Probabilistic Sorting Memory Constrained Tree Search Algorithm for MIMO System

Xiaoping Jin[1], Zheng Guo[1(✉)], Ning Jin[1], and Zhengquan Li[2,3]

[1] Department of Information Engineering, China Jiliang University, Hang Zhou 310018, China
{jxp1023, jinning1117}@cjlu.edu.cn, Guozheng311@sina.com

[2] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
lzq722@sina.com

[3] National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China

Abstract. Considering the shortcomings of large storage space requirements and high complexity in multiple-symbol differential detection algorithm in current Multiple Input Multiple Output (MIMO) system, this paper proposes a probabilistic sorting memory constrained tree search algorithm (PSMCTS) by using performance advantage of sorting algorithm and storage advantage of memory constrained tree search (MCTS). Based on PSMCTS, a pruning PSMCTS named PPSMCTS is put forward. Simulation results show that the performance of PSMCTS is approach to that of ML algorithm under fixed memory situations, while the computational complexity is lower than that of MCTS algorithm in small storage capacity conditions under low signal noise ratio (SNR) region. PPSMCTS has more prominent advantages on reduction of computational complexity than PSMCTS algorithm. Theoretical analysis and simulation demonstrate that the two proposed algorithms can effectively inherit the good feature of MCTS algorithm, which are suitable for hardware implementation.

Keywords: MIMO · Probabilistic sorting · Memory constrained tree search
Pruning algorithm

1 Introduction

Recent years, the combination of Multiple Input Multiple Output (MIMO) technology and Orthogonal Frequency Division Multiplexing (OFDM) technology expands the application of MIMO system greatly, makes the system works more efficiently in frequency selective fading environment. However, under severe channel states, for example high-speed mobile condition, it is very difficult for the receiver to obtain the channel state information. Therefore, differential encoded signaling combined with low-complexity differential detection at the receiver becomes an attractive design alternative. But 3 dB performance loss would be paid compared with traditional correlation detection [1]. Then multiple-symbol differential detection (MSDD) which

L. Meng and Y. Zhang (Eds.): MLICOM 2018, LNICST 251, pp. 411–425, 2018.
https://doi.org/10.1007/978-3-030-00557-3_41

using N + 1 received symbols to detect N symbols (N is regarded as observation window length or block length) is proposed as an effective solution to this problem. The increasing length of observation window can effectively shorten the performance gap of 3 dB [2].

At present, most multiple-symbol differential detection algorithms are based on tree-searching principle [1, 3]. Maximum likelihood (ML) detection [2] is the most representative algorithm of best performance. But the exhaustive search strategy makes its complexity increase in an exponential relationship with the block length and the number of antenna, which leads to a computationally intractable problem. Therefore, some detection algorithms with lower complexity were proposed [4–9]. Around these algorithms, there are three kinds of search strategies in general: depth-first [4, 5], breadth-first, and metric-first. Sphere detection (SD) is a typical depth-first searching detection algorithm. Due to its continuous backtracking, this algorithm has different throughput when in different channel environment, which does not lend itself to parallel and pipeline processing. Breadth-first search strategy [6, 10, 11], such as K-BEST algorithm, has high throughput and stable complexity, which is suitable for pipeline processing, but the K value constraint brings loss in performance. Stack algorithm [7] mainly based on metric-first strategy, as named as Dijkstra algorithm [8, 9], it always extends the node with the minimum metric value in measure list, so it has least visited node number among the three search strategies.

However, the hardware implementation of these algorithms usually has high computational complexity and requires large storage capacity. But in practice, the storage space is confining, which limits the algorithm performance. MCTS (Memory Constrained Tree Search) proposed in paper [12] gave a solution to this problem. It can approximate the performance of ML algorithm under any storage space situation. When the storage space is set as minimum value or maximum value, the performance of the MCTS algorithm approaches to that of the SD algorithm and stack algorithm respectively. Moreover, the average computational complexity reduces with the increasing of storage space. It possesses a good compromise between memory requirement and computational complexity. But, the complexity is still high when in small memory space because of approximating SD strategy.

DSPS (Dijkstra Search with Probabilistic Sorting) algorithm is a new tree search algorithm proposed by Chang [13, 14]. Compared with the Dijkstra algorithm of full search, DSPS greatly reduces the number of visited nodes and effectively enhances the bit error ratio (BER) performance by using mathematical statistical probability on the nodes. It has high research value in respect of saving storage space and reducing complexity.

In this context, we focus on using efficient methods to improve MCTS algorithm, which aiming to reduce the computational complexity of MSDD MIMO system, especially under memory constrained situation. We make the following contributions:

1. We propose a new memory constrained search algorithm - PSMCTS (probabilistic sorting memory constrained tree search), in which the DSPS merges into the MCTS algorithm to reduce the access node number and improve the decision accuracy of the MCTS algorithm.

2. To enhance the PSMCTS's advantage of low complexity under small memory size, a pruning algorithm is applied into PSMCTS. The improved scheme is called PPSMCTS. The key work here is how to decide the pruning threshold and use it to prune the searching tree layer by layer.

The rest of the paper is organized as follows. Section 2 presents the system model and signal construction. Section 3 introduces the PSMCTS algorithm applied in our MSDD MIMO system. In Sect. 4, PPMCTS algorithm is proposed. Section 5 provides system complexity and performance analysis, and Sect. 6 concludes the paper.

2 System Model

We consider a MIMO-OFDM system with N_R receive and N_T transmit antennas and communicating over a quasi-static, frequency-flat fading channel. The system diagram is shown in Fig. 1, in which the MSDD block is the focus of our research.

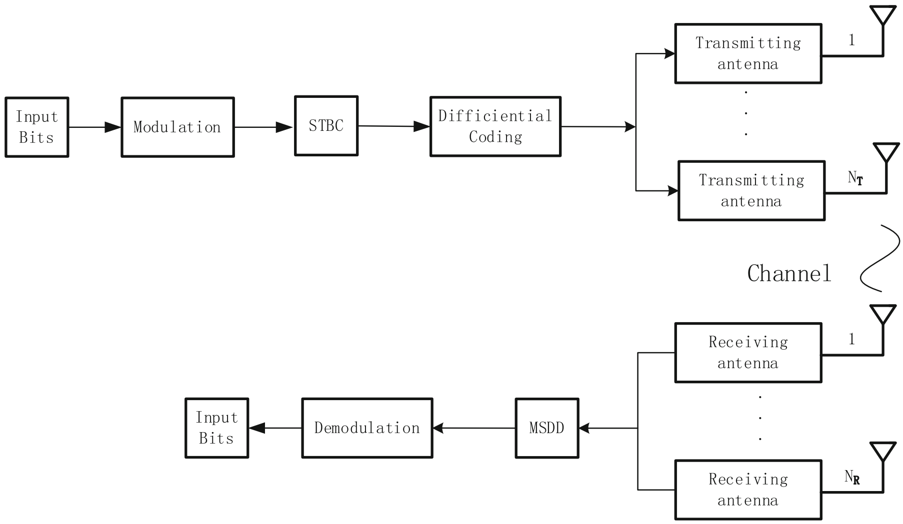

Fig. 1. Block diagram of MIMO system

In Fig. 1, the space-time block coding (STBC) module is constructed on Alamouti's transmit diversity scheme when $N_T = 2$. Other scheme can also be used, depending on the number of antennas. Define information matrix $S_t = \begin{bmatrix} s_{1,t} & s_{2,t} \\ -s_{2,t}^* & -s_{1,t}^* \end{bmatrix}$, where $S_{1,t}$ and $S_{2,t}$ belongs to a L-PSK modulation constellation collection V and

$$V = \{e^{j2\pi(m-1)/L} | m = 1, 2 \cdots, L\} \tag{1}$$

S_t satisfies $S_t S_t^H = I_2$, $(\cdot)^*$ means conjugate. For differential coding, setting reference matrix $C_0 = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}$, the coding rule is

$$C_t = S_t C_{t-1} \tag{2}$$

where C_t denotes differential coding matrix. After differential coding, the data is transmitted through the space-time matrix using different multipath channel model. Now the receive signal at t time is

$$R_t = C_t H_t = W_t \tag{3}$$

where $R_t = \begin{bmatrix} r_{1,t} \\ r_{2,t} \end{bmatrix}, C_t = \begin{bmatrix} c_{1,t} & c_{2,t} \\ -c_{2,t}^* & -c_{1,t}^* \end{bmatrix}, H_t = \begin{bmatrix} h_{1,1} \\ h_{2,1} \end{bmatrix}$ is channel matrix, in which each element $h_{i,1}(i = 1, 2)$ follows Gauss distribution with 0 mean and variance σ_H^2, $W_t = \begin{bmatrix} w_{1,t} \\ w_{2,t} \end{bmatrix}$ is a noise matrix, in which each element follows Gauss distribution with mean 0 and variance σ_W^2.

Assuming the window length of multiple-symbol differential detection (MSDD) is N + 1, namely the receiver continuously receives N + 1 symbols to detect N symbols. The ML decision criterion is based on the following formula (proof see Appendix A)

$$\hat{V}_{ML} = \underset{V_{t+1},\ldots,V_{t+N}}{\arg\min} \sum_{i=1}^{N} \sum_{l=i+1}^{N+1} ||R[l+t-1] - (\prod_{m=i+t}^{l+t-1} V[m]) \times R[i+t-1]||_F^2 \tag{4}$$

where t denotes the start time of detection.

3 PSMCTS Algorithm

In MCTS algorithm, (4) is used as metric decision, and the visiting node selection is restricted to the storage space and the metric value. It visits the node with minimum metric value per time. This feature can reduce the requirement of storage space. But in small storage space condition, MCTS tends to use a depth-first search strategy, like sphere detection (SD) algorithm, which still needs to backtrack to visit a large number of nodes and the computational complexity is still large.

To optimize formula (4), we use the cumulative distribution function of [10] as the decision metric

$$\hat{F} = \arg\min F(D; k) = \arg\min \frac{\gamma(k/2, D/\sigma^2)}{\Gamma(k/2)} \tag{5}$$

where $D = \sum_{i=1}^{N} \sum_{l=i+1}^{N+1} ||R[l+t-1] - (\prod_{m=i+t}^{l+t-1} V[m]) \times R[i+t-1]||$ has k-dimensional chi-square distribution (proof see Appendix B).

Based on MCTS and (5), combined with Dijkstra algorithm characteristic, we propose a probabilistic sorting memory constrained tree search algorithm (PSMCTS), its search procedure is as follows:

1. According to the system requirements, modulation constellation size is L. Initialize the available storage space number with $M, M \geq (N-1)(L-1)+1$. Initialize multiple window length with N, and send the receiving signals with block size $N + 1$ into PSMCTS decoder.
2. Set tree level $K = N$, which represents the search starts from the tree root. If $K \neq 2$, expend the root node to L child nodes, save them into memory storage and delete the root node. Do search process of MCTS algorithm according to (5).
   (a) Start from $K = N$, namely from $V[t+N]$. According to PSMCTS, expand L child nodes, save them into memory storage and delete the node itself. Choose the best node from storage which satisfies (5), then expand it to next level. Save L expanding nodes into storage and delete the chosen best node.
   (b) Expand the best node from upper level, and save L child nodes into memory storage. In condition of low storage space, due to the stored branch node number is small, choose the best branch node to expand directly. If $K = 1$, output the node with minimum value directly. If storage space is enough, retain multiple branch nodes. Expand the best node and add L child nodes into memory storage. Do stack algorithm in the storage. Repeat the above steps until the best leaf node is found.
   (c) Repeat the above steps until the bottom of the tree is reached. Then output the best path.
3. If $K = 2$, find the best leaf node, and output the best path.

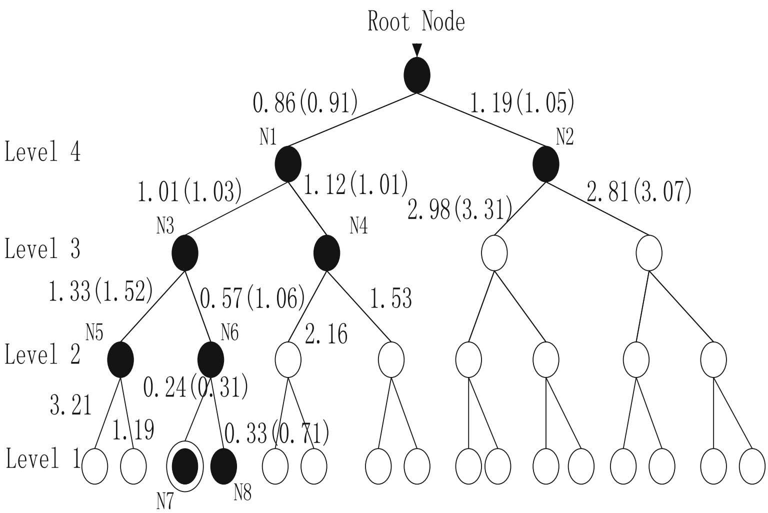

Fig. 2. Tree search analysis of PSMCTS

The search demos of PSMCTS and MCTS are shown as Figs. 2 and 3, respectively. In the figures, ○ denotes the branch nodes which are not visited. ● denotes the visited nodes. ⦿ denotes the best path node. Figure 2 shows tree search analysis of PSMCTS. In this figure, numerical value is probabilistic metric value, which in parenthesis is traditional metric value. After the iterative computation runs to the step when *N*3 is selected, the node list contains four nodes *N*2, *N*4, *N*5, *N*6, which ordered as *N*2, *N*6, *N*4, *N*5 by traditional metric, and *N*2 will be selected as the best node to iterative computation for next round. But the four nodes will be ordered as *N*6, *N*4, *N*5, *N*2 by probabilistic metric, *N*6 will be chosen directly as the best node for next round iterative computation. Figure 3 is the chart of tree search analysis for MCTS. In this figure, numerical value is traditional metric value. It is intuitively show that the PSMCTS algorithm has the advantage of less visited nodes as compare to MCTS. In addition, in order to express the search process advantage more intuitively, Tables 1 and 2 show the specific search storage state of PSMCTS and MCTS respectively, where the bold number denotes visited node chosen for expanding.

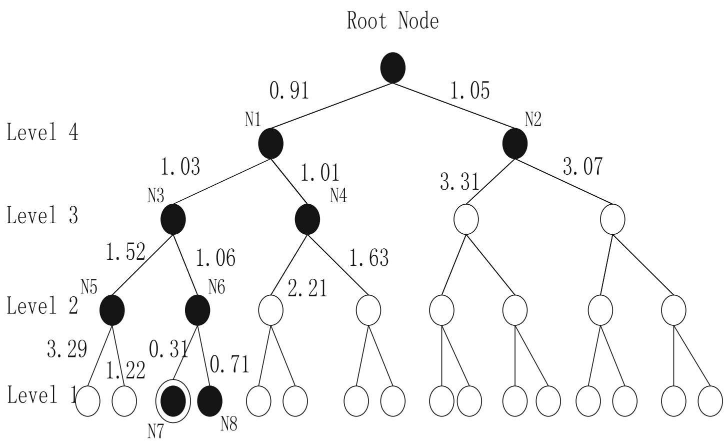

Fig. 3. Tree search analysis of MCTS

The two tables below show the use of storage of the two algorithms respectively. In PSMCTS, the storage space of each layer only needs to retain 3 branch nodes, while MCTS needs to retain 4 nodes at least. This reveals the PSMCTS algorithm has advantage in reducing storage space requirement, and this advantage will increase along with the number of constellation points. The tables also show the advantages of simplified steps in PSMCTS algorithm. Because it can accurately represent node metrics, the storage space is reduced, the number of visited nodes is reduced and the tree search process is accelerated, which makes the algorithm more effective and fast.

Table 1. PSMCTS search state

Visited node	Memory nodes metric (level)	Visited level
(1)	**0.86(4)**, 1.19(4)	4
(2)	**1.01(3)**, 1.21(3), 1.19(4)	3
(3)	1.33(2), **0.57(2)**, 1.21(3)	2
(4)	**0.24(1)**, 0.33(1)	2
(5)	**0.24(1)**	1

Table 2. MCTS search state

Visited node	Memory nodes metric (level)	Visited level
(1)	**0.91(4)**, 1.05(4)	4
(2)	**1.01(3)**, 1.03(3), 1.05(4)	3
(3)	1.03(3), 1.63(2), 2.21(2), 1.05(4)	2
(4)	**1.06(2)**, 1.52(2), 2.21(2), 1.63(2)	3
(5)	**0.31(1)**, 0.71(1), 1.52(2)	2
(6)	**0.31(1)**	1

4 PPSMCTS

As introduced in Sect. 3, the PSMCTS algorithm can adjust its search strategy dynamically according to the memory size. When the pre-specified memory size is very small, the computational complexity cannot but become large. As the pre-specified memory size increases, the average of the computational complexity will decrease. The key point is to find the best trade-off between memory requirement and computational complexity. Aiming at reducing the computational complexity further on the constraint of small memory, this part puts forward a pruning PSMCTS algorithm called PPSMCTS.

The PPSMCTS algorithm is proposed based on the PSMCTS algorithm. The difference is that, in PPSMCTS, we set a pre-specified pruning threshold and use it to prune the searching tree layer by layer, only those nodes whose metric are smaller than the pruning threshold are retained. The pruning process can reduce the total number of visited nodes throughout detecting and it is especially effective in low SNR region with small memory constraint. When SNR is high, the number of visited nodes can reach to only N nodes.

The pruning threshold is a key fact. If the value is too large, the memory will contain lots of useless nodes and will not be able to reduce the complexity. And if the value is too small, the probability of removing the maximum likelihood solution will be increased. It will affect the performance of detection.

In PPSMCTS algorithm, from formula (A.3 in Appendix A), the metric (i.e., weight) of the node $S(n)$ atn $(1 \leq n \leq N)$ level of weighted L-ray tree is

$$d_n = \sum_{n_R}^{N_R} \mathbf{R}_{n_R}^H(n)\Lambda_n^{-1}\mathbf{R}_{n_R}(n) \tag{6}$$

where $\Lambda_n = S(n)(\mathbf{C}_{R,n} \otimes \mathbf{I}_{N_T})S^H(n)$. For the root node, $d_0 = 0$. Th_n denotes the corresponding pruning threshold of the n-level. The retained probability of node $S(n)$ after pruning operation is taken as $\Pr\{d_n \leq Th_n\} = 1 - \varepsilon$. In order to ensure the BER performance, the pre-specified probability can be set as $P_0 = 1 - \varepsilon$. Then the pruning threshold Th_n should satisfy

$$\Pr\left\{ d_n = \sum_{n_R}^{N_R} R_{n_R}^H(n)\Lambda_n^{-1}R_{n_R}(n) \leq Th_n \right\} = P_0 = 1 - \varepsilon \tag{7}$$

where $R_{n_R}(n)$ meets $CN(0, \Lambda_n)$ distribution, and its quadratic form $2R_{n_R}^H(n)\Lambda_n^{-1}R_{n_R}(n)$ meets $\chi^2_{2(n+1)N_T}$ distribution of $2(n+1)N_T$ degree of freedom. Owing to the statistical independence of $R_{n_R}(n), n_R = 1, 2, \ldots, N_R, 2d_n = 2\sum_{n_R}^{N_R} R_{n_R}^H(n)\Lambda_n^{-1}R_{n_R}(n)$ meets $\chi^2_{2(n+1)N_TN_R}$ distribution of $2(n+1)N_TN_R$ degree of freedom. Substitute these data into (7), we get

$$Th_n = \frac{\left(\chi^2_{2(n+1)N_TN_R}(P_0)\right)^{-1}}{2} \tag{8}$$

Here, the superscript '−1' represents the inverse of Chi square distribution. (9) is then obtained according to the Chi square distribution of $2(n+1)N_TN_R$ degree of freedom.

$$\int_0^{\alpha} \frac{1}{2^{(n+1)N_TN_R}\Gamma((n+1)N_TN_R)} x^{(n+1)N_TN_R-1} e^{-\frac{x}{2}} dx = P_0 = 1 - \varepsilon \tag{9}$$

Here, ε is very small, such as 0.1, 0.01, etc. α can be obtained from (9), which is equivalent to $\left(\chi^2_{2(n+1)N_TN_R}(P_0)\right)^{-1}$. In order to find ML solution, all children of the root node will remain without being pruned. We adopt the pruning threshold which is expressed as

$$Th_n = \frac{n}{2}\alpha, n = 1, 2\ldots, N \tag{10}$$

Where α can be obtained from (9), and n is an empirical value related to the tree level which is set to increase the threshold value [16].

5 Complexity and Performance Analysis

In order to verify the effectiveness of PSMCTS, we analyze the complexity and the bit error ratio (BER) performance in this section, where the noise is a Gaussian white noise with a mean 0 and a variance of 1, the channel is quasi-static frequency flat fading channel and it remains constant within an observation window.

5.1 Complexity Analysis

In MCTS algorithm, M must has a minimum bound to ensure the MCTS algorithm can be achieved. According to the proof about M minimum bound in [12], set M value as $(N-1)(L-1)+1$ in multi-symbol differential system, where L is the number of constellation. Here, window length $N=4$, with QPSK modulation $L=4$, and $M \geq (N-1)(L-1)+1$. For ML algorithm, the visited node number is $L^0 + \ldots + L^{N-1} = (L^N - 1)/(L-1)$. The visited node number of MCTS and PSMCTS are shown in Figs. 4 and 5 respectively.

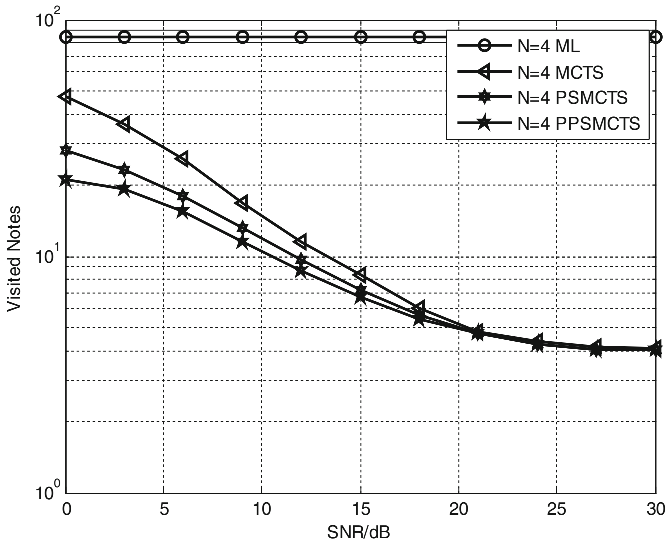

Fig. 4. Complexity analysis comparison chart

In Fig. 4, all kinds of comparisons are discussed in the condition that the window length is 4. The horizontal and vertical coordinates denotes signal-to-noise ratio and visited nodes number respectively. The result shows that under the same window length and storage space, for the same kind of modulation, PSMCTS shows more advantages compared with MCTS, especially in low SNR, which is conductive to a lower average complexity. Furthermore, based on pruning algorithm, PPSMCTS has the lowest calculate complexity among the four mentioned algorithms.

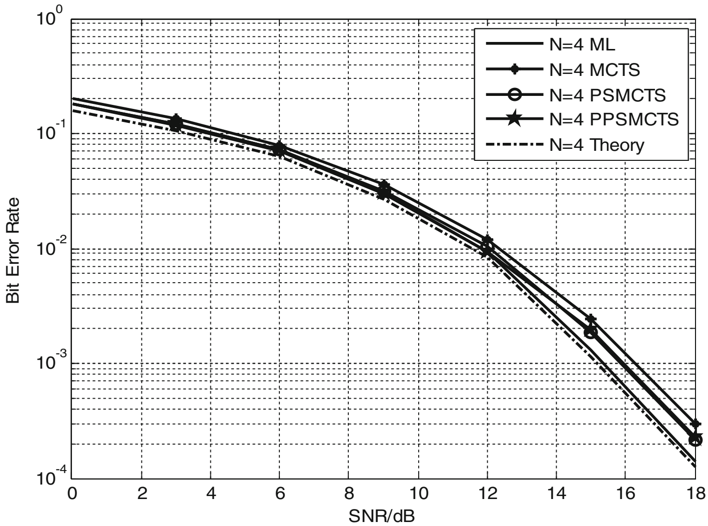

Fig. 5. Performance analysis diagram

5.2 Performance Analysis

This part mainly conducts performance simulation and analysis on algorithms in the test environment. In Alamouti STBC system, the binary bits is modulated to signal set S = {exp[2πj(i − 1)/M]} after MPSK mapping, where i = 1, …, M. After differential coding and 2 × 2 matrix transformation, each group symbol denotes as S_t, and 2 pair symbols $\left[s_{1,t} - s_{2,t}\right]$ and $\left[-s_{2,t}^*, s_{1,t}^*\right]$ are sent through two different antennas. Assuming the MPSK signal amplitude is A, single antenna transmit power $P = A^2$. Total transmit power is $P_T = A^2$. If Rayleigh channel h1 and h2 keep unchanged in the two symbol times, the received signals are:

$$\begin{aligned} r_1 &= h_1 s_1 + h_2 s_2 + n_1 \\ r_2 &= -h_1 s_2^* + h_2 s_1^* + n_2 \end{aligned} \tag{11}$$

where n_1, n_2 is AWGN channel with zero mean, and r_1, r_2 is received signals at two time slots.

$$s_1' = h_1^* r_1 + h_2 r_2^* = \alpha s_1 + h_1^* n_1 + h_2 n_2^* \tag{12}$$

$$s_2' = h_2^* r_1 - h_1 r_2^* = \alpha s_2 + h_2^* n_1 - h_1 n_2^* \tag{13}$$

where $\alpha = |h_1|^2 + |h_2|^2$. Refer to MPSK theoretical bit error rate formula in Ref. [17], we have

$$SER = \frac{M-1}{M} - \left(\frac{2a(2a^2+3)\tan^{-1}\left(\sqrt{\frac{\bar{\gamma}_s - 2a^2}{2a^2+2}}\right)}{4\pi(a^2+1)^{3/2}} + \frac{a\,\sin\left(2\,\tan^{-1}\left(\sqrt{\frac{\bar{\gamma}_s - 2a^2}{2a^2+2}}\right)\right)}{4\pi(a^2+1)^{3/2}} + \frac{a(2a^2+3)}{4(a^2+1)^{3/2}} \right) \tag{14}$$

where $a = \sqrt{\bar{\gamma}_s/2}\sin(\pi/M), \bar{\gamma}_s = 2A^2/N_0$ is average SNR.

The simulation results are shown in Fig. 5. Using theoretical BER for comparison, ML algorithm which has best performance is most close to theoretical BER. Under the same storage space, both PSMCTS and PPSMCTS have certain performance improvement compared with MCTS and are more approximate to ML, this is due to the use of probabilistic sorting algorithm effectively.

6 Summary

Considering the large computational complexity problem under the condition of the hardware storage space constraints and the small storage space, this paper proposed PSMCTS algorithm, which effectively provides better performance by using the advantage of DSPS algorithm and MCTS algorithm. Overall, PSMCTS algorithm not only has low storage space demand and easy hardware implementation, but also reduces the computational complexity in the low SNR region, which reduces the average system complexity. With using pruning algorithm, PPSMCTS has more obvious advantage in reducing complexity. At the same time, the detection performance of the two algorithms this paper proposed approach ML algorithm. Therefore, PSMCTS and PPSMCTS both are good detection algorithms.

Acknowledgement. This work was supported by Zhejiang Provincial Natural Science Foundation of China (no. LY17F010012), the Natural Science Foundation of China (no. 61571108), the open Foundation of State key Laboratory Of Networking and Switching Technology (Beijing University of Posts and Telecommunication no. SKLNST-2016-2-14).

Authors' Contributions. Xiaoping Jin conceived the idea of the system model and designed the proposed schemes. Zheng Guo has done a part of basic work in this article. Ning Jin performed simulations of the proposed schemes. Zhengquan Li provided substantial comments on the work and supported and supervised the research. All of the authors participated in the project, and they read and approved the final manuscript.

Competing Interests. The authors declare that they have no competing interests.

Appendix A

On the basis of the signal model given in Sect. 2, we define an additional $2(N+1) \times 2(N+1)$ information matrix as $\mathbf{S} = diag\{S_k, S_{k-1}, \ldots, S_{k-N}\}$. Within one observation window, the received matrix R conditioned on the message matrix S has a multivariate Gaussian conditional Probability Density Function (PDF)

$$p(\mathbf{R}|\mathbf{S}) = \frac{1}{\pi^{4(N+1)}\det\Lambda}\exp\{-tr(\mathbf{R}^H\Lambda^{-1}\mathbf{R})\} \tag{A.1}$$

where $\Lambda = \mathbf{S}(\mathbf{C}_R \otimes \mathbf{I}_{N_T})\mathbf{S}^H$. Here, $\mathbf{C}_R = \sigma_n^2\mathbf{I}_{N+1} + \mathbf{C}_h$ is the covariance matrix of R [18], $\otimes$ denotes the Kronecker product of two matrices or vectors and $\mathbf{C}_h$ denotes the autocorrelation matrix of the channel which can be expressed as

$$\mathbf{C}_h = \begin{bmatrix} C_h(0) & \cdots & C_h(N) \\ \vdots & \ddots & \vdots \\ C_h(-N) & \cdots & C_h(0) \end{bmatrix}.$$

Thus, the ML decision metric within the observation window can be written as

$$S_{ML} = \arg\min\left\{tr(\mathbf{R}^H\Lambda^{-1}\mathbf{R}) + \ln\det(\Lambda)\right\} \tag{A.2}$$

Considering that $\det(\Lambda)$ can be ignored because it is independence with the transmitted information, (A.2) becomes

$$S_{ML} = \arg\min\left\{tr(\mathbf{R}^H\Lambda^{-1}\mathbf{R})\right\} \tag{A.3}$$

Using the results of the literature [19], (A.3) can be simplified to (A.4).

$$\begin{aligned}\hat{V}_{ML} &= \mathop{\arg\min}_{V_{t+1},\ldots,V_{t+N}} \sum_{i=1}^{N}\sum_{l=i+1}^{N+1} -\tilde{c}_{i,l}||R[i+t-1](\prod_{m=i+t}^{l+t-1} V[m])^H \times R[l+t-1]||_F^2 \\ &= \mathop{\arg\min}_{V_{t+1},\ldots,V_{t+N}} \sum_{i=1}^{N}\sum_{l=i+1}^{N+1} ||R[l+t-1] - \tilde{c}_{i,l}(\prod_{m=i+t}^{l+t-1} V[m]) \times R[i+t-1]||_F^2\end{aligned} \tag{A.4}$$

In formula (A.4), $c_{i,l}$ is the entity element of Λ [15]. Normalize $c_{i,l}$ as follows, $c_m = \max|c_{k,k+1}|, k = 1,\ldots,N$ or $c_m = c_{\lfloor N/2\rfloor,\lfloor N/2\rfloor+1}$, $\tilde{c}_{i,l} = c_{i,l}/c_m$, where $\lfloor\cdot\rfloor$ denotes the floor operation, $|\bullet|$ denotes the absolute value. When the channel condition remains within an observation window, $C_h(n) = 1$. Therefore $\tilde{c}_{i,l} = 1(i = 1,2,\ldots,N, l = 2,\ldots,N+1 \text{ and } i \neq l)$. So (A.4) can be simplified to (A.5).

$$\hat{V}_{ML} = \mathop{\arg\min}_{V_{t+1},\ldots,V_{t+N}} \sum_{i=1}^{N}\sum_{l=i+1}^{N+1} ||R[l+t-1] - (\prod_{m=i+t}^{l+t-1} V[m]) \times R[i+t-1]||_F^2 \tag{A.5}$$

When N = 1, (A.5) can be simplified to (A.6)

$$\hat{V} = \mathop{\arg\min}_{V_{t+1},\ldots,V_{t+N}} ||R[t+1] - V[t+1] \times R[t]||_F^2 \tag{A.6}$$

Appendix B

When observation window N = 1, from formula (A.6), we obtain

$$\begin{aligned} D &= ||R[t+1] - V[t+1]R[t]||_F^2 \\ &= ||C[t+1]H[t+1] + W[t+1] - V[t+1](C[k]H[t] + W[t])||_F^2 \\ &= ||C[t+1]H[t+1] + W[t+1] - C[k+1]H[t] - V[t+1]W[t]||_F^2 \end{aligned} \tag{B.1}$$

Since it is assumed that the channel remains unchanged at an adjacent interval, i.e. $H[t+1] = H[t]$, so

$$D = ||W[t+1] - V[t+1]W[t]||_F^2 \tag{B.2}$$

In this paper, the $W[n], n = t, t+1, \ldots, t+N$ is a matrix with NT rows and NR columns, each element follows Gauss distribution with 0 mean and variance σ_W^2. It can be seen that $D/2\sigma_w^2$ is a chi-square random variable with a degree of freedom of $N_R N_T$. Thus, from formula (A.5), it can be deduced to (B.3) and (B.4) when the length of the observation window is N + 1 in the multi-symbol differential detection system.

$$\begin{aligned} D = {} & ||C[t+N]H[t+N] + W[t+N] - V[t+N]C[t+N-1]H[t+N-1] - V[t+N]W[t+N-1]||_F^2 \\ & + \ldots + ||C[t+N]H[t+N] + W[t+N] - V[t+N-1]V[t+N]C[t+N-2]H[t+N-2] \\ & - V[t+N-1]V[t+N]W[t+N-2]||_F^2 + \ldots + ||C[t+N]H[t+N] + W[t+N] \\ & - V[t+1]\ldots V[t+N-1]V[t+N]C[t]H[t] - V[t+1]\ldots V[t+N-1]V[t+N]W[t]||_F^2 \\ = {} & ||C[t+N]H[t+N] + W[t+N] - C[t+N]H[t+N-1] - V[t+N]W[t+N-1]||_F^2 \\ & + \ldots + ||C[t+N]H[t+N] + W[t+N] - C[t+N-1]H[t+N-2] - V[t+N-1]V[t+N]W[t+N-2]||_F^2 \\ & + \ldots + ||C[t+N]H[t+N] + W[t+N] - C[t+1]H[t] - V[t+1]\ldots V[t+N-1]V[t+N]W[t]||_F^2 \\ = {} & ||W[t+N] - V[t+N]W[t+N-1]||_F^2 + \ldots + ||W[t+N] - V[t+N-1]V[t+N]W[t+N-2]||_F^2 \\ & + \ldots + ||W[t+N] - V[t+1]\ldots V[t+N-1]V[t+N]W[t]||_F^2 \end{aligned} \tag{B.3}$$

In the derivation of (B.3), the third equal sign assumes that the channel remains constant within an observation interval, resulting in the formula (B.4)

$$D = \sum_{i=1}^{N} \sum_{l=i+1}^{N+1} ||W[l+t-1] - (\prod_{m=i+t}^{l+t-1} V[m]) \times W[i+t-1]||_F^2 \tag{B.4}$$

At this point, according to the chi-square random variable degrees of freedom of the nature of the cumulative, $D/2\sigma_w^2$ is a chi-square random variable with a degree of freedom of $N(N+1)N_R N_T$. So, the decision metrics distributed according to the chi-square distribution with $k = 2N(N+1)N_R N_T \sigma_w^2$ degrees of freedom [13]. Its cumulative distribution function (cdf) is given by

$$F(D;k) = \frac{\gamma(k/2, D/\sigma^2)}{\Gamma(k/2)} \tag{B.5}$$

where σ^2 is variance of $W[l+t-1] - (\prod_{m=i+t}^{l+t-1} V[m]) \times W[i+t-1]$ in formula (B.4). According to formulas (2) and (3), and the distribution character of channel and noise, σ^2 is equal to $2\sigma_W^2$. Both $\gamma(.)$ and $\Gamma(.)$ are Gamma functions and show as

$$\gamma(s,x) = \int_0^x t^{s-1} e^{-t} dt \tag{B.6}$$

$$\Gamma(x) = \int_0^{+\infty} t^{x-1} e^{-1} dt \tag{B.7}$$

References

1. Wei, R.Y.: Differential encoding by a look-up table for quadrature-amplitude modulation. IEEE Trans. Commun. **59**(1), 84–94 (2011)
2. Kim, J.-S., Moon, S.-H., Lee, I.: A new reduced complexity ML detection scheme for MIMO systems. IEEE Trans. Commun. **58**(4), 1302–1310 (2010)
3. Bello, I.A., Halak, B., El-Hajjar, M., Zwolinski, M.: A survey of VLSI implementations of tree search algorithms for MIMO detection. Circ. Syst. Signal Process. **35**(10), 3644–3674 (2016)
4. Schenk, A., Fischer, R.F.H.: A stopping radius for the sphere decoder: complexity reduction in multiple-symbol differential detection. In: International ITG Conference on Source and Channel Coding, pp. 1–6. IEEE (2010)
5. Takahashi, T., Fukuda, T., Sun, C.: An appropriate radius for reduced-complexity sphere decoding. In: International Conference on Communications, Circuits and Systems (ICCCAS), 28–30 July 2010, Chengdu, China, pp. 41–44 (2010)
6. Jin, N., Jin, X.P., Ying, Y.G., Wang, S., Lou, X.Z.: Research on low-complexity breadth-first detection for multiple-symbol differential unitary space-time modulation systems. IET Commun. **5**(13), 1868–1878 (2011)
7. Mao, X., Ren, S.: Adjustable reduced metric-first tree search. In: International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), 23–25 September 2011, Wuhan, China, pp. 1–4 (2011)
8. Kim, T., Park, I.: High-throughput and area efficient MIMO symbol detection based on modified Dijkstra search. IEEE Trans. Circuits Syst. I Regul. Pap. **57**(7), 1756–1766 (2010)
9. Jasika, N., Alispahic, N., Elma, A.: Dijkstra's shortest path algorithm serial and parallel execution performance analysis. In: MIPRO 2012 Proceedings of the 35th International Convention, 21–25 May 2012, Opatija, pp. 1811–1815 (2012)
10. Suh, S., Barry, J.R.: Reduced-complexity MIMO detection via a slicing breadth-first tree search. IEEE Trans. Wirel. Commun. **16**(3), 1782–1790 (2017)
11. Sah, A.K., Chaturvedi, A.K.: Stopping rule-based iterative tree search for low-complexity detection in MIMO systems. IEEE Trans. Wirel. Commun. **16**(1), 169–179 (2017)

12. Dai, Y., Yan, Z.: Memery constrained tree search detection and new ordering schemes. IEEE J. Sel. Top. Signal Process. **3**(6), 1026–1037 (2009)
13. Chang, R.Y., Chung, W.-H.: Efficient tree-search MIMO detection with probabilistic node ordering. In: IEEE International Conference on Communications, 5–9 June, 2011, Kyoto, pp. 1–5 (2011)
14. Chang, R.Y., Chung, W.-H.: Best-first tree search with probabilistic node ordering for MIMO detection: generalization and performance-complexity tradeoff. IEEE Trans. Wirel. Commun. **11**(2), 780–789 (2012)
15. Cui, T., Tellambura, C.: Bound-intersection detection for multiple-symbol differential unitary space–time modulation. IEEE Trans. Commun. **53**(12), 2114–2123 (2005)
16. Li, Y., Wei, J.B.: Multiple symbol differential detection algorithm based on the sphere decoding in unitary space time modulation system. Sci. China Ser. F-Inf. Sci. **39**(5), 569–578 (2009)
17. Hu, X., Gao, Y., Pan, Y.: Error rates calculation and performance analysis of (2,1) STBC systems. In: 7th International Conference on Signal Processing Proceedings ICSP, 31 August–4 September 2004, Beijing, pp. 1902–1905 (2004)
18. Cui, T., Tellambura, C.: On multiple symbol detection for diagonal DUSTM over ricean channels. IEEE Trans. Wirel. Commun. **7**(4), 1146–1151 (2008)
19. Bhukania, B., Schniter, P.: On the robustness of decision-feedback detection of DPSK and differential unitary space-time modulation in Rayleigh-fading channels. IEEE Trans. Wirel. Commun. **3**(5), 1481–1489 (2004)