Learning-based MIMO Detection with Dynamic Spatial Modulation

Le He, Lisheng Fan, Xianfu Lei, Xiaohu Tang, Pingzhi Fan, *Fellow, IEEE*, and Arumugam Nallanathan, *Fellow, IEEE*

***Abstract*—In this paper, we investigate signal detection in emerging dynamic spatial modulation (DSM) based MIMO systems, where existing mapping methods do not work efficiently. Therefore, we propose a combinatorial mapping-based DSM (CM-DSM) scheme in this work. The proposed CM-DSM scheme employs a combinatorial 3D mapping to avoid the detection ambiguity and achieve a lower average number of active antennas in the system. Additionally, this mapping helps construct an appropriate decision tree for optimal signal detection. By leveraging the combinatorial nature of CM-DSM, we propose a memory-bounded tree search (METS) algorithm, which efficiently finds the maximum likelihood (ML) estimate. To further enhance detection efficiency, we propose a deep learning boosted version of METS (DL-METS) that learns the optimal heuristic function from implicit data patterns. Simulation results indicate that both the proposed METS and DL-METS work well in the system. In particular, the proposed DL-METS achieves nearly optimal detection performance while maintaining almost the lowest expected computational complexity. This strongly validates the effectiveness of the algorithm proposed in this work.**

***Index Terms*—Spatial modulation, variable active antennas, MIMO detection, deep learning, tree search.**

I. Introduction

MASSIVE multiple-input multiple-out (MIMO) communication systems have been considered as one of the fundamental components for next-generation wireless communication networks [1]. The usage of large antenna arrays at both transmitter and receiver can significantly improve transmission rate and spectral efficiency of wireless communication. However, conventional MIMO has the disadvantage that the system complexity and deployment cost increase rapidly with the number of antennas [1]. Taking this issue into consideration, the concept of spatial modulation (SM) [2]–[5], which carries information by using both the spatial dimension and conventional 2D symbols, has recently attracted much attention. In MIMO communication systems, SM only activates a part of the transmit antennas for conveying information. At the same time, extra bits can be transmitted via the active antenna combination (AAC). Hence, we can still achieve a high transmission data rate while employing fewer radio frequency (RF) chains. In other words, SM achieves a comprehensive balance between energy efficiency and spectral efficiency compared to traditional MIMO systems and has been considered a promising technique for achieving ultra-reliable and low-latency communication (URLLC) in future wireless networks [3].

During the past decade, many variants of the SM family have emerged. The original version of SM was first introduced in [6], and it selects a dedicated antenna at each time slot. Despite the fact that the transmitted information can be identified by the AAC together with the transmitted symbol, the overall performance improvement is limited, especially when the number of antennas is large. Generalized spatial modulation (GSM) was thereby studied in [7]–[9], which activates a fixed number of transmit antennas at each instant to further improve the spectral efficiency. In this case, GSM has the potential to support a higher data transmission rate than the original version. Since the number of active antennas is fixed in GSM-MIMO systems, the information bits can be separately mapped into AAC and symbols [7], and the number of transmitted bits is proportional to the number of active antennas. However, if the number of active antennas is variable, then the lack of such information at the receiver side will result in detection ambiguity, which will significantly degrade the system bit error rate (BER) performance [10]. To tackle this issue and fully exploit the available RF chains at the transmitter, dynamic spatial modulation (DSM) with varying active transmit antennas has been investigated in [10]–[12]. In contrast to GSM, information bits in DSM based MIMO systems are jointly mapped into the complete 3D constellation consisting of both the AAC and conventional 2D symbols, such that we can alleviate the detection ambiguity at the receiver [10]. Besides, this kind of mapping is also known as 3D-mapping in the literature, and the choice is unlimited. For example, the authors in [10] proposed JM-VSM, which fixes the number of transmitted bits with pseudo-Gray mapping. By using variable active antennas, DSM is able to provide a much higher data rate compared to GSM, and meanwhile achieves a higher energy efficiency compared to the conventional systems.

Unfortunately, due to the increasing number of active antennas, signal detection in SM family-based MIMO systems becomes much more challenging compared to that in the conventional MIMO systems [13]–[15]. In addition, the inter-channel interference (ICI), signal sparsity and varying antenna combinations also make the detection of transmitted signal much more difficult [16], [17]. Specifically, a brute force

L. He, and L. Fan are both with Guangzhou University, Guangzhou 510006, China (e-mail: hele20141841@163.com, lsfan@gzhu.edu.cn).

X. Lei, X. Tang and P. Fan are all with the School of Information Science and Technology, Institute of Mobile Communications, Southwest Jiaotong University, Chengdu 610031, China (e-mail: xflei@home.swjtu.edu.cn, xhutang@swjtu.edu.cn, p.fan@ieee.org).

A. Nallanathan is with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London, U.K. (e-mail: a.nallanathan@qmul.ac.uk)

enumeration to find the maximum likelihood (ML) estimate in SM family-based MIMO systems will typically have to search over the whole set of all possible AACs and symbols, which can rapidly become computationally infeasible for massive MIMO. To tackle this issue, some low-complexity and sub-optimal detection algorithms have been proposed for the SM and GSM variants, including the ordered-block minimum mean squared error (OB-MMSE) algorithm [13], Gaussian approximation [14], Bayesian cooperative detection [18] and message passing-based detection algorithms [15], [19], [20]. The aforementioned algorithms all have lower computational complexities than the brute force enumeration, but the BER performances can not approach the optimal one.

To reduce the computational complexity while pursuing the optimal BER performance, researchers have turned to tree search-based signal detection algorithms [16], [17], [21]–[24]. Tree search algorithms convert the detection problem into a search for the closest path on a decision tree, allowing for a significant reduction in average computational complexity, although the complexity of all tree search algorithms is still exponential in the worst case [21]. For example, some approaches employ an additional symbol “0” or NULL to identify inactive antennas in GSM and use conventional sphere decoding (SD) or K-best search algorithms to find the optimal solution [21]–[24]. However, these approaches still have to check each node, even if it is not a valid candidate, which limits their complexity reduction. Sorting assisted successive sphere decoding (SA-SSD) algorithm was developed in [16] to address this issue by searching only on decision trees with valid AACs. SA-SSD sorts all possible AACs and finds the closest path accordingly, which is efficient for small numbers of overall antenna combinations but not for larger scales. Furthermore, a similar strategy that builds on SA-SSD was proposed in [17], reducing computational complexity further by using box-optimization and initial radius broadcasting.

Besides, in recent years, researchers have explored learning-based methods to improve the efficiency of MIMO detection, inspired by the unprecedented progress made in deep learning. For instance, in [25], a model-driven MIMO detection method was proposed that used deep learning. The approach modified a conventional iterative detection algorithm by unfolding it into a deep neural network (DNN) structure. Two auxiliary parameters were introduced at each layer of the DNN for better canceling multi-user interference (MUI). The learning to learn iterative search algorithm (LISA) proposed in [26] employed different DNN-based architectures as the parameterized policy function based on the channel model. Through learning the optimal parameters of this function, LISA provides improved BER performance compared to its predecessors, and it is feasible even with imperfect channel state information (CSI). Additionally, meta-learning-based search algorithms have recently been proposed in [27] and [28]. The EPNet in [27] employs a deep unfolded DNN structure with the expectation propagation algorithm for iterative detection and decoding. A meta-learning-based mechanism was developed for online training to provide significant robustness in practical deployment. Furthermore, an intelligent MIMO detection network was proposed in [28], which used DNN and meta learning to significantly reduce the complexity of the K-best search algorithm.

While many algorithms have been proposed to address the signal detection problem in MIMO systems based on SM or GSM variants, there is still a lack of studies on efficient detection algorithms for DSM in the literature. This is because existing detection methods were mainly designed for spatial modulation systems with a fixed number of active antennas and separate mapping. In this case, most existing detection algorithms fail to work well due to the variable number of active antennas and joint 3D mapping in DSM. For example, existing message passing-based detection algorithms typically assume knowledge of the fixed sparsity constraint [15], [19], [20], which is impractical due to the variable sparsity in DSM. To the best of our knowledge, two most recent attempts for designing an effective detection algorithm for DSM with variable number of active antennas were introduced in [29] and [30]. In [29], the authors modified the OB-MMSE algorithm and proposed OWMMSE-CML to support variable active antenna and joint 3D mapping. However, this algorithm still tries to enumerate all possible antenna combinations, which will bring degraded performance in large-scale systems. In [30], the authors proposed a partitioning and sorting based sphere decoding (PS-SD) detector for the signal detection with variable active antennas and the JM-VSM 3D-mapping scheme [10]. Firstly, PS-SD partitions and sorts the transmission vectors according to the number of active antennas as well as the probability of antenna being activated. After that, PS-SD employs sphere decoding to estimate the sorted transmission vectors. However, the complexity of PS-SD still grows rapidly with the increasing number of antennas. Therefore, designing a low-complexity optimal signal detection algorithm with variable number of active antennas still remains a very challenging issue, which is also the motivation of this work.

In this paper, we are interested in designing an efficient search algorithm for the fast optimal signal detection in DSM-MIMO. To this end, we first propose a novel combinatorial mapping based DSM (CM-DSM) MIMO system, which enables us to utilize the combinatorial nature of DSM to construct a combinatorial tree for the signal detection. Based on CM-DSM, we then develop a memory-bounded tree search (METS) algorithm which can utilize the proposed CM-DSM structure to find the optimal ML estimate with reduced complexity. In further, we propose a deep learning boosted version of METS (DL-METS), which provides almost the optimal detection performance and meanwhile can achieve nearly the lowest complexity. To summarize our work, the main contributions of this paper are,

- We propose a novel 3D-mapping approach, namely CM-DSM, to address the detection ambiguity problem by leveraging the combinatorial nature of the spatially modulated signal. In contrast to the existing 3D-mapping schemes which rely on pre-stored mapping tables, the proposed CM-DSM is a real-time scheme which maps binary bits according to their computed combinatorial order. In this way, not only hardware and operation costs are reduced, but also the detection complexity can be reduced by exploiting the resultant combinatorial decision tree.

- We propose a low-complexity optimal detection algorithm, namely METS, in order to address the optimal detection problem in spatial modulations with variable number of active antennas. The proposed METS utilizes the combinatorial decision tree of the CM-DSM scheme to prune many unpromising nodes, resulting in that it can achieve the exact optimal detection performance with considerably reduced complexity.
- We further propose an improved version of METS, namely DL-METS, by combining deep learning and the proposed CM-DSM scheme to guide the METS towards the most promising path during the search process. We show that by using DNN to predict the least remaining cost of each branches, we can further avoid visiting many unpromising intermediate nodes. By comparing to the existing detection approaches, this strategy enables further significant complexity reduction, which does not compromise on the detection performance.

The rest of this paper is arranged as follows. Section II details the proposed CM-DSM-based MIMO system and explains how the proposed CM-DSM scheme can further decrease the average number of active antennas in the system. Section III proposes the METS algorithm for effective detection in the above system, while Section IV presents the DL-METS algorithm, a deep learning boosted version of the METS algorithm that enhances its detection efficiency. Finally, Section V discusses related simulation results, and Section VI presents the conclusion of this paper.

II. Proposed CM-DSM based MIMO System

A. System Model for the CM-DSM Scheme

In this section, we will first describe the proposed CM-DSM based MIMO system. Then, we will detail the proposed METS algorithm, and show how it can efficiently utilize the CM-DSM structure to find the optimal ML estimate.

As illustrated in Fig. 1, we consider a MIMO system in which N_t and N_r antennas are deployed at the transmitter and receiver, respectively. In addition, only N_{RF} ($N_r \geqslant N_t \geqslant N_{RF}$) RF chains are available at the transmitter. Let N_a denote the number of active antennas at the transmitter, and then obviously we have $N_a \leqslant N_{RF}$. Unlike GSM, the number of active antennas in DSM is varied, and the total number of bits that can be transmitted per instant is given by

$$K = \left\lfloor \log_2 \left\{ \sum_{N_a=1}^{N_{RF}} \binom{N_t}{N_a} |\mathcal{X}|^{N_a} \right\} \right\rfloor, \qquad (1)$$

where $\binom{N_t}{N_a} = \frac{N_t!}{N_a!(N_t-N_a)}$ denotes the binomial coefficient, $\mathcal{X}$ is the constellation set for quadrature amplitude modulation (QAM) or phase shift keying (PSK) modulation, and $\lfloor \cdot \rfloor$ denotes the floor operation. As pointed out in [10], a major issue of variable activate antennas is that it will cause detection ambiguity at the receiver if we map information bits separately into AACs and symbols. With separate mapping methods, a variable number of bits is transmitted, and the receiver will not know how many bits are transmitted exactly. This detection ambiguity will result in a substantial degradation of the detection performance of the system. To solve this issue, one possible approach is to employ 3D-mapping methods which jointly map information bits over the complete set of all possible combinations of AACs and symbols. However, the existing mapping methods like the JM-VSM introduced in [10] will increase the expected number of active antennas, since the priority of AACs is not exploited.

Algorithm 1 Combinatorial Encoding and Decoding

```
procedure ENCODING($\boldsymbol{c}$)
    Calculate the combinatorial index $k$ based on (2)
    return $k$
end procedure

procedure DECODING($k$)
    Set $\boldsymbol{c} = [c_0, c_1, \ldots, c_n, \ldots, c_{N_t-1}]$ as zeros
    $r = k$
    for $k = N_a$ to 1 do
        for $n = N_e + k - 1$ to $k - 1$ do
            if $r \geqslant \binom{n}{k}$ then
                $c_n = 1$
                $r = r - \binom{n}{k}$
                break
            end if
        end for
    end for
    return $\boldsymbol{c}$
end procedure
```

Consequently, we hereby propose a novel combinatorial mapping method, which takes the priority of AACs into account, and jointly maps binary bits according to their combinatorial order. The main idea of the proposed CM-DSM is to rank all the 2^K possible combinations of AACs and symbols in the order of fewer number of active antennas first. This enables us to reduce the average number of active antennas. It is important to note that not all combinations are valid for mapping, since only 2^K combinations can be used to convey information. Basically, all combinations will firstly be ranked lexicographically according to their AACs, and then they are ordered in terms of the transmitted symbols. In this case, the number of transmitted bits is fixed to K, and we can thereby avoid the detection ambiguity at the receiver.

In the proposed CM-DSM scheme, we first lexicographically order all the AACs that have an identical number of active antennas. The ranking can be done based on the combinatorial ranking method introduced in [31], [32]. Specifically, given an AAC of $\boldsymbol{c} = [c_0, c_1, \cdots, c_{N_t-1}]$, the matching index can be computed as

$$k = \sum_{n=0}^{N_t-1} c_n \binom{n}{k_n}, \qquad (2)$$

where $k_n = \sum_{i=0}^{n} c_i$, and $\binom{n}{k_n} = 0$ holds for any $n < k$. Based on (2), we are able to construct a coding trellis for converting AACs to indices and vice versa. The coding and decoding procedures of this coding trellis are shown in Algorithm 1.

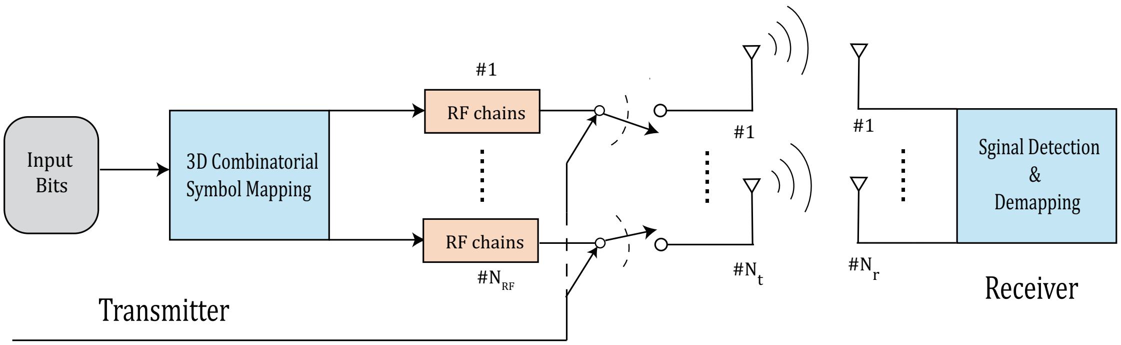

Fig. 1. Diagram of the proposed CM-DSM based MIMO system.

Algorithm 2 Mapping Algorithm for CM-DSM based MIMO Systems

```
Input: Information bits $\boldsymbol{b} = [b_K, b_{K-1}, \cdots, b_1]$ of length $K$;
Output: Transmitted signal $\boldsymbol{s}$;
Convert the binary sequence $\boldsymbol{b}$ to decimal number $n = \sum_{i=0}^{K} b_i 2^{i-1}$;
for $N_a = 1, 2, \cdots, N_{RF}$ do
    if $n < \sum_{i=1}^{N_a} \binom{N_t}{i} |\mathcal{X}|^i$ then
        $r = n - \sum_{i=1}^{N_a - 1} \binom{N_t}{i} |\mathcal{X}|^i$;
        $k = \left\lfloor \frac{r}{|\mathcal{X}|^{N_a}} \right\rfloor$;
        $q = r - k|\mathcal{X}|^{N_a}$;
        Interpret decimal number $k$ as AAC according to the decoding procedure in Algorithm 1;
        Interpret decimal number $q$ as binary sequence $\hat{\boldsymbol{b}} = [\hat{b}_{N_a}, \hat{b}_{N_a-1}, \cdots, \hat{b}_1]$ of length $N_a$;
        Convert the binary sequence $\hat{\boldsymbol{b}}$ as symbols according to the selected conventional constellation;
        Construct transmitted signal $\boldsymbol{s}$ according to the selected AAC and symbols;
        return $\boldsymbol{s}$;
    end if
end for
```

Algorithm 3 Demapping Algorithm for CM-DSM based MIMO Systems

```
Input: Transmitted signal $\boldsymbol{s}$;
Output: Information bits $\boldsymbol{b} = [b_K, b_{K-1}, \cdots, b_1]$ of length $K$ and the corresponding index number $n$;
Get the AAC $\boldsymbol{c}$ and number of active antennas $N_a$ from the signal vector $\boldsymbol{s}$;
Encode the AAC $\boldsymbol{c}$ as an index number $k$ according to (2);
Convert the $N_a$ non-zero symbols of $\boldsymbol{s}$ to a binary sequence $\hat{\boldsymbol{b}}$ of length $N_a \log_2(|\mathcal{X}|)$;
Compute the corresponding decimal number $q$ from the binary sequence $\hat{\boldsymbol{b}}$;
$r = k|\mathcal{X}|^{N_a} + q$;
$n = r + \sum_{i=1}^{N_a - 1} \binom{N_t}{i} |\mathcal{X}|^i$;
Convert the decimal number $n$ to a binary sequence $\boldsymbol{b}$ of length $K$;
return $\boldsymbol{b}$ and $n$;
```

By exploiting this coding trellis, the combinatorial 3D-mapping scheme can be constructed by jointly selecting the AAC and symbols according to the information bits, and then recovering the bits from the transmitted symbols. Specifically, we provide the details of the proposed mapping and demapping algorithms in Algorithms 2 and 3, respectively. As shown in Algorithm 2, while mapping bits into signal vectors, we first determine the number of active antennas by checking whether the total number of possible combinations activating at most N_a antennas is larger than the decimal number represented by the information bits. After that, we determine which kind of combination should be used according to the index number k of the AAC. Then, the symbols can be determined according to the residual number q and conventional BPSK or QAM constellations. While demapping signal vectors into bits, we just apply the reverse procedure of the mapping process, as shown in Algorithm 3. In addition, we also provide examples for the proposed combinatorial mapping scheme in Appendix A, in order to help better understand the proposed mapping and demapping procedures.

Intuitively, the proposed combinatorial mapping can reduce the average number of active transmit antennas, since the AACs with more active antennas can be excluded as many as possible. Specifically, all possible antenna combinations can be encoded as binary sequences. These binary sequences can then be ranked lexicographically, which means that all AACs are also ranked in a lexicographical order of fewer active antennas first. Moreover, we order all possible transmit signal vectors in terms of both the corresponding AAC and symbols, which eventually results in a combinatorial mapping between the input bits and signal candidates. In this manner, it is clear that the signal candidates with more active antennas will be considered in a low priority. Since not all AACs will be used for mapping, those AACs with more active antennas will be excluded from this mapping as many as possible. Therefore, the mapping in the proposed scheme can actually achieve a lower average number of active antennas if we consider that there is no prior information on the incoming data. Mathematically, the average number of active antennas of the proposed CM-DSM based MIMO system can be given by

$$\overline{N_a} = \frac{1}{2^K} \sum_{i=1}^{N_{RF}} N(i), \tag{3}$$

where

$$N(i) = \min\left\{ \binom{N_t}{i} |\mathcal{X}|^i, \max\left\{ 0, 2^K - \sum_{j=1}^{i-1} \binom{N_t}{j} |\mathcal{X}|^j \right\} \right\} \tag{4}$$

represents the number of valid candidates with i active antennas. In contrast, the other mapping methods fail to incorporate such priority, e.g., JM-VSM [10], and take into account the AACs with more active antennas in the mapping table, which thereby results in a higher average number of active antennas.

III. Memory-Bounded Tree Search for CM-DSM based MIMO Systems

After mapping the information bits into signal vectors, the formed signal vector $\boldsymbol{s} \in \mathcal{S}^{N_t}$ will be transmitted via the wireless channel, and thus we have the received signal as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{s} + \boldsymbol{w}, \tag{5}$$

where $\boldsymbol{H} \in \mathbb{C}^{N_r \times N_t}$ represents the full channel state information (CSI) matrix, $\boldsymbol{w} \in \mathbb{C}^{N_r \times 1}$ stands for the additive white Gaussian noise (AWGN) having a zero mean and unit variance. It should be noted that the effective constellation set of the CM-DSM based MIMO system is $\mathcal{S} = \mathcal{X} \cup \{0\}$, where $\mathcal{X}$ is the alphaset of conventional 2D modulation and "0" is used for identifying inactive antennas (e.g. $\mathcal{X} = \{-1, +1\}$ and $\mathcal{S} = \{-1, 0, +1\}$ if BPSK modulation is adopted). Moreover, the system model described in (5) can be represented alternatively as the effective subsystems given by

$$\boldsymbol{y} = \sum_{j=1}^{N_a} \boldsymbol{h}_j s_j + \boldsymbol{w} = \bar{\boldsymbol{H}}\boldsymbol{s} + \boldsymbol{w}, \tag{6}$$

where $\boldsymbol{h}_j$ represents the column of $\boldsymbol{H}$ corresponding to the j-th non-zero element s_j of $\boldsymbol{s}$, and $\bar{\boldsymbol{H}} = [\boldsymbol{h}_1, \cdots, \boldsymbol{h}_j, \cdots, \boldsymbol{h}_{N_a}]$ is the effective sub-CSI matrix.

Based on (5), the mathematically optimal approach to estimate signals in CM-DSM based MIMO systems is provided by

$$\boldsymbol{s}^* = \arg\min_{\boldsymbol{s} \in \mathcal{S}^{N_t}} \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{s}\|^2, \tag{7}$$

which exhaustively searches the closest Euclidean point inside the search space. Note that the optimal estimate in (7) involves a computational complexity exponentially increasing with the problem scale. To reduce the complexity, we will formulate the problem as a search on the decision tree, such that we can find the optimal estimate efficiently according to a specific search strategy. Specifically, we first employ the QR factorization on the CSI matrix, which is given by

$$\boldsymbol{H} = \underbrace{\begin{bmatrix} \boldsymbol{Q}_1 & \boldsymbol{Q}_2 \end{bmatrix}}_{\boldsymbol{Q}} \begin{bmatrix} \boldsymbol{R} \\ \boldsymbol{0} \end{bmatrix} = \boldsymbol{Q} \begin{bmatrix} \boldsymbol{R} \\ \boldsymbol{0} \end{bmatrix}, \tag{8}$$

where $\boldsymbol{Q}_1 \in \mathbb{C}^{N_r \times N_t}$ and $\boldsymbol{Q}_2 \in \mathbb{C}^{N_r \times (N_r - N_t)}$ consist of orthogonal columns, and $\boldsymbol{R} \in \mathbb{C}^{N_t \times N_t}$ denotes an upper triangular matrix. By applying the orthogonal transformation into (7), we then have

$$\boldsymbol{s}^* = \arg\min_{\boldsymbol{s} \in \mathcal{S}^{N_t}} \left\| \boldsymbol{Q}^H \boldsymbol{y} - \begin{bmatrix} \boldsymbol{R} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{s} \right\|^2, \tag{9}$$

$$= \arg\min_{\boldsymbol{s} \in \mathcal{S}^{N_t}} \sum_{j=1}^{N_t} \left\| z_k - \sum_{j=0}^{k} r_{k,j} s_j \right\|^2, \tag{10}$$

$$= \arg\min_{\boldsymbol{s} \in \mathcal{S}^{N_t}} \sum_{j=1}^{N_t} b(\boldsymbol{s}^k), \tag{11}$$

where $\boldsymbol{z} \triangleq \boldsymbol{Q}_1^H \boldsymbol{y}$, $r_{i,j}$ represents the (i,j)-th component of $\boldsymbol{R}$ by counting inversely, and the j-th increment $b(\boldsymbol{s}^k) = \left\| z_k - \sum_{j=0}^{k} r_{k,j} s_j \right\|^2$ can be determined by the partially determined vector $\boldsymbol{s}^k = [s_k, s_{k-1}, \cdots, s_1]$ only. Moreover, we denote the cumulative cost as $g(\boldsymbol{s}^k) = \sum_{i=1}^{k} b(\boldsymbol{s}^i)$. Obviously, we have the successive identity $g(\boldsymbol{s}^{k+1}) = g(\boldsymbol{s}^k) + b(\boldsymbol{s}^{k+1})$, and a decision tree is constructed. Thus, we can find the optimal estimate by searching the corresponding decision tree for the shortest path.

An example of the decision tree is presented in Fig. 2 for illustration, where $N_{RF} = 2$, $N_t = 3$, and BPSK modulation is adopted. In this decision tree, a node located on the k-th level of the tree is denoted as $\boldsymbol{s}^k$, which also denotes the path starting from the root to that node. In particular, we refer to the nodes as goal nodes when they are located on the deepest level, i.e., when $k = N_t$, since they represent complete solutions. Obviously, the cumulative cost of goal nodes also represents the Euclidean distance of the associated signal vector. We shall emphasize that not all the goal nodes are associated with valid DSM signals, and only the goal nodes with valid DSM signals are valid candidates. We can now easily check whether an intermediate node is leading to valid goal nodes or not through exploiting the structure of the proposed CM-DSM, and checking whether the index number of the intermediate node calculated from Algorithm 3 is smaller than 2^K or not.

Since invalid intermediate nodes can be quickly identified by using the CM-DSM structure, we can now find the optimal estimate of the transmitted signal, i.e., the shortest valid path of the corresponding decision tree. To this end, we propose an memory-bounded tree search (METS) algorithm, which can utilize the combinatorial nature of CM-DSM to find the optimal solution. Basically, our proposed METS is developed based on the tree search algorithm introduced in [33], where the original algorithm counts the tree level from the leaf nodes while our proposed METS counts the tree level from the root. With different sizes of available memory space, METS will search the optimal solution with different policies. Specifically, it will search the tree in a best-first manner if the memory space is large enough. Otherwise, it will work in a depth-first manner. Since best-first runs much faster than depth-first, the algorithm can search the optimal path faster with enough memory space. Mathematically, the minimum memory size requirement of the algorithm can be expressed as [33]

$$L = (N_t - 1)(|\mathcal{S}| - 1) + 1. \tag{12}$$

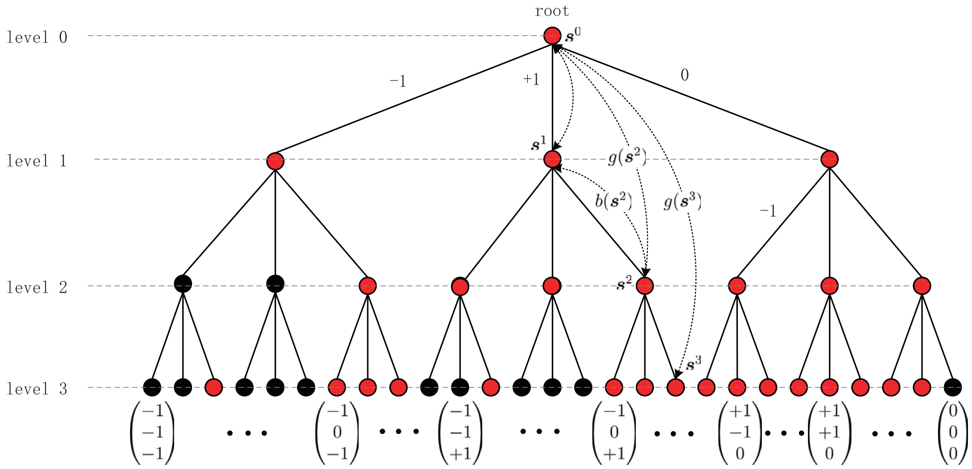
root
level 0
level 1
level 2
level 3
s^0
−1
+1
0
s^1
$g(s^2)$
$b(s^2)$
$g(s^3)$
−1
s^2
s^3
$\begin{pmatrix}-1\\-1\\-1\end{pmatrix}$ ··· $\begin{pmatrix}-1\\0\\-1\end{pmatrix}$ ··· $\begin{pmatrix}-1\\-1\\+1\end{pmatrix}$ ··· $\begin{pmatrix}-1\\0\\+1\end{pmatrix}$ ··· $\begin{pmatrix}+1\\-1\\0\end{pmatrix}$ ··· $\begin{pmatrix}+1\\+1\\0\end{pmatrix}$ ··· $\begin{pmatrix}0\\0\\0\end{pmatrix}$

Fig. 2. Structure of the CM-DSM based MIMO decision tree with $N_t = 3$, $N_{RF} = 2$, and BPSK modulation. A path is valid if and only if it contains only red nodes.

Accordingly, given the currently available memory size of $\hat{L}$, the minimum visitable tree level is given by [33]

$$k_{min}(\hat{L}) = \max\left\{N_t - 1 - \left\lfloor \frac{\hat{L}}{|\mathcal{S}| - 1} \right\rfloor, 0\right\}. \quad (13)$$

By jointly using the above two equations as well as the demapping algorithm presented in Algorithm 3, our proposed METS is established, and the pseudo codes are presented in Algorithm 4. Specifically, METS will always start searching from a dummy root $\boldsymbol{s}^0$. At each iteration, it always visits the best node among the visitable nodes in memory according to (13). When visiting the node $\boldsymbol{s}^k$, METS will find all the children which not only fall within the predefined sphere, but also have an index number calculated from Algorithm 3 smaller than 2^K. It should be noted that if the intermediate node has only zero elements, then its index is calculated by setting its first element to the least element of the alphaset $\mathcal{X}$. For example, if $N_t = 4$ and $\mathcal{X} = \{-1, +1\}$ (BPSK), then the index number for an intermediate node $\boldsymbol{s}^2 = [0, 0]$ is calculated from $\boldsymbol{s}^4 = [-1, 0, 0, 0]$. In this case, many unpromising nodes can be pruned to make the search much faster. In particular, METS will update the radius and prune the tree whenever a valid goal node is generated. Under this strategy, the size of search space will be reduced multiple times during the search, and the algorithm will find the optimal solution if a goal node is eventually visited.

IV. Proposed Deep Learning Boosted Memory-Bounded Tree Search

In this section, we will first describe the proposed deep learning based memory efficient search algorithm for signal detection in DSM-MIMO. Then, we will discuss its computational complexity.

A. Estimating the Optimal Heuristic with Deep Learning

Generally, the best-first search based algorithm will employ a heuristic function to guide the search process towards the

Algorithm 4 METS Algorithm for CM-DSM based MIMO Systems

```
Input: Full CSI matrix $\boldsymbol{H}$ and received signal $\boldsymbol{y}$;
Output: Estimated signal $\hat{\boldsymbol{s}}$, and the number of visited nodes $N_v$;
Initialize the search radius $R = \infty$;
Initialize the currently available memory size as $\hat{L} = L$;
Push the dummy root $\boldsymbol{s}^0$ into memory, and set $\hat{L} = \hat{L} - 1$;
Set the currently visiting tree level $k = 0$, and set the node to be visited as $\boldsymbol{s}^k = \boldsymbol{s}^0$;
Set the number of currently visited nodes as $N_v = 0$;
while $k \neq N_t$ do
    Start visiting $\boldsymbol{s}^k$, and set $N_v = N_v + 1$;
    Find all the $M$ ($M \leqslant |\mathcal{S}|$) children $\boldsymbol{s}^{k+1}$ of $\boldsymbol{s}^k$ which not only fall within the sphere $g(\boldsymbol{s}^{k+1}) \leqslant R$, but also have an index number calculated from Algorithm 3 smaller than $2^K$;
    Remove $\boldsymbol{s}^k$ from memory, and then append the $M$ children into the memory;
    Set $\hat{L} = \hat{L} - M + 1$;
    if any child $\boldsymbol{s}^{k+1}$ is a goal node with $k = N_t - 1$, and $g(\boldsymbol{s}^{k+1}) < R$ then
        Update the radius as $R = g(\boldsymbol{s}^{k+1})$;
        Prune the tree with the updated radius, and update the memory accordingly;
    end if
    Suppose that the nodes inside memory are grouped according to their levels, and the nodes of each group are sorted in an ascending order of their cost $g(\boldsymbol{s}^k)$. Then, we find the best node among the first nodes of each group, with a restriction to only consider the nodes with a tree level not less than $k_{\min}(\hat{L})$. After finding the best node, we denote it as $\boldsymbol{s}^l$.
    Set the next node to be visited as $\boldsymbol{s}^k = \boldsymbol{s}^l$;
end while
return $\hat{\boldsymbol{s}} = \boldsymbol{s}^k$, and $N_v$;
```

most promising path. Mathematically, the remaining cost from the node $\boldsymbol{s}^k$ to another node $\boldsymbol{s}^j$ is defined as

$$h(\boldsymbol{s}^j, \boldsymbol{s}^k) = g(\boldsymbol{s}^j) - g(\boldsymbol{s}^k), \quad \forall \boldsymbol{s}^j \supseteq \boldsymbol{s}^k, \tag{14}$$

where $\boldsymbol{s}^j \supseteq \boldsymbol{s}^k$ denotes that $\boldsymbol{s}^j$ is the descendent of $\boldsymbol{s}^k$. Then, the optimal heuristic $h^*(\boldsymbol{s}^k)$ estimates the least remaining cost from $\boldsymbol{s}^k$ to all goal nodes of the sub-tree of that node, given by

$$h^*(\boldsymbol{s}^k) = \min_{\boldsymbol{x}^{N_t} \supseteq \boldsymbol{s}^k} h(\boldsymbol{s}^{N_t}, \boldsymbol{s}^k). \tag{15}$$

Moreover, we will use the optimal heuristic to estimate the cost of the shortest complete path via that node, which is given by

$$f^*(\boldsymbol{s}^k) = g(\boldsymbol{s}^k) + h^*(\boldsymbol{s}^k). \tag{16}$$

Now it is clear that if we are able to estimate the optimal f-cost precisely, then we can always guide the search towards the optimal path. Unfortunately, it is very difficult to estimate the optimal heuristic since it requires to check all the goal nodes of the sub-tree. Therefore, we hereby propose to use deep learning to find the best heuristic function. Once the optimal heuristic can be estimated precisely by learning the implicit patterns from data, we can significantly boost the search process to improve the detection efficiency.

Specifically, we denote the DNN based heuristic function as $h(\boldsymbol{s}^k|\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is the set of learnable parameters of the DNN. Then, we can compute the f-cost of a node as $f(\boldsymbol{s}^k|\boldsymbol{\theta}) = g(\boldsymbol{s}^k) + h(\boldsymbol{s}^k|\boldsymbol{\theta})$. In order to find the best heuristic, a fully connected neural network (FCNN) with I layers is considered in our proposed algorithm. For the i-th layer, we denote the number of neurons as n_i, and the activation function of each layer is the rectified linear unit (ReLU) based activation function. Thus, the output of the i-th layer can be expressed by

$$\boldsymbol{p}^i = \max\{\boldsymbol{W}^i \boldsymbol{p}^{i-1} + \boldsymbol{b}^i, \boldsymbol{0}\}, \tag{17}$$

where $\max(\cdot,\cdot)$ executes element-wisely, $\boldsymbol{p}^{i-1}$ represents the input of layer i, and $\boldsymbol{W}^i \in \mathbb{R}^{n_i \times n_{i-1}}$ and $\boldsymbol{b}^i \in \mathbb{R}^{n_i \times 1}$ are the learnable parameters, respectively. In particular, we denote the input of the first layer as $\boldsymbol{p}^0 = [\mathrm{R}(\tilde{\boldsymbol{w}}), \mathrm{I}(\tilde{\boldsymbol{w}})]^T$, where $\mathrm{R}(\cdot)$ and $\mathrm{I}(\cdot)$ denote the real and imaginary parts, respectively, and the complex vector $\tilde{\boldsymbol{w}}$ is given by

$$\tilde{\boldsymbol{w}} = \boldsymbol{z} - \boldsymbol{R} \begin{bmatrix} \boldsymbol{0}_{(N_t-k)\times 1} \\ \boldsymbol{s}^k \end{bmatrix}, \tag{18}$$

where the rest $(N_t - k)$ elements of $\boldsymbol{s}^k$ are set to zero.

By leveraging deep learning to find the optimal heuristic function, METS algorithm can be significantly improved. The pseudo codes of the resulting deep learning boosted METS (DL-METS) algorithm are presented in Algorithm 5. Generally, the DL-METS algorithm runs just like the METS algorithm, except the ability to estimate the initial radius and guide the search towards the shortest path. Specifically, DL-METS will estimate the initial radius as $R = f(\boldsymbol{s}^0|\boldsymbol{\theta})$, by estimating the heuristic of the dummy root node with the well-trained DNN. Moreover, when visiting nodes, DL-METS will check the estimated f-cost rather than g-cost, which will make the algorithm prune many unnecessary nodes. In particular, DL-METS employs an increase factor λ ($\lambda > 1$) to increase the radius as $R = \lambda R$ if the search is failed. This is because that the estimated heuristic may cause the algorithm fail to append any child into the memory such that the algorithm can not find the next node to be visited. When the number of failures grows, the radius will eventually be large enough to ensure a success search, such that the algorithm will eventually find an appropriate estimate of the transmitted signal. In conclusion, the major differences between DL-METS and METS are two-folds:

- DL-METS can estimate a good initial radius to reduce the size of search space before searching. As a contrast, METS initializes the radius to infinity at the beginning, which will significantly increases the size of search space.
- Instead of using the cumulative g-cost in METS, DL-METS can guide the search process towards the shortest path, by leveraging a deep network to estimate the shortest remaining cost at each node. This enables DL-METS to find the solution much faster.

B. Training Procedure

Now it is obvious that our training objective is to train the DNN for estimating the optimal heuristic as precise as possible. Then, a straightforward approach is to minimize the estimation error between the estimated heuristic and the optimal heuristic. Moreover, for every node on the best path of each sub-tree, its f-cost should be equal to the g-cost of the full path. It is easy to prove this fact by the definition of the optimal heuristic. Following this, the training objective is to minimize the mean squared estimation error,

$$\mathcal{L}_{\mathcal{D}}(\boldsymbol{\theta}) = \mathbb{E}_{\{\boldsymbol{s}^k, \boldsymbol{s}^{N_t}\} \in \mathcal{D}} \left\{ \| g(\boldsymbol{x}^{N_t}) - f(\boldsymbol{s}^k|\boldsymbol{\theta}) \|^2 \right\}, \tag{19}$$

in which the empirical expectation is taken on the dataset,

$$\mathcal{D} = \{\boldsymbol{z}_t, \boldsymbol{R}_t, \boldsymbol{s}_t^1, \boldsymbol{s}_t^2, \cdots, \boldsymbol{s}_t^k, \cdots, \boldsymbol{s}_t^{N_t}\}_{t=1}^T, \tag{20}$$

where the time index t denotes that the corresponding data item is randomly drawn from different decision trees of different time slots, and the total number of time slots is T. It should be noted that the nodes $\boldsymbol{s}^k$ inside the dataset $\mathcal{D}$ are only taken from the optimal complete path. Based on the loss function and data set, we can employ the mini-batch gradient descent optimizer to train the network as

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{D}}(\boldsymbol{\theta}). \tag{21}$$

However, finding the ML estimate is still needed during the training. For simplicity, we will take the nodes from the path leading to the transmitted signal rather than the shortest path. This is because that the transmitted signal is most likely to be the shortest path, such that finding the ML estimate is no longer needed during the training. Based on this strategy, the DNN can be trained well very fast, while the proposed algorithm's performance is still good. The complete training procedure of DL-METS is presented in Algorithm 6.

It should be emphasized that the network structure in this paper is not limited to the FCNN, and it can be other forms by jointly exploiting the number of antennas and the available

Algorithm 5 DL-METS Algorithm for CM-DSM based MIMO Systems

```
Input: Full CSI matrix $\boldsymbol{H}$, received signal $\boldsymbol{y}$ and the increase factor $\lambda > 1.0$;
Output: Estimated signal $\hat{s}$, and the number of visited nodes $N_v$;
Initialize the search radius $R = f(\boldsymbol{s}^0|\boldsymbol{\theta})$;
Initialize the currently available memory size as $\hat{L} = L$;
Push the dummy root $\boldsymbol{s}^0$ into memory, and set $\hat{L} = \hat{L} - 1$;
Set the currently visiting tree level $k = 0$, and set the node to be visited as $\boldsymbol{s}^k = \boldsymbol{s}^0$;
Set the number of currently visited nodes as $N_v = 0$;
while $k \neq N_t$ do
    Start visiting $\boldsymbol{s}^k$, and set $N_v = N_v + 1$;
    Find all the $M$ ($M \leqslant |\mathcal{S}|$) children $\boldsymbol{s}^{k+1}$ of $\boldsymbol{s}^k$ which not only fall within the sphere $f(\boldsymbol{s}^{k+1}|\boldsymbol{\theta}) \leqslant R$, but also have an index number calculated from Algorithm 3 smaller than $2^K$;
    Remove $\boldsymbol{s}^k$ from memory, and then append the $M$ children into the memory;
    Set $\hat{L} = \hat{L} - M + 1$;
    if any child $\boldsymbol{s}^{k+1}$ is a goal node with $k = N_t - 1$, and $f(\boldsymbol{s}^{k+1}|\boldsymbol{\theta}) < R$ then
        Update the radius as $R = f(\boldsymbol{s}^{k+1}|\boldsymbol{\theta})$;
        Prune the tree with the updated radius, and update the memory accordingly;
    end if
    Suppose that the nodes inside memory are grouped according to their levels, and the nodes of each group are sorted in an ascending order of their cost $g(\boldsymbol{s}^k)$. Then, we find the best node among the first nodes of each group, with a restriction to only consider the nodes with a tree level not less than $k_{\min}(\hat{L})$. After finding the best node, we denote it as $\boldsymbol{s}^l$.
    if $\boldsymbol{s}^l$ is found then
        Set the next node to be visited as $\boldsymbol{s}^k = \boldsymbol{s}^l$;
    else
        Initialize the search radius $R = \lambda R$;
        Initialize the currently available memory size as $\hat{L} = L$;
        Push the dummy root $\boldsymbol{s}^0$ into memory, and set $\hat{L} = \hat{L} - 1$;
        Set the node to be visited as $\boldsymbol{s}^k = \boldsymbol{s}^0$;
    end if
end while
return $\hat{s} = \boldsymbol{s}^k$ and $N_v$;
```

Algorithm 6 Training Procedure for DL-METS

```
// Randomly generate symbols for the training data set;
Set the total number of time slots as $T$;
Initialize the data set $\mathcal{D}_{all} = \varnothing$;
for $t = 1, 2, \cdots, T$ do
    Randomly generate $\boldsymbol{z}_t$, $\boldsymbol{R}_t$ and $\boldsymbol{s}_t^{N_t}$ with random signal-to-noise ratio (SNR);
    $\mathcal{D}_{all} = \mathcal{D}_{all} \cup \{\boldsymbol{z}_t, \boldsymbol{R}_t, \boldsymbol{s}_t^{N_t}\}$;
    // Generate samples by enumerating the nodes on the transmitted signal;
    for $k = 0, 1, \cdots, N_t - 1$ do
        Select the node $\boldsymbol{s}_t^k$ on the $k$-th level of the path $\boldsymbol{s}_t$;
        $\mathcal{D}_{all} = \mathcal{D}_{all} \cup \{\boldsymbol{s}_t^k\}$;
    end for
end for
// Training the model with the generated samples;
Initialize the network parameters $\boldsymbol{\theta}$ to random values;
loop
    Randomly pick a subset of samples $\mathcal{D}$ from the whole data set $\mathcal{D}_{all}$;
    According to (19), we then calculate the average loss on the data set $\mathcal{D}$;
    Update the network's parameters $\boldsymbol{\theta}$ with stochastic gradient decent;
end loop
```

resources. In this paper, FCNN is adequate as the heuristic network implementation, since it is simple and hardware-friendly. In general, FCNN is suitable for low-dimensional tasks, since the MIMO signal detection problem typically involves much smaller dimensions compared to other tasks like image recognition. Besides, this paper is mainly concerned with verifying the effectiveness of the deep heuristic network, and the structure of the network considered in this paper is merely intended as a guide. In practice, it is important to employ convolutional neural networks (CNNs) to reduce the computational complexity for very large-scale systems. Moreover, recursive neural networks (RNNs) can be helpful for temporally correlated channels. Therefore, selecting an appropriate heuristic network for a given system should be based on its particular requirements and resource limitations.

C. Computational Complexity

Essentially, there are two factors that play a dominant role together in measuring the computational complexity of DL-METS algorithm: the number of visited nodes and the cost of visiting nodes. Mathematically, the cost of visiting node $\boldsymbol{s}^k$ is given by

$$\mathcal{O}\left(k + \sum_{i=1}^{I} n_i n_{i-1} + n_i\right), \tag{22}$$

which is the corresponding computational complexity when computing the f-cost. This includes the costs of computing the increment $b(\boldsymbol{s}^k)$ as well as the heuristic $h(\boldsymbol{s}^k|\boldsymbol{\theta})$. In particular, when the optimal heuristic is estimated precisely by the heuristic network, i.e., $h(\boldsymbol{s}^k|\boldsymbol{\theta}) = h^*(\boldsymbol{s}^k)$, the search algorithm will visit the least number of nodes. In this case, only the N_t nodes that lie on the optimal path will eventually be visited. Thus, we have the lower bound on the computational complexity of the proposed DL-METS as

$$\mathcal{O}\left(N_t^2 + N_t\left(\sum_{i=1}^{I} n_i n_{i-1} + n_i\right)\right). \tag{23}$$

On the other hand, if imperfect estimate of the optimal heuristic occurs, there will be no guarantee on the optimality. In other words, it may result in exponential complexity growing with the problem scale when meeting the worst case. However, the computational complexity of the proposed DL-METS can still be significantly improved if we provide high quality estimates of the optimal heuristic. As we will show in the following simulation results, the BER performance and search speed can be improved significantly with low estimation errors. In particular, the proposed DL-METS can achieve almost the optimal BER performance, while it still visits nearly the least number of nodes in low SNR regimes. More specifically, when the number of antennas is greater than 24, we achieve a four-fold complexity reduction compared with PS-SD. From the superiority of the proposed method, we can obtain a useful insight on the system design that the proposed deep model-driven learning can serve as a promising detection approach for future low-complexity and high-performance communication systems.

V. Simulations Results and Discussions

In this section, simulation results will be presented to validate the effectiveness of the proposed METS and DL-METS algorithms.

A. Environment Setup

In general, the simulations are performed under several different CM-DSM based MIMO systems, where we employ QPSK as the modulation scheme. To show the robustness of the proposed algorithm, we consider the practical scenarios with spatially correlated Rayleigh channels, where the Kronecker model [34] is employed. Formally, the channel model is given by

$$\boldsymbol{H} = \boldsymbol{R}_r^{\frac{1}{2}} \tilde{\boldsymbol{H}} \boldsymbol{R}_t^{\frac{1}{2}}, \tag{24}$$

where $\tilde{\boldsymbol{H}}$ represents an independent and identically distributed (i.i.d.) Gaussian matrix, $\boldsymbol{R}_r$ represents the correlation matrix at the receiver, while $\boldsymbol{R}_t$ represents the correlation matrix at the transmitter. In our simulations, the channel correlation only occurs at the transmitter. Thus, $\boldsymbol{R}_r$ and $\boldsymbol{R}_t$ are given by

$$\boldsymbol{R}_r = \boldsymbol{I}, \quad \boldsymbol{R}_t = \begin{pmatrix} 1 & \tau & \cdots & \tau \\ \tau & 1 & \cdots & \tau \\ \vdots & \vdots & \ddots & \vdots \\ \tau & \cdots & \tau & 1 \end{pmatrix}, \tag{25}$$

where $\tau \in [0, 1]$ denotes the correlation coefficient. In particular, the channel is uncorrelated if $\tau = 0$. Otherwise, the channel is somehow correlated, and the correlation level increases with a larger τ.

For the implementation of DNN, the same DNN structure is used for all the simulations. Specifically, a FCNN with 4 hidden layers is employed, and the sizes of these layers are 128, 64, 32, and 16, respectively. Moreover, the Adam optimizer with a learning rate of 10^{-6} is employed for the stochastic gradient descending during training. Each mini-batch contains 128 time slots, and the training set has a total batch size of 10 million time slots. In further, we randomly collect training samples with a SNR range from 0 dB to 30 dB.

B. Competing Algorithms

To validate the effectiveness of the proposed METS and DL-METS algorithms, the proposed algorithms will be contrasted with several competitive rivals. However, due to the lack of signal detection algorithms in DSM, we will mainly compare our proposed algorithms with the optimal MLD for the BER performance comparison, since the optimal MLD performance can be treated as the lower-bound of all detection algorithms. On the other hand, we will show the efficiency of the proposed algorithms based on the number of visited nodes during searching. In summary, we use the following abbreviations in the simulation results,

1) OWMMSE-CML: The ordered weight MMSE based conditional maximum likelihood search algorithm introduced in [29]. This algorithm will firstly enumerate all possible antenna combinations, and then compute the MMSE estimate of each antenna combination. After that, it finds the best MMSE estimate by maximum likelihood search.
2) PS-SD: Partitioning-and-ordering aid sphere decoding introduced in [30]. This algorithm can achieve the optimal BER performance as ML.
3) METS(L): The memory-bounded tree search algorithm introduced in this paper, where L denotes the number of available memory size.
4) DL-METS-Basic(L, λ): The deep learning boosted version of METS introduced in this paper, where L represents the number of available memory size, and λ is the increase factor. The notation "Basic" indicates that the algorithm will only estimate the initial radius, and it will still use the g-cost to guide the search.
5) DL-METS(L, λ): This is the full-power version of DL-METS. It will not only estimate the initial radius, but also use the f-cost to guide the search.
6) ML: the conventional maximum likelihood detection algorithm in (7), whose BER performance can serve as the lower-bound of all algorithms.

The main purpose to compare DL-METS-Basic and DL-METS is to show the effect of the following two major differences between DL-METS and METS. Specifically, by only estimating the initial radius, DL-METS-Basic is able to run much faster than METS, while it still guarantees to find the optimal ML estimate. As a contrast, DL-METS will not only estimate the initial radius, but also estimate the f-cost to boost the search process. This enables DL-METS to find the solution much faster than DL-METS-Basic. However, it becomes possible for DL-METS to miss the optimal ML estimate due to the estimation error. In other words, the optimality is not guaranteed in DL-METS, as it depends on the accuracy of the estimated heuristic.

C. Results and Discussions

Fig. 3 depicts the average number of active antennas versus the number of antennas for the proposed CM-DSM based MIMO system. In this experiment, we compare the proposed scheme with JM-VSM, which is a joint mapping method introduced in [10]. According to this figure, we can observe

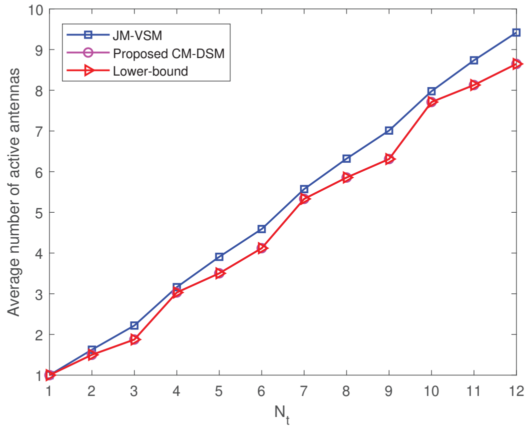

Fig. 3. Average number of active antennas versus the number of antennas for CM-DSM based MIMO systems with $N_r = N_t = N_{RF}$ and QPSK modulation.

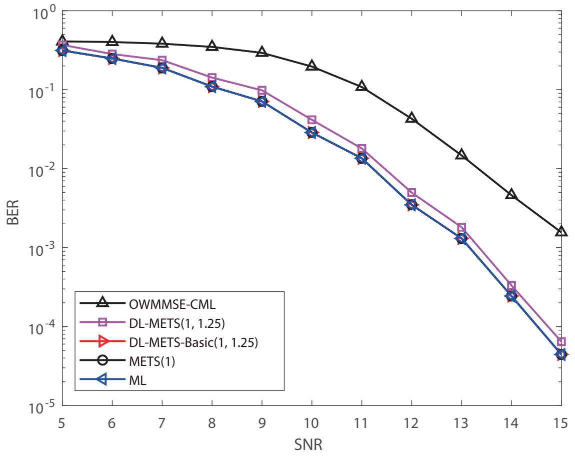

Fig. 5. BER versus SNR for CM-DSM based MIMO systems with $N_r = N_t = 12$, $N_{RF} = 8$, $\tau = 0$ and QPSK modulation.

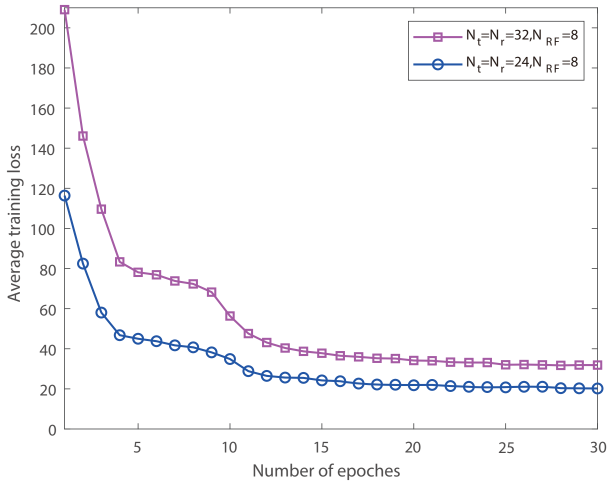

Fig. 4. Average traning loss for the proposed DL-METS in the CM-DSM based MIMO systems with $N_{RF} = 8$, $\tau = 0$ and QPSK modulation.

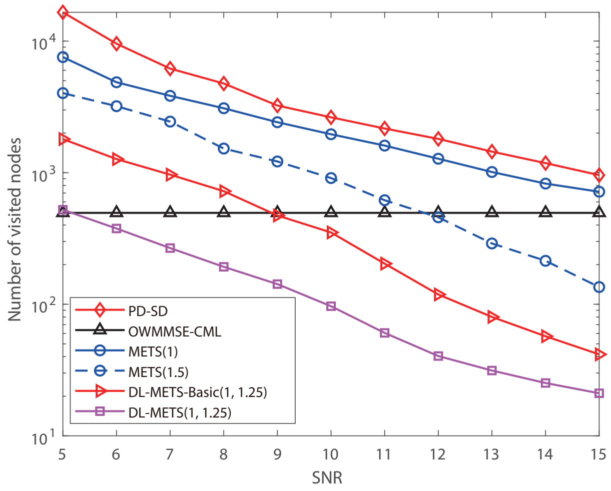

Fig. 6. Number of visited nodes versus SNR for CM-DSM based MIMO systems with $N_r = N_t = 12$, $N_{RF} = 8$, $\tau = 0$ and QPSK modulation.

that the proposed CM-DSM outperforms JM-VSM as it always activates fewer transmit antennas at different problem scales. Specifically, when $N_t = N_r = N_{RF} = 12$, the proposed CM-DSM will be expected to activate 8.6 transmit antennas while this number is 9.4 for JM-VSM. By activating fewer transmit antennas on average, the proposed CM-DSM scheme can help reduce the operating cost in practice.

Fig. 4 shows the average training loss of the proposed DL-METS in the proposed CM-DSM system with QPSK modulation, where $N_{RF} = 8$ and $\tau = 0$. We can find from this figure that there are several inflections in the deep model training of DL-METS due to the usage of stochastic gradient descending, which can help avoid local optimums in the deep search. After some numbers of searching epoches along with several inflections, our proposed training strategy can eventually converge to a low level very fast. This indicates the effectiveness of the proposed DL-METS strategy.

Fig. 5 demonstrates the BER versus SNR for the aforementioned algorithms in the proposed CM-DSM based MIMO system with QPSK modulation, where the wireless channel is uncorrelated, and the numbers of transmit antennas, receive antennas, and RF chains are set to 12, 12, and 8, respectively. From Fig. 5, we can find that the BER performances of METS and DL-METS-Basic both achieve the exact optimal ML performance, which is because that these two algorithms both guarantee to find the shortest path on the decision tree. As a contrast, the BER performance of OWMMSE-CML is far from the optimal ML performance. In addition, the BER performance of DL-METS is very close to the optimal ML performance. Specifically, when the SNR is 15 dB, the DL-METS algorithm only produce 132% error with respect to the optimal ML detection. Moreover, when the SNR is 15 dB, DL-METS can reduce the error of OWMMSE-CWL to about 4%. These results imply that the estimated heuristic is very close to the optimal heuristic such that the search is always towards to the shortest path.

In order to show the efficiency of the proposed algorithms, we further present the complexity comparison results in Fig. 6, where the results are taken under the same setting of Fig. 5. Note that we measure the complexity of the aforementioned

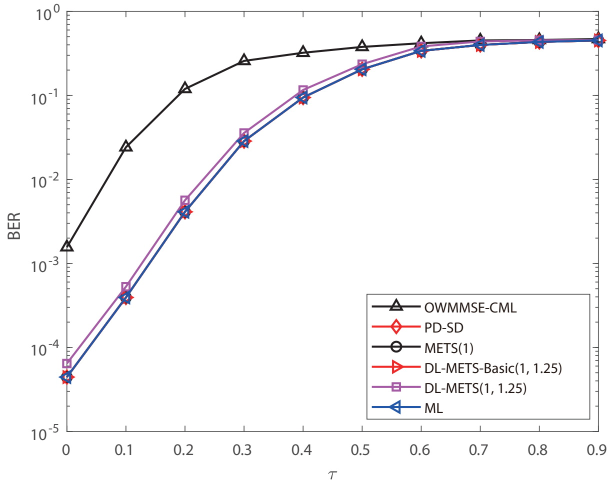

Fig. 7. BER versus correlation level for CM-DSM based MIMO systems with QPSK modulation, where $N_r = N_t = 12$, $N_{RF} = 8$, and SNR= 15 dB.

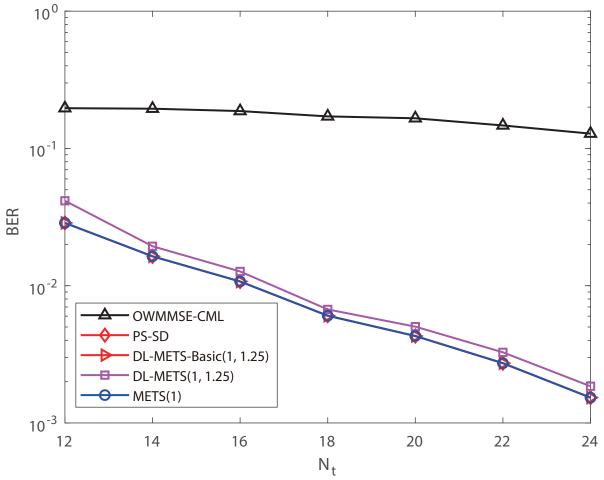

Fig. 8. BER versus the number of antennas for CM-DSM based MIMO systems with QPSK modulation, where $N_r = N_t$, $N_{RF} = 8$, $\tau = 0$ and SNR= 10 dB.

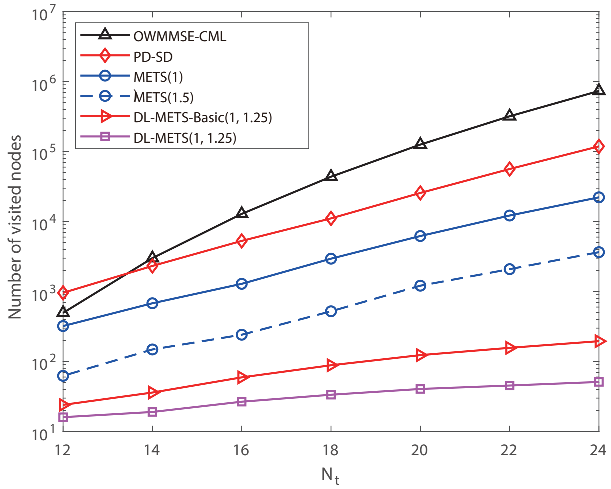

Fig. 9. Number of visited nodes versus the number of antennas for CM-DSM based MIMO systems with QPSK modulation, where $N_r = N_t$, $N_{RF} = 8$, $\tau = 0$ and SNR= 15 dB.

algorithms based on the number of visited nodes, as the computational complexity of search algorithms is mainly determined by the number of visited nodes. In particular, the total number of signal candidates in this system is 2^{25}, which can be treated as the complexity of the brute-force ML detection. From Fig. 6, we can find that the number of visited nodes of the proposed METS and DL-METS algorithms both converge to a low value with the increasing SNR. In addition, the complexity of OWMMSE-CL almost remains unchanged for different SNRs since it searches a fixed number of candidates regardless of the SNR. Obviously, by comparing to the brute-force enumeration and OWMMSE-CWL, significant complexity reduction is achieved by both METS and DL-METS. Specifically, within a wide range of SNR, the number of nodes visited by both DL-METS-Basic and DL-METS maintains a very low number which is very close to the lower bound of $N_t = 12$. Moreover, we can find that DL-METS-Basic visits much fewer nodes than METS, which indicates that the estimated initial radius is accurate such that the initial search space is significantly reduced. In further, DL-METS outperforms DL-METS-Basic in the low SNR region, which reveals that the estimated heuristic is accurate, and it can significantly boost the search process.

To further show the robustness of the proposed algorithm, simulation results under spatially correlated channels are presented in Fig. 7, where $N_r = N_t = 12$, $N_{RF} = 8$, and SNR is set to 15 dB. In this figure, the spatial correlation coefficient τ varies from 0 to 0.9. From Fig. 7, we can find that even when the channel is somehow correlated in the spatial domain, METS and DL-METS-Basic both guarantee to find the shortest path on the decision tree, as their BER performances are the same as the optimal ML performance. In addition, in the high correlated channel, DL-METS still outperforms OWMMSE-CWL, and achieves almost the optimal ML detection performance, which shows that the proposed algorithms are robust and can be applied in practical scenarios.

In order to further verify the effectiveness of the proposed algorithms, we provide the simulation results versus the number of antennas in Fig. 8 and Fig. 9, where $N_r = N_t$, $N_{RF} = 8$, and $\tau = 0$. In addition, the number of transmit antennas N_t ranges from 12 to 24, and SNR is specified as 10 dB and 15 dB in Fig. 8 and Fig. 9, respectively. We can find from Fig. 8 that the proposed METS and DL-METS still outperform OWMMSE-CML at different number of antennas. Specifically, when SNR $=$ 10 dB and $N_t = 24$, DL-METS leads to a significant reduction in the detection errors of OWMMSE-CML to about 1.4%. In addition, it can also be concluded from Fig. 8 that DL-METS maintains the near-optimal BER performance, since METS guarantees to find the shortest path and DL-METS achieves nearly the same BER performance as METS.

Moreover, by combining the results in both Figs. 8 and 9, we can find that the proposed METS and DL-METS not only achieve nearly the optimal BER performance, but also achieve much lower computational complexity with comparison to OWMMSE-CML. This is because that OWMMSE-CML enumerates all possible antenna combinations, while the proposed algorithms can efficiently prune many unnecessary nodes. In further, the complexity of the proposed DL-METS algorithm grows slowly with the increasing number of antennas, while the other competing algorithms can quickly become computationally intractable. This is because that the proposed DL-METS can accurately estimate the optimal heuristic function, which results in a very low complexity even for large-scale MIMO systems. These results provide further evidence of the effectiveness of the proposed DL-METS algorithm.

VI. Conclusions

In this paper, we investigated the signal detection problem in DSM enabled MIMO systems, where we firstly addressed the detection ambiguity problem in DSM by proposing a novel CM-DSM scheme based on the combinatorial nature of the signal structure. The proposed CM-DSM can map the information bits with fixed length into the complete set of antenna combinations and symbols. Moreover, it can reduce the number of active antennas by incorporating the priority of AACs into the mapping table. Based on the proposed CM-DSM scheme, we then constructed a decision tree for the optimal signal detection in variable active antenna systems. To find the optimal ML estimate efficiently, the METS algorithm was further proposed to prune the tree based on the combinatorial nature of CM-DSM. Furthermore, we improved the METS with a DNN boosted heuristic function to reduce the computational complexity, and thus the DL-METS was proposed to find the solution more efficiently. The proposed DL-METS can not only efficiently estimate the initial radius of search space, but also guide the search towards the most promising path by estimating the shortest remaining cost of each node. This strategy enables DL-METS to find the nearly optimal path, and meanwhile significantly reduces the number of visited nodes.

Appendix A
Examples for the Proposed Combinatorial Mapping Scheme

In this part, we provide examples for mapping bits and demapping signal vectors in the proposed CM-DSM system. For reference, the resulting mapping table of the proposed CM-DSM system with QPSK modulation is illustrated in Table I, where $N_t = N_{RF} = 3$. In this table, the zero elements in AACs and transmitted signal vectors indicate that the corresponding transmit antenna is inactivated. It should be noted that the following examples will be presented based on this system as well.

TABLE I
3D-Mapping Table for CM-DSM based MIMO Systems with QPSK modulation, where $N_t = N_{RF} = 3$.

Index	Bits	AAC	Transmitted Signal Vector
0	[0 0 0 0 0 0]	[1 0 0]	$[-1-1j,+0+0j,+0+0j]$
1	[0 0 0 0 0 1]	[1 0 0]	$[-1+1j,+0+0j,+0+0j]$
2	[0 0 0 0 1 0]	[1 0 0]	$[+1-1j,+0+0j,+0+0j]$
3	[0 0 0 0 1 1]	[1 0 0]	$[+1+1j,+0+0j,+0+0j]$
4	[0 0 0 1 0 0]	[0 1 0]	$[+0+0j,-1-1j,+0+0j]$
5	[0 0 0 1 0 1]	[0 1 0]	$[+0+0j,-1+1j,+0+0j]$
6	[0 0 0 1 1 0]	[0 1 0]	$[+0+0j,+1-1j,+0+0j]$
7	[0 0 0 1 1 1]	[0 1 0]	$[+0+0j,+1+1j,+0+0j]$
8	[0 0 1 0 0 0]	[0 0 1]	$[+0+0j,+0+0j,-1-1j]$
9	[0 0 1 0 0 1]	[0 0 1]	$[+0+0j,+0+0j,-1+1j]$
10	[0 0 1 0 1 0]	[0 0 1]	$[+0+0j,+0+0j,+1-1j]$
11	[0 0 1 0 1 1]	[0 0 1]	$[+0+0j,+0+0j,+1+1j]$
12	[0 0 1 1 0 0]	[1 1 0]	$[-1-1j,-1-1j,+0+0j]$
13	[0 0 1 1 0 1]	[1 1 0]	$[-1-1j,-1+1j,+0+0j]$
14	[0 0 1 1 1 0]	[1 1 0]	$[-1-1j,+1-1j,+0+0j]$
15	[0 0 1 1 1 1]	[1 1 0]	$[-1-1j,+1+1j,+0+0j]$
16	[0 1 0 0 0 0]	[1 1 0]	$[-1+1j,-1-1j,+0+0j]$
17	[0 1 0 0 0 1]	[1 1 0]	$[-1+1j,-1+1j,+0+0j]$
18	[0 1 0 0 1 0]	[1 1 0]	$[-1+1j,+1-1j,+0+0j]$
19	[0 1 0 0 1 1]	[1 1 0]	$[-1+1j,+1+1j,+0+0j]$
20	[0 1 0 1 0 0]	[1 1 0]	$[+1-1j,-1-1j,+0+0j]$
21	[0 1 0 1 0 1]	[1 1 0]	$[+1-1j,-1+1j,+0+0j]$
22	[0 1 0 1 1 0]	[1 1 0]	$[+1-1j,+1-1j,+0+0j]$
23	[0 1 0 1 1 1]	[1 1 0]	$[+1-1j,+1+1j,+0+0j]$
24	[0 1 1 0 0 0]	[1 1 0]	$[+1+1j,-1-1j,+0+0j]$
25	[0 1 1 0 0 1]	[1 1 0]	$[+1+1j,-1+1j,+0+0j]$
26	[0 1 1 0 1 0]	[1 1 0]	$[+1+1j,+1-1j,+0+0j]$
27	[0 1 1 0 1 1]	[1 1 0]	$[+1+1j,+1+1j,+0+0j]$
28	[0 1 1 1 0 0]	[1 0 1]	$[-1-1j,+0+0j,-1-1j]$
29	[0 1 1 1 0 1]	[1 0 1]	$[-1-1j,+0+0j,-1+1j]$
30	[0 1 1 1 1 0]	[1 0 1]	$[-1-1j,+0+0j,+1-1j]$
31	[0 1 1 1 1 1]	[1 0 1]	$[-1-1j,+0+0j,+1+1j]$
32	[1 0 0 0 0 0]	[1 0 1]	$[-1+1j,+0+0j,-1-1j]$
33	[1 0 0 0 0 1]	[1 0 1]	$[-1+1j,+0+0j,-1+1j]$
34	[1 0 0 0 1 0]	[1 0 1]	$[-1+1j,+0+0j,+1-1j]$
35	[1 0 0 0 1 1]	[1 0 1]	$[-1+1j,+0+0j,+1+1j]$
36	[1 0 0 1 0 0]	[1 0 1]	$[+1-1j,+0+0j,-1-1j]$
37	[1 0 0 1 0 1]	[1 0 1]	$[+1-1j,+0+0j,-1+1j]$
38	[1 0 0 1 1 0]	[1 0 1]	$[+1-1j,+0+0j,+1-1j]$
39	[1 0 0 1 1 1]	[1 0 1]	$[+1-1j,+0+0j,+1+1j]$
40	[1 0 1 0 0 0]	[1 0 1]	$[+1+1j,+0+0j,-1-1j]$
41	[1 0 1 0 0 1]	[1 0 1]	$[+1+1j,+0+0j,-1+1j]$
42	[1 0 1 0 1 0]	[1 0 1]	$[+1+1j,+0+0j,+1-1j]$
43	[1 0 1 0 1 1]	[1 0 1]	$[+1+1j,+0+0j,+1+1j]$
44	[1 0 1 1 0 0]	[0 1 1]	$[+0+0j,-1-1j,-1-1j]$
45	[1 0 1 1 0 1]	[0 1 1]	$[+0+0j,-1-1j,-1+1j]$
46	[1 0 1 1 1 0]	[0 1 1]	$[+0+0j,-1-1j,+1-1j]$
47	[1 0 1 1 1 1]	[0 1 1]	$[+0+0j,-1-1j,+1+1j]$
48	[1 1 0 0 0 0]	[0 1 1]	$[+0+0j,-1+1j,-1-1j]$
49	[1 1 0 0 0 1]	[0 1 1]	$[+0+0j,-1+1j,-1+1j]$
50	[1 1 0 0 1 0]	[0 1 1]	$[+0+0j,-1+1j,+1-1j]$
51	[1 1 0 0 1 1]	[0 1 1]	$[+0+0j,-1+1j,+1+1j]$
52	[1 1 0 1 0 0]	[0 1 1]	$[+0+0j,+1-1j,-1-1j]$
53	[1 1 0 1 0 1]	[0 1 1]	$[+0+0j,+1-1j,-1+1j]$
54	[1 1 0 1 1 0]	[0 1 1]	$[+0+0j,+1-1j,+1-1j]$
55	[1 1 0 1 1 1]	[0 1 1]	$[+0+0j,+1-1j,+1+1j]$
56	[1 1 1 0 0 0]	[0 1 1]	$[+0+0j,+1+1j,-1-1j]$
57	[1 1 1 0 0 1]	[0 1 1]	$[+0+0j,+1+1j,-1+1j]$
58	[1 1 1 0 1 0]	[0 1 1]	$[+0+0j,+1+1j,+1-1j]$
59	[1 1 1 0 1 1]	[0 1 1]	$[+0+0j,+1+1j,+1+1j]$
60	[1 1 1 1 0 0]	[1 1 1]	$[-1-1j,-1-1j,-1-1j]$
61	[1 1 1 1 0 1]	[1 1 1]	$[-1-1j,-1-1j,-1+1j]$
62	[1 1 1 1 1 0]	[1 1 1]	$[-1-1j,-1-1j,+1-1j]$
63	[1 1 1 1 1 1]	[1 1 1]	$[-1-1j,-1-1j,+1+1j]$

A. Mapping Bits

Suppose that the input bits are $\boldsymbol{b} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$, which corresponds to the decimal number of $n = 38$. At the beginning, we need to find the number of active antennas. This is done by finding the least N_a satisfying $\sum_{i=1}^{N_a} \binom{N_t}{i} |\mathcal{X}|^i > 38$. In this case, we will have $N_a = 2$, which means that we should activate 2 antennas for transmission. With $N_a = 2$, we know that the input bits should be the $r = 38 - \sum_{i=1}^{N_a} \binom{N_t}{i} |\mathcal{X}|^i = 26$-th candidate from the set of all the possible candidates with $N_a = 2$. Since each AAC will cover at most $|\mathcal{X}|^{N_a} = 16$

symbol combinations, we know that the combinatorial index of the corresponding AAC should be $k = \left\lfloor \frac{r}{|\mathcal{X}|^{N_a}} \right\rfloor = 1$. Based on this index, we can get the AAC as $\begin{bmatrix}1 & 0 & 1\end{bmatrix}$ by decoding the combinatorial index k with the decoding algorithm in Algorithm 1. After that, we need to determine the symbols to be transmitted with the active antennas. Since $q = r - k|\mathcal{X}|^{N_a} = 10$, we know that the corresponding symbol combination should be the 10-th combination with the same AAC of $\begin{bmatrix}1 & 0 & 1\end{bmatrix}$. Base on this, we convert q into a binary sequence of $\hat{\boldsymbol{b}} = \begin{bmatrix}1 & 0 & 1 & 0\end{bmatrix}$. By mapping $\hat{\boldsymbol{b}}$ with conventional QPSK constellation, we have the symbol vector as $\hat{\boldsymbol{s}} = \begin{bmatrix}1-1j & 1-1j\end{bmatrix}$. Finally, the complete signal vector $\boldsymbol{s} = \begin{pmatrix}1-1j & 0+0j & 1-1j\end{pmatrix}$ can be generated by combining the QPSK symbols with the selected AAC.

B. Demapping Signal Vector

In principle, the demapping process is just the reversion of mapping bits. Specifically, for the received signal $\boldsymbol{s} = \begin{pmatrix}1-1j & 0+0j & 1-1j\end{pmatrix}$, we have $N_a = 2$, and the corresponding AAC is $\begin{bmatrix}1 & 0 & 1\end{bmatrix}$. Then, we can get the combinatorial index of this AAC by applying the encoding algorithm in Algorithm 1, which should be $k = 1$. Besides, we extract the QPSK symbols from the received signal as $\hat{\boldsymbol{s}} = \begin{bmatrix}1-1j & 1-1j\end{bmatrix}$, and demap it as the bits of $\hat{\boldsymbol{b}} = \begin{bmatrix}1 & 0 & 1 & 0\end{bmatrix}$. After that, we get the decimal number of $\hat{\boldsymbol{b}}$ as $q = 10$, and we have $r = k|\mathcal{X}|^{N_a} + q = 26$. Hence, the decimal number for the resulting bits is given by $n = r + \sum_{i=1}^{N_a-1} \binom{N_t}{i} |\mathcal{X}|^i = 38$, which corresponds to the binary sequence of $\boldsymbol{b} = \begin{bmatrix}1 & 0 & 0 & 1 & 1 & 0\end{bmatrix}$.

References

[1] M. Giordani, M. Polese, M. Mezzavilla, S. Rangan, and M. Zorzi, "Toward 6G networks: Use cases and technologies," *IEEE Commun. Mag.*, vol. 58, no. 3, pp. 55–61, 2020.

[2] M. D. Renzo, H. Haas, A. Ghrayeb, S. Sugiura, and L. Hanzo, "Spatial modulation for generalized MIMO: Challenges, opportunities, and implementation," *Proc. IEEE*, vol. 102, no. 1, pp. 56–103, 2014.

[3] D. A. Basnayaka, M. D. Renzo, and H. Haas, "Massive but few active MIMO," *IEEE Trans. Veh. Technol.*, vol. 65, no. 9, pp. 6861–6877, 2016.

[4] M. Wen, B. Zheng, K. J. Kim, M. D. Renzo, T. A. Tsiftsis, K. Chen, and N. Al-Dhahir, "A survey on spatial modulation in emerging wireless systems: Research progresses and applications," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 9, pp. 1949–1972, 2019.

[5] M. Wen, X. Cheng, and L. Yang, *Index Modulation for 5G Wireless Communications*. Cham, Switzerland: Springer, 2017.

[6] R. Mesleh, H. Haas, S. Sinanovic, C. W. Ahn, and S. Yun, "Spatial modulation," *IEEE Trans. Veh. Technol.*, vol. 57, no. 4, pp. 2228–2241, 2008.

[7] J. Wang, S. Jia, and J. Song, "Generalised spatial modulation system with multiple active transmit antennas and low complexity detection scheme," *IEEE Trans. Wirel. Commun.*, vol. 11, no. 4, pp. 1605–1615, 2012.

[8] A. Younis, R. Mesleh, M. D. Renzo, and H. Haas, "Generalised spatial modulation for large-scale MIMO," in *22nd European Signal Processing Conference, EUSIPCO 2014, Lisbon, Portugal, September 1-5, 2014*. IEEE, 2014, pp. 346–350.

[9] J. Li, S. Dang, Y. Yan, Y. Peng, S. Al-Rubaye, and A. Tsourdos, "Generalized quadrature spatial modulation and its application to vehicular networks with NOMA," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4030–4039, 2021.

[10] S. Gadhai and R. Budhiraja, "Joint-mapping-based variable active antenna spatial modulation," *IEEE Wirel. Commun. Lett.*, vol. 9, no. 10, pp. 1668–1672, 2020.

[11] P. Liu, M. D. Renzo, and A. Springer, "Variable-nu generalized spatial modulation for indoor LOS mmWave communication: Performance optimization and novel switching structure," *IEEE Trans. Commun.*, vol. 65, no. 6, pp. 2625–2640, 2017.

[12] Y. Yang and M. Guizani, "Mapping-varied spatial modulation for physical layer security: Transmission strategy and secrecy rate," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 4, pp. 877–889, 2018.

[13] Y. Xiao, Z. Yang, L. Dan, P. Yang, L. Yin, and W. Xiang, "Low-complexity signal detection for generalized spatial modulation," *IEEE Commun. Lett.*, vol. 18, no. 3, pp. 403–406, 2014.

[14] C. Lin, W. Wu, and C. Liu, "Low-complexity ML detectors for generalized spatial modulation systems," *IEEE Trans. Commun.*, vol. 63, no. 11, pp. 4214–4230, 2015.

[15] L. Xiao, P. Xiao, Y. Xiao, H. Haas, A. Mohamed, and L. Hanzo, "Compressive sensing assisted generalized quadrature spatial modulation for massive MIMO systems," *IEEE Trans. Commun.*, vol. 67, no. 7, pp. 4795–4810, 2019.

[16] T. Liu, C. Chen, and C. Liu, "Fast maximum likelihood detection of the generalized spatially modulated signals using successive sphere decoding algorithms," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 656–659, 2019.

[17] V. M. García-Molla, F. Martínez-Zaldívar, M. Á. Simarro, and A. González, "Maximum likelihood low-complexity GSM detection for large MIMO systems," *Signal Process.*, vol. 175, p. 107661, 2020.

[18] Z. Ma, F. Gao, J. Jiang, and Y. Liang, "Cooperative detection for ambient backscatter assisted generalized spatial modulation," in *2019 IEEE Global Communications Conference, GLOBECOM 2019, Waikoloa, HI, USA, December 9-13, 2019*. IEEE, 2019, pp. 1–6.

[19] C. Wang, P. Cheng, Z. Chen, J. A. Zhang, Y. Xiao, and L. Gui, "Near-ML low-complexity detection for generalized spatial modulation," *IEEE Commun. Lett.*, vol. 20, no. 3, pp. 618–621, 2016.

[20] L. Xiao, P. Yang, Y. Xiao, S. Fan, M. D. Renzo, W. Xiang, and S. Li, "Efficient compressive sensing detectors for generalized spatial modulation systems," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 1284–1298, 2017.

[21] J. Cal-Braz and R. S. Neto, "Low-complexity sphere decoding detector for generalized spatial modulation systems," *IEEE Commun. Lett.*, vol. 18, no. 6, pp. 949–952, 2014.

[22] B. Zheng, X. Wang, M. Wen, and F. Chen, "Soft demodulation algorithms for generalized spatial modulation using deterministic sequential Monte Carlo," *IEEE Trans. Wirel. Commun.*, vol. 16, no. 6, pp. 3953–3967, 2017.

[23] B. Zheng, M. Wen, F. Chen, N. Huang, F. Ji, and H. Yu, "The K-best sphere decoding for soft detection of generalized spatial modulation," *IEEE Trans. Commun.*, vol. 65, no. 11, pp. 4803–4816, 2017.

[24] T. Q. Tran, S. Sugiura, and K. Lee, "Ordering-and partitioning-aided sphere decoding for generalized spatial modulation," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 10 087–10 091, 2018.

[25] J. Liao, J. Zhao, F. Gao, and G. Y. Li, "A model-driven deep learning method for massive MIMO detection," *IEEE Commun. Lett.*, vol. 24, no. 8, pp. 1724–1728, 2020.

[26] J. Sun, Y. Zhang, J. Xue, and Z. Xu, "Learning to search for MIMO detection," *IEEE Trans. Wirel. Commun.*, vol. 19, no. 11, pp. 7571–7584, 2020.

[27] J. Zhang, Y. He, Y. Li, C. Wen, and S. Jin, "Meta learning-based MIMO detectors: Design, simulation, and experimental test," *IEEE Trans. Wirel. Commun.*, vol. 20, no. 2, pp. 1122–1137, 2021.

[28] H. Huo, J. Xu, G. Su, W. Xu, and N. Wang, "Intelligent MIMO detection using meta learning," *IEEE Wirel. Commun. Lett.*, vol. 11, no. 10, pp. 2205–2209, 2022.

[29] V. V. Gudla and V. B. Kumaravelu, "Dynamic spatial modulation for next generation networks," *Phys. Commun.*, vol. 34, pp. 90–104, 2019.

[30] B. Li, Y. Wu, X. Jiang, and E. Bai, "Transmit antenna combination optimization and detection algorithm for joint-mapping-based variable active antenna spatial modulation system," in *2021 7th International Conference on Computer and Communications (ICCC)*. IEEE, 2021, pp. 2190–2194.

[31] J. P. M. Schalkwijk, "An algorithm for source coding," *IEEE Trans. Inf. Theory*, vol. 18, no. 3, pp. 395–399, 1972.

[32] K. He, L. He, L. Fan, X. Lei, Y. Deng, and G. K. Karagiannidis, "Efficient memory-bounded optimal detection for GSM-MIMO systems," *IEEE Trans. Commun.*, vol. 70, no. 7, pp. 4359–4372, 2022.

[33] Y. Dai and Z. Yan, "Memory-constrained tree search detection and new ordering schemes," *IEEE J. Sel. Top. Signal Process.*, vol. 3, no. 6, pp. 1026–1037, 2009.

[34] G. Zhu, K. Huang, V. K. N. Lau, B. Xia, X. Li, and S. Zhang, “Hybrid beamforming via the Kronecker decomposition for the millimeter-wave massive MIMO systems,” *IEEE J. Sel. Areas Commun.*, vol. 35, no. 9, pp. 2097–2114, 2017.