MULTIDIMENSIONAL COMPLEX NUMBER PARAMETRIC MODEL ORDER AND PARAMETERS ESTIMATION

D. Kouamé, C. Garnier, J.M. Grégoire, J.M. Girault

LUSSI, FRE 2448 CNRS , University of Tours, France. E-mail: kouame@univ-tours.fr

ABSTRACT

This paper presents a technique for accessing multidimensional complex number AR model order and parameters through matrix factorization. The principle of this technique consists of the transformation of the multidimensional model to a pseudo SISO (Simple Input Simple Output) AR model then performing factorization of the covariance matrix of the data. This factorization then leads to a recursive form of the parameter and order estimation. The methodology developed here may be applied to an AR model of any dimension. Computer simulation results are provided to illustrate the behavior of this method.

1. INTRODUCTION

A multidimensional or N-dimensional complex-number AR model (or ND-AR) represents $y(n_1, n_2, ..., n_N)$, the components of the complex number signal y at location $(n_1, n_2, ..., n_N)$, as a linear combination of the complex number components $y(n_1 - k_1, n_2 - k_2, ..., n_N - k_N)$ and an additive noise $w(n_1, n_2, ..., n_N)$, where $(k_1, k_2, ..., k_N)\epsilon I$, and I is a set of neighbors excluding $(0, 0, ..., 0)$. In recent years in the field of multidimensional parametric modeling, two-dimensional ($N = 2$) autoregressive (2D-AR) modeling has received very high levels of attention in many areas, particularly in digital signal and image processing areas. Many works have accordingly merged, *e.g.* [1]-[2] and relevant references therein. Although three dimensional ($N = 3$) AR (3D-AR) has received some attention, studies concerning more than three dimensions are rare, particularly in contrast to their potential applications [3],[4],[5]. This is mainly due to the high complexity of the generalization of 2D-AR cases. The aim of this paper is to propose a general framework for estimation of general ND-AR model parameters and order . Based on matrix factorization, the recursive form of this algorithm for both dimensions (time, space,...) and order is straightforward. It allows easy AR modeling of, for example, three, four (or more) dimension signals such as analysis of video sequences . The properties of the algorithm are introduced and its behavior is illustrated with $N = 3$ using a numerical example.

Thanks to French Department of Research for funding.

2. FORMULATION OF MULTIDIMENSIONAL COMPLEX AR PROBLEM

Consider the second order stationary multidimensional complex AR process (ND-AR) defined by :

$$y(n_1, n_2, ..., n_N) = \sum ... \sum_{(k_1,k_2,...,k_N)\epsilon I} a(k_1, k_2, ..., k_N)$$
$$\times y(n_1 - k_1, n_2 - k_2, ..., n_N - k_N) + w(n_1, n_2, ..., n_N), \quad (1)$$

where $u(n_1, n_2, ..., n_N)$ is a field of zero mean constant variance-independent random noise and the parameters $a(k_1, k_2, ..., k_N)$ are complex numbers and provide a stable system. In the following, attention is focused on the first hyperplane model, without loss of generality (the methodology may be applied to the other hyperplanes) *i.e.*, the set of neighbors is $I = \{(k_1, k_2, ..., k_N)|k_i = 1, 2, ...p_i, i = 1, 2, ...N\}$. For convenience we assume that $p_1 = p_2 = ...p_N = m$, *i.e.* the model orders are identical in all directions. The following methodology is a generalization of UDU^H factorization, introduced in [6] and used in [7] and [8], respectively, for one dimensional real and complex number cases. This factorization made it possible to access the parameter and the order from 0 to m through an algorithm having good numerical properties and concise computation.

Let us define the following vectors in which elements y and a are stacked.

$$\begin{aligned}\boldsymbol{\phi}_m^T(n_1, n_2, ..., n_N) = \\ [y(n_1, n_2, ..., n_N - 1)...y(n_1, n_2, ..., n_N - m)... \\ y(n_1, n_2 - 1, ..., n_N)...y(n_1, n_2 - m, ..., n_N)... \\ y(n_1 - m, n_2 - m, ..., n_N - m)\, y(n_1, n_2, ..., n_N)]\end{aligned}$$

$$\begin{aligned}\boldsymbol{\theta}_m^T(n_1, n_2, ..., n_N) = \\ [a(0, 0, ...1,)...a(0, 0, ...m)\, a(0, 1, ...0,)... \\ a(0, m, ...0,)...a(m, 0, ..., 0)...a(m, m, m...m)\; 1]\end{aligned} \quad (2)$$

Note that $y(n_1, n_2, ..., n_N)$ and 1 are part of these vectors.

Let also

$$\boldsymbol{x}_m^T(n_1,n_2,...,n_N) = [y(n_1,n_2,...,n_N-1)...y(n_1,n_2,...,n_N-m)... y(n_1,n_2-1,...,n_N)...y(n_1,n_2-m,...,n_N)... y(n_1-m,n_2-m,...,n_N-m)] \quad (3)$$

This means

$$\boldsymbol{\phi}_m^T(n_1,n_2,...,n_N) = [x_m^T(n_1,n_2,...,n_N) \quad y(n_1,n_2,...,n_N)] \quad (4)$$

Defining data covariance matrix :

$$P_m(n_1,n_2,...,n_N) = \left[\sum_{i_1=1}^{n_1} ... \sum_{i_N=1}^{n_N} \phi_m(n_1,n_2,...,n_N)\phi_m^H(n_1,n_2,...,n_N)\right]^{-1} \quad (5)$$

and assuming that $p = (m+1)^N$, the size of this matrix is $p \times p$. For convenience the following notations will be used when no confusion is possible.

$$\boldsymbol{n} = (n_1,n_2,...,n_N) \quad (6)$$
$$\boldsymbol{k} = (k_1,k_2,...,k_N)$$

and generally any index variable $(t_1,t_2,...,t_N)$ will be denoted :$\boldsymbol{t} = (t_1,t_2,...,t_N)$. Also $\sum_{i_1=1}^{n_1} ... \sum_{i_N=1}^{n_N}$ will be denoted $\sum_{\boldsymbol{i}=1}^{\boldsymbol{n}}$ and $\sum_{i_1=1}^{n_1-i} ... \sum_{i_N=1}^{n_N-i}$ will be denoted $\sum_{\boldsymbol{i}=1}^{\boldsymbol{n}-i}$ We then write P_m in factored form as follows:

$$P_m(\boldsymbol{n}) = U_m(\boldsymbol{n})D_m(\boldsymbol{n})U_m^H(\boldsymbol{n}); \quad (7)$$

where H denotes the Hermitian matrix transpose and U is an upper triangular matrix with all diagonal elements equal to unity. The elements of this upper triangular matrix are column vectors with dimension 1 to p defined as follows :

$$U_m(\boldsymbol{n}) = [\quad 1 \quad col\{\vartheta_{0,p}(\boldsymbol{n}) \quad 1\}... \quad col\{\vartheta_{p-i,i}(\boldsymbol{n}) \quad 1\}... \quad col\{\vartheta_{p-1,1}(\boldsymbol{n}) \quad 1\}... \quad col\{\vartheta_{p,0}(\boldsymbol{n}) \quad 1\} \quad] \quad (8)$$

$D_m(\boldsymbol{n})$ in eq.(7) is a diagonal matrix containing loss functions for order 1 to m, as seen below.

Remark:

* $\vartheta_{p-i,i}(\boldsymbol{n})$ is a column vector. Its dimension is $p-i$.
* $col\{\vartheta_{p-i,i}(\boldsymbol{n}) \quad 1\} = \begin{bmatrix} \vartheta_{p-i,i}(\boldsymbol{n}) \\ 1 \end{bmatrix}$ is the $(p-i+1)^{th}$ column.
* Due to the structure of the model defined in eq.(2), $\vartheta_{p-i,i}(\boldsymbol{n})$ consists of part or all of the *true* parameters of the model, depending on whenever the model order is less or greater than the dimension of $\vartheta_{p-i,i}(\boldsymbol{n})$

Eq.(7) is achieved from successive decompositions. Indeed, using eq.(4) in eq.(5), it comes :

$$P_m^{-1}(\boldsymbol{n}) = \begin{bmatrix} \sum_{\boldsymbol{j}=1}^{\boldsymbol{n}} x_m(\boldsymbol{j})x_m^H(\boldsymbol{j}) & \sum_{\boldsymbol{j}=1}^{\boldsymbol{n}} x_m(\boldsymbol{j})y(\boldsymbol{j}) \\ \sum_{\boldsymbol{j}=1}^{\boldsymbol{n}} y(\boldsymbol{j})x_m^H(\boldsymbol{j}) & \sum_{\boldsymbol{j}=1}^{\boldsymbol{n}} |y(\boldsymbol{j})|^2 \end{bmatrix} \quad (9)$$

From classical Least squares estimation theory(see e.g.[9]), it is known that :

$$\vartheta_{p,0}(\boldsymbol{n}) = \left[\sum_{\boldsymbol{j}=1}^{\boldsymbol{n}} x_m(\boldsymbol{j})x_m^H(\boldsymbol{j})\right]^{-1} \sum_{\boldsymbol{j}=1}^{\boldsymbol{n}} x_m(\boldsymbol{j})y(\boldsymbol{j}) \quad (10)$$

or similarly

$$\vartheta_{p-i,i}(\boldsymbol{n}) = \left[\sum_{\boldsymbol{j}=1}^{\boldsymbol{n}-i} \phi_m(\boldsymbol{j})\phi_m^H(\boldsymbol{j})\right]^{-1} \sum_{\boldsymbol{j}=1}^{\boldsymbol{n}-i} \phi_m(\boldsymbol{j})y(\boldsymbol{j}) \quad (11)$$

Thus eq.(9) may also be written

$$P_m^{-1}(\boldsymbol{n}) = \begin{bmatrix} I_{p-1} & 0 \\ \vartheta_{p,0}^H(\boldsymbol{n}) & 1 \end{bmatrix} \begin{bmatrix} Q_m(\boldsymbol{n}) & 0 \\ 0 & C_{p,0}(\boldsymbol{n}) \end{bmatrix} \begin{bmatrix} I_{p-1} & \vartheta_{p,0}^H(\boldsymbol{n}) \\ 0 & 1 \end{bmatrix} \quad (12)$$

where

$$Q_m(\boldsymbol{n}) = \sum_{\boldsymbol{i}=1}^{\boldsymbol{n}} x_m(\boldsymbol{i})x_m^H(\boldsymbol{i}) \quad (13)$$

and

$$C_{p,0}(\boldsymbol{n}) = \sum_{\boldsymbol{j}=1}^{\boldsymbol{n}} |y|^2(\boldsymbol{j}) - \vartheta_{p,0}^H(\boldsymbol{k}) \sum_{\boldsymbol{j}=1}^{\boldsymbol{n}} x_m(\boldsymbol{j})x_m^H(\boldsymbol{j})\vartheta_{p,0}(\boldsymbol{n}) \quad (14)$$

$C_{p,0}$ is equivalent to a loss function. Indeed eq.(10) is also equivalent to :

$$\left[\sum_{\boldsymbol{j}=1}^{\boldsymbol{k}} x_m(\boldsymbol{j})x_m^H(\boldsymbol{j})\right]\vartheta_{p,0}(\boldsymbol{n}) = \sum_{\boldsymbol{j}=1}^{\boldsymbol{k}} x_m(\boldsymbol{j})y(\boldsymbol{j}) \quad (15)$$

Defining

$$\hat{y}(\boldsymbol{n}) = \sum_{\boldsymbol{j}=1}^{\boldsymbol{n}} x_m^H(\boldsymbol{j})\vartheta_{p,0}(\boldsymbol{n}) \quad (16)$$

yields

$$C_{p,0}(\boldsymbol{n}) = \sum_{\boldsymbol{j}=1}^{\boldsymbol{n}} |y(\boldsymbol{j}) - \hat{y}(\boldsymbol{j})|^2 \quad (17)$$

and generally,

$$C_{p-i,i}(\boldsymbol{n}) = \sum_{\boldsymbol{j}=1}^{\boldsymbol{n}-\boldsymbol{i}} |y(\boldsymbol{j}) - \hat{y}(\boldsymbol{j})|^2 \quad (18)$$

Thus $C_{p-i,i}(\boldsymbol{n})$ are positive. Now, returning to eq.(13): $Q_m(\boldsymbol{n})$, which is in the form of eq.(5), is then iteratively decomposed

in the same way, and ultimately, after taking the inverse of the matrices involved in the decomposition, eq.(7) is obtained. As mentioned above, $U_m(\boldsymbol{n})$ is an upper triangular complex number matrix, the elements of which are $\vartheta_{p-i,i}(\boldsymbol{n})$.
$D_m(\boldsymbol{m})$ is a diagonal matrix, the elements of which are the inverse of the loss function, *i.e.* $C^{-1}_{p-i,i}(\boldsymbol{n})$ Now, it is important to note that the similarity with a 1 D case is just apparent. Indeed, by its construction the model defined in eq.(2) has the following characteristics:

- The *true* parameters as defined in eq.(2) are not directly the columns of U_m, but are included in these columns. This is due to the definitions of the model in eq.(2). Indeed assuming the order of the model is $m = m_0$, then the first m_0-elements of each column of $U_{m;\, m>m_0}$ are actually the first estimated m_0-parameters of the model. The elements from m_0 to m have no physical meaning. This is repeated periodically to obtain all the parameters by segments of m_0 elements. Since the dimension of U_m is $(m+1)^N \times (m+1)^N$ *true* parameters are located in column $(m_0+1)(m+1)^{N-1} - 1$ since the $a(0,0,...,0) = 1$ is not being estimated. Note that this is a generalization which is also valid in the 1-D case.

- Accordingly the elements of D_m are pseudo-periodic, with a period $m+1$ in each direction (dimension). The minimum within each period constitutes the actual loss function, and the first minimum of this cost function gives the *true* order m_0 of the model. Note that all elements of the matrices are stacked and thus D_m has $(m+1)^N \times (m+1)^N$ elements. Thus, to find the *true* order of the model the following steps may be used :

  1. split elements of D_m into successive segments of $m+1$ elements, and create a new vector, M_1 consisting of the minima of the segments.
  2. repeat 1) using the above set of minima.
  3. stop when the size of the vector of minima is $m+1$. This vector is called M_N. This procedure needs N steps.
  4. Finally the minimum of M_N gives the *true* order m_0

It is clear that the order-recursion is achieved from the above. The dimension (time, space,...)-recursion may be obtained by generalizing the complex 1-D case, using UDU^H decomposition. This last recursion is obtained by defining $\boldsymbol{n} - 1 = (n_1 - 1, n_2 - 1, ..., n_N - 1)$ and writing the relevant recursions. Due to space limitation, we only provide below the main steps of the method. From eq.(5), it can be written :

$$P_m(\boldsymbol{n}) = [P_m^{-1}(\boldsymbol{n}-1) + \phi_m(\boldsymbol{n})\phi_m^H(\boldsymbol{n})]^{-1} \qquad (19)$$

Defining the variables $f = U_m^T(\boldsymbol{n}-1)\phi_m(\boldsymbol{n})$, $g = D_m(\boldsymbol{n}-1)f^*$, and $\beta(\boldsymbol{n}) = 1 + f^T g$, where the asterisk denotes the complex conjugate, at the rank one update expression, the matrix $P_m(\boldsymbol{n})$ can be expressed by :

$$P_m(\boldsymbol{n}) = U_m(\boldsymbol{n})D_m(\boldsymbol{n})U_m^H(\boldsymbol{n}) =$$
$$U_m(\boldsymbol{n}-1)\left[D_m(\boldsymbol{n}-1) - \frac{gg^H}{\beta(\boldsymbol{n})}\right]U_m^H(\boldsymbol{n}-1) \qquad (20)$$

From these recursions, only elements of U_m with physical meanings are retained. To speed up computations, the recursions may be applied only to the elements with physical meanings.

3. NUMERICAL SIMULATION

For simplicity in the presentation of the parameters of the model , we set $N = 3$, $m = 4$, $m_0 = 2$. We consider the following 3D complex AR model as defined in eq.(2), where $a(k_1,k_2,k_3), k_1 = 0,1,2;\ k_2 = 0,1,2;\ k_3 = 0,1,2;$ defined as below. $a(0,0,0) = 1$ is not being estimated.

$$y(n_1,n_2,...,n_3) = \sum_{k_1=0}^{2}\sum_{k_2=0}^{2}\sum_{k_3=0;(k_1,k_2,k_3)\neq(0,0,0)}^{2} a(k_1,k_2,...,k_N)$$
$$\times y(n_1-k_1, n_2-k_2, ..., n_N-k_N) + w(n_1,n_2,...,n_N),$$

where y is a $16\times16\times16$ complex field driven by a complex gaussian random field with variance 0.1 : The results

Table 1.

$a(:,:,0) =$

1	-0.8000 - 1.1000i	-0.1500 + 0.4500i
-0.8000 - 1.1000i	-0.5700 + 1.7600i	0.6150 - 0.1950i
-0.1500 + 0.4500i	0.6150 - 0.1950i	-0.1800 - 0.1350i

$a(:,:,1) =$

-0.8000 - 1.1000i	-0.5700 + 1.7600i	0.6150 - 0.1950i
-0.5700 + 1.7600i	2.3920 - 0.7810i	-0.7065 - 0.5205i
0.6150 - 0.1950i	-0.7065 - 0.5205i	-0.0045 + 0.3060i

$a(:,:,2) =$

-0.1500 + 0.4500i	0.6150 - 0.1950i	-0.1800 - 0.1350i
0.6150 - 0.1950i	-0.7065 - 0.5205i	-0.0045 + 0.3060i
-0.1800 - 0.1350i	-0.0045 + 0.3060i	0.0877 - 0.0607i

Theoretical parameters

are shown in fig.1 and Table 2. D_m is shown in fig.1 with $m = 4$ here (upper curve) and the result after minima extraction (lower curve), it can be seen that the order found is 2. Table 2 shows the estimated parameters extracted from $U_m;\ m = 4$ at column $(m0+1)(m+1)^2 - 1 = 74$.

Table 2.

$$a(:,:,0) =$$

1	-0.8003 - 1.0995i	-0.1492 + 0.4500i
-0.8007 - 1.1003i	-0.5685 + 1.7608i	0.6149 - 0.1970i
-0.1502 + 0.4501i	0.6149 - 0.1960i	-0.1816 - 0.1345i

$$a(:,:,1) =$$

-0.8002 - 1.0988i	-0.5700 + 1.7594i	0.6155 - 0.1937i
-0.5700 + 1.7605i	2.3919 - 0.7826i	-0.7079 - 0.5199i
0.6155 - 0.1942i	-0.7074 - 0.5199i	-0.0030 + 0.3071i

$$a(:,:,2) =$$

-0.1491 + 0.4497i	0.6148 - 0.1948i	-0.1795 - 0.1360i
0.6154 - 0.1940i	-0.7072 - 0.5211i	-0.0045 + 0.3069i
-0.1794 - 0.1361i	-0.0046 + 0.3067i	0.0879 - 0.0614i

Estimated parameters

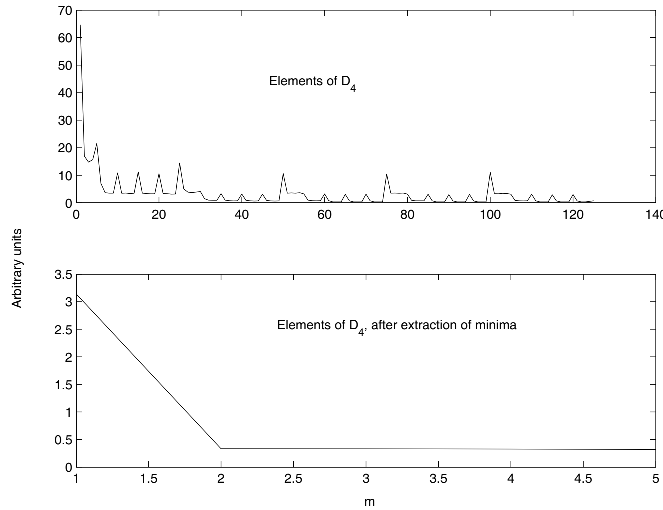

Fig. 1.
Loss fonction (elements of Diagonal matrix D_m $m = 4$). Upper curve shows all the elements of D_m. Lower curve shows loss function after extraction of minima . As can be seen, in the lower curve the minimum is achieved at $m_0 = 2$

4. CONCLUSION AND GENERAL REMARKS

We have presented a new estimation of multidimensional complex number AR model order and parameters. This algorithm was based on UDU^H matrix factorization which generalized the $1D$ cases. Any other matrix factorization may be used instead. The proposed approach is an order recursive algorithm which allows simultaneously to access the parameters of the model with any order from 0 to a given order m. The dimension (time, space,...) recursive form of the algorithm is straightforward, as in the 1-D case. This thus makes it possible to perform AR modeling of video sequences for example. However, for very long data sets, this algorithm may be time consuming and bias may appear due to the ill-conditioning. In this case any fast techniques and regularization techniques may be used, (see e.g.[10]). The algorithm is finally illustrated using simulation data.

5. REFERENCES

[1] B.F.McGruffin and B. Liu, "An efficient algorithm for two dimensional autoregressive spectrum estimation," *IEEE Trans. ASSP*, vol. 37, pp. 106–117, 1989.

[2] A.H. Kayran and S.R. Parker, "Optimum quater-plane autoregressive modeling of 2-d fields using four-field lattice approach," *IEEE Trans. Sig. Process.*, vol. 45, pp. 2363–2373, 1997.

[3] H. K. Kwan and Y. C. Lui, "Lattice predictive modeling of 3-d random fields with application to interframe predictive coding of picture sequences," *Int. J. Electron.,*, vol. 66, pp. 489505, 1989.

[4] B. Choi, "An order-recursive algorithm to solve the 3-d yule walker equations of causal 3-d ar models,," *IEEE Trans. Sig. Process.*, vol. 47.

[5] C.L. Nikias and R. R. Raghuveer, "Multi-dimensional parametric spectral estimation," *Signal Processing*, vol. 9, pp. 191–205, 1985.

[6] G.J. Bierman, *Factorization Methods for Discrete Sequentiel Estimation*, New York Academic, 1977.

[7] S. Niu, D. Xiao, and D.G. Fisher, "A recursive algorithm for simultaneous identification of model order and parameters," *IEEE Trans. ASSP.*, vol. 38, pp. 884–886.

[8] J.P. Remenieras, D. Kouame, J.M.Gregoire, and F. Patat, "Multiphase pipe flow velocity profile measurements by doppler ultrasound containing a high level of colored noise," in *Proc. of IEEE Int. Ultr. Symposium*, 1996.

[9] L Ljung, *System Identification, Theory for users*, Printice Hall, 1987.

[10] L Ljung, *Fast algorithm for integral estimation and least squares identification problems*, Linkoping University, Sweden, Dissertation 93, 1985.