Maximum Likelihood Low-Complexity GSM Detection for Large MIMO Systems

Victor M. Garcia-Molla[a,1], F. J. Martínez-Zaldívar[a], M. Angeles Simarro[a], Alberto Gonzalez[a]

[a] *Victor M. Garcia-Molla, F. J. Martínez-Zaldívar, M. Angeles Simarro, and Alberto Gonzalez are with Universitat Politècnica de València, SPAIN e-mail: {vmgarcia@upv.es, mdesiha@dcom.upv.es, fjmartin@dcom.upv.es, agonzal@dcom.upv.es}*

Abstract

Hard-Output Maximum Likelihood (ML) detection for Generalized Spatial Modulation (GSM) systems involves obtaining the ML solution of a number of different MIMO subproblems, with as many possible antenna configurations as subproblems. Obtaining the ML solution of all of the subproblems has a large computational complexity, especially for large GSM MIMO systems. In this paper, we present two techniques for reducing the computational complexity of GSM ML detection.

The first technique is based on computing a box optimization bound for each subproblem. This, together with sequential processing of the subproblems, allows fast discarding of many of these subproblems. The second technique is to use a Sphere Detector that is based on box optimization for the solution of the subproblems. This Sphere Detector reduces the number of partial solutions explored in each subproblem. The experiments show that these techniques are very effective in reducing the computational complexity in large MIMO setups.

Keywords: MIMO, Signal Detection, Maximum Likelihood Detection, GSM

---

*Corresponding author

Email address: vmgarcia@upv.es (Victor M. Garcia-Molla)

1. Introduction

GSM (Generalized Spatial Modulation) is a recent transmission technique in MIMO (Multiple Input-Multiple Output) systems [1, 2]. The distinguishing feature of this technique is that only a subset of the available transmit antennas are activated for each transmission. The subsets of antennas (named configurations in this paper) that can be used to transmit are fixed, numbered, and known in advance by the receiver. As a consequence, the configuration of active antennas in each transmission is used to convey extra bits.

GSM has several advantages with respect to conventional MIMO systems. For instance, this technique alleviates the problem of hardware complexity and inter-antenna synchronization [3]. A drawback of GSM is that the detection process becomes more involved. The receiver needs to detect both the transmitted symbols and the configuration of antennas chosen for the transmission.

Most methods proposed for GSM detection have two stages. First, the antenna configuration is detected. Then, the transmitted symbols can be estimated by solving a smaller MIMO detection problem, involving only the configuration of active antennas. Among the existing methods, several are fast (but suboptimal) methods, like the methods proposed in [1, 4, 5, 6, 7]. Another popular idea that has sometimes been proposed in two-stage detectors is the use of extended constellations, which include "0" as a symbol in order to identify the inactive antennas. This is often used in methods that perform some kind of tree detection, using schemes that are popular in standard MIMO detection such as Fixed Complexity Sphere Decoding [8] or the K-best method [9].

Maximum Likelihood (ML) detection for GSM problems offers the optimum performance in terms of detection accuracy. Thus, ML-GSM provides an upper bound on the attainable detection accuracy and it is of great interest for researchers. However, it has been considered unfeasible for large MIMO systems because of the high computational complexity of exhaustive detectors. One of the known proposals for GSM-ML detection was described in [10]. In that paper, the detection of the antenna configuration and the detection of the symbols are carried out jointly. However, this algorithm depends on obtaining the Cholesky decomposition of the channel matrix, which can be computed only if the channel matrix is square. This is a serious limitation to its applicability.

Recently a new ML method for GSM problems has been proposed in [11]. That method uses successive applications of a standard MIMO ML Sphere Decoder (SD) (one for each valid antenna configuration). The method features the use of an adjustable radius in all of the configurations as well as a previous ordering of the configurations so that the ones with a higher likelihood of being the correct configuration are processed first. This method is very efficient for GSM-MIMO setups with moderate-size constellations (4QAM, 16QAM) and/or a small to moderate number of antennas ($2-4$ active antennas in each transmission, $4-6$ receive antennas). However, as will be shown later, the computational cost renders the method impractical for larger problems.

In this paper, we propose two techniques that are designed to be used along with the solving strategy described in [11]. Both techniques are based on box optimization [12, 13, 14]. The first proposal is the computation of a new bound, which is based on box optimization. This bound can be used to reduce the number of subproblems/configurations that must be studied to achieve the GSM ML solution.

The second proposal is to switch from the standard Schnorr-Euchner decoder used in [11] to the box optimization-based sphere decoder described in [14]. This Sphere Decoder reduces the number of partial solutions that must be examined in each configuration.

The proposed techniques have been tested in Matlab [15] under four large MIMO setups. We use the number of information bits per transmission (spectral efficiency) as a measure of the size of the GSM-MIMO problems. The results show that, for large GSM-MIMO problems, both proposals are necessary in order to perform ML detection with acceptable computing times.

2. Problem description

Let us consider a GSM-MIMO system with n_T transmit antennas, n_A active antennas, and n_R receive antennas, with $n_A < n_T$. In this paper, we explore the case $n_T > n_R$ and $n_R \geq n_A$.

Given that n_A is the number of antennas that can be activated in each transmission, then the total number of possible subsets of active antennas is $\binom{n_T}{n_A}$. Usually, not all of the possible configurations are considered as valid configurations (i.e., not all possible configurations are used for transmission). If the selection of antenna will convey n_b bits, then $n_c = 2^{n_b}$ valid antenna configurations are selected. Each configuration can be described as a set of antenna indexes, $\{i_{k_1}, i_{k_2}, \cdots, i_{k_{n_A}}\}, 1 \leq i_{k_j} \leq n_T, j = 1, .., n_A$.

Let Ω be the constellation of complex symbols, of size $|\Omega| = L$. Hence, each symbol carries $\log_2 L$ code bits each. Thus, the bits to be transmitted are grouped in blocks of $n_A \cdot log_2(L)$ $+log_2(n_c)$. The first $n_A \cdot log_2(L)$ bits are mapped into a symbol vector $\mathbf{s} = [s_1, ..., s_{n_A}]$. The remaining $n_b = log_2(n_c)$ bits are used to select the antenna configuration.

Let $\mathbf{H} \in \mathbb{C}^{n_R \times n_T}$ be the MIMO overall channel matrix, with independent elements $h_{ij} \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$. The k-th antenna configuration (with antennas $\{i_{k_1}, i_{k_2}, \cdots, i_{k_{n_A}}\}$) defines its corresponding channel submatrix $\mathbf{H_k}$, which is formed by the columns $\{i_{k_1}, i_{k_2}, \cdots, i_{k_{n_A}}\}$ of the overall channel matrix $\mathbf{H}$. If the transmission is carried out through the k-th configuration, the received vector can be written as

$$\mathbf{y} = \mathbf{H_k} \cdot \mathbf{s} + \mathbf{v}. \tag{1}$$

where $\mathbf{v}$ denotes a white-Gaussian noise (AWGN) complex vector with elements $v_i \sim \mathcal{N}(\mathbf{0}, \sigma^2)$. Thus, the ML detector for the GSM problem can be described as:

$$\{\hat{k}, \hat{s}\} = \underset{k \in \{1, \ldots, n_c\}, \mathbf{s} \in \Omega^{n_A}}{\arg\min} \|\mathbf{y} - \mathbf{H_k} \cdot \mathbf{s}\|^2. \tag{2}$$

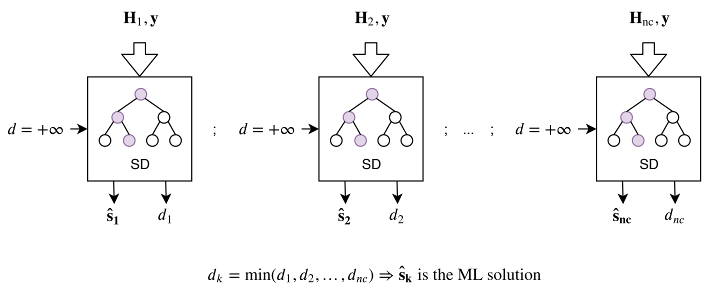

Figure 1: GSM-ML basic detection procedure

3. GSM-ML Detection

Standard ML MIMO detection methods cannot be applied directly to GSM problems when $n_T > n_R$ because it is not possible to obtain the required triangular factorization of the channel matrix. In such cases, the only way available for computing the ML solution (to the best of our knowledge) is to decouple the problem in n_c ML MIMO detection subproblems, one for each antenna configuration:

$$\hat{s}_k = \underset{\mathbf{s}\in\Omega^{n_A}}{\arg\min} \|\mathbf{y} - \mathbf{H_k} \cdot \mathbf{s}\|^2. \tag{3}$$

Equation (3) defines the ML estimator for the k-th antenna configuration. A trivial approach to GSM-ML detection would be to use a standard ML MIMO SD to solve subproblems (3), for all k. By comparing the optimal Euclidean distances $d_k = \|\mathbf{y} - \mathbf{H_k} \cdot \hat{\mathbf{s}}_\mathbf{k}\|^2$ for $k = 1, ..., n_c$, we can obtain the minimal distance, which will indicate the optimal configuration and, therefore, the ML solution.

However, the cost of this procedure is very high because n_c different ML subproblems must be solved.

Figure 1 illustrates the procedure. Each box represents the resolution of a MIMO subproblem (3), returning its ML solution and the associated distance. The GSM-ML solution is the ML solution of the subproblem with the smallest associated distance.

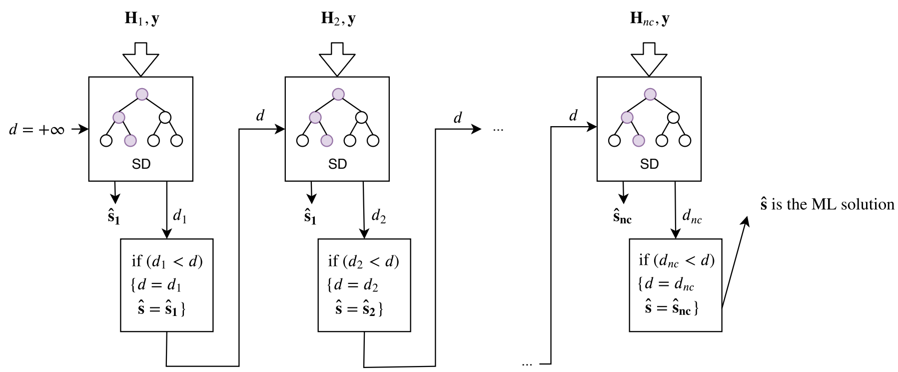

Figure 2: Procedure of sequential detection with adjustable radius.

3.1. Sequential detection with adjustable radius and ordering

The main goal of the idea proposed in [11] is to decrease the computational cost of GSM-ML detection through sequential detection and the use of an adjustable radius across all of the subproblems (3). The idea of the adjustable radius in GSM-ML detection comes from a similar technique that is used in most standard MIMO SD detectors.

In MIMO SD detectors, the initial value of the radius is chosen as the squared Euclidean distance of the best feasible solution obtained so far. MIMO SD detectors search among the possible solutions looking for the one with the smallest Euclidean distance. When a partial solution has a larger distance than the actual radius, this partial solution is discarded. When a solution is found with a distance that is smaller than the radius, the radius is updated as the squared Euclidean distance of the new solution. [16, 17, 18, 19].

The selection of the initial radius has a strong impact in the performance of SD MIMO detectors. When the initial radius is too large, too many partial solutions are examined, with a high computational cost; on the other hand, when the initial radius is smaller than the distances of all of the possible solutions, the detection ends very fast and no solution is returned. [16, 17, 18, 19]

The technique of the adjustable radius was extended to GSM problems in [11], combined with sequential detection. An initial radius d is considered, chosen initially as $+\infty$. Then, the subproblems (3) are solved in order, using a MIMO SD detector with adjustable radius. After the k-th subproblem is solved (returning d_k and $\hat{s}_k$), the radius d is compared with the radius d_k. If $d_k < d$, the radius is updated as d_k and the best solution obtained is updated as $\hat{s}_k$. The new best radius d is then used as the initial radius for the next configuration.

Figure 2 describes the procedure. Let us assume that the solution of the k_{opt}-th configuration is the actual GSM-ML solution and has radius $d_{k_{opt}}$. Then, the distances of all of the possible solutions in subproblems $k_{opt}+1, ..., n_c$ are larger than $d_{k_{opt}}$. Therefore, the SD detector applied to subproblems $k_{opt}+1, ..., n_c$ will not return a new solution (which is correct because the GSM ML solution has already been found) and will end very fast.

If the correct configuration (the configuration whose ML solution is the overall GSM-ML solution) is among the first positions on the list of configurations (i.e., k_{opt} is close to 1), then only a few MIMO ML subproblems must be solved, and the process will be quite efficient. On the other hand, if k_{opt} is close to n_c, then many subproblems must be solved and the process will be slow. Therefore, for the sake of efficiency, the configurations must be reordered so that the correct configuration has a high probability of being located among the first positions.

The ordering method proposed in [11] depends on the QR decompositions [20] of the channel submatrices $\mathbf{H_k} \in \mathbb{C}^{n_R \times n_A}$, $k = 1, ..., n_c$. The QR decomposition of $\mathbf{H_k}$ gives as result a unitary matrix $\mathbf{Q_k} \in \mathbb{C}^{n_R \times n_R}$ and an upper triangular matrix $\mathbf{R_k} \in \mathbb{C}^{n_R \times n_A}$ such that $\mathbf{H_k} = \mathbf{Q_k}\mathbf{R_k}$. Given that the last $n_R - n_A$ rows of $\mathbf{R_k}$ are zeros, the QR decomposition is usually rewritten as:

$$\mathbf{H_k} = \mathbf{Q_k} \cdot \mathbf{R_k} = \mathbf{Q_k} \cdot \begin{pmatrix} \mathbf{R_{k1}} \\ 0 \end{pmatrix} = (\mathbf{Q_{k1}}\mathbf{Q_{k2}}) \cdot \begin{pmatrix} \mathbf{R_{k1}} \\ 0 \end{pmatrix}, \tag{4}$$

where $\mathbf{R_{k1}} \in \mathbb{C}^{n_A \times n_A}$, $\mathbf{Q_{k1}} \in \mathbb{C}^{n_R \times n_A}$ and $\mathbf{Q_{k2}} \in \mathbb{C}^{n_R \times (n_R - n_A)}$. Given a received signal $\mathbf{y}$, for any sent signal $\mathbf{s}$, the Euclidean distance in the k-th configurations is $\|\mathbf{y} - \mathbf{H_k} \cdot \mathbf{s}\|^2$. Given that $\mathbf{Q_k}$ is unitary and using the QR decomposition for rectangular channel submatrices $\mathbf{H_k} \in \mathbb{C}^{n_R \times n_A}$, with

$n_R > n_A$, we have:

$$\|\mathbf{y} - \mathbf{H_k} \cdot \mathbf{s}\|^2 = \|\mathbf{Q_k^H} \cdot (\mathbf{y} - \mathbf{Q_k} \cdot \mathbf{R_k} \cdot \mathbf{s})\|^2 = \| \begin{pmatrix} \mathbf{Q_{k1}^H} \\ \mathbf{Q_{k2}^H} \end{pmatrix} \cdot \mathbf{y} - \begin{pmatrix} \mathbf{R_{k1}} \\ 0 \end{pmatrix} \cdot \mathbf{s}\|^2$$
$$= \| \mathbf{Q_{k1}^H} \cdot \mathbf{y} - \mathbf{R_{k1}} \cdot \mathbf{s}\|^2 + \| \mathbf{Q_{k2}^H} \cdot \mathbf{y}\|^2. \tag{5}$$

The ordering method proposed in [11] sorts the configurations depending on the value of the term $\| \mathbf{Q_{k2}^H} \cdot \mathbf{y}\|^2$. However, if $n_R = n_A$ (which is a reasonable setup), this term does not exist and therefore cannot be used for ordering.

We have preferred to use a similar ordering method, which was proposed in [5] as the basis of a suboptimal, non-ML GSM detection method. It amounts to obtaining the zero-forcing (ZF) estimator of each subproblem, i.e., solving the unconstrained versions of the problem (3),

$$\mathbf{zf_k} = \arg\min_{\mathbf{s} \in \mathbb{C}^{n_A}} \|\mathbf{y} - \mathbf{H_k} \cdot \mathbf{s}\|^2, \tag{6}$$

obtaining the unconstrained solutions $\mathbf{zf}_k$. These solutions are readily found using the QR decompositions of the matrices $\mathbf{H_k}$ (first compute $\mathbf{z_k} = \mathbf{Q_k^H} \cdot \mathbf{y}$, and then solve the upper triangular linear system $\mathbf{R_{k1}} \cdot \mathbf{zf_k} = \mathbf{z_k}(1 : n_A)$, obtaining $\mathbf{zf_k}$, the solution of (6); the unconstrained solutions $\mathbf{zf}_k$ can be quantized, forming the ZF estimators $\hat{\mathbf{zf}}_\mathbf{k}$). The squared Euclidean distances of these estimators are computed: $dz_k = \|\mathbf{y} - \mathbf{H_k} \cdot \hat{\mathbf{zf}}_\mathbf{k}\|^2$. Then, the configurations are sorted according to the distances dz_k, from smallest to largest. This method does not have any restrictions and has worked quite well in all of the tested cases.

Next, we describe the base algorithm proposed in [11] as pseudo-code. The first step would be to compute the QR decompositions of the matrices $\mathbf{H_k} = \mathbf{Q_k} \cdot \mathbf{R_k}$. These QR decompositions can be reused as long as the channel matrix does not change.

Algorithm 1 implements the ordering phase and is also used to obtain an initial radius and the vectors $\mathbf{z_k} = \mathbf{Q_k^H} \cdot \mathbf{y}$, which are needed in the search phase.

Algorithm 1 ML GSM ordering phase

Input:

- $n_c \in \mathcal{N}$ number of antenna configurations

- channel submatrices $\mathbf{H_k} \in \mathbf{C}^{n_R \times n_A}, k = 1, ..., n_c$

- unitary submatrices $\mathbf{Q_k} \in \mathbf{C}^{n_R \times n_R}, k = 1, ..., n_c$

-upper triangular submatrices $\mathbf{R_k} \in \mathbf{C}^{n_R \times n_A}, k = 1, ..., n_c$

- received signal $\mathbf{y} \in \mathbf{C}^{n_R}$

Output:

- ordering vector $v_{order} \in \mathcal{N}^{n_c}$

- initial radius r

- vectors $\mathbf{z_k}$

/* Start*/

if $n_R > n_A$ **then**

    for $k = 1$ **to** nc **do**

        $\mathbf{z_k} = \mathbf{Q_k^H} \cdot \mathbf{y};$

        $\mathbf{zf}_k = (\mathbf{R_{k1}})^{-1} \cdot \mathbf{z_k}(1 : n_A);$

        $\hat{\mathbf{zf}}_\mathbf{k} = quantized(\mathbf{zf}_k);$

        $d_k = \|\mathbf{y} - \mathbf{H_k} \cdot \hat{\mathbf{zf}}_\mathbf{k}\|^2$

    end for

else if $n_R = n_A$ **then**

    for $k = 1$ **to** nc **do**

        $\mathbf{z_k} = \mathbf{Q_k^H} \cdot \mathbf{y};$

        $\mathbf{zf}_k = \mathbf{R_k^{-1}} \cdot \mathbf{z_k};$

        $\hat{\mathbf{zf}}_\mathbf{k} = quantized(\mathbf{zf}_k);$

        $d_k = \|\mathbf{y} - \mathbf{H_k} \cdot \hat{\mathbf{zf}}_\mathbf{k}\|^2$

    end for

end if

$\mathbf{v_{order}} = sort(\mathbf{d});$ /* v_{order} is a vector of indices that sort the vector $\mathbf{d}$ from smallest to largest */

$r = d(v_{order}(1));$

For the second phase (Algorithm 2, successive search among configurations), an implementation of an ML Sphere Decoder is needed. We used a Schnorr-Euchner decoder [17], named SD-SE, taking as the input arguments the triangular matrix coming from the QR decomposition, the received signal (premultiplied by the transposed orthogonal matrix coming from the QR decomposition), and the initial radius. The usual preprocessing (computation of the $\mathbf{z_k}$ vectors) is carried out in the ordering phase (Algorithm 1).

There is an important difference between the case $n_R = n_A$ and the case $n_R > n_A$. In the case $n_R = n_A$, equation (5) simplifies to $\|\mathbf{y} - \mathbf{H_k} \cdot \mathbf{s}\|^2 = \| \mathbf{Q_1^{H_k}} \cdot \mathbf{y} - \mathbf{R_{k1}} \cdot \mathbf{s}\|^2$. In this case, the initial radius is the radius that was computed in previous iterations. However, in the case $n_R > n_A$, an extra term appears in equation (5), $\| \mathbf{Q_{k2}^{H}} \cdot \mathbf{y}\|^2$. This term is fixed in the sense that it does not depend on the sent signal (it does not depend on the solution of the sphere decoder). Then, this fixed amount can be subtracted from the radius, obtaining a tighter bound. This is done in line 18 of Algorithm 2. After including these details, Algorithm 2 is as follows:

Algorithm 2 ML GSM search phase

Input:
- channel submatrices $\mathbf{H_k} \in \mathbf{C}^{n_R \times n_A}, k = 1, ..., n_c$
- unitary submatrices $\mathbf{Q_k} \in \mathbf{C}^{n_R \times n_T}, k = 1, ..., n_c$
- square upper triangular submatrices $\mathbf{R_{k1}} \in \mathbf{C}^{n_A \times n_A}, k = 1, ..., n_c$
- received signal $\mathbf{y} \in \mathbf{C}^{n_R}$
- ordering vector $v_{order} \in \mathcal{N}^{n_c}$
- initial radius r
- vectors $\mathbf{z_k} = \mathbf{Q_k^H} \cdot \mathbf{y}$
Output:
- index of optimal configuration i_{opt}
- ML solution sol_optim
/*Start*/
$rad = r$;
for $k = 1$ **to** n_c **do**
    $i = v_{order}(k)$
    if $n_R > n_A$ **then**
        $dist\_extra = \|\mathbf{z_i}(\mathbf{n_A+1:n_R})\|^2$ /* $= \| \mathbf{Q_k^H}_2 \cdot \mathbf{y}\|^2$ */
        $dist\_aux = rad - dist\_extra$
    end if
    $x = SD\_SE(\mathbf{R_{i1}}, \mathbf{z_i}(\mathbf{1:n_A}), dist\_aux)$
    $new\_rad = \|\mathbf{y} - \mathbf{H_i} \cdot \mathbf{x}\|^2$
    if $new\_rad <= rad$ **then**
        $rad = new\_rad$
        $sol\_optim = x$
        $i_{opt} = i$
    end if
end for

Algorithms 1 and 2 describe the method proposed in [11], with only the modification of the ordering phase, using the ZF-based ordering described above.

4. Proposed Techniques

In this section, we describe our two proposals for improving the computational cost of Algorithms 1 and 2 in large GSM-MIMO problems.

4.1. Box optimization bound

Box optimization has been proposed as a help for MIMO detection in different papers [12, 13, 14]. Here we have adapted some of the proposals in those papers to the GSM problem.

The first proposal is to compute the solution of the continuous least squares problem for each configuration k:

$$\hat{\mathbf{sr}}_{\mathbf{k}} = \underset{\substack{\mathbf{s}\in\mathbb{C}^{n_A} \\ \min(Re(\Omega))\leq Re(s_i)\leq\max(Re(\Omega))\ ,1\leq i\leq m \\ \min(Im(\Omega))\leq Im(s_i)\leq\max(Im(\Omega))\ ,1\leq i\leq m}}{\arg\min} \left\| \mathbf{H_k}\cdot\mathbf{s}-\mathbf{y}\right\|^2 , \tag{7}$$

where $s_i, 1 \leq i \leq n_A$ are the components of the vector $\mathbf{s}$. This problem is derived from (3), discarding the condition that the components of the solution belong to the constellation Ω.

Compared to (3), this is a continuous problem. The components of the solution vector do not need to belong to Ω; the only restriction is that the search zone be bounded. The limits of the search zone are $[\min(Re(\Omega))$, $\max(Re(\Omega))]$ for the real part of the components of the solution vector $\mathbf{s}$, and $[\min(Im(\Omega)), \max(Im(\Omega))]$ for the imaginary part. This search zone has the form of a box, hence the name of box optimization. An efficient algorithm for solving this problem, which has been adapted to the MIMO problem, was proposed in [14].

The box defined by the constellation and used in (7) contains, by definition, all of the possible solutions of the MIMO problem, i.e., all vectors $\mathbf{s} \in \Omega^{n_A}$. Therefore, for each configuration k, the distance $dr_k = \|\mathbf{y} - \mathbf{H_k}\cdot\hat{\mathbf{sr}}_{\mathbf{k}}\|^2$ (where $\hat{\mathbf{sr}}_{\mathbf{k}}$ is the solution of the minimization problem (7)) is a lower bound of the minimum Euclidean distance $\|\mathbf{y} - \mathbf{H_k}\cdot\mathbf{s}\|^2$ for all of the possible transmitted signals $\mathbf{s}$ in configuration k. Consequently, dr_k is a lower bound of the distance

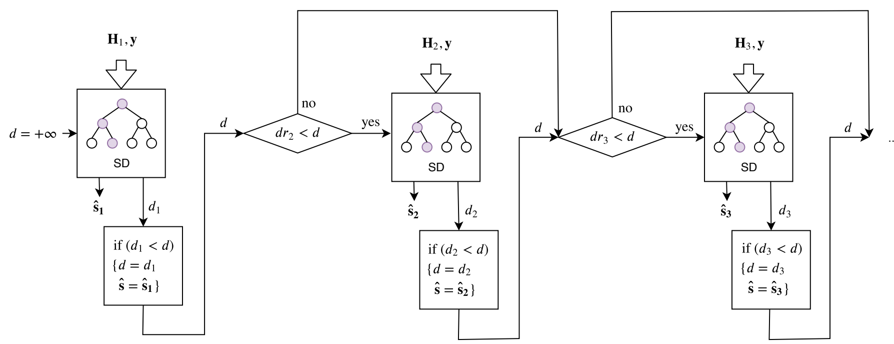

Figure 3: Proposal 1, use of distances dr_k to discard configurations

d_k obtained solving subproblems (3). The cost of the computation of the dr_k distances is smaller than the cost of solving subproblems (3). Moreover, the difference between these two costs is more pronounced when the size of the MIMO systems increases.

We propose computing the distances dr_k prior to the start of the detection, using the box-optimization solver proposed in [14], and using these distances to discard configurations. The computation of these distances could be done in Algorithm 1, after line 17.

These distances can be used as follows: let us consider the sequential process proposed in [11]. When a new configuration/subproblem $k, k > 1$ has to be explored, a radius has already been computed, which is the Euclidean distance of the best solution obtained so far. Then, if the present radius is smaller than the distance dr_k, the k-th configuration can be safely ignored/pruned because the distance of any signal in this subproblem will have a larger Euclidean distance than dr_k.

This procedure is graphically described in Figure 3, and a pseudocode implementation is given in Algorithm 3.

Algorithm 3 ML GSM search phase with Box Optimization aid

Input:
- channel submatrices $\mathbf{H_k} \in \mathbf{C}^{n_R \times n_A}, k = 1, ..., n_c$
- unitary submatrices $\mathbf{Q_k} \in \mathbf{C}^{n_R \times n_T}, k = 1, ..., n_c$
- square upper triangular submatrices $\mathbf{R_{k1}} \in \mathbf{C}^{n_A \times n_A}, k = 1, ..., n_c$
- received signal $\mathbf{y} \in \mathbf{C}^{n_R}$
- ordering vector $v_{order} \in \mathcal{N}^{n_c}$
- initial radius r
- vectors $\mathbf{z_k} = \mathbf{Q_k^H} \cdot \mathbf{y}$
- distances $dr_k, k = 1, ..., n_c$

Output:
- index of optimal configuration i_{opt}
- ML solution sol_optim

/*Start*/
$rad = r$;
for $k = 1$ **to** n_c **do**
    $i = v_{order}(k)$
    if $dr_i < rad$ **then**
        if $n_R > n_A$ **then**
            $dist\_extra = \|\mathbf{z_i}(\mathbf{n_A} + \mathbf{1} : \mathbf{n_R})\|^2$ /* $= \| \mathbf{Q^H_{k\,2}} \cdot \mathbf{y}\|^2$ */
            $dist\_aux = rad - dist\_extra$
        end if
        $x = SD\_SE(\mathbf{R_{i1}}, \mathbf{z_i}(\mathbf{1} : \mathbf{n_A}), dist\_aux)$
        $new\_rad = \|\mathbf{y} - \mathbf{H_i} \cdot \mathbf{x}\|^2$
        if $new\_rad <= rad$ **then**
            $rad = new\_rad$
            $sol\_optim = x$
            $i_{opt} = i$
        end if
    end if
end for

As Figure 3 shows, the search is carried out only in those configurations whose minimal possible distance (dr_k) is smaller than the best radius obtained. This reduces the number of configurations explored. The computational cost of computing the distances dr_k is not negligible and in the case of relatively small problems (bits per transmission of around 20 or less), it is not worthwhile. However, for larger problems ($30-40$ bits per transmission, or more), the reduction in examined configurations is significant enough to counterbalance the extra computational cost coming from the computation of the dr_k distances.

4.2. Box-optimization-aided sphere decoder

The bound described in Section 4.1 allows reasonable computing times for problems with numbers of bits per transmission of around $30 - 40$. However, for larger problems (and especially for large noise), the computational cost per signal can be unpredictably large.

This problem can be alleviated to a certain extent by switching from the Schnorr-Euchner decoder (line 22, Algorithm 3) to the hard-output box-optimization-aided sphere decoder proposed in [14] (available in [21]). This sphere decoder has the same basic structure of the SD-SE decoder; that is, it performs a tree search among partial solutions, looking for the solution with minimum Euclidean distance and using an adjustable radius to improve efficiency. Box optimization is used in the sphere decoder proposed in [14] as a means of discarding partial solutions (i.e., branches of the tree) quite fast, especially in cases with large noise.

The computational cost of this sphere decoder is substantially smaller than standard SD detectors for large problems with large noise because the number of examined partial solutions is greatly reduced. The number of examined partial solutions is also very small for small MIMO problems. However, in this case the computational cost of the box optimizations may be excessive.

In the GSM case, the situation is similar. For small GSM-MIMO systems, this technique may not be worthwhile. However, as shown below, the combined use of the algorithm proposed in [11] and the two proposals described above allows reasonable execution times for GSM-MIMO configurations over 50 bits per transmission.

It is important to note that the distances dr_k can also be used to sort the configurations. However, we preferred to compare the algorithm described in [11] with our proposals using the same ordering in all cases: the ordering based on distances from ZF estimators, which was described at the end of section 3.1. The goal of selecting this ordering was to evaluate our proposals in a setting that was independent from the ordering method.

5. Numerical simulations, Results, and Discussion

We chose four setups to test our proposals, which are shown in Table 1. We estimated the average number of expanded nodes and execution time by means of Monte Carlo simulation. The experiments were carried out varying the signal-to-noise ratio between 5 and 40 dB in increments of 5 dB. We generated 10000 complex Gaussian channel matrices for each value of the signal-to-noise ratio. Each matrix was used for an equivalent channel coherence time of 5 sent GSM-MIMO signals. The tests were carried out running Matlab R2017, using a single core of an Intel Xeon CPU X5680 processor with the Ubuntu operating system.

Table 1: Setups for computer Simulations

	n_T	n_A	n_R	Modulation	n_c	bps/Hz
Setup 1	32	6	6	4QAM	64	18
Setup 2	32	6	6	16QAM	64	30
Setup 3	32	6	6	64QAM	64	42
Setup 4	32	8	8	64QAM	256	56

The proposed methods are ML as long as the detector used for the subproblems (3) is ML. All of the Sphere Decoders used are ML; therefore, all of them display the same Bit-Error-Rate (BER) curve. For a visual comparison, we also implemented the popular, suboptimal OB-MMSE method [4]. Figure 4 shows the BER curves for the setups 1, 2, and 3 (which differ only in the constella-

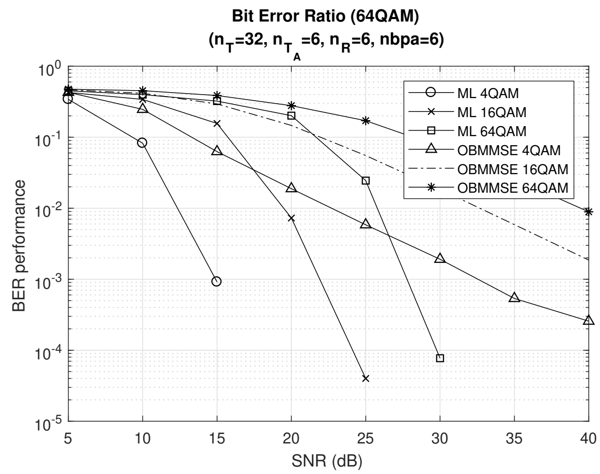

Figure 4: BER comparison between OB-MMSE (non-ML method) and the proposed ML methods

tions used) along with the corresponding curves for the OB-MMSE method. Of course, the usual trade-off between accuracy and computing efficiency appears here; the OB-MMSE method is much faster than any ML method.

Next, we discuss the computational efficiency of the proposed ML techniques. We compare three methods: SE_1, which is the method proposed in [11]; SE_2, which is the same SE_1 method but including the proposal 1 (the box-optimization bound); and BO_1, which is the SE_1 method but including proposals 1 and 2 (the box-optimization bound and box-optimization-aided sphere decoder). In the three methods, the configurations were ordered using the ZF-based method described in 3.1.

5.1. Results with Setups 1 (4QAM) and 2 (16QAM).

Figures 5 and 6 show the average expanded nodes in the smaller Setups 1 and 2. In these two cases, the average number of expanded nodes does not give a reliable indication of the computational cost due to the higher preprocessing cost of the box optimization. This is clearly seen in Figures 7 and 8, where the average computing times per GSM-MIMO symbol are shown. It can be observed that the SE_1 method is faster in these smaller problems than the SE_2

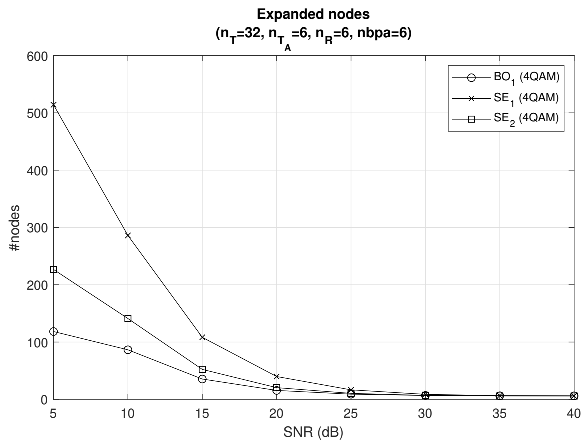

Figure 5: Average expanded nodes in Setup 1 (4QAM) .

and BO$_1$ methods.

5.2. Results with Setups 3 and 4 (64 QAM)

It is well known that when the size of the problem increases, all ML sphere decoders increase the number of nodes (in most cases, exponentially). This problem is even more acute in a GSM setting.

Table 2: Average computing times (seconds) in Setup 3.

SNR	SE$_1$	SE$_2$	BO$_1$
5	1.7E+00	5.8E-01	8.4E-02
10	2.0E-01	1.2E-01	6.3E-02
15	6.9E-02	5.4E-02	5.2E-02
20	2.9E-02	3.8E-02	4.5E-02
25	1.6E-02	2.9E-02	3.5E-02
30	9.9E-03	2.6E-02	2.9E-02
35	9.2E-03	2.5E-02	2.7E-02
40	8.6E-03	2.5E-02	2.6E-02

Tables 2 and 3 show that, in Setup 3, SE$_2$ and BO$_1$ have similar results, which are far better than the results of the SE$_1$ method. The box optimization

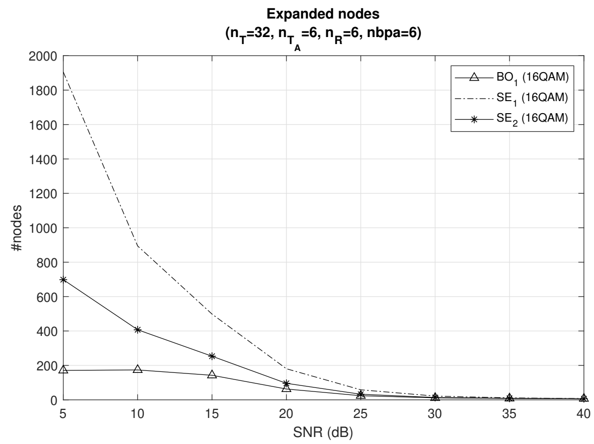

Figure 6: Average expanded nodes in Setup 2 (16QAM).

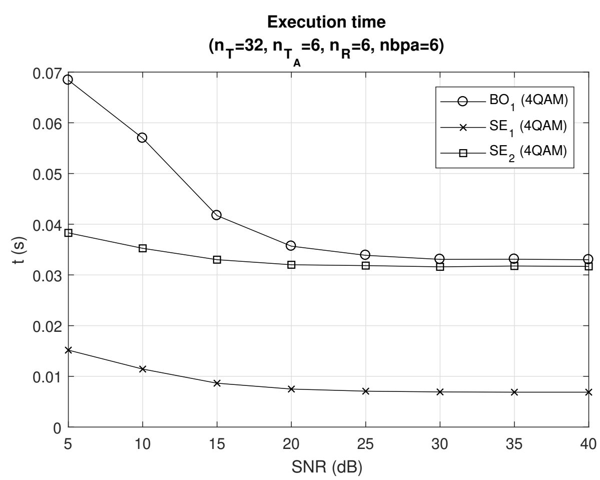

Figure 7: Average computing times (seconds) in Setup 1 (4QAM).

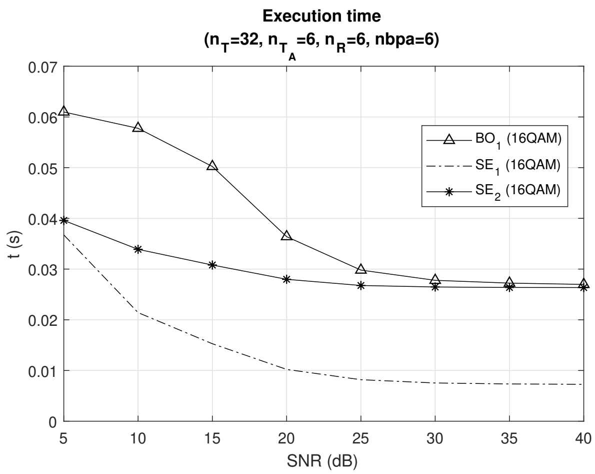

Figure 8: Average computing times (seconds) in Setup 2 (16QAM).

bound is very effective in this case, especially in the low SNR regime. The BO_1 method is very stable in terms of computing time and expanded nodes. In this setup, the differences between methods are very large, which makes a graphic representation inappropriate. This is the reason why the results of this setup have been presented as tables.

The results show a clear trend, favoring simple algorithms for small problems, while sophisticated algorithms perform better in large problems. We checked this idea by performing a larger experiment (Setup 4), which uses 56 bits per transmission. In this setup, the SE_1 method was far too slow. We reduced the experiment to only 500 channel matrices. The computing times obtained are shown in Table 4.

The results show that the computing times needed by SE_1 are not acceptable, especially for large noise. For example, for a SNR of $15dB$, the computational cost of method SE_1 is three times larger than the methods that use our proposals, SE_2 and BO_1. When the noise increases, the difference becomes much larger. We performed another experiment using Setup 4 and 10000 channel matrices, but involving only the methods SE_2 and BO_1. The results are shown in Figures 9 and 10.

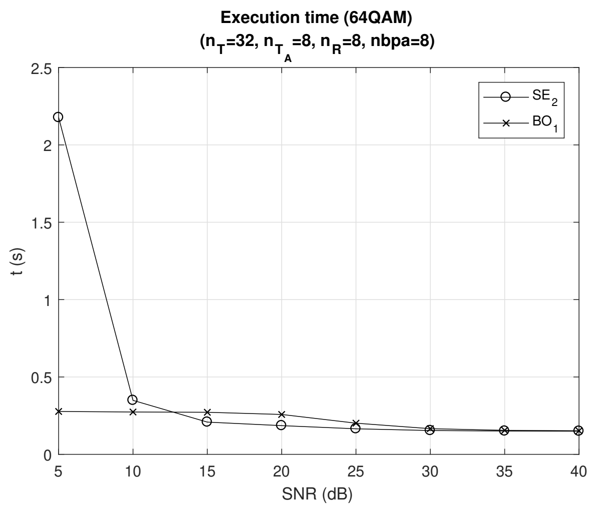

Figure 9: Average computing times (seconds) in Setup 4 (experiment with 10000 channel matrices).

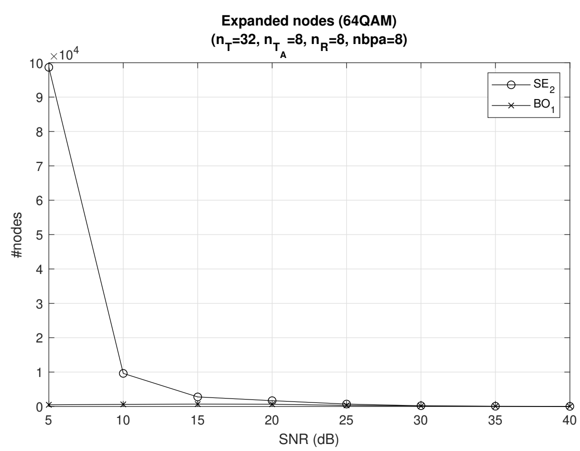

Figure 10: Average expanded nodes in Setup 4 (experiment with 10000 channel matrices).

Table 3: Average expanded nodes in Setup 3.

SNR	SE_1	SE_2	BO_1
5	8.3E+04	2.6E+04	3.4E+02
10	9.2E+03	4.8E+03	2.5E+02
15	3.0E+03	1.4E+03	2.1E+02
20	9.9E+02	6.2E+02	1.7E+02
25	3.5E+02	1.8E+02	8.8E+01
30	7.6E+01	4.4E+01	3.2E+01
35	4.1E+01	3.0E+01	1.5E+01
40	1.6E+01	1.3E+01	8.9E+00

The results indicate that, as long as the noise is moderate ($SNR \geq 10$), the SE_2 method is slightly better. However, in the presence of large noise, it becomes necessary to switch to the box optimization sphere detector (method BO_1) in order to obtain acceptable computing times.

Figure 11 shows the effect of the proposed techniques when the size of the problem increases. We have chosen spectral efficiency (bps/Hz) as a measure of the size of the GSM-ML problem, see Table 1. For ease of interpretation of the graph, we have chosen a single SNR (10dB) as a representative SNR. Then, the average computing times of each method for a SNR of 10dB are displayed, versus bps/Hz.

6. Conclusion

The algorithm proposed in [11] allows ML detection in GSM-MIMO problems of small and moderate size. However, when this algorithm is applied to large MIMO problems, its computational cost becomes excessive, even for research simulations. In this paper, we propose two new techniques that can be used along with the algorithm proposed in [11] for large GSM-MIMO problems.

The first proposal is the use of a bound based on box optimization that can be used to discard configurations. This proposal is very useful for large GSM

Table 4: Average computing times (seconds) in Setup 4 (experiment with only 500 channel matrices)

SNR	SE_1	SE_2	BO_1
5	7.2E+01	1.8E+00	2.6E-01
10	4.9E-00	3.1E-01	2.5E-01
15	7.4E-01	1.9E-01	2.5E-01
20	5.5E-01	1.7E-01	2.3E-01
25	1.1E-01	1.5E-01	1.8E-01
30	7.8E-02	1.4E-01	1.5E-01
35	4.2E-02	1.3E-01	1.4E-01
40	3.5E-02	1.3E-01	1.3E-01

MIMO detection problems with moderate noise. The second proposal is to use a box-optimization-aided sphere decoding solver for the MIMO subproblems. The experimental results show that this technique becomes necessary in order to obtain reasonable computing times in large MIMO-GSM detection problems, if the noise is also large.

7. Acknowledgements

This work has been partially supported by Spanish Ministry of Science, Innovation and Universities and by European Union through grant RTI2018-098085-BC41 (MCUI/AEI/FEDER), by GVA through PROMETEO/2019/109 and by Catedra Telefonica-UPV through SSENCE project.

References

[1] J. Wang, S. Jia, J. Song, Generalised Spatial Modulation System with Multiple Active Transmit Antennas and Low Complexity Detection Scheme, IEEE Trans. Wireless Commun. 11-4 (2012) 1605–1615.

[2] M. Di Renzo, H. Haas, A. Ghrayeb, S. Sugiura, L. Hanzo, Spatial Modulation for Generalized MIMO: Challenges, Opportunities, and Implemen-

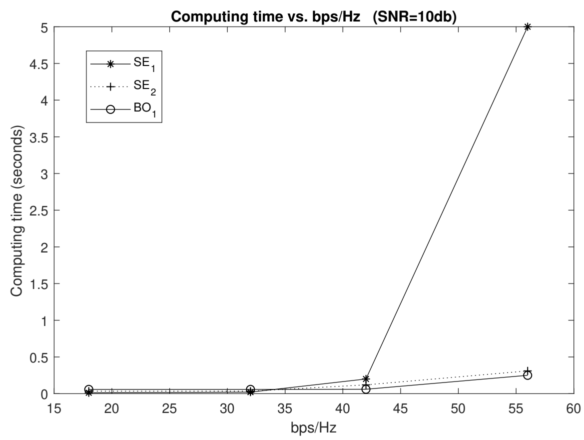

Figure 11: Average computing times vs bps/Hz, SNR=10dB.

tation, Proceedings of the IEEE 102 (2014) 56–103. doi:10.1109/JPROC.2013.2287851.

[3] P. Patcharamaneepakorn, S. Wu, C.-X. Wang, H. Aggoune, M. Alwakeel, X. Ge, M. Di Renzo, Spectral, Energy, and Economic Efficiency of 5g Multicell Massive MIMO systems with Generalized Spatial Modulation, IEEE Transactions on Vehicular Technology 65 (2016) 1–1. doi:10.1109/TVT.2016.2526628.

[4] Y. Xiao, Z. Yang, L. Dan, P. Yang, L. Yin, W. Xiang, Low-Complexity Signal Detection for Generalized Spatial Modulation, IEEE Commun. Lett. 18-3 (2014) 403–406.

[5] G. Altin, M. Çelebi, A Simple Low-Complexity Algorithm for Generalized Spatial Modulation, AEU-Int J. Electron. C. 97 (2018) 63–67.

[6] L. Xiao, P. Yang, Y. Xiao, S. Fan, M. D. Renzo, S. Li, Efficient Compressive Sensing Detectors for Generalized Spatial Modulation Systems, IEEE Trans. Veh. Technol. 66-2 (2017) 1284–1298.

[7] C. Li, J. Wang, W. Liu, Y. Cheng, Y. Huang, Low-Complexity Detection for GSM-MIMO Systems via Spatial Constraint, in 2016 IEEE

83rd Vehicular Technology Conference (VTC Spring) (2016). doi:DOI: 10.1109/VTCSpring.2016.7504330.

[8] B. Zheng, S. Lin, M. Wen, Q. Li, Y. Huang, F. Chen, Fixed-Complexity Sphere Decoding for Soft detection for Generalized Spatial Modulation, in 2017 IEEE Globecom Workshops (GC Wkshps) (2017). doi:10.1109/GLOCOMW.2017.8269179.

[9] B. Zheng, M.Wen, F. Chen, N. Huang, F. Ji, H. Yu, The K-Best Sphere Decoding for Soft Detection of Generalized Spatial Modulation, IEEE Trans. Commun. 65-11 (2017) 4803–4816.

[10] J. Cal-Braz, R. Sampaio-Neto, Low-Complexity Sphere Decoding Detector for Generalized Spatial Modulation Systems, IEEE Commun. Lett. 18-6 (2014) 949–952.

[11] T. Liu, C. Chen, C. Liu, Fast Maximum Likelihood Detection of the Generalized Spatially Modulated Signals Using Successive Sphere Decoding Algorithms, IEEE Commun. Lett. 23-4 (2019) 656–659. doi:doi: 10.1109/LCOMM.2019.2898398.

[12] M. Stojnic, H. Vikalo, B. Hassibi, Speeding up the Sphere Decoder with H^∞ and SDP Inspired Lower Bounds, IEEE Transactions on Signal Processing 56 (2) (2008) 712–726.

[13] X. Wen, Q. Han, Solving Box–Constrained Integer Least Squares Problems, IEEE Trans. on Wireless Communications 7 (1) (2008) 277–287.

[14] V. M. Garcia-Molla, A. Vidal, A. Gonzalez, S. Roger, Improved Maximum Likelihood Detection through Sphere Decoding combined with Box Optimization, Signal Processing, Elsevier 98 (2014) 287–294.

[15] MATLAB, (R2018b), The MathWorks Inc., Natick, Massachusetts, 2018.

[16] E. Agrell, T. Eriksson, A. Vardy, K. Zeger, Closest point search in Lattices, IEEE Trans. Commun. 48 (2002) 2201–2214.

[17] C. Schnorr, M. Euchner, Lattice Basis Reduction: Improved Practical Algorithms and Solving Subset Sum Problems, Math. Programming 48 (66) (1994) 181–191.

[18] B. Hassibi, H. Vikalo, On Sphere Decoding algorithm. I. Expected Complexity, IEEE Trans. Signal Process. 53 (2005) 2806–2818.

[19] U. Fincke, M. Pohst, Improved Methods for Calculating Vectors of Short Length in a Lattice, Including a Complexity Analysis, Math. Comput. 44 (170) (1985) 463–471.

[20] G. Golub, C. Van Loan, Matrix Computations, 3rd Edition, John Hopkins University Press, Baltimore, Maryland, 1996.

[21] V. M. Garcia-Molla, MIMO Hard ML detectors and Soft-output Detectors Accelerated through Box Optimization.
URL `http://www.inco2.upv.es/Software.php`