International Journal of Advanced Research in

Electronics, Communication & Instrumentation Engineering and Development

Volume: 1 Issue: 1 08-Nov-2013,ISSN_NO: 2347 -7210

Algorithm for Reduced Complexity and High Performance Embedded MIMO Receivers

D.JESLINE ANTONIO, M.E (Assist.prof)

SUN College of Engg & Tech.

Mail id: jesline.antonio6@gmail.com

ALLAN J WILSON, M.E (Assist.prof)

SUN College of Engg & Tech.

Mail id: allan94449@gmail.com

***Abstract** –Matrix inversion is a key enabling technology in MIMO (Multi Input Multi Output) communication systems*. **To date, no matrix inversion implementation has been devised which supports real-time operation for these standards. In this, we overcome this barrier by presenting a novel matrix inversion algorithm which is ideally suited to high performance floating-point implementation. Specifically, we present a matrix inversion approach based on modified squared Givens rotations (MSGR). This is a new QR decomposition algorithm which overcomes critical limitations in other QR algorithms that prohibits their application to MIMO systems. In addition, we present a novel modification that further reduces the complexity of MSGR by almost 20%. This enables real-time implementation with negligible reduction in the accuracy of the inversion operation, or the BER of a MIMO receiver based on this.**

Index terms**-BLAST, matrix inversion, multiple input multiple output (MIMO), QR decomposition**

I. INTRODUCTION

MULTIINPUT–MULTIOUTPUT (MIMO) technology, such as BLAST [1]–[3] for WiFi or WiMAX offers the potential to exploit spatial diversity in a communications channel to increase its bandwidth without sacrificing larger portions of the radio spectrum. The general form of a MIMO system composed of n_t transmit and n_r receive antennas is outlined in Fig. 1. Implementation of these systems involves satisfying the real-time performance requirements of the application (in terms of metrics such as throughput, latency, etc.), in a manner which efficiently exploits the embedded device(s) to implement such systems. A key feature of MIMO receivers is their reliance on matrix computations such as addition, multiplication and inversion for example, to enable operations such as minimum mean square error (MMSE) equalization during channel detection [3]. As a result, to implement algorithms such as MMSE designers currently have to develop custom algorithms which avoid explicit matrix inversion [8]. This severely complicates the implementation process.

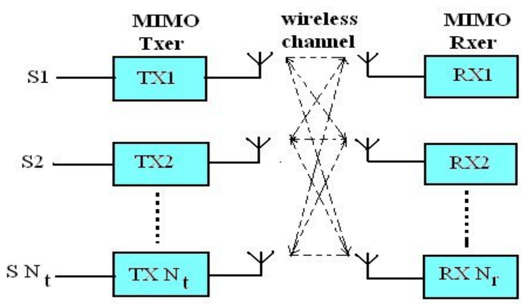

Fig. 1. MIMO system overview.

This paper presents a new matrix inversion approach which overcomes this real-time performance barrier. We develop a general purpose complex-valued matrix inversion algorithm and study its application to and integration in MIMO receiver algorithms and embedded architectures. Specifically, we make two main contributions.

1) We derive a new QR decomposition (QRD)-based algorithm known as modified squared Givens’ rotations (MSGR), which overcomes key limitations with other QRD-based approaches which hinder their adoption in MIMO receiver architectures.

2) We show how the complexity of MSGR-based matrix inversion may be further reduced by almost 20% by removing a scale factor term, with little effect on its numerical stability, or the perceived BER of a MIMO receiver in which it is integrated.

This paper is structured as follows. Section II summarises real-time matrix inversion approaches, justifying the use of QR-based approaches, and systems. Section III resolves these issues, deriving the MSGR algorithm. Section IV describes how we can reduce the complexity of MSGR.

International Journal of Advanced Research in
Electronics, Communication & Instrumentation Engineering and Development
Volume: 1 Issue: 1 08-Nov-2013,ISSN_NO: 2347 -7210

outlines barriers to their adoption in BLAST MIMO

TABLE I
MATRIX INVERSION COMPLEXITY COMPARISON

Approach	Additions (n_r=4)	Multiplications (n_r=4)	Divisions (n_r=4)	Square roots (n_r=4)
Gaussian [20]	$2n_r^3-n_r^2+n$ (116)	$2(n_r^3+n_r)$ (136)	$2n_r^2-n$ (28)	n_r(4)
Cholesky decomposition [20]	$2n_r^3-n_r^2+n$ (116)	$2(n_r^3+n_r)$ (136)	$2n_r^2-n$ (28)	n_r(4)
Square root decomposition [21]	$5n_r^3+3n_r^2$ (368)	$9n_r^3+7n_r^2$ (688)	n_r^2(16)	0(0)
Conventional givens rotations QRD [20]	$8n_r^3+n_r^2-6n_r$ (504)	$32/3\ n_r^3+11n_r^2-47/3\ n_r$ (796)	$3n_r^2-n_r$ (44)	$3/2(n_r^2-n_r)$ (18)
Squared givens rotations QRD [16]	$8n_r^3-4n_r^2$ (448)	$28/3n_r^3+7/2n_r^2-7/6\ n_r$	$5/2n_r^2-3/2n_r$ (34)	0(0)

II. BACKGROUND

MIMO systems grow larger to incorporate more antennas, both solutions will tend to the same complexity. However, given that the matrix inversion approaches in [4]–[7] are all either throughput or latency deficient by a factor of at least 2, there appear to be substantial issues to be overcome before real-time matrix inversion in systems such as 802.11n is feasible.

Matrix inversion techniques are, generally, either *iterative* or *direct* [9]. Iterative methods, such as the Jacobi or Gauss-Seidel methods, start with an estimate of the solution and iteratively update the estimate based on calculation of the error in the previous estimate, until a sufficiently accurate solution is derived. The sequential nature of this process can limit the amount of available parallelism and make high throughput implementation problematic [9]. On the other hand, direct methods such as Gaussian elimination (GE), Cholesky decomposition (CD), and QRD typically compute the solution in a known, finite number of operations and exhibit plentiful data and task-level parallelism. Table I describes the complexity of a number of direct matrix inversion algorithms in MIMO communications systems composed of n_t=n_r antennas, as well as an absolute complexity measure for n_r=4.

As Table I shows, CD suffers from the drawback of requiring symmetric matrices, a condition not guaranteed to occur in MIMO systems, limiting its applicability. Despite their relatively high complexity as compared with CD, QRD approaches

are an attractive alternative not only because of their ability to overcome the symmetric restriction, but also because of their innate numerical stability [10]. Furthermore, the plentiful data and task level parallelism available in these algorithms has previously been comprehensively exploited in a range of algorithms and architectures for recursive least squares (RLS) in adaptive beam forming and RADAR [11]–[13].

no reported variants which exhibit significantly lower complexity whilst maintaining accuracy.

3) There is no currently reported implementation of SGR based matrix inversion which can meet the high real-time performance demands of modern MIMO receivers.

III. MODIFIED SGR FOR COMPLEX MATRIX INVERSION

As regards QRD algorithms, Givens' rotations [14] QRD algorithms are more easily parallelized than Householder Transformations [15], but the methods for implementing Givens' rotations, i.e., conventional Givens' rotations (CGR) [14], squared Givens' rotations (SGR) [16], and CORDIC [17] all place different constraints on implementations. The authors in [18] show that fixed-point CORDIC-based QR algorithms are more accurate for linear MMSE detection of practical MIMO-OFDM channels than SGR employing conventional arithmetic. However this comes at an excessively high area cost due to the use of CORDIC operators; indeed [8] and [19] report 3.5:1 and 3:1 area efficiency advantages when employing conventional mathematical operators as opposed to CORDIC; [19] also describes a 25:1 sample rate advantage associated with employing conventional arithmetic and demonstrates that floating-point arithmetic can be employed to overcome the precision issues outlined in [18] at no area cost. These factors seem to favour SGR-based implementation over CORDIC. Whilst CGR does not fundamentally require CORDIC for implementation, it does require widespread use of costly square-root operations, and is generally more computationally demanding than SGR (see Table I).

As such, the ability of SGR to avoid the use of CORDIC and square-root operations, reduce overall complexity and exploit floating-point arithmetic at no area cost promises an appealing blend of numerical stability and computational complexity.There are, however, a number of critical deficiencies which restrict its adoption in MIMO systems.

1) As described in Section III-B, SGR produces erroneous results when zeros occur on the diagonal elements of either the input matrix or the partially decomposed matrices generated during the triangularization.
2) Complex-valued SGR is highly computationally demanding and there are

To illustrate how MSGR extends SGR for complex-valued data, we use an example. Consider a 3×4 matrix of complex values. MSGR generates an upper triangular matrix, eliminating a_1, $b_1$and b_2 in a three-stage approach.

$$\begin{bmatrix} r \\ a \\ b \end{bmatrix} = \begin{bmatrix} r_1 & r_2 & r_3 & r_4 \\ a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \end{bmatrix} \quad \ldots\ldots\ldots(5)$$

Stage 1: Rotate rows **r** and **a** to eliminate $\mathbf{a_1}$ element.

A. *Modified Squared Givens Rotations*

Inversion of a matrix **A** can be performed by firstly decomposing this into an upper triangular form, which can be more easily inverted. QRD is one such approach to perform this triangularization by decomposing **A** into two resulting matrices **Q** and **R**

$$\mathbf{A= QR} \quad \ldots\ldots\ldots\ldots (1)$$

Where, Q- unitary matrix.

R- upper triangular matrix.

Here Q^{-1} = Hermitian transpose Q^{H}

$$A^{-1} = (QR)^{-1} = R^{-1}Q^{H} \quad \ldots\ldots\ldots\ldots(2)$$

After QRD, inversion is much simpler because the inversion of the upper triangular matrix can be derived using back substitution as in,

$$w_{ij} = \begin{cases} -\sum_{k=1}^{j-1} w_{ik}\, r_{kj}/r_{jj} & i < j \\ 1/r_{jj} & i = j \\ 0 & otherwise \end{cases} \quad \ldots(3)$$

Using SGR, the matrix **A** is decomposed in to **QA** matrix, and **U** is an upper triangular matrix.

The inversion of a matrix using **SGR is**

A^{-1} = $(Q_A D_U U)^{-1} = U^{-1} (Q_A D_U^{-1})^{-1}$

$(QA\ D_U^{-1})^{-1} = D_R\, Q^{H} = QA^{H}$ .... (4)

$$\bar{b} = w_b^{\frac{1}{2}} \left[\frac{\bar{r}_1}{r_1} \, vb - \frac{vb1}{r_1}\bar{r} \right]$$

$$= w_b^{\frac{1}{2}} \left[\frac{\bar{r}_1}{r_1} \, vb - \frac{vb1}{u_1}\bar{u} \right] = \bar{w}_b^{\frac{1}{2}} \bar{v}_b \quad \ldots\ldots\ldots(11)$$

Stage 3: Rotate $\bar{a}$ and $\bar{b}$ to eliminate. To carry out these rotations, translated from **V** -space to **U** –space.

$$q = \left(r_1^* r_1 + a_1^* a_1\right)^{1/2}$$
$$\bar{r} = q^{-1}(r_1^* r_1 + a_1^* a)$$
$$\bar{a} = q^{-1}(-a_1 r + r_1 a) \qquad \text{……….(6)}$$

Introducing **U** , the update process for **r** to be expressed as,

$$u = r_1^* r_1$$
$$\bar{u} = \bar{r}_1^* \bar{r} \qquad \text{………………..(7)}$$
$$\bar{u} = u + a_1^* a$$

Similarly, introducing **v**, where $w_a > 0$ is a scale factor

$$a = w_a^{\frac{1}{2}} v$$

$$\bar{a} = w_a^{\frac{1}{2}} \left(\frac{r_1}{\bar{r}} v - \frac{v_1}{u} r \right) = \bar{w}_a^{\frac{1}{2}} \left(\frac{r_1}{\bar{r}} v - \frac{v_1}{\bar{u}} r \right)$$

…………..(8)

Updating of row can be written as,

Stage 2: Rotate $\bar{r}$ and **b** TO ELIMINATE $\mathbf{b_1}$, row **b** must now be translated to **V**-space for rotation according to,

$$\bar{w}_a = w_a \frac{u_1}{\bar{u}_1} \qquad \text{……….. (9)}$$

$$v_b = w^{-\frac{1}{2}} \bar{b} = w^{-\frac{1}{2}} \bar{w}_b^{\frac{1}{2}} \bar{v}_b \qquad \text{…….. (13)}$$

The final phase of translating the last row to **U**-space is necessary in order to make the diagonal element in this row a real value. The MSGR method for the general case is described using where k is the k^th column being processed.

$$\bar{u} = u + w v_k^* v \qquad \text{……… (14)}$$

$$\bar{v} = v - \left(\frac{v_k}{u_k}\right) u \qquad \text{…………. (15)}$$

$$\bar{w} = w \frac{u_k}{\bar{u}_k} \qquad \text{………….. (16)}$$

$$u_a = \bar{a}_2^* \bar{a}$$
$$= \frac{1}{2}\left(\frac{r}{\bar{r}} v_2 - \frac{v_1}{u_1} r\right)^* \frac{1}{2}\left(\frac{r}{\bar{r}} v - \frac{v_1}{u_1} r\right)$$
$$= \bar{w}_a \bar{v}_2^* \bar{v}$$

…….…..(12)

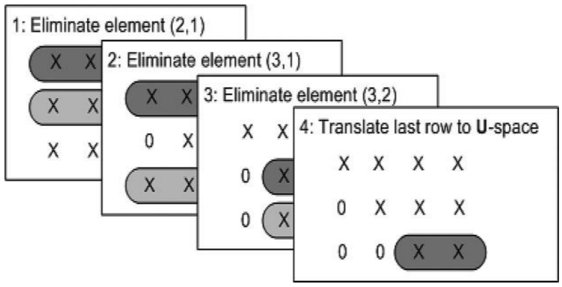

Fig.2. MSGR operation sequence

$$b = w_b^{\frac{1}{2}} v_b$$

Effectively, **r** and **a** have been translated to **U** and **V** -space, respectively.

C. Back Substitution

MSGR decomposes the input matrix as **A**, which may be inverted as $\mathbf{U}^{-1}(\mathbf{Q_A D_U}^{-1})^{-1}$. Inversion of the upper triangular matrix **U** can be computed using back-substitution on the result of the MSGR operation.

Given this back-substitution operation, MSGR-based matrix inversion may be split into three sub operations:

Decomposition of the input matrix **A** into the upper triangular matrix **U** , formation of $\mathbf{U}^{-1}$ from **U** via back-substitution, and finally multiplication by $(\mathbf{Q_A D_U}^{-1})^{-1}$ to form the product $\mathbf{U}^{-1}(\mathbf{Q_A D_U}^{-1})^{-1}$. We can use this to formulate an operational model of MSGR-based matrix inversion to complement the mathematical model described thus far. The

B. *Processing Zero Values Incurred on the Matrix Diagonal*

Dohler proposed a solution for dealing with U_k=0, as given in

$$for\ u_k = 0, v_k \neq 0 \begin{cases} \bar{u} = wvk^* v \\ \bar{v} = arbitrary \\ \bar{w} = 0 \end{cases} \quad ..... (17)$$

Despite this, the condition of both U_k=V_k=0 has not previously been considered, and the solutions proposed in prohibit further processing based on these updated values.

$$u_k = 0, v_k \neq 0 \begin{cases} \bar{u} = wvk^* v \\ \bar{v} = -u \\ \bar{w} = w \end{cases} \quad .. (18)$$

This caveat fully defines MSGR based matrix triangulation and has resolved the outstanding barriers to employing SGR for complex valued matrix inversion. It remains to convert the triangularized matrix to the inverse, which requires a suitable back-substitution operation.

~~formation of from via back-substitution and~~ subsequent multiplication by is then performed by an ***Invert and Multiply (IAM) Array***.

$$G_{ij} = \begin{cases} -\frac{1}{u_{jj}} \left(\sum_{ik}^{j-1} G_{ik} u_{kj} \right) & i < j \\ \frac{1}{u_{ij}} & i = j \\ 0 & i > j \end{cases} \quad ..(19)$$

IV. REDUCED COMPLEXITY MSGR-BASED MATRIX INVERSION FOR BLAST MIMO

Reduced Complexity MSGR-Based Matrix Inversion

We propose to remove the scale factor **w** and its associated computations from the MSGR cells to reduce the complexity of MSGR-based matrix inversion. Using MSGR without the **w**-factor, the matrix **A** is decomposed as in

$$\mathbf{A} = \mathbf{Q_W}\, \mathbf{U_W} \quad .... (20)$$

MSGR without the **W**-factor, $\mathbf{U_W}$ is an invertible upper triangular matrix, and the inversion of **A** can be given as in

$$\mathbf{A^{-1} = (Q_W\, U_W)^{-1} = U_W^{-1}\, Q_W^{-1}}. \quad \text{……… (21)}$$

Once all **w**-related computations have been removed, the MSGR array now produces an upper triangular matrix $\mathbf{U_W}$ given an input matrix **A**.

TABLE II
MSGR COMPLEXITY ANALYSIS

Operation	Multiplications	Additions	Divisions
MSGR *w*-factor included	658	448	34
MSGR *w*-factor excluded	532	448	28
% saving	19	0	18

However, whilst the complexity reduction offered by removing the **w**-factor is clearly attractive, it is only advantageous if it does not significantly reduce the accuracy of the resulting inverted matrices, or the accuracy of any MIMO receiver algorithm built upon MSGR-based matrix inversion.

V. RESULTS AND DISCUSSION

The effect of removing the **w**-factor on the accuracy of the resulting inverted matrix is described in the graphs of Fig.4. These two graphs measure the deviation of the product of the original matrix and its inverse from the matrix (y- axis) for each of 200, 4× 4 MIMO rich scattering Rayleigh-fading channel matrices, which are enumerated on the x-axis. The complexity of MSGR is reduced by exclude the w-factor.

In other words, the complexity is reduced by reduce the number of operations performed in MSGR matrix inversion.

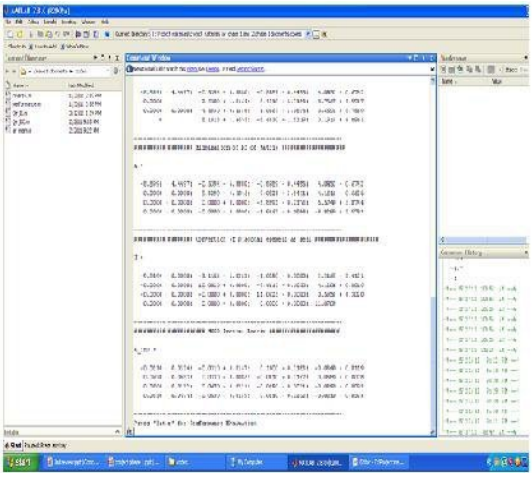

Fig.4 (a)

Fig 4 (a) shows the MSGR method for matrix inversion using matlab software. This method enables a suitable mechanism for complex matrix inversion and also it reduces the computation complexity almost 20 percentages.

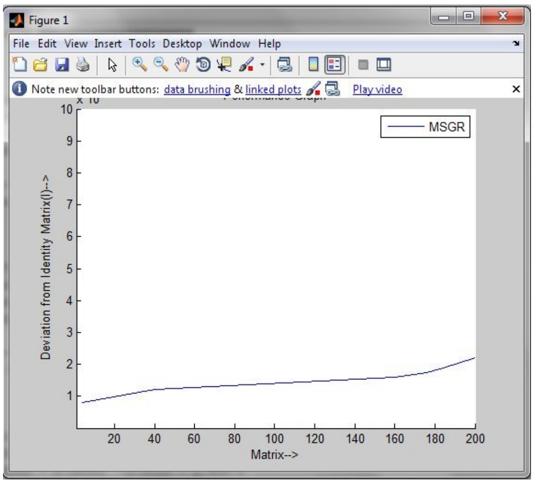

Fig.4 (b)

Fig 4(b) shows the error plot for MSGR method. It is the deviation value of identity matrix in y axis corresponds to the number of matrices in x axis.

Fig 4(c) shows the comparison plot for existing method (CORDIC) and the MSGR method. From this plot clearly shows that the Error rate is less compared to the existing method. It is because of

International Journal of Advanced Research in
Electronics, Communication & Instrumentation Engineering and Development
Volume: 1 Issue: 1 08-Nov-2013,ISSN_NO: 2347 -7210

number of operations reduced in the MSGR matrix or exclude the w-factor in the MSGR method. So that it gives better performance without reducing the accuracy.

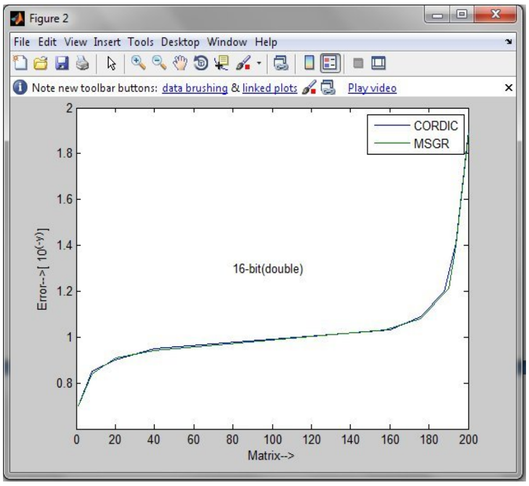

Fig.4 (c)

Fig. 4. w-less MSGR-based matrix inversion accuracy comparison. (b) Floating-point matrix inversion error. (c) Fixed-point matrix inversion error.

VI. CONCLUSION

Explicit matrix inversion is a major bottleneck in the design of embedded MIMO transceiver architectures. Until now, there has been no appropriate solution to this problem for state-of the- art MIMO systems, such as those in 802.11n systems, with the large disparities between required and actual performance indicating the need for a thorough review of both algorithms and architectures employed.

The work presented in this paper has solved this problem. We have derived Modified Squared Givens' rotations (MSGR), an algorithm for QR-based matrix triangularization and inversion which overcomes deficiencies in the standard SGR algorithms. This provides a complex-valued matrix inversion method which not only overcomes key factors for integration in MIMO systems, but also enables a suitable mechanism for complex matrix inversion more generally. Moreover, we have shown that the computational complexity of the algorithm may be further reduced by almost 20% with minimal impact on the accuracy of the inverted matrix or the perceived BER of a BLAST MIMO receiver based upon it.

VI. REFERENCES

[1] G. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas,"*Bell Lab. Tech. J.*, pp. 41–59, 1996.

[2] P. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, "V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel," in *Proc. URSI Int. Symp. Signals, Syst., Electron.*, 1998, pp. 295–300.

[3] M. Sellathurai and S. Haykin, "Turbo-blast for wireless communications: Theory and experiments," *Bell Lab. Tech. J.*, vol. 50, pp. 2538–2546, 2002.

[4] F. Echman and V. Owall, "A scalable pipelined complex valued matrix inversion architecture," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS 2005)*, May 2005, pp. 4489–4492.

[5] I. LaRoche and S. Roy, "An efficient regular matrix inversion circuit architecture for MIMO processing," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS 2006)*, May 2006, pp. 4819–4822.

[6] M. Myllyla, J.-H. Hintikka, J. Cavallaro, M. Juntti, M. Limingoja, and A. Byman, "Complexity analysis of MMSE detector architectures for MIMO OFDM systems," in *Conf. Rec. 39th Asilomar Conf. Signals, Syst., Comput. (2005)*, vol. 1, pp. 75–81.

[7] M. Karkooti, J. Cavallaro, and C. Dick, "FPGA implementation of matrix inversion using QRD-RLS algorithm," in *Conf. Rec. Asilomar Conf. Signals, Syst. Comput.*, 2005, pp. 1625–1629.

[8] H. S. Kim, W. Zhu, J. Bhatia, K. Mohammed, A. Shah, and B. Daneshrad, "A practical, hardware friendly MMSE detector for MIMO-OFDM-based systems," *EURASIP J. Adv. Signal Process.*, vol. 2008, pp. 1–14, Jan. 2008.

[9] M. Ylinen, A. Burian, and J. Takala, "Updating matrix inverse in fixed-point representation: Direct versus iterative methods," in *Proc. Int. Symp. System-on-Chip*, 2003, pp. 45–48.

[10] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice- Hall, 1991.

[11] G. Lightbody, R.Walke, R.Woods, and J. McCanny, "Linear QR architecture for a single chip adaptive beamformer," *J. VLSI Signal Process. Syst. Signal, Image, and Video Technol.*, vol. 24, pp. 67–81, 2000.

[12] Z. Liu, J. McCanny, and R. Walke, "Generic Soc QR array processor for adaptive beamforming," *IEEE Trans. Circuits Syst., Part II—Analog Digit. Signal Process.*, vol. 50, no. 4, pp. 169–175, Apr. 2003.

[13] R. Hamill, J. McCanny, and R. Walke, "On-line CORDIC algorithm and VLSI architecture for implementing QR-array processors," *IEEE Trans. Signal Process.*, vol. 48, no. 2, pp. 592–598, Feb. 2000.

[14] W. Givens, "Computation of plane unitary rotations transforming a general matrix to triangular form," *J. Soc. Indust. Appl. Math*, vol. 6, pp. 26–50, 1958.

[15] G. Golub, "Numerical methods for solving least-squares problems," *Num. Math.*, vol. 7, pp. 206–216, 1965.

[16] R. Dohler, "Squared givens rotations," *IMA J. Numer. Anal.*, vol. 11, pp. 1–5, 1991.