

УДК 519.873

The Synthesis Algorithm for Spatial Filtering to Maintain a Constant Level of the Useful Signal

Valeriy N. Tyapkin*

Dmitry D. Dmitriev†

Yuri L. Fateev‡

Nikolay S. Kremez§

Military Engineering Institute

Siberian Federal University

Svobodny, 79, Krasnoyarsk, 660041

Received 10.22.2015, received in revised form 26.02.2016, accepted 22.03.2016

The article discusses the theoretical basis of the two classical recursive algorithms beamforming - algorithm of the Kalman filter and least squares algorithm based on QR decomposition. On the basis of these two algorithms synthesized algorithm to minimize noise at the output of the antenna array, which allows to maintain a constant level of the received useful signal.

Keywords: phased array, adaptive algorithms, Kalman filter, recursive least squares method, QR decomposition.

DOI: 10.17516/1997-1397-2016-9-2-258-268.

1. Introduction

Modern navigation receivers global satellite systems have a significant drawback – low interference immunity [1]. To solve this problem, radio interference suppressed navigation receivers are based on phased array antennas using adaptive beamforming techniques.

Pattern control is widely used in applications such as navigation, radar, wireless communication and others. Currently, the most popular methods of digital typed beamforming. Their main advantage is the possibility of formation of special types of the antenna system navigation receiver — lows (failures) in the appropriate direction to the source of interference and highs in the direction of the useful signal. Due to this compensated (suppressed) interference and the useful signal is accumulated from directions other than the interference. The overall effect of this treatment is determined by the level of use and completeness of the differences, as well as taking into account the quality of each of them, depending on the degree of knowledge the statistical characteristics of the signals and interference.

Since beamforming takes place in digital form, it is necessary to use digital techniques could be performed in real time. Therefore, the solution of this task occurs both in hardware and in software.

*tyapkin58@mail.ru

†dmitriev121074@mail.ru

‡fateev_yury@inbox.ru

§Nkremez@sfu-kras.ru

© Siberian Federal University. All rights reserved

The article deals with the theoretical foundations of digital beamforming algorithms: recursive algorithm Kalman filtering and recursive least squares algorithm based on QR decomposition (QR-RLS)[2].

On the basis of these two algorithms in an article synthesized algorithm to minimize noise at the output of the antenna array, which allows to maintain a constant level of the desired signal.

2. Kalman filtering algorithm

Kalman filtering algorithm is an important adaptive digital signal processing algorithms. It is an efficient recursive filter estimating the state vector of the dynamic system, and uses a series of incomplete and noisy measurements. Kalman filter for recursive estimation of the state vector of the dynamic system known a priori, that is to calculate the current state of the system is necessary to know the current measurement and the previous state of the filter. Thus, the Kalman filter, like other recursive filters implemented in the time, rather than in the frequency domain, but, unlike other similar filters, Kalman filter operates not only estimates of the states, and more and uncertainty estimates (distribution density) of the state vector, based on Bayes' formula of conditional probability.

Traditional Kalman filter has the following parameters:

$$\begin{aligned}\mathbf{K}(n) &= \mathbf{K}(n, n-1) - \mathbf{F}(n, n+1) \cdot \mathbf{G}(n) \cdot \mathbf{C}(n) \cdot \mathbf{K}(n, n-1), \\ \mathbf{G}(n) &= \mathbf{F}(n+1, n) \cdot \mathbf{K}(n, n-1) \cdot \mathbf{C}^H \cdot [\mathbf{C}(n) \cdot \mathbf{K}(n, n-1) \cdot \mathbf{C}^H(n) + \mathbf{Q}_2(n)]^{-1}, \\ \alpha(n) &= \mathbf{y}(n) - \mathbf{C}(n) \cdot \mathbf{x}(n|y_{n-1}),\end{aligned}\quad (1)$$

where $\mathbf{C}(n)$ is observation matrix, $\mathbf{F}(n, n+1)$ is state transition matrix.

For variant not induced dynamic model of the noise process converges to white noise with zero mean and unit variance. Therefore, some parameters of the state space converge to a constant and can be prepared as follows:

$$\begin{aligned}\mathbf{F}(n+1, n) &= \sqrt{\lambda} \mathbf{I}, \quad \mathbf{C}(n) = \mathbf{u}^H(n), \quad \mathbf{Q}_2(n) = 1, \\ \mathbf{K}(n) &= \frac{\mathbf{K}(n-1)}{\lambda} - \frac{\mathbf{g}(n) \cdot \mathbf{u}(n)^H \cdot \mathbf{K}(n-1)}{\sqrt{\lambda}}, \\ \mathbf{g}(n) &= \frac{\mathbf{K}(n-1) \cdot \mathbf{u}(n)}{\sqrt{\lambda} \left(1 + \mathbf{u}(n)^H \cdot \mathbf{K}(n-1) \cdot \mathbf{u}(n) \right)}, \\ \alpha(n) &= y(n) - \mathbf{u}^H(n) \cdot \mathbf{x}(n|y_{n-1}).\end{aligned}\quad (2)$$

By manipulating the expression obtained in (2), we can get the parameters for Kalman filter:

$$\begin{aligned}\mathbf{g}(n) &= \frac{\mathbf{K}^{-1}(n) \cdot \mathbf{u}(n)}{\sqrt{\lambda}}, \\ \mathbf{K}^{-1}(n) &= \lambda \cdot \mathbf{K}^{-1}(n-1) + \mathbf{u}(n) \cdot \mathbf{u}(n)^H, \\ \mathbf{K}^{-1}(n) \cdot \mathbf{x}(n+1|y_n) &= \frac{\mathbf{K}^{-1}(n-1) \cdot \mathbf{x}(n|y_{n-1})}{\sqrt{\lambda}} + \frac{\mathbf{u}(n) \cdot y(n)}{\sqrt{\lambda}}.\end{aligned}\quad (3)$$

Then, by rewriting the expression of the $\mathbf{K}^{-1}(n) = \mathbf{K}^{-H/2}(n) \cdot \mathbf{K}^{-1/2}$ and using the lemma on matrix inversion, we can determine the Kalman filter in the form of pre-array and post-array

submatrix replaced $\tilde{\mathbf{G}}_i = \begin{pmatrix} c & s \\ -s & c \end{pmatrix}$ with the element c and s , which satisfy the condition

$$s^2 + c^2 = 1. \quad (8)$$

With this condition, the normalization matrix $\tilde{\mathbf{G}}_i$ and, respectively, \mathbf{G}_i are orthogonal matrix and, moreover, justifies the name \mathbf{G}_i performed by the conversion namely converting the rotation plane, since the elements c and s can be interpreted as a cosine and sine of the angle of rotation.

By imposing conditions on the c and s in the transformation Givens, seeking to ensure that by a sequence of orthogonal transformation matrices $\mathbf{G}_1, \mathbf{G}_2, \dots, \mathbf{G}_{n-1}$ of the form (8), we managed to reduce the matrix to the right triangular form, consistently nullifying subdiagonal elements first, second, ..., $(n-1)$ -th columns.

Further, using these expressions, proceed directly to the synthesis QR-RLS algorithm.

Assume that the navigation signal amplitude significantly lower than the intensity interference, so the signal does not have significant influence on the estimate of the correlation matrix of interference $\bar{\Phi}$.

Suppose that at the inputs M -element phased arrays to interference with the instantaneous values of the oscillation $u_1(t), u_2(t), \dots, u_M(t)$, phase shifted relative to each other due to the path difference ΔD from element to element by an amount $\Delta\varphi = \frac{2\pi}{\lambda} \cdot \Delta D$. Interfering signals at the output of the i -th element phased array can be represented as a set of discrete samples of instantaneous amplitude u_{il} with sampling period $T_D = 1/2f_m$ (f_m – the maximum frequency in the spectrum of the received signal interference) in accordance with the Kotel'nikov theorem (Fig. 1). Here, $i = 1, 2, \dots, M$; $l = 1, 2, \dots, k$ are the number of discrete samples.

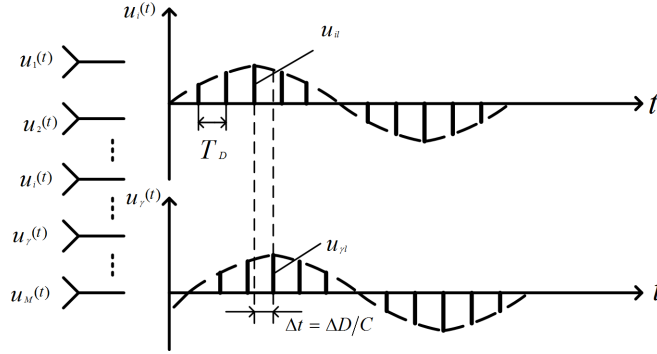


Fig. 1. Sampling signal interference

The cross-correlation between the signals of the channels i and j is defined as the arithmetic mean of the correlation points $u_i u_j$ by the number of counts l from 1 to $k - \varphi_{ij} = \frac{1}{k} \sum_{l=1}^k u_{il} u_{jl}$.

Introducing the complex amplitudes of the instantaneous values of discrete \mathbf{u}_{il} and \mathbf{u}_{jl} integrated mutual correlation time is defined as

$$\hat{\mathbf{V}} = \frac{1}{k} \sum_{l=1}^k u_{il} u_{jl}^* / 2. \quad (9)$$

Weight vector at stage n is defined as:

$$\mathbf{w}(n) = [w_0(n), w_1(n), \dots, w_{M-1}(n)]^T. \quad (10)$$

The objective function for the RLS algorithm is defined as:

$$\delta(n) = \sum_{i=1}^n \lambda^{n-i} |e(i)|^2, \quad (11)$$

where $e(i) = d(i) - \mathbf{w}^H(n) \cdot \mathbf{u}(i)$. Let \mathbf{w}^H be optimum value, at which the target function $\delta(n)$ is minimized can be determined in the normal equation form as follows [4]:

$$\Phi(n) \cdot \mathbf{w}(n) = \mathbf{z}(n), \quad (12)$$

where $\Phi(n) = \sum_{i=1}^n \lambda^{n-1} \cdot \mathbf{u}(i) \cdot \mathbf{u}(i)^H$, $\mathbf{z}(n) = \sum_{i=1}^n \lambda^{n-1} \cdot \mathbf{u}(i) \cdot \mathbf{d}^*(i)$.

The last two equations can be written in a recursive form, as follows:

$$\begin{aligned} \Phi(n) &= \lambda \cdot \Phi(n-1) + \mathbf{u}(n) \cdot \mathbf{u}(n)^H, \\ \mathbf{z}(n) &= \lambda \cdot \mathbf{z}(n-1) + \mathbf{u}(n) \cdot \mathbf{d}^*(n). \end{aligned} \quad (13)$$

For the next step, use the lemma on the treatment of matrices. Matrices $\mathbf{A} + \alpha \mathbf{B} \mathbf{B}^{*T}$ Duayra treated according to the rule [5]:

$$(\mathbf{A} + \alpha \mathbf{B} \mathbf{B}^{*T})^{-1} = \mathbf{A}^{-1} - \frac{\alpha \mathbf{A}^{-1} \mathbf{B} \mathbf{B}^{*T} \mathbf{A}^{-1}}{1 + \alpha \mathbf{B}^{*T} \mathbf{A}^{-1} \mathbf{B}}. \quad (14)$$

Using this ratio, we get

$$\mathbf{P}(n) = \frac{\mathbf{P}(n-1)}{\lambda} - \frac{\mathbf{k}(n) \cdot \mathbf{u}(n)^H \cdot \mathbf{P}(n-1)}{\lambda}, \quad (15)$$

where $\mathbf{P}(n) = \Phi^{-1}(n)$, $\mathbf{k}(n) = \frac{\lambda^{-1} \cdot \mathbf{P}(n-1) \cdot \mathbf{u}(n)}{1 + \lambda^{-1} \cdot \mathbf{u}(n)^H \cdot \mathbf{P}(n-1) \cdot \mathbf{u}(n)} = \mathbf{P}(n) \cdot \mathbf{u}(n) = \Phi^{-1}(n) \cdot \mathbf{u}(n)$.

With the above relations can calculate the expression for the weight vector of the constraint equation (4):

$$\begin{aligned} \mathbf{w}(n) &= \Phi^{-1}(n) \cdot \mathbf{z}(n), \\ \mathbf{w}(n) &= \mathbf{P}(n) \cdot \mathbf{z}(n), \\ \mathbf{w}(n) &= \lambda \cdot \mathbf{P}(n) \cdot \mathbf{z}(n-1) + \mathbf{P}(n) \cdot \mathbf{u}(n) \cdot \mathbf{d}^*(n), \\ \mathbf{w}(n) &= \mathbf{P}(n-1) \cdot \mathbf{z}(n-1) - \mathbf{k}(n) \cdot \mathbf{u}(n)^H \cdot \mathbf{P}(n-1) \cdot \mathbf{z}(n-1) + \mathbf{P}(n) \cdot \mathbf{u}(n) \cdot \mathbf{d}^*(n), \\ \mathbf{w}(n) &= \mathbf{w}(n-1) - \mathbf{k}(n) \cdot \mathbf{u}(n)^H \cdot \mathbf{w}(n-1) + \mathbf{P}(n) \cdot \mathbf{u}(n) \cdot \mathbf{d}^*(n), \\ \mathbf{w}(n) &= \mathbf{w}(n-1) + \mathbf{k}(n) \cdot \xi^*(n), \end{aligned} \quad (16)$$

where $\xi^*(n) = d(n) - \mathbf{w}^H(n-1) \cdot \mathbf{u}(n)$ From equation (15) and equation (3) is observed correspondence between the RLS algorithm and Kalman filter. Rewriting the previously obtained expression

$$\Phi(n) = \Phi^{-\frac{1}{2}}(n) \cdot \Phi^{-\frac{H}{2}}(n), \quad (17)$$

and entering a new variable

$$\mathbf{p}(n) = \Phi^{\frac{H}{2}}(n) \cdot \mathbf{w}(n) = \Phi^{-\frac{1}{2}}(n) \cdot \mathbf{z}(n). \quad (18)$$

RLS algorithm can be represented as an array of pre-and post-array, similar to the previously described algorithm Kalman:

$$\begin{bmatrix} \sqrt{\lambda} \cdot \Phi^{\frac{1}{2}}(n-1) & \mathbf{u}(n) \\ \sqrt{\lambda} \cdot \mathbf{p}^H(n-1) & d(n) \\ \mathbf{0}^T & 1 \end{bmatrix} \theta(n) = \begin{bmatrix} \Phi^{\frac{1}{2}}(n) & \mathbf{0} \\ \mathbf{p}^H(n) & \frac{\xi(n)}{\sqrt{\gamma(n)}} \\ \mathbf{u}^H(n) \cdot \Phi^{-\frac{H}{2}}(n) & \sqrt{\gamma(n)} \end{bmatrix}, \quad \mathbf{w}^H(n) = \mathbf{p}^H(n) \cdot \Phi^{-\frac{1}{2}}(n). \quad (19)$$

QR-RLS algorithm is used to solve the problem of adaptive beamforming due to its recursive nature, efficient computing structures and numerical stability [6].

4. An algorithm for minimizing the noise at the output of the antenna array so as to maintain a constant level of the useful signal

One of the most promising methods for adaptation of the antenna arrays is currently the method of minimum variance distortionless response (MVDR) the antenna array. MVDR algorithm allows you to gain in the signal / noise ratio in the desired direction while suppressing the deep interference received signals, combining the merits of QR-RLS algorithm.

The problem of minimizing the noise at the antenna array while maintaining a constant level of the useful signal can be formulated as follows:

$$\min_{\mathbf{w}(n)} \sum_{i=1}^n \lambda^{n-i} |e(i)|^2, \quad (20)$$

where $e(i) = \mathbf{w}^H(n) \cdot \mathbf{u}(i)$, under restriction

$$\mathbf{w}^H(n) \cdot \mathbf{s}(\theta_0) = 1, \quad (21)$$

where $\mathbf{s}(\theta_0)$ is vector determining the direction of the source signal.

If the constraints (21), a weight vector for the antenna array can be written as follows [7]:

$$\mathbf{w}(n) = \frac{\Phi^{-1}(n) \cdot \mathbf{s}(\theta_0)}{\mathbf{s}^H(\theta_0) \cdot \Phi^{-1}(n) \cdot \mathbf{s}(\theta_0)}. \quad (22)$$

Define additional vector as $\alpha(n) = \Phi^{-\frac{1}{2}}(n) \cdot \mathbf{s}(\theta_0)$ we can rewrite the expression for the weight vector and error estimates, and write a new expression of error estimates as follows:

$$\begin{aligned} \mathbf{w}(n) &= \frac{\Phi^{-\frac{H}{2}}(n) \cdot \alpha(n)}{\|\alpha(n)\|^2}, \\ \mathbf{e}(n) &= \frac{\alpha^H(n) \cdot \Phi^{-\frac{1}{2}}(n) \cdot \mathbf{u}(n)}{\|\alpha(n)\|^2}, \\ \mathbf{e}'(n) &= \alpha^H(n) \cdot \Phi^{-\frac{1}{2}}(n) \cdot \mathbf{u}(n). \end{aligned} \quad (23)$$

With the set parameters of the algorithm for solving the problem MVDR is possible to solve the problem of maintaining a constant level the useful signal in the QR-RLS algorithm, and present it in the form of an array of pre-and post-array as follows:

$$\begin{bmatrix} \sqrt{\lambda} \cdot \Phi^{\frac{1}{2}}(n-1) & \mathbf{u}(n) \\ \sqrt{\lambda} \cdot \alpha^H(n-1) & d(n) \\ \mathbf{0}^T & 1 \end{bmatrix} \theta(n) = \begin{bmatrix} \Phi^{\frac{1}{2}}(n) & \mathbf{0} \\ \alpha^H(n) & \frac{-e'(n)}{\sqrt{\gamma(n)}} \\ \mathbf{u}^H(n) \cdot \Phi^{-\frac{H}{2}}(n) & \sqrt{\gamma(n)} \end{bmatrix}. \quad (24)$$

After calculating the parameter using (24) we can estimate the error:

$$e(n) = \frac{-(-e'(n) \cdot \sqrt{\gamma(n)})}{\|\mathbf{a}(n)\|^2}. \quad (25)$$

5. Submission MVDR algorithm as a systolic array

From these expressions it can be concluded that the calculation of the response phased array antenna using MVDR-algorithm or its precursor QR-RLS algorithm includes a matrix multiplication, which involves a series of Givens rotations. It is known that the consistent implementation of the matrix multiplication are generally inefficient and slow. For signal processing systems in real time, using the QR-RLS and MVDR beamforming algorithms, a more efficient method of computing.

In 1978, Kung and Leiserson offered systolic arrays for matrix calculations in systems of signal processing on VLSI. The systolic array based on the method of triangular complex rotations and provides a significant gain productivity calculations compared with the traditionally used method of complex Givens rotations.

The systolic array system, there are individual processing cells arranged in a specific structure. Each individual cell has its own system of processing functionality and its own local memory. Furthermore, only the neighboring cells are connected to each other, and there is no direct connection between the cells that are not adjacent.

Thus, when data is written in a systolic array of processing cells on the front of the system will process the data necessary to store data in its own local memory, and then transmit towards its neighboring cells. Cells that receive forwarded data from the interface elements, in turn, process the data, store the necessary data and send the results to its neighboring cells. This processing and sending the processed data in each cell lasts as long as the data stream reaches the end of the system, where the final results of calculation are removed. Thus, the processing passes through the system in the same manner as the blood of a human heart rhythmically fed through the vessels to the necessary body organs. Therefore, this processing system, by analogy with the human body, called a systolic array processor. The proposed architecture allows a significant reduction in the time required to perform QR-decomposition using the same computational resources (cell CORDIC computation). A second advantage of the proposed scheme is that the QR-decomposition is executed so that the upper triangular matrix \mathbf{R} has only the diagonal elements are real. This greatly simplifies the subsequent handling of the matrix \mathbf{R} using back-substitution algorithm, which requires dividing by the diagonal elements of the matrix \mathbf{R} .

The algorithm MVDR (24) by multiplying the matrix an pre-array with a number of Givens rotations in the post array turns lower triangular matrix and reset input vector $\mathbf{u}(n)$. The number of elements in the input vector $\mathbf{u}(n)$ corresponds to the number of antennas in the case of the beamforming algorithm MVDR.

To zero the elements of the input vector, it is necessary to apply a series of operations Givens rotations. The number of operations required is the number of input elements, because each Givens rotation resets one element of the input vector. Thus, in general, MVDR beamforming algorithm with K antennas are required to K Givens rotation operation based post-array.

In the calculations of the Givens rotation operation can be performed in parallel since there is no data dependency between the operation Givens rotation on one of the inputs, and the Givens rotation operation at the same positions in subsequent iterations. That is, the calculation of the

Givens rotation to a single element of the input vector does not require the calculation results to other elements.

When considering the Givens rotation operations described expressions (7) and (8), it is evident that for each Givens rotation sequence of the corresponding entry in the array of post-change only certain elements of the pre-array, and the remaining elements are not changed. To illustrate the above, we present a step-by-step operation of calculation algorithm for MVDR the antenna array consisting of three elements.

Step 1:

$$\begin{bmatrix} \Phi_{1,1} & 0 & 0 & u_1 \\ \Phi_{2,1} & \Phi_{2,2} & 0 & u_2 \\ \Phi_{3,1} & \Phi_{3,2} & \Phi_{3,3} & u_3 \\ \alpha_1 & \alpha_2 & \alpha_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos & 0 & 0 & -\sin' \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \sin & 0 & 0 & \cos \end{bmatrix}, \quad (26)$$

where

$$\begin{aligned} \Phi'_{1,1} &= (\Phi_{1,1} \cdot \cos) + (u_1 \cdot \sin), \\ u'_1 &= (\Phi_{1,1} \cdot (-\sin')) + (u_1 \cdot \cos) = 0, \\ \Phi'_{2,1} &= (\Phi_{2,1} \cdot \cos) + (u_2 \cdot \sin), \\ u'_2 &= (\Phi_{2,1} \cdot (-\sin')) + (u_2 \cdot \cos), \\ \Phi'_{2,2} &= \Phi_{2,2}, \\ \Phi'_{3,1} &= (\Phi_{3,1} \cdot \cos) + (u_3 \cdot \sin), \\ u'_3 &= (\Phi_{3,1} \cdot (-\sin')) + (u_3 \cdot \cos), \\ \Phi'_{3,2} &= \Phi_{3,2}, \\ \Phi'_{3,3} &= \Phi_{3,3}, \\ \alpha'_1 &= \alpha_1 \cdot \cos, \\ \alpha'_2 &= \alpha_2, \\ \alpha'_3 &= \alpha_3, \\ \beta' &= \alpha_1 \cdot (-\sin'). \end{aligned}$$

Step 2:

$$\begin{bmatrix} \Phi'_{1,1} & 0 & 0 & 0 \\ \Phi'_{2,1} & \Phi'_{2,2} & 0 & u'_2 \\ \Phi'_{3,1} & \Phi'_{3,2} & \Phi'_{3,3} & u'_3 \\ \alpha'_1 & \alpha'_2 & \alpha'_3 & \beta' \\ \beta' & \beta' & \beta' & \beta' \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos & 0 & -\sin' \\ 0 & 0 & 1 & 0 \\ 0 & \sin & 0 & \cos \end{bmatrix}, \quad (27)$$

where

$$\begin{aligned} \Phi''_{1,1} &= \Phi'_{1,1}, \\ u''_1 &= 0, \\ \Phi''_{2,1} &= \Phi'_{2,1}, \\ \Phi''_{2,2} &= (\Phi'_{2,2} \cdot \cos) + (u'_2 \cdot \sin), \\ u''_2 &= (\Phi'_{2,2} \cdot (-\sin')) + (u'_2 \cdot \cos) = 0, \\ \Phi''_{3,1} &= \Phi'_{3,1}, \\ \Phi''_{3,2} &= (\Phi'_{3,2} \cdot \cos) + (u'_3 \cdot \sin), \\ u''_3 &= (\Phi'_{3,2} \cdot (-\sin')) + (u'_3 \cdot \cos), \\ \Phi''_{3,3} &= \Phi'_{3,3}, \\ \alpha''_1 &= \alpha'_1, \\ \alpha''_2 &= (\alpha'_2 \cdot \cos) + (\beta' \cdot \sin), \end{aligned}$$

$$\alpha''_3 = \alpha'_3,$$

$$\beta'' = ((\alpha'_2 \cdot (-\sin^*)) + (\beta' \cdot \cos)).$$

Step 3:

$$\begin{bmatrix} \Phi''_{1,1} & 0 & 0 & 0 \\ \Phi''_{2,1} & \Phi''_{2,2} & 0 & 0 \\ \Phi''_{3,1} & \Phi''_{3,2} & \Phi''_{3,3} & u''_3 \\ \alpha''_1 & \alpha''_2 & \alpha''_3 & \beta'' \\ \beta'' & \beta'' & \beta'' & \beta'' \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos & -\sin' \\ 0 & 0 & \sin & \cos \end{bmatrix}, \quad (28)$$

where

$$\begin{aligned} \Phi'''_{1,1} &= \Phi''_{1,1}, \\ u'''_1 &= 0, \\ \Phi'''_{2,1} &= \Phi''_{2,1}, \\ u'''_2 &= 0, \\ \Phi'''_{2,2} &= \Phi''_{2,2}, \\ \Phi'''_{3,1} &= \Phi''_{3,1}, \\ \Phi'''_{3,2} &= \Phi''_{3,2}, \\ \Phi'''_{3,3} &= (\Phi''_{3,3} \cdot \cos) + (u''_3 \cdot \sin), \\ u'''_3 &= (\Phi''_{3,3} \cdot (-\sin^*)) + (u''_3 \cdot \cos), \\ \alpha'''_1 &= \alpha''_1, \\ \alpha'''_2 &= \alpha''_2, \\ \alpha'''_3 &= (\alpha''_3 \cdot \cos) + (\beta'' \cdot \sin), \\ \beta''' &= ((\alpha''_3 \cdot (-\sin^*)) + (\beta'' \cdot \cos)). \end{aligned}$$

$$\begin{bmatrix} \Phi'''_{1,1} & 0 & 0 & 0 \\ \Phi'''_{2,1} & \Phi'''_{2,2} & 0 & 0 \\ \Phi'''_{3,1} & \Phi'''_{3,2} & \Phi'''_{3,3} & 0 \\ \alpha'''_1 & \alpha'''_2 & \alpha'''_3 & \beta''' \\ \beta''' & \beta''' & \beta''' & \beta''' \end{bmatrix}. \quad (29)$$

Fig. 2 is a block diagram of a triangular systolic array which may be used to calculate the QR-decomposition matrix of dimension 3×3 .

Elementary while the triangular systolic array is called the systolic cycle. During the systolic cycle is executed in a single operation of computer processors and transfer the data to other nodes of the triangular systolic array, and then begins the next systolic cycle. Also in Fig. 2 shows the organization of the input and output data streams. The input vector is loaded with the shift lines in time to one cycle of systole. The square cells form inside cells, responsible for computing the elementary Givens rotations for the matrix elements in accordance with the expression (7).

Also it carried out additional calculations error $e(n)$ according to the expression $e(n) = \frac{-(-e'(n) \cdot \gamma^{-\frac{1}{2}}(n) \cdot \gamma^{\frac{1}{2}}(n))}{||a(n)||^2}$ of the right circle in the cells marked with an "x".

Cells depicted circle, in the standard terminology called border cells, that is, the values calculated on the border of the cell which is the diagonal elements of the transformed matrix.

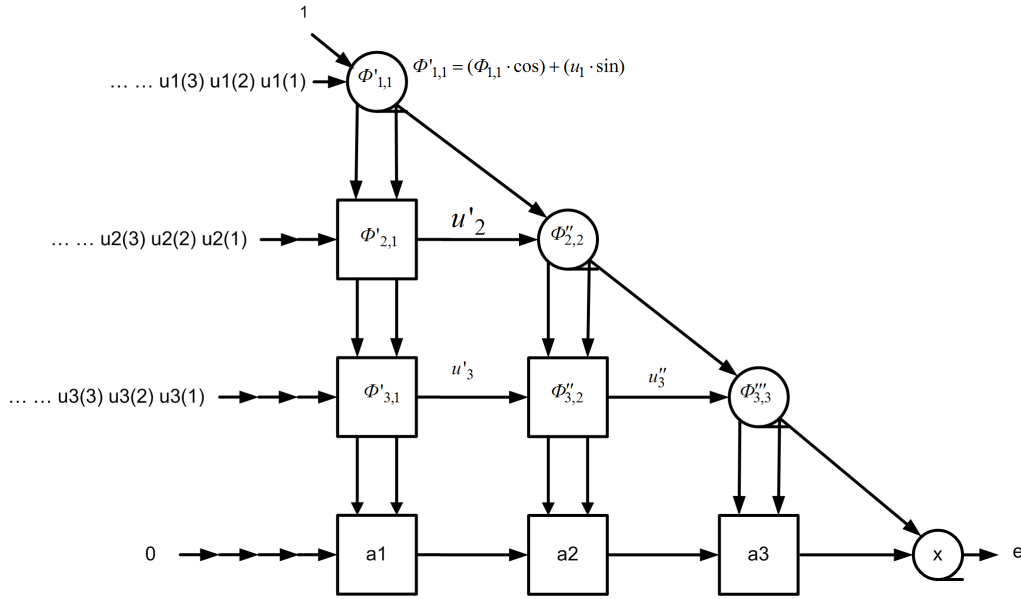


Fig. 2. Block diagram of the triangular array diastolic

Conclusion

Thus, the proposed architecture of the triangular systolic array on the basis of the triangular complex rotations optimized for implementation in large scale integrated circuits, enabling the effective implementation of the operation QR-decomposition of complex matrices. Compared with the QR-RLS algorithm, the proposed architecture can provide a gain of up to 35% of the computation time of QR-decomposition.

The research was financed by the Ministry of Education and Science of the Russian Federation (agreement no. 14.575.21.0081, unique project RFMEFI57514X0081).

References

- [1] V.N.Tyapkin, D.D.Dmitriev, T.G.Moshkina, The potential noise immunity of consumer navigation equipment satellite navigation systems, *Vestnik SibGAU*, **43**(2012), no. 3, 113–119 (in Russian).
- [2] C.Dick, F.Harris, M.Pajic, D.Vuletic, Implementing a Real-Time Beamformer on an FPGA Platform: We designed a flexible QRD-based beamforming engine using Xilinx System Generator, *XCell Journal*, (2007), 36–40.
- [3] D.Bindel, J.Demmel, W.Kahan, On Computing Givens Rotations Reliably and Efficiently., *Acm Trans. Mathematical Software*, (2002), 206–238.
- [4] Ed. A.I.Perov, V.N.Kharisov., GLONASS. Principles of construction and operation. 4th ed., Revised, Moscow, Radio, 2010 (in Russian).
- [5] Ya.D.Shearman, Electronic systems: Basics of construction and theory. Resource Book. Ed. 2nd, Revised. and add., Moscow, Radio, 2007 (in Russian).

- [6] V.N.Tyapkin, D.D.Dmitriev, E.N.Garin, A.V.Sokolovsky, Management of amplitude-phase distributed adaptive phased array , *Industrial Automatic Control Systems and Controllers*, 4(2013), 66–71 (in Russian).
- [7] V.N.Tyapkin, D.D.Dmitriev, V.G Konnov, A.N.Fomin, The method of determining the vector of spectral coefficients of the likelihood ratio, *Vestnik SibGAU*, 43(2012), no. 3, 76–79 (in Russian).

Синтез алгоритма пространственной фильтрации с сохранением постоянного уровня полезного сигнала

**Валерий Н. Тяпкин
Дмитрий Д. Дмитриев
Юрий Л. Фатеев
Николай С. Кремез**

В статье рассматриваются теоретические основы формирования диаграммы направленности на основе двух классических рекурсивных алгоритмов — алгоритм фильтра Калмана и алгоритм наименьших квадратов на основе QR-разложения. На основе этих двух алгоритмов синтезирован алгоритм минимизации шума на выходе антенной решетки, что позволяет поддерживать постоянный уровень принимаемого полезного сигнала.

Ключевые слова: фазированная антенная решетка, адаптивные алгоритмы, фильтр Калмана, рекурсивный метод наименьших квадратов, QR-разложение.