Spectral estimation fast algorithm with longterm numerical stability

V.V.Zaharov (vick@xanum.uam.mx),
F.S.Casco (alfa@xanum.uam.mx).
Departamento de Ingenieria Electrica,
Universidad Autonoma Metropolitana-Iztapalapa,
Mexico D.F., MEXICO

The recursive fast RLS-algorithm of spectral estimation with longterm numerical stability is offered. The numerical stability in offered algorithm is achieved by synthesis of recursive computing procedure of QR Householder decomposition of the input data correlation matrix. It is shown, that offered algorithm saves a longterm numerical stability even if input correlation matrix has bad condition. Applying of the synthesized algorithm for spectral estimation of time series allows to avoid of the spectral density distortion arising in real computing digital systems due to final words length.

I. INTRODUCTION

The basic task of the spectral analysis of time series is the reception of authentic spectral estimations of random process on its unique realization of final duration. From existing methods of spectral estimations the special place occupy methods which are used autoregressive (AR) model. It is explained that AR estimation of spectral density has the sharp peaks and therefore it is possible to receive the higher resolution of harmonic components than classical spectral estimation methods [1]. Let us consider AR–model for an estimation of a forward linear prediction of element x_n, n=1,2...,N+ p

$$\hat{x}_n = -\sum_{k=1}^{p} a_k x_{n-k} , \tag{1}$$

where $\hat{x}_n$ is estimation of element x_n , a_k is parameter of a forward linear prediction, N is number of elements of a time series, p is order of AR – model.
If minimize an error of a linear prediction

$$\left(x_n - \hat{x}_n\right)^2 = \Delta_n^2 \equiv min, \tag{2}$$

then a vector of a linear prediction parameters a_k which minimizes Δ_n^2 is the solution of the normal equations. For this purpose the eq.(1) using eq.(2) may be written as

$$\Delta_n = x_n + \sum_{k=1}^{p} a_p x_{n-k} . \tag{3}$$

Notice that n =1,2...,N + p and x_n = 0 at n < 1, n > N . Then eq.(3) may be written in the matrix form

$$\Delta = X \cdot A, \tag{4}$$

where $\Delta = \left|\Delta_1, \Delta_2, ..., \Delta_N, ..., \Delta_{N+p}\right|^T$, $A = \left|1, a_1, a_2, ..., a_p\right|^T$, T is transpose symbol,

X is $(N+p)\times(p+1)$ matrix of input data with structure as shown below

$$X = \begin{vmatrix} x_1 & 0 & 0 & \ldots & 0 \\ x_2 & x_1 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ x_N & x_{N-1} & x_{N-2} & \ldots & x_{N-p} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & 0 & x_N \end{vmatrix}, \qquad (5)$$

As shown in [1], equation that minimizes the average square of an error has a form

$$RA = P, \qquad (6)$$

where $P = |\rho, 0, \ldots, 0|^T$ is $(p+1)$ vector,

$$\rho = \sum_{n=1}^{N+p} |\Delta_n|^2 = \Delta^{\sim}\Delta, \qquad (7)$$

~ is symbol of transpose and conjugate, $R = X^{\sim} X$ is Toeplitz $(p+1)\times(p+1)$ correlation matrix of input data. The eq.(6) is the Yule–Walker equation for autoregressive process and can be solved concerning of parameters a_i, $i=1,2\ldots, p$ by various known algorithms, for example by Levinson algorithm [1].

However at processing input data in real time the methods of recursive estimation of AR–parameters are more preferable. That allows after receipt of the next input data to update parameters and diagram of an estimated spectral density. Spectral density of input data for the received parameters will be determined by expression [1]

$$S(f) = \frac{T\rho}{\left|1 + \sum_{n=1}^{p} \hat{a}_n e^{-j2\pi f_n T}\right|} \qquad (8)$$

were $\hat{a}_n$ is estimation of parameter a_n, $e^{-j2\pi f_n T}$ is complex harmonic of frequency f_n, T is a sampling interval.

Such procedure allows to trace slowly varied parameters of nonstationary sequences by putting some exponential window which is gone along the input data, creating the least change of the current errors and very strongly reducing of older ones. Thus eq.(7) we put as

$$\rho = \sum_{n=1}^{N+p} w^{N+p-n} |\Delta_n|^2, \qquad (9)$$

were w is positive real scalar satisfying to a condition $0 < w \le 1$.
Then expression for a matrix R we can write

$$R = \sum_{n=1}^{N+p} w^{N+p-n} \chi_n \chi_n^{\sim}, \qquad (10)$$

where $\chi_n = \left| x_n, x_{n-1}, \ldots, x_{n-p} \right|^T$ is n-th row of matrix X, $n = 1,2\ldots, N+p$.

The most simple procedures of the current parameters estimation of AR–model are the algorithms of the least mean squares (LMS algorithm) such as gradient procedure of the quickest descent, for example Griffith´s algorithm [2]. However, strong dependence of convergence rate of these algorithms from the characteristics of an input data, in particular from a correlation matrix condition, does not allow to use ones for tracking parameters of AR–model if the change in time of an input data statistics occurs rather quickly in comparison with convergence rate of these algorithms. For this reason more preferable is using of recursive algorithms of the least squares (RLS algorithms) allowing to receive convergence rate to an optimum estimation of AR–parameters for much smaller number of steps than algorithms LMS and also independence of a correlation matrix condition. Payment for these advantages is the additional computational complexity of RLS algorithms. However high sensitivity of RLS algorithms to final word length can result in bad numerical stability. That means that after performance of the certain number of iterations the output error of the prediction filter can sharply increase (in other words filter has not long-term numerical stability) that can lead to essential distortions of estimated spectral density. In [1],[2] for reduction of influence of bad numerical stability on the received solution is offered to add “white” noise to input data or to add a small constant in a diagonal of a correlation matrix (to make regularization of a matrix). However such methods result in displacement of received estimated spectral density that in some cases is inadmissible. Other method to improve the numerical stability is the choice w<<1, that allows to limit sharply the error on the previous iterations. However, it results in loosing of useful information (in other words parameter w is obliged to trace nonstationarity which is not connected in any way to condition of input correlation matrix). Obviously the most perspective way of increase of numerical stability of recursive algorithms is the synthesis of algorithms using more stability algebraic computing procedures than used in known recursive algorithms RLS the formula Morison–Woodberry for the inversing of the one-rank modified matrixes [1],[2]. Fundamental works on study of influence of effects of final word length in various computing procedures is the works [3], [4], from that follows, that better in computing sense are the Householder transform and modified algorithm of Gram – Schmidt, then follow a Choletsky method and Givens rotation method.

The purpose of the present work is the synthesis of spectral estimation algorithm which has long-term numerical stability having low sensitivity to bad input data correlation matrix condition and computational complexity of one is linear function of N. For this purpose, it is offered recursive Householder transform method which is applied to RLS algorithm of spectral estimation and as consequence allows to define high immunity to final word length errors in recursive RLS procedure [5].

2. SYNTHESIS OF ALGORITHM

For synthesis of algorithm we will write the eq. (6) as

$$LL^{\sim} A = P\,, \tag{11}$$

where L and $L^{\sim}$ accordingly lower and upper decomposition of a matrix R, such that $R = LL^{\sim}$. From eq.(11) follows, that the vector A can be found by the solution of two systems of the linear equations

$$LB = P\,, \qquad L^{\sim} A = B\,. \tag{12}$$

The matrixes L and $L^{\sim}$ can be obtain using Choletsky factorization [4]. However, in this case improvement of numerical stability is insignificant and the matrix R should be formed in an obvious form i.e. it requires additional memory. Then can be made it factorization that requires also additional computational complexity.

The offered algorithm of spectral estimation allows to make recursive updating of AR– parameters and spectral estimation and also allows no form a matrix R in an obvious form but elements of a matrix L, $L^{\sim}$ to receive directly from a matrix of input data X. We suppose that the input data come in real time. It is necessary after receipt of the next input element to make updating of a spectral density. We will present algorithm as consecutive performance of the following steps.

Step 1. Obtaining of current input data element x_n, $n=1,2,\ldots,N+p$

Step 2. Forming and waiting of $n\times(p + 1)$ input data matrix X

$$X=WX', \tag{13}$$

where $W = \left| w^{n-1}, w^{n-2}, \ldots, w^{2}, w^{1} \right|$

$$X' = \begin{vmatrix} x_1 & 0 & \ldots & 0 \\ x_2 & x_1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ x_{n-1} & x_{n-2} & \ldots & x_{n-p-1} \\ x_n & x_{n-1} & \ldots & x_{n-p} \end{vmatrix}, \tag{14}$$

where $n = 1,2,\ \ldots,\ N + p$ is number of the current element.

Step 3. Calculation of a matrix L

Step 4. Solving the equations system (12) and reception of vector A.

Step 5. Calculation of estimated spectral density according to eq.(8).

Let us consider recursive procedure of updating of elements of a matrix $L^{\sim}$ and obtaining of vector A by solving of system of the eq.(12). We assume, that the matrix X is generated on n complex elements of the input data. We present a matrix $L_n^{\sim}$ as

$$\left|\begin{matrix} L^{\sim} \\ \cdots \\ 0 \end{matrix}\right| = QX \text{ ,} \tag{15}$$

where Q is $(n\times n)$ orthogonal Householder matrix, $L^{\sim}$ is $(p+1)\times(p+1)$ upper triangular matrix. As follows from eq.(15) the elements of a matrix $L^{\sim}$ can be obtained using orthogonal transformation above a matrix X, not forming in an obvious form of a matrix R.

We present a matrix Q in factorized form

$$\left|\begin{matrix} L^{\sim} \\ \cdots \\ 0 \end{matrix}\right| = Q_{p+1}Q_p...Q_i X^{i-1}, \tag{16}$$

where $X^0 = X$, $X^{i-1} = Q_{i-1}...Q_1 X = \left| X_1^{i-1} \vdots \ldots \vdots X_j^{i-1} \vdots \ldots \vdots X_{p+1}^{i-1} \right|$, i=1,2...,p+1, thus column X_j^{i-1} of matrix X^{i-1}, j=1,2..., p+1, have passed the orthogonalization by matrixes Q_1, Q_2,..., Q_{i-1} (i–1) times. The matrixes Q_i are determined as follows [3]

$$Q_i = I_n - \frac{U_i U_i^{\sim}}{u_i C_i}, \; i = 1,2...,p+1, \tag{17}$$

where I_n is $(n\times n)$ identity matrix. In eq.(17) scalar C_i has a form

$$C_i = X_i^{i-1\sim} X_i^{i-1}, \; i = 1,2..., p+1, \tag{18}$$

where $X_i^{i-1} = \left| 0,0,\ldots,0, x_{i,i}^{i-1}, x_{i,i+1}^{i-1}, \ldots, x_{i,n}^{i-1} \right|$ is i-th column of a matrix X_{i-1}. The column vector U_i in eq.(17) has dimension n and is determined by expression

$$U_i = \left| X_i^{i-1} + \delta e_i C_i \right|, \tag{19}$$

Where $\delta = \begin{cases} 1, & if \quad Re\left\{x_{ii}^{i-1}\right\} \geq 0, \\ -1, & if \quad Re\left\{x_{ii}^{i-1}\right\} < 0, \end{cases}$, $Re\left\{x_{ii}^{i-1}\right\}$ is the real part of an element x_{ii}^{i-1}, $e_i = \left|0,\ldots,0,1,0,\ldots,0\right|$ is column vector of dimension n, where unit is in the i-th place. The value u_i

in eq.(17) is *i*-th element of a vector U_i. Using the eq.(16) *i*-th step of column ortogonalization of a matrix X^{i-1} is expressed as

$$X^{i} = Q_i X^{i-1}. \tag{20}$$

We present eq.(20) using eq.(17) as operations above vectors

$$X_j^{i} = Q_i X_j^{i-1} = \left(I_n - \frac{U_i U_i^{\sim}}{u_i C_i} \right) X_j^{i-1} = X_j^{i-1} - U_i \frac{U_i^{\sim} X_j^{i-1}}{u_i C_i}, \quad j = i, i+1, \dots, p+1. \tag{21}$$

Finally using eq.(19) we present eq.(21) as

$$\begin{vmatrix} x_{ji}^{i} \\ \cdots \\ X_j^{i} \end{vmatrix} = \begin{vmatrix} x_{ji}^{i-1} \\ \cdots \\ X_j^{i-1} \end{vmatrix} - \begin{vmatrix} x_{ii}^{i-1} + C_i \\ \cdots \\ X_i^{i-1} \end{vmatrix} \Theta_i \left(G_{ji} + C_i x_{ji}^{i-1} \right), \tag{22}$$

where $G_{ji} = X_j^{i-1\sim} X_i^{i-1}$

The scalar Θ_i in eq.(22) is defined by the following expression

$$\Theta_i = \frac{1}{u_i C_i}. \tag{23}$$

The eq.(22) is vector algorithm of obtaining of elements l_{ij} and orthogonalization of vectors X_j^{i} using Householder transform. From eq.(22) can be obtained expression for elements l_{ij} of a matrix L

$$l_{ij} = x_{ji}^{i} = x_{ji}^{i-1} - \left(x_{ii}^{i-1} + C_i \right) \Theta_i \left(G_{ji} + C_i x_{ji}^{i-1} \right), \tag{24}$$

where $i = 1,2..., p+1$, $j = i,i+1..., p+1$.
Expression for vectors orthogonalization yielding

$$X_j^{i} = X_j^{i-1} - X_i^{i-1} \Theta_i \left(G_{ji} + C_i x_{ji}^{i-1} \right). \tag{25}$$

Thus recursive computing process of reception of elements l_{ij} of a matrix L with applying of Householder transform on *i*-th iteration ($i = 1,2..., p+1$) consists of the following stages:

- Forming of a scalar C_i according to eq.(18);
- Forming of a vector U_i agrees eq.(19);
- The calculation l_{ij} agrees eq.(24);
- Ortogonalization of vectors X_j^{i} according to eq.(25).

How follows from eq.(18),(19),(24),(25) the performance of one step of vector algorithm requires approximately 2*n* multiplications and 2*n* adds.

We show below that receiving of the next input data element no require more than 6 operations of multiplications and 7 adds for obtaining update element of matrix *L*. We assume that through sampling interval of time *T* is received (*n*+1)–th element of input data. Then a matrix *X* can be presented as

$$X = \left| \begin{array}{cccc} x_1 & 0 & \cdots & 0 \\ x_2 & x_1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ x_{n-1} & x_{n-2} & \cdots & x_{n-p-1} \\ x_n & x_{n-1} & \cdots & x_{n-p} \\ \hline x_{n+1} & x_n & \cdots & x_{n-p+1} \\ \hline \end{array} \right|, \tag{26}$$

i.e. in a matrix the line is added (is allocated with a framework).

It is obvious if at *n*-th stage the value C_i, U_i, l_{ij}, X_j^i were received then these values can be easily modified at (*n*+1)-th stage using the following ratios

$$C_{i(n+1)} = C_{i(n)} + x_{n+2-i}^{i-1\sim} x_{n+2-i}^{i-1}, \quad i = 1,2,\dots,p+1, \tag{27}$$

$$G_{ji(n+1)} = G_{ji(n)} + x_{n+2-j}^{i-1\sim} x_{n+2-i}^{i-1}, \quad j = 1,2,\dots,p+1, \tag{28}$$

where x_{n+2-i}^{i-1} is (*n*+2-i)-th element of the vector X_i^{i-1}

$$U_{i(n+1)} = \left| \begin{array}{c} X_i^{i-1} \\ \cdots \\ x_{n+2-i}^{i-1} \end{array} \right| + \delta e_i C_{i(n+1)}; \tag{29}$$

$$l_{ij(n+1)} = x_{ji}^{i-1} - \left(x_{ii}^{i-1} + C_{i(n+1)} \right) \Theta_{i(n+1)} \left(G_{ji(n+1)} + C_{i(n+1)} x_{ji}^{i-1} \right); \tag{30}$$

$$X_{j(n+1)}^i = \left| \begin{array}{c} X_j^{i-1} \\ \cdots \\ x_{n+2-i}^{i-1} \end{array} \right| - \left| \begin{array}{c} X_i^{i-1} \\ \cdots \\ x_{n+2-i}^{i-1} \end{array} \right| \Theta_{i(n+1)} \left(G_{ji(n+1)} + C_{i(n+1)} x_{ji}^{i-1} \right); \tag{31}$$

where $\Theta_{i(n+1)} = \dfrac{1}{u_i C_{i(n+1)}}$.

How follows from eq.(27)-(31), for obtaining of elements $l_{ij(n+1)}$ from elements $l_{ij(n)}$ is required of the 6 multiplications and 7 adds. If take into account that matrix L has $p^2/2$ elements than updating of full matrix L requires $3p^2$ multiplications and $3.5p^2$ adds. After obtaining of elements l_{ij} of a matrix L, the elements of vectors B and A from eq.(12) can be also obtained using following equations

$$b_1 = \rho / l_{11} , \tag{32}$$

$$b_j = \left(-\sum_{n=1}^{j-1} l_{jn} b_n \right) / l_{jj} , \quad j = 2,3,\ldots, p+1, \tag{33}$$

$$a_{p+1} = b_{p+1} / l_{p+1,p+1} , \tag{34}$$

$$a_j = \left(b_j - \sum_{n=p+1}^{j} l_{jn} a_n \right) / l_{jj} , \quad j = p, p-1,\ldots,1 . \tag{35}$$

The coefficients from eq.(34), (35) are used for updating of spectral density in according with eq.(8). Solving of systems (32),(33) and (34),(35) require additional $(p^2 + p)$ operation of multiplications and adds. Thus the total complexity of algorithm $4.5p^2 + p$.

3.NUMERICAL STABILITY TEST

For comparison of known fast algorithm [1] and synthesized their modeling on the PC with 32–bit floating point arithmetic. Two cases were simulated. In first case the average square of filter output error (p = 10) was estimated when its input was one “powerful” harmonic and additive “white” noise. The ratio P_S / P_N was 60*db*, where P_S – harmonic power, P_N – noise power. This case should be considered as “worse” because the maximal eigenvalue of a matrix R in this case will be approximately equal P_S, and minimal – P_N. Hence the condition number of a matrix R is equal P_S / P_N. The result of dependence ρ / ρ_{opt} from number of input data n is shown on Figure1 (curves1) where ρ_{opt} is received from the direct solution of the eq.(6) using the Levinson algorithm. The number of tests equal 100. In the second case the sequence consist 8 harmonic of total power $\sum_{i=1}^{8} P_S / P_N = 60db$. Such case should be considered as “average” from the point of view of numerical stability because at the completely resolved spectral components the condition number of a matrix R will be proportional to P_S / P_N, that is lower than in the worse case. The results are shown on Figure1 (curves 2). As follows from the given results of simulation, at the finite word length the increase of an output error of the filter with increase of number of input data (shaped curve) is appeared. Such effect will result in distortion of the spectrum form when apply recursive algorithms for spectral estimation. That is undesirable and practically compensates the advantages of recursive algorithms.

The results of the synthesized algorithm simulation (the solid curve in figure1) shows that this effect can partially be excluded by applying offered recursive procedure for obtaining of autoregressive parameters. In particular, for the given examples the synthesized algorithm allows to keep long-term numerical stability down to n = 10000 input elements that lays within the limits of reasonable maximal length of time series of a various nature.

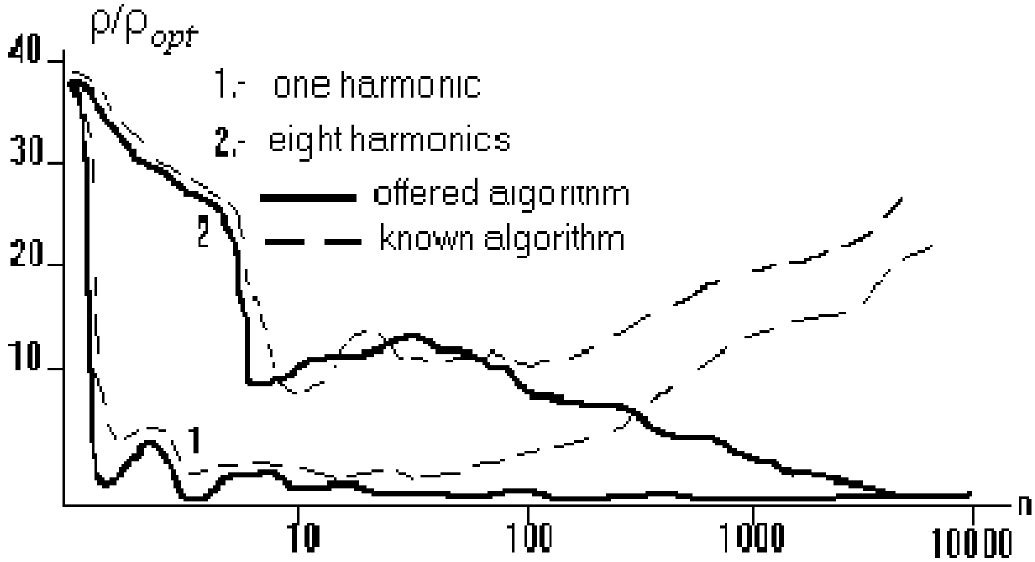

Fig. 1. *Dependence of* ρ / ρ_{opt} *from number of input data* n .

4. CONCLUSIONS

The effective RLS fast algorithm of spectral estimation with longterm numerical stability is synthesized. It is achieved by using in the synthesized algorithm the Householder transform procedure. Is offered recursive version of the specified procedure of RLS spectral estimation algorithm. It is shown, that offered recursive algorithm has allowed keeping numerical stability up to 10000 elements of input data even if correlation matrix of input data has bad condition. In the same time computational complexity of the offered algorithm is no more then known one. Thus, the application of the synthesized algorithm in spectral estimation problem allow to avoid effect of spectral density distortion arising in real computing systems due to final word length.

References

1. Marple S.L. Jr. ”Digital spectral analysis with application”(Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1987)
2. Haykin S. “Adaptive Filter Theory”(Prentice Hall, 1996, 989p)
3. Wilkinson J..H. “The algebraic eigenvalue problem: monographs in Numerical Analysis”(Clarendon Press, Oxford, 1988)
4. Golub G.H., Van Loan C.F. Matrix Computations. – Johns Hopkins University Press, 1993.
5. Householder A.S. “A class of method for inverting matrices”. *J. Sos. Indust. Appl. Math,* 1958, Vol.6, # 2, pp. 189–195.