the algorithm, it does not fit well for parallel implementations which is a necessity when N_T is large.

1.1.4 Fixed Complexity Sphere Decoding

To address the drawbacks of the SD algorithm, the authors in [5, 39] proposed an approach that makes the complexity of SD invariable and provides a highly parallelizable structure. In order to explain the fixed complexity SD (FCSD) method in [39], for fixed $r \in \{0, \ldots, N_\mathrm{T}-1\}$, we define the following partitioning of the model in (2)

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{s} + \boldsymbol{e} = \underbrace{\begin{bmatrix}\overline{\boldsymbol{H}} & \widetilde{\boldsymbol{H}}\end{bmatrix}}_{\text{col. permut. of } \boldsymbol{H}} \underbrace{\begin{bmatrix}\bar{\boldsymbol{s}}^T & \tilde{\boldsymbol{s}}^T\end{bmatrix}^T}_{\text{permut. of } \boldsymbol{s}} + \ \boldsymbol{e} = \overline{\boldsymbol{H}}\bar{\boldsymbol{s}} + \widetilde{\boldsymbol{H}}\tilde{\boldsymbol{s}} + \boldsymbol{e}, \tag{8}$$

where $\overline{\boldsymbol{H}} \in \mathbb{R}^{N_\mathrm{R} \times r+1}$, $\widetilde{\boldsymbol{H}} \in \mathbb{R}^{N_\mathrm{R} \times (N_\mathrm{T}-r-1)}$, $\bar{\boldsymbol{s}} \in \mathcal{S}^{r+1}$, and $\tilde{\boldsymbol{s}} \in \mathcal{S}^{N_\mathrm{T}-r-1}$. The choice of partitioning involves the choice of a permutation, and how to perform this choice is not obvious. In fact, there are $\binom{N_\mathrm{T}}{r+1}$ possible partitionings in (8). The aim in FCSD is to find a partitioning such that the condition number of the matrix $\widetilde{\boldsymbol{H}}$ is minimized and it will be clear why in what follows.

The FCSD method offers a trade-off between exact and approximate computation of (4) via the parameter r. More specifically, the FCSD splits the minimization in (4) into two parts

$$\hat{\boldsymbol{s}} = \underset{\boldsymbol{s} \in \mathcal{S}^{N_\mathrm{T}}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{s}\| = \underset{\bar{\boldsymbol{s}} \in \mathcal{S}^{r+1}}{\operatorname{argmin}} \ \underset{\tilde{\boldsymbol{s}} \in \mathcal{S}^{N_\mathrm{T}-r-1}}{\operatorname{argmin}} \left\|\boldsymbol{y} - \overline{\boldsymbol{H}}\bar{\boldsymbol{s}} - \widetilde{\boldsymbol{H}}\tilde{\boldsymbol{s}}\right\|, \tag{9}$$

and then approximates the second minimization by a simple sub-optimal hard detector such as ZF-DF. Thus, the enumeration is only performed over the $\bar{\boldsymbol{s}}$ part of the vector $\boldsymbol{s}$. This can be viewed in the tree-search problem in Fig. 2 as a full search (follows all branch paths) down to tree-level $r+1$ followed by a greedy search that only explores one of all possible paths down to the last tree-level. The ZF-DF method is computationally much more efficient than the exact minimization in (9), but it performs well only for well-conditioned matrices. However, the secondary minimization problems in (9) are generally well-conditioned since the matrices $\widetilde{\boldsymbol{H}}$ are tall. In addition to that, when forming the partitioning in (8), the original symbol order in $\boldsymbol{s} = [s_1, \ldots, s_{N_\mathrm{T}}]^T$ is permuted in (8) so that the condition number of $\widetilde{\boldsymbol{H}}$ is minimized. Notably, FCSD performs ZF-DF for $r = 0$ and solves the exact ML problem (as defined by (4)) for $r = N_\mathrm{T} - 1$.

1.1.5 Reduced-Dimension Maximum Likelihood Search

The reduced-dimension maximum-likelihood search (RD-MLS) method [22] uses the same core idea as the FCSD. It splits the minimization problem in (4), by using the partitioned model in (8), into two parts as in (9). Then, as opposed to the FCSD method that uses a simple hard detector to solve the secondary problem, it uses solely a linear estimator $\boldsymbol{F}$ such as the least squares estimator (ZF filter) to approximate the secondary problem without performing any quantization. Hence,

$$\hat{\bar{\boldsymbol{s}}} = \underset{\bar{\boldsymbol{s}} \in \mathcal{S}^{r+1}}{\operatorname{argmin}} \left\| \boldsymbol{y} - \overline{\boldsymbol{H}}\bar{\boldsymbol{s}} - \widetilde{\boldsymbol{H}}\boldsymbol{F}(\boldsymbol{y} - \overline{\boldsymbol{H}}\bar{\boldsymbol{s}}) \right\| = \underset{\bar{\boldsymbol{s}} \in \mathcal{S}^{r+1}}{\operatorname{argmin}} \left\| \boldsymbol{z} - \boldsymbol{G}\bar{\boldsymbol{s}} \right\|, \qquad (10)$$

where $\boldsymbol{z} = \boldsymbol{y} - \widetilde{\boldsymbol{H}}\boldsymbol{F}\boldsymbol{y}$ and $\boldsymbol{G} = (\boldsymbol{I} - \widetilde{\boldsymbol{H}}\boldsymbol{F})\overline{\boldsymbol{H}}$. This reduces the dimension of the minimization to that of the space of $\bar{\boldsymbol{s}}$. Then, the problem with the reduced dimension is solved using an SD type of algorithm. Using the solution in the reduced dimension problem $\hat{\bar{\boldsymbol{s}}}$, the rest of the symbol vector is detected with a simple hard detector such as the MMSE with successive interference cancellation method. The main difference between the RD-MLS and the FCSD method is that FCSD uses the output of a simple hard detector in the secondary minimization when solving the primary minimization whereas the RD-MLS does not. The disadvantage of the RD-MLS algorithm is that it does not improve the conditioning of the reduced problem (matrix $\boldsymbol{G}$) compared to the original one (matrix $\boldsymbol{H}$), as is done in the FCSD method. The reason is the unquantized linear estimator (matrix $\boldsymbol{F}$) which essentially results in a projection of the reduced dimension space $\overline{\boldsymbol{H}}\bar{\boldsymbol{s}}$ onto the orthogonal complement of the column-space of $\widetilde{\boldsymbol{H}}$.

This algorithm provides two parameters with which complexity can be traded for performance: the dimension parameter r and the sphere radius R of the SD algorithm. The RD-MLS algorithm reduces the complexity of the SD method and it inherits the SD algorithm's properties, both good and bad. For instance, the disadvantage with variable complexity comes along.

1.1.6 Lattice-Reduction Aided Detectors

The lattice-reduction (LR) aided MIMO detection algorithms build upon the idea of converting an ill-conditioned problem into an equivalent well-conditioned problem via a linear transform $\boldsymbol{T}$ that fulfills certain conditions. Once the problem is well-conditioned, simple hard detectors such as ZF can be used to achieve near-optimal performance. That said, LR type of methods are preprocessing procedures that operate on the channel matrix only rather than pure detection methods that operate on the received data vector $\boldsymbol{y}$ as well.

To explain LR preprocessing, the finite constellation $\mathcal{S}$ (assuming uniformly spaced constellation points) is first scaled with some scalar $1/\alpha$ and extended to enumerate all integers $\mathbb{Z}$. The infinite constellation $\mathbb{Z}^{N_\mathrm{T}}$ represents a square lattice with dimension N_T. Then the ML problem in (4) is relaxed to find the point in the lattice (after applying $\alpha\boldsymbol{H}$) that is closest to the received data $\boldsymbol{y}$. Hence, the relaxed ML problem becomes

$$\hat{\boldsymbol{x}} = \underset{\boldsymbol{x}\in\mathbb{Z}^{N_\mathrm{T}}}{\operatorname{argmin}} \|\boldsymbol{y} - \alpha\boldsymbol{H}\boldsymbol{x}\| , \tag{11}$$

which is equivalent to finding the point in the lattice having the basis consisting of the column vectors of $\alpha\boldsymbol{H}$ that is closest to $\boldsymbol{y}$, i.e., $\operatorname{argmin}_{\{\boldsymbol{x}';\boldsymbol{x}'=\alpha\boldsymbol{H}\boldsymbol{x},\,\boldsymbol{x}\in\mathbb{Z}^{N_\mathrm{T}}\}} \|\boldsymbol{y} - \boldsymbol{x}'\|$. The basis matrix $\alpha\boldsymbol{H}$ does not contain a unique set of basis vectors for the particular lattice $\{\boldsymbol{x}';\boldsymbol{x}' = \alpha\boldsymbol{H}\boldsymbol{x},\, \boldsymbol{x}\in\mathbb{Z}^{N_\mathrm{T}}\}$. Indeed, there is an infinite number of different basis vectors that span the same lattice; some yield well-conditioned basis matrices and some do not, see Fig. 3. By finding an appropriate transformation matrix $\boldsymbol{T}$, an ill-conditioned realization of the problem in (11) can be transformed into an equivalent well-conditioned problem. The $\boldsymbol{T}$ matrix must be invertible and endomorphic (maps a space onto itself) with respect to the lattice spanned by $\alpha\boldsymbol{H}$. Hence,

$$\begin{aligned}\hat{\boldsymbol{x}} &= \underset{\boldsymbol{x}\in\mathbb{Z}^{N_\mathrm{T}}}{\operatorname{argmin}} \|\boldsymbol{y} - \alpha\boldsymbol{H}\boldsymbol{x}\| = \underset{\boldsymbol{x}\in\mathbb{Z}^{N_\mathrm{T}}}{\operatorname{argmin}} \left\|\boldsymbol{y} - \alpha\boldsymbol{H}\boldsymbol{T}\boldsymbol{T}^{-1}\boldsymbol{x}\right\| \\ &= \boldsymbol{T}\,\underset{\boldsymbol{x}\in\mathbb{Z}^{N_\mathrm{T}}}{\operatorname{argmin}} \|\boldsymbol{y} - \alpha\boldsymbol{H}\boldsymbol{T}\boldsymbol{x}\| .\end{aligned} \tag{12}$$

With a well-conditioned problem (matrix $\alpha\boldsymbol{H}\boldsymbol{T}$), finding the closest lattice point is easy. The main difficulty has been shifted from finding the closest point in a lattice to finding an appropriate transformation matrix $\boldsymbol{T}$.

There are certainly many ways to find a good matrix $\boldsymbol{T}$: some are computationally demanding and some are not. One of the most well-known and computationally cheap algorithms is the LLL lattice reduction (LLL-LR) algorithm introduced by A.K. Lenstra, H.W. Lenstra, and L. Lovász in [40]. This algorithm has, as many other LR algorithms do due to their iterative nature, a variable complexity. For more detail on LR aided MIMO detection, see [7] and the references therein.

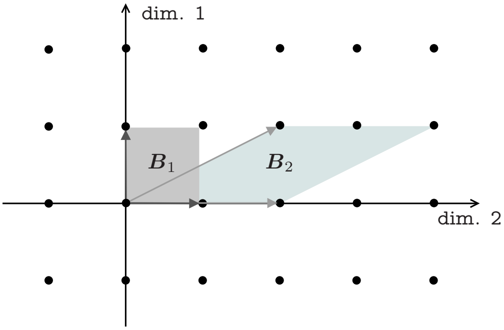

Figure 3: Square lattice generated by two different bases. The two bases are given by the column vectors in $\boldsymbol{B}_1 = \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}$ and $\boldsymbol{B}_2 = \begin{bmatrix}1 & 0\\2 & 2\end{bmatrix}$, respectively. The basis matrix $\boldsymbol{B}_1$ is obviously much better conditioned than $\boldsymbol{B}_2$ and thus finding a closest lattice point in the basis given by $\boldsymbol{B}_1$ is much simpler than in that given by $\boldsymbol{B}_2$.

1.1.7 Semidefinite-Relaxation Detection

The main idea behind the semidefinite-relaxation (SDR) approach to hard-decision MIMO detection [41–44] is to first pose the finite constellation requirement as a low-rank (in this case rank one) constraint on a matrix whose diagonals belong to a finite constellation. These two constraints on this matrix are then relaxed to a positive semidefinite constraint, which makes the resulting problem convex and enables the use of semidefinite programming to solve it. More specifically, we can rewrite

$$\begin{aligned}\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{s}\|^2 &= \boldsymbol{s}^T\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{s} - 2\boldsymbol{y}^T\boldsymbol{H}\boldsymbol{s} + \|\boldsymbol{y}\|^2\\ &= \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \|\boldsymbol{y}\|^2,\end{aligned} \tag{13}$$

where $\boldsymbol{x} \triangleq \begin{bmatrix}\boldsymbol{s}\\1\end{bmatrix}$ and $\boldsymbol{Q} \triangleq \begin{bmatrix}\boldsymbol{H}^T\boldsymbol{H} & -\boldsymbol{H}^T\boldsymbol{y}\\ -\boldsymbol{y}^T\boldsymbol{H} & 0\end{bmatrix}$.

Then, SDR utilizes $\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} = \operatorname{tr}\left(\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x}\right) = \operatorname{tr}\left(\boldsymbol{Q}\boldsymbol{x}\boldsymbol{x}^T\right)$, which lets the MIMO detection problem, $\min \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{s}\|^2$ subject to $\boldsymbol{s} \in \mathcal{S}^{N_\mathrm{T}}$, to be equivalently written as

$$\underset{\boldsymbol{x},\boldsymbol{X}}{\operatorname{argmin}} \operatorname{tr}\left(\boldsymbol{Q}\boldsymbol{X}\right) \tag{14}$$

$$\text{subject to } \boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^T, \tag{15}$$

$$X_{i,i} \in \{s^2 : s \in \mathcal{S}\}, i = 1, \dots, N_\mathrm{T} - 1, \tag{16}$$

$$X_{N_\mathrm{T},N_\mathrm{T}} = 1. \tag{17}$$

By relaxing the first constraint, i.e., (15), to $\boldsymbol{X} \succeq 0$, the problem is simplified to a convex optimization problem, which can be solved using techniques from semidefinite programming. Examples of such techniques are interior point methods [45]. These relaxations may not lead to a solution that belongs to the finite constellation and some extra steps are needed in order to find a feasible $\boldsymbol{s} \in \mathcal{S}^{N_\mathrm{T}}$ that is close to $\boldsymbol{X}$. This issue is part of the relaxations made so far, which induces errors in the solution. Another disadvantage is that interior point methods do not have a deterministic search complexity, which can vary for different realizations of the problem. However, interior point methods (and hence SDR) converge typically very fast to a solution that is close to the optimal one and can therefore be terminated rather early. For instance, it has been shown in [46] that SDR can achieve high performance at high signal-to-noise ratio (SNR).

1.1.8 Likelihood Ascent Search

The likelihood ascent search (LAS) method, which performs a very simple procedure, had its main application within image restoration [47–49] and is based on Hopfield neural networks. In [50, 51], LAS was presented in the context of CDMA detection. In, [23], the same algorithm was presented in the context of large (and square) MIMO detection. What LAS does is essentially a bit-flipping procedure. At a given symbol vector, say $\hat{\boldsymbol{s}}$, LAS changes one bit at a time, say bit i, and checks whether the likelihood is increased or decreased. In our system model, this is equivalent to evaluating whether the following is satisfied or not,

$$\|\boldsymbol{y} - \boldsymbol{H}\hat{\boldsymbol{s}}\|^2 > \|\boldsymbol{y} - \boldsymbol{H}(\hat{\boldsymbol{s}} \oplus \boldsymbol{u}_i)\|^2 , \tag{18}$$

where $\boldsymbol{u}_i$ has only the ith bit equal to binary one and $\oplus$ is the bit-wise "exclusive or" logical operator. If the likelihood is increased, i.e., the inequality in (18) is satisfied, then the algorithm moves to the particular symbol vector $\hat{\boldsymbol{s}} \oplus \boldsymbol{u}_i$. If it is not increased, then a different bit in $\hat{\boldsymbol{s}}$ is flipped and so on. If none of the bit-flips yield a likelihood ascent, the algorithm is terminated and the current symbol vector is returned as the solution. The LAS procedure has shown to converge rather fast to a solution [50, 51]. However, this solution is only guaranteed to be a local maximum (of the likelihood) and the global optimum may not be found. This is something that methods like Tabu search, see Sec. 1.1.9, try to fix.

There are several variations of the LAS algorithm. An evident extension is to choose properly in which order to flip the bits to avoid getting stuck in a "wrong" local minimum and to converge fast to a global minimum [52]. Another extension is to let the number of bits flipped concurrently vary, and not flip just one bit at a time [52]. The complexity of the LAS algorithm is not fixed and it highly depends on the realization of the channel matrix and the noise.

1.1.9 Tabu Search

The tabu search (TS) methodology was first developed in [53] and later refined in [54] as a general mathematical tool to solve combinatorial optimization problems. In summary, a TS algorithm performs a search in a neigbourhood of the current point (similar to LAS for instance) while keeping track of a finite list of so-called tabu points. The algorithm makes sure not to visit such tabu points and that way tries to avoid searching in cycles and getting stuck in local optima, which other algorithms such as LAS do not. Different implementations construct the tabu list in different ways. One common approach is to store the recently visited points in order not to revisit them again. Other methods apply a varying tabu list size in order to make the procedure adaptive and more efficient. For long list sizes, getting stuck in local optima is less likely than for short lists at the expense of a higher search complexity since the tabu list will be searched at each step and vice versa for the opposite case. One example of such an implementation is the reactive tabu search (RTS) [55]. The RTS method increases the tabu list size when none (or very few) points in the current neighborhood are found in the list. Analogously, it decreases the list size if all (or most) points in the neighborhood are found in the list. In addition, the RTS method adds an escape strategy that restarts the algorithm at a different point to further reduce the risk of cyclic searches. This is typically done when the tabu list mechanism is not sufficient, which happens for instance, if the tabu list and the neighborhood size becomes prohibitively large but still not large enough to avoid cycles. In [55], the escape procedure is initiated if in many consecutive iterations, the neighborhood points are found in the tabu list.

Particularly for MIMO detection, the TS methodology was applied in [56, 57]. One difficulty there, as in other tabu search problems, was to find a proper neighborhood structure since this has a direct impact on the effectivness of the search. There the neighborhood was chosen to be the set of points within a certain predetermined Hamming distance (computed with the operator $d_{\mathrm{hamm}}(\cdot,\cdot)$) from the current point, i.e.,

$$\mathcal{N}(\boldsymbol{s}) \triangleq \big\{\boldsymbol{x} \in \mathcal{S}^{N_{\mathrm{T}}}; d_{\mathrm{hamm}}(\boldsymbol{s},\boldsymbol{x}) \leq \delta\big\}. \tag{19}$$

The TS type of methods are ad-hoc schemes whose convergence cannot be guaranteed. Their complexity is also difficult to determine since it varies from realization to realization of the problem. However, many empirical studies have shown that the applicability is high and these methods can outperform, in terms of average complexity versus accuracy, many other MIMO detection methods, especially in cases when the dimensions of the MIMO detection problem are large in number [57].

1.2 Soft MIMO Detection

The optimal soft information desired by the channel decoder is the a posteriori log-likelihood ratio

$$l(b_i|\boldsymbol{y}) \triangleq \log\left(\frac{P(b_i = 1|\boldsymbol{y})}{P(b_i = 0|\boldsymbol{y})}\right), \tag{20}$$

where b_i is the i:th bit of the transmitted vector $\boldsymbol{s}$. The quantity in (20) tells us, given some $\boldsymbol{y}$, how likely it is that the i:th bit of $\boldsymbol{s}$ is equal to zero or one, respectively. This quantity is typically referred to as making a soft decision on the bit b_i, also referred to as the bit of interest throughout this text. By using Bayes' rule, performing marginalization over all bits except the i:th bit, and assuming uniform a priori probabilities, the log-likelihood ratio (LLR) becomes

$$l(b_i|\boldsymbol{y}) = \log\left(\frac{\sum_{\boldsymbol{s}:b_i(\boldsymbol{s})=1}\exp\left(-\frac{1}{N_0}\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{s}\|^2\right)}{\sum_{\boldsymbol{s}:b_i(\boldsymbol{s})=0}\exp\left(-\frac{1}{N_0}\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{s}\|^2\right)}\right). \tag{21}$$

where the notation $\sum_{\boldsymbol{s}:b_i(\boldsymbol{s})=x}$ means the sum over all possible vectors $\boldsymbol{s} \in \mathcal{S}^{N_\mathrm{T}}$ for which the i:th bit is equal to x. In (21), there are $|\mathcal{S}|^{N_\mathrm{T}}$ terms that need to be evaluated and added, which again results in an exponential complexity in N_T. This is a significant obstacle when it comes to realizing such an algorithm. Therefore, many different approximate methods have been developed.

1.2.1 Max-Log Detection

A very good approximation of (21) is the so called max-log approximation where the sums in both the numerator and the denominator are replaced by their corresponding largest term. Why max-log is a very good approximation follows from the fact that

$$\log(e^a + e^b) = \max(a,b) + \log(1 + e^{-|a-b|}) \approx \max(a,b) + e^{-|a-b|} \approx \max(a,b), \tag{22}$$

where the approximations are very close to equalities when $e^{-|a-b|}$ is very small. The quantity $e^{-|a-b|}$ is generally very small in the relevant MIMO detection problems.

The max-log approximation of (21) is

$$l(b_i|\boldsymbol{y}) \approx \frac{1}{N_0}\Big(\min_{\boldsymbol{s}:b_i(\boldsymbol{s})=0}\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{s}\|^2 - \min_{\boldsymbol{s}:b_i(\boldsymbol{s})=1}\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{s}\|^2\Big). \tag{23}$$

The complexity of this approximation remains exponential in the number N_T since a full enumeration of all the $|\mathcal{S}|^{N_\mathrm{T}}$ candidates is still performed. With this approximation, we end up in taking hard decisions, i.e., solving constrained minimization problems as written in (4), in order to obtain soft decisions. Therefore, any hard detector of choice can produce soft values via this approximation. In particular, there are variations of the SD algorithm that utilize this in order to produce soft values [27, 58–60]. In [58], a list of a fixed number of solution candidates is stored during the SD search through the tree in Fig. 2. Depending on which paths the SD takes, it might find the candidates that are the optimal solutions to the individual minimization problems in (23) (but it also might not). In [60] and [27], smart book-keeping techniques are used in order to find the optimal solutions to the individual minimization problems in (23). For more details, see Sec. 2.2 in Paper C.

1.2.2 Soft ZF

A very crude approximation to (21), as in the hard decision case, is the soft zero forcing approximation. Similarly to hard decision ZF, soft decision ZF first decouples the symbols in $\boldsymbol{s}$ via

$$\boldsymbol{z} = \underbrace{(\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T}_{\triangleq \boldsymbol{H}^\dagger}\boldsymbol{y} = \boldsymbol{s} + \boldsymbol{H}^\dagger \boldsymbol{e} = \boldsymbol{s} + \boldsymbol{n}, \quad \boldsymbol{n} \sim \mathcal{N}(\boldsymbol{0}, \tfrac{N_0}{2}(\boldsymbol{H}^T\boldsymbol{H})^{-1}).$$

The decoupled vector model is split up in N_T scalar models

$$z_k = s_k + n_k, \quad k = 1, \ldots, N_\mathrm{T}. \tag{24}$$

Then, by approximating the noise terms n_k as uncorrelated for all $k = 1, \ldots, N_\mathrm{T}$, soft decisions are computed on the bits in s_k (independently of the bits in $s_\ell \forall \ell \neq k$) by applying (21) on the scalar model in (24). The approximation from the exact LLR becomes explicit by rewriting (21) as

$$l(b_i|\boldsymbol{y}) = \log\left(\frac{\sum_{\boldsymbol{s}:b_i(\boldsymbol{s})=1} \exp\left(-\frac{1}{N_0}\big(\boldsymbol{H}^\dagger\boldsymbol{y} - \boldsymbol{s}\big)^T\boldsymbol{H}^T\boldsymbol{H}\big(\boldsymbol{H}^\dagger\boldsymbol{y} - \boldsymbol{s}\big)\right)}{\sum_{\boldsymbol{s}:b_i(\boldsymbol{s})=0} \exp\left(-\frac{1}{N_0}\big(\boldsymbol{H}^\dagger\boldsymbol{y} - \boldsymbol{s}\big)^T\boldsymbol{H}^T\boldsymbol{H}\big(\boldsymbol{H}^\dagger\boldsymbol{y} - \boldsymbol{s}\big)\right)}\right). \tag{25}$$

The approximation step the ZF method makes from exact LLR is that it approximates $\boldsymbol{H}^T\boldsymbol{H}$ in (25) as diagonal by keeping only the diagonal elements in the effective covariance matrix $\frac{N_0}{2}(\boldsymbol{H}^T\boldsymbol{H})^{-1}$. The complexity of this algorithm is of the same order of magnitude as in the hard decision case.

1.2.3 Soft MMSE

The soft MMSE method applies a linear MMSE filter on the received data, which consists of the ZF filter with a regularization term inside the inverse that makes the inverse a bit numerically friendlier to compute. To derive the MMSE filter, the signal vector $\boldsymbol{s}$ is assumed to be random with an associated prior distribution (Gaussian with zero-mean and identity covariance matrix). The signal vector is also typically independent of the channel matrix and the received noise. Now, following the construction of the linear MMSE filter [61], the derivations lead to

$$\mathbb{E}\{\boldsymbol{s}\boldsymbol{y}^T\}\left(\mathbb{E}\{\boldsymbol{y}\boldsymbol{y}^T\}\right)^{-1} = \boldsymbol{H}^T(\tfrac{N_0}{2}\boldsymbol{I}+\boldsymbol{H}\boldsymbol{H}^T)^{-1} = (\tfrac{N_0}{2}\boldsymbol{I}+\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T. \qquad (26)$$

Then, similar to soft ZF, before the computation of soft decisions, the effective noise, which is the noise after the MMSE filter is applied, is approximated as uncorrelated. Hence, this enables the computation of the soft decisions for one symbol (element in the symbol vector) at the time independent of the adjacent symbols. One major advantage of the MMSE filter over the ZF filter is that it can be used both for underdetermined ($N_\mathrm{T} > N_\mathrm{R}$) and overdetermined ($N_\mathrm{T} < N_\mathrm{R}$) systems thanks to the last equality in (26). A simple proof of this equality is to apply $\frac{N_0}{2}\boldsymbol{I}+\boldsymbol{H}^T\boldsymbol{H}$ from the left and $\frac{N_0}{2}\boldsymbol{I}+\boldsymbol{H}\boldsymbol{H}^T$ from the right on both the left-hand and right-hand side of the last equality in (26). Hence,

$$(\tfrac{N_0}{2}\boldsymbol{I}+\boldsymbol{H}^T\boldsymbol{H})\boldsymbol{H}^T = \tfrac{N_0}{2}\boldsymbol{H}^T+\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{H}^T = \boldsymbol{H}^T(\tfrac{N_0}{2}\boldsymbol{I}+\boldsymbol{H}\boldsymbol{H}^T). \qquad (27)$$

■

Another approach to arrive at the same procedure is to consider one symbol (and the containing bits), say $\bar{s}$, as the signal of interest and the rest $\tilde{\boldsymbol{s}}$ as Gaussian interference. Then, the MIMO model can be written as

$$\boldsymbol{y} \;=\; \bar{\boldsymbol{h}}\bar{s} \;+\; \underbrace{\widetilde{\boldsymbol{H}}\tilde{\boldsymbol{s}} \;+\; \boldsymbol{e}}_{\text{interference+noise}} \;\approx\; \bar{\boldsymbol{h}}\bar{s} \;+\; \boldsymbol{n}, \qquad (28)$$

where $\boldsymbol{n} \sim \mathcal{N}\big(\boldsymbol{0},(\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^T+\frac{N_0}{2}\boldsymbol{I})^{-1}\big)$. Computing the soft values using the resulting single-input multiple-output model is equivalent to computing the soft values with the procedure explained first.

1.2.4 Soft MMSE with Parallel Interference Cancellation

By analyzing the soft MMSE procedure as written in (28), one can see that the interfering term introduces a bias once $\boldsymbol{y}$ is given since the then received symbols are not zero-mean. This is what the methods based on parallel interference cancellation (PIC), such as MMSE-PIC, try to reduce by simply approximating the mean of the bias and then subtracting it. Hence

$$\boldsymbol{y}-\widetilde{\boldsymbol{H}}\mathbb{E}\{\tilde{s}|\boldsymbol{y}\} = \bar{\boldsymbol{h}}\bar{s}+\widetilde{\boldsymbol{H}}\left(\tilde{s}-\mathbb{E}\{\tilde{s}|\boldsymbol{y}\}\right)+\boldsymbol{e} \approx \bar{\boldsymbol{h}}\bar{s}+\tilde{\boldsymbol{n}}, \qquad (29)$$

where the approximation stems from the assumption that the interference is Gaussian, i.e., saying that $\tilde{\boldsymbol{n}}$ is Gaussian. The elements in $\tilde{\boldsymbol{n}}$ have smaller variances than the elements in $\boldsymbol{n}$ in (28). The non-trivial part is to compute $\mathbb{E}\{\tilde{\boldsymbol{s}}|\boldsymbol{y}\}$ since the required marginal distributions are as numerically burdensome to compute as the exact LLRs. Therefore, approximations are required. One very accurate, although not very computationally efficient approximation, is to use the estimated marginals that are delivered by the outer decoder. This is commonly done by iterating soft information between the soft MMSE-PIC detector and the outer decoder.

In [21], this procedure was further generalized to enable the useful signal to incorporate more than one symbol. This was done by using the partitioned model in (8)

$$\boldsymbol{y} = \bar{\boldsymbol{H}}\bar{\boldsymbol{s}} + \underbrace{\widetilde{\boldsymbol{H}}\tilde{\boldsymbol{s}} + \boldsymbol{e}}_{\text{interference+noise}} \approx \bar{\boldsymbol{H}}\bar{\boldsymbol{s}} + \boldsymbol{n}, \tag{30}$$

and considering $\bar{\boldsymbol{s}}$ as the useful signal instead of just $\bar{s}$. A soft value of a bit of interest is then computed by marginalizing out all the bits in $\bar{\boldsymbol{s}}$ except the bit of interest.

1.2.5 Partial Marginalization

The soft-output via partial marginalization (PM) method in [6] offers a trade-off between exact and approximate computation of (21), via a parameter $r \in \{0, \ldots, N_{\mathrm{T}} - 1\}$. The PM method is an extension of the hard decision detector FCSD to the case of pure soft decision detection. It retains the highly parallelizable structure. The slightly modified version in [62] of the method in [6] is presented here. It is simpler than that in [6] but without comprising performance.

Consider again the partitioned model in (8), where now $\bar{\boldsymbol{s}} \in \mathcal{S}^{r+1}$ contains the i:th bit in the original symbol vector $\boldsymbol{s}$. How the partitioning is chosen (8) is analogous to that in FCSD, i.e., the condition number of the matrix $\widetilde{\boldsymbol{H}}$ is aimed to be minimized. The PM method implements a two-step approximation of (21). More specifically, in the first step it approximates the sums of (21) that correspond to $\tilde{\boldsymbol{s}}$ with a maximization,

$$l(b_i|\boldsymbol{y}) \approx \log\left(\frac{\sum\limits_{\bar{\boldsymbol{s}}:b_i(\boldsymbol{s})=1} \max\limits_{\tilde{\boldsymbol{s}}} \exp\Big(-\frac{1}{N_0}\,\|\boldsymbol{y} - \bar{\boldsymbol{H}}\bar{\boldsymbol{s}} - \widetilde{\boldsymbol{H}}\tilde{\boldsymbol{s}}\|^2\Big)}{\sum\limits_{\bar{\boldsymbol{s}}:b_i(\boldsymbol{s})=0} \max\limits_{\tilde{\boldsymbol{s}}} \exp\Big(-\frac{1}{N_0}\,\|\boldsymbol{y} - \bar{\boldsymbol{H}}\bar{\boldsymbol{s}} - \widetilde{\boldsymbol{H}}\tilde{\boldsymbol{s}}\|^2\Big)}\right). \tag{31}$$

In the second step, the maximization in (31) is approximated, as in the FCSD method, with a simple hard detector such as the ZF-DF detector [6]. Why this approximation is reasonable follows from the discussions in Sec. 1.1.4 and Sec. 1.2.1. Notably, PM performs ZF-DF aided max-log detection for $r = 0$ and computes the exact LLR values (as defined by (21)) for $r = N_{\mathrm{T}} - 1$.

1.3 Summary

1.3.1 Different MIMO Settings

There are many different smart techniques to solve a MIMO detection problem with a certain accuracy and computational complexity. However, which method to choose is not always easy to decide. Some of the simple methods work well if the channel conditions are favorable, i.e., yielding close-to-orthogonal columns in $\boldsymbol{H}$, and using any of the more advanced and computationally heavy techniques becomes a waste of computational resources. However, if one uses simple techniques when the channel conditions are not favorable, the performance losses may be grave and in the worst case outside the tolerance threshold.

Further, in medium-sized MIMO systems, for which most of the methods above are developed for, LAS which is primarily developed for large-sized MIMO does not work so well. For large-sized MIMO systems with square structure, LAS and Tabu Search methods work quite well. In very large MIMO systems that are highly overdetermined [63], i.e., a base-station with roughly ten times more receiving antennas than the number of transmitting users (typically equipped with very few antennas each), the simplest algorithms such as MMSE and ZF become optimal. The focus in such systems is therefore not to approximate the exact LLR procedure, as opposed to in medium-sized MIMO, but rather these linear detection techniques. That said, a shift in focus is imposed in order to further reduce the number of operations performed. Even more distinguished is the focus on where the operations are performed, distributively or centrally. Since these types of systems are sought to have many antennas at the base station, maybe hundreds or even thousands, keeping the antenna units as modular and independent as possible is highly desirable. The main reasons are that the computational burden (which in such high-dimensional systems can quickly become intolerable) needs to be evenly distributed, and that if one unit breaks down no other is affected. It is however not possible to have completely distributed processing, since the information from the different base-station antennas needs to be combined at some point—information fusion is the main strength of the MIMO technology. Nevertheless, there are ways to distribute some of the computational burden to the antenna units, in order to be administered locally, before the processed information reaches the central processing unit.

1.3.2 MIMO Detection Timeline

Here, the main contributions in literature related to MIMO detection are summarized pictorially. In Fig. 4, the horizontal axis shows the time from the beginning of CDMA detection in the 90's up to now with large-sized MIMO detection. The vertical axis distinguishes between the four main categories of detectors: soft/hard decision detectors with fixed/random computational complexity. Since the field of MIMO detection has grown extensively over the years, with a huge number of publications, the presented list of references is far from exhaustive. The shown references are chosen based on their significance and their breadth. Several of the methods discussed in these references have been published in earlier works.

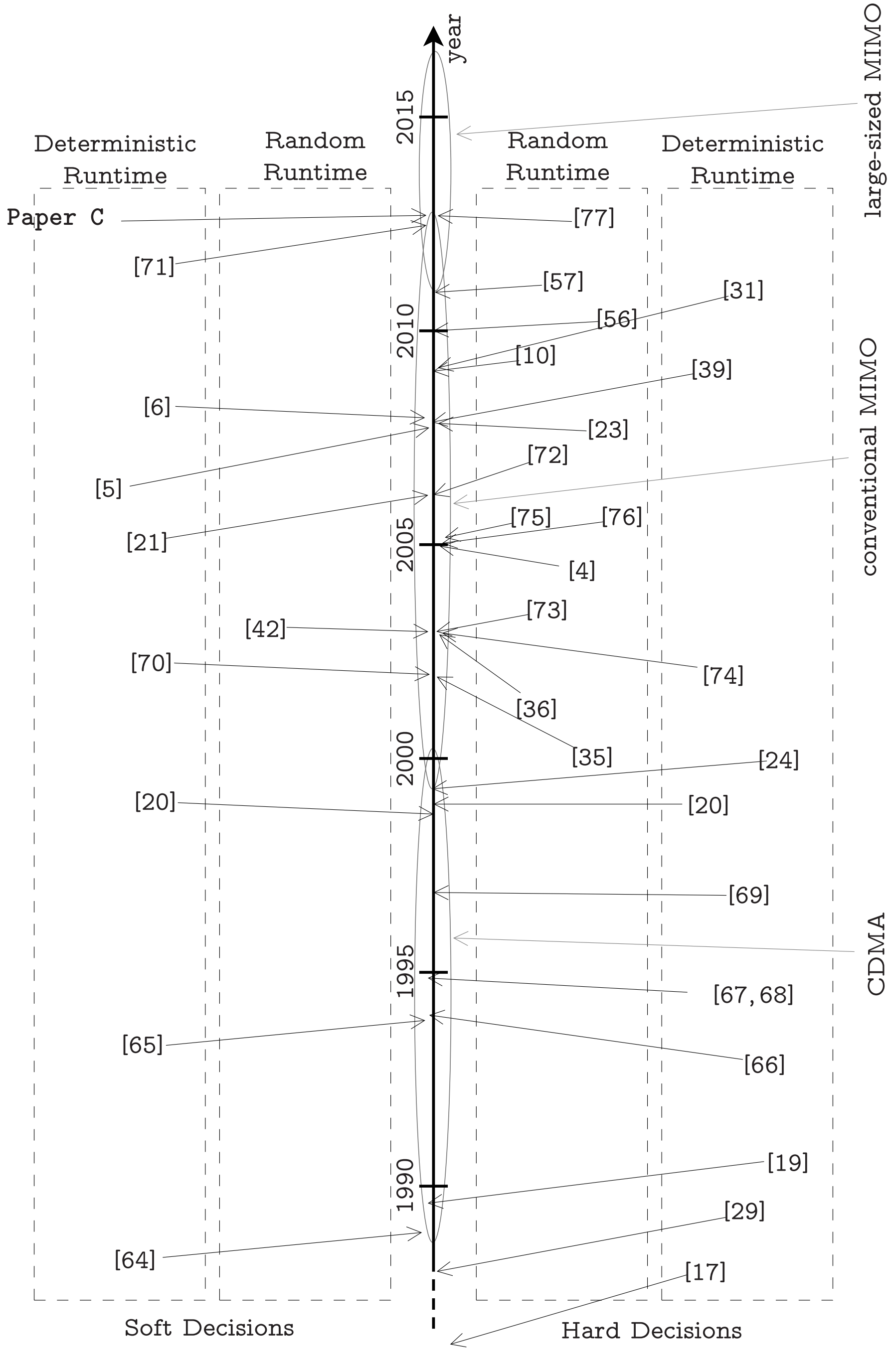

Figure 4: Timeline of relevant litterature related to MIMO detection. The references are sorted in four groups of detection methods: those with soft/hard decisions and those with random/predictable complexity (run time). The timeline marks the three important phases of the development of MIMO detection: CDMA, conventional MIMO, and very large MIMO.

Chapter 2

Contributions of the Thesis

This thesis addresses the general MIMO detection problem in two ways. Part III treats a development of new and more efficient detection techniques for the different MIMO settings. Part II[1]is related to adaptive means of allocating computational resources at the receiver during the signal separation (detection) process. The main idea is to utilize already proposed detection algorithms and their advantages versus disadvantages in an adaptive manner which is possible since there is always a trade-off between performance and complexity in MIMO detection in a coded system. For instance, for well-conditioned channels, low-complexity detection methods are often sufficiently accurate. In such cases, performing computationally very expensive optimal detection would be a waste of computational power. A general framework is presented, which is not specific to certain detection algorithms nor scenarios. The main ideas of this part are exemplified with the PM method, but can be applied to any MIMO detector of choice. The techniques that are proposed facilitate fixed complexity detection and incorporate the possibility of tuning the trade-off between complexity and detection accuracy with arbitrarily fine (discrete) resolution. The work in [78–83] can be thought of as special cases of the general framework presented here.

The main content of this thesis consists of three published papers along with some additional unpublished material. The first paper aims to address the allocation problem directly and as a result brings out new interesting problems, such as identifying quantities (measures) on which the adaptive allocation should be based. One of the fundamental and promising measures proposed there requires the knowledge of the probability distribution of the detector outputs, which is difficult to acquire. In the second paper, this difficulty is addressed and a good approximate distribution is found. Additionally, supplementary unpublished simulation results are presented, which conclude Part II. The third paper

[1] Much of Part II has been published as part of my Licentiate thesis [84].

begins Part III by presenting a new MIMO detection technique which brings unprecedented performance. This detection technique has shown to be applicable to and well-suited for a wide range of MIMO detection problems. To this paper, supplementary unpublished simulations results are added as additional support to some of the interesting conclusions that were drawn. The last paper, which consists of unpublished material, treats MIMO detection for large-sized systems. The focus relative to conventional medium-sized MIMO is a bit different and the work brings to light some insights that have not been brought up before within the large-sized MIMO context. This paper completes Part III and the thesis itself.

2.1 Adaptive Computational Resource Allocation

2.1.1 Related Work

In recent years, a limited amount of litterature has been produced that considers adaptive detection. Some general ideas were outlined in [85] and some more specific aspects of the problem were addressed in [78–82].

The work of [78] is specific to the SD algorithm where the idea is to let the algorithm decide whether it requires more or less processing power to solve a detection problem. This is possible due to the fact that the SD algorithm has a variable complexity that depends on the channel conditions. For ill-conditioned channels, it requires more time (processing power) to find the solution to the detection problem than for well-conditioned channels. In [78], a maximum allowed per-received-data-packet as well as a per-detection-problem time (processing power) limit is set. The SD algorithm is then executed to solve the detection problems as they appear without violating the specified time limits. A received data packet generally contains multiple detection problems. The first detection problems that fit within the predetermined time limits are solved, and the problems that do not are simply ignored.

In [79], the ideas of [78] are extended to adaptively vary the user parameter (sphere radius) of the SD algorithm in an iterative decoding setting; in iterative decoding, the detector and decoder interchange soft information in several iterative steps in order to improve the overall performance. For SD, using a larger sphere radius means better performance with higher complexity and smaller means vice versa. The idea in [79] is to set a minimum accuracy threshold for each detected bit and use that threshold to determine the initial smallest allowed sphere radius that will assure a certain accuracy of each detected bit. For some bits, it is computationally more expensive to meet the accuracy threshold than for others. Then in subsequent iterations, the sphere radius is decreased for those bits that pass well over the accuracy threshold in order not to spend unnecessary computational power. Since SD has a variable complexity, so do the techniques in [78] and [79], which is not a desirable property due to the necessity of over-dimensioned hardware; the utilization will not be full whenever the channel conditions are good and the SD algorithm finishes early. Additionally, the techniques in [78] and [79] do not give priority to the detection problems that are most "beneficial" to solve, which is something that can, as we will see in this thesis, yield large performance gains.

The approach in [80] tries to predict whether a detection problem is simple or difficult. An approximate bit-error-rate expression is derived given the current channel conditions for the ZF detector and for the optimal detector. This measure is then used to predict whether it is necessary to use the optimal detector or if it is sufficient to use the simple ZF detector on different detection problems. The aim is to reduce processing power without violating a predetermined minimum bit error rate. This approach is crude in the sense that it performs either computationally cheap ZF detection or expensive optimal detection, but nothing in between. In [86], they adapt and solve a knapsack problem that switches between SD and ZF to save computational resources.

A more sophisticated approach is given in [81], which is specific to the FCSD algorithm. Recall that the FCSD algorithm performs ZF-DF and ML detection for the user parameter $r = 0$ and $r = N_\mathrm{T} - 1$, respectively. This multi-mode detector has been successfully implemented in hardware [71]. The FCSD user parameter is adjusted beforehand using a rule that is based on the estimation variance such that a predetermined tolerance level is met with the aim to reduce the required computational power. This procedure makes the FCSD algorithm adapt to the effective channel conditions in a less crude manner than in [80]. Similarly in [82], such a procedure is performed but instead using a rule that is based on the condition number of the effective channel matrix. The better the conditioning is, the smaller user-parameter r is used.

In [83] a K-best extension of the SD method is employed in an adaptive manner. The K-best detector has a parameter $\mathcal{K}$ that sets the maximum number of paths to be traversed at each level of the tree-search. By varying this parameter, the tradeoff between performance and complexity can be controlled. In [83], a bit-error-rate approximation is used in order to predict what $\mathcal{K}$ should be used with the objective to minimize the spent power directly instead of indirectly, via for instance minimization of the number of operations.

2.1.2 Papers

Paper A Allocation of Computational Resources for Soft MIMO Detection

Authored by Mirsad Čirkić, Daniel Persson, and Erik G. Larsson.

Published in the IEEE Journal on Selected Topics in Signal Processing, Special Issue on Soft MIMO Detection, Dec., 2011. The work is mainly based on the conference papers in [87,88].

The work considers soft MIMO detection for the case of block fading. That is, the transmitted codeword spans over several independent channel realizations and several instances of the detection problem must be solved for each such realization. It develops methods that adaptively allocate computational resources to the detection problems of each channel realization, under a total per-codeword complexity constraint. The main results are a formulation of the problem as a mathematical optimization problem with a well-defined objective function and constraints, and algorithms that solve this optimization problem efficiently computationally.

Paper B Approximating the LLR Distribution for a Class of Soft-Output MIMO Detectors

Authored by Mirsad Čirkić, Daniel Persson, Jan-Åke Larsson, and Erik G. Larsson.

Published in the IEEE Transactions on Signal Processing, Dec., 2012. The work is mainly based on the conference paper [89].

This paper presents approximations of the LLR distribution for a class of fixed-complexity soft-output MIMO detectors, such as the optimal soft detector and the soft-output via partial marginalization detector. More specifically, in a MIMO AWGN setting, we approximate the LLR distribution conditioned on the transmitted signal and the channel matrix with a Gaussian mixture model (GMM). Our main results consist of an analytical expression of the GMM model (including the number of modes and their corresponding parameters) and a proof that, in the limit of high SNR, this LLR distribution converges in probability towards a unique Gaussian distribution.

2.1.3 Future Work

The work related to such adaptive allocation of computational resources has closed some open problems but in doing so, it has opened others. Finding efficient algorithms that perform the adaptive allocation was not the difficult part in this work. This part has been mainly resolved with several efficient sub-optimal and optimal algorithms. The difficult part that still contains open questions is to find good quantitative accuracy measures that predict the detector accuracy given the effective channel conditions. The results in Paper A indicate that there exist measures that can yield better performance than what is presented. Thus, finding such accuracy measures is definitely a future work worth considering.

So far, Part II has not considered detection with iterative decoding. It is nevertheless fully possible to develop such extensions but it would require more care when determining the detector accuracy measures. Similarly, higher order constellations are not considered neither and implementing such extensions are straightforward but it requires finding appropriate detector accuracy measures as well.

The fundamental ideas of this work can be extended further to other applications, such as energy efficient detection [83]. One could control the processing frequency [90] and or adapt the signal sampling in order to save energy. Energy efficient detection can be performed by controlling the processing frequency of the different computational units that perform detection. Easy detection problems that can be solved using computationally cheap detection methods with processing units that work at a lower frequency. On the contrary, difficult detection problems that require computationally heavy and advanced detection methods, can be processed with units using higher frequency in order for all problems (simple and difficult) to be solved during the same amount of time. This is beneficial since, as pointed out in [90], processing units performing the same task at a lower frequency in a longer time interval require less energy than at a higher frequency in a shorter time interval. In addition to that, the hardware utilization will be very high since none of the computational units will wait for one another, which is a very desirable property.

Adaptive signal sampling can be utilized to control the sampling frequency at each receive antenna in order to simplify the computations without comprising performance. Higher resolution arithmetics require more sophisticated hardware, more energy to acquire, and more computational power to process [91]. Now, this technique would be beneficial since some antennas would in one extreme case only see noise for which high resolution sampling would be a waste of energy and computational resources. In such an extreme case, one could just simply shut down that antenna and the chain of units that follows. The technique where antennas are shut down due to bad channel conditions is largely known as antenna selection [92]. This technique can be extended in a more delicate manner by adjusting the number of significant bits at each antenna based on the effective channel conditions (information throughput) at each antenna.

2.2 Soft MIMO Detection

This part of the thesis includes two contributions that aim to solve the soft decision MIMO detection problem without employing the max-log approximation. The first contribution, i.e., Paper C, proposes a new method for soft MIMO detection. This method has shown to have an unprecedented complexity versus performance trade-off, both for medium-sized and large-sized MIMO systems. To Paper C, extended simulation results for large-sized MIMO (with square structure) are added. These results support the that the max-log approximation can have worse performance than simple linear MMSE detection when the antenna arrays are increased in size. In particular the method that Paper C proposes, which consists of a smart extension of the MMSE method, closes the gap even more to exact LLR. The second contribution, i.e., Paper D, which utilizes the techniques in Paper C, includes new algorithmic steps specifically applicable for large-sized MIMO systems as in [63].

2.2.1 Related Work

Much of the related work is discussed in Chap. 1. It is worth noting that a huge part of the literature considers hard detection and tree search problems in particular. One particular reason for this is the very tight max-log approximation in Sec. 1.2.1, which enables any hard decision detector to produce soft decisions. There are not many methods that have the main focus to address the problem of producing soft decisions without the need to apply the max-log approximation. A few examples are those in Sec. 1.2.3, 1.2.4, 1.2.5, and Paper C.

2.2.2 Papers

Paper C SUMIS: Near-Optimal Soft-In Soft-Out MIMO Detection With Low and Fixed Complexity

Authored by Mirsad Čirkić and Erik G. Larsson.

To appear in the IEEE Transactions on Signal Processing, 2014. The work is mainly based on the conference paper [93].

The fundamental problem of interest here is soft-input soft-output multiple-input multiple-output (MIMO) detection. The work proposes a method, referred to as subspace marginalization with interference suppression (SUMIS), that yields unprecedented performance at low and fixed (deterministic) complexity. The method provides a well-defined tradeoff between computational complexity and performance. Apart from an initial sorting step consisting of selecting channel-matrix columns, the algorithm involves no searching nor algorithmic branching; hence the algorithm has a completely predictable run-time and allows for a highly parallel implementation. The work numerically assess the performance of SUMIS in different practical settings: full/partial channel state information, sequential/iterative decoding, and low/high rate outer codes. It also includes comments on how the SUMIS method performs in systems with a large number of transmit antennas.

Paper D On Near-Optimal Very Large Multi-User MIMO Detection

Authored by Mirsad Čirkić and Erik G. Larsson.

Unpublished material that will be submitted for publication in the near future.

This work discusses efficient techniques for detection in large-dimension multiuser multiple-input multiple-output (MIMO) systems that are highly overdetermined. We exemplify the application of conjugate gradient methods in the setup of our interest and compare its performance with respect to methods based on the Neumann series expansion. We bring to light some important insights on the performance versus complexity tradeoffs that have not been uplifted before.

2.2.3 Future Work

The field of MIMO detection has been attractive for more than two decades and even though it has lost some of its popularity, it is still very popular. The focus has shifted more from fundamental investigations to exploring and exploiting specialized tricks and tweaks that are suited for some particular setting. It is now very common to reuse some already proposed method and utilize the usefulness of the method (specialize it) in order to get the most out of it for the particular setup of interest. There is a lot more room left for future work in specialization of different MIMO detection techniques. For instance, the newly emerged large MIMO setups bring a new structure to the detection problem that can be utilized and that significantly changes the focus. Also, there are not many actual hardware implementations of the different already proposed MIMO detectors. Relevant and fair real-life comparisons are of extreme interest to the field.

Bibliography

[1] M. Meeker, S. Flannery, L. Wu, and et al., "The mobile internet report," Tech. Rep., Morgan Stanley Research, Dec. 2009.

[2] E. Telatar, "Capacity of multi-antenna Gaussian channels," *European Transactions on Telecommunications*, vol. 10, no. 6, pp. 585–596, Jun. 2001.

[3] D. Tse and P. Viswanath, *Fundamentals of wireless communication*, Cambridge Uni. Press, New York, NY, USA, 2005.

[4] B. Hassibi and H. Vikalo, "On the sphere-decoding algorithm I. Expected complexity," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2806–2818, Aug. 2005.

[5] L. G. Barbero and J. S. Thompson, "Extending a fixed-complexity sphere decoder to obtain likelihood information for turbo-MIMO systems," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 5, pp. 2804 –2814, Sept. 2008.

[6] E. G. Larsson and J. Jaldén, "Fixed-complexity soft MIMO detection via partial marginalization," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3397–3407, Aug. 2008.

[7] D. Wübben, D. Seethaler, J. Jaldén, and G. Matz, "Lattice reduction," *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 70–91, May 2011.

[8] J. Choi and H. Nguyen, "SIC-based detection with list and lattice reduction for MIMO channels," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 7, pp. 3786–3790, Sept. 2009.

[9] D. L. Milliner, E. Zimmermann, J. R. Barry, and G. Fettweis, "A fixed-complexity smart candidate adding algorithm for soft-output MIMO detection," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 6, pp. 1016–1025, Dec. 2009.

[10] L. Bai and J. Choi, "Partial MAP-based list detection for MIMO systems," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 5, pp. 2544–2548, June 2009.

[11] Y. Li and J. Moon, "Reduced-complexity soft MIMO detection based on causal and noncausal decision feedback," *IEEE Transactions on Signal Processing*, vol. 56, no. 3, pp. 1178–1187, Mar. 2008.

[12] P. Aggarwal, N. Prasad, and X. Wang, "An enhanced deterministic monte carlo method for near-optimal MIMO demodulation with QAM constellations," *IEEE Transactions on Signal Processing*, vol. 55, no. 6, pp. 2395–2406, June 2007.

[13] J. Choi, Y. Hong, and J. Yuan, "An approximate MAP-based iterative receiver for MIMO channels using modified sphere detection," *IEEE Transactions on Wireless Communications*, vol. 5, no. 8, pp. 2119–2126, Aug. 2006.

[14] Z. Guo and P. Nilsson, "Algorithm and implementation of the K-best sphere decoding for MIMO detection," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 3, pp. 491–503, Mar. 2006.

[15] I. Nevat, T. Yang, K. Avnit, and J. Yuan, "MIMO detection with high-level modulations using power equality constraints," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 7, pp. 3383–3392, Sept. 2010.

[16] Z. Muhammad and Z. Ding, "Blind multiuser detection for synchronous high rate space-time block coded transmission," *IEEE Transactions on Wireless Communications*, vol. 10, no. 7, pp. 2171–2185, July 2011.

[17] K.S. Schneider, "Optimum detection of code division multiplexed signals," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-15, no. 1, pp. 181–185, 1979.

[18] S. Verdú, "Minimum probability of error for asynchronous Gaussian multiple access channels," *IEEE Transactions on Information Theory*, vol. 32, no. 1, pp. 85–96, Jan. 1986.

[19] S. Verdú, "Computational complexity of optimum multiuser detection," *Algorithmica*, vol. 4, no. 3, pp. 303–312, 1989.

[20] S. Verdú, *Multiuser Detection*, Cambridge Uni. Press, New York, NY, USA, 1st edition, 1998.

[21] A. Elkhazin, K.N Plataniotis, and S. Pasupathy, “Reduced-dimension map turbo-blast detection,” *IEEE Transactions on Communications*, vol. 54, no. 1, pp. 108–118, Jan. 2006.

[22] J. W. Choi, B. Shim, A.C. Singer, and N. I. Cho, “Low-complexity decoding via reduced dimension maximum-likelihood search,” *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1780–1793, Mar. 2010.

[23] K. Vardhan, S.K., Mohammed, A. Chockalingam, and B.S. Rajan, “A low-complexity detector for large MIMO systems and multicarrier CDMA systems,” *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 3, pp. 473–485, Apr. 2008.

[24] E. Viterbo and J. Boutros, “A universal lattice code decoder for fading channels,” *IEEE Transactions on Information Theory*, vol. 45, no. 5, pp. 1639–1642, Jul. 1999.

[25] L. Bai and J. Choi, *Low Complexity MIMO Detection*, Springer, Boston, MA, USA, 2012.

[26] G. Papa, D. Ciuonzo, G. Romano, and P. S. Rossi, “Soft-input soft-output king decoder for coded MIMO wireless communications,” in *Proc. IEEE 8th International Symposium on Wireless Communication Systems (ISWCS)*, 2011, pp. 760–763.

[27] C. Studer and H. Bölcskei, “Soft-input soft-output single tree-search sphere decoding,” *IEEE Transactions on Information Theory*, vol. 56, no. 10, pp. 4827–4842, Oct. 2010.

[28] E. G. Larsson, “MIMO detection methods: How they work,” *IEEE Signal Processing Magazine*, vol. 26, no. 3, pp. 91–95, May 2009.

[29] R. Harris, D. M. Chabries, and F. Bishop, “A variable step (VS) adaptive filter algorithm,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 2, pp. 309–316, Apr. 1986.

[30] G.H. Golub and C.F. Van Loan, *Matrix computations*, The Johns Hopkins Uni. Press, Baltimore, Maryland, USA, 1996.

[31] J. Maurer, J. Jaldén, D. Seethaler, and G. Matz, “Achieving a continuous diversity-complexity tradeoff in wireless MIMO systems via pre-equalized sphere-decoding,” *IEEE Journal ofSelected Topics in Signal Processing*, vol. 3, no. 6, pp. 986–999, Dec. 2009.

[32] Y. Shang and X.-G. Xia, “On fast recursive algorithms for v-blast with optimal ordered sic detection,” *IEEE Transactions on Wireless Communications*, vol. 8, no. 6, pp. 2860–2865, June 2009.

[33] Y. Dai and Z. Yan, "Memory-constrained tree search detection and new ordering schemes," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 6, pp. 1026–1037, Dec. 2009.

[34] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Mathematics of Computations*, vol. 44, no. 170, pp. 463–471, 1985.

[35] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Transactions on Information Theory*, vol. 48, no. 8, pp. 2201–2214, Aug. 2002.

[36] M. O. Damen, H. El Gamal, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Transactions on Information Theory*, vol. 49, no. 10, pp. 2389–2402, Oct. 2003.

[37] T. Cui and C. Tellambura, "An efficient generalized sphere decoder for rank-deficient MIMO systems," *IEEE Communication Letters*, vol. 9, no. 5, pp. 423–425, May 2005.

[38] J. Jaldén and B. Ottersten, "On the complexity of sphere decoding in digital communications," *IEEE Transactions on Signal Processing*, vol. 53, no. 4, pp. 1474–1484, Apr. 2005.

[39] L. G. Barbero and J. S. Thompson, "Fixing the complexity of the sphere decoder for MIMO detection," *IEEE Transactions on Wireless Communications*, vol. 7, no. 6, pp. 2131–2142, Jun. 2008.

[40] A. K. Lenstra, H. W. Lenstra, and L. Lovász, "Factoring polynomials with rational coefficients," *Mathematische Annalen*, vol. 261, no. 4, pp. 515–534, 1982.

[41] P. H. Tan and L. K. Rasmussen, "The application of semidefinite programming for detection in CDMA," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 8, pp. 1442–1449, Aug. 2001.

[42] B. Steingrimsson, Z.-Q. Luo, and K. M. Wong, "Soft quasi-maximum-likelihood detection for multiple-antenna wireless channels," *IEEE Transactions on Signal Processing*, vol. 51, no. 11, pp. 2710–2719, Nov. 2003.

[43] N. D. Sidiropoulos and Z.-Q. Luo, "A semidefinite relaxation approach to MIMO detection for high-order QAM constellations," *IEEE Signal Processing Letters*, vol. 13, no. 9, pp. 525–528, Sep. 2006.

[44] A. Mobasher, M. Taherzadeh, R. Sotirov, and A. K. Khandani, "A near-maximum-likelihood decoding algorithm for MIMO systems based on semidefinite programming," *IEEE Transactions on Information Theory*, vol. 53, no. 11, pp. 3869–3886, Nov. 2007.

[45] J. Nocedal and S. J. Wright, *Numerical optimization*, Springer, New York, NY, 2:nd edition, 2006.

[46] J. Jaldén and B. Ottersten, "The diversity order of the semidefinite relaxation detector," *IEEE Transactions on Information Theory*, vol. 54, no. 4, pp. 1406–1422, Apr. 2008.

[47] Y.-T. Zhou, R. Chellappa, A. Vaid, and B. K. Jenkins, "Image restoration using a neural network," *IEEE Transactions on Acoustic, Speech, Signal Processing*, vol. 36, no. 7, pp. 1141–1151, July 1988.

[48] Y. Sun, "Hopfield neural network based algorithms for image restoration and reconstruction - part i: Algorithms and simulations," *IEEE Transactions on Signal Processing*, vol. 48, no. 7, pp. 2105–2118, July 2000.

[49] Y. Sun, "Hopfield neural network based algorithms for image restoration and reconstruction - part ii: Performance analysis," *IEEE Transactions on Signal Processing*, vol. 48, no. 7, pp. 2119–2131, July 2000.

[50] Y. Sun, "Eliminating-highest-error and fastest-metric-descent criteria and iterative algorithms for bit synchronous CDMA multiuser detection," in *Proc. IEEE International Conference on Communications*, 1998, pp. 1576–1580.

[51] Y. Sun, "A family of linear complexity likelihood ascent search detectors for CDMA multiuser detection," in *Proc. IEEE International Symposium on Spread Spectrum Techniques and Applications*, 2000, pp. 713–717.

[52] S. K. Mohammed, A. Zaki, A. Chockalingam, and B. S. Rajan, "High-rate space-time coded large-MIMO systems: Low-complexity detection and channel estimation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 6, pp. 958–974, Dec. 2009.

[53] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers and Operations Research*, vol. 13, no. 5, pp. 533–549, 1986.

[54] F. Glover, "Tabu search - part 1," *ORSA Journal on Computing*, vol. 1, no. 2, pp. 190–206, 1989.

[55] R. Battiti, "The reactive tabu search," *ORSA Journal on Computing*, vol. 6, no. 2, pp. 126–140, Spring 1994.

[56] T. Datta, N. Srinidhi, A. Chockalingam, and B. S. Rajan, "Random-restart reactive tabu search algorithm for detection in large-mimo systems," *IEEE Communications Letters*, vol. 14, no. 12, pp. 1107–1109, Dec. 2010.

[57] N. Srinidhi, T. Datta, A. Chockalingam, and B. S. Rajan, "Layered tabu search algorithm for large-MIMO detection and a lower bound on ML performance," *IEEE Transactions on Communications*, vol. 59, no. 11, pp. 2955–2963, Nov. 2011.

[58] B.M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Transactions on Communications*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

[59] B. Mennenga, A. von Borany, and G. Fettweis, "Complexity reduced soft-in soft-out sphere detection based on search tuples," in *IEEE International Conference on Communications*, 2009, pp. 1–6.

[60] R. Wang and G. B. Giannakis, "Approaching MIMO channel capacity with soft detection based on hard sphere decoding," *IEEE Transacations on Communications*, vol. 54, no. 4, pp. 587–590, Apr. 2006.

[61] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*, Prentice Hall, New Jersey, NJ, USA, 2000.

[62] D. Persson and E. G. Larsson, "Partial marginalization soft MIMO detection with higher order constellations," *IEEE Transactions on Signal Processing*, vol. 59, no. 1, pp. 453–458, Jan. 2011.

[63] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Transactions on Wireless Communications*, vol. 9, no. 11, pp. 3590–3600, Nov. 2010.

[64] H.V. Poor and S. Verdú, "Single-user detectors for multiuser channels," *IEEE Transactions on Communications*, vol. 36, no. 1, pp. 50–60, Jan. 1988.

[65] M. Brandt-Pearce and B. Aazhang, "Multiuser detection for optical code division multiple access systems," *IEEE Transactions on Communications*, vol. 42, no. 234, pp. 1801–1810, Feb. 1994.

[66] U. Madhow and M.L. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA," *IEEE Transactions on Communications*, vol. 42, no. 12, pp. 3178–3188, Dec. 1994.

[67] A. Duel-Hallen, J. Holtzman, and Z. Zvonar, "Multiuser detection for CDMA systems," *IEEE Personal Communications*, vol. 2, no. 2, pp. 46–58, Apr. 1995.

[68] M.K. Varanasi, "Group detection for synchronous Gaussian code-division multiple-access channels," *IEEE Transactions on Information Theory*, vol. 41, no. 4, pp. 1083–1096, July 1995.

[69] H.V. Poor and S. Verdú, "Probability of error in MMSE multiuser detection," *IEEE Transactions on Information Theory*, vol. 43, no. 3, pp. 858–871, May 1997.

[70] E. G. Larsson, P. Stoica, and J. Li, "On maximum-likelihood detection and decoding for space-time coding systems," *IEEE Transactions on Signal Processing*, vol. 50, no. 4, pp. 937–944, Apr. 2002.

[71] L. Liu, J. Löfgren, P. Nilsson, and V. Öwall, "VLSI implementation of a soft-output signal detector for multimode adaptive multiple-input multiple-output systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 12, pp. 2262–2273, Dec. 2013.

[72] A. D. Murugan, H. El Gamal, M. O. Damen, and G. Caire, "A unified framework for tree search decoding: Rediscovering the sequential decoder," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 933–953, Mar. 2006.

[73] H. Artes, D. Seethaler, and F. Hlawatsch, "Efficient detection algorithms for MIMO channels: a geometrical approach to approximate ML detection," *IEEE Transactions on Signal Processing*, vol. 51, no. 11, pp. 2808–2820, Nov. 2003.

[74] J. Benesty, Y. Huang, and J. Chen, "A fast recursive algorithm for optimum sequential signal detection in a BLAST system," *IEEE Transactions on Signal Processing*, vol. 51, no. 7, pp. 1722–1730, Jul. 2003.

[75] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, and H. Bolcskei, "VLSI implementation of MIMO detection using the sphere decoding algorithm," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 7, pp. 1566–1577, July 2005.

[76] A. Wiesel, Y.C Eldar, and S. Shamai, "Semidefinite relaxation for detection of 16-QAM signaling in MIMO channels," *IEEE Signal Processing Letters*, vol. 12, no. 9, pp. 653–656, Sep. 2005.

[77] T. Datta, N. A. Kumar, A. Chockalingam, and B. S. Rajan, "A novel monte carlo sampling based receiver for large-scale uplink multiuser MIMO systems," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 7, pp. 3019–3038, Sept. 2013.

[78] C. Studer, A. Burg, and H. Bölcskei, "Soft-output sphere decoding: algorithms and VLSI implementation," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 2, pp. 290–300, Feb. 2008.

[79] K. Nikitopoulos and G. Ascheid, "Complexity adjusted soft-output sphere decoding by adaptive LLR clipping," *IEEE Communications Letters*, vol. 15, no. 8, pp. 810–812, Aug. 2011.

[80] I-W. Lai, G. Ascheid, H. Meyr, and T.-D. Chiueh, "Low-complexity channel-adaptive MIMO detection with just-acceptable error rate," in *Proc. IEEE 69th Vehicular Technology Conference (VTC)*, 2009, pp. 1–5.

[81] K.-C. Lai, C.-C. Huang, and J.-J. Jia, "Variation of the fixed-complexity sphere decoder," *IEEE Communications Letters*, vol. 15, no. 9, pp. 1001–1003, Sept. 2011.

[82] X. Wu and J. S. Thompson, “FPGA implementation of an efficient high-throughput sphere decoder for MIMO systems based on the smallest singular value threshold,” in *Proc. IEEE NASA/ESA Conference on Adaptive Hardware and Systems*, 2010, pp. 340–345.

[83] L. Liu, “Energy-efficient MIMO detection using link-adaptive parameter adjustment,” in *Proc. IEEE Vehicular Technology Conference (VTC Spring)*, 2013, pp. 1–5.

[84] Mirsad Čirkić, *Optimization of Computational Resources for MIMO Detection*, Linköping Studies in Science and Technology. Licentiate Thesis No. 1514, Linköping University, Sweden, 2011.

[85] D. W. Waters, N. Sommer, A. Batra, and S. Hosur, “Dynamic resource allocation to improve MIMO detection performance,” U.S. Patent Application 0 137 762 A1, Jun. 2008.

[86] I-W. Lai, C.-H. Lee, G. Ascheid, and T.-D. Chiueh, “Channel-adaptive MIMO detection based on the multiple-choice knapsack problem (MCKP),” *IEEE Wireless Communications Letters*, vol. 1, no. 6, pp. 633–636, Dec. 2012.

[87] M. Čirkić, D. Persson, and E. G. Larsson, “Optimization of computational resource allocation for soft MIMO detection,” in *Proc. 43:rd Asilomar Conference on Signals, Systems and Computers*, 2009, pp. 1488–1492.

[88] M. Čirkić, D. Persson, and E. G. Larsson, “New results on adaptive computational resource allocation in soft MIMO detection,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2011, pp. 2972–2975.

[89] M. Čirkić, D. Persson, E. G. Larsson, and J.-Å. Larsson, “Gaussian approximation of the LLR distribution for the ML and partial marginalization MIMO detectors,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2011, pp. 3232–3235.

[90] E. G. Larsson and O. Gustafsson, “The impact of dynamic voltage and frequency scaling on multicore DSP algorithm design,” *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 127–144, May 2011.

[91] B. Parhami, *Computer arithmetic: algorithms and hardware designs*, Oxford University Press, Inc., New York, NY, USA, 2nd edition, 2009.

[92] S. Sanayei and A. Nosratinia, “Antenna selection in MIMO systems,” *IEEE Communications Magazine*, vol. 42, no. 10, pp. 68–73, Oct. 2004.

[93] M. Čirkić and E. G. Larsson, “Near-optimal soft-output fixed-complexity MIMO detection via subspace marginalization and interference suppression,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Mar. 2012, pp. 2805–2808.

Part II

Adaptive Computational Resource Allocaiton

Papers

Part III

Soft MIMO Detection

Papers