Proceedings of APCC2008 copyright (c) 2008 IEICE 08 SB 0083

A Dynamic Layer Ordering M-paths MIMO Detection Algorithm and its Implementation

Jianping Chen, En Zhou, Xiaolin Hou, Zhan Zhang and Hidetoshi Kayama
{j.chen,zhou,hou,zhan and kayama}@docomolabs-beijing.com.cn

***Abstract*—During the past decade, Multiple-Input-Multiple-Output (MIMO) systems have received a significant attention because of its promising capacity increase over Single-Input-Single-Output (SISO) systems. In this paper, as a key technology, we present a novel M-paths MIMO detection algorithm with dynamic-layer-ordering for spatial-multiplexing MIMO systems over rich scattering wireless environment. This algorithm obtains a near-ML detection performance (e.g., only 0.2 dB performance degradation for 4-TX and 4-RX MIMO with 16QAM) with a much lower computational complexity compared to the ML detection. We also develop an recursive procedure for dynamic-layer-ordering and reduce the ordering complexity from $\mathcal{O}(N_{\mathrm{T}}^4)$ to $\mathcal{O}(N_{\mathrm{T}}^3)$, where N_{T} is the number of TX antennas. In order to verify the feasibility and performance, a pipelined structure is designed and implemented in FPGA with limited hardware resources and a 200Mbps throughput.**

I. INTRODUCTION

During the past decade, multiple-input-multiple-output (MIMO) technology over rich scattering wireless channel has received a significant attention due to the promising capacity increase of multiple-input-multiple-out (MIMO) channel over single-input-single-output (SISO) channel[2], [1]. MIMO technique offers the potential of large data transmission rate and reliability without any cost of frequency spectrum[1]. It has been one of the hot technologies for fourth-generation (4G) communications and is becoming a key part in almost every new wireless standards, such as HSDPA,802.11n, 802.16e and 802.10.

The reliability advantage of MIMO systems comes from the inherently available diversity, however, can not be exploited well by the various existing detection methods. The maximum-likelihood (ML) detection can exploit all the available diversity information, but it also tends to prohibitively high computational complexity. On the other hand, a variety of existing *sub-optimal* detection methods try to balance the computational complexity and performance, such as the equalization based technique (like ZF detection and MMSE detection), the successive interference cancellation(SIC) based techniques (like SQRD). These detection methods have much less computational complexity than ML detection, but their performance is also significantly inferior than that of ML detection[4], [5].

This paper provides a Dynamic-layer-Ordering Multiple-paths (DOM) algorithm to balance the computational complexity and detection performance. This algorithm absorbs and improves the merits of Dynamic Nulling-and-Cancelling (DNC)[4], [5] and multiple-paths detection[6] technologies and avoids their weak points. With this detection algorithm, systems can obtain only 0.2dB performance degradation than ML detection for 4×4 MIMO setting with 16QAM.

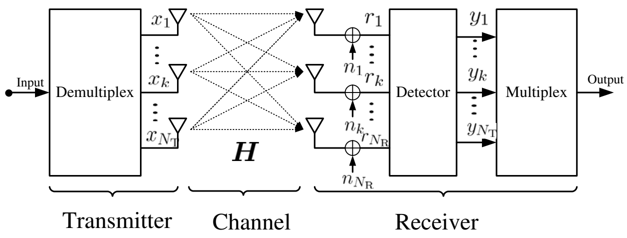


Figure 1. Spatial Multiplexing $N_{\mathrm{T}} \times N_{\mathrm{R}}$ MIMO System Model

II. THE DYNAMIC-LAYER-ORDERING M-PATHS (DOM) DETECTION ALGORITHM

In this section, we will introduce the principle of the Dynamic-layer-Ordering M-paths (DOM) algorithm first. And based on this algorithm, we will extend it to a MMSE based algorithm and a real-valued based algorithm.

A. Principle

In this paper, we assume a linear spatial-multiplexing MIMO System model (fig.1) where the transmitted vector with $\boldsymbol{x} = (x_1, \cdots, x_{N_{\mathrm{T}}})^T$ with N_{T} transmission antennas and unit transmission power $E\left\{\boldsymbol{x}\boldsymbol{x}^\dagger\right\} = \boldsymbol{I}$, the received vector $\boldsymbol{r} = (r_1, \cdots, r_{N_{\mathrm{R}}})^T$ with N_{R} antennas ($N_{\mathrm{R}} \geq N_{\mathrm{T}}$) are related according to:

$$\boldsymbol{r} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} \tag{1}$$

where $\boldsymbol{n} = (n_1, \cdots, n_{N_{\mathrm{R}}})^T$ is *Additive White Gaussian Noise* (AWGN) vector with $E\left\{\boldsymbol{n}\boldsymbol{n}^\dagger\right\} = \sigma_n^2\boldsymbol{I}$; $\boldsymbol{H}$ is a $N_{\mathrm{R}} \times N_{\mathrm{T}}$ channel matrix, therein, each single element of $\boldsymbol{H}$ is satisfying complex Gaussian distribution.

The ML detection target is to find the vector has the *maximal* probability:

$$\hat{\boldsymbol{x}} \triangleq \arg\max_{\boldsymbol{x} \in \mathcal{C}^{N_{\mathrm{T}}}} P\left(\boldsymbol{r}|\boldsymbol{x}\right) \tag{2}$$

where $\mathcal{C}$ is the symbol alphabet of each antenna.

When the channel matrix $\boldsymbol{H}$ is full rank, the ML criterion is equivalent to:

$$\hat{\boldsymbol{x}} \triangleq \arg\max_{\boldsymbol{x} \in \mathcal{C}^{N_{\mathrm{T}}}} P\left(\boldsymbol{y}|\boldsymbol{x}\right) \tag{3}$$

where $\boldsymbol{y} = (\boldsymbol{H}^\dagger\boldsymbol{H})^{-1}\boldsymbol{H}^\dagger\boldsymbol{r}$ is the *Zero-Forcing (ZF)* equalization result of the received vector $\boldsymbol{r}$. For most cases, the channel matrix $\boldsymbol{H}$ is full rank, the criterion decided by formula (3) is approaching to the ML criterion decided by formula (2).

For an antenna set $\mathcal{A}_l = \{a_1, \cdots, a_l\}$, the optimal vector $\hat{\boldsymbol{x}}_{[\mathcal{A}_l]} = (\boldsymbol{x}_{[a_1]} \cdots \boldsymbol{x}_{[a_l]})^T$ with l entries is given by the *maximum likelihood* (ML) rule that maximizes the conditional probability of $\boldsymbol{y}_{[\mathcal{A}_l]} = (\boldsymbol{y}_{[a_1]} \cdots \boldsymbol{y}_{[a_l]})^T$. Therein, a vector subscripted by a index a_i means the a_i'th element of this vector; and a vector $(\cdot)$ subscripted by a set $\mathcal{A}_l$ declares a new vector $(\cdot)_{[\mathcal{A}_l]} = ((\cdot)_{[a_1]}, \cdots, (\cdot)_{[a_l]})^T$:

$$\hat{\boldsymbol{x}}_{[\mathcal{A}_l]} \triangleq \underset{\boldsymbol{x}_{[\mathcal{A}_l]} \in \mathcal{C}^{\mathcal{A}_l}}{\arg\max}\, P\left(\boldsymbol{y}_{[\mathcal{A}_l]} | \boldsymbol{x}_{[\mathcal{A}_l]}\right) \tag{4}$$

where the set $\mathcal{C}^{\mathcal{A}_l}$, $|\mathcal{C}^{\mathcal{A}_l}| = |\mathcal{C}|^l$, is the search space composed by antenna set $\mathcal{A}_l$. Obviously, when $l = N_\mathrm{T}$, $\hat{\boldsymbol{x}}_{[\mathcal{A}_{N_\mathrm{T}}]}$ is identical to the ML detection denoted by formula(3).

The error propagation of Successive Interference Cancellation (SIC) causes detection performance degradation significantly, the *most reliable* antennas shall be canceled first [6].

For layer l, the *most reliable* antenna set $\mathcal{A}_l$ also can be expressed by the *maximum likelihood* (ML) rule:

$$\mathcal{A}_l \triangleq \underset{\mathcal{A} = \{a'_1 \cdots a'_l\} \subseteq \{1 \cdots N_\mathrm{T}\}}{\arg\max} P\left(\boldsymbol{y}_{[\mathcal{A}]} | \hat{\boldsymbol{x}}_{[\mathcal{A}]}\right) \tag{5}$$

where $\hat{\boldsymbol{x}}_{[\mathcal{A}]}$ is defined in formula(4).

Evidently, to search $\hat{\boldsymbol{x}}_{[\mathcal{A}]}$ directly with exponential computational complexity should be avoided. For layer l, assume that we have gotten an *most reliable* antennas set $\mathcal{A}_l = \{a_1 \cdots a_l\}$ based on criterion(5) and M vectors $\mathcal{X}_l = \{\boldsymbol{x}_m^{(l)} | 1 \le m \le M\}$. Therein, the i'th element $\boldsymbol{x}_{m,[i]}^{(l)}$ of a vector $\boldsymbol{x}_m^{(l)}$ corresponds to the transmitted symbol of the a_i'th antenna; and these vectors have relationship as $P(\boldsymbol{y}_{[\mathcal{A}_l]} | \boldsymbol{x}_1^{(l)}) \ge \cdots \ge P(\boldsymbol{y}_{[\mathcal{A}_l]} | \boldsymbol{x}_M^{(l)}) \ge P(\boldsymbol{y}_{[\mathcal{A}_l]} | \text{any other vector})$, and we call them *survival-paths* in the view of the *tree search* algorithm.

Based on the SIC principle, replace the search $\mathcal{C}^{\mathcal{A}_l}$ with these M survival-paths, the *most reliable* antenna of layer $l+1$ is given by:

$$a_{l+1} \triangleq \underset{a'_{l+1} \in \{1 \cdots N_\mathrm{T}\} \setminus \mathcal{A}_l}{\arg\max} P\left(\boldsymbol{y}_{[\mathcal{A}_l \cup \{a'_{l+1}\}]} | \hat{\boldsymbol{x}}^{(l+1)}\right) \tag{6}$$

where $\hat{\boldsymbol{x}}^{(l+1)} = (\hat{\boldsymbol{x}}_{[a_1]}, \cdots, \hat{\boldsymbol{x}}_{[a_l]}, \hat{\boldsymbol{x}}_{[a']})^T$ is a path (vector) with $l+1$ elements and maximizes the likelihood:

$$\hat{\boldsymbol{x}}^{(l+1)} \triangleq \underset{\substack{\boldsymbol{x}'_{[1:l]} \in \mathcal{X}_l \\ \boldsymbol{x}'_{[l+1]} \in \mathcal{C}}}{\arg\max} P\left(\boldsymbol{y}_{[\mathcal{A}_l]}, \boldsymbol{y}_{[a'_{l+1}]} | \boldsymbol{x}'\right) \tag{7}$$

where a vector $(\cdot)$ subscripted by a range $[1:l]$ declares a new vector $\left((\cdot)_{[1]} \cdots (\cdot)_{[l]}\right)^T$.

In order to control the computational complexity of dynamic layer further more, we limit M to 1:

$$a_{l+1} \triangleq \underset{a'_{l+1} \in \{1 \cdots N_\mathrm{T}\} \setminus \mathcal{A}_l}{\arg\max} P\left(\boldsymbol{y}_{[\mathcal{A}_l]}, \boldsymbol{y}_{[a'_{l+1}]} | \boldsymbol{x}_1^{(l)}, \hat{x}\right) \tag{8}$$

where $\hat{x} \triangleq \arg\max_{\hat{x} \in \mathcal{C}} P(\boldsymbol{y}_{[\mathcal{A}_l \cup \{a'_{l+1}\}]} | \boldsymbol{x}_1^{(l)}, \hat{x})$ is the symbol which has the *maximum* likelihood. In this way, the *most reliable* antennas set for layer $l+1$ is given by $\mathcal{A}_{l+1} = \mathcal{A}_l \cup \{a_{l+1}\}$.

Similarly, after deciding the *most reliable* antennas set $\mathcal{A}_{l+1}$ for layer $l+1$, the M survival-paths can be updated by:

$$\mathcal{X}_{l+1} \triangleq \underset{\substack{\boldsymbol{x}_{[1:l]} \in \mathcal{X}_l \\ \boldsymbol{x}_{[l+1]} \in \mathcal{C}}}{\arg \overset{[M]}{\mathrm{SCG}}} P\left(\boldsymbol{y}_{[\mathcal{A}_l]}, \boldsymbol{y}_{[a_{l+1}]} | \boldsymbol{x}\right) \tag{9}$$

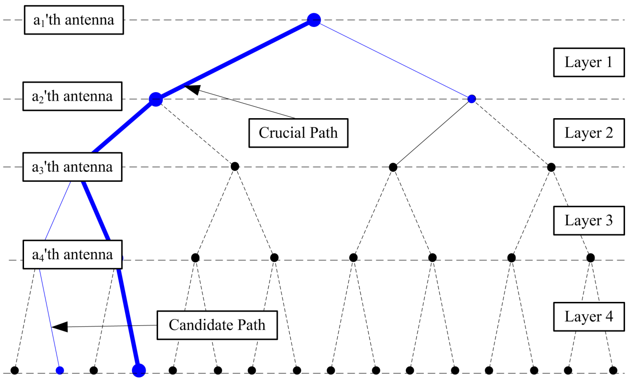


Figure 2. An Illustration of the DOM algorithm with M=2 for 4x4 MIMO with BPSK

where operator $\mathrm{SCG}^{[M]}$ means **S**ort and **C**hoose M **G**reatest values to compose a new set with M elements.

Recursively, l from 1 to N_T, we first decide the most reliable antenna set $\mathcal{A}_l$ with the aid of survival-paths $\mathcal{X}_{l-1}$ of layer $l-1$; then, based on $\mathcal{A}_l$ and $\mathcal{X}_{l-1}$, update the new survival-paths $\mathcal{X}_l$ for layer l. In this way, until the $\mathcal{X}_{N_\mathrm{T}}$ is decided and select the first vector $\boldsymbol{x}_1^{(N_\mathrm{T})}$ of $\mathcal{X}_{N_\mathrm{T}}$ as the final detection result (fig.2).

B. Algorithm

The DOM algorithm is an iterative algorithm: in the l'th iterative step, one data stream will be canceled and a $(N_\mathrm{T} - l + 1) \times N_\mathrm{R}$ MIMO will be reduced to a $(N_\mathrm{T} - l) \times N_\mathrm{R}$ MIMO; recursively, until all the data stream is canceled.

1) Initialization Step: Perform the *ZF* equalization as follows:

$$\boldsymbol{y}^{(1)} = \boldsymbol{D}^{(1)} \boldsymbol{H}^\dagger \boldsymbol{r} \tag{10}$$

where

$$\boldsymbol{D}^{(1)} = (\boldsymbol{H}^\dagger \boldsymbol{H})^{-1} \tag{11}$$

The overall computational complexity of initialization step is $\mathcal{O}(N_\mathrm{T}^3)$ because of the complexity of matrix inversions and multiplications.

2) Iterations Step (iteration variable l from 1 to N_T):

- Precalculation:
  Calculate the Post-Signal-to-Noise Rate (PSNR) of each active antenna $a \in \{1, \cdots, N_\mathrm{T}\} \setminus \{a_1, \cdots, a_{l-1}\}$, (the active antennas are defined as the antennas which have not been canceled), where a_i, $1 \le i \le l-1$, is the antenna index which has been canceled in layer l:

$$\mathrm{PSNR}_a = \frac{1}{\sigma_n^2 \cdot \boldsymbol{D}_{[k,k]}^{(l)}} \propto \frac{1}{\boldsymbol{D}_{[k,k]}^{(l)}} \tag{12}$$

  where $\boldsymbol{D}_{[k,k]}^{(l)}$ is the k'th (corresponds to the a'th antenna) diagonal element of $(N_\mathrm{T}-l+1) \times (N_\mathrm{T}-l+1)$ matrix $\boldsymbol{D}^{(l)}$ defined in formula (11); it maps to the a'th antenna.
  At the first blush, the computational complexity of calculating $\boldsymbol{D}^{(l)}$ is approaching to $\mathcal{O}(N_\mathrm{T}^3)$. With a recursive algorithm introduced by [6] (see section-II-C), the computational complexity of calculating $\boldsymbol{D}^{(l)}$ can be reduced to $\mathcal{O}\left(N_T^2\right)$.

- Dynamic Layer Ordering:
Calculate the reliability for all the active antennas $a \in \{1, \cdots, N_{\mathrm{T}}\} \setminus \{a_1, \cdots, a_{l-1}\}$ and select the most reliable one out of them as the most reliable antenna:

$$a_l = \underset{a \in \{1 \cdots N_{\mathrm{T}}\} \setminus \{a_1 \cdots a_{l-1}\}}{\arg\max} \mathrm{PSNR}_a \cdot \mathrm{I}_a \tag{13}$$

where $\mathrm{I}_a = \min_{\boldsymbol{x}_{[a]} \in \mathcal{C} \setminus \{\hat{\boldsymbol{x}}_{[a]}\}} \|\boldsymbol{y}_{1,[a]}^{(l)} - \boldsymbol{x}_{[a]}\|^2 - \|\boldsymbol{y}_{1,[a]}^{(l)} - \hat{\boldsymbol{x}}_{[a]}\|^2$ is the *instantaneous reliability factor* (IRF)[4]. It is defined as the difference between the minimal Euclidian distance and the second minimal Euclidian distance of the ZF equalization result $\boldsymbol{y}_{1,[a]}$ in constellation $\mathcal{C}$.
- M survival-paths detection:
For all M survival paths $\boldsymbol{x}_1^{(l-1)}, \cdots, \boldsymbol{x}_M^{(l-1)}$, assume $\Lambda_1^{(l-1)}, \cdots, \Lambda_M^{(l-1)}$ and $\boldsymbol{y}_1^{(l)}, \cdots, \boldsymbol{y}_M^{(l)}$ are their corresponding accumulated metrics of layer $l-1$ and interference-cleaned *ZF* equalization results respectively of layer l, calculate all the possible $M \times |\mathcal{C}|$ accumulated metrics and choose best M out of $M \times |\mathcal{C}|$ metrics by:

$$\left\{\Lambda_1^{(l)} \cdots \Lambda_M^{(l)}\right\} = \underset{\substack{1 \le m \le M \\ x \in \mathcal{C}}}{\overset{[M]}{\mathrm{SCL}}} \left\{\Lambda_m^{(l-1)} + \mathrm{PSNR}_{a_l} \cdot \|\boldsymbol{y}_{m,[a_l]}^{(l)} - x\|^2\right\} \tag{14}$$

where operator $\mathrm{SCL}^{[M]}$ means **S**ort and **C**hoose M **L**east values for a set. Instantaneously, record the corresponding index m and symbol x, update the new survival-paths $\boldsymbol{x}_1^{(l)}, \cdots, \boldsymbol{x}_M^{(l)}$.
- Update the corresponding M *ZF* equalization results $\boldsymbol{y}_1^{(l+1)}, \cdots, \boldsymbol{y}_M^{(l+1)}$ with $\boldsymbol{y}_1^{(l)}, \cdots, \boldsymbol{y}_M^{(l)}$ and update the matrix $\boldsymbol{D}^{(l)}$ with $\boldsymbol{D}^{(l-1)}$ recursively (subsection-II-C).

3) Output Step: Select the best (first) survival-path $\boldsymbol{x}_1^{(N_{\mathrm{T}})}$ as the final detection result.

C. Recursive Calculation

Assume that the interference-cleaned channel matrix $\boldsymbol{H}^{(l+1)}$ of layer $l+1$ is the result of the channel matrix $\boldsymbol{H}^{(l)}$ of layer l with k'th column removed, the symmetric matrix $\boldsymbol{D}^{(l+1)} = (\boldsymbol{H}^{(l+1)\dagger}\boldsymbol{H}^{(l+1)})^{-1}$ can be calculated by updating the matrix $\boldsymbol{D}^{(l)} = (\boldsymbol{H}^{(l)\dagger}\boldsymbol{H}^{(l)})^{-1}$ with $\mathcal{O}(n^2)$ computational complexity as follows:

$$\boldsymbol{D}^{(l+1)} = \begin{bmatrix} \boldsymbol{D}_{11} & \boldsymbol{D}_{12} \\ \boldsymbol{D}_{21} & \boldsymbol{D}_{22} \end{bmatrix} - \frac{1}{\delta_k} \begin{bmatrix} \boldsymbol{d}_1 \\ \boldsymbol{d}_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{d}_1^\dagger & \boldsymbol{d}_2^\dagger \end{bmatrix} \tag{15}$$

where $\boldsymbol{D}_{11}$, $\boldsymbol{D}_{12}$, $\boldsymbol{D}_{21}$, $\boldsymbol{D}_{22}$, $\boldsymbol{d}_1$, $\boldsymbol{d}_2$ and δ_k are parts of matrix $\boldsymbol{D}^{(l)}$:

$$\begin{array}{cc} & \begin{array}{ccc} & k'\text{th column} & \\ & \downarrow & \end{array} \\ \boldsymbol{D}^{(l)} = & \begin{bmatrix} \boldsymbol{D}_{11} & \boldsymbol{d}_1 & \boldsymbol{D}_{12} \\ \boldsymbol{d}_1^\dagger & \delta_k & \boldsymbol{d}_2^\dagger \\ \boldsymbol{D}_{21} & \boldsymbol{d}_2 & \boldsymbol{D}_{22} \end{bmatrix} \leftarrow k'\text{th row} \end{array}$$

Similarly, the corresponding *ZF* equalization result $\boldsymbol{y}^{(l+1)}$ be calculated based on the channel matrix $\boldsymbol{D}^{(k)}$ and the previous layer's result $\boldsymbol{y}^{(l)}$ as follows:

$$\boldsymbol{y}^{(l+1)} = \begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \end{bmatrix} - \frac{1}{\delta_k} \begin{bmatrix} \boldsymbol{d}_1 \\ \boldsymbol{d}_2 \end{bmatrix} (\varphi_k - x_k) \tag{16}$$

where x_k is the detection result of k'th element; $\boldsymbol{y}_1$, $\boldsymbol{y}_2$ and φ_k are parts of $\boldsymbol{y}^{(l)}$ as follows:

$$\boldsymbol{y}^{(l)} = \begin{bmatrix} \boldsymbol{y}_1 \\ \varphi_k \\ \boldsymbol{y}_2 \end{bmatrix} \leftarrow t'\text{th element}$$

The total computational complexity of updating *ZF* results for M survival-paths is approaching to $\mathcal{O}(N_{\mathrm{T}}^2)$.

D. MMSE Extension

In previous subsections, the deduction is based on *ZF* equalization, in deed, an algorithm based on the *ZF* equalization can always be extended to a *MMSE* based one by:

$$\begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{0} \end{bmatrix} = \begin{bmatrix} \boldsymbol{H} \\ -\sigma_n \boldsymbol{I} \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} \boldsymbol{n} \\ \sigma_n \boldsymbol{I} \end{bmatrix} \tag{17}$$

The only change is to replace formula (11) by $\boldsymbol{D}^{(1)} = (\boldsymbol{H}^\dagger \boldsymbol{H} + \sigma_n^2 \boldsymbol{I})^{-1}$. As simulation result shows that the extension will only enhance a little of computational complexity but enhance the performance significantly (section-IV).

E. Real-Valued System Model versus Complex-Valued System Model

The performance of MIMO detection is affected by the cancellation ordering significantly, the more elaborate ordering, the higher detection performance. Conventional MIMO detection algorithms are done in complex field because all the elements are complex numbers in system model (1). A complex-valued MIMO system model always can be reinterpreted as an equivalent real-valued MIMO system model [7]:

$$\begin{bmatrix} \Re\{\boldsymbol{r}\} \\ \Im\{\boldsymbol{r}\} \end{bmatrix} = \begin{bmatrix} \Re\{\boldsymbol{H}\} & -\Im\{\boldsymbol{H}\} \\ \Im\{\boldsymbol{H}\} & \Re\{\boldsymbol{H}\} \end{bmatrix} \begin{bmatrix} \Re\{\boldsymbol{x}\} \\ \Im\{\boldsymbol{x}\} \end{bmatrix} + \begin{bmatrix} \Re\{\boldsymbol{n}\} \\ \Im\{\boldsymbol{n}\} \end{bmatrix} \tag{18}$$

It is always possible for rectangle constellations, such as QAM; their symbol alphabet can be divided to two independent real symbol alphabets because of the independence of the real part and imagery part. For example, an 16QAM symbol always can be divided to two independent 4PAM symbols. This extension will enhance the detection performance significantly for DOM algorithm because of more elaborate dynamic layer ordering.

III. Hardware Structure

In practice, we have implemented a pipelined structure for the DOM algorithm and implemented and verified it in FPGAs. This is a implementation based on DOM-R-MMSE algorithm (M=4) for $N_{\mathrm{T}} = 4$ and $N_{\mathrm{R}} = 4$ MIMO with 16QAM.

Because the DOM algorithm is an iterative algorithm, we convert each iteration step into a single block which can process one received vector in 8 clock cycles. In order to avoid overflow, we also designed a floating-point data type in FPGA and applied it into crucial part who need accurate calculation, such as matrix inversion.

Each iteration block includes 3 sub-blocks:

1) Search for the most reliable antenna accords to formula (13)
2) According to formula (14), the second block is to search the survival-paths which have least accumulated-metrics.
3) Finally, update the matrix $\boldsymbol{D}^{(l)}$ and MMSE/ZF equalization result $\boldsymbol{y}_{\text{zf}}^{(l)}$ according to formula (15) and (16) respectively.

In order to avoid to estimate the noise power σ_n^2, we used a fixed value 10^{-3} to replace it:

$$\boldsymbol{D}^{(1)} = \left(\boldsymbol{H}^{\dagger}\boldsymbol{H} + 10^{-3}\boldsymbol{I}\right)^{-1}$$

this approximation simplifies the complexity of the receiver largely, but only has a little of performance degradation compared to DOM-R with *MMSE* extension.

IV. Performance Evaluation

A. Software Verifications

We evaluated the BER performance of DOM for both complex-valued/real-valued algorithms (denoted by DOM-C/DOM-R) and the *ZF/MMSE* (denoted by DOM-ZF/DOM-MMSE) based algorithms with different M. Here, we only provide the results of the 4×4 MIMO settings with 16QAM, the similar conclusions have been inspected and verified for other MIMO settings.

Fig.4 shows the BER performance versus SNR for both DOM-C/DOM-R and ZF/MMSE algorithms with $M = 4$. We find that:

1) Near-ML performance: DOM-C-ZF, DOM-C-MMSE, DOM-R-ZF and DOM-R-MMSE have near-ML performance with the increase of the number of survival-paths M. When M is set to 4, DOM-R-MMSE only has 0.2 dB performance degradation.
2) The real-valued algorithm has much better performance than the complex-valued algorithm because of the more elaborate dynamic ordering.
3) The MMSE based algorithm has better performance than ZF-based one: when $M = 4$, DOM-C-MMSE have about 2dBs gains than DOM-C-ZF because that the MMSE balanced inter-antenna interference and noise power amplification, but the ZF based one need not to estimate the noise power σ_n^2, it simplifies the complexity of the receiver.

B. Hardware Verifications

We have verified the performance of the pipeline hardware implementation of the DOM algorithm in our OFDM MIMO system. In this system, the radio frequency is set to 2.35GHz; the whole bandwidth is 6.25MHz; each frame includes 32 OFDM symbols; and the length of IFFT and the guard interval is set to 1024 and 128 respectively. The whole system is implemented in FPGAs and DSPs. The channel is emulated by the C8 channel emulator of Elektrobit Ltd.

Fig.5(a), fig.5(b) and fig.5(c) show the performances of the DOM algorithms both in hardware verifications and

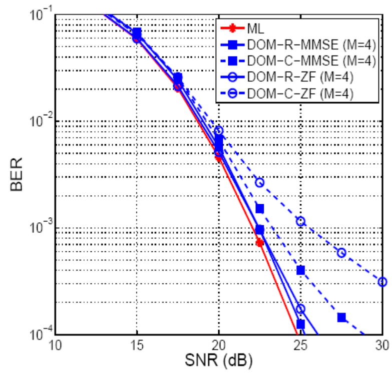


Figure 4. Performance of the DOM algorithm for $N_{\text{T}} = 4$ and $N_{\text{R}} = 4$ with 16QAM

Relative Path Power (dB)	Delay (ns)
0.0	0
-1.0	310
-9.0	710
-10.0	1090
-15.0	1730
-20.0	2510

Table I
Channel Model (3GPP TR 25.996-Case 2)

software verifications with a robust channel estimation algorithm named 2D-EDFTI algorithm[8] over th 3GPP TR 25.996-case 2 channel model(reference to tab.I). We find that the DOM algorithm has at least 10 dBs performance enhancement than MMSE detection, when M is set to 4, for 4×4 MIMO. For other channel model, similar results will also hold based on our observations.

V. Conclusions

In this paper, we present a near-ML dynamic-layer-ordering M-paths (DOM) MIMO detection method and develop a pipeline hardware structure for it. This algorithm is deduced from ML criterion, and it achieves near-ML performance through the dynamic-layer-ordering and M-paths detection technologies (about 0.2dB performance degradation with a low computational complexity when M = 4, section-IV). The dynamic layer ordering reduces the effect of error propagation; the M-paths detection improves the detection accuracy. We also extend the DOM algorithm to MMSE system model (16) and real-valued system model (17) to enhance the performance by a more elaborate and accurate ordering. A recursive parameters estimation algorithm is also introduced to reduce the computational complexity to $\mathcal{O}(N_{\text{T}}^3)$. It has good scalability to balance computational complexity and performance by adjusting the number of survival-paths M.

In practice, we have implemented and verified this algorithm in hardware (FPGA and DSP) in a pipeline structure with limited resources. The simulation results (section-IV-A) and hardware verification results (section-IV-B) show that the DOM algorithm has a near-ML performance with a low computational complexity; it is a promising reception technology for spatial multiplexing MIMO system.

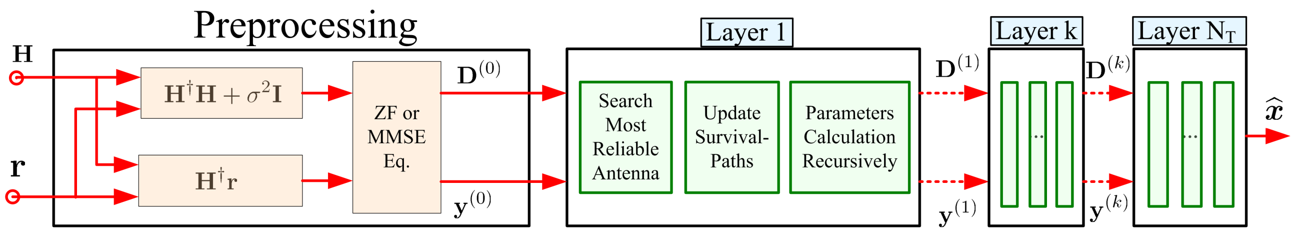
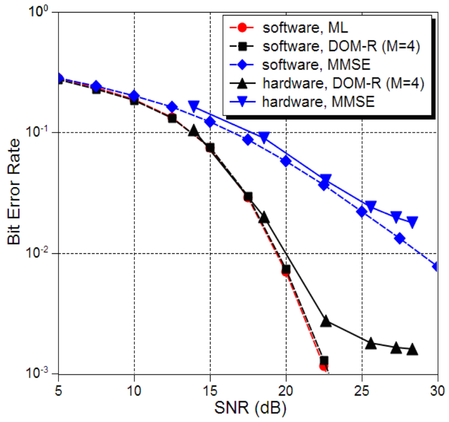
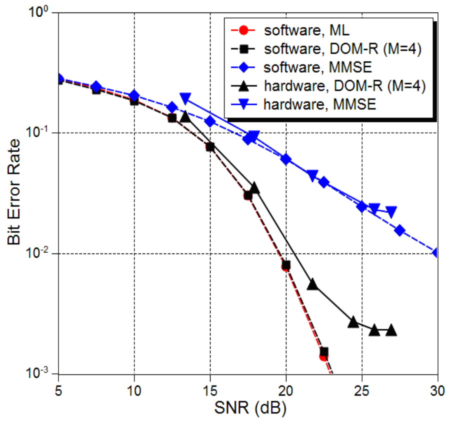


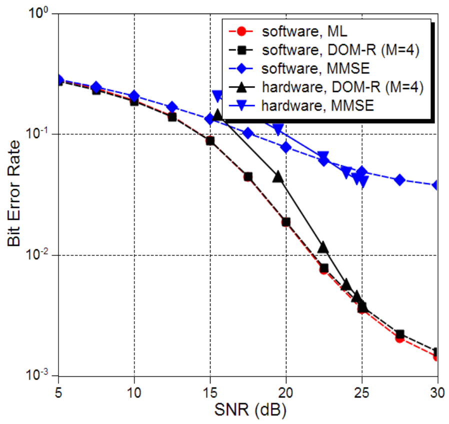
Figure 3. A Pipeline Hardeare Structure for the DOM algorithm



(a) 3km/h case



(b) 30km/h case



(c) 120km/h case

Figure 5. Performance Measurement over 3GPP TR 25.996 Channels

REFERENCES

[1] G. Foschini, "Layered space-time architecture for wireless communication in fading environments when using multiple antennas," Bell Labs. Tech. Journal, vol.2, pp.41–59, 1996.

[2] G. Golden, G. Golden, C. Foschini, R. Valenzuela, and P. Wolniansky, "Detection algorithm and initial laboratory results using V-BLAST space-time communication architecture," Electronics Letters, vol.35, no.1, pp.14–16, 1999.

[3] U. Fincke and P. M., "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," Mathematics of Computation, vol. 44, pp. 463–471, 1985.

[4] D. Seethaler, D. Seethaler, H. Artes, and F. Hlawatsch, "Dynamic nulling-and-cancelling with near-ML performance for MIMO communication systems," in Proc. IEEE ICASSP '04, H. Artes, Ed., vol. 4, pp. 777–780, 2004.

[5] D. Seethaler, D. Seethaler, H. Artes, and F. Hlawatsch, "Dynamic Nulling-and-Canceling for Efficient Near-ML Decoding of MIMO Systems," in Proc. IEEE Trans. on Signal Processing, vol.54, pp.4741–4752, 2006

[6] Y. Dai, Y. Dai, S. Sun, and Z. Lei, "A comparative study of QRD-M detection and sphere decoding for MIMO-OFDM systems," in Proc. IEEE 16th International Symposium on PIMRC 2005, S. Sun, Ed., vol. 1, pp. 186–190, 2005.

[7] R. Fischer, R. Fischer, and C. Windpassinger, "Real versus complex-valued equalisation in V-BLAST systems," Electronics Letters, vol. 39, no. 5, pp. 470–471, 2003.

[8] X. Hou, Z. Zhang and H. Kayama, "Low-complexity enhanced DFT-based channel estimation for OFDM systems with virtual subcarriers," in Proc. IEEE PIMRC'07, Sept. 3-7, 2007.