

DOI: 10.17516/1997-1397-2020-13-2-160-169

УДК 519.873

A Recursive Algorithm for Estimating the Correlation Matrix of the Interference Based on the QR Decomposition

Valery N. Tyapkin*

Dmitry D. Dmitriev†

Andrey B. Gladyshev‡

Peter Yu. Zverev§

Siberian Federal University
Krasnoyarsk, Russian Federation

Received 02.11.2019, received in revised form 10.12.2019, accepted 20.01.2020

Abstract. Many tasks of digital signal processing require the implementation of matrix operations in real time. These are operations of matrix inversion or solving systems of linear algebraic or differential equations (Kalman filter). The transition to the implementation of digital signal processing on programmable logic device (FPGAs), as a rule, involves calculations based on the representation of numbers with a fixed point. This makes solving spatio-temporal processing problems practically impossible based on conventional computational methods. The article discusses the implementation of spatial-temporal signal processing algorithms in satellite broadband systems using QR decomposition. The technologies of CORDIC computations required for recurrent QR decomposition when used together in systolic algorithms are presented.

Keywords: phased antenna array, adaptive algorithms, Kalman filter, recursive least squares algorithm (RLS), QR decomposition, systolic algorithm.

Citation: V.N.Tyapkin, D.D.Dmitriev, A.B.Gladyshev, P.Yu.Zverev, A Recursive Algorithm for Estimating the Correlation Matrix of the Interference Based on the QR Decomposition, J. Sib. Fed. Univ. Math. Phys., 2020, 13(2), 160–169. DOI: 10.17516/1997-1397-2020-13-2-160-169.

Introduction

Modern satellite broadband radio communication systems have a significant drawback - low noise immunity. The solution to this problem is based on the use of phased array antennas. Such antennas are controlled using adaptive algorithms, the parameters of which can be quickly changed in accordance with the emerging interference environment.

To work effectively in real conditions of parametric a priori uncertainty and a dynamic change in the statistical characteristics of interference, adaptive systems are required, the parameters of which can be quickly changed in accordance with the interference environment. Currently, theoretically substantiated and tested in practice, the methods of the Markov theory of optimal filtration. This theory has been fully and strictly developed in a number of books and articles

*tyapkin58@mail.ru

†dmitriev121074@mail.ru <https://orcid.org/0000-0001-6438-6094>

‡a-ghonass@yandex.ru

§peter676@mail.ru

© Siberian Federal University. All rights reserved

[1–12]. In [6, 10, 11], the solution to the problem of adaptive filtering of signals based on the use of the lemma on the inversion of the correlation matrix (MIL) of input signals of an adaptive filter is considered. This solution leads to a recursive least squares (RLS) algorithm.

The same problem can also be solved by reducing the matrix of input samples of the adaptive filter signals to a triangular form. In this case, the range of numbers involved in the calculations is reduced by comparing the solution to this problem with the estimation of the inverse correlation matrix of interference using MIL. This increases the stability of QRD RLS algorithms.

Assume that the signal and interference affect the input of a multichannel M -element phased array antenna. The set of signals from the outputs of the M -element antenna array is described by the time function $y_1(t), y_2(t), \dots, y_m(t)$ and form a column vector $y_1(t) = [y_1(t), y_2(t), \dots, y_m(t)]^T$. Moreover, a single-channel reception ($M = 1$) is considered as a special case of multi-channel.

Discretization of the received useful and interference signals is performed at the radio frequency f_0 . A feature of this is the small sampling interval Δt , which is approximately half the period of the carrier frequency $T_d = 1/f_0$, $\Delta t \approx 1/2f_0 = T_0/2$. Discrete interference values obtained from the antenna are random numbers that are conveniently represented as a column vector $y_i = [y_i(k\Delta t)] = [y_i(k)]$, $k = \overline{1, L}$, where L determines the duration of the observation interval $T - L = T/\Delta t$. In the case of multichannel reception, the vector of received oscillations will have the following form $\mathbf{Y} = (\mathbf{Y}_1 \mathbf{Y}_2 \dots \mathbf{Y}_k \dots \mathbf{Y}_T)^T$ and dimension $(T \times M)$.

Most QR decomposition algorithms are based on Householder reflection and Givens rotations [13]. For the implementation of space-time processing, the most useful is the recursive version of the Givens method, which provides updates to the solutions at the rate of arrival of the input samples of the signal. High real-time performance provides a systolic version of the QR algorithm using pipelined implementation of Givens rotations on FPGA. High speed fixed-point number calculations on FPGAs are provided by the CORDIC processor. The principle of its operation differs significantly from the arithmetic-logical devices of existing processors. To implement Givens rotation, 10 shift-addition operations are sufficient. In this case, an accuracy sufficient to achieve an interference suppression ratio of more than 50 dB is ensured.

Thus, the development of this research area promises a significant improvement in the quality of reception and processing of broadband signals and noise immunity based on the existing element base and is relevant. Consider the implementation of spatial-temporal signal processing algorithms using QR decomposition. To solve the problem of recurrent QR decomposition, we will develop CORDIC computing technology in systolic algorithms.

1. Recursive adaptation algorithm using QR decomposition

A recursive adaptation algorithm using QR decomposition estimates the filter coefficients at the current time step through the calculated filter coefficient at the previous step. Due to its recursive nature, the algorithm is called QRD — a recursive QR decomposition algorithm. QR factorization consists in reducing a linear system to a triangular one. For this, the original matrix is represented as the product of the upper triangular matrix \mathbf{R} and the orthogonal matrix \mathbf{Q} .

Consider a system of linear equations

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \quad (1)$$

where $\mathbf{A} - (n \times m)$ is the matrix, \mathbf{x} is the vector of derivatives (for example, weights for the adaptive antenna array), \mathbf{b} is the m -vector. The QR decomposition of matrix \mathbf{A} of size $(n \times m)$

for any $n \gg m$ can be described as:

$$\mathbf{A} = \mathbf{Q} \cdot \mathbf{R}, \quad (2)$$

\mathbf{Q} is a unitary matrix: $\mathbf{Q} \cdot \mathbf{Q}^H = \mathbf{I}$, where \mathbf{I} is the identity matrix, \mathbf{R} is $(n \times m)$ the upper right triangular matrix. Equation (2) can be written in divided form:

$$\mathbf{A} = \mathbf{Q} \cdot \mathbf{R} = \mathbf{Q} \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix} = [\mathbf{Q}_1 \ \mathbf{Q}_2] \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix} = \mathbf{Q}_1 \cdot \mathbf{R}_1, \quad (3)$$

where \mathbf{R}_1 is the $(m \times m)$ triangular matrix, \mathbf{Q}_1 is the $(m \times n)$ matrix and \mathbf{Q}_2 is the $((n-m) \times m)$ matrix.

In the spatio-temporal processing of broadband signals, the system of equations is usually redefined because $n \gg m$. Solution (2) minimizing the norm of the residual $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|$ has the form $\mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}$.

Substitution of equality $\mathbf{A} = \mathbf{Q}^T \mathbf{R}$ gives the following form

$$\mathbf{A} = (\mathbf{R}^T \mathbf{Q}^T \mathbf{Q} \mathbf{R})^{-1} \mathbf{R}^T \mathbf{Q}^T \mathbf{b} = \mathbf{R}^{-1} \mathbf{Q}^T \mathbf{b}.$$

Here the triangular $(m \times m)$ -matrix is subject to circulation, which requires $(m^2)/2$ operations of addition and multiplication. Let us synthesize a recursive algorithm for estimating the correlation matrix of interference based on QR decomposition. From the theory of matrices it is known that there exists a unitary matrix $\mathbf{Q}_k(k)$, that for any \mathbf{A}_{kM} matrix can be obtained by decomposition

$$\mathbf{Q}_k^H \mathbf{A}_{kM} = \begin{bmatrix} \mathbf{F}_M \\ \mathbf{0}_{(T-M)M} \end{bmatrix}, \quad (4)$$

where \mathbf{F}_M is the upper triangular square matrix, called the Cholesky decomposition, $\mathbf{0}_{(T-M)M}$ is the zero rectangular matrix. The superscript H means complex conjugation and transposition.

For the unitary matrices $\mathbf{Q}_k(k)$, the equalities $\mathbf{Q}_k(k) \cdot \mathbf{Q}_k^H(k) = \mathbf{Q}_k^H(k) \cdot \mathbf{Q}_k(k) = \mathbf{I}$ and $\mathbf{Q}_k^H(k) = \mathbf{Q}_k^{-1}(k)$.

If designated

$$\mathbf{A}_{kM} = \mathbf{\Lambda}_k^{0.5}(k) \mathbf{Y}_{Mk}^H(k), \quad (5)$$

where

$$\mathbf{\Lambda}_k^{0.5} = \text{diag} \left\{ \sqrt{\lambda^{k-1}}, \sqrt{\lambda^{k-2}}, \dots, \sqrt{\lambda^1}, 1 \right\} = \begin{bmatrix} \sqrt{\lambda^{k-1}} & 0 & \dots & 0 & 0 \\ 0 & \sqrt{\lambda^{k-2}} & \dots & 0 & 0 \\ \vdots & 0 & \ddots & 0 & \vdots \\ 0 & 0 & \dots & \sqrt{\lambda} & 0 \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix}. \quad (6)$$

The parameter λ is used to weight the signals and allows you to take into account changes in the statistics of signals if they are non-stationary and their statistical parameters change over time. The parameter λ is also called the exponential weighting parameter or the "forgetting factor" parameter. Its value is usually limited by the limits $(1 - 0.4K) \leq \lambda \leq 1$ [14, 15]. For example, at $\lambda = 0.9$, $\lambda_0 = 1$, $\lambda_1 = 0.9$, $\lambda_2 = 0.81$, $\lambda_3 = 0.72$, $\lambda_4 = 0.66$, $\lambda_5 = 0.59, \dots, \lambda_{10} = 0.35, \dots, \lambda_{20} = 0.12, \dots, \lambda_{30} = 0.04, \dots, \lambda_{40} = 0.015, \dots, \lambda_{50} = 0.005, \dots, \lambda_{100} = 0.000027$, that is, the parameter λ determines the "memory" of the algorithm for solving the recursive problem.

For $0 < \lambda < 1$ and small values of the difference $k - i$, the summed terms are weighted with large weights, and for large values of this difference, with smaller weights. For $\lambda = 1$ this

"memory" is equal to k samples of the observed signals. For $0 < \lambda < 1$, the contribution of the same i -th samples to the sum is different for different λ . This contribution is greater for large λ and smaller for smaller λ . That is, with decreasing λ , the effective memory decreases and vice versa. Substituting equation (5) in (4) we obtain

$$\mathbf{Q}_k^H \Lambda_k^{0.5}(k) \mathbf{Y}_{Mk}^H(k) = \begin{bmatrix} \hat{\mathbf{R}}_M(k) \\ \mathbf{0}_{(T-M)M} \end{bmatrix}, \quad (7)$$

where

$$\mathbf{Y}_{Mk}(k) \Lambda_k^{0.5}(k) {}_{kM} \mathbf{Q}_k(k) = \begin{bmatrix} \hat{\mathbf{R}}_M(k) \\ \mathbf{0}_{(T-M)M} \end{bmatrix}^H = \begin{bmatrix} \hat{\mathbf{R}}_M^H(k), \mathbf{0}_{(T-M)M}^T \end{bmatrix}. \quad (8)$$

Reducing the matrix $\hat{\mathbf{R}}_M(k)$ to a triangular form using the observation matrix $\mathbf{Y}_{Mk}^H(k)$ can be performed in various ways, the main of which is Givens rotation. The matrix $\hat{\mathbf{R}}_M(k)$ can be obtained recursively in time, performing calculations for the k -th iteration using the results from the previous, $(k-1)$ -th iteration. This is as follows.

Let us assume that at iteration $k-1$ there is a decomposition:

$$\mathbf{Q}_{k-1}^H(k-1) \Lambda_{k-1}^{0.5}(k-1) \mathbf{Y}_{M(k-1)}^H(k-1) = \begin{bmatrix} \hat{\mathbf{R}}_M(k-1) \\ \mathbf{0}_{(k-1-M)M} \end{bmatrix} \quad (9)$$

and conversion is required

$$\mathbf{Q}_k^H(k) \Lambda_k^{0.5}(k) \mathbf{Y}_{Mk}^H(k) = \begin{bmatrix} \hat{\mathbf{R}}_M(k) \\ \mathbf{0}_{(k-M)M} \end{bmatrix}. \quad (10)$$

Using the result of (9), we define the matrix

$$\tilde{\mathbf{Q}}_k^H(k) = \begin{bmatrix} \mathbf{Q}_{k-1}^H(k-1) & \mathbf{0}_{k-1} \\ \mathbf{0}_{k-1}^T & 1 \end{bmatrix}. \quad (11)$$

If the matrix $\Lambda_k^{0.5}(k) \mathbf{Y}_{Mk}^H(k)$ multiplied from left on the matrix (11), this operation modifies equation (9), adding it to the matrix in the right side of $(k+1) - w$ (bottom) row:

$$\begin{aligned} \tilde{\mathbf{Q}}_k^H(k) \Lambda_k^{0.5}(k) \mathbf{Y}_{Mk}^H(k) &= \tilde{\mathbf{Q}}_k^H(k) \begin{bmatrix} \lambda^{0.5} \Lambda_{k-1}^{0.5}(k-1) \mathbf{Y}_{M(k-1)}^H(k-1) \\ \mathbf{Y}_M^H(k) \end{bmatrix} = \\ &= \begin{bmatrix} \mathbf{Q}_{k-1}^H \lambda^{0.5} \Lambda_{k-1}^{0.5}(k-1) \mathbf{Y}_{M(k-1)}^H(k-1) \\ \mathbf{Y}_M^H(k) \end{bmatrix} = \begin{bmatrix} \lambda^{0.5} \hat{\mathbf{R}}_M(k-1) \\ \mathbf{0}_{(k-1-M)M} \\ \mathbf{Y}_M^H(k) \end{bmatrix}. \end{aligned} \quad (12)$$

To perform the transformation (10), in equation (12) it is necessary to zero the last line. From equations (10) and (12) it follows that

$$\begin{aligned} \mathbf{Q}_k^H \Lambda_k^{0.5}(k) \mathbf{Y}_{Mk}^H(k) &= \hat{\mathbf{Q}}_k^H(k) \tilde{\mathbf{Q}}_k^H(k) \Lambda_k^{0.5}(k) \mathbf{Y}_{Mk}^H(k) = \\ &= \hat{\mathbf{Q}}_k^H(k) = \begin{bmatrix} \lambda^{0.5} \hat{\mathbf{R}}_M(k-1) \\ \mathbf{0}_{(k-1-M)M} \\ \mathbf{Y}_M^H(k) \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{R}}_M(k) \\ \mathbf{0}_{(k-1-M)M} \\ \mathbf{0}_M \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{R}}_M(k) \\ \mathbf{0}_{(k-M)M} \end{bmatrix}, \end{aligned} \quad (13)$$

where the matrix $\mathbf{Q}_k^H(k)$ is the product of two matrices

$$\mathbf{Q}_k^H(k) = \hat{\mathbf{Q}}_k^H(k) \tilde{\mathbf{Q}}_k^H(k). \quad (14)$$

transformation (15), for which matrix $\mathbf{G}_{M+1}^{(l)}(k)$ is used. Then

$$\begin{aligned}
 \mathbf{G}_{M+1}^{(1)}(k) \times \begin{bmatrix} \lambda^{0.5} \hat{\mathbf{R}}_N(k-1) \\ \tilde{\mathbf{Y}}_M^{(0)H}(k) \end{bmatrix} &= \begin{bmatrix} c_{M,1}(k) & 0 & \cdots & 0 & -S_{M,1}^*(k) \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \mathbf{I}_{N-3} & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ s_{M,1}(k) & 0 & \cdots & 0 & c_{M,1}(k) \end{bmatrix} \times \\
 \times \begin{bmatrix} \lambda^{0.5} \hat{R}_{M,11}(k-1) & \lambda^{0.5} \hat{R}_{M,12}(k-1) & \cdots & \lambda^{0.5} \hat{R}_{M,1(M-1)}(k-1) & \lambda^{0.5} \hat{R}_{M,1M}(k-1) \\ 0 & \lambda^{0.5} \hat{R}_{M,22}(k-1) & \cdots & \lambda^{0.5} \hat{R}_{M,2(M-1)}(k-1) & \lambda^{0.5} \hat{R}_{M,2M}(k-1) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \hat{R}_{M,(M-1)(M-1)}(k-1) & \hat{R}_{M,(M-1)M}(k-1) \\ 0 & 0 & \cdots & 0 & \hat{R}_{M,MM}(k-1) \\ \tilde{y}_{M,1}^{(0)*}(k) & \tilde{y}_{M,2}^{(0)*}(k) & \cdots & \tilde{y}_{M,(M-1)}^{(0)*}(k) & \tilde{y}_{M,M}^{(0)*}(k) \end{bmatrix} = \\
 = \begin{bmatrix} \lambda^{0.5} \hat{R}_{M,11}(k) & \lambda^{0.5} \hat{R}_{M,12}(k-1) & \cdots & \lambda^{0.5} \hat{R}_{M,1(M-1)}(k-1) & \lambda^{0.5} \hat{R}_{M,1M}(k-1) \\ 0 & \lambda^{0.5} \hat{R}_{M,22}(k-1) & \cdots & \lambda^{0.5} \hat{R}_{M,2(M-1)}(k-1) & \lambda^{0.5} \hat{R}_{M,2M}(k-1) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \hat{R}_{M,(M-1)(M-1)}(k-1) & \hat{R}_{M,(M-1)M}(k-1) \\ 0 & 0 & \cdots & 0 & \hat{R}_{M,MM}(k-1) \\ 0 & \tilde{y}_{M,2}^{(1)*}(k) & \cdots & \tilde{y}_{M,(M-1)}^{(1)*}(k) & \tilde{y}_{M,M}^{(1)*}(k) \end{bmatrix}. \tag{19}
 \end{aligned}$$

In equation (19), vector $\tilde{\mathbf{Y}}_N^{(0)H}(k)$ is defined as

$$\tilde{\mathbf{Y}}_N^{(0)H}(k) = [\tilde{y}_{M,1}^{(0)*}(k), \tilde{y}_{M,2}^{(0)*}(k), \dots, \tilde{y}_{M,M}^{(0)*}(k)] = \mathbf{Y}_N^H(k). \tag{20}$$

The superscript in parentheses means the number of the transformation performed on variable $\tilde{y}_{M,i}^{(0)*}(k)$ with number i in the vector. Such a conversion over $\tilde{y}_{M,1}^{(0)*}(k)$ is performed once (after the first time the variable is reset), over $\tilde{y}_{M,2}^{(0)*}(k)$ — twice (after the second time, the variable is reset), etc. and over $\tilde{y}_{M,M}^{(0)*}(k)$ — M times (after the M -th time, the variable is reset).

From equation (19) it follows that

$$c_{M,1}(k) \lambda^{0.5} \hat{R}_{M,11}(k-1) - s_{M,1}^*(k) \tilde{y}_{M,1}^{(0)*}(k) = \hat{R}_{M,11}(k) \tag{21}$$

and

$$s_{M,1}(k) \lambda^{0.5} \hat{R}_{M,11}(k-1) + c_{M,1}(k) \tilde{y}_{M,1}^{(0)*}(k) = 0. \tag{22}$$

Performing similar conversion of all $i = 1, 2, \dots, M$, can be established that

$$c_{M,i}(k) \lambda^{0.5} \hat{R}_{M,ii}(k-1) - s_{M,i}^*(k) \tilde{y}_{M,i}^{(i-1)*}(k) = \hat{R}_{M,ii}(k), \tag{23}$$

$$s_{M,i}(k) \lambda^{0.5} \hat{R}_{M,ii}(k-1) + c_{M,i}(k) \tilde{y}_{M,i}^{(i-1)*}(k) = 0. \tag{24}$$

Then from equation (22) we can determine that

$$s_{M,i}(k) = -c_{M,i}(k) \tilde{y}_{M,i}^{(i-1)*}(k) [\lambda^{0.5} \hat{R}_{M,ii}(k-1)]^{-1},$$

and, given that $c^2 + ss^* = 1$, from the equation

$$\begin{aligned}
 & c_{M,i}^2(k) + c_{M,i}(k)\tilde{y}_{M,i}^{(i-1)*}(k)[\lambda^{0.5}\hat{R}_{M,ii}(k-1)]^{-1}c_{M,i}(k)\tilde{y}_{M,i}^{(i-1)*}(k)[\lambda^{0.5}\hat{R}_{M,ii}(k-1)]^{-1} = \\
 & = c_{M,i}^2(k)[\lambda^{0.5}\hat{R}_{M,ii}(k-1)]^2 + c_{M,i}^2(k)[\tilde{y}_{M,i}^{(i-1)*}(k)\tilde{y}_{M,i}^{(i-1)}(k)] = \\
 & = c_{M,i}^2(k)[\lambda\hat{R}_{M,ii}^2(k-1) + \tilde{y}_{M,i}^{(i-1)*}\tilde{y}_{M,i}^{(i-1)}][\lambda\hat{R}_{M,ii}^2(k-1)]^{-1} = 1
 \end{aligned} \tag{25}$$

can determine that

$$\begin{aligned}
 c_{M,i}(k) &= \sqrt{\lambda\hat{R}_{M,ii}^2(k-1)[\lambda\hat{R}_{M,ii}^2(k-1) + \tilde{y}_{M,i}^{(i-1)*}(k)\tilde{y}_{M,i}^{(i-1)}(k)]^{-1}} = \\
 &= \sqrt{\lambda\hat{R}_{M,ii}^2(k-1)}\sqrt{[\lambda\hat{R}_{M,ii}^2(k-1) + \tilde{y}_{M,i}^{(i-1)*}(k)\tilde{y}_{M,i}^{(i-1)}(k)]^{-1}} = \lambda^{0.5}\hat{R}_{M,ii}(k-1)\hat{R}_{M,ii}^{-1}(k),
 \end{aligned} \tag{26}$$

where

$$\hat{R}_{M,ii}(k) = \sqrt{\lambda\hat{R}_{M,ii}^2(k-1) + \tilde{y}_{M,i}^{(i-1)*}(k)\tilde{y}_{M,i}^{(i-1)}(k)}. \tag{27}$$

It is taken into account that the diagonal elements $\hat{R}_{N,ii}(k)$ of the matrix $\hat{\mathbf{R}}_M(k)$ are real numbers. Then, using (26) in equation (24), we can determine

$$\begin{aligned}
 s_{N,i}(k) &= -\lambda^{0.5}\tilde{R}_{M,ii}(k-1)\tilde{R}_{M,ii}^{-1}(k) + \tilde{y}_{M,i}^{(i-1)*}(k)[\lambda^{0.5}\tilde{R}_{M,ii}^2(k-1)]^{-1} = \\
 &= -\tilde{y}_{M,i}^{(i-1)*}(k)\tilde{R}_{M,ii}^{-1}(k).
 \end{aligned} \tag{28}$$

Thus, equations (26)–(28) allow us to calculate the cosine and sine of a certain rotation angle. According to (19), they calculate the elements of the i -th row of matrix $\hat{\mathbf{R}}_M(k)$, zero out the i -th element of vector $\tilde{\mathbf{Y}}_M^{(i-1)H}(k)$, and modify the remaining nonzero elements of this vector as $[0, 0, \dots, 0, \tilde{y}_{M,i+1}^{(i)*}(k), \dots, \tilde{y}_{M,M}^{(i)*}(k)]$, i.e.

$$\tilde{\mathbf{Y}}_M^{(i-1)H}(k) = [0, 0, \dots, 0, \tilde{y}_{M,i}^{(i-1)*}(k), \tilde{y}_{M,i+1}^{(i-1)*}(k), \dots, \tilde{y}_{M,M}^{(i-1)*}(k)] \tag{29}$$

and

$$\tilde{\mathbf{Y}}_M^{(i)H}(k) = [0, 0, \dots, 0, \tilde{y}_{M,i+1}^{(i)*}(k), \dots, \tilde{y}_{M,M}^{(i)*}(k)]. \tag{30}$$

These transformations for each value of i include calculations (26)–(28), and for all $j = i + 1, \dots, M$, calculations similar to (23) and (24), i.e.

$$\hat{R}_{M,ij}(k) = c_{M,i}(k)\lambda^{0.5}\hat{R}_{M,ij}(k-1) - s_{M,i}^*(k)\tilde{y}_{M,j}^{(i-1)*}(k) \tag{31}$$

and

$$\tilde{y}_{M,j}^{(i)*}(k) = s_{M,i}(k)\lambda^{0.5}\hat{R}_{M,ij}(k-1) + c_{M,i}(k)\tilde{y}_{M,j}^{(i-1)*}(k). \tag{32}$$

Thus, transformation (13) can be performed either as $\mathbf{Q}_k^H(k)\mathbf{\Lambda}_k^{0.5}(k)\mathbf{Y}_{Mk}^H(k)$, applying a $k \times k$ matrix $\mathbf{Q}_k^H(k)$ to a $k \times M$ matrix $\mathbf{\Lambda}_k^{0.5}(k)\mathbf{Y}_{Mk}^H(k)$ at each iteration, or using a matrix

$$\hat{\mathbf{Q}}_k^H(k) = \prod_{i=1}^M \mathbf{G}_M^{(i)}(k) \text{ applied to a } k \times M \text{ matrix } \begin{bmatrix} \lambda^{0.5}\hat{\mathbf{R}}_M(k-1) \\ \mathbf{0}_{(k-1-M)M} \\ \mathbf{Y}_M^H(k) \end{bmatrix}.$$

2. Parallel implementation of the QR algorithm in a triangular systolic array

Givens transformation has good properties for use in a triangular systolic array. The architecture of the basic computations of the algorithm using such calculators is given in [16] and in Fig. 1.

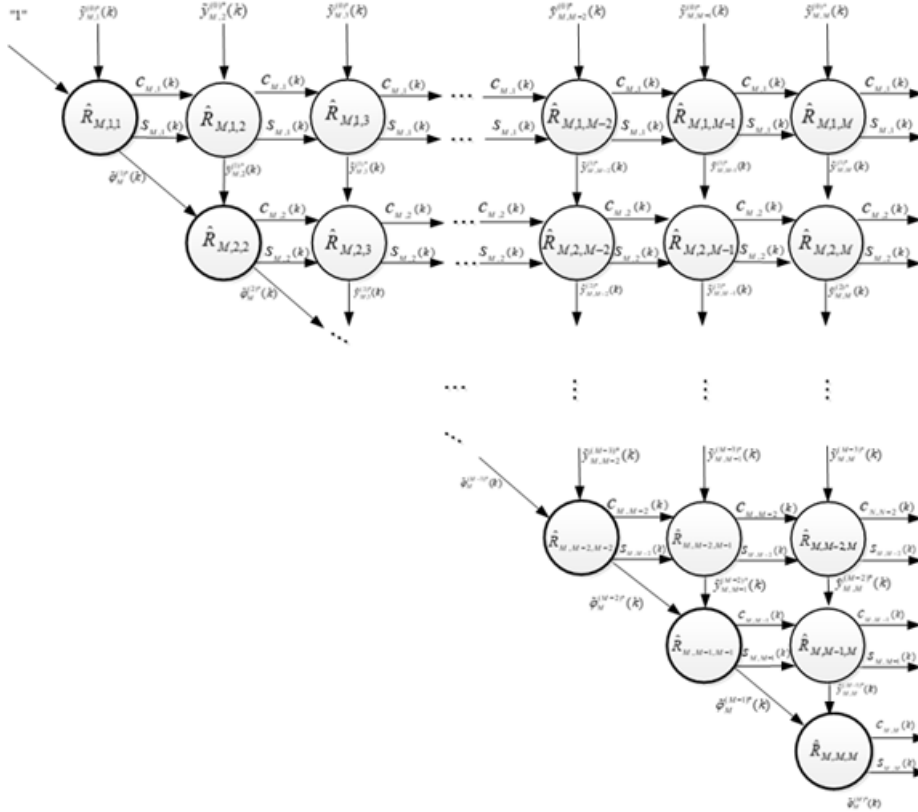


Fig. 1. Block diagram of a triangular systolic array

The systolic array is based on the method of triangular complex rotations and allows to obtain a significant performance gain in comparison with the method of complex rotations of Givens.

In the system of a triangular systolic array, there are individual processing cells located in an ordered structure. Each individual cell of the system has its own processing functionality and local memory. Only neighboring cells are connected to each other and there is no direct connection between cells that are not adjacent. When data is fed into the systolic array system, the processing cells at the front end of the system will process the data, store them in local memory, and then forward them to adjacent cells. This processing and transfer of the processed data in each cell continues until the data stream reaches the end of the system, where the final calculation results are obtained. The proposed architecture provides a significant reduction in the time required to perform QR decomposition using the same computing resources (CORDIC computational cells). Another advantage of the proposed scheme is that during QR decomposition, the upper triangular matrix \mathbf{R} has only real diagonal elements. This simplifies the subsequent inversion of the matrix \mathbf{R} using the backward substitution algorithm, which requires division by the diagonal elements of the matrix \mathbf{R} .

Algorithm (15) shows that the application of Givens rotations in a post-array to multiply matrices on the left side of the preliminary array allows one to obtain a triangular matrix and zero out the input vector $\mathbf{Y}(k)$. The number of elements in the input vector $\mathbf{Y}(k)$ corresponds to the number of antennas. The number of required Givens rotation operations is the same as the number of elements in the input. That is, each rotation of Givens will reset exactly one element of the input vector. Thus, the algorithm for generating a radiation pattern with K antennas requires the Givens rotation operation K in the calculation of the post-array.

In calculations, the Givens rotation operation can be performed in parallel, because there is no data dependence between the Givens rotation operation at one of the inputs and the Givens rotation operation at the same position in subsequent iterations. Thus, Givens rotation operations can be performed in parallel.

Conclusion

The proposed architecture of the triangular systolic array using the method of the triangular complex rotations optimized for implementation in large-scale integrated circuits, allowing you to effectively perform the operation QR-decomposition of complex matrices. Compared with the QR-RLS algorithm, the proposed architecture can provide a gain of up to 35% in the time of calculating the QR decomposition. The synthesized algorithm will make it possible to implement the methods of spatio-temporal processing of broadband signals of satellite communication systems.

This work was supported by the Ministry of Science and Higher Education of the Russian Federation in the framework of the Federal target program «Research and development on priority directions of development of the scientific-technological complex of Russia for 2014-2020» (agreement no. 05.605.21.0185, unique ID project RFMEFI60519X0185).

References

- [1] Ya.D.Shirman, Radio-electronic systems: Fundamentals of construction and theory. Directory. Ed. 2nd, rev. and add., Moscow, Radio Engineering, 2007 (in Russian).
- [2] V.N.Tyapkin, I.N.Kartsan, D.D.Dmitriev, S.V.Efremova, Algorithms for adaptive processing of signals in a flat phased antenna array, International Siberian Conference on Control and Communications (SIBCON), 2017.
- [3] A.I.Perov, Statistical theory of radio systems, Moscow, Radio Engineering, 2003 (in Russian).
- [4] I.N.Kartsan, V.N.Tyapkin, D.D.Dmitriev, A.E.Goncharov, P.V.Zelenkov, I.V.Kovalev, *IOP Conf. Series: Materials Science and Engineering*, **155**(2016) no. 1.
DOI: 10.1088/1757-899X/155/1/012019
- [5] I.N.Kartsan, V.N.Tyapkin, D.D.Dmitriev, A.E.Goncharov, I.V.Kovalev, *IOP Conf. Series: Materials Science and Engineering*, **255**(2017). DOI: 10.1088/1757-899X/255/1/012009
- [6] V.N.Tyapkin, D.D.Dmitriev, Yu.L.Fateev, N.S.Kremez, *Journal of Siberian Federal University. Mathematics & Physics*, **9**(2016), no. 2, 258–268.
DOI: 10.17516/1997-1397-2016-9-2-258-267

- [7] I.N.Kartsan, Y.L.Fateev, V.N.Tyapkin, D.D.Dmitriev, A.E.Goncharov, P.V.Zelenkov, I.V.Kovalev, *IOP Conf. Series: Materials Science and Engineering*, **155**2016, no. 1.
DOI: 10.1088/1757-899X/155/1/012020
- [8] R.A.Monzingo, T.W.Miller, Introduction to Adaptive Arrays, New York, John Wiley & Sons, 1980.
- [9] Ya.D.Shirman et al., The first domestic studies of the adaptation of antenna systems to interfering influences, Moscow, Radio engineering, no. 11, 1989 (in Russian).
- [10] V.N.Tyapkin, D.D.Dmitriev, V.G.Konnov, A.N.Fomin, *Bulletin of the Siberian state Aerospace University named after Acad. M. F. Reshetneva*, **43**(2012), no. 3, 76–79.
- [11] V.I.Dzhigan, *Telecommunications*, **6**(2006), 6–11 (in Russian).
- [12] I.A.Lubkin, V.N.Tyapkin, *Bulletin of the Siberian state Aerospace University named after Acad. M.F. Reshetneva*, **28**(2010), no. 2, 39–43 (in Russian).
- [13] G.H.Golub, C.F.Van, Lone Matrix calculations, The Johns Hopkins University Press, Baltimore, 1996.
- [14] D.T.M.Slock, T.Kailath, *IEEE Trans. Signal Processing*, **39**(1991), no. 1, 92–114.
- [15] A.Benallal, A.Gillioire, A new method to stabilize fast RLS algorithm based on the first-order model of the propagation of numerical errors, Proceedings of the International Conference on Acoustic, Speech and Signal Processing, Vol. 5, 1988, 1373–1376.
- [16] J.G.McWhirter, Recursive least-squares minimization using a systolic array, Proceedings of the SPIE Intern. Sic. Opt. Eng. Vol. 43, 1983, 105–112.

Рекурсивный алгоритм оценивания корреляционной матрицы помех, основанный на QR-разложении

Валерий Н. Тяпкин
Дмитрий Д. Дмитриев
Андрей Б. Гладышев
Пётр Ю. Зверев
Сибирский федеральный университет
Красноярск, Российская Федерация

Аннотация. Многие задачи цифровой обработки сигналов требуют выполнения матричных операций в режиме реального времени. Это операции обращения матрицы или решения систем линейных алгебраических или дифференциальных уравнений (фильтр Калмана). Переход к реализации цифровой обработки сигналов на программируемых логических интегральных схемах (ПЛИС), как правило, предполагает расчеты, основанные на представлении чисел с фиксированной точкой. Это делает практически невозможным решение задач пространственно-временной обработки на основе традиционных вычислительных методов. В статье рассматривается реализация алгоритмов пространственно-временной обработки сигналов в широкополосных спутниковых системах с использованием QR-разложения. Представлены технологии вычислений CORDIC, необходимые для повторного QR-разложения при совместном использовании в систолических алгоритмах.

Ключевые слова: фазированная антенная решетка, адаптивные алгоритмы, фильтр Калмана, рекурсивный алгоритм по критерию наименьших квадратов, QR-разложение, систолические алгоритмы.