We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Chapter

A Review to Massive MIMO Detection Algorithms: Theory and Implementation

Bastien Trotobas, Amor Nafkha and Yves Louët

Abstract

Multiple-input multiple-output (MIMO) systems entered most major standards in the past decades, including IEEE 802.11n (Wi-Fi) and long-term evolution (LTE). Moreover, MIMO techniques will be used for 5G by increasing the number of antennas at the base station end. MIMO systems enable spatial multiplexing, which has the potential of increasing the capacity of the communication channel linearly with the minimum of the number of antennas installed at both sides without sacrificing any additional bandwidth or power. To handle the space-division multiplexing (SDM), receivers have to implement new algorithms to exploit the spatial information in order to distinguish the transmitted data streams. This chapter provides an overview of the most well-known and promising MIMO detectors, as well as some unusual-yet-interesting ones. We focus on the description of the different paradigms to highlight the different approaches that have been studied. For each paradigm, we describe the mathematical framework and give the underlying philosophy. When hardware implementations are available in the literature, we provide the results reported and give the according references.

Keywords: MIMO systems, MIMO detectors, space-division multiplexing, SDM-MIMO, linear detection, interference cancelation, tree-search

1. Introduction

In multiple-input multiple-output (MIMO) communication systems, both the transmitters and receivers are equipped with several antennas which will help in achieving high gains in spectral, power, and energy efficiency compared to conventional single-input single-output (SISO) systems where both the transmitters and receivers have only one antenna each. As a matter of fact, the MIMO systems have the ability to turn multipath propagation and multipath delay spread into a benefit for the receiver. The key advantage of MIMO systems is the many orders of magnitude of the signal-to-noise ratio (SNR) at no extra bandwidth. However, a non-negligible software and hardware processing complexity is added at both sides (transmitter and receiver). Present wireless communication standards including Wi-Fi standards like IEEE 802.11n/ac, long-term evolution (LTE), and WiMAX are considering MIMO technology as a key element. Moreover, in the next generation of wireless technology systems (i.e., 5G), massive MIMO is emerging as a new research field in which base stations are equipped with 100 or more antennas. At

the receiver side, designing reliable and energy-efficient MIMO detectors is a very challenging task, because of the complexity of the implementation of the signal detection due to the interfering sub-streams. The signal detection problem refers to finding the most probable transmitted symbols based on the perfect channel state information (CSI) available at the receiver and the received signal.

The hardware implementation of massive MIMO detector is of particular interest to deal with 5G wireless technology. Optimal massive detectors such as the maximum likelihood detector (MLD) or the sphere decoding (SD) are considered infeasible given their high computational complexity. Hence, low computational complexity algorithm achieving near-optimal performance is required; many existing detection algorithms like zero forcing (ZF), minimum mean-square error

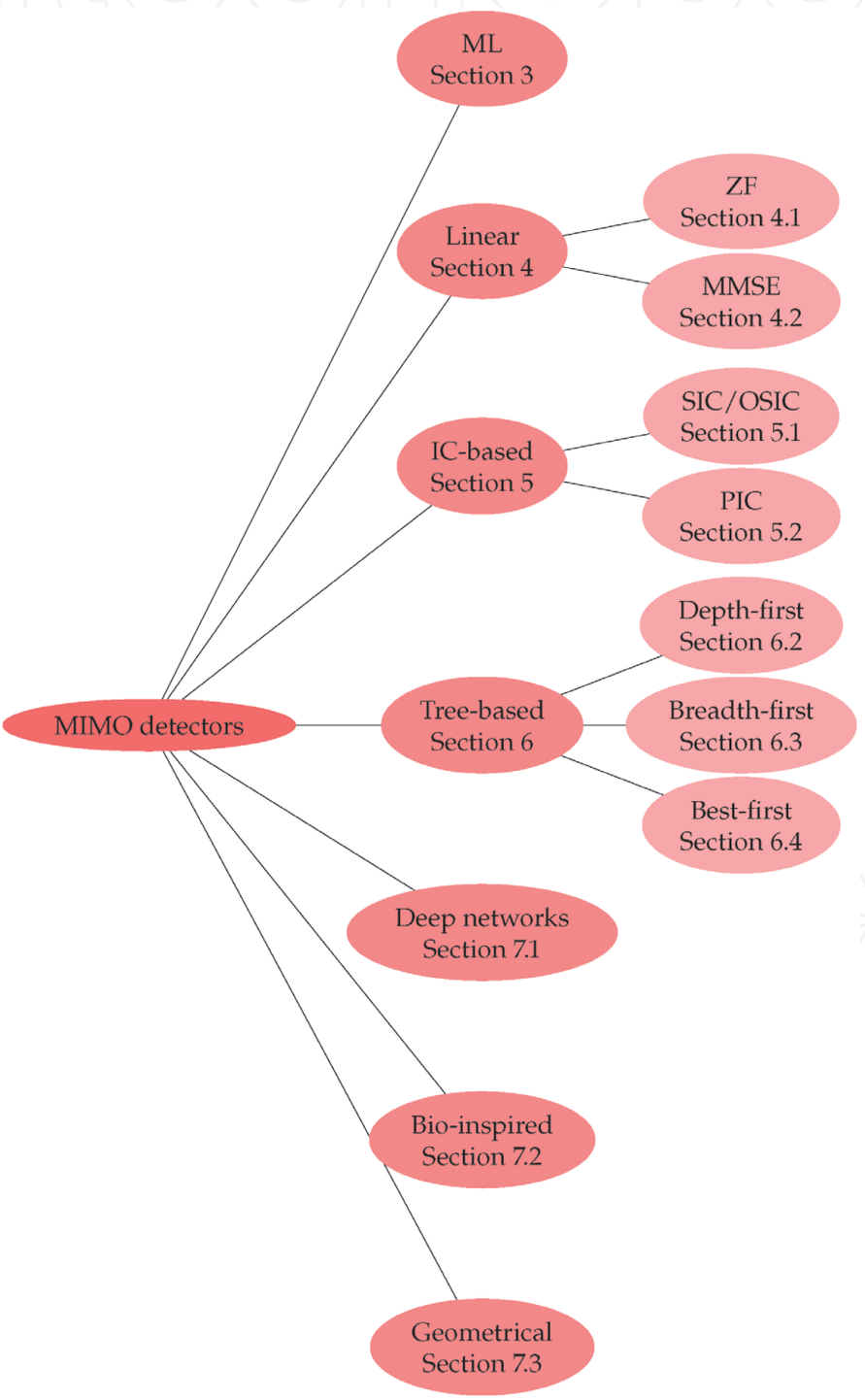


Figure 1.
Tree mind map of the detectors described in this chapter and the number or the corresponding section.

(MMSE), and successive interference cancelation (SIC) are used to deal with massive MIMO detection. In [1, 2], the authors presented surveys on various MIMO and massive MIMO detection techniques from algorithmic viewpoints. Although many classical massive MIMO detectors have been proposed in the literature, herein, new recent algorithms based on the application of machine learning, geometrical techniques, and bioinspired methods are presented and discussed.

In this chapter, we propose an overview of the SDM detection algorithms. We specifically stress out the different paradigms that are used to solve the detection problem and compare all of them. Thus, we describe the most well-known and promising MIMO detectors, as well as some unusual-yet-interesting ones. Section 2 presents the framework and the assumptions that are used in the remainder section. Section 3 introduces the maximum likelihood (ML) optimal detector, and then Section 4 describes the linear ones. Section 5 details algorithms based on the interference cancelation, and Section 6 discusses the one based on tree-search. Finally, Section 7 highlights unusual-yet-interesting detectors before Section 8 concludes the chapter. **Figure 1** provides an overview of all the detectors described in this chapter as a tree mind map.

2. Introduction to MIMO detection algorithms

In the SDM framework, data streams are transmitted at the same time and at the same frequency, and the receiver relies on spatial consideration to distinguish the streams. Herein, we assume that the MIMO transmitter does not use any spatial coding and that all data streams are independent. To give the reader a unified mathematical description through this chapter, we adopt the following notation: scalars, vectors, and matrices are denoted by lower-case, bold-face lower-case, and bold-face higher-case letters, respectively. We call $\mathbf{v}(i)$ the ith coefficient of the vector $\mathbf{v}$, and $\mathbf{H}(i,j)$ is the element of the ith row and jth column in the $\mathbf{H}$ matrix.

In the linear input–output MIMO model where data are transmitted as the symbols of a constellation Φ, the received vector $\mathbf{y} \in \mathbb{C}^M$ is the result of the emitted symbols $\mathbf{x} \in \Phi^N$ propagated through the channel $\mathbf{H}$ and added to an additive noise $\mathbf{w}$. This model leads to the following equation:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w} \tag{1}$$

and the MIMO detection problem then refers to the combinatorial optimization problem:

$$\arg \min_{\mathbf{x} \in \Phi^N} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2. \tag{2}$$

Assuming a circularly symmetric Gaussian noise, solving Eq. (2) is equivalent to searching the most probable emitted symbol vector based on the signal on each receive antennas and the channel state. Even if $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2$ is a convex function with respect to x, the detection problem is not a convex optimization problem due to the discrete feasible solution set Φ^N. As a result, a special algorithm has to be used, and this chapter will describe the most common ones.

Let us start by outlining the traditional assumptions that we will use in the present chapter. Although many constellation types could be used in MIMO systems, we limit the discussion to the square quadrature amplitude modulations (QAMs) that are most commonly investigated. Besides, the channel is considered

memoryless, linear, and flat and with a block fading[1]. In this chapter, we assume that channel state information (CSI) is correctly estimated at the receiver side but not at the transmitter side. The impact of imperfect CSI at the receiver on the performance of detection algorithms is not addressed in the present chapter. Some known training symbols are sent from the transmitter, based on which the receiver estimates the channel before proceeding to the detection of the transmitted data symbols.

The channel matrix is modeled as a complex matrix $\mathbf{H} \in \mathbb{C}^{(M,N)}$. In that case, the element $\mathbf{H}(i,j)$ refers to the complex channel gain between the jth transmit antenna to the ith receive antenna. Many channel models can fit in this framework, and we stick to the most popular one: the uncorrelated Rayleigh fading channel [3, 4]. The uncorrelated channel model provides a good approximation of propagating environments with rich scattering where the signals between the transmitter and the receiver experience many different paths and no strong line of sight between the transmitter and the receiver. This situation occurs, for instance, in urban and indoor conditions. In these conditions, each receiver antenna receives a sum of a large number of signal paths, and the channel transfer functions can be modeled as the realization of a circularly symmetric normal distribution.

3. Maximum likelihood detector

Obtaining the optimal result requires, in the most straightforward approach, the use of the ML detector that solves Eq. (2) using an exhaustive search. Even if this method gives the best result since all $x \in \Phi^N$ are evaluated, it is not suitable for real implementation. Indeed, the number of vectors to be tested grows exponentially with the number of transmit antenna and the constellation size. Thus, the computational cost of evaluating Eq. (2) requires an unrealistic quantity of resources to detect the transmitted vector $\mathbf{x}$. That is why a variety of detection algorithm has been developed throughout the past year to achieve the same detection performance of ML detectors while having a tractable complexity.

From a computational theory perspective, the detection problem is an instance of the closest lattice-point search (CLPS) problem with a specified lattice [5]. It has been proved that regardless of the preprocessing on the lattice (i.e., the channel matrix), the problem is always NP-hard [5]. The NP-hardness implies that it is not possible, at the moment, to find any detector that is sure to have both an optimal performance and a polynomial complexity[2]. For that reason, all the following detectors have suboptimal performance (which can be very close to optimal) or a non-polynomial worst-case complexity (which can be polynomial is the average case).

4. Linear detectors

4.1 Zero forcing (ZF) detector

Linear detectors are the most simple algorithms to solve the detection problem. The most basic one is the ZF algorithm that follows a two-step process. First, the ZF

[1] The nomenclature section provides definitions.

[2] It is widely believed that it does not exist any solution to solve NP-hard problems in polynomial time. Yet, this assumption is not proven so that there is still a possibility that such an algorithm exists and is just not discovered nowadays.

detector solves Eq. (2) transforming the constraint from $\mathbf{x} \in \Phi^N$ to $\mathbf{x} \in \mathbb{C}^N$ such that the problem become an easy-to-solve convex optimization with a known mathematical solution:

$$\mathbf{x}_0 = \mathbf{H}^{+}\mathbf{y} \tag{3}$$

with $\mathbf{H}^{+} = \left(\mathbf{H}^H\mathbf{H}\right)^{-1}\mathbf{H}^H$ being the left Moore-Penrose pseudoinverse. Then, the constraint on x is reintroduced by quantizing the vector accordingly to the constellation in use. This quantization should lead to a good estimation as after the application of the detection matrix $\mathbf{T}_{\mathrm{ZF}} = \mathbf{H}^{+}$, Eq. (1) becomes

$$\mathbf{T}_{\mathrm{ZF}}.\mathbf{y} = \mathbf{x} + \mathbf{H}^{+}\mathbf{w} \tag{4}$$

highlighting that all the interference are canceled. The previous equation is also the proof that the ZF detector is the optimal linear one regarding the signal-to-interference ratio (SIR) criteria. Indeed, one can see that the vector $\mathbf{T}_{\mathrm{ZF}}.\mathbf{y}$ contains each stream independently plus some noise but without any interference.

4.2 Minimum mean-square error (MMSE) detector

By only focusing on the interference, the ZF detector performance suffers from not taking the noise into account. Indeed, if the noise level is known to the receiver, a Bayesian estimator including this information can provide a better detection. A linear Bayesian estimator minimizing the mean-square error can be derived using the orthogonality principle [6] leading to

$$\mathbf{T}_{\mathrm{MMSE}} = \left(\mathbf{H}^H\mathbf{H} + 2\sigma^2\mathbf{I}\right)^{-1}\mathbf{H}^H \tag{5}$$

with σ^2 being the noise variance per real direction. The detector based on this detection matrix, followed by the quantization, is called the minimum mean-square error (MMSE), and it is known to maximize the signal-to-noise-plus-interference ratio (SINR). When the signal-to-noise ratio (SNR) is low (i.e., σ^2 is high), the MMSE detector provides better results, jointly minimizing the interference and the noise. Otherwise, when σ^2 is very low, the corrective term becomes negligible, and the ZF and MMSE detectors overlap.

5. Interference cancellation detectors

To improve further the performance, it is necessary to drop the linear detector approach and look for more elaborate decoding algorithms. Historically, the first nonlinear detector type is still based on the principle of canceling signal interference. This concept leads to two approaches: an iterative one named successive interference cancelation (SIC) and a simultaneous version named parallel interference cancelation (PIC).

5.1 Successive interference cancellation (SIC) detector

The SIC detectors opt for a two-step iterative process: first, a decision is taken on the first position $\mathbf{x}_1$, and then assuming that the decision was right, the detector corrects $\mathbf{y}$ by removing the interference that would have been generated by $\mathbf{x}_1$.

Then, SIC detectors repeat this process on the next $\mathbf{x}$'s entry until the whole vector is received.

Even if the performance is better than with the linear detectors, the SIC process is very prone to error, given that the assumption at an iteration has an impact on all the following ones. For this reason, the simple SIC detector has quickly been replaced by a variant seeking for an optimal iteration order [7]. This variant called ordered successive interference cancelation (OSIC) aims to make the first assumption on the position that leads to the better SNR or SINR.

To select the best symbol to detect at each iteration, the OSIC detector computes the post-SNR or post-SINR for each symbol, assuming that the kth element is canceled using the detection matrix $\mathbf{T}$. Most of the time, the detection matrix is chosen to be $\mathbf{T}_{\mathrm{ZF}}$ or $\mathbf{T}_{\mathrm{MMSE}}$ optionally updated after each iteration. When using the SNR criterion, the value of the kth post-SNR is computed as in [7, 8].

$$\mathrm{SNR}_k = \frac{<\mathbf{x}(k)^2>_{\Phi} |\mathbf{T}_k \mathbf{h}_k|^2}{\sigma^2 \|\mathbf{T}_k\|^2} \tag{6}$$

with $\mathbf{T}_k$ being the kth row of $\mathbf{T}$, $\mathbf{h}_k$ the k^{th} column of $\mathbf{H}$, and $<\mathbf{x}_k^2>_{\Phi}$ the expected value over the constellation set. The latter term is the average signal power of the kth data stream that can be computed, assuming that each symbol is equiprobable, as

$$<\mathbf{x}_k^2>_{\Phi} = \frac{1}{\varphi} \sum_{\mathbf{x} \in \Phi} \mathbf{x}(k)^2 \tag{7}$$

with φ the number of symbols in constellation Φ.

When using the SINR criterion, the post-SINR expression becomes slightly more complex as the post-processed power of each other channel appears in the expression

$$\mathrm{SINR}_k = \frac{<\mathbf{x}(k)^2>_{\Phi} |\mathbf{T}_k \mathbf{h}_k|^2}{\sum_{l \neq k} <\mathbf{x}(l)^2>_{\Phi} |\mathbf{T}_l \mathbf{h}_l|^2 + \sigma^2 \|\mathbf{T}_k\|^2}. \tag{8}$$

For clarity sake, **Figure 2** sums up the OSIC detection algorithm introducing a process $t : \mathbf{H} \mapsto T$ to build a detection matrix from a channel one. This process is most of the time the Moore-Penrose pseudoinverse or the process described in Eq. (5). We also denote by $\mathcal{D}$ the set of the symbol index to be decoded and by $\leftarrow$ the affectation. One must note that an instruction is optional and may be skipped. If this instruction is applied, performance is increased by canceling the interference in the post-criterion computation and so is the complexity.

5.2 Parallel interference cancellation (PIC) detector

The main drawback of the OSIC algorithm is that the number of iterations grows linearly with the number of antennas. The number of stage becomes an issue for large MIMO system since each stage adds a reception delay. For that reason, a detector capable of canceling the interference for all antennas at once was developed.

The first application of such an algorithm to SDM systems dates from the early 2000s, and it is based on a few basic steps summed up in **Figure 3** as published in [9]. The main point of the PIC algorithm is to start by using a simple detector with poor performance, most of the time a linear one, and cancel the interference on all antennas at once based on the assumption. If better performance is required, it is possible to iterate the last three instructions as many times as needed by using the new detected symbol as the new assumption.

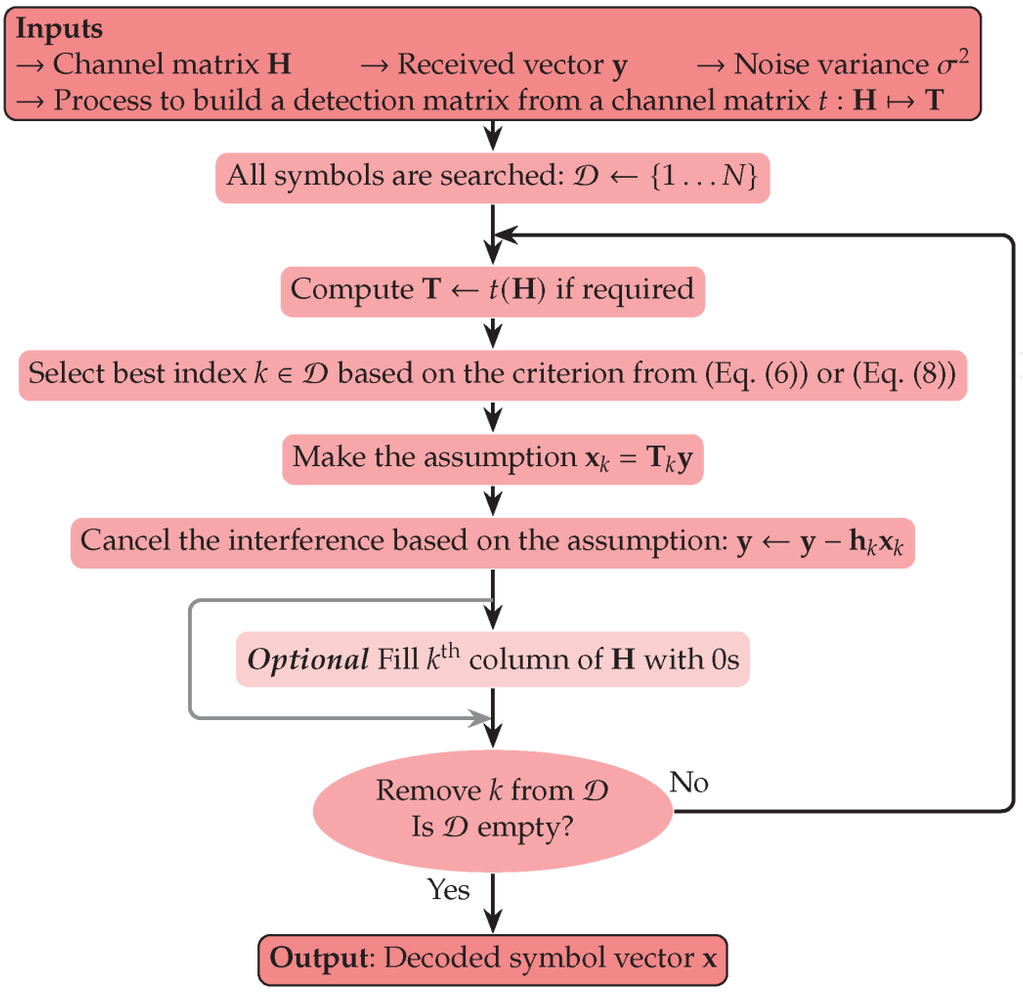


Figure 2.
OSIC algorithm outline.

Simultaneously, the iterative reception techniques developed for turbo codes and single-input single-output channels are adapted to MIMO systems. The goal is to receive a coded message by alternating between soft-input soft-output detector and decoder. Each algorithm uses a priori information from the other to improve its performance [10]. This method leads to one of the current most accomplished version of the PIC family: a soft-input soft-output detector to be used in iterative decoding with any message coding [11].

This version adds several improvements to the basic algorithm described in **Figure 3**. First, it uses the soft symbols from [12] that are defined as the expected value of the symbols knowing the a priori. The reliability of a soft symbol is computed as its variance. Then, the parallel cancelation (see the third instruction in **Figure 3**) is performed using the soft symbols in place of the rough estimation. Finally, a last MMSE filtering is performed before the computation of the log-likelihood ratios (LLRs). Further reductions in complexity are also used, such as the max-log approximation [10, 13, 14] or the channel Gram matrix [15]. Thanks to all of these improvements, an application-specific integrated circuit (ASIC) is reported to achieve a throughput greater than 750 Mb/s with good BER performance [11].

5.3 Selecting between SIC and PIC detectors

The key idea to select between SIC and PIC detectors is to compare the relative quality of data streams. As stated earlier, SIC algorithms guess the best data stream and then process the other one based on this assumption. This process makes the

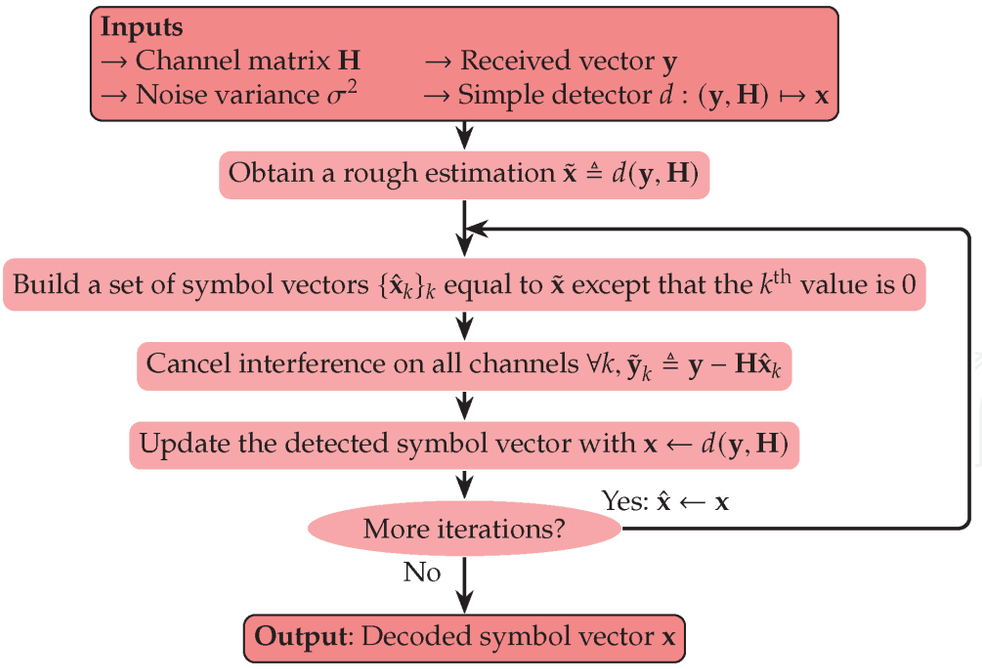


Figure 3.
PIC algorithm outline as published in [9].

SIC algorithms very prone to error propagation. Indeed, if an assumption is wrong, the error has consequences on all the data streams to be detected. Hence, SIC detectors should be used when there is a net ranking in the quality of each data stream. A basic scenario for this would be a MIMO system receiving data from several users with different channel qualities.

On the contrary, PIC detectors process all the data streams at once so that they are more resilient to interstream error propagation. However, the parallel computations assume that every data stream is as reliable as the other. Due to this assumption, a poor-quality data stream propagates its error to the whole system. For that reason, PIC detectors are well suited when all data streams have the same quality level.

6. Tree-search-based detectors

Tree-search-based detectors are the current most investigated algorithms. They use a different framework than the linear and the interference cancelation detector. As the name suggests, the tree-search detector interprets the detection problem from Eq. (2) as the search for the best path in a tree. Tree-search-based detectors can either be optimal with a non-polynomial yet small complexity or quasi-optimal yet not optimal with a polynomial convexity.

Figure 4 gives an example of the tree interpretation for a constellation with four symbols and two data streams. In this configuration, solving the detection problem is equivalent to find two symbols in the set $\Phi = \{s_1, s_2, s_3, s_4\}$ that minimize the objective function. This process can be seen as finding the path in the tree that leads to the best objective function. The first tree level corresponds to the first symbol and so on for each level.

In this paradigm, the exhaustive search detector described in Section 3 computes the objective function for each leaf node and then selects the best path. Tree-search-based detectors search for the best leaf without trying every path. This leads to three enumeration paradigms: depth-first, breadth-first, and best-first. These

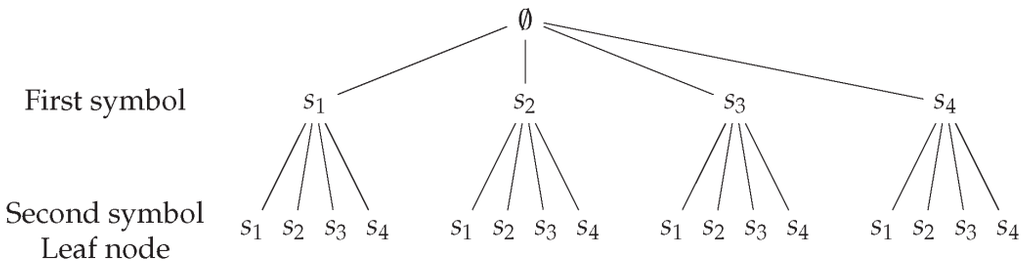


Figure 4.
Tree view of the detection problem: Example for $\Phi = \{s_1, s_2, s_3, s_4\}$ and two data streams.

paradigms will be detailed after the description of the preprocessing used by all the variants.

6.1 Preprocessing using QR decomposition

All tree-search-based detectors use the same preprocessing. Let be $\mathbf{H} = \mathbf{QR}$ the QR decomposition of the channel matrix with $\mathbf{Q}$ a unitary matrix and $\mathbf{R}$ an upper triangular one. The decomposition is computed only once per coherence block leading to a negligible overhead of the complexity per received symbol. Using the QR decomposition, we have

$$\|\mathbf{y} - \mathbf{Hx}\| = \|\mathbf{y} - \mathbf{QRx}\| = \|\mathbf{Q}^H\mathbf{y} - \mathbf{Q}^H\mathbf{QRx}\| \tag{9}$$

as unitary matrices act as isometries. Thus, by exploiting the property of unitary matrices $\mathbf{Q}^H = \mathbf{Q}^{-1}$, this norm can be rewritten as

$$\|\mathbf{y} - \mathbf{Hx}\| = \|\tilde{\mathbf{y}} - \mathbf{Rx}\| \tag{10}$$

with $\tilde{\mathbf{y}} \triangleq \mathbf{Q}^H\mathbf{y}$. Computing $\tilde{\mathbf{y}}$ is the only overhead in complexity that is required on a symbol basis as it cannot be preprocessed for the whole coherence block.

The point of this QR preprocessing is that the triangularity of $\mathbf{R}$ allows to compute the objective function iteratively. Indeed, we can introduce for all $k \in \{1, \ldots, N\}$, $\mathbf{d}(k) \triangleq \tilde{\mathbf{y}}(k) - (\mathbf{Rx})(k)$ with $(\mathbf{Rx})(k)$ being the kth coefficient of the product Rx. Given this definition, the objective function is written as

$$\|\tilde{\mathbf{y}} - \mathbf{Rx}\|^2 = \sum_{k=1}^{n} \mathbf{d}(k)^2. \tag{11}$$

Moreover, the triangularity of $\mathbf{R}$ gives

$$\forall k \in \{1, \ldots, N\}, (\mathbf{Rx})(k) = \sum_{j=1}^{n} \mathbf{R}(k,j)\mathbf{x}(j) = \sum_{j=k}^{n} \mathbf{R}(k,j)\mathbf{x}(j) \tag{12}$$

that leads to

$$\forall k \in \{1, \ldots, N\}, \mathbf{d}(k) = \tilde{\mathbf{y}}(k) - \sum_{j=k}^{n} \mathbf{R}(k,j)\mathbf{x}(j). \tag{13}$$

With this expression, it is clear that we can compute partial estimations of the objective function and $\mathbf{d}(k)$ coefficients while the symbol vector is built. Indeed,

starting from the last component, Eq. (13) is fully evaluated for the jth position as soon as a hypothesis is made on $\mathbf{x}(j)$. Thus, it is possible to add a new operand in Eq. (11) and to have an idea of how promising the partial symbol vector is. The partial objective function from Eq. (11) is traditionally called the partial Euclidean distance (PED).

6.2 Depth-first tree-search detection: sphere decoding

The depth-first paradigm is the oldest one, and it is commonly known in the communication field as the sphere decoding (SD). SD is the transposition of the mathematical Fincke-Pohst algorithm in the telecommunication field [16]. The basic principle of this algorithm is to define an upper-bound for the objective function named the radius r^2 and then to use it to prune paths as early as possible. Reintroducing x_0 from Eq. (3), the upper-bound constraint gives

$$\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 = \|\mathbf{H}(\mathbf{x}_0 - \mathbf{x})\| \leq r^2. \tag{14}$$

This inequality highlights that constraining the objective function may be interpreted as looking for solution no so far from $\mathbf{x}_0$. As stated in Eq. (4), the only deviation from $\mathbf{x}_0$ is due to the noise so that the choice of r must be adapted to the SNR. Thus, if the SNR is high, the radius can be small, while in the contrary scenario, the radius should be increased so that there is at least one vector $\mathbf{x}$ satisfying the constraint from Eq. (14).

In the remainder of this section, we assume that r^2 is adequately chosen and that there is at least a solution. As stated in Section 6.1, the QR decomposition allows us to compute a PED at each level. As the PED is a sum of squares, it can only increase during the decoding process. Thus, if at some point, a PED violates the constraint from Eq. (14), then all the vectors build upon this partial solution are bound to be infeasible. From a tree-search perspective, this means that if a node already breaks the constraint, all its children will do the same. Thus, all paths starting from this node can be pruned without performance loss.

The SD is referred to as a depth-first detector as starting from the root node, it goes as depth as possible until it reaches a leaf or violates Eq. (14). If a leaf is reached, it is compared to the best leaf so far and saved if it is the new best leaf. If Eq. (14) is violated, the SD algorithm backtracks and explores a new path. When all paths are either completed or pruned, the result is the best leaf reached.

The Schnorr-Euchner (SE) enumeration is another depth-first enumeration known to perform better by using a lattice reduction method [17, 18]. The basic idea is to explore the node's children by the increasing order of their PED. This is particularly useful when using the radius reduction technique that sets an infinite r^2 at the beginning and then updates it to the best objective function for a leaf encounter so far.

The SD algorithm and its SE version are optimal as they ensure to find the exact solution of the detection problem. Indeed, the best leaf is obviously the best leaf among all the completed paths, and the pruned paths cannot lead to a better point due to their already worst PED. The NP-hardness argument detailed in Section 3 implies that SD has a non-polynomial worst-case complexity. Moreover, SD expected complexity is also non-polynomial even if the exponential growth is slow enough to compete with polynomial detectors under certain circumstances [19].

A very efficient soft-input soft-output depth-first algorithm is the single tree-search sphere decoding (STS-SD) [20]. To produce its soft-output, it uses the max-log approximation [10, 13, 14] and makes some changes on the pruning

criterion. The max-log approximation avoids the computation of the exact LLRs by claiming that

$$L_i \approx \frac{1}{2\sigma^2}\left(\min_{x\in\chi_i^0}\|\mathbf{y}-\mathbf{Hx}\|^2 - \min_{x\in\chi_i^1}\|\mathbf{y}-\mathbf{Hx}\|^2\right) \tag{15}$$

where $\chi_i^k = \{x \in \Phi^N : b_i = k\}$ is the set of all symbols with the ith bit set to k. Thus, to compute the max-log approximation, one must know the objective function of the best leaf (i.e., one of the minimum in Eq. (15)) but also the objective function of each best counter-hypotheses (the other minimum). A path should then be pruned only if its PED is greater than the current radius r^2 and if this path cannot lead to a better counter-hypothesis. This can be implemented by adding another radius called the hypothetical radius constraint.

One of the most advanced ASIC-implemented depth-first reported so far uses a two-dimensional Schnorr-Euchner enumeration. This implementation reaches a throughput higher than 600 Mb/s for the soft-output version and exceeds 1.2 Gb/s for the hard-output one while keeping an excellent energy efficiency [21]. The high throughput is achieved by using several SD cores in parallel to decode several vectors simultaneously.

6.3 Breadth-first tree-search detectors: K-best and M-algorithm

Breadth-first detectors drop out of optimality for better implementability. Indeed, they address the two main issues of the depth-first paradigm: the unpredictable complexity that depends on the SNR through r^2 and the depth-and-backtrack enumeration that prevents the use of hardware pipelines. Breadth-first detectors achieve this goal by removing the pruning criterion and always keep the same number of paths instead. At each level, the detector compares all the current paths' PED and keeps only the best ones. This number is traditionally called K for the K-best algorithm [22] or M for the M-algorithm [23]. A recent work mixed this approach with the upper-bound radius from death-first to prune even more path per level and reduce the complexity further [24]. This method converges to the breadth-first if all PED are always under the upper-bound, but if some PEDs overgrow, it can reduce the number of surviving paths.

The restricted number of surviving paths induces that the right one can be pruned early if its PED has grown too quickly in the early levels. This is the reason for the optimality loss. Then, some detectors implement a post-detection SNR criterion to reorder the tree levels such that the most certain one is at the top. Thus, the right path is less likely to be pruned by mistake in the early stages. Thanks to this reordering, the K-best algorithm performs very well yet, not optimally in a mathematical sense.

From a hardware implementation perspective, breadth-first tree-search detectors are very efficient. They do not require any backtrack so that an expanded node can be safely deleted as it will never be revisited. Moreover, the number of visited nodes per level is fixed. Thus, ASIC can embed the exact amount of resources required. These two characteristics allow the construction of efficient hardware pipelines that substantially increase the throughput. K-best can achieve at least the same throughput as depth-first without the need for parallel cores. Hard-output implementations exceeding 2.5 Gb/s are reported using the breadth-first paradigm [25]. Another study focused on energy efficiency designed a breadth-first variant that can handle both the channel noise and the hardware noise generated by the voltage over the scaling method in memories [26].

Breadth-first detectors can provide soft-output using the max-log approximation and a list. This list approach is used by several detectors, including some other tree-search algorithm and detectors from other families. The point of this approach is to produce a list Γ of point associated with their objective function and to approximate Eq. (15) as

$$L_i \approx \frac{1}{2\sigma^2}\left(\min_{x\in\Gamma\cap\chi_i^0}\|\mathbf{y}-\mathbf{Hx}\|^2 - \min_{x\in\Gamma\cap\chi_i^1}\|\mathbf{y}-\mathbf{Hx}\|^2\right). \tag{16}$$

For most breadth-first algorithms, Γ is the list of all completed paths [22, 25].

6.4 Best-first tree-search detector: fast descent tree-search and parallel tree-search

Best-first detectors are also sometimes called metric-first. The basic idea besides this enumeration is to not favor depth or breadth over each other. Instead, the node with the best PED is expanded, regardless of its level. The best-first algorithm keeps a node pool with nodes to be expanded. First, the pool is initialized with the root node. Then, at each iteration, the node with the lower PED is popped out from the pool, and all its children are computed and pushed in the pool unless they are leaves. If they are, a comparison with the best leaf so far allows us to keep track of the best result. When a leaf is reached, its objective function may be used as an upper-bound to prune the pool for each node with a PED higher than the new reference. The detection ends when the pool is empty.

This simple method quickly overfills the pool as several nodes are added when only one is popped out. Rather than providing a huge pool to contain all the nodes, improvement is to convert the φ-ary tree (with φ the constellation size) to a first-child next-sibling binary tree [27]. This method is called the modified best-first algorithm (MBF). With this variant, the only nodes added in the pool after an expansion are the best child and the best yet-to-visit siblings. Then, the growth rate of the pool size is controlled. However, this method struggles to provide a full path solution quickly as the popped out node is the one with the lower PED that is often close to the root. To solve this issue, a variant implementation called MBF fast descent (MBF-FD) changes the expansion rule. When a node is expanded, the process goes through the best child until reaching a leaf, pushing in all best siblings along the way [28].

Recently, a best-first algorithm ASIC is reported to reach 800 Mb/s in a soft-input soft-output framework [29]. The modified algorithm, called cross-level parallel tree-search, splits the pool node into several pools, one per level. At each iteration, a node from each pool in popped out expanded using the best-child/best-sibling framework, and the new nodes are pushed in the according pool. Moreover, the presented detector prune nodes in each pool using the upper-bound to keep only the one that can improve the result or the counter-hypothesis (see Section 6.2 for details). The slit pool helps the parallelization process so that this algorithm variant is very suitable for hardware implementation.

7. Other unusual detectors: bioinspired and geometrical detectors

7.1 Deep neural MIMO detection: learning to detect

The rise of deep learning leads to the search for an efficient neural network to solve the detection problem such as DetNet [30]. This network is inspired by the projected gradient descent algorithm that is a major option to solve convex

optimization. It is trained for both static channel (**H** is fixed) and on a time-varying channel (the same condition as previously). As the errors are sometimes unavoidable due to a bad channel realization, the loss function should not be the objective function. Thus, the DetNet designers opt for a

$$\sum_{k=1}^{L} \log(k) \frac{\|\mathbf{x} - \mathbf{x}_k\|}{\|\mathbf{x} - \mathbf{x}_{\mathrm{ZF}}\|} \tag{17}$$

with $\mathbf{x}_{\mathrm{ZF}}$ the result of ZF detection and $\mathbf{x}_k$ the detected symbol of the kth layer. Then, the normalization with the ZF result avoids over-penalizing the situation with bad **H** realization. Moreover, the logarithm weights the result from each layer to give more credit to the final ones.

7.2 Bioinspired detectors

The most studied bioinspired decoders fall into two categories: ant colony optimizations (ACO) and particle swarm optimizations (PSO) that include the firefly algorithm (FA). These techniques are often very complex compared to the previously described algorithms, but they claim to be resilient to challenging conditions. Bioinspired algorithms should be able to decode messages with imperfect CSI, or the data streams are correlated.

ACO-based detectors simulate several ants that choose a path randomly to follow with a nonuniform probability function [31]. Each antenna is processed independently. At each iteration, an ant selects the symbol $s \in \Phi$ with the probability

$$p_s = \frac{\tau_s^{\alpha} \eta_s^{\beta}}{\sum_{s \in \Phi} \tau_s^{\alpha} \eta_s^{\beta}} \tag{18}$$

with τ_s the pheromone level on the path, η_s an image of the objective function, most of the time through a log-sigmoid function, and (α, β) the two parameters that balance the relative importance of each term. After each iteration, the pheromone level is updated according to the following principle: the better the objective function the ant achieves, the more pheromone it dropped off. Thus, the ant selects more often the path that seems more promising regarding the objective function and the previous runs while preserving some chance of exploring a new path.

FA-based detectors simulate several fireflies that try to find the best mating partner. The objective function determines the attractiveness of a firefly. Then a firefly goes toward more attractive congeners biased with a random influence to promote exploration [32]. Some FA variant implements a memory effect that makes it even closer to a PSO-based algorithm [33]. This framework is applied to MIMO detection using the QR decomposition described in Section 6.1. The FA represents each symbol to decode as a nest containing as many fireflies as the constellation size. Thus, the fireflies have to select a partner in each nest from the last symbol to the first based on the biased attractiveness. When the firefly population searched all the nests, the best path represents the decoded symbol vector. FA-based detectors can be related to tree-search-based algorithm with a randomness exploration and a fixed number of path allowed.

7.3 Geometrical detectors

Geometrical detectors are based on a two-step process: an exploration to find a small set of promising solutions and an exploitation to improve this set at a small cost. It follows the traditional approach in nonconvex optimization to perform

simple descents that can be stuck in local optimums from several starting points. Geometrical detectors use a real-valued model and the singular value decomposition (SVD) rather than the QR one [34]. Let us rewrite the objective function by introducing the SVD of $\mathbf{H} = \mathbf{UDV}^T$ with $\mathbf{U}$ and $\mathbf{V}$ two orthogonal matrices and $\mathbf{D}$ the diagonal matrix containing the singular values $\{\lambda_i : 1 \leq i \leq n\}$ in ascending order.

Consequently, the objective function can be rewritten as

$$\|\mathbf{y} - \mathbf{Hx}\|^2 = \left(\mathbf{V}^T(\mathbf{x} - \mathbf{x}_0)\right)^T \mathbf{DU}^T\mathbf{UD}\left(\mathbf{V}^T(\mathbf{x} - \mathbf{x}_0)\right) \tag{19}$$

using $\mathbf{x}_0$ from Eq. (3). As the vectors of $\mathbf{V}$, named $\{\mathbf{v}_i : 1 \leq i \leq n\}$, constitute a basis, we can define $\{\alpha_i : 1 \leq i \leq n\}$ the coordinates of $\mathbf{x} - \mathbf{x}_0$ on this basis. Using the orthogonality of $\mathbf{U}$ and $\mathbf{V}$ and the diagonality of $\mathbf{D}$, Eq. (19) leads to

$$\|\mathbf{y} - \mathbf{Hx}\|^2 = \sum_{i=1}^{n} \alpha_i^2 \lambda_i^2. \tag{20}$$

Let Δ_i be the straight line passing through x_0 and directed by $\mathbf{v}_i$. One can note that Eq. (20) highlights that the objective function grows more slowly along the first Δ_i rather than along the last ones so that promising points must be around these first straight lines. Then the solution is most likely to be found along this line. The geometrical exploration step is then performed, selecting some points near the first Δ_i. Then a straightforward descent algorithm is performed by looking for the best point in the close neighborhood until convergence.

A soft-output version of this algorithm is possible using the max-log approximation and the list approach detailed in [35], Section 5.2. A field-programmable gate array (FPGA) implementation has recently been proposed. This groundwork points out that geometrical detectors may achieve good performance in the future yet being far from mature at that point [36].

Detector	BER	Complexity	Comment
ML	Optimal	Dramatically complex	
ZF	Very poor	Very simple	Best linear detector regarding SNR criterion
MMSE	Poor	Simple	Best linear detector regarding SINR criterion
SIC/OSIC	Good	Rather complex	Best when there is a clear ranking in the quality of each data stream
PIC	Good	Rather complex	Best when all data streams have the same quality level
Depth-first	Optimal	Very complex	
Breadth-first	Good	Rather complex	Possible trade-off between BER and complexity via the number of surviving paths
Best-first	Good	Less complex	
Deep neural	Good	Rather complex	Possible trade-off between BER and complexity via the number of layers
Bioinspired	Good	Very complex	Resilient to imperfect CSI and channel correlation
Geometrical	Rather good	Rather complex	Possible trade-off between BER and complexity via the number of descents

Table 1.
Summary of all detectors described in this chapter.

8. Conclusions and summary

MIMO detection is a well-studied problem that has been tackled from several perspectives. The mathematical interpretation, as a combinatorial optimization problem, leads to the optimal and linear detectors. From the signal processing perspective, detecting a signal means improving the SNR or SINR so that the direct answer is to cancel the interference and to remove the noise. From an algorithmic perspective, the detection problem is the search for the best path in a weighted tree that relies on some well-known algorithms. Other sources of inspiration, such as nature or geometry, provide some interesting perspectives. These paradigms and the associated detectors are summed up in **Table 1**, and we compare all of them according to the BER-complexity trade-off.

All these perspectives shed a different light on the problem, leading to fruitful experimentation. Indeed, some methods take inspiration from others to keep on improving. Therefore, some improvement axes remain open, for instance, the permanent decrease of complexity with equal performance, the development for efficient hardware implementations, or the optimization of the interaction with decoders to exploit channel codings better.

Conflict of interest

The authors declare no conflict of interest but to research and publish in this area, in particular on geometrical detectors.

Nomenclature

Memoryless	the channel outputs only depend on its state and its inputs.
Linear	the channel outputs are linear combinations of the inputs.
Flat	the channel coherence bandwidth is large enough to consider that all frequencies experience the same transfer function.
Block fading	the channel states vary slow enough to be considered constant over many symbols named coherence block.

Abbreviations

ACO	ant colony optimization
ASIC	application-specific integrated circuit
CDMA	code-division multiple access
CLPS	closest lattice-point search
CSI	channel state information
FA	firefly algorithm
FPGA	field-programmable gate array
LLR	log-likelihood ratio
MBF	modified best-first
MBF-FD	modified best-first fast descent
ML	maximum likelihood
MMSE	minimum mean-square error
OSIC	ordered successive interference cancelation
PED	partial Euclidean distance
PIC	parallel interference cancelation

PSO	particle swarm optimization
QAM	quadrature amplitude modulation
SD	sphere decoding
SE	Schnorr-Euchner
SDM	space-division multiplexing
SIC	successive interference cancelation
SIR	signal-to-interference ratio
SINR	signal-to-noise-plus-interference ratio
SNR	signal-to-noise ratio
SVD	singular value decomposition
ZF	zero forcing

IntechOpen

Author details

Bastien Trotobas, Amor Nafkha*† and Yves Louët†
SCEE/IETR UMR CNRS 6164, CentraleSupélec, Cesson Sévigné, France

*Address all correspondence to: amor.nafkha@centralesupelec.com

† These authors contributed equally.

IntechOpen

© 2020 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. CC BY

References

[1] Yang S, Hanzo L. Fifty years of MIMO detection: The road to large-scale MIMOs. IEEE Communication Surveys and Tutorials. 2015;**17**(4):1941-1988. ArXiv: 1507.05138

[2] Idowu-Bismark OB, Kennedy O, Idachaba F, Atayero AA. A primer on MIMO detection algorithms for 5G communication network. International Journal on Communications Antenna and Propagation. 2018;**8**(3):194

[3] Saleh AAM, Valenzuela R. A statistical model for indoor multipath propagation. IEEE Journal on Selected Areas in Communications. 1987;**5**(2): 128-137

[4] Debbah M, Muller RR. MIMO Channel modeling and the principle of maximum entropy. IEEE Transactions on Information Theory. 2005;**51**(5): 1667-1690

[5] Micciancio D. The hardness of the closest vector problem with preprocessing. IEEE Transactions on Information Theory. 2001;**47**(3): 1212-1215

[6] Kay SM. Fundamentals of Statistical Signal Processing. Englewood Cliffs, NJ: Prentice-Hall; 1993

[7] Wolniansky PW, Foschini GJ, Golden GD, Valenzuela RA. V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel. In: International Symposium on Signals, Systems, and Electronics; 1998. pp. 295-300

[8] Riadi A, Boulouird M, Hassani MM. ZF/MMSE and OSIC detectors for UpLink OFDM massive MIMO systems. In: IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology; 2019. pp. 767-772

[9] Chin WH, Constantinides AG, Ward DB. Parallel multistage detection for multiple antenna wireless systems. Electronics Letters. 2002; **38**(12):597

[10] Hochwald BM, ten Brink S. Achieving near-capacity on a multiple-antenna channel. IEEE Transactions on Communications. 2003;**51**(3):389-399

[11] Studer C, Fateh S, Seethaler D. ASIC implementation of soft-input soft-output MIMO detection using MMSE parallel interference cancellation. IEEE Journal of Solid-State Circuits. 2011; **46**(7):1754-1765

[12] Tuchler M, Singer AC, Koetter R. Minimum mean squared error equalization using a priori information. IEEE Transactions on Signal Processing. 2002;**50**(3):673-683

[13] Koch W, Baier A. Optimum and sub-optimum detection of coded data disturbed by time-varying intersymbol interference. In: IEEE Global Telecommunications Conference and Exhibition; 1990. pp. 1679-1684

[14] Robertson P, Villebrun E, Hoeher P. A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain. In: IEEE International Conference on Communications, Vol. 2; 1995. pp. 1009-1013

[15] Tomasoni A, Ferrari M, Gatti D, Osnato F, Bellini S. A low complexity turbo MMSE receiver for W-LAN MIMO systems. In: IEEE International Conference on Communications, Vol. 9; 2006. pp. 4119-4124. ISSN: 1938-1883

[16] Fincke U, Pohst M. Improved methods for calculating vectors of short length in a lattice, including a complexity analysis. Mathematics of Computation. 1985;**44**(170):463-471

[17] Schnorr CP, Euchner M. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. Mathematical Programming. 1994;**66**(1–3):181-199

[18] Agrell E, Eriksson T, Vardy A, Zeger K. Closest point search in lattices. IEEE Transactions on Information Theory. 2002;**48**(8):2201-2214

[19] Jalden J, Ottersten B. On the complexity of sphere decoding in digital communications. IEEE Transactions on Signal Processing. 2005;**53**(4): 1474-1484

[20] Studer C, Bolcskei H. Soft–input soft–output single tree-search sphere decoding. IEEE Transactions on Information Theory. 2010;**56**(10): 4827-4842

[21] Chuang PIJ, Sachdev M, Gaudet VC. VLSI implementation of high-throughput, low-energy, configurable MIMO detector. In: IEEE International Conference on Computer Design; New York, USA; 2015. pp. 335-342

[22] Guo Z, Nilsson P. Algorithm and implementation of the K-best sphere decoding for MIMO detection. IEEE Journal on Selected Areas in Communications. 2006;**24**(3):491-503

[23] Wong KKY, McLane PJ. A low-complexity iterative MIMO detection scheme using the soft-output M-Algorithm. In: IST Mobile Summit; 2005. p. 5

[24] Choi SJ, Shim SJ, You YH, Cha J, Song HK. Novel MIMO detection with improved complexity for near-ML detection in MIMO-OFDM systems. IEEE Access. 2019;7:60389-60398

[25] Yan Z, He G, Ren Y, He W, Jiang J, Mao Z. Design and implementation of flexible dual-mode soft-output MIMO detector with channel preprocessing. IEEE Transactions on Circuits and Systems I: Regular Papers. 2015;**62**(11): 2706-2717

[26] Khairy MS, Shen CA, Eltawil AM, Kurdahi FJ. Algorithms and architectures of energy-efficient error-resilient MIMO detectors for memory-dominated wireless communication systems. IEEE Transactions on Circuits and Systems I: Regular Papers. 2014; **61**(7):2159-2171

[27] Knuth DE. The Art of Computer Programming: Volume 1: Fundamental Algorithms. 3rd ed. Boston: Addison-Wesley Professional; 1997

[28] Liao CH, Wang TP, Chiueh TD. A 74.8 mW soft-output detector IC for 8 x 8 spatial-multiplexing MIMO communications. IEEE Journal of Solid-State Circuits. 2010;**45**(2):411-421

[29] He G, Zhang X, Liang Z. Algorithm and architecture of an efficient MIMO detector with cross-level parallel tree-search. IEEE Transactions on Very Large Scale Integration Systems. 2020; **28**(2):467-479

[30] Samuel N, Diskin T, Wiesel A. Deep MIMO detection. In: International Workshop on Signal Processing Advances in Wireless Communications; 2017. pp. 1-5. ISSN: 1948-3252

[31] Marinello JC, Abrão T. Lattice reduction aided detector for MIMO communication via ant colony optimisation. Wireless Personal Communications. 2014;**77**(1):63-85

[32] Datta A, Bhatia V. A near maximum likelihood performance modified firefly algorithm for large MIMO detection. Swarm and Evolutionary Computation. 2019;**44**:828-839

[33] Nasiri B, Meybodi MR. History-driven firefly algorithm for optimisation in dynamic and uncertain environments. International Journal of

Bio-Inspired Computation. 2016; **8**(5):326

[34] Nafkha A, Boutillon E, Roland C. Quasi-maximum-likelihood detector based on geometrical diversification greedy intensification. IEEE Transactions on Communications. 2009; **57**(4):926-929

[35] Trotobas B, Nafkha A. Fixed complexity soft-output detection algorithm through exploration and exploitation processes. In: AICT 2018; Barcelona, Spain; 2018. pp. 89-93

[36] Trotobas B, Akourim Y, Nafkha A, Loüet Y, Weiss J. Evaluation of the complexity, performance and implementability of geometrical MIMO detectors: The example of the exploration and exploitation list detector. International Journal on Advances in Telecommunications. June 2020;**13**(1&2)