

REPORT DOCUMENTATION PAGE			Form Approved OMB NO. 0704-0188		
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA, 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p>					
1. REPORT DATE (DD-MM-YYYY)		2. REPORT TYPE Technical Report		3. DATES COVERED (From - To) -	
4. TITLE AND SUBTITLE Fixed-point Design of the Lattice-reduction-aided Iterative Detection and Decoding Receiver for Coded MIMO Systems			5a. CONTRACT NUMBER W911NF-11-1-0542		
			5b. GRANT NUMBER		
			5c. PROGRAM ELEMENT NUMBER 611102		
6. AUTHORS Qingsong Wen, Minzhen Ren, Xiaoli Ma			5d. PROJECT NUMBER		
			5e. TASK NUMBER		
			5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAMES AND ADDRESSES Georgia Tech Research Corporation Office of Sponsored Programs Georgia Tech Research Corporation Atlanta, GA 30332 -0420			8. PERFORMING ORGANIZATION REPORT NUMBER		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Research Office P.O. Box 12211 Research Triangle Park, NC 27709-2211			10. SPONSOR/MONITOR'S ACRONYM(S) ARO		
			11. SPONSOR/MONITOR'S REPORT NUMBER(S) 60712-NS-II.6		
12. DISTRIBUTION AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.					
13. SUPPLEMENTARY NOTES The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.					
14. ABSTRACT This technical report summarizes the fixed point implementation for lattice-reduction aided detectors. More important, this report illustrates the performance of coded LR aided detectors.					
15. SUBJECT TERMS lattice reduction, fixed point realization, error correction code, soft decoding					
16. SECURITY CLASSIFICATION OF:		17. LIMITATION OF ABSTRACT		15. NUMBER OF PAGES	
a. REPORT UU	b. ABSTRACT UU	c. THIS PAGE UU	UU	19a. NAME OF RESPONSIBLE PERSON Xiaoli Ma	
				19b. TELEPHONE NUMBER 404-385-7456	

## **Report Title**

Fixed-point Design of the Lattice-reduction-aided Iterative Detection and Decoding Receiver for Coded MIMO Systems

### **ABSTRACT**

This technical report summarizes the fixed point implementation for lattice-reduction aided detectors. More important, this report illustrates the performance of coded LR aided detectors.

# Fixed-point Design of the Lattice-reduction-aided Iterative Detection and Decoding Receiver for Coded MIMO Systems

Qingsong Wen, Minzhen Ren, Xiaoli Ma

School of Electrical and Computer Engineering,  
Georgia Institute of Technology, Atlanta, Georgia 30332

## I. INTRODUCTION

With the evolution of wireless communication systems, the multiple input multiple output (MIMO) system has been adopted to provide higher data rate [1]. In addition, error control codes (ECC) are usually included in the system to enhance the information reliability, e.g., Turbo Codes [2] and Low Density Parity Check (LDPC) codes [3]. The challenge to apply both MIMO and ECC into wireless systems is on designing a reliable but low-complexity receiver.

The optimal receiver for coded MIMO systems is to use a joint detector and decoder for the whole coded data block, which is extremely complex and infeasible in the practical system due to the long length of coded data block. Although decoupled detectors and decoders can significantly reduce the complexity, the performance would be largely degraded compared to the optimal receiver. In order to balance the complexity and performance, the receiver with iterative detection and decoding (IDD) is proposed in [4], where the separate soft-input soft-output (SISO) detector and SISO decoder are used to achieve the near-optimal performance by exchanging extrinsic information iteratively.

The optimal SISO detector under IDD for coded MIMO systems would be the maximum a posteriori (MAP) detector, which is often with high complexity especially when the constellation size and/or the channel dimension are high. The list MIMO detectors, such as the list sphere detector [4] and the list sequential detector [5], are an attractive choice as they allow a flexible

tradeoff between performance and complexity. One key issue of the list MIMO detector is to generate a list of candidates containing the transmitted symbol vectors with low complexity. The way to find the list and the number of candidates in the list are directly related to both performance and complexity. So it is desirable if the detector can obtain the near-optimal performance only using a small number of candidates.

Recently, lattice reduction (LR) technique has been proposed to improve the performance of MIMO detector in [6], [7], and [8], by transforming the channel matrix into a better-conditioned matrix. It is shown that LR-aided linear detectors can achieve the full diversity of the maximum likelihood (ML) receiver. Furthermore, the combination of LR with list MIMO detection like K-best detector [9] shows that it can maintain near-ML performance even with very low K values (the number of candidates), which means much lower complexity of the detector. The LR-aided IDD algorithms with list MIMO detector have been well studied in the literature [10]. However, there are few papers focusing on the fixed-point design for the whole LR-aided IDD system, which is a key step for practical hardware implementation in VLSI chips or FPGAs. In this paper, we evaluated the LR-aided IDD performance under finite precision in operands and arithmetic operations, and designed the detailed fixed-point implementation for the whole LR-aided IDD receiver based on that the bit error rate (BER) performance of the fixed-point system could be within 0.2dB degradation compared to the performance of the corresponding floating-point system.

The rest of this paper is organized as follows. Section II presents the system model of the LR-aided IDD receiver for MIMO coded systems. Section III introduces the key algorithms used in the fixed-point LR-aided IDD receiver. Section IV provides the detailed fixed-point implementation for the whole LR-aided IDD receiver followed by the conclusion in Section V.

## II. SYSTEM MODEL

Consider a coded multiplexing transmission system depicted in Fig. 1. At the transmitter, a sequence of binary information bits  $\mathbf{b}$  is random produced, passed the ECC, and interleaved. Then the coded sequence  $\mathbf{c}$  is mapped into a symbol sequence  $\mathbf{s}$  where the constellation size is  $k$  bits/symbol. For the system with  $N$  transmit and  $M$  receive antennas, the MIMO transmission can be expressed as:

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{w}, \quad (1)$$

where the  $\mathbf{H}$  is assumed as a  $M \times N$  complex Gaussian channel matrix with zero mean and unit variance, the  $N \times 1$  vector  $\mathbf{c}$  consists of the information symbols drawn from a constellation  $S$ ,  $\mathbf{y}$  is the  $M \times 1$  received vector, and  $\mathbf{w}$  is the complex additive white Gaussian noise with variance  $\sigma_w^2$ . Suppose that  $E[\mathbf{s}\mathbf{s}^H] = \mathbf{I}_N$ , and  $E[\mathbf{w}\mathbf{w}^H] = \sigma_w^2 \mathbf{I}_M$ . We assume that the channel matrix  $\mathbf{H}$  is time-invariant during a certain block which is greater than a symbol period and change independently from block to block, and it is known at the receiver but unknown at the transmitter.

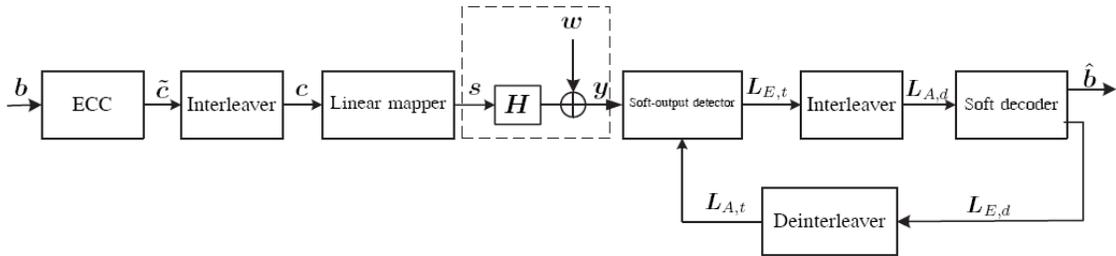


Fig. 1. Block diagram of LR-aided IDD receiver for coded MIMO systems

At the receiver, LR-aided IDD structure is adopted to exchange extrinsic information between the SISO detector and the SISO decoder. The extrinsic information  $\mathbf{L}_{E,t}$  is first calculated by the SISO detector based on the observation  $\mathbf{y}$ , the channel  $\mathbf{H}$ , and the prior information  $\mathbf{L}_{A,t}$  which is fed back by the SISO decoder. Then, the extrinsic information from the detector is passed through the interleaver to the SISO decoder, which takes it as priori information  $\mathbf{L}_{A,d}$  to obtain the information bits and calculate new extrinsic information  $\mathbf{L}_{E,d}$  to feed back to the detector. Thus, the receiver is designed in an iterative way between the detection and decoding.

### III. KEY ALGORITHMS IN LR-AIDED IDD RECEIVER

#### A. Lattice Reduction

In the MIMO transmission model in Eq. (1), the received signal vector  $\mathbf{y}$  is the noisy observation of the vector  $\mathbf{H}\mathbf{s}$ , which is in the lattice spanned by the columns of  $\mathbf{H}$  since all the entries of  $\mathbf{s}$  can be transformed to complex integers by shifting and scaling. In general, a lattice has more than one set of basis vectors.

There exist some bases that span the same lattice as  $\mathbf{H}$  but are closer to orthogonality than  $\mathbf{H}$ . The process of finding a basis closer to orthogonality is called LR. Theoretically, finding an optimal set of bases (closest to orthogonality) in a lattice is computationally expensive. Thus, the ultimate goal of LR algorithms is to find a "better" channel matrix  $\tilde{\mathbf{H}} = \mathbf{H}\mathbf{T}$  where  $\mathbf{T}$  as a unimodular matrix, which means that all the entries of  $\mathbf{T}$  and  $\mathbf{T}^{-1}$  are complex integers and the determinant of  $\mathbf{T}$ , is  $\pm 1$  or  $\pm j$ . The restrictions on the matrix  $\mathbf{T}$  ensure that the lattice generated by  $\tilde{\mathbf{H}}$  is the same as that of  $\mathbf{H}$ .

Generally, LR techniques involve preprocessing  $\mathbf{H}$  to produce a reduced-lattice basis  $\tilde{\mathbf{H}} = \mathbf{H}\mathbf{T}$ . This factorization allows us to rewrite the system in Eq. (1) as

$$\mathbf{y} = \mathbf{H}\mathbf{T}(\mathbf{T}^{-1}\mathbf{s}) + \mathbf{w} = \tilde{\mathbf{H}}\mathbf{z} + \mathbf{w}. \quad (2)$$

Here we adopt the complex LLL (CLLL) algorithm [8], [11] to perform the LR on the channel matrix  $\mathbf{H}$ . The detailed pseudo-code of the CLLL algorithm can be summarized as follows in Fig. 2 [8].

---

INPUT:  $\mathbf{H}$ ; OUTPUT:  $\tilde{\mathbf{Q}}, \tilde{\mathbf{R}}, \mathbf{T}$

---

```

(1)  $[\tilde{\mathbf{Q}}, \tilde{\mathbf{R}}] = \text{QR Decomposition}(\mathbf{H})$ ;
(2)  $\delta \in (\frac{1}{2}, 1)$ ;
(3)  $m = \text{size}(\mathbf{H}, 2)$ ;
(4)  $\mathbf{T} = \mathbf{I}_m$ ;
(5)  $k = 2$ ;
(6) while  $k \leq m$ 
(7)   for  $n = k - 1 : -1 : 1$ 
(8)      $u = \text{round}((\tilde{\mathbf{R}}(n, k) / \tilde{\mathbf{R}}(n, n)))$ ;
(9)     if  $u \neq 0$ 
(10)       $\tilde{\mathbf{R}}(1 : n, k) = \tilde{\mathbf{R}}(1 : n, k) - u \cdot \tilde{\mathbf{R}}(1 : n, n)$ ;
(11)       $\mathbf{T}(:, k) = \mathbf{T}(:, k) - u \cdot \mathbf{T}(:, n)$ ;
(12)    end
(13)  end
(14)  if  $\delta |\tilde{\mathbf{R}}(k-1, k-1)|^2 > |\tilde{\mathbf{R}}(k, k)|^2 + |\tilde{\mathbf{R}}(k-1, k)|^2$ 
(15)    Swap the (k-1)th and kth columns in  $\tilde{\mathbf{R}}$  and  $\mathbf{T}$ 
(16)     $\Theta = \begin{bmatrix} \alpha^* & \beta \\ -\beta & \alpha \end{bmatrix}$  where  $\alpha = \frac{\tilde{\mathbf{R}}(k-1, k-1)}{\|\tilde{\mathbf{R}}(k-1:k, k-1)\|}$ ;
       $\beta = \frac{\tilde{\mathbf{R}}(k, k-1)}{\|\tilde{\mathbf{R}}(k-1:k, k-1)\|}$ ;
(17)     $\tilde{\mathbf{R}}(k-1 : k, k-1 : m) = \Theta \tilde{\mathbf{R}}(k-1 : k, k-1 : m)$ ;
(18)     $\tilde{\mathbf{Q}}(:, k-1 : k) = \tilde{\mathbf{Q}}(:, k-1 : k) \Theta^H$ ;
(19)     $k = \max(k-1, 2)$ ;
(20)  else
(21)     $k = k + 1$ ;
(22)  end
(23) end

```

---

Fig. 2. Pseudo-code of CLLL algorithm

For the CLLL algorithm, the main computation parts contain the QR Decomposition (Line 1 in Fig. 2), the Size Reduction part (Line 7-13 in Fig. 2), the Complex Lovasz Condition (Line 14 in Fig. 2), and the Basis Updating part (Line 16-18 in Fig. 2). In order to facilitate the fixed-point design while to keep the performance at the same time, some modifications are adopted in [12], where the Relaxed Size Reduction Condition is defined for the calculating of Size Reduction part, the Complex Lovasz Condition is replaced by the Siegel Condition, the integer-rounded division (Line 8 in Fig. 2) is implemented by using a single Newton-Raphson (NR) iteration method, and the calculation of  $\Theta$  (Line 16 in Fig. 2) is completed by Householder CORDIC algorithm.

### B. List MIMO detector

For the list MIMO detector in LR-aided IDD receiver, the authors in [10] proposed three methods, i.e. fixed radius algorithm (FRA), fixed candidates algorithm (FCA), and fixed memory-usage algorithm (FMA). FRA as well as FMA is a combination of sphere decoding [13] and LR, which searches all possible candidates in the sphere. In this case the number of candidates is random, which may cause difficulty on hardware implementation. FCA is a combination of K-best algorithm [14] and LR, which applies an element-by-element searching with a fixed number of points on each layer so that it is suitable for the hardware implementation.

For LR-aided linear hard detectors, LR is first applied on the channel matrix  $\mathbf{H}$  followed by the linear equalization based on the reduced-lattice basis  $\tilde{\mathbf{H}}$ . For example, when Zero Forcing (ZF) equalizer is adopted, we can get

$$\mathbf{x} = \tilde{\mathbf{H}}^\dagger \mathbf{y} = \mathbf{T}^{-1} \mathbf{s} + \tilde{\mathbf{H}}^\dagger \mathbf{w} \triangleq \mathbf{z} + \mathbf{n}. \quad (3)$$

Then we need to obtain an estimate of  $\mathbf{z}$  in Eq. (3) and next the  $\mathbf{s}$  is estimated through one-to-one mapping, which implies we need to get a candidate list of  $\mathbf{z}$  in the list MIMO detector. Different from the SD method in [13], here the sphere is built in the  $\mathbf{z}$ -domain centered at LR-aided estimate instead of the  $\mathbf{s}$ -domain centered at ZF estimate or other estimate from preprocessing. However, because of matrix  $\mathbf{T}$ , the constellation of  $\mathbf{z}$  is not ready. Some candidates  $\hat{\mathbf{z}}$  on integer lattice may not generate valid candidates in  $\mathbf{s}$ -domain. One way is to find all possible  $\mathbf{z}$ 's and then perform searching, which costs high computational complexity. Since our final goal is to obtain  $\mathbf{s}$  not  $\mathbf{z}$  and the alphabet of  $\mathbf{s}$  is known, so we need to find the list of candidates on  $\mathbf{s}$ ,

$\mathcal{C}_s$  as:

$$\mathcal{C}_s = \{\tilde{\mathbf{s}} : \|\mathbf{T}^{-1}\tilde{\mathbf{s}} - \mathbf{x}\|^2 < r_z\}. \quad (4)$$

To further reduce the complexity, we can apply QR decomposition for  $\mathbf{T}^{-1}$  so that  $\mathbf{T}^{-1} = \mathbf{Q}_T \mathbf{R}_T$ , then we obtain

$$\|\mathbf{T}^{-1}\tilde{\mathbf{s}} - \mathbf{x}\|^2 = \|\mathbf{Q}_T^H \mathbf{x} - \mathbf{R}_T \tilde{\mathbf{s}}\|^2. \quad (5)$$

Here low complexity tree-searching methods can be performed by starting from the bottom layer. In order to facilitate the hardware implementation, we select the FCA as the list MIMO detector in the LR-aided IDD receiver for fixed-point design because its breadth-first tree-search method has a fixed throughput like K-best method. Furthermore, FCA always includes the LR-aided hard-decision in the candidate list to guarantee diversity. The detailed pseudo-code of the FCA algorithm can be summarized as follows in Fig. 3 [10].

Input:  $\mathbf{y}$ ,  $\mathbf{H}$ ,  $K_p$ ; Output:  $\mathcal{C}_s$   
Initialize:  $\mathbf{Dist} = \text{zeros}(1, K_p)$ ; and  $\mathcal{C}_s = \emptyset$   
 $\mathbf{Dist}$  records the distance between  $\mathbf{R}_T \tilde{\mathbf{s}}$  and  $\mathbf{Q}_T^H \mathbf{x}$ , for  $\tilde{\mathbf{s}} \in \mathcal{C}_s$

---

S1.  $[\tilde{\mathbf{H}}, \mathbf{T}] = \text{CLLL}(\mathbf{H})$ ;  
S2. Hard-decision solution:  $\hat{\mathbf{s}}_{hd}$ ;  
S3.  $[\mathbf{Q}_T, \mathbf{R}_T] = \text{QR decomposition}(\mathbf{T}^{-1})$ ;  
S4.  $\mathbf{x} = \tilde{\mathbf{H}}^\dagger \mathbf{y}$ ;  
S5.  $\mathbf{q} = \mathbf{Q}_T^H \mathbf{x}$ ;  
S6. For  $n = N : (-1) : 1$   
S7.     For each partial candidate vector  $\tilde{\mathbf{s}}_i \in \mathcal{C}_s$ ,  $i \in [1, K_p]$   
S8.         For each symbol  $u_l \in \mathcal{S}$ ,  $l \in [1, |\mathcal{S}|]$  except the one in  $\hat{\mathbf{s}}_{hd}$   
S9.              $\mathbf{D}_t(i, l) = \mathbf{Dist}(i) + |\mathbf{q}(n) - \mathbf{R}_T(n, n : N)[u_l; \tilde{\mathbf{s}}_i]^T|^2$ ;  
S10.             end  
S11.         end  
S12.     Find the  $K_p$  minimum values in  $\mathbf{D}_t$  and save them as  $\mathbf{Dist}$ ;  
S13.     Save the corresponding vectors  $[u_l; \tilde{\mathbf{s}}_i]$  as  $\mathcal{C}_s$ ;  
S14. end

Fig. 3. Pseudo-code of List MIMO detector with FCA

### C. QR Decomposition

QR decomposition (QRD) is an essential component for both above-mentioned CLLL and FCA algorithms. The QRD transform a matrix  $\mathbf{H}$  into a unitary matrix  $\mathbf{Q}$  and an upper triangular

matrix  $\mathbf{R}$ , i.e.,  $\mathbf{H} = \mathbf{QR}$ . Three well known algorithms have been proposed to perform QRD: Gram-Schmidt (GS) algorithm, Householder transformation (HT), and Givens rotation (GR).

In [15], [16], it has been shown that GS can be efficiently implemented through Coordinat Rotation Digital Computer (CORDIC) and Triangular Systolic Array (TSA) algorithms. So GS does not require norm and division operations by CORDIC algorithm, and it can easily adopt parallelism when processing a large matrix by TSA algorithm. Furthermore, GS demonstrates higher numerical stability with VLSI implementation in the QRD process compared with GS and HT methods. Due to these reasons, we select the GS method as the QRD algorithm in the LR-aided IDD receiver.

The QRD process under GS algorithm with TSA and CORDIC [15] can be illustrated on a  $2 \times 2$  complex matrix  $\mathbf{H}$  as:

$$\mathbf{H} = \begin{pmatrix} Ae^{j\theta_a} & Ce^{j\theta_c} \\ Be^{j\theta_b} & De^{j\theta_d} \end{pmatrix}, \quad (6)$$

where  $j = \sqrt{-1}$ ,  $A, B, C, D$  represent the magnitudes, and  $\theta_a, \theta_b, \theta_c, \theta_d$  stand for the angles of the matrix entries. In order to get QRD of the  $\mathbf{H}$  matrix, the  $\mathbf{H}$  is first transformed by the unitary matrix  $\mathbf{Q}_1$  expressed by:

$$\mathbf{Q}_1 = \begin{pmatrix} \cos\theta_1 e^{j\theta_2} & \sin\theta_1 e^{j\theta_3} \\ -\sin\theta_1 e^{j\theta_2} & \cos\theta_1 e^{j\theta_3} \end{pmatrix}, \quad (7)$$

where the three angles  $\theta_1, \theta_2, \theta_3$  are calculated as follows:

$$\begin{aligned} \theta_1 &= \tan^{-1}(C/A), \\ \theta_2 &= -\theta_a, \\ \theta_3 &= -\theta_b. \end{aligned} \quad (8)$$

After the above transformation, we can get an upper triangular matrix  $\mathbf{R}_1$  as:

$$\mathbf{R}_1 = \mathbf{Q}_1 \mathbf{H} = \begin{pmatrix} \cos\theta_1 e^{j\theta_2} & \sin\theta_1 e^{j\theta_3} \\ -\sin\theta_1 e^{j\theta_2} & \cos\theta_1 e^{j\theta_3} \end{pmatrix} \begin{pmatrix} Ae^{j\theta_a} & Ce^{j\theta_c} \\ Be^{j\theta_b} & De^{j\theta_d} \end{pmatrix} = \begin{pmatrix} X & Ye^{j\theta_y} \\ 0 & Ze^{j\theta_z} \end{pmatrix}. \quad (9)$$

Next, the  $\mathbf{R}_1$  is transformed by another simple unitary matrix  $\mathbf{Q}_2$  expressed by:

$$\mathbf{Q}_2 = \begin{pmatrix} 1 & 0 \\ 0 & e^{-j\theta_z} \end{pmatrix}. \quad (10)$$

So that we get the last  $\mathbf{R}$  matrix of the QRD process as follows:

$$\mathbf{R} = \mathbf{Q}_2 \mathbf{R}_1 = \begin{pmatrix} 1 & 0 \\ 0 & e^{-j\theta_z} \end{pmatrix} \begin{pmatrix} X & Y e^{j\theta_y} \\ 0 & Z e^{j\theta_z} \end{pmatrix} = \begin{pmatrix} X & Y e^{j\theta_y} \\ 0 & Z \end{pmatrix}. \quad (11)$$

Based on the above procedure, for a  $4 \times 4$  matrix  $\mathbf{H}$ , the QRD can be implemented through the CORDIC-based systolic array as depicted in Fig. 4.

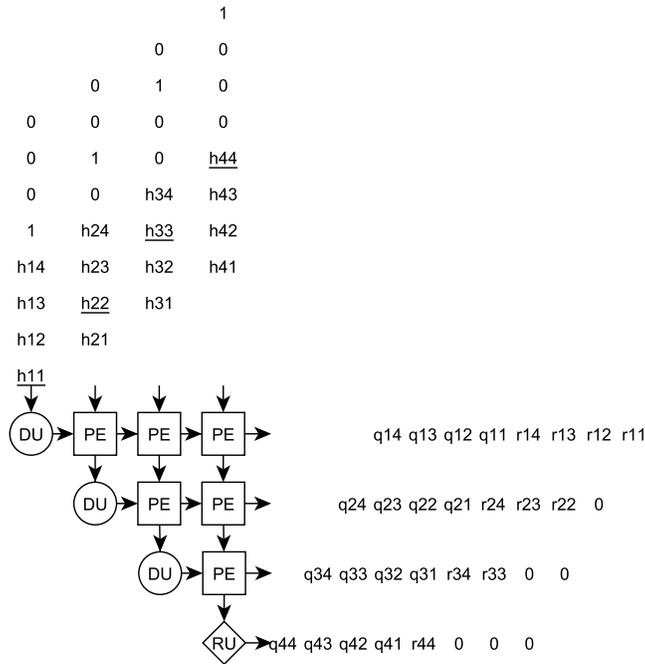


Fig. 4. Architecture diagram of CORDIC-based triangular systolic array to solve systolic array.

Three different types of cells are shown in Fig. 4: delay unit(DU), processing element(PE), and rotational unit(RU) [15]. DU delays the incoming data by number of clock cycles that neighboring cell takes to process the data, then deliver it to PE when it is available. PE, as the most complex unit, can operate in either vectoring mode or rotation mode. In vectoring mode, PE calculates the three angles described in (8), stores them into the cell memory, and meanwhile computes the norm of the complex vector. The computed norm is passed to the east of the cell with a flag that requests the next PE to operate in vectoring mode. In rotation mode, PE rotates the incoming complex vector with the angles stored in the cell memory, and passes the results

from north to south port and west to east port. Fig. 5 depicts the structure of PE in both modes with data flows. Similarly, RU has the same operation modes as PE, but operates in vectoring mode only when a diagonal element from a channel matrix enters from the north port.

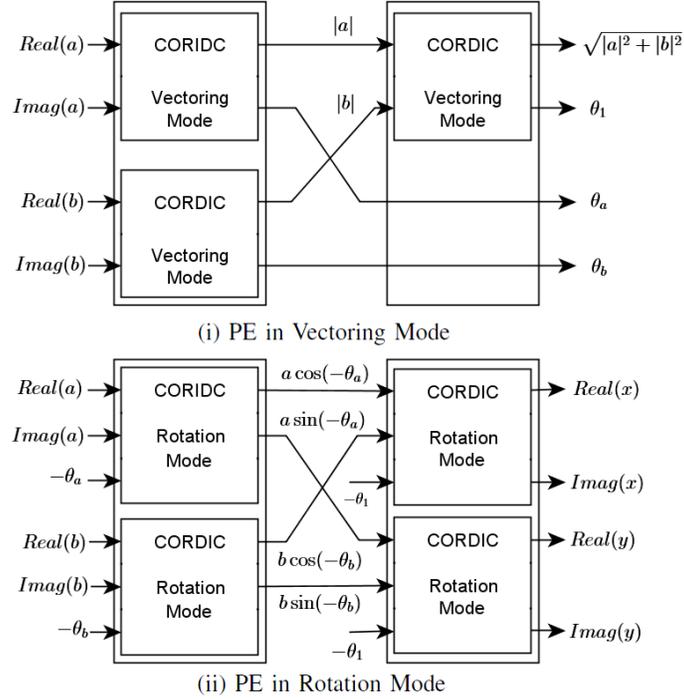


Fig. 5. PE cell structure in (i) vectoring mode and (ii) rotation mode ( $a/b$  denotes the data that is coming into the cell from west/north entrance).

#### D. LLR Computing between the detector and the decoder

The extrinsic information  $L_{E,t}$  shown in Fig. 1 is usually expressed by the log-likelihood ratio (LLR) of each transmitted bit as follows [10]:

$$L_{E,t}(c_i|\mathbf{y}) \approx \frac{1}{2} \max_{\mathbf{c} \in C_s \cap S_{i,+1}} \left\{ -\frac{2}{\sigma_w^2} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|_2^2 + \mathbf{c}^T \mathbf{L}_{A,t} - L_{A,t}(c_i) \right\} - \frac{1}{2} \max_{\mathbf{c} \in C_s \cap S_{i,-1}} \left\{ -\frac{2}{\sigma_w^2} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|_2^2 + \mathbf{c}^T \mathbf{L}_{A,t} + L_{A,t}(c_i) \right\}, \quad (12)$$

where  $C_s$  denotes the candidate list from the list MIMO detector in the LR-aided IDD receiver,  $S_{i,+1}$  represents the subset of  $C_s$  with the  $i$ th bit as  $+1$ , and similarly defined  $S_{i,-1}$ , so that  $C_s = S_{i,+1} \cap S_{i,-1}$ .

Now both complexity and performance of the list MIMO detector depend on the size of the candidate list  $C_s$ . If the list of candidates is too long, the results will be near to the optimal MAP while the complexity is too high (near the exhaustive search). On the other hand, if the list is too short, the performance will be degraded due to the inaccurate  $L_{E,t}$  values. Furthermore, the error of  $L_{E,t}$  is especially large in the case when the output list  $C_s$  includes only candidates with  $c_i$  either  $+1$  or  $-1$ , which may result in very large values in Eq. (12) that would cause the decoder from correcting the falsely detected data.

The undesirable effect of the small candidates in the list MIMO detector can be reduced by LLR clipping [4], which limits the dynamic range of LLR values so that the decoder can still overcome the error data from the detector. The LLR clipping is defined as follows:

$$L_{E,t}^{clip}(c_i|\mathbf{y}) = \begin{cases} L_{E,t}(c_i|\mathbf{y}) & |L_{E,t}(c_i|\mathbf{y})| \leq L_{max}, \\ \text{sign}(L_{E,t}(c_i|\mathbf{y})) \cdot L_{max} & |L_{E,t}(c_i|\mathbf{y})| > L_{max}. \end{cases} \quad (13)$$

where the  $L_{E,t}^{clip}(c_i|\mathbf{y})$  is the clipped LLR and the  $L_{max}$  is the predefined maximum LLR value for  $L_{E,t}$ . Besides improving the performance of the list MIMO detector, LLR clipping can also reduce the word-length of the fixed-point design and decrease the complexity of the hardware implementation.

### E. Turbo Decoding

The Turbo decoder contains two elementary MAP decoders interconnected to each other by interleavers ( $\pi$ ) and deinterleavers ( $\pi^{-1}$ ) in serial way as shown in Fig. 6.

Each elementary decoder has three inputs: the systematic bit ( $y_{ks}$ ), the parity bits from the component encoder ( $y_{kp1}$  or  $y_{kp2}$ ), and the extrinsic information from the other component decoder ( $L(u_k)$ ), also known as a-priori information of the systematic bit. During the Turbo decoding, the component decoders iteratively exchange the probabilities for each information bit represented by LLR, which could ameliorates the LLRs of the information bits and improves the decoding accuracy.

For the fixed-point implementation, here we adopt the well known Max-Log-MAP algorithm, which has near the same performance as the optimal MAP algorithm while with much lower complexity [17]. For the Max-Log-MAP algorithm, the calculation process of each constituent decoder can be summarized in the following parts:

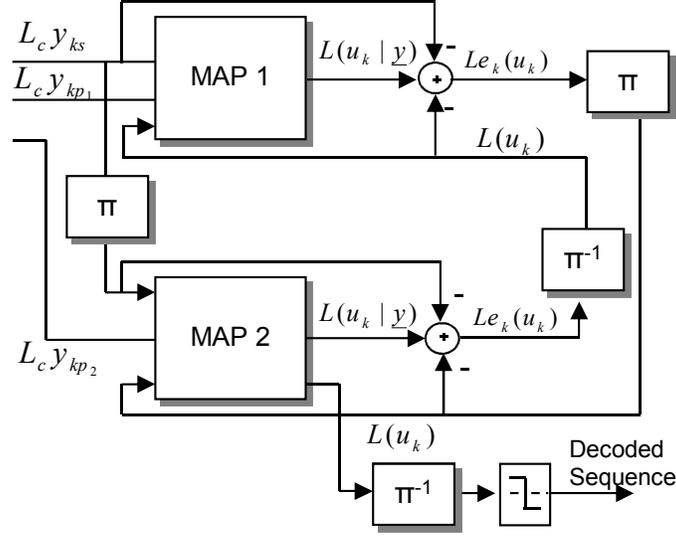


Fig. 6. Architecture diagram of Turbo decoder

#### 1, Branch Metric computing (BM)

$$\gamma_k(s', s) = \frac{1}{2}u_k L(u_k) + \frac{L_c}{2} \sum_{l=0}^{N-1} x_{kl} y_{kl}. \quad (14)$$

#### 2, Forward Recursion computing (FW)

$$\alpha_k(s) = \max_{s'} \{ \alpha_{k-1}(s') + \gamma_k(s', s) \}; k = 0, 1, \dots, N - 1. \quad (15)$$

where  $\alpha_{k=0}(s = 0) = 0$ , and  $\alpha_{k=0}(s \neq 0) = -\infty$ .

#### 3, Backward Recursion computing (BW)

$$\beta_{k-1}(s') = \max_{s'} \{ \beta_k(s) + \gamma_k(s', s) \}; k = N, N - 1, \dots, 1. \quad (16)$$

where  $\beta_{k=N}(s = 0) = 0$ , and  $\beta_{k=N}(s \neq 0) = -\infty$ .

#### 4, LLR computing

$$L(u_k | y) = \max_{(s', s) \Rightarrow u_k = +1} \{ \alpha_{k-1}(s') + \gamma_k(s', s) + \beta_k(s) \} - \max_{(s', s) \Rightarrow u_k = -1} \{ \alpha_{k-1}(s') + \gamma_k(s', s) + \beta_k(s) \}. \quad (17)$$

#### 5. Extrinsic information computing

$$Le_k(u_k) = L(u_k | y) - L_c Y_{ks} - L(u_k). \quad (18)$$

In the Equ. (14)-(18),  $u_k$  is the information bit which produces the transition from state  $s'$  to state  $s$  in the Turbo trellis.  $L(u_k)$  is a priori information and  $x_{kl}$  and  $y_{kl}$  are the expected transmitted symbols and the actual received symbols, respectively.  $L_c$  is the channel reliability defined as  $L_c = 2/\sigma^2$ , where  $\sigma^2$  is the noise average power.

#### IV. FIXED-POINT DESIGN FOR LR-AIDED IDD RECEIVER

In this section, the fixed-point design for the whole LR-aided IDD receiver will be analyzed and decided based on the algorithms of the above section. For the fixed-point simulation, let  $FP(iwl, fwl)$  be the finite representation of an  $wl$ -bit two's complement number where  $fwl$  is the fractional wordlength and  $iwl$  is the integer wordlength including a sign bit, so  $wl = iwl + fwl$ . In order to compare the practical fixed-point performance under different wordlength accuracy with the ideal floating-point performance, all the simulations are based on the same system parameters assumed in the following paragraph.

In this paper, the LR-aided IDD receiver is applied in the i.i.d. Rayleigh fading channel with  $M = N = 4$ , i.e., spatial multiplexing MIMO systems under 4 transmit antennas and 4 receive antennas. The channel is time-invariant for one symbol period and changes independently from symbol to symbol. Modulation scheme is QPSK and the SNR is defined as symbol energy versus noise power, i.e.,  $E[|s|^2]/\sigma_w^2$ . For the FCA algorithm in the list MIMO detector,  $K_p$  is set to 2 except the candidate from LR-aided hard detection. Simulation results show that the number of the candidate list in FCA is almost 3. For the ECC, the parallel rate 1/2 Turbo code is adopted with the generator  $(1, \frac{1+D^2}{1+D+D^2})$ . The information bit sequence is of length 1024. For each information sequence, we perform up to 4 IDD iterations and up to 8 iterations within the turbo decoder as suggested in [4].

##### A. LLR clipping between the detector and the decoder

To study the LLR clipping effect and to find the optimal clipping threshold of the LR-aided IDD receiver, we examined the BER performance under different clipping values as shown in Fig. 7. The simulation results demonstrate that the performance of the system can be clearly improved by applying a proper LLR clipping threshold. On the other hand, either too large clipping values or too small clipping values would degrade the system performance.

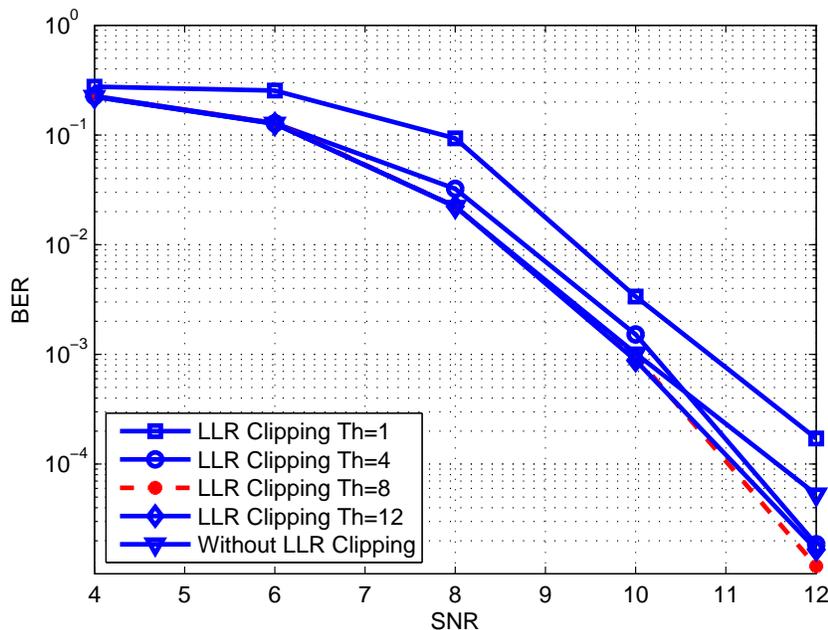


Fig. 7. BER performance under different LLR clipping threshold

Based on the simulation, LLR clipping threshold with  $L_{max} = 8$  is shown to be a appropriate and simple choice to be used, which is also consistent with the results in [18] and [4].

We also examined the effect of iteration times for the IDD and Turbo decoding on the system performance under the above selected clipping threshold with  $L_{max} = 8$ . The simulation results are shown in Fig. 8. It can be seen that as the number of iterations increases, the performance becomes better. However, if we keep increasing the number of iterations, the performance improvement becomes marginal. From Fig. 8, the performance of the system with 3 IDD iterations and 4 Turbo decoding iterations is very near to the performance of the system under 4 IDD iterations and 8 Turbo decoding iterations. In the hardware implementation, low complexity and delay would be desirable when facing the cost. So in the following parts, we only investigate the LR-aided IDD receiver with 3 IDD iterations and 4 Turbo decoding iterations.

### B. Fixed-point design for the List MIMO detector

The QRD part in the list MIMO detector is located in the CLLL and FCA algorithms, where QRD is used for the channel matrix  $\mathbf{H}$  and the unimodular matrix  $\mathbf{T}^{-1}$ .

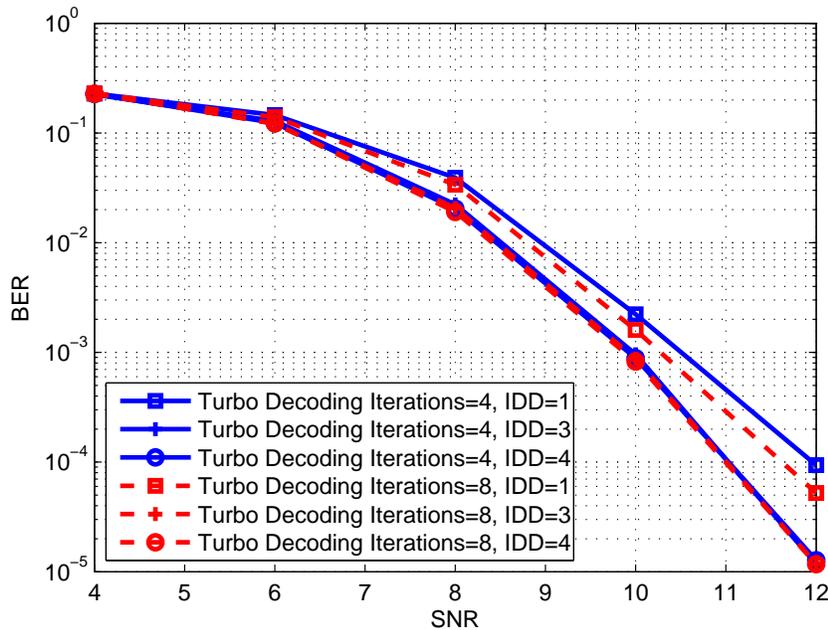


Fig. 8. BER performance under different iteration in IDD and Turbo decoding

For the i.i.d. Rayleigh fading channel  $\mathbf{H}$  with variance of one in our system configuration, the probability that its element energy exceeds 16 is approximately  $6.4 \times 10^{-58}$ , which one can practically ignore. Therefore 5 bits are enough to represent the integer part of  $\mathbf{H}$  elements. During the QRD processing, since the angles in the transforming unitary matrix are well-contained within  $[-\pi, \pi]$ , 4 bits are sufficient to represent the integer part of angles. For the fractional bit width in both  $\mathbf{H}$  data and the angles data, we examined the accuracy of the QRD under different fractional bit width as shown in Fig. 9, where the accuracy is defined as the difference of Frobenius norm between the channel matrix  $\mathbf{H}$  and the product of  $\mathbf{Q}$  and  $\mathbf{R}$ , i.e.,

$$\text{Accuracy(QR model)} = \|\mathbf{H} - \mathbf{QR}\|_F \quad (19)$$

Fig. 9 shows that 16 bits are enough for the fractional wordlength in QRD since in this case both data and angles can achieve an accuracy within 0.14%. In sum,  $FP(5, 16)$  and  $FP(4, 16)$  are suitable for the data and the angles in the QRD module, respectively.

For the QRD of the unimodular matrix  $\mathbf{T}^{-1}$ , the angles property is the same as that in the QRD of the  $\mathbf{H}$ , so the same  $FP(4, 16)$  is adopted; for the data part, because all the entries are Gaussian integers, we can reduce the fractional wordlength and increase the integer wordlength

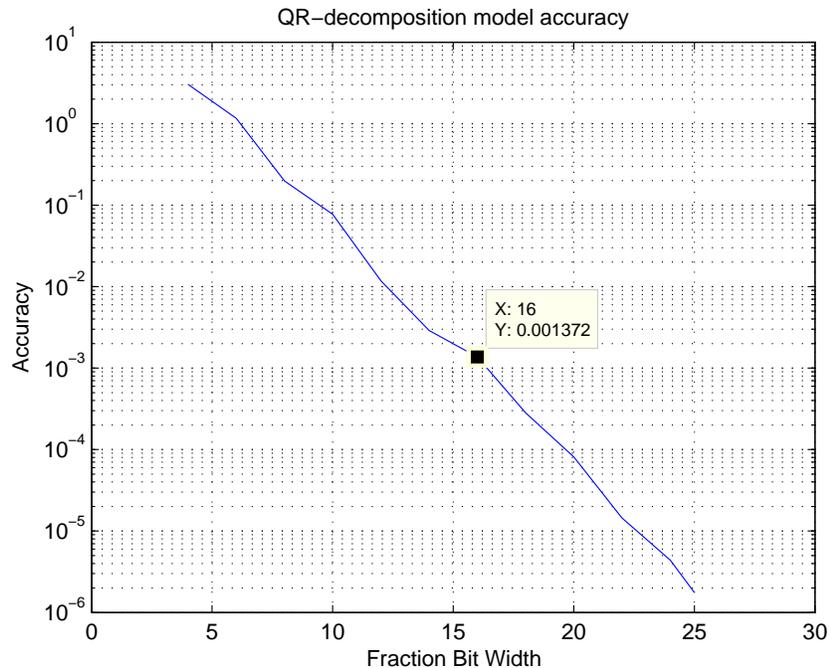


Fig. 9. QR-dcomposition Triangular Systolic Array fixed-point model accuracy

while keep the whole wordlength invariable. Here we adopt  $FP(13, 8)$  for the data, which shows that the performance of system would be almost the same as that of floating-point system in the following simulation. Besides, due to the identical whole wordlength compared with the  $\mathbf{H}$ , the same QRD hardware implementation can be used for both unimodular matrix  $\mathbf{T}^{-1}$  and channel matrix  $\mathbf{H}$ .

For the CLLL part, the fixed-point design is mainly referred to our former work in [12]. The fixed-point representation of some key parameters in CLLL are as follows: the integer bits for  $u$ ,  $\mathbf{T}$ , and internal datapath of Householder CORDIC are 11 bits, 9 bits, and 5 bits respectively; the fraction bits for both  $\mathbf{Q}$  and  $\mathbf{R}$  are 13 bits; the integer bits of  $\mathbf{R}$  after size reduction and basis updating are 5 bits at most.

When only applying the fixed-point design for the list MIMO detector under the above analysis in the LR-aided IDD receiver, its performances compared with the floating-point system under LLR clipping are depicted in the Fig. 10. The results show that the BER performance degradations of the fixed-point design for the list MIMO detector are kept less than 0.2 dB.

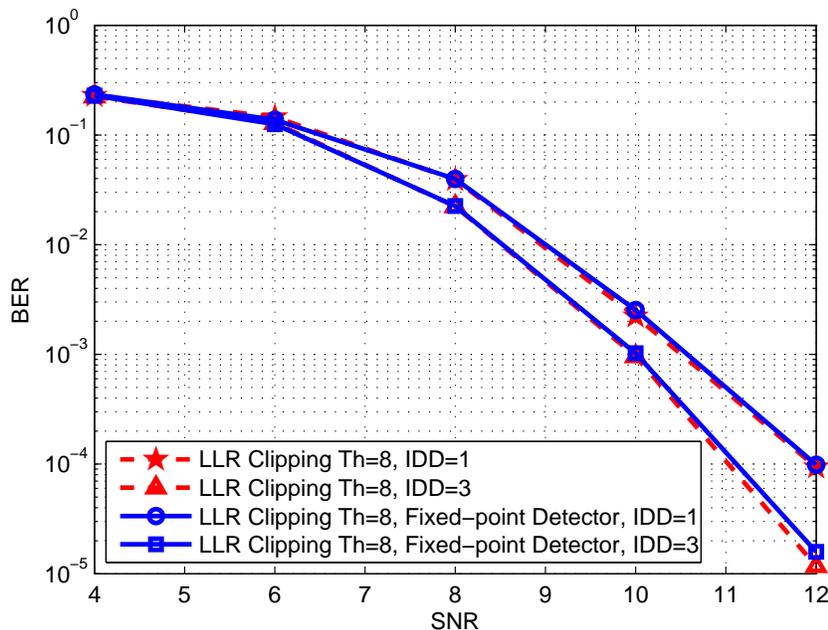


Fig. 10. BER performance under fixed-point MIMO detector in LR-aided IDD receiver

### C. Fixed-point design for the Turbo decoder

Fixed-point design for Turbo decoding has been well studied in the literature [19], [20], [21], and [22]. The most important parts of the fixed-point implementation for Turbo decoding are the BM, FW, and BW parts as shown in Section III-E. The fixed-point implementation in this paper is mainly based on the results in [22]. Here the bits width for the BM, FW, and BW we adopted are  $FP(5, 3)$ ,  $FP(7, 3)$ , and  $FP(7, 3)$ , respectively. And the bits width for both the extrinsic information and prior information is  $FP(5, 3)$ .

When only applying the fixed-point design for the Turbo decoder under the above analysis in the LR-aided IDD receiver, its performance differences compared with the floating-point system under LLR clipping are depicted in the Fig. 11. The results show that the BER performance degradations of the fixed-point design for the Turbo decoder are kept within 0.1 dB.

### D. Fixed-point performance of the whole LR-aided IDD receiver

Based on the above finite wordlength analysis for the MIMO detector and the Turbo decoder, and by adding the fixed-point design for the LLR information between detector and decoder,

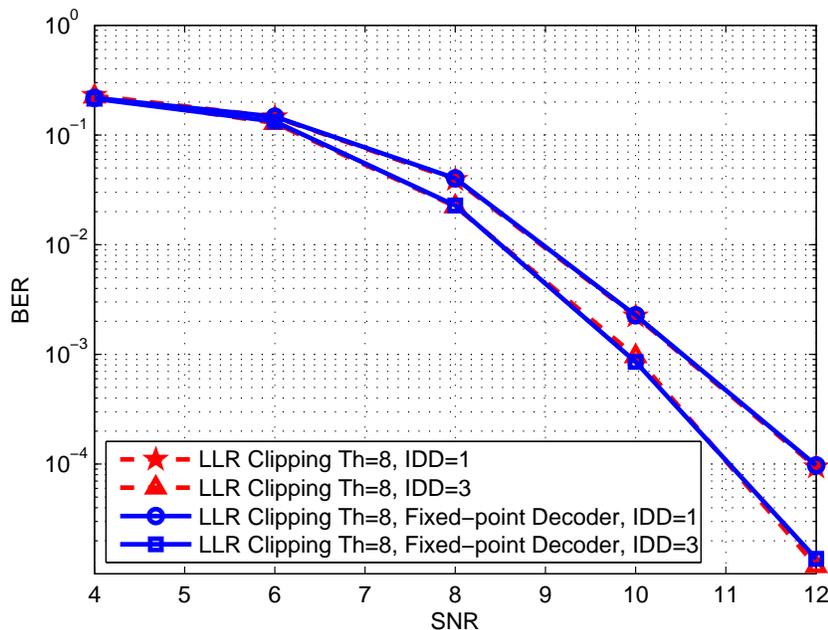


Fig. 11. BER performance under fixed-point Turbo decoder in LR-aided IDD receiver

we can get the fixed-point performance of the whole LR-aided IDD receiver for coded MIMO systems. Because the clipping threshold is set to 8 under the former simulation verification, the fixed-point design of the LLR values can adopt 4 bits integer wordlength with saturation operation. For the fraction wordlength, our simulations show that even 2 bits can keep a desirable performance, which could reduce the complexity in the VLSI implementation.

By using  $FP(4, 2)$  fixed-point representation for the LLR information between the MIMO detector and the decoder, and adding all fixed-point designs of the former analysis in the LR-aided IDD receiver, its performance differences compared with the floating-point system under LLR clipping are depicted in the Fig. 12. The results show that the BER performance degradation of the fixed-point design for the whole LR-aided IDD receiver is kept less than 0.2 dB.

## V. CONCLUSION

In this paper we have demonstrated fixed-point implementation for the whole LR-aided IDD receiver in the MIMO coded systems, which includes the fixed-point design for the key algorithms like CLLL, FCA, QRD, LLR clipping, and Turbo decoding. The results of the fixed-point system

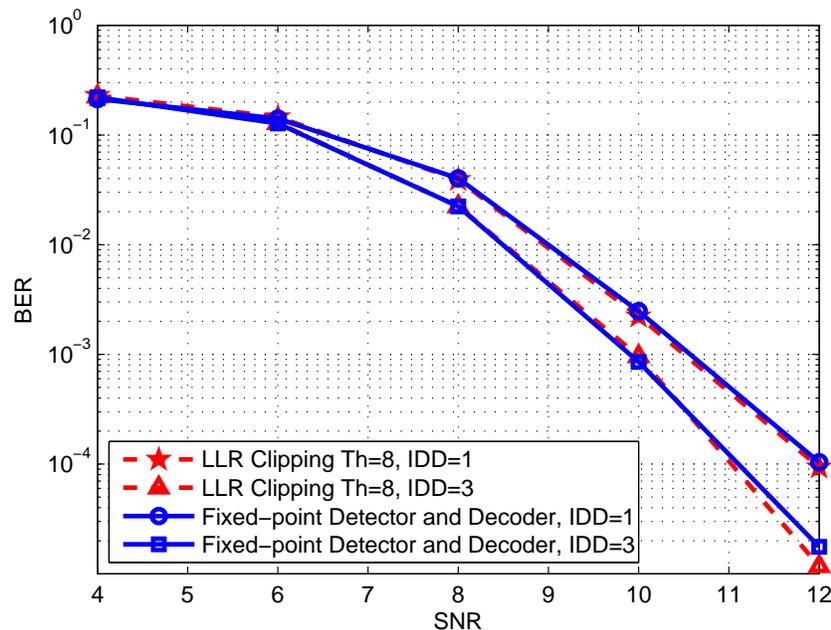


Fig. 12. BER performance under fixed-point LR-aided IDD receiver

show that its BER performance degradation is within 0.2dB compared with the floating-point system. With these results the hardware implementation of the LR-aided IDD receiver can be straightforwardly implemented in VLSI and FPGA, which is also our next work consideration.

## REFERENCES

- [1] L. Zheng and D. Tse, "Diversity and multiplexing: a fundamental tradeoff in multiple-antenna channels," *Information Theory, IEEE Transactions on*, vol. 49, no. 5, pp. 1073 – 1096, may 2003.
- [2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo-codes. 1," in *Communications, 1993. ICC 93. Geneva. Technical Program, Conference Record, IEEE International Conference on*, vol. 2, may 1993, pp. 1064 –1070 vol.2.
- [3] R. Gallager, "Low-density parity-check codes," *Information Theory, IRE Transactions on*, vol. 8, no. 1, pp. 21 –28, january 1962.
- [4] B. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *Communications, IEEE Transactions on*, vol. 51, no. 3, pp. 389 – 399, march 2003.
- [5] J. Hagenauer and C. Kuhn, "The list-sequential (liss) algorithm and its application," *Communications, IEEE Transactions on*, vol. 55, no. 5, pp. 918 –928, may 2007.
- [6] H. Yao and G. Wornell, "Lattice-reduction-aided detectors for mimo communication systems," in *Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE*, vol. 1, nov. 2002, pp. 424 – 428 vol.1.

- [7] D. Wubben, R. Bohnke, V. Kuhn, and K.-D. Kammeyer, "Near-maximum-likelihood detection of mimo systems using mmse-based lattice reduction," in *Communications, 2004 IEEE International Conference on*, vol. 2, june 2004, pp. 798 – 802 Vol.2.
- [8] X. Ma and W. Zhang, "Performance analysis for mimo systems with lattice-reduction aided linear equalization," *Communications, IEEE Transactions on*, vol. 56, no. 2, pp. 309 –318, february 2008.
- [9] M. Shabany and P. Glenn Gulak, "The application of lattice-reduction to the k-best algorithm for near-optimal mimo detection," in *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on*, may 2008, pp. 316 –319.
- [10] W. Zhang and X. Ma, "Low-complexity soft-output decoding with lattice-reduction-aided detectors," *Communications, IEEE Transactions on*, vol. 58, no. 9, pp. 2621 –2629, september 2010.
- [11] Y. H. Gan, C. Ling, and W. H. Mow, "Complex lattice reduction algorithm for low-complexity full-diversity mimo detection," *Signal Processing, IEEE Transactions on*, vol. 57, no. 7, pp. 2701 –2710, july 2009.
- [12] B. Gestner, W. Zhang, X. Ma, and D. Anderson, "Lattice reduction for mimo detection: From theoretical analysis to hardware realization," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 58, no. 4, pp. 813 –826, april 2011.
- [13] B. Hassibi and H. Vikalo, "On the sphere-decoding algorithm i. expected complexity," *Signal Processing, IEEE Transactions on*, vol. 53, no. 8, pp. 2806 – 2818, aug. 2005.
- [14] K. wai Wong, C. ying Tsui, R.-K. Cheng, and W. ho Mow, "A vlsi architecture of a k-best lattice decoding algorithm for mimo channels," in *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on*, vol. 3, 2002, pp. III–273 – III–276 vol.3.
- [15] A. Maltsev, V. Pestretsov, R. Maslennikov, and A. Khoryaev, "Triangular systolic array with reduced latency for qr-decomposition of complex matrices," in *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*, may 2006, p. 4 pp.
- [16] Y.-T. Hwang and W.-D. Chen, "A low complexity complex qr factorization design for signal detection in mimo ofdm systems," in *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on*, may 2008, pp. 932 –935.
- [17] P. Robertson, E. Villebrun, and P. Hoher, "A comparison of optimal and sub-optimal map decoding algorithms operating in the log domain," in *Communications, 1995. ICC '95 Seattle, 'Gateway to Globalization', 1995 IEEE International Conference on*, vol. 2, jun 1995, pp. 1009 –1013 vol.2.
- [18] M. Myllyla, J. Antikainen, M. Juntti, and J. Cavallaro, "The effect of llr clipping to the complexity of list sphere detector algorithms," in *Signals, Systems and Computers, 2007. ACSSC 2007. Conference Record of the Forty-First Asilomar Conference on*, nov. 2007, pp. 1559 –1563.
- [19] G. Montorsi and S. Benedetto, "Design of fixed-point iterative decoders for concatenated codes with interleavers," *Selected Areas in Communications, IEEE Journal on*, vol. 19, no. 5, pp. 871 –882, may 2001.
- [20] Y. Tong, T.-H. Yeap, and J.-Y. Chouinard, "Vhdl implementation of a turbo decoder with log-map-based iterative decoding," *Instrumentation and Measurement, IEEE Transactions on*, vol. 53, no. 4, pp. 1268 – 1278, aug. 2004.
- [21] M. Castellon, I. Fair, and D. Elliott, "Fixed-point turbo decoder implementation suitable for embedded applications," in *Electrical and Computer Engineering, 2005. Canadian Conference on*, may 2005, pp. 1065 –1068.
- [22] A. Morales-Cortes, R. Parra-Michel, L. Gonzalez-Perez, and T. Cervantes, "Finite precision analysis of the 3gpp standard turbo decoder for fixed-point implementation in fpga devices," in *Reconfigurable Computing and FPGAs, 2008. ReConFig '08. International Conference on*, dec. 2008, pp. 43 –48.