An MMP-Based Approach for Detection in Large MIMO Systems Using Sphere Decoding

Abhishek, Abhay Kumar Sah, and A. K. Chaturvedi, *Senior Member, IEEE*

***Abstract*—Sphere decoders (SDs) are known to provide near maximum-likelihood performance in multiple-input multiple-output systems. But their ability to provide good error performance has not evinced any interest in large antenna systems because of their very high complexity. However, if the number of erroneous dimensions can be made low, say by some pre-processing, we can use SD for those dimensions. This could potentially improve the error performance at a reasonable complexity. To identify such dimensions, we present a multipath matching pursuit-based approach. Simulation results show that compared to existing algorithms, the proposed approach can provide a significant improvement in error performance.**

***Index Terms*—Large MIMO, massive MIMO, sphere decoder, mutlipath matching pursuit.**

I. Introduction

In RECENT times there has been an increased interest in large multiple-input multiple-output (MIMO) (often called massive MIMO) systems [1], [2]. However, the advantages promised by large/massive MIMO systems can be realized only if several hurdles can be overcome. One of them is the design of a reliable and computationally efficient detector. Sphere decoder (SD) and its variants [3], [4] are known to provide close to maximum likelihood (ML) performance but their computational complexity increases rapidly with the number of transmit antennas and therefore, in their present form they are not suited for large antenna systems [1].

In this letter, we attempt to exploit the potential of a sphere decoder to provide superior error performance by applying it only on the dimensions which cannot be detected correctly by a low complexity detector like zero forcing (ZF) or minimum mean square error (MMSE). Since the number of such erroneous dimensions are likely to be low, the complexity of SD will not be high. Approaches to identifying such dimensions have been discussed in [5] and [6] by using matching pursuit (MP) algorithms [7]–[9]. The multipath MP (MMP) algorithm is known to be the best MP algorithm in the literature. However, application of MMP algorithm to the detection problem requires us to address the uncertainty in the number of erroneous dimensions. We begin by using a residual vector to identify the locations of the errors in the low complexity initial solution. Next, we use an improved residual update strategy to improve the localization accuracy, iteratively, and use a thresholding function to determine the likely erroneous locations. Finally, to reduce the variation in the number of errors in a block, we combine a concatenation based approach [10] with the above ideas to propose the SD-MMP algorithm.

The rest of the letter is organized as follows. Section II describes the preliminaries. We propose the algorithm in Section III and its complexity analysis has been discussed in Section IV. Simulation results are presented in Section V. Finally, we conclude the letter in Section VI. $\mathbf{A}^{\dagger}$ stands for the Moore-Penrose pseudoinverse of matrix $\mathbf{A}$. The set operators $A\backslash B$ and $|A|$ denote, respectively, the elements in set A but not in set B and the cardinality of set A.

II. Preliminaries

Consider a MIMO system with n_t transmit antennas and n_r receive antennas ($n_r \geq n_t$). Mathematically,

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

where $\mathbf{y}$ is an $n_r \times 1$ receive vector, $\mathbf{x}$ is an $n_t \times 1$ transmit vector, $\mathbf{H}$ is an $n_r \times n_t$ channel matrix and $\mathbf{n}$ is an $n_r \times 1$ noise vector. The elements of the channel matrix are distributed as $\mathcal{CN}(0, 1)$ and those of the noise vector as $\mathcal{CN}(0, \sigma^2)$. The elements of $\mathbf{x}$ are drawn from a finite alphabet set $\mathcal{A}$.

The aim of the detection problem is to find the vector which minimizes the Euclidean cost. Since there are $|\mathcal{A}|^{n_t}$ possible transmit vectors, it is infeasible to find such a vector even for small MIMO systems. Therefore, an SD restricts the search within a sphere of radius d, as

$$\hat{\mathbf{x}}_{SD} = \underset{\mathbf{x}\in\mathcal{A}^{n_t}}{\operatorname{argmin}} \{\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 \leq d^2\}, \tag{2}$$

and for a suitable choice of d it can achieve near-ML performance at a polynomial complexity.

In the proposed approach, we assume that the erroneous locations in the initial solution $\hat{\mathbf{x}}$ obtained using a linear detector (LD) are contained in set $\mathcal{S}$. Using this information, the effect of correctly detected symbols can be subtracted from the original received vector. For initial solutions satisfying $|\mathcal{S}| < n_t$, the dimension of the MIMO system will be reduced, thereby facilitating application of sphere decoding for achieving better performance.

One way to determine the locations of the incorrectly detected symbols in $\hat{\mathbf{x}}$ could be to use a MP algorithm [7]–[9], which works if the following condition [11] is satisfied

$$n_r > c\,|\mathcal{S}| \log\left(\frac{n_t}{|\mathcal{S}|}\right), \tag{3}$$

Manuscript received November 21, 2016; revised December 25, 2016; accepted December 25, 2016. Date of publication December 29, 2016; date of current version April 7, 2017. The associate editor coordinating the review of this paper and approving it for publication was J. Wu. *(All the authors contributed equally to this work.)*

The authors are with the Department of Electrical Engineering, Indian Institute of Technology Kanpur, Kanpur 208016, India (e-mail: abhie@iitk.ac.in; abhaysah@iitk.ac.in; akc@iitk.ac.in).

Digital Object Identifier 10.1109/LWC.2016.2646368

where c is a constant. In the next section, we propose an algorithm to improve error localization leading to the application of sphere decoding for low complexity detection.

III. SD-MMP Algorithm

We employ the MMP algorithm to determine the erroneous locations in the initial solution. In contrast to other matching pursuit algorithms, MMP maintains multiple promising candidates throughout the search process. The MMP algorithm limits the number of promising candidates to a predefined maximum value l_{max} and investigates them via a serial search which stops when all the l_{max} candidates have been examined. The proposed SD-MMP is given in Algorithm 1. The key steps of the algorithm follow.

A. Initialization

A concatenation based approach, where a block of K initial solution vectors are processed together, has been proposed in [10] to improve the error localization capability of gOMP [9]. Taking a cue from there, we initialize the algorithm with K MMSE solution vectors $\{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_K\}$ corresponding to K consecutive instances of receive vectors $\{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_K\}$, where $\hat{\mathbf{x}}_i$ represents the MMSE solution for $\mathbf{y}_i$ given by

$$\hat{\mathbf{x}}_i = (\mathbf{H}_i^H \mathbf{H}_i + \sigma^2 \mathbf{I})^{-1} \mathbf{H}_i^H \mathbf{y}_i. \tag{4}$$

We denote the error corresponding to $\hat{\mathbf{x}}_i$ as $\mathbf{x}_{e,i} = \hat{\mathbf{x}}_i - \mathbf{x}_i$. Next, we determine the support sets $\mathbf{x}_{e,i}\ \forall\, i = 1$ to K.

B. Localization of Errors

The support sets corresponding to the K sub-blocks for the l^{th} candidate vector at the qth iteration are given by $\mathcal{S}_{l,i}^q \forall\, i = 1$ to K, with initial support sets $\mathcal{S}_{l,i}^0$ being the null set '$\emptyset$'. The residual vectors $\mathbf{r}_i^q = \mathbf{y}_i - \mathbf{H}_i \hat{\mathbf{x}}_i\ \forall\, i = 1$ to K are used to compute the corresponding metric $\mathbf{f}_i = \mathbf{H}_i^H \mathbf{r}_i^q$. We determine the search order of the candidate vectors using the relation [8]

$$l = 1 + \sum_{k=1}^{N_{itr}} L^{k-1}(c_k - 1), \tag{5}$$

where L is the number of child paths for each candidate and $(c_1, c_2, \ldots, c_{N_{itr}})$ denotes the order in which these paths will be searched. We denote by λ and ϕ, the sub-block to which the c_q^{th} largest element of the vector $|\mathbf{f}| = [|\mathbf{f}_1|^T\, |\mathbf{f}_2|^T \ldots |\mathbf{f}_K|^T]$[1] belongs and the index of this element relative to the λ^{th} sub-block respectively. The support set $\mathcal{S}_{l,\lambda}$ is then augmented and is used for estimating the corresponding error vector, which is then used for updating the respective residual vector for the $(q+1)$th iteration as

$$\mathbf{r}_\lambda^{q+1} = \mathbf{r}_\lambda^0 - \mathbf{H}_\lambda \left(\mathcal{S}_{l,\lambda}^{q+1}\right) \mathbf{x}_{e,\lambda}. \tag{6}$$

It may be noted that this strategy is similar to [10] but unlike the MMP in [8] where the receive vector $\mathbf{y}$ is used instead of the current residual vector. This is useful in improving the localization accuracy of the algorithm. Since in each iteration the support set corresponding to only one of the K sub-blocks gets updated, the residual vector and support sets for the remaining sub-blocks are the same as those in the previous iteration. The next iteration uses the updated residual vector to compute $|\mathbf{f}_\lambda|$ and the process repeats. At the end of N_{itr} iterations, we obtain the error vectors and the support sets for the sub-blocks contained in $\Lambda_l^{N_{itr}}$. The entire update process is captured within the **for** loop in Algorithm 1.

[1]The modulus operator on a vector $\mathbf{a} = [a_1, a_2, \ldots, a_n]$ is defined as $|\mathbf{a}| = [|a_1|, |a_2|, \ldots, |a_n|]$.

Algorithm 1: SD-MMP Algorithm

Input : $\{\mathbf{y}_1, \cdots, \mathbf{y}_K\}, \{\hat{\mathbf{x}}_1, \cdots, \hat{\mathbf{x}}_K\}, \{\mathbf{H}_1, \cdots, \mathbf{H}_K\}$ η, L, N_{itr}

Output: $\{\check{\mathbf{x}}_1, \cdots, \check{\mathbf{x}}_K\}$

Initialization: $\check{\mathbf{x}}_i = \mathbf{x}_i^0 = \hat{\mathbf{x}}_i\ \forall\, i = 1, 2, \cdots, K$;
i^{th} residual $\mathbf{r}_i^{0,0} = \mathbf{y}_i - \mathbf{H}_i \hat{\mathbf{x}}_i\ \forall\, i$; $\mathbf{r}_i^{-1,0} = \infty\ \forall\, i$;
$p = 0$;
while $\sum_{i=1}^K \|\boldsymbol{r}_i^{p,0}\|^2 < \sum_{i=1}^K \|\boldsymbol{r}_i^{p-1,0}\|^2$ **do**
  $p = p + 1,\ l = 0,\ \|\mathbf{r}^{p,0}\| = \infty$;
  while $l < l_{max}$ **do**
    $l = l + 1,\ \Lambda_l^0 \leftarrow \emptyset\ \mathcal{S}_{l,i} \leftarrow \emptyset\, \forall i = 1, 2, \cdots, K$;
    $[c_1, \cdots, c_{N_{itr}}] =$ **compute-ck** (l, L);
    for $q \leftarrow 1$ **to** N_{itr} **do**
      $\mathbf{f}_i = \mathbf{H}_i^H \mathbf{r}_i^{p-1,q-1} \quad \forall\, i$;
      $\omega_{c_q} \leftarrow$ select index of c_q^{th} largest element in $[|\mathbf{f}_1|^T\, |\mathbf{f}_2|^T \cdots |\mathbf{f}_K|^T]$;
      $\lambda = \lceil \omega_{c_q} / n_t \rceil$;
      $\Lambda_l^q = \Lambda_l^{q-1} \cup \{\lambda\}$;
      $\phi = (\,(\omega_{c_q} - 1)$ **mod** $n_t\,) + 1$;
      $\mathcal{S}_{l,\lambda}^q \leftarrow \mathcal{S}_{l,\lambda}^q \cup \{\phi\}$;
      $\mathbf{x}_{e,\lambda}^q = \mathbf{H}_\lambda^\dagger(\mathcal{S}_{l,\lambda}^q)\, \mathbf{r}_\lambda^{p-1,0}$;
      $\mathbf{r}_\lambda^{p-1,q} = \mathbf{r}_\lambda^{p-1,0} - \mathbf{H}_\lambda(\mathcal{S}_{l,\lambda}^q) \mathbf{x}_{e,\lambda}^q$;
    end
    $\mathcal{S}_{l,i}^p = \{\mathcal{S}_{l,i}^{N_{itr}}(j) \mid |\mathbf{x}_{e,i}(j)| > \eta, j = 1, 2, \cdots, N_{itr}\}$ $\forall\, i \in \Lambda_l^{N_{itr}}$;
    $\mathcal{S}_{l,i}^p \leftarrow$ select m most significant elements;
    $\mathbf{x}_{l,i}^{p-1} = \mathbf{x}_i^{p-1}$;
    $\mathbf{y}_{r,i} = \mathbf{y}_i - \mathbf{H}_i((\mathcal{S}_{l,i}^p)^c) \mathbf{x}_{l,i}^{p-1}((\mathcal{S}_{l,i}^p)^c)\ \forall\, i \in \Lambda_l^{N_{itr}}$;
    Apply SD algorithm on $\mathbf{y}_{r,i}$ using $\mathbf{H}_i((\mathcal{S}_{l,i}^p)^c)$ to obtain $\mathbf{x}_{l,i}^{p-1}((\mathcal{S}_{l,i}^p)^c)\ \forall\, i \in \Lambda_l^{N_{itr}}$;
    if $\sum_{i=1}^K \|\boldsymbol{y}_i - \boldsymbol{H}_i \boldsymbol{x}_{l,i}^{p-1}\|^2 < \sum_{i=1}^K \|\boldsymbol{r}_i^{p,0}\|^2$ **then**
      $\mathbf{x}_i^p = \mathbf{x}_{l,i}^{p-1}\ \forall\, i$;
      $\mathbf{r}_i^{p,0} = \mathbf{y}_i - \mathbf{H}_i \mathbf{x}_i^p\ \forall\, i$;
    end
  end
end
return $\check{\mathbf{x}}_i \leftarrow \mathbf{x}_i^{p-1}\ \forall\, i$.

C. Shrinkage of Support Sets

Using an entire support set for decoding is unnecessary, thus the support sets $\mathcal{S}_{l,i}^{N_{itr}}\ \forall\, i \in \Lambda_l^{N_{itr}}$ are acted upon by a shrinkage operation through a hard thresholding. Only those elements of $\mathcal{S}_{l,i}^{N_{itr}}$ whose $|\mathbf{x}_{e,i}|$ exceed a pre-defined parameter η are retained. Since the elements of $\hat{\mathbf{x}}_i$ and $\mathbf{x}_i$ belong to the

alphabet set $\mathcal{A}$, the non-zero elements of $|\mathbf{x}_{e,i}|$ will take values from a difference set $\mathcal{D} = \{|\alpha_i - \alpha_j| \,|\forall \alpha_i \neq \alpha_j \,\&\, \alpha_i, \alpha_j \in \mathcal{A}\}$. Therefore, $|\mathbf{x}_{e,i}|$ greater than half of $\min \mathcal{D}$ can be used as an indicator for the symbols which are likely to be in error. Hence, we propose to select the value of η to be $d_{min}/2$, where d_{min} is the minimum distance between any two points in the constellation from which the alphabet $\mathcal{A}$ is chosen.

We further limit the cardinality of $\mathcal{S}_{l,i}$ to m ($m \ll n_t$) by retaining only those elements in $\mathcal{S}_{l,i}$ which correspond to the m largest entries in the vector $|\mathbf{x}_{e,i}|$. The value of m is chosen such that the complexity of implementing SD is kept in check.

D. Decoding of Erroneous Symbols

The effect of the correctly detected symbols, i.e., the elements of $\mathcal{S}^c_{l,i} = [1 : n_t] \backslash \mathcal{S}_{l,i}$ are subtracted and we obtain

$$\mathbf{y}_{r,i} = \mathbf{y}_i - \mathbf{H}_i(\mathcal{S}^c_{l,i})\, \hat{\mathbf{x}}_i(\mathcal{S}^c_{l,i}) = \mathbf{H}_i(\mathcal{S}_{l,i})\, \mathbf{x}_i(\mathcal{S}_{l,i}) + \mathbf{n}_i. \quad (7)$$

This reduced dimension MIMO system can now be detected by an SD. After performing error recovery for all the possible candidate vectors, the one with the least norm is compared with the residual vector of the previous iteration. If its norm is less, the previous solution is updated and the steps in Sections III-B and III-C are performed over the new residual vector. This process continues until there is a reduction in the norm of the residual vector. This part has been captured in the outer while loop in Algorithm 1. In Algorithm 1, $(\cdot)^\dagger$, $\lceil \cdot \rceil$, mod, and compute-ck denote the Moore-Penrose pseudo inverse, the ceiling function, the standard modulo operation and the operation in (5), respectively.

E. Choosing the Values of L and N_{itr}

The number of iterations in the **for** loop of Algorithm 1 is denoted by N_{itr}. Since at each iteration, only one of the support sets $\mathcal{S}^q_{l,i}\, \forall\, i = 1$ to K is augmented by one, the sum of cardinalities of the support sets after N_{itr} iterations can be at most N_{itr}. Thus, the value of N_{itr} should ideally be variable and depend on the number of errors in the solution vector before the **for** loop begins. For convenience, we have taken N_{itr} to be constant and assumed it to be the smallest integer greater than the average number of errors in the MMSE solution vectors. Usually, the average symbol error rate of an initial solution obtained using MMSE detector is less than 0.1. Hence, we choose N_{itr} to be $\lceil 0.1\, n_t K \rceil$, where $n_t K$ is the length of the solution vectors. The value of L is based on the relation [6] $LN_{itr} = \lfloor K n_r / \log(K n_t) \rfloor$, where $\lfloor \cdot \rfloor$ denotes the floor function. This relation ensures that sufficient number of child paths have been explored for each candidate.

IV. Complexity Analysis

In this section, we determine the complexity of the key operations which define the MMP-SD algorithm.

1) Computation of $\mathbf{f}$*:* In the first iteration of the **for** loop inside Algorithm 1, $\mathbf{f}_i$ needs to be computed for K sub-blocks thus requiring $K(2n_r - 1)n_t$ flops, followed by the modulus operation which requires $4Kn_t$ flops. Finding the index corresponding to the c_q^{th} largest element of the vector $[|\mathbf{f}_1|^T\, |\mathbf{f}_2|^T \ldots |\mathbf{f}_K|^T]$ has $O(Kn_t)$ complexity. Since only one of the $\mathbf{f}_i$s needs to be updated in each of the subsequent iterations, a total of $(N_{itr} - 1)n_r n_t$ flops are required. Thus, the overall complexity of this operation is $O(N_{itr} n_r n_t)$ where we have taken $K < N_{itr}$.

2) Computation of $\mathbf{x}_{e,i}$*:* For the computation of the least squares solution $\mathbf{x}_{e,i}$, we need the QR decomposition of $\mathbf{H}_i$. For this, we employ the modified Gram-Schmidt (MGS) algorithm which makes use of the QR decomposition at the previous iteration. If the i^{th} sub-block is not contained in Λ_l^{q-1}, the complexity of this operation is $O(n_r)$. Otherwise, the complexity is $O(n_r N_i)$ where N_i is the number of times the i^{th} sub-block occurs in Λ_l^q. The worst-case complexity results when Λ_l^q contains the index of only one sub-block in each iteration. This results in an overall complexity of $O(n_r N_{itr}^2)$.

3) Residual Update: This step requires $(2N_i - 1)n_r$ flops for multiplication followed by n_r flops for subtraction, resulting in a complexity of $2N_i n_r$. The worst-case complexity occurs when $\Lambda_l^{N_{itr}}$ contains only one sub-block, leading to $n_r N_{itr}(N_{itr} + 1)$ flops. This is followed by the calculation of the norm of the residual which is compared with the previous residual, requiring $(4n_r - 1)|\mathcal{S}_l^{N_{itr}}|$ flops. Subsequent update requires a total of $2n_r|\mathcal{S}_l^{N_{itr}}|$ flops, resulting in an overall complexity of $O(n_r N_{itr}^2)$. It may be noted that the complexity of shrinkage operation is negligible, hence we ignore it here.

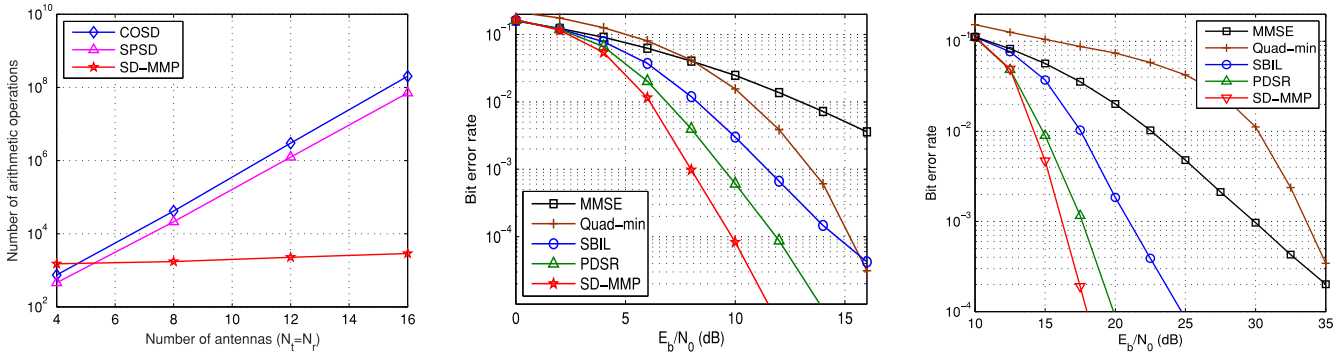
4) Complexity of SD: The expected complexity of SD is known to be polynomial (often cubic) in the number of antennas [12]. Since the dimensionality of the reduced system is at max m ($\ll n_t$), the complexity of SD will be $O(m^3)$.

Considering the facts that $m \ll n_t$ and N_{itr} is of the order of n_t, the total complexity obtained by adding complexities of all the above key operations can be seen to be $O(N_{itr} n_r n_t)$ and the overall complexity for l_{max} candidates will be $O(N_{itr} l_{max} n_r n_t)$. Since we are processing K sub-blocks together, the complexity per sub-block or per receive vector is $O(N_{itr} l_{max} n_r n_t / K)$. If we take $N_{itr} = 0.1\, n_t K$ and $l_{max} < n_t$, the complexity of SD-MMP can be expressed in terms of n_r and n_t as $O(n_t^2 n_r)$. This analysis assumes the search orders of the candidate vectors to be disjoint, which will be rarely true. Thus, the actual complexity will often be lower. It may be noted that the above expressions of complexity do not consider the complexity of initialization, which is again $O(n_t^2 n_r)$ for MMSE based initialization and hence does not change the order of overall complexity.

V. Simulation Results

The simulation set up is the same as described after (1). In Fig. 1(a) we compare the overall complexity of SD-MMP with two recent variants of SD - statistical pruning SD (SPSD) [3] and channel ordering based SD (COSD) [4]. The results shown are for 16-QAM modulation with $E_b/N_0 = 5$ dB. For obtaining the performance of SD-MMP, we have used Schnorr-Euchner SD [3] and the simulation parameters were taken to be $K = 10$, $m = 4$, and $l_{max} = 20$. From the figure, it can be seen that the number of arithmetic operations of both SPSD and COSD increase rapidly with the number of antennas while for SD-MMP, the increase is so low that it appears independent of the number of antennas.

Since obtaining the error performance of COSD or SPSD is not feasible for large systems, we compare the error



(a) (b) (c)

Fig. 1. (a) Number of arithmetic operations and bit error rates for (b) a 32 × 32 4-QAM system and (c) a 128 × 128 16-QAM system.

TABLE I
COMPARISON BASED ON ARITHMETIC OPERATIONS ($\times 10^3$) PER BIT

Detection Algorithm	4-QAM Modulation at $E_b/N_0 = 10$ dB			
	32×32		64×64	
	Additions	Multiplications	Additions	Multiplications
SBIL	0.930	0.948	2.823	2.853
PDSR	6.862	8.568	18.354	27.931
SD-MMP	12.18	12.85	15.61	16.81
	16-QAM Modulation at $E_b/N_0 = 15$ dB			
	64×64		128×128	
SBIL	2.936	2.966	13.494	13.557
PDSR	23.61	32.24	65.71	79.77
SD-MMP	21.73	26.12	49.385	59.811

performance of SD-MMP with existing algorithms for large MIMO systems, specifically with the algorithms which use a similar sparsity based framework like SBIL [6], PDSR [5] and Quad-min [13]. Thus, we consider a 32×32 4-QAM system and a 128×128 16-QAM system. The bit error rate (BER) performances for the respective systems are shown in Figs. 1(b) and 1(c), and the overall number of arithmetic operations per bit are compared in Table I for four different cases. From both the figures, it can be seen that SD-MMP outperforms the others and there is a clear order in which the four algorithms can be placed with respect to error performance. In fact, except for one case, SD-MMP is better than the second best PDSR in complexity as well.

For example, for a 128×128 16-QAM system, SD-MMP requires 2 dB less E_b/N_0 compared to PDSR to achieve a BER of 10^{-4} and there is approximately 24% savings in the total number of arithmetic operations. Compared to SBIL, the gain at a BER of 10^{-4} is approximately 7 dB while compared to Quad-min the gain is at least 15 dB. It may be noted that for a 32×32 4-QAM system, Quad-min is inferior to MMSE up to an $E_b/N_0 = 8$ dB while for a larger system and a bigger constellation such as a 128×128 16-QAM system, Quad-min is inferior to MMSE up to a considerably higher value of E_b/N_0 (see Fig. 1(c)). This deterioration in the performance of Quad-min is similar to several other algorithms [2] like lattice reduction aided linear detectors and semi-definite relaxation based detectors, the performances of which are known to deteriorate once the size of the MIMO system becomes very large, if the SNR is not very high. Given the significantly inferior error performance of Quad-min, we have not included its complexity in Table I.

VI. CONCLUSION

We present an MMP based approach for using sphere decoding over the dimensions which are erroneously detected by a linear detector. This approach helps in better identification of such dimensions, paving the way for the application of sphere decoding to the dimensions where it is needed most. Thus, the proposed SD-MMP algorithm is able to exploit the power of an SD to provide improved error performance while keeping the overall complexity low. Using simulations, we show that SD-MMP offers a better alternative for large MIMO systems compared to existing sparsity based schemes.

REFERENCES

[1] F. Rusek *et al.*, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, Jan. 2013.
[2] S. Yang and L. Hanzo, "Fifty years of MIMO detection: The road to large-scale MIMOs," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1941–1988, 4th Quart., 2015.
[3] T. Cui, S. Han, and C. Tellambura, "Probability-distribution-based node pruning for sphere decoding," *IEEE Trans. Veh. Technol.*, vol. 62, no. 4, pp. 1586–1596, May 2013.
[4] B. S. Thian and A. Goldsmith, "A reduced-complexity MIMO receiver via channel ordering," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, Honolulu, HI, USA, 2009, pp. 1–6.
[5] J. W. Choi and B. Shim, "New approach for massive MIMO detection using sparse error recovery," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, Austin, TX, USA, Dec. 2014, pp. 3754–3759.
[6] X. Peng, W. Wu, J. Sun, and Y. Liu, "Sparsity-boosted detection for large MIMO systems," *IEEE Commun. Lett.*, vol. 19, no. 2, pp. 191–194, Feb. 2015.
[7] J. Wang, S. Kwon, P. Li, and B. Shim, "Recovery of sparse signals via generalized orthogonal matching pursuit: A new analysis," *IEEE Trans. Signal Process.*, vol. 64, no. 4, pp. 1076–1089, Feb. 2016.
[8] S. Kwon, J. Wang, and B. Shim, "Multipath matching pursuit," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2986–3001, May 2014.
[9] J. Wang, S. Kwon, and B. Shim, "Generalized orthogonal matching pursuit," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6202–6216, Dec. 2012.
[10] A. Abhishek, A. K. Sah, and A. K. Chaturvedi, "Improved sparsity behaviour and error localization in detectors for large MIMO systems," presented at IEEE GLOBECOM Workshops, Washington, DC, USA, 2016.
[11] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.
[12] B. Hassibi and H. Vikalo, "On the sphere-decoding algorithm I. Expected complexity," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2806–2818, Aug. 2005.
[13] Y. Fadlallah, A. Aïssa-El-Bey, K. Amis, D. Pastor, and R. Pyndiah, "New iterative detector of MIMO transmission using sparse decomposition," *IEEE Trans. Veh. Technol.*, vol. 64, no. 8, pp. 3458–3464, Aug. 2015.