# Implementation-Friendly QRM-MLD Using Trellis-Structure Based on Viterbi Algorithm

Sang-Ho Choi, Jun Heo, and Young-Chai Ko

*Abstract:* **The maximum likelihood detection with QR decomposition and M-algorithm (QRM-MLD) has been presented as a suboptimum multiple-input multiple-output (MIMO) detection scheme which can provide almost the same performance as the optimum maximum likelihood (ML) MIMO detection scheme but with the reduced complexity. However, due to the lack of parallelism and the regularity in the decoding structure, the conventional QRM-MLD which uses the tree-structure still has very high complexity for the very large scale integration (VLSI) implementation. In this paper, we modify the tree-structure of conventional QRM-MLD into trellis-structure in order to obtain high operational parallelism and regularity and then apply the Viterbi algorithm to the QRM-MLD to ease the burden of the VLSI implementation. We show from our selected numerical examples that, by using the QRM-MLD with our proposed trellis-structure, we can reduce the complexity significantly compared to the tree-structure based QRM-MLD while the performance degradation of our proposed scheme is negligible.**

*Index Terms:* **Maximum likelihood detection (MLD), maximum likelihood detection with QR decomposition and M-algorithm (QRM-MLD), multiple input multiple output (MIMO), QR decomposition, very large scale integration (VLSI).**

## I. INTRODUCTION

The employment of multiple-input multiple-output (MIMO) systems has received much interest because it can provide an increase in data rate and diversity gain in wireless communication [1], [2]. MIMO detection, which is one of the essential part in MIMO systems, has been of major research issues since the performance of the MIMO systems depends highly on the detection scheme.

While the maximum likelihood detection (MLD) is optimum in terms of the performance [2], the MLD has prohibitive complexity to implement. In particular, the complexity of MLD is exponentially increasing as the number of antennas and/or the constellation size increase since it needs to make an exhaustive search over all the possible transmitted symbol combinations. As such, some suboptimal approaches such as zero forcing detection (ZFD), minimum mean square error detection (MMSED), and some variants of ZFD and MMSED have been presented [2], [3]. However, although the complexity of ZFD and MMSED is much reduced compared to the MLD, the performance of ZFD and MMSED is often unacceptable, in particular, for the system to require high throughput.

Recently, the MLD with QR decomposition and M-algorithm (QRM-MLD) was proposed in [4], where the performance is near the performance of MLD but with quite reduced complexity compared to the optimal MLD. The QRM-MLD basically selects a certain limited number of surviving set of paths at each MIMO detection stage according to the threshold, used for the purpose of selecting the qualified paths and reducing the number of surviving paths, instead of selecting all the possible set of paths such as in MLD. However, the conventional QRM-MLD which is using the tree-structure, still requires high complexity for the implementation mainly due to the lack of parallelism and regularity in the decoding structure which are critical factors for low-power and high speed very large scale integration (VLSI) implementation. The same problem has occurred in the MLD and the modification of tree-structure for the MLD has been presented in [5].

Note that it has been known in VLSI implementation that the delay time is approximately proportional to the inverse of power consumption, which means that the reduction of power consumption results in the longer delay time [6]. Therefore, in case of saving the power, a certain finite number of parallel functions are used, instead of the serial functions, in order to compensate for longer delay time. As such the operational frequency can be maintained. This approach is often adopted for the channel decoding to reduce the power consumption [7]. In addition to parallelism, regularity in an algorithm, which refers to the repeated occurrence of computational patterns, can reduce the power consumption by reusing the connections within a pattern [8].

In this paper, using [4] and [5], we show that the tree-structure of conventional QRM-MLD can be reconstructed into the trellis-structure in order to obtain high operational parallelism and regularity and then apply the Viterbi algorithm to the QRM-MLD to ease the burden of the VLSI implementation.[1] Furthermore, we show that, using our proposed two different thresholds in selecting the survival paths and trellis-structure based QRM-MLD algorithm, the number of surviving paths can be much reduced which results in much lowered calculation complexity while the performance degradation of our proposed scheme is negligible compared to the conventional tree-structure based QRM-MLD.

This paper is organized as follows. Section II gives a system and channel model to be considered in this paper and Section III gives brief summary of conventional QRM-MLD. Section IV introduces our proposed MIMO detection and also describes its differences from conventional QRM-MLD. Then, we sug-

Manuscript received April 26, 2007; approved for publication by Ravi Narasimhan, Division II Editor, September 23, 2007.

This paper was presented in part at the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Athens, Greece, September 2007.

This research is supported by the Ubiquitous Computing and Network (UCN) Project, Knowledge and Economy Frontier R&D Program of the Ministry of Knowledge Economy (MKE) in Korea as a result of UCN's subproject 09C1-C2-31T.

The authors are with the School of Electrical Engineering, Korea University, Seoul 136-701, Korea, email: {dosiba, junheo, koyc}@korea.ac.kr.

[1] In essence, the Viterbi algorithm make use of the parallelism and the regularity for the decoding.

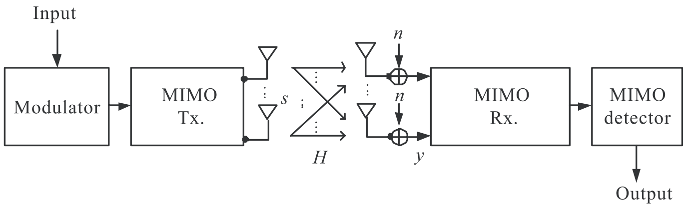



Fig. 1. Uncoded MIMO system model.

gest the scheme to use two different thresholds for selecting reliable paths to decrease complexity. The simulation results of performance and complexity of proposed MIMO detection are presented and discussed in Section V, followed by conclusion.

## II. SYSTEM AND CHANNEL MODEL

Fig. 1 depicts the MIMO system with $n_T$ transmit antennas and $n_R$ receive antennas ($n_T \leq n_R$). The system adopts spatial multiplexing signaling (i.e., the signals transmitted from multiple antennas are independent each other) and $M$-QAM modulation with $M = 2^b$, where $M$ is the number of constellations in modulation and $b$ is the number of bits per symbol. When $\mathbf{s}$ is denoted by transmitted symbol vector with the size of $n_T \times 1$, to be passed through the Rayleigh fading channel with the $n_R \times n_T$ channel matrix, $\mathbf{H}$, and $\mathbf{y}$ is denoted by received symbol vector with the size of $n_R \times 1$, we can express MIMO system as

$$\mathbf{y} = \mathbf{Hs} + \mathbf{n} \tag{1}$$

where $\mathbf{s} = [s_{n_T}\ s_{n_T-1} \cdots s_1]^T$ and $\mathbf{n}$ is a $n_R \times 1$ additive white gaussian noise (AWGN) vector with zero mean and variance $N_0/2$.

## III. CONVENTIONAL QRM-MLD

### *A. Algorithm*

In this section, we briefly summarize the QRM-MLD scheme based on [4]. Let us assume that the maximum rank of $\mathbf{H}$ is equal to $n_T$. First, taking the QR decomposition into $\mathbf{H}$, we can represent $\mathbf{H}$ as

$$\mathbf{H} = \mathbf{Q}\acute{\mathbf{R}} \tag{2}$$

where $\mathbf{Q}$ is unitary matrix with the size of $n_R \times n_R$ and $\acute{\mathbf{R}}$ is given as

$$\acute{\mathbf{R}} = \begin{bmatrix} \mathbf{R} \\ \mathbf{0}_{(n_R-n_T)\times n_T} \end{bmatrix}. \tag{3}$$

Note that in (3), $\mathbf{R}$ is an $n_T \times n_T$ upper triangular matrix and $\mathbf{0}_{(n_R-n_T)\times n_T}$ is a zero matrix of size $(n_R - n_T) \times n_T$. Multiplying $\mathbf{Q}^H$ to both sides of (1) and using $\mathbf{Q}^H\mathbf{Q} = \mathbf{I}$ yield

$$\mathbf{z} = \mathbf{Rs} + \acute{\mathbf{n}} = \mathbf{R}\begin{bmatrix} s_{n_T} \\ s_{n_T-1} \\ \vdots \\ s_1 \end{bmatrix} + \begin{bmatrix} \acute{n}_{n_T} \\ \acute{n}_{n_T-1} \\ \vdots \\ \acute{n}_1 \end{bmatrix} \tag{4}$$

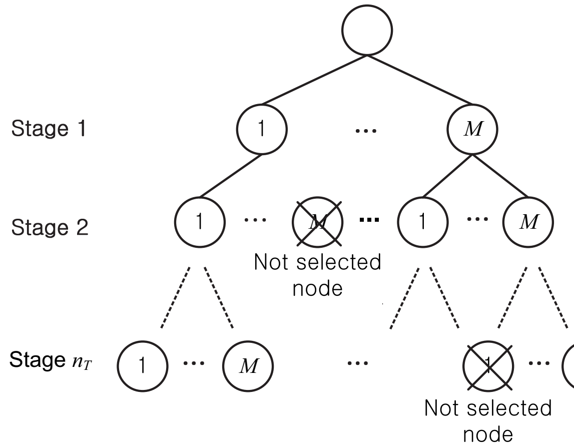



Fig. 2. Tree-structured QRM-MLD.

where $\mathbf{z} = \mathbf{Q}^H\mathbf{y} = [z_{n_T}\ z_{n_T-1} \cdots z_1]^T$, $\acute{\mathbf{n}} = \mathbf{Q}^H\mathbf{n}$, and

$$\mathbf{R} = \begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,n_T} \\ 0 & r_{2,2} & \cdots & r_{2,n_T} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & r_{n_T,n_T} \end{bmatrix}. \tag{5}$$

Next, we describe the operation of the M-algorithm [9]. The decoding starts from finding the transmitted symbol candidates for $s_1$ by comparing the squared Euclidean distance between the received symbol $z_1$ and all the possible symbols, $c_x$, for $1 \leq x \leq M$, given as

$$|z_1 - r_{n_T,n_T} c_x|^2. \tag{6}$$

By comparing the metrics given in (6) for all the possible transmitted symbols, the transmitted symbol candidates are arranged from the one with the smallest metric in increasing order. Then, the $S_1$ transmitted symbol candidates with minimum value of the metric in (6) are selected and the selected symbol candidates are transferred to the next stage along with the corresponding metric values. From the second stage, the decoding process can be generalized. In the $i$th stage, for $2 \leq i \leq n_T$, the combinations of surviving transmitted symbol candidate sets, $c_{x'}$ $(1 \leq x' \leq S_{i-1})$, from the first to the ($i$-1)th symbols and all the possible transmitted symbols, $c_x$ $(1 \leq x \leq M)$, for the current $i$th symbol, $s_i$, are generated. Then, the accumulated metric of all the combinations that are generated above is calculated as

$$|z_i - \mathbf{R}_{n_T-i+1,n_T-i+1:n_T}[c_x, c_{x'}]^T|^2 + E_{x'} \tag{7}$$

where $1 \leq x \leq M$, $1 \leq x' \leq S_{i-1}$ and $E_{x'}$ is accumulated metric that is transferred from the $(i-1)$th stage along with the transmitted symbol candidate sets, $c_{x'}$. In (7), $\mathbf{R}_{n_T-i+1,n_T-i+1:n_T}$ is denoted as vector with elements from the $(n_T - i + 1)$th to the $(n_T)$th in the $(n_T - i + 1)$th row of $\mathbf{R}$. Since the number of $c_{x'}$, which is surviving transmitted symbol candidates for the first to the $(i-1)$th symbol is, $S_{i-1}$, the number of all the transmitted symbol candidate sets, $[c_x, c_{x'}]$, for the $i$th to the first symbols is $MS_{i-1}$. Among those combinations, $S_i$ surviving transmitted symbol candidate sets for the $i$th to the

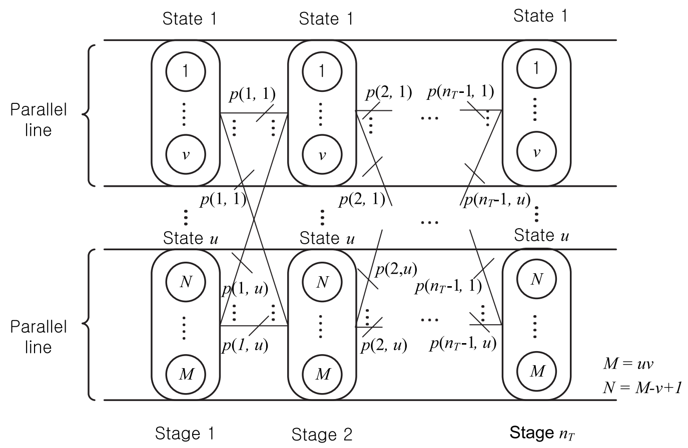



Fig. 3. Trellis-structured QRM-MLD.

first symbols are selected in the ascending order of accumulated metric in (7). Then, $S_i$ selected transmitted symbol candidate sets, $[c_x, c_{x'}]$, in the $i$th stage are transferred to the next $(i+1)$th stage along with their accumulated metric values. The process as above is repeated up to the final stage, that is, $i = n_T$. At the final stage, the combinations of all the possible transmitted symbol candidate for $s_{n_T}$ and the transmitted symbol candidate sets, $c_{x'}$, which survive throughout all the previous stages are considered for the final detection of symbol vector, $\mathbf{s}$, and one combination with the smallest accumulated metric is selected. The elements of selected combination are hard decision output of QRM-MLD.

### *B. Decoding Structure*

MLD is composed of tree-structure with $n_T$ stage, and all the paths in tree-structure are considered as candidates to determine exact symbols. The conventional QRM-MLD is also composed of tree-structure as shown in Fig. 2. However, conventional QRM-MLD searches only some selected paths instead of all the paths. Note that, though conventional QRM-MLD has less complexity than that of MLD, it still has low parallelism/regularity because it is tree-structured detection as mentioned above.

## IV. TRELLIS-STRUCTURED QRM-MLD

Using the trellis-structure proposed for MLD in [5] and applying it to QRM-MLD, we have trellis-structured QRM-MLD as shown in Fig. 3, where the three parameters are defined as in [5]:

(i) $u$: The number of states in trellis-structured QRM-MLD.
(ii) $v$: The number of sub-states per each state in trellis-structured QRM-MLD. Therefore, we have $M = uv$.
(iii) $p(i, j)$ $(1 \leq i \leq n_T, 1 \leq j \leq u)$: The required number of surviving paths from each state in the $i$th stage to the $(i+1)$th stage. Then, the number of total surviving paths from the $i$th stage to the $(i+1)$th stage is $\sum_j p(i, j)$.

Note that our trellis-structured QRM-MLD differs from [5] in that the number of surviving paths varies in every stage and state. In [5], a certain number of paths are selected as surviving paths at all the states. Hence, it cannot assign efficient number of surviving paths at each state. On the other hand, in our proposed algorithm, the number of surviving paths for the $j$th state in the $i$th stage is independently controlled by two thresholds which will be explained in more detail soon. Hence, our proposed algorithm can assign the efficient number of surviving paths at each state. Fig. 4 depicts an example of operation using the two thresholds at each stage. Using [4] and [5], the threshold, $T_i$, which is one of the two thresholds, at the $i$th stage is calculated as

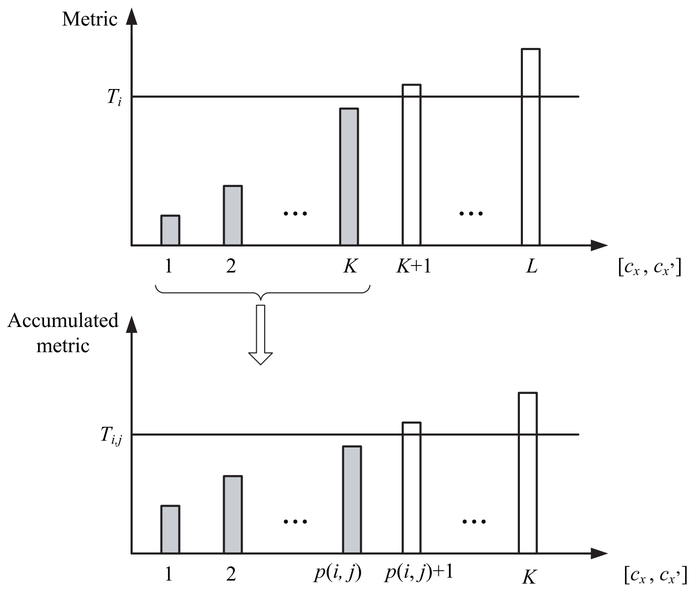



Fig. 4. An example of selecting paths by thresholds in (8) and (10) for the $j$th state of the $i$th stage. $L = v\sum_j p(i-1, j)$, $p(i, j) \leq v$.

$$T_i = |z_i - \mathbf{R}_{n_T-(i-1)}\hat{\mathbf{s}}| + X\sigma \tag{8}$$

where $\mathbf{R}_{n_T-(i-1)}$ is $(n_T - (i-1))$th row vector in (5), $\hat{\mathbf{s}} = \underset{\mathbf{c}}{\text{argmin}}\, \|(\mathbf{R}^H\mathbf{R})^{-1}\mathbf{R}^H\mathbf{z} - \mathbf{c}\|$ for $n_T \times 1$ vector, $\mathbf{c}$, whose elements are composed of $M$ constellations in $M$-QAM, $X$ is predetermined value and $\sigma$ is noise standard deviation. The $\sigma$ is obtained from $E_b/N_o$ as defined in [10]

$$\frac{E_b}{N_o} = \frac{n_R}{b}\frac{E_s}{N_o} \tag{9}$$

which can be regarded as the total received $E_b/N_o$ per transmit antenna, where $b$ is the number of bits per symbol and $E_b$ and $E_s$ are average energy per bit and symbol, respectively.

In the $i$th stage, the paths with metric smaller than threshold in (8) are selected as reliable paths at each state as shown in Fig. 4. In the high SNR region, the more reliable bound for path selection is provided by threshold in (8) than that in the low SNR region since the probability that $\hat{\mathbf{s}}$ is the same as the transmitted symbol vector is getting higher as the SNR increases. Then, due to the second term in (8), the threshold for selecting the reliable path provides small bound when SNR region is high. In this reason, the threshold in (8) leads to select the efficient number of paths which will be transferred to the next step to determine the surviving paths. The parameter, $X$, in (8) provides a trade-off between computational complexity and performance. In the case of large value of $X$, the performance can be improved by reducing the miss-selection probability at the cost of increased computational complexity.

Now, we present how to determine the surviving paths at each state. Using the method of [4], the number of surviving paths, $p(i,j)$ $(1 \leq i \leq n_T,\ 1 \leq j \leq u)$ is controlled by the other threshold which can be calculated as

$$T_{i,j} = E_{(i,j),min} + Y\sigma \tag{10}$$

where $E_{(i,j),min}$ is the smallest accumulated metric at the $j$th state in the $i$th stage, $Y$ is predetermined value such as $X$ in the above, and $\sigma$ is noise standard deviation obtained from (9). The paths with accumulated metric smaller than the threshold, $T_{i,j}$, are selected as the surviving paths as shown in Fig. 4. From [4], we know that selecting paths with accumulated metric of smaller than $T_{i,j}$ as surviving paths means that the paths whose probability to be correct path is higher than $P_{(i,j),min}/e^{(Y/\sigma)}$ are selected as surviving paths at each state, where $P_{(i,j),min}$ is the probability that the path with $E_{(i,j),min}$ as accumulated metric is the correct path. In addition, $Y$ in (10) provides a trade-off between computational complexity and performance as the same principles with $X$ in (8). As the values of $X$ and $Y$ are larger, performance is improved at the expense of an increase in computational complexity. It may not be feasible to find the optimal values of $X$ and $Y$ in terms of the performance and complexity according to the several communication conditions (i.e., $n_t, n_r, u, v$). The appropriate values of $X$ and $Y$ can be properly determined based on the several communication situations from some simulation results. We show in Section V from Fig. 6 how the $X$ and $Y$ effect on the performance for various values of $X$ and $Y$. To stop increasing computational complexity, at each state, $v$ surviving paths are selected at most even though the number of paths that are satisfied with the threshold in (10) is more than $v$. If there does not exist any path that is passed through the threshold in (8) at the $j$th state of the $i$th stage, we select only one surviving path with the smallest accumulated metric among $v\sum_j p(i-1,j)$ paths generated by incoming paths from the $(i-1)$th stage. After the final stage, we obtain $\sum_j p(n_T, j)$ surviving paths with $bn_T$ bits per each path. We select one final path with the smallest accumulated metric, and $bn_T$ bits along this final path are hard decision output. The flowchart for the proposed algorithm is shown in Fig. 5.

Note that the setting of parameters, $u, v$ and $p(i,j)$, plays a critical role in the trade-off among performance, computational complexity and operation speed [5]. The larger surviving paths $\sum_j p(i,j)$ there are in the $i$th stage, the better performance the scheme has. When $\sum_j p(i,j)$ is constant in all the stages, the smaller the value of $u$, the better the scheme performance. The computational complexity grows when incoming paths from previous stage increase, because their metrics should be computed to select surviving paths at each state. That is, the larger $\sum_j p(i,j)$, the higher the increasing complexity. The complexity of metric calculation is increased by $O(N)$ and that of sorting using quicksort algorithm for selecting the paths is increased by $O(N \log_2 N)$ [11] as the number of paths is increased, where $N$ is the number of paths. The high speed VLSI

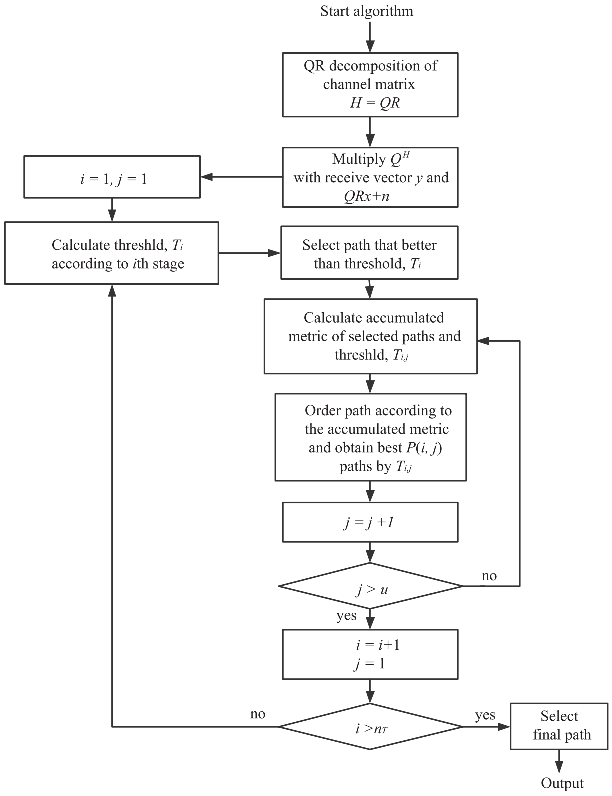



Fig. 5. Trellis-structured QRM-MLD algorithm.

implementation can be acquired by incrementing $u$, because detection processes in $u$ parallel lines are independently proceeded at the same time along the parallel line.

## V. SIMULATION RESULTS

In this section, we provide simulation results to illustrate the performance and computational complexity of our proposed trellis-structured QRM-MLD. We assume that the channel is block fading and the channel coefficients, generated according to the Rayleigh distribution, are constant for a period of one transmit time slot at the transmitter (i.e., the path gains are constant over $n_T$ transmitted symbols in this paper). In our simulation, we consider $n_T = n_R = 2$ and $n_T = n_R = 4$ MIMO system with 16-QAM modulation and assume perfect channel state information (CSI) at the receiver is available. In Fig. 6, we present performance of trellis-structured QRM-MLD for the several $(u, v)$ cases when $X$ and $Y$ are set to the same values as $4.2, 2.7,$ and $1$. The parameters, $u$ and $v$, are the number of states and sub-states in the trellis-structured QRM-MLD, respectively. Note that $X$ and $Y$, are predetermined parameters of the two different thresholds in (8) and (10), respectively. The curves in Fig. 6 are obtained by applying the two thresholds for selecting paths as mentioned in section IV. We can see from Fig. 6 a slight performance degradation comparing with the performance of MLD. This is because two thresholds for selecting paths result in deprivation of opportunities for searching the paths. Instead, we can reduce the complexity in calculating and ordering metrics. Fig. 7 shows the average number of metric calculation per

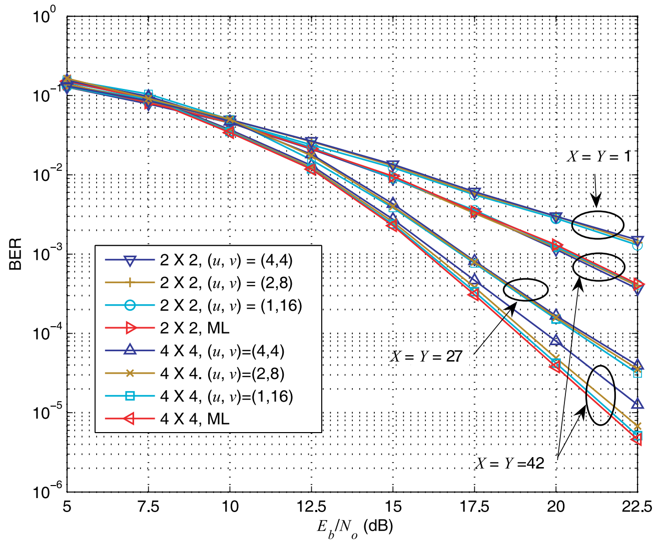



Fig. 6. Performance of the trellis-structured QRM-MLD for the several cases in the $2 \times 2$ and $4 \times 4$ MIMO systems with 16-QAM. The predetermined values, $X$ and $Y$, in the two different thresholds are set to the same values as $1,\ 2.7$, and $4.2$. $u$ and $v$ are the number of states and sub-states, respectively.

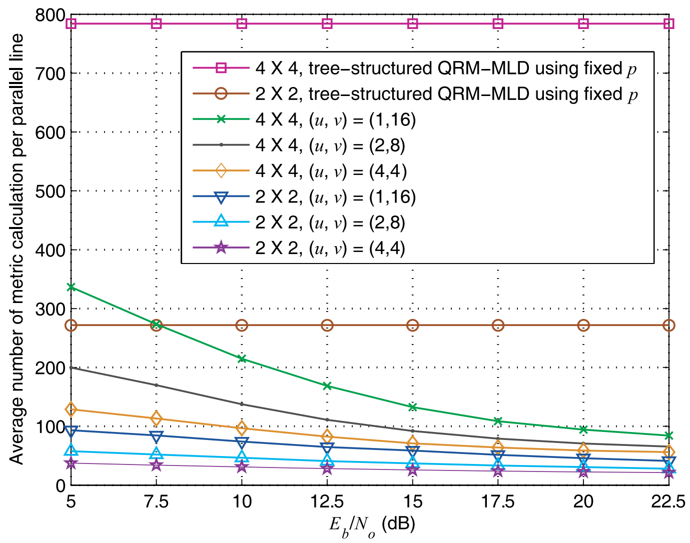



Fig. 7. The average number of metric calculation per parallel line for the trellis-structured QRM-MLD and the tree-structured QRM-MLD using the fixed number of surviving paths, $p$ ($p = 16$), in the $2 \times 2$ and $4 \times 4$ MIMO systems with 16-QAM. The predetermined values, $X$ and $Y$, in the two different thresholds are set to the same value as $4.2$. $u$ and $v$ are the number of states and sub-states, respectively.

parallel line according to the SNR. The complexity on the y-axis in Fig. 7 is measured by counting the metric calculations in (6) and (7) that are calculated for determining the symbols which are transmitted during one transmit time slot. We average the number of metric calculations per parallel line that are counted as above over 10,000 simulation results. The threshold parameters, $X$ and $Y$, are set to the same value as 4.2 in Fig. 7. For the tree-structured QRM-MLD using the fixed number of surviving paths, $p$, [12] the average values of metric calculation, calculated as $M + pM(n_t - 1)$, are constant and larger than those of the trellis-structured QRM-MLD over all the SNR regions. In contrast, the average value of metric calculation per parallel line in the trellis-structured QRM-MLD for each $(u, v)$ case converges to $(M + uM(n_t - 1))/u$ as the SNR increases to infinity. The reason for converging to the certain number is that the average number of surviving paths from the each sub-state to the next stage is reduced to the only one as the SNR increases. Overall the SNR regions, the averaged complexity of metric calculation for the $(u, v) = (1, 16)$ case will be decreased to approximately 8% that of the tree-structured QRM-MLD using the fixed number of surviving paths, $p$ ($p = 16$), in the $4 \times 4$ MIMO system with 16-QAM modulation. We can also see from Fig. 7 that average metric computational complexity per parallel line is reduced as the number of states in the trellis-structured QRM-MLD increases. The low average number of metric calculation per parallel line leads to enhance the operation speed in VLSI implementation.

## VI. CONCLUSION

In this paper, we proposed trellis-structured QRM-MLD which provides the regularity and parallelism to ease the VLSI implementation. We also proposed two thresholds for determining the surviving paths and selecting the high reliable paths. From our analysis and selected numerical examples, we showed that our proposed trellis-structured QRM-MLD with two thresholds scheme can provide quite a low computational complexity for MIMO detection while the performance loss compared to the MLD is negligible.

## REFERENCES


[1] E. Telatar, "Capacity of multi-antenna gaussian channels," *European Trans. Telecommun.*, vol. 10, pp. 585–595, Nov. 1999.

[2] B. A. Bjerke and J. G. Proakis, "Multiple-antenna diversity techniques for transmission over fading channels," in *Proc. IEEE WCNC, Italy*, Sept. 1999, pp. 1038–1042.

[3] G. D. Golden, C. J. Foschini, R. A. Valenzuela, and P. W. Wolniansky, "Detection algorithm and initial laboratory results using V-BLAST space-time communication architecture," *Electron. Lett.*, vol. 35, pp. 14–16, Jan. 1999.

[4] H. Kawai, K. Higuchi, N. Maeda, and M. Sawahashi, "Adaptive control of surviving symbol replica candidates in QRM-MLD for OFDM MIMO multiplexing," *IEEE J. Sel. Areas Commun.*, vol. 24, pp. 1130–1140, June 2006.

[5] T. Zhang, Y. Xin, and S. Chen, "Parallelism/regularity-driven MIMO detection algorithm design," in *Proc. IEEE ISCAS, Kobe, Japan*, 2005, pp. 4959–4962.

[6] S. M. Kang and Y. Leblebici, *CMOS Digital Integrated Circuits: Analysis and Design*. McGraw-Hill, 3rd ed., 2002.

[7] F. Sun and T. Zhang, "Parallel high-througput limited search trellis decoder VLSI design," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 13, pp. 1013–1022, Sept. 2005.

[8] R. Mehra and J. Rabaey, "Exploiting regularity for low-power design," in *Proc. IEEE ICCAD*, Nov. 1996, pp. 166–172.

[9] J. B. Anderson and S. Mohan, "Sequential coding algorithms: A survey and cost analysis," *IEEE Trans. Commun.*, vol. 32, pp. 169–176, Feb. 1984.

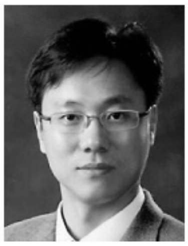
[10] X. Zhu and R. D.Murch, "Performance analysis of maximum likelihood detection in a MIMO antenna system," *IEEE Trans. Commun.*, vol. 50, pp. 187–191, Feb. 2002.

[11] C. A. R. Hoare, "Quicksort," *Comput. J.*, vol. 5, no. 1, pp. 10–15, 1962.

[12] K. J. Kim and J. Yue, "Joint channel estimation and data detection algorithms for MIMO-OFDM systems," in *Proc. 36th Asilomar Conf. Signals, Syst., Compute.*, Nov. 2002, pp. 1857–1861.

**Sang-Ho Choi** received the B.S. and M.S. degrees in Electrical Engineering from the Korea University, Seoul, Korea, in 2006 and 2008, respectively. He is now at Samsung Electro-Mechanics. His current research interests are wireless MODEM design.

**Jun Heo** received the B.S. and M.S. degrees in Electronics Engineering from Seoul National University, Seoul, Korea in 1989 and 1991, respectively and the Ph.D. degree in Electrical Engineering from the University of Southern California, Los Angeles, USA in 2002. During the 1991–1997, he was a Senior Research Engineer at LG Electronics Co., Inc. During the 2003–2006, he was an Assistant Professor in the Electronics Engineering Departmemt, Konkuk University, Seoul, Korea. He is presently an Assistant Professor in the School of Electrical Engineering at Korea University, Seoul, Korea. His research interests include channel coding theory and digital communication systems.



**Young-Chai Ko** received the B.Sc. degree in Electrical and Telecommunication Engineering from the Hanyang University, Seoul, Korea, and the M.S.E.E. degree and the Ph.D. degree in Electrical Engineering from the University of Minnestoa, Minneapolis, MN, USA in 1999 and 2001, respectively. In March 2001 he joined the Texas Instruments, Inc. San Diego Wireless Center in San Diego, CA, where he conducted the system design including the physical layer synchronization, channel estimation, MIMO systems, and the optimization of the RAKE receiver for the chipset development of the 3G UMTS standards. He has been with Department of Electronics and Computer Engineering, Korea University as Assistant Professor since March in 2004. His research focus is on the performance analysis and the design of wireless communication systems. He received the Best Paper Award of the IEEE Vehicular Technology Conference (VTC'99-Fall), Amsterdam, Netherlands, for his contribution to the performance analysis on the diversity systems.