A Givens Rotation-based QR Decomposition for MIMO Systems

Wen Fan and Amir Alimohammad

***Abstract*—QR decomposition is an essential operation in various detection algorithms utilized in multiple-input multiple-output (MIMO) wireless communication systems. This paper presents a Givens rotation-based QR decomposition for 4×4 MIMO systems. Instead of performing QR decomposition by CORDIC algorithms, lookup table (LUT) compression algorithms are employed to rapidly evaluate the trigonometric functions. The proposed approach also provides greater accuracy compared to the CORDIC algorithms. QR decomposition is performed by complex Givens rotations cascaded with real Givens rotations. In complex Givens rotations, a modified triangular systolic array (TSA) is adopted to reduce the delay units of the design and hence, reducing the hardware complexity. The proposed QR decomposition algorithm is implemented in TSMC 90-nm CMOS technology. It achieves the throughput of 53.5 million QR decompositions per second (MQRD/s) when operating at 214 MHz.**

***Index Terms*—QR decomposition, Givens rotation, lookup table compression, MIMO detection.**

I. Introduction

MULTIPLE-input multiple-output (MIMO) technique has attracted significant interest due to the substantial increment of system capacity and spectral efficiency. As a reliable technology supporting high throughput data transmission, MIMO has been widely adopted by recent wireless communication standards, such as IEEE 802.11n, IEEE 802.16m (WiMAX) and 3GPP-LTE [1], [2]. One of the challenges for MIMO systems is to design a high-throughput and accurate detector for the receiver. QR decomposition is an essential operation to convert a MIMO channel into multiple layered sub-channels and thus, reduce the computational complexity of MIMO detection. The accuracy of QR decomposition will directly impact the bit error rate (BER) performance and symbol detection throughput of MIMO systems significantly.

Three QR factorization methods have been widely used in MIMO systems: Gram-Schmidt, Householder transformation, and Givens rotation [3]–[14]. Since the complexity of Householder transformation is relatively high, related studies on the QR decomposition architectures are typically classified into two main categories. One is based on the modified Gram-Schmidt algorithm (MGS) [3]–[8], which performs QR decomposition in parallel and requires many norm and division operations. The other category is based on Givens rotation and utilizing triangular systolic array architecture [9]–[14], which implements the rotation operation by the coordinate rotation digital computer (CORDIC) algorithms. Compared to MGS, Givens rotation has the advantage of lower hardware complexity, however, the long latency is the main obstacle of the Givens rotation approach.

CORDIC algorithms are commonly used to implement Givens rotation-based QR decomposition for their low hardware complexity. However, the number of iterations will be large if the system requires high accuracy, which leads to a relatively long latency. This article intends to utilize alternative algorithms with greater computational accuracy than CORDIC algorithms for Givens rotation-based QR decomposition. After comparing the mean square error (MSE) performance and arithmetic complexity of the lookup table (LUT) compression, linear approximation, and CORDIC algorithms, LUT compression scheme is selected to implement trigonometric functions in Givens rotation as it has lower computational complexity than the linear approximation technique and also provides improved MSE performance compared to the CORDIC-based solution.

To reduce the hardware complexity, the proposed QR decomposition approach consists of a complex-valued decomposition (CVD) followed by a real-valued decomposition (RVD). The RVD is designed with the triangular systolic array (TSA) architecture and the CVD is implemented using the modified TSA architecture. Compared with the conventional architectures, the proposed scheme reduces the number of delay units and also shortens the latency of the design.

The rest of this article is organized as follows. In Section II, the MIMO system model and the important role of QR decomposition in detection algorithms are briefly reviewed. Section III describes the complex-valued and real-valued decomposition algorithms in the proposed design. Implementing trigonometric functions using our proposed LUT compression scheme is also presented and compared with the linear approximation and CORDIC-based realizations. In Section IV, the architectures of complex Givens rotation, real Givens rotation, and the divider used in the arctangent function are presented and discussed. The implementation results and comparisons are provided in Section V. Finally, Section VI makes some concluding remarks.

II. MIMO System Model

Consider a spatial multiplexing MIMO system [15] with N_T transmit and N_R receive antennas. The equivalent baseband model of the channel can be described by a complex-valued $N_R \times N_T$ matrix $\mathbf{H}$. The relation between transmit and receive signal vectors is can be written as:

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}, \tag{1}$$

where $\mathbf{y} = [y_1, y_2, ..., y_{N_R}]^T$ denotes the N_R-dimensional receive signal vector, $\mathbf{s} = [s_1, s_2, ..., s_{N_T}]^T$ is the $N_T \times 1$ transmit signal vector, and $\mathbf{n} = [n_1, n_2, ..., n_{N_R}]^T$ denotes the

$N_R \times 1$ noise vector with independent identically-distributed (i.i.d) zero-mean Gaussian noise variates [16].

QR decomposition is an essential preprocessing unit in various MIMO detection techniques, such as zero-forcing, sphere decoding, and K-best detection algorithms [17]–[20]. By using QR decomposition of the channel matrix $\mathbf{H} = \mathbf{QR}$, where $\mathbf{Q}$ is a unitary matrix and $\mathbf{R}$ is an upper triangular matrix, the detected signal vector $\hat{\mathbf{s}}$ can be expressed as follows:

$$\begin{aligned} \hat{\mathbf{s}} &= \arg\min \|\mathbf{y} - \mathbf{Hs}\|^2 \\ &= \arg\min \|\mathbf{Q}^H \mathbf{y} - \mathbf{Rs}\|^2. \end{aligned} \tag{2}$$

In efficient MIMO detection schemes, such as the K-best algorithm, sorting of the expanded traversal paths is an important step. If the CVD system model is chosen, additional multiplication and addition will be necessary before sorting can take place. In order to obtain the best candidates without a complicated sorting step, many MIMO detection schemes utilize the RVD system model [18], in which (1) can be expressed as:

$$\begin{aligned} \begin{bmatrix} \mathrm{Re}\{\mathbf{y}\} \\ \mathrm{Im}\{\mathbf{y}\} \end{bmatrix} &= \begin{bmatrix} \mathrm{Re}\{\mathbf{H}\} & -\mathrm{Im}\{\mathbf{H}\} \\ \mathrm{Im}\{\mathbf{H}\} & \mathrm{Re}\{\mathbf{H}\} \end{bmatrix} \begin{bmatrix} \mathrm{Re}\{\mathbf{s}\} \\ \mathrm{Im}\{\mathbf{s}\} \end{bmatrix} \\ &+ \begin{bmatrix} \mathrm{Re}\{\mathbf{n}\} \\ \mathrm{Im}\{\mathbf{n}\} \end{bmatrix}, \end{aligned} \tag{3}$$

where $\mathrm{Re}\{\cdot\}$ and $\mathrm{Im}\{\cdot\}$ denote the real and imaginary parts, respectively. In this case, the dimension of the real-valued channel matrix becomes $2N_R \times 2N_T$. For the MIMO systems with a relatively large number of transmit and receive antennas (e.g., 4×4 or more), it is shown in [11] that the direct RVD of the channel matrix will be more complicated than the CVD. However, MIMO detection will be computationally-intensive without RVD, especially for high order modulation schemes. One efficient approach is to first perform the CVD of the channel matrix and then perform the RVD of the complex triangular matrix $\mathbf{R}$ [11]. This approach is applied in the proposed QR decomposition scheme and is discussed in the following section.

III. Proposed QR Decomposition Algorithm

A. Complex-Valued Decomposition

Givens rotation technique zeros one element of a matrix at a time by applying a two-dimensional rotation. Therefore, rotation matrix plays an important role on the performance of QR decomposition. The idea of CVD-based Givens rotation can be illustrated using the polar representation. Consider a 4×4 complex-valued matrix

$$\mathbf{H} = \begin{bmatrix} |h_{11}|e^{j\theta_{11}} & h_{12} & h_{13} & h_{14} \\ |h_{21}|e^{j\theta_{21}} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \\ h_{41} & h_{42} & h_{43} & h_{44} \end{bmatrix}, \tag{4}$$

where $|h_{i1}|$ and θ_{i1} (i = 1,2) represent the magnitude and the angle of the matrix entries, respectively, and $j = \sqrt{-1}$. The Givens rotation matrix $\mathbf{G}_1$ targets at eliminating h_{21} by h_{11} and can be expressed as:

$$\mathbf{G}_1 = \begin{bmatrix} c & s & 0 & 0 \\ -s^* & c^* & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5}$$

The complex triangular matrix $\mathbf{R}$ and the complex unitary matrix $\mathbf{Q}$ can be obtained as:

$$\begin{aligned} \mathbf{R} &= \mathbf{G}_6\mathbf{G}_5\mathbf{G}_4\mathbf{G}_3\mathbf{G}_2\mathbf{G}_1\mathbf{H}, \\ \mathbf{Q} &= (\mathbf{G}_6\mathbf{G}_5\mathbf{G}_4\mathbf{G}_3\mathbf{G}_2\mathbf{G}_1)^H, \end{aligned} \tag{6}$$

where $\mathbf{G}_2, \ldots, \mathbf{G}_6$ are rotation matrices to zero h_{31}, h_{41}, h_{32}, h_{42} and h_{43}, respectively, and $(\cdot)^H$ denotes the Hermitian of a complex matrix. The c and s parameters can be calculated using the three-angle complex rotation (Three-ACR) technique [11], [21], where:

$$\begin{aligned} c &= \frac{h_{11}^*}{\sqrt{|h_{11}|^2 + |h_{21}|^2}} = \cos\theta_a e^{-j\theta_{11}}, \\ s &= \frac{h_{21}^*}{\sqrt{|h_{11}|^2 + |h_{21}|^2}} = \sin\theta_a e^{-j\theta_{21}}, \\ \theta_a &= \arctan\left(\frac{|h_{21}|}{|h_{11}|}\right). \end{aligned} \tag{7}$$

In order to avoid the square root operation in calculating θ_a, (7) can be further optimized as:

$$\theta_a = \arctan\left|\frac{\mathrm{Re}\{h_{21}\}}{\cos\theta_{21}} \times \frac{\cos\theta_{11}}{\mathrm{Re}\{h_{11}\}}\right|. \tag{8}$$

After rotation, the rotated matrix $\mathbf{G}_1\mathbf{H}$ becomes

$$\mathbf{G}_1\mathbf{H} = \begin{bmatrix} h_{11}^{(1)} & h_{12}^{(1)} & h_{13}^{(1)} & h_{14}^{(1)} \\ 0 & h_{22}^{(1)} & h_{23}^{(1)} & h_{24}^{(1)} \\ h_{31} & h_{32} & h_{33} & h_{34} \\ h_{41} & h_{42} & h_{43} & h_{44} \end{bmatrix}, \tag{9}$$

where $h_{ij}^{(k)}$ denotes the h_{ij} entry of the matrix after k-th rotation. An important feature of the Three-ACR technique is that it causes the triangular matrix $\mathbf{R}$ to have only real diagonal elements. Therefore, an additional rotation step is required to remove the imaginary part of $h_{22}^{(1)}$ in $\mathbf{G}_1\mathbf{H}$. The additional rotation matrix can be written as:

$$\mathbf{G}_1' = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{-j\theta_{22}^{(1)}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{10}$$

Another approach to calculate the c and s parameters is the two-angle complex rotation (Two-ACR) [21], where

$$\begin{aligned} c &= \cos\theta_a, \\ s &= \sin\theta_a e^{j\theta_b}, \\ \theta_b &= \theta_{11} - \theta_{21}. \end{aligned} \tag{11}$$

The Three-ACR technique based on the architecture of TSA reduces the latency and area by using the same hardware resources of the CORDIC modules, but the throughput will be lower than that of the Two-ACR. Three-ACR saves four

TABLE I
ARITHMETIC COMPLEXITY OF THE THREE-ANGLE COMPLEX ROTATION AND TWO-ANGLE COMPLEX ROTATION SCHEMES

	Two-ACR		Three-ACR	
$N_T \times N_R$	2×2	4×4	2×2	4×4
Multiplication	18	54	26	74
Addition	6	24	12	42
Arctangent	3		4	
Sine	3		4	
Cosine	3		4	

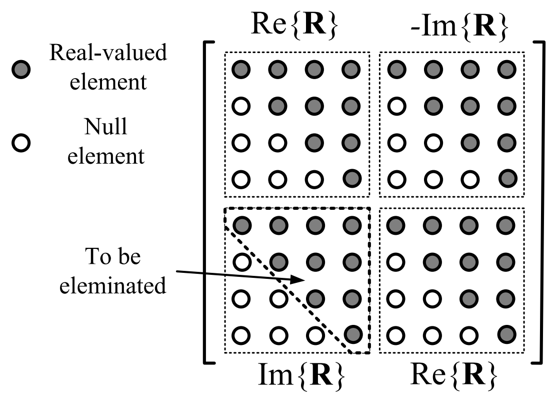

Fig. 1. Real-valued decomposition matrix $\tilde{\mathbf{R}}$ for a 4×4 MIMO system.

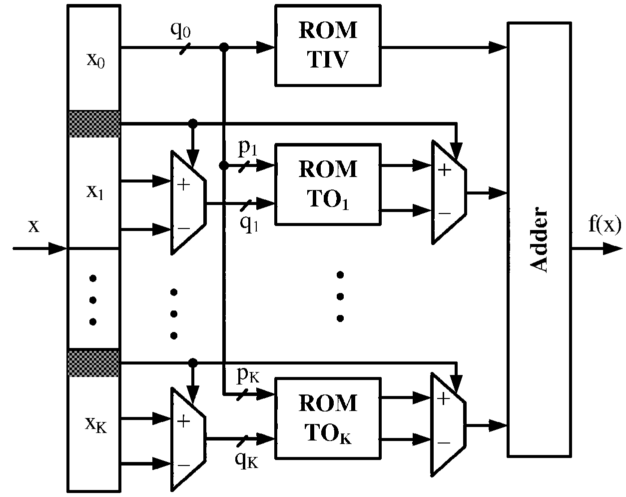

Fig. 2. Architecture of the LUT compression algorithm.

rotation stages in the RVD section compared to Two-ACR as it makes the complex-valued triangular matrix $\mathbf{R}$ have only real-valued diagonal elements. However, Three-ACR adds six additional rotations in the CVD section according to (6) and (10), which implies a longer latency and a larger area. Furthermore, both parameters c and s calculated with the Three-ACR technique are complex, while the parameter c calculated with the Two-ACR approach is real and therefore, the Two-ACR has lower arithmetic complexity than the Three-ACR technique for the LUT compression-based designs.

Table I shows the arithmetic complexity of the Three-ACR and Two-ACR approaches. As shown in Table I, the Two-ACR requires fewer operations than Three-ACR. For 4×4 MIMO systems or systems with more antennas, the complexity of the Two-ACR in (11) does not increases as much as that of Three-ACR. Therefore, we use the Two-ACR scheme to compute c and s parameters in the rotation matrices.

B. Real-Valued Decomposition

Fig. 1. shows the real-valued matrix $\tilde{\mathbf{R}}$, which is expanded from the complex-valued matrix $\mathbf{R}$. The task for RVD is to eliminate the elements in the lower-left section of the matrix $\tilde{\mathbf{R}}$, as shown by the dashed lines in Fig. 1. To guarantee the minimum number of operations, the processing sequence is important [11]. For example, for 4×4 matrices, the elements of $\tilde{\mathbf{R}}$ should be nulled in $\tilde{\mathbf{R}}_{51}$, $\tilde{\mathbf{R}}_{62}$, $\tilde{\mathbf{R}}_{73}$, $\tilde{\mathbf{R}}_{84}$, $\tilde{\mathbf{R}}_{52}$, $\tilde{\mathbf{R}}_{63}$, $\tilde{\mathbf{R}}_{74}$, $\tilde{\mathbf{R}}_{53}$, $\tilde{\mathbf{R}}_{64}$, $\tilde{\mathbf{R}}_{54}$ order. The RVD of $\mathbf{R}$ and $\mathbf{Q}$ can be written as:

$$\begin{aligned} \mathbf{R}_{RVD} &= \tilde{\mathbf{G}}\tilde{\mathbf{R}}, \\ \mathbf{Q}_{RVD} &= \tilde{\mathbf{G}}^H\tilde{\mathbf{Q}}, \end{aligned} \quad (12)$$

where $\tilde{\mathbf{G}}$ is the real-valued Givens rotation matrix and $\tilde{\mathbf{Q}}$ is the real-valued $2N_R \times 2N_T$ matrix expanded from $\mathbf{Q}$.

C. Lookup Table Compression Algorithm

In the design of Givens rotation-based QR decomposition, the chosen vector rotation technique has a direct impact on the throughput and the hardware complexity of the design. CORDIC technique has been extensively applied in the Givens rotation-based QR decomposition algorithms [9]–[14], [22]. CORDIC has the advantage of implementing vector rotations using only adders and shifters, however, due to the iterative nature of CORDIC algorithms, it is challenging to achieve high throughputs and high accuracies. The LUT compression and linear approximation approaches are significantly more accurate than the CORDIC technique when the same word-lengths are utilized. This will be briefly discussed here.

The LUT compression technique is an effective approach for approximating trigonometric functions utilizing very small read-only memories (ROMs) and simple arithmetic circuitries. The architecture of the LUT compression method is shown in Fig. 2. The principle of the LUT compression technique is to decompose the input signal $x \in [0, 1)$ into $K + 1$ non-overlapping sub-words, $x_0, x_1, \ldots, x_K$, each with $q_0, q_1, \ldots,$ and q_K bits, respectively. The interval [0,1) of x has been divided into 2^{q_0} subintervals. x_0 represents the starting point of each subinterval and $x_1 + \ldots + x_K$ is the offset in each interval between x and x_0. A piecewise linear approximation of $f(x)$ can be expressed as:

$$\begin{aligned} f(x) &= f(x_0 + x_1 + \cdots + x_K) \\ &\approx A(x_0) + B(x_0)(x_1 + \cdots + x_K) \\ &= A(x_0) + B(x_0)x_1 + \cdots + B(x_0)x_K, \end{aligned} \quad (13)$$

The term $B(x_0)x_1$ can be approximated as $B_1(\alpha_1)x_1$, where α_1 is the sub-word of x_0 including its $p_1 \leq q_0$ most significant bits (MSBs). Likewise, the term $B(x_0)x_2$ is approximated as $B_2(\alpha_2)x_2$, where α_2 is a sub-word of x_0 including its $p_2 \leq p_1$ MSBs. Similar approximations can be conducted on the

TABLE II
ARITHMETIC COMPLEXITY OF DIFFERENT ALGORITHMS FOR THREE TRIGONOMETRIC FUNCTIONS

	LUT compression		Linear approximation		CORDIC	
Functions	sin/cos	$arctan$	sin/cos	$arctan$	sin/cos	$arctan$
WL.WF	$I: 12.9$ $O: 11.9$	$I: 14.9$ $O: 12.9$	$I: 12.9$ $O: 11.9$	$I: 14.9$ $O: 12.9$	$I: 12.9$ $O: 11.9$	$I: 14.9$ $O: 12.9$
Parameters	$q_0 = 6$ $q_1 = 3$ $p_1 = 2$	$q_0 = 6$ $q_1 = 3$ $p_1 = 2$	$s = 128$	$s = 128$	$n = 9$	$n = 9$
ROM size (bits)	464	464	2304	2304	117	117
Mul.	-	-	1	1	-	-
Add.	1	2	1	1	12	24
Div.	-	1	-	1	-	-
Shifter	-	-	-	-	8	16

$B(x_0)x_i$, $i = 3, \ldots, K$, terms. Therefore, the expression (13) can be approximated as:

$$f(x) \approx A(x_0) + B_1(\alpha_1)x_1 + \cdots + B_K(\alpha_K)x_K \qquad (14)$$

where $A(x_0)$ can be realized with a ROM with 2^{q_0} entries, and is called table of initial values (TIV). $B(\alpha_i)x_i$ can be implemented with K ROMs with $2^{p_i+q_i}$ entries each, which is called table of offsets (TO) [23]. The three trigonometric functions, sine, cosine, and arctangent, can be computed by conditionally adding or subtracting the values in TOs from the selected values in TIV.

The size of ROMs can be reduced to half by making the TOs symmetric. Then $f(x)$ in (14) can be written as:

$$\begin{aligned} f(x) \approx \tilde{A}(x_0) \quad &+ \quad B_1(\alpha_1)(x_1 - \frac{\delta_1}{2}) + \cdots \\ &+ \quad B_K(\alpha_K)(x_K - \frac{\delta_K}{2}), \end{aligned}$$

where δ_i is the weight of the least significant bit (LSB) of the i-th sub-word, and can be written as:

$$\delta_i = (2^{q_i} - 1)2^{-\varepsilon_i},$$

where

$$\varepsilon_i = \sum_{j=0}^{i} q_j.$$

The coefficients stored in TIV and TOs can then be calculated by minimizing the maximum approximation error as [23]:

$$\tilde{A}(x_0) = \frac{f(x_0) + f(x_0 + \Delta_0)}{2},$$

$$B_i(\alpha_i) = \frac{f(\alpha_i + \delta_i) - f(\alpha_i) + f(\alpha_i + \delta_i + \sigma_i) - f(\alpha_i + \sigma_i)}{2\delta_i},$$

where $\tilde{A}(x_0)$ is the content of the TIV and

$$\begin{aligned} \sigma_i \quad &= \quad 2^{-p_i} - 2^{q_i - \varepsilon_i}, \\ \Delta_0 \quad &= \quad \sum_{j=1}^{K} \delta_j. \end{aligned}$$

The content of the i-th TO, $i = 1, \ldots, K$, can be computed as:

$$TO_i(\alpha_i, x_i) = B_i(\alpha_i)(x_i + 2^{-\varepsilon_i - 1}).$$

Linear approximation is another approach to evaluate various trigonometric functions accurately. It is based on the idea of segmenting the interval $[0, 1)$ of the input signal x into $s = 2^u$ sub-intervals. The u MSBs of x encode the segment starting point x_k and are used to address the LUTs that store the linear function coefficients. The remaining bits of x represent the offset $x - x_k$ [24]. The linear approximation of a trigonometric function can be calculated as:

$$f(x) = n_k + m_k(x - x_k), \ k = 1, 2, \ldots, s, \qquad (15)$$

where n_k and m_k are constants and linear coefficients, respectively, $x_k \leq x < x_{k+1}$, $x_1 = 0$, and $x_{s+1} = 1$. The constant coefficients can be calculated by setting $x = x_k$ and the linear coefficients can be obtained by setting $x = (x_k + x_{k+1})/2$, where $x_k = (k-1)/s$. They are expressed as:

$$\begin{aligned} n_k \quad &= \quad f(\frac{\pi}{4}x_k), \\ m_k \quad &= \quad 2s\left[f(\frac{\pi}{4} \times \frac{2k-1}{2s}) - n_k\right]. \end{aligned} \qquad (16)$$

Both LUT compression and linear approximation techniques can be employed to evaluate the arctangent, sine, and cosine functions in the QR decomposition algorithm. The three trigonometric functions can be expressed as:

$$\begin{aligned} f_{arctan}/(\frac{\pi}{4}) \quad &= \quad (\arctan y)/(\frac{\pi}{4}) = a, \\ f_{sin} \quad &= \quad \sin(\frac{\pi}{4}a), \\ f_{cos} \quad &= \quad \cos(\frac{\pi}{4}a), \end{aligned} \qquad (17)$$

where $a \in [0, 8)$ and $y \in (0, 1)$. Note that only $arctan$, sin and cos values are required to be calculated with the rotation angles within $[0, \pi/4)$. If the rotation angles are within $[\pi/4, 2\pi)$, the trigonometric function values can be obtained by optionally performing input/output complements and the output swap operations. For example, if the rotation angles are within $[\pi/4, \pi/2]$, then $a \in [1, 2)$ and the three trigonometric

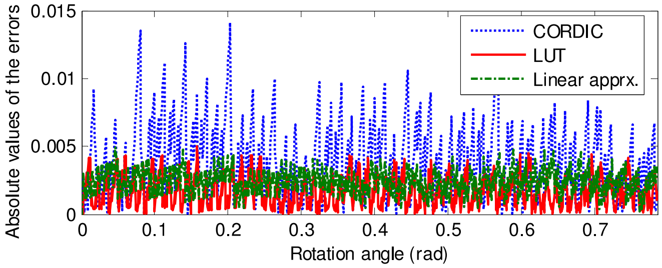

Fig. 3. Sine function errors of the three approaches with the rotation angle within $(0, \pi/4)$.

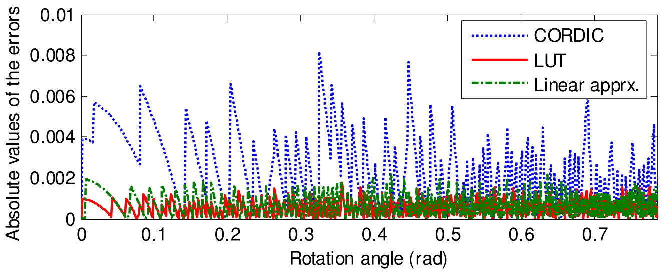

Fig. 4. Cosine function errors of the three approaches with the rotation angle within $(0, \pi/4)$.

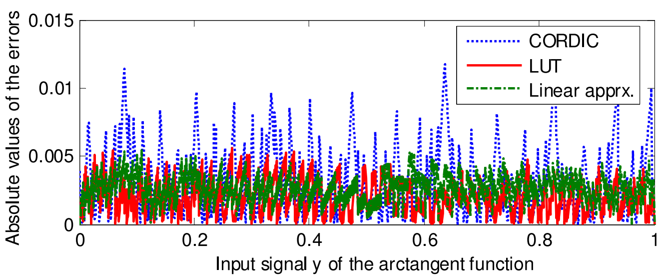

Fig. 5. Arctangent function errors of the three approaches with the input within $(0, 1)$.

functions can be expressed as:

$$
\begin{aligned}
f_{arctan}/(\frac{\pi}{4}) &= (\frac{\pi}{2} - \arctan\frac{1}{y})/(\frac{\pi}{4}) = 2 - a, \\
f_{sin} &= \cos[\frac{\pi}{4}(2-a)], \\
f_{cos} &= \sin[\frac{\pi}{4}(2-a)]. \qquad (18)
\end{aligned}
$$

By scaling the rotation angle's range from $[0, 2\pi)$ to $[0, 8)$ only 3 MSBs of a are required to control the octants instead of using the whole word-length of the angle. For QR decomposition, the result of the arctangent function will be the input signal to the sine or cosine functions and hence, it is not necessary to perform scaling by hardware. The scaling can be done in software when computing the coefficients stored in LUTs. Therefore, the hardware cost is reduced by scaling the range of the rotation angle.

Table II shows the arithmetic complexity of the LUT compression, linear approximation, and CORDIC approaches when implementing trigonometric functions in fixed-point format. The word-length (WL) and fraction length (WF) of the inputs and outputs of the trigonometric functions are also listed in Table II. The parameters of the LUT compression

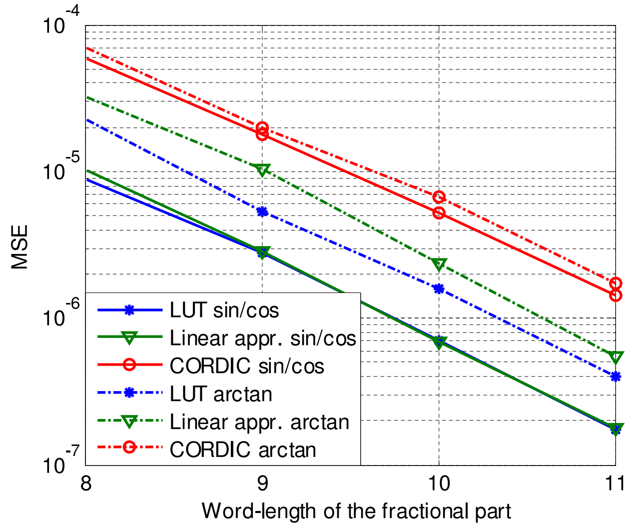

Fig. 6. MSE performances versus fractional word-lengths for three different approaches.

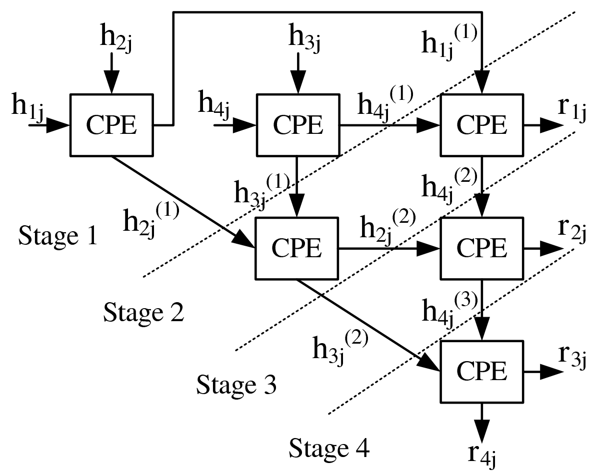

Fig. 7. Proposed TSA architecture for the complex-valued Givens rotation.

and the linear approximation approaches are set such that their mean square errors (MSEs) are relatively close. The MSEs of the CORDIC approaches are significantly larger than those of the LUT compression and the linear approximation schemes, independent of the parameters' values. If the WF is fixed, even increasing the number of iterations to more than nine, the MSE performance of CORDIC will not be improved because after nine iterations, there will not be any rotations. Therefore, the number of iterations in CORDIC algorithms is set to nine. According to the results in Table II, the LUT compression method requires considerably smaller ROMs than the linear approximation and without requiring any multipliers. Although it uses one more adder to implement the arctangent function compared to the linear approximation scheme, the overall arithmetic complexity of the LUT compression technique is significantly lower than that of the linear approximation method. CORDIC utilizes smaller ROMs than those in the LUT compression scheme and it does not need dividers, however, it uses many adders and shifters.

Figs. 3, 4, and 5 show the absolute values of the approx-

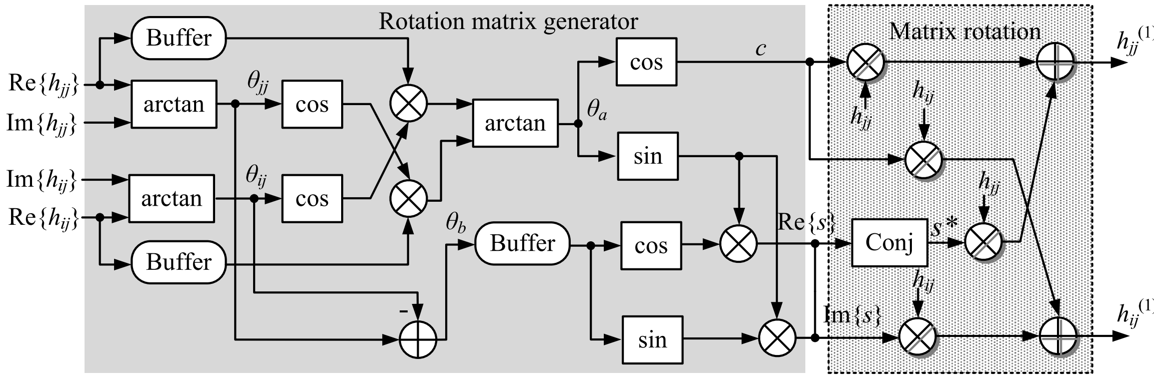

Fig. 8. Architecture of the complex processing element (CPE).

imation errors for three trigonometric functions using three different discussed schemes. For sine and cosine functions, the rotation angles are within $(0, \pi/4)$ and for arctangent function, the input signal is within $(0, 1)$, which means the output angle of the arctangent function is within the $(0, \pi/4)$ interval. Figs. 3, 4, and 5 demonstrate that the approximation errors of the LUT compression and linear approximation techniques are considerably smaller than the approximation errors of the CORDIC scheme. The MSE performance of the three approaches with various fractional bit-widths is shown in Fig. 6. For the sine, cosine and arctangent functions, the CORDIC algorithm has the worst MSE performance among the three. For sine and cosine functions, the MSE performance of the LUT compression and linear approximation schemes are close. However, LUT compression provides the best MSE performance for implementing the arctangent function. Considering the arithmetic complexity and the MSE performance, the LUT compression approach is chosen for implementing the three trigonometric functions in the proposed QR decomposition.

IV. Architecture Design of QR Decomposition

We utilize our proposed QR decomposition architecture in a 4×4 MIMO system. In order to achieve high throughputs, one column of the channel matrix **H** is processed in every clock cycle and therefore, the proposed architecture performs QR decomposition of a 4×4 matrix over four clock cycles. The required projection of $\mathbf{Q}^H \mathbf{y}$ in MIMO detection algorithms is then generated in one additional clock cycle.

A. Architecture of the Complex-Valued Givens Rotation

The conventional TSA architecture has been previously used for QR decomposition [14], [21]. The TSA architecture was improved in [11] by reducing the delay units, which is also adopted in our design for complex-valued Givens rotation, as shown in Fig. 7. It performs complex-valued Givens rotation in four stages, while conventional TSA architecture requires five stages. Each complex processing element (CPE) zeros one entry in the left-bottom triangular section of the **H** matrix. The outputs r_{1j}, r_{1j}, r_{3j} and r_{4j} are four rows of complex-valued triangular matrix **R**. In the first stage, h_{21} and h_{31} are zeroed by h_{11} and h_{41}, respectively, and four rows of the matrix are rotated by the rotation matrix accordingly, expressed as $h_{ij}^{(1)}$

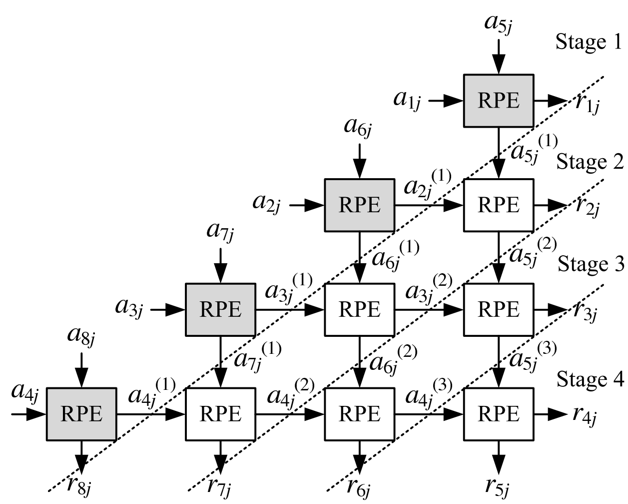

Fig. 9. Proposed architecture for the real-valued Givens rotation.

$(i, j = 1, 2, 3, 4)$. In the second stage, $h_{41}^{(1)}$ and $h_{32}^{(1)}$ are zeroed by $h_{11}^{(1)}$ and $h_{22}^{(1)}$. In the third stage, $h_{42}^{(2)}$ is zeroed by $h_{22}^{(2)}$, and $h_{43}^{(3)}$ is zeroed by $h_{33}^{(3)}$ in the final stage.

The detailed architecture of the CPE is shown in Fig. 8. It executes the complex-valued Givens rotation in two steps. In the first step, the rotation matrix is built by calculating the c and s parameters according to equations (5), (8) and (11), where $c = \cos\theta_a$ and $s = \sin\theta_a e^{j\theta_b}$. The critical operations for calculating the rotation matrix are arctangent, cosine and sine functions. These operations are implemented with the LUT compression method using the architecture shown in Fig. 2, where we choose $q_0 = 6$, $q = 3$, and $p = 2$. In the second step, the channel matrix **H** is multiplied by the generated rotation matrix **G** to zero one of its element. Since c is real-valued, the multipliers and adders in this part are half-complex (the ones with shadows in Fig. 8), which have lower complexity than the complex-valued multipliers and adders. While the traditional complex multipliers consist of four real multipliers and two adders, the half-complex multipliers in Fig. 8. use two real multipliers and no adders.

B. Architecture of the Real-Valued Givens Rotation

The architecture of the real-valued Givens rotation is shown in Fig. 9, where a_{ij} and r_{ij}, $i = 1, \ldots, 8$, are the entries of

TABLE III
IMPLEMENTATION RESULTS OF DIFFERENT QR DECOMPOSITION TECHNIQUES

	[3]	[4]	[6]	[9]	[11]	[25]	[26]	This work
Algorithm	MGS	Interpolation-based	MGS	GR and Householder	GR	GR	GR	GR
Matrix dimension	4×4 Complex	$2 \times 2 \sim 4 \times 4$ Complex	4×4 Complex	4×4 Complex	8×8 Real	4×4 Complex	4×4 Complex	8×8 Real
Technology	0.18-μm	90-nm	0.18-μm	0.13-μm	0.18-μm	90-nm	90-nm	90-nm
Max. freq. (MHz)	400	140	162	278	100	125	116	214
Processing cycles	35	4	104	40	4	4	4	4
Gate count (K)	32.6	318	61.8	36	152	115	437.5	262
Throughput (MQRD/s)	11.4	35	1.6	7	25	31.4	29	53.5
Normalized throughput [1]	22.8	35	3.2	10	50	31.4	29	53.5
Gate efficiency [2]	0.699	0.110	0.052	0.269	0.329	0.273	0.066	0.20

[1] Throughput $\times$ (Technology/90-nm)/$(64/n^2)$, where $n = 2N$ for $N \times N$ complex matrices and $n = N$ for real matrices.
[2] Normalized throughput / Gate count.

real-valued matrix $\tilde{\mathbf{R}}$ for RVD and the triangular matrix $\mathbf{R}_{RVD}$ after RVD, respectively. The real processing element (RPE) has a similar, but simpler architecture to that of a CPE. It only uses one arctangent, one cosine, and one sine function to build the rotation matrix and uses four real multipliers and two real adders to implement the matrix rotation operation. The entries in the lower left corner of the 8×8 matrix $\tilde{\mathbf{R}}$, built by the CVD of $\mathbf{R}$, are zeroed in four stages. In the first stage, only the left four columns of the matrix are required to be computed. The four columns on the right of the matrix can be derived by the left part according to the relationship in Fig. 1, which can be expressed as $r_{i,j} = -r_{i+4,j-4}, i = 1, \ldots, 4, j = 5, \ldots, 8$, and $r_{i,j} = r_{i-4,j-4}, i, j = 5, \ldots, 8$.

C. Datapath of the Divider

Divider is an important unit in the arctangent function, which directly impacts the latency and area of the system significantly. Higher-radix dividers compute more bits per iteration, so they are faster, at the expense of a larger area. Lower-radix dividers can be designed with smaller area, but they are slow as they compute fewer bits per iteration. In the proposed design, division is performed in radix-4, which computes two bits per iteration for a moderate speed and its relatively small area utilization [27].

The divider is designed with five pipeline stages, as shown in Fig. 10. Each stage processes two iterations. d, x, w and Q are divisor, dividend, remainder and quotient, respectively, and $q_j, j = 1, 2, ..., 6$, denotes the redundant digit set of $\{-2, -1, 0, 1, 2\}$. Normalizer is used to scale d into the range of [0.5, 1). By setting the initial value of the residual to $x/4$, the final quotient will be four times of the obtained quotient. Four MSBs of the normalized divisor are used to select the boundary of the digit set. Quotient digit selection is implemented with LUTs. The next residual will be calculated as $w_{j+1} = 4w_j - q_{j+1} \times d$. $4w_j$ can be implemented by left-shifting two bits of w_j. The logic between the second and the fifth pipeline stage is not shown in Fig. 10 as is the same logic used between $Reg1$ and $Reg2$. The final quotient can be derived by an on-the-fly conversion of the q_j [27].

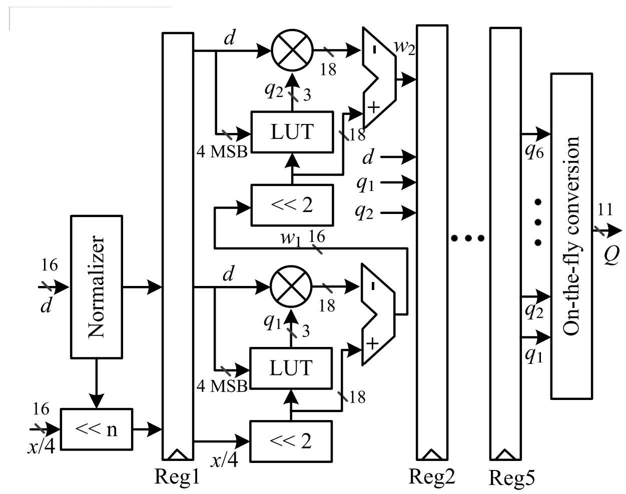

Fig. 10. Datapath of the radix-4 divider.

V. IMPLEMENTATION RESULTS AND COMPARISONS

The proposed Givens rotation-based QR decomposition architecture is designed, simulated in floating-point and fixed-point representations, implemented and verified. The word-lengths of the signals in the QR decomposition architecture are chosen based on the fixed-point simulation results of different MIMO detection algorithms utilizing various modulation schemes. Assume that the channel matrix $\mathbf{H}$ is perfectly known at the receiver. For i.i.d. channels with additive White Gaussian noise (AWGN) model at the receiver, five bits are allocated to the integer part of the channel matrix entries. The fixed-point simulations are performed for different fractional bit-widths of the variables, which impacts the accuracy of the results.

The BER performance of a 64-QAM modulated 4×4 MIMO system using the K-best detection algorithm (K = 10) and for different word-lengths are shown in Fig. 11. According to the simulation results, the BER performance starts to degrade significantly with 8 bits fraction at 20 dB signal-to-noise ratio (SNR). Fig. 12 shows the symbol error rate (SER) performance of a 16-QAM modulated 4×4 MIMO system with zero-forcing (ZF) detection. In this case, the SER performance does not degrade much as long as 9 bits are used for the fractional part of the signals. Therefore, the QR decomposition can be reliably designed with 14 bits (5 bits for the integer part and 9 bits for the fractional part) either employed in the K-best or ZF detection algorithms. In Fig. 13, the BER performance of the LUT and CORDIC scheme for a 64-QAM modulated 4×4 MIMO system using K-best detection technique is shown. Both the LUT compression and CORDIC schemes are with the word-length of 9 bits. The BER performance of

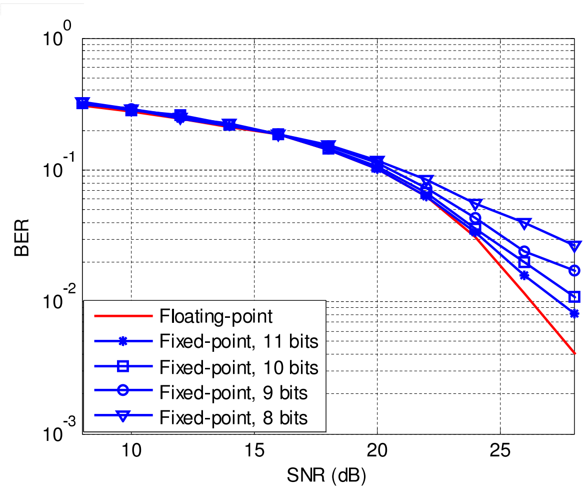

Fig. 11. The BER performance of the fixed-point simulations for a 4×4 MIMO system utilizing 64-QAM modulation and K-best detection technique for different word-lengths.

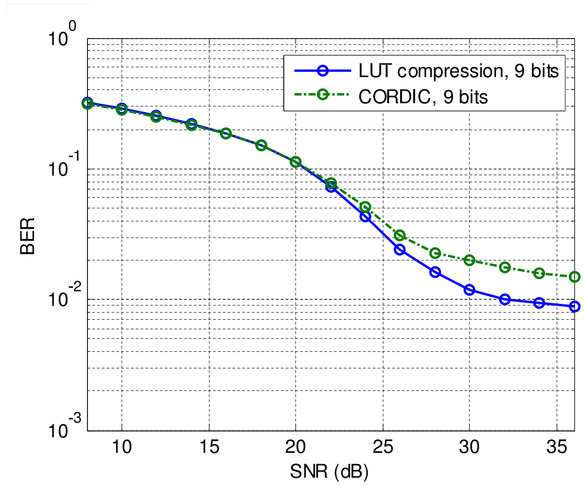

Fig. 13. The BER performance of the LUT compression and CORDIC approaches for a 4×4 MIMO system utilizing 64-QAM modulation and K-best detection technique.

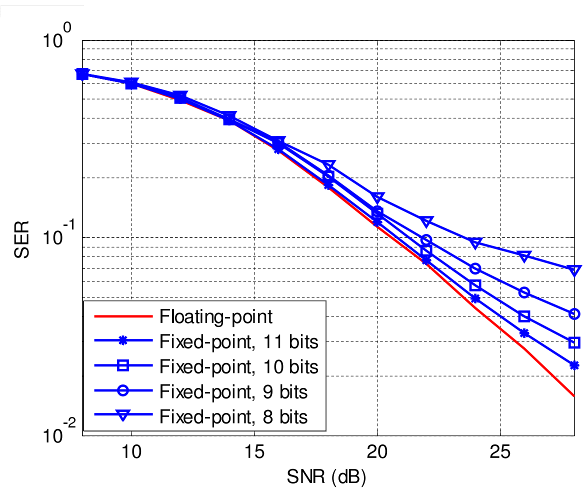

Fig. 12. The SER performance of the fixed-point simulations for a 4×4 MIMO system utilizing 16-QAM modulation and ZF detection technique for different word-lengths.

the LUT compression scheme is better than CORDIC-based implementation when the SNR is larger than 30 dB. Fig. 13 proves again that the LUT compression algorithm is more accurate than the CORDIC scheme.

The proposed QR decomposition architecture is synthesized in TSMC 90-nm technology. The gate count is about 262K and it can operate at 214 MHz. This architecture can decompose a new 4×4 complex-valued matrix every four clock cycles. Therefore, it achieves the throughput of 53.5 MQRD/s. The implementation results of our work are compared with several other published works in Table III. Our design and the designs in [4], [11], [25], [26] can perform a new QR decomposition operation every four clock cycles, however, our design has the highest throughput. The hardware complexity of our QR decomposition is lower than the design in [4]. Although the designs in [3], [6], [9], [25] have smaller gate counts than the proposed scheme, their throughputs and normalized throughputs are relatively low. Both the proposed QR decomposition and the design in [11] support 4×4 complex matrix decompositions and also 8×8 real matrix decompositions, which simplify the sorting operation if utilized in the K-best or the sphere decoding MIMO detection algorithms [28], [29]. Although [11] has a fewer gate count than ours, the proposed design provides an improved accuracy, which has been demonstrated in Fig. 3 to Fig. 6. With a greater accuracy (i.e., lower error rate for 25$\sim$35 dB SNR), transmit signal power can be scaled down as long as the target error rate can be supported. According to the implementation results in Table III, the proposed QR decomposition can be used in MIMO systems requiring high throughputs and greater accuracy with moderate hardware complexity.

VI. Conclusion

A high throughput and accurate QR decomposition architecture for 4×4 MIMO systems was proposed. Instead of using traditional CORDIC algorithms to implement the phase calculations and rotations, we employed the LUT compression method for approximating trigonometric functions rapidly, which also provides improved mean square error (MSE) performance compared to CORDIC algorithms. In addition, two-angle rotation was selected for complex Givens rotations as it has lower arithmetic complexity than three-angle rotation. The proposed Givens rotation-based QR decomposition architecture was implemented using TSMC 90-nm technology. It can operate at 214 MHz and it achieves the throughput of 53.5 MQRD/s. The implementation results indicate that the proposed QR decomposition approach has great potential for applications in MIMO wireless systems by achieving a higher throughput and improved accuracy compared to CORDIC-based implementations.

References

[1] E. Dahlman, S. Parkvall, and J. Skold, *4G LTE/LTE-Advanced for Mobile Broadband*. Elsevier Ltd., 2011.

[2] S. Sesia, I. Toufik, and M. Baker, *LTE-The UMTS Long Term Evolution: From Theory to Practice*. John Wiley & Sons Ltd., 2009.

[3] R. C-H Chang *et al.*, "Iterative QR decomposition architecture using the modified Gram-Schmidt algorithm for MIMO systems," *IEEE Trans. Circuits Syst. I*, vol. 57, no. 5, pp. 1095–1102, 2010.

[4] P-L Chiu *et al.*, "Interpolation-based QR decomposition and channel estimation processor for MIMO-OFDM system," *IEEE Trans. Circuits Syst. I*, vol. 58, no. 5, pp. 1129–1141, 2011.

[5] Y. Miyaoka, Y. Nagao, and M. K. et al., "Sorted QR decomposition for high-speed MMSE MIMO detection based wireless communication systems," in *Proc. IEEE Int. Symp. Circuits. Syst.*, 2012, pp. 2857–2860.

[6] P. Luethi *et al.*, "Gram-Schmidt-based QR decomposition for MIMO detection: VLSI implementation and comparison," in *Proc. IEEE Asian-Pacific Conference on Circuits and Systems*, 2008, pp. 830–833.

[7] C. K. Singh, S. H. Prasad, and P. T. Balsara, "VLSI architecture for matrix inversion using modified Gram-Schmidt based QR decomposition," in *Proc. Int. Conf. VLSI Design*, 2007, pp. 836–841.

[8] P. Salmela, A.Burian, H. Sorokin, and J. Takala, "Complex-valued QR decomposition implementation for MIMO receivers," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Process*, 2008, pp. 1433–1436.

[9] M. Shabany, D. Patel, and P. G. Gulak, "A low-latency low-power QR-decomposition ASIC implementation in 0.13 um CMOS," *IEEE Trans. Circuits Syst. I*, vol. 60, no. 2, pp. 327–340, 2013.

[10] D. Patel, M. Shabany, and P. G. Gulak, "A low-complexity high-speed QR decomposition implementation for MIMO receivers," in *Proc. IEEE Int. Symp. Circuits. Syst.*, 2009, pp. 33–36.

[11] Z.-Y. Huang and P.-Y. Tsai, "Efficient implementation of QR decomposition for Gigabit MIMO-OFDM systems," *IEEE Trans. Circuits Syst. I*, vol. 58, no. 10, pp. 2531–2542, 2011.

[12] M.-W. Lee, J.-H. Yoon, and J. Park, "High-speed tournament Givens rotation-based QR decomposition architecture for MIMO receiver," in *Proc. IEEE Int. Symp. Circuits. Syst.*, 2012, pp. 21–24.

[13] J-Y Wang *et al.*, "A 2x2 - 8x8 sorted QR decomposition processor for MIMO detection," in *Proc. IEEE Asian Solid-State Circuits Conference*, 2010, pp. 1–4.

[14] Z. h Liu, K. Dickson, and V. McCanny, "Application-specific instruction set processor for SoC implementation of mordern signal processing algorithms," *IEEE Trans. Circuits Syst. I*, vol. 52, no. 4, pp. 755–765, 2005.

[15] A. Paulraj, R. Nabar, and D. Gore, *Introduction to Space-Time Wireless Communications*. Cambridge University Press, 2003.

[16] J. G. Proakis, *Digital Communications*. 4th ed., McGraw-Hill, 2001.

[17] T.-H. Liu, "Comparisons of two real-valued MIMO signal models and their associated ZF-SIC detectors over the Rayleigh fading channel," *IEEE Trans. Wireless Commun.*, vol. 12, no. 12, pp. 6054–6066, 2013.

[18] M. Shabany and P. G. Gulak, "A 675 Mbps, 4×4 64-QAM K-best MIMO detector in 0.13 um CMOS," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 20, no. 1, pp. 135–147, 2012.

[19] S. Mondal *et al.*, "Design and implementation of a sort-free K-best sphere decoder," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 18, no. 10, pp. 1497–1501, 2010.

[20] C.-A. Shen and A. Eltawil, "A radius adaptive K-best decoder with early termination: algorithm and VLSI architecture," *IEEE Trans. Circuits Syst. I*, vol. 57, no. 9, pp. 2476–2486, 2010.

[21] A. Maltsev *et al.*, "Triangular systolic array with reduced latency for QR-decomposition of complex matrices," in *Proc. IEEE Int. Symp. Circuits. Syst.*, 2006, pp. 385–388.

[22] Y.-T. Hwang and W.-D. Chen, "Design and implementation of a high-throughput fully parallel complex-valued QR factorisation chips," *IET Circuits, Devices, and Systems*, vol. 5, no. 5, pp. 424–432, 2011.

[23] A. G. M. Strollo, D. D. Caro, and N. Petra, "A 630 MHz, 76 mW direct digital frequency synthesizer using enhanced ROM compression technique," *IEEE J. Solid-State Circuits*, vol. 42, no. 2, pp. 350–360, 2007.

[24] D. D. Caro and A. G. M. Strollo, "High-performance direct digital frequency synthesizers using piecewise-polynomial approximation," *IEEE Trans. Circuits Syst. I*, vol. 52, no. 2, pp. 423–337, 2005.

[25] T.-H. Liu, C.-N. Chiu, P.-Y. Liu, and Y.-S. Chu, "Block-wise QR-decomposition for the layered and hybrid Alamouti STBC MIMO systems: algorithms and hardware architectures," *IEEE Trans. Signal Process.*, vol. 62, no. 18, pp. 4737–4747, 2014.

[26] W.-Y. Chen, D. Guenther, and C.-A. S. et. al., "Design and implementation of a low-latency, high-throughput sorted QR decomposition circuit for MIMO communications," in *in Proc. IEEE Asian Pacific Conf. on Circuits Syst.*, 2016, pp. 277–280.

[27] M. D. Ercegovac and T. Lang, *Division and Square Root: Digit Recurrence Algorithms and Implementations*. Kluwer Academic, 1994.

[28] A. Burg and M. Borgmann and M. Wenk and M. Zellweger and W. Fichtner and H. Bolcskei, "VLSI implementation of MIMO detection using the sphere decoding algorithm," *IEEE J. Solid-State Circuits*, vol. 40, no. 7, pp. 1566–1577, 2005.

[29] M. Myllyla, J. R. Cavallaro, and M. Juntti, "Architecture design and implementation of the metric first list sphere detector algorithm," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 19, no. 5, pp. 895–899, 2011.