A Recursive Instrumental Variable (RIV) Lattice Algorithm for Adaptive Identification of Non Gaussian Multichannel AR Processes

V. Buzenac, R. Settineri, M. Najim

Equipe Signal et Image de l'ENSERB and GDR 134
351, cours de la Libération, 33405 Talence cedex, FRANCE
Tel: (+33) 56.84.61.40, Fax: (+33) 56.84.84.06, Email: vero@goelette.tsi.u-bordeaux.fr

RESUME

Dans cet article, le filtrage adaptatif de modèles AR multicanaux est considéré. Afin d'obtenir des estimées des paramètres AR insensibles aux bruits gaussiens, on propose un algorithme en treillis permettant de résoudre les équations « normales aux ordres supérieurs ». Notre algorithme peut être considéré comme un calcul récursif de la pseudoinverse d'une matrice bloc-Toeplitz dont les blocs ne sont pas carrés. La méthode proposée tient compte des problèmes d'identifiabilité dus à l'utilisation de cumulants, et fournit des estimées consistentes quel que soit le modèle considéré.

ABSTRACT

This paper deals with the adaptive filtering of non gaussian multichannel AR models. In order to obtain insensitive to gaussian noise estimates of the AR parameters, we propose a lattice algorithm allowing the resolution of the « higher order normal equations ». It can be considered as a recursive computation of the pseudoinverse of block-Toeplitz matrix with non square blocks. The proposed method takes identifiability problems due to cumulants into consideration and provides consistent AR estimates for any model.

Multivariate system identification is used in many applications, such as multichannel equalization, source separation, econometry ... (for a bibliography, see [1] for instance), and has been widely studied under the assumption of gaussian inputs. But the Higher Order Statistics (HOS) properties may be of use if the systems are not minimumphase, or if the outputs are disturbed by gaussian noise. Multichannel ARMA identification with HOS has been dealt with by some authors [2, 3], but as far as we know, no adaptive approach has been proposed, in spite of the often encountered nonstationarity of the signals under study. The aim of this paper is to provide such an approach. The algorithm we propose is an extension of Swami & Mendel's univariate double lattice [7], and uses an instrumental variable to compute recursively the pseudo-inverse of a block-Toeplitz matrix with rectangular blocks.

The paper is organized as follows. In the first part, the reader is introduced to multichannel ARMA processes, and to cumulants of vectorial processes. In part II, we derive the proposed adaptive lattice algorithm. In part III, we propose a modification of the used equations in order to guarantee identifiability. Part IV provides results of simulations.

I - Multichannel AR estimation using cumulants

In the whole paper, we consider the multivariate causal ARMA model :

$$\sum_{i=0}^{p}\mathbf{A}(i)\,x(n-i)=\sum_{i=0}^{q}\mathbf{B}(i)\,w(n-i) \qquad (1)$$

where $\mathbf{A}(i)$ and $\mathbf{B}(i)$ are respectively the $r\times r$ AR and MA matrix coefficients, w(n) is the input vector process and x(n) is the output vector process. w(n) is supposed to be non gaussian, and x(n) may be corrupted by a gaussian vectorial noise v(n). We call y(n) = x(n) + v(n) the observed vector process.

Several definitions have been given for cumulants of vector processes. The most convenient in our case has been given by Swami et al. [2]. If y(n) is a r-element vector, the autocorrelation and 3[rd] order cumulant are respectively given by:

$$C_{2x}(\tau)=E\{y(n)\otimes y(n+\tau)\}$$

$$C_{3x}(\tau_1,\tau_2)=E\{y(n)\otimes y(n+\tau_1)\otimes y(n+\tau_2)\}$$

where $\otimes$ means the Kronecker product.If the reader is not familiar to the Kronecker product and the unvec operator, we refer him to [4] for definitions and properties. The higher order cumulants can be defined in the same way, but they involve permutation matrix. Note that a k-th order cumulant of a r-element vector is a r^k-element vector.

Let $\mathbf{H}(k)$ be the impulse response coefficient matrix, and $\mathbb{H}(z)$ its z-transform. We will suppose our ARMA model has the following properties :

1 - $\mathbf{A}(0)=\mathrm{I}$, $\mathbf{H}(0)=\mathrm{I}$, $\mathbf{B}(0)$, $\mathbf{B}(q)$ and $\mathbf{A}(p)$ are full rank.

2 - $\mathbb{H}(z)$ has no pole-zero cancellation,

3 - rank $(\overline{\Gamma}_{kw})=r$, with $\overline{\Gamma}_{kw}=\mathrm{unvec}_{r\times r^{k-1}}(\underline{\Gamma}_{kw})$, where $\underline{\Gamma}_{kw}$ is the k-th order cumulant of the input process.

Under these properties, the multichannel higher order equations can be written :

$$\sum_{i=0}^{p}\mathbf{A}(i)\,\overline{c}_{ky}(t,\tau-i)=0, \qquad \tau>q \qquad (2.a)$$

$$\sum_{i=0}^{p}\mathbf{A}(i)\,\overline{c}_{ky}(t,q-i)=(\mathbf{I}_{k-2}\otimes\mathbf{H}(t))\overline{\Gamma}_{kw} \qquad (2.b)$$

with $\quad \overline{c}_{ky}(m,n)=\mathrm{unvec}_{r\times r^{k-1}}(\underline{c}_{ky}(m,n))$

II. The adaptive lattice algorithm.

In the derivation, we consider a purely AR model for the sake of simplicity, and we use third order cumulants. The extensions to the estimation of AR part of an ARMA model and to higher order cumulants are straightforward and will be given afterwards.

In equations (2), let t=0 and collect them for τ=0 to p. The following system is obtained:

$$\mathbf{R}\,\mathbf{A} = e_0 \otimes \overline{\Gamma}_{kw}^{T} \qquad (3)$$

with :

$$e_0 = [1 \; 0 \; \ldots \; 0]^T$$

$$\mathbf{A} = [\,\mathbf{A}(0)\,\mathbf{A}(1)\,\ldots\,\mathbf{A}(p)]^T$$

$$\mathbf{R} = \begin{bmatrix} \overline{c}_{3y}^{T}(0,0) & \overline{c}_{3y}^{T}(0,-1) & \cdots & \overline{c}_{3y}^{T}(0,-p) \\ \overline{c}_{3y}^{T}(0,1) & \ddots & & \overline{c}_{3y}^{T}(0,-p+1) \\ \vdots & & \ddots & \vdots \\ \overline{c}_{3y}^{T}(0,p) & \cdots & \cdots & \overline{c}_{3y}^{T}(0,0) \end{bmatrix}$$

R is a nonsymmetric block Toeplitz matrix with rectangular blocks.

Suppose **R** is full rank (this will be discussed in part III). The **A** matrix can be obtained using the Moore-Penrose pseudo-inverse of **R**. To derive an adaptive algorithm, we seek to inverse recursively the **R** matrix. Symmetric block Toeplitz matrix recursive inversion has been dealt with by Akaike [5], and the case of non symmetric ones has been considered by Swami [6], but both of them have supposed the blocks to be square. Our approach cover the case of rectangular blocks, and can thus be considered as a recursive computation of the pseudo-inverse of block Toeplitz matrix with rectangular blocks.

Let us define the inner product as :

$$< \underline{u}(n), \underline{v}(n) > \; = \sum_{i=1}^{n} \lambda^{n-i}\, \underline{u}(i) \otimes \underline{v}^T(i)$$

$<\underline{u}(n),\underline{v}(n)>$ is the weighted estimation of $E[\underline{u}(n) \otimes \underline{v}^T(n)]$. Note that:

$$< \underline{u}(n), \underline{v}(n) > \; = \sum_{i=1}^{n} \lambda^{n-i}\, \underline{u}(i)\underline{v}^T(i),$$

but we have chosen the Kronecker product notation in order to make the link with the definition of vector processes cumulants.

Let $\underline{z}(n)$ be an instrumental process associated to $\underline{y}(n)$. $\underline{z}(n)$ is a s-element vector, and we allow s to be different from r.

Let $\qquad \Phi_{p+1}(n) = < \underline{Z}_{p+1}(n), \underline{Y}_{p+1}(n) >$

$$\text{with: } \underline{Z}_{p+1}(n) = \begin{bmatrix} \underline{z}(n) \\ \vdots \\ \underline{z}(n-p) \end{bmatrix}, \qquad \underline{Y}_{p+1}(n) = \begin{bmatrix} \underline{y}(n) \\ \vdots \\ \underline{y}(n-p) \end{bmatrix}$$

In the stationnary case, $\Phi_{p+1}(n)$ is a non symmetric block Toeplitz matrix, and the blocks are rectangular if s≠r. If z(n) is chosen equal to y(n) ⊗ y(n), $\Phi_{p+1}(n)$ is an estimate of R.

Consider the forward linear prediction of y(n) so that the prediction error $\underline{f}_p(n)$ is orthogonal to $\underline{z}(n-i) = \underline{y}(n-i) \otimes \underline{y}(n-i)$ for i=1,2, ...,p. It can be shown that this prediction problem is equivalent to the resolution of the system (3). Note that the orthogonality introduced here is not the conventionnal orthogonality condition.

This prediction problem is solved using a double lattice structure involving the following four linear predictions:

1. the forward linear prediction of $\underline{y}(n)$ defined above.
2. the backward linear prediction of $\underline{y}(n-p)$ so that the prediction error $\underline{b}_p(n)$ is orthogonal to $\underline{z}(n-i)$ for i=p-1, ..., 1, 0.
3. the forward linear prediction of $\underline{z}(n)$ so that the prediction error $\underline{\tilde{f}}_p(n)$ is orthogonal to $\underline{y}(n-i)$ for i=1,2, ...,p.
4. the backward linear prediction of $\underline{z}(n-p)$ so that the prediction error $\underline{\tilde{b}}_p(n)$ is orthogonal to $\underline{y}(n-i)$ for i=p-1, ..., 1, 0.

The structure is made of two lattices, excited respectively by the original process $\underline{y}(n)$ and the associated instrumental process $\underline{z}(n)$, and coupled by their reflexion coefficients.

The algorithm is given in table 1. Its complete derivation is too long to be given here but a longer paper is currently into preparation. It will be sent to any interested reader on request to the authors.

Remarks:

1. Note that this algorithm involves two More Penrose pseudoinverses, while the instrumental variable double lattice derived by Swami [7] involves two matrix inversions.
2. In spite of significant differences in their derivation, mainly due to the fact that pseudoinverses do not obey to the same properties as inverses, the proposed algorithm and the double lattice of [7] have similar equations but with different matrix dimensions.
3. The proposed algorithm can be used to compute recursively any block Toeplitz matrix M with rectangular block, provided M can be written as the intercorrelation matrix of a r-channel process y and a s-channel instrumental process z.
4. The extension to ARMA(p,q) models is obvious. In such a case, the matrix R of equation (3) is

$$\mathbf{R} = \begin{bmatrix} \overline{c}_{3y}^{T}(0,q) & \overline{c}_{3y}^{T}(0,q-1) & \cdots & \overline{c}_{3y}^{T}(0,q-p) \\ \overline{c}_{3y}^{T}(0,q+1) & \ddots & & \overline{c}_{3y}^{T}(0,q-p+1) \\ \vdots & & \ddots & \vdots \\ \overline{c}_{3y}^{T}(0,q+p) & \cdots & \cdots & \overline{c}_{3y}^{T}(0,q) \end{bmatrix}$$

which can be built by choosing:

z(n) = y(n-q) ⊗ y(n-q)

III Identifiability.

In section II, we have supposed the matrix R to be full rank. This may not be true in some cases [2]. However, Swami et al. have shown that the AR parameters of of a multichannel system could be uniquely determined from all the cumulants taken into

$$M_{p,q} = \{\; C(k,t) \;/\; k = q-p, \ldots, q \quad t = q+1, \ldots, q+p \;\}$$

In the previous section, we have only taken the cumulants of $M_{p,q}$ with k=q (=0 in the AR case). In order to assure identifiability, we propose to use a linear combination of the cumulants of $M_{p,q}$. This can be seen as an extension of the w-slice method of Fonollosa et al. [8] to the multichannel case.

We will use the notations of [6] and [8]. Let us call w-slice the following matrix:

$$\overline{w}(k,-t) = \sum_{i=0}^{p} \overline{c}_3(k-i,t)\mathbf{W}(i)$$

where the W(i) are constant $r^2 \times r^2$ matrices.

$$\sum_{j=0}^{p}\mathbf{A}(j)\overline{w}(k,j+t)=\sum_{i=0}^{p}\Big(\sum_{j=0}^{p}\mathbf{A}(j)\overline{c}_3(k-i,t-j)\Big)\mathbf{W}(i)$$

$$\sum_{j=0}^{p}\mathbf{A}(j)\overline{w}(k,j+t)=0 \quad \text{if } t>q \qquad (4)$$

Equations (4) are the wslice counterparts of the higher order normal equations (2).

Let $\mathbf{W}(i)=\mathbf{I}_r\otimes\mathbf{A}(i)$, and calculate the wslice for t=q:

$$\overline{w}(q,t)=\sum_{i=0}^{p}\overline{c}_3(q-i,-t)(\mathbf{I}_r\otimes\mathbf{A}(i))$$

$$\overline{w}(q,t)=\sum_{i=0}^{p}\sum_{j=0}^{+\infty} unvec\,[(\mathbf{H}(j)\otimes\mathbf{H}(j+q-i)\otimes \mathbf{H}(j-t))\overline{\Gamma}_{3w}](\mathbf{I}_r\otimes\mathbf{A}(i))$$

$$\overline{w}(q,t)=\sum_{j=0}^{+\infty}\sum_{i=0}^{p}\mathbf{H}(j-t)\overline{\Gamma}_{3w}(\mathbf{H}(j)\otimes \mathbf{H}(j+q-i))^T(\mathbf{I}_r\otimes\mathbf{A}(i))$$

Taking the z transform of the last equation, we finally obtain:

$$\mathrm{W}(q,z)=\mathrm{H}(z^{-1})\overline{\Gamma}_{3w}[\mathbf{H}^T(0)\otimes\mathbf{B}^T(q)]$$

which can also be written:

$$\mathrm{A}(z^{-1})\,\mathrm{W}(q,z)=\mathrm{B}(z^{-1})\overline{\Gamma}_{3w}[\mathbf{H}^T(0)\otimes\mathbf{B}^T(q)] \quad (5)$$

As $\mathbf{A}(z)$ and $\mathbb{B}(z)$ have no common root, the recursion (5) holds with minimal order p. Therefore $\mathrm{W}(q,z)$ is a full rank slice (see [6] for definitions about full rank slices).

With this result, consistent AR estimates can always be obtained from equations (4). The system to solve is then made of linear combinations of cumulants instead of cumulants. The recursive resolution can be achieved with the algorithm of section II with the following choice of $\underline{z}(n)$:

$$\underline{z}(n)=\sum_{i=0}^{p}\underline{y}(n-q)\otimes\mathbf{A}(i)\underline{y}(n-q-i)$$

Note that in the AR case, the matrix of the system is block triangular, and the elements of the diagonal are $\overline{\Gamma}_{3w}[\mathbf{H}^T(0)\otimes\mathbf{B}^T(q)]$. Such a matrix is full rank, which confirms the previous result deduced from recursion (5).

Of course, the real $\mathbf{A}$(i) are not known as they are precisely what we look for. Therefore, in the computation of z(n), the current estimates of the A(i) instead of the real ones. The convergence of this approach has not been shown yet, but, as simulations give encouraging results, further research are carried out on this topic.

IV- Simulations.

In this part, we give results of simulations in order to show the performances of the new algorithm. Two models are under study:

<u>model 1</u>: AR(2) model with 3channels:

$$A(1)=\begin{pmatrix}0.5 & -0.4 & 0.2\\ -0.6 & 0 & 0.1\\ 0.1 & 0 & 0.3\end{pmatrix} \quad A(2)=\begin{pmatrix}0.5 & -0.2 & 0.8\\ 0.3 & 0.1 & 0\\ 0.5 & 0.2 & 0\end{pmatrix}$$

<u>model 2</u>: ARMA(1,1) model with 2 channels:

$$A(1)=\begin{pmatrix}0.5 & 0\\ 0.8 & -0.7\end{pmatrix}$$

$$B(0)=\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} \qquad B(1)=\begin{pmatrix}2 & 0\\ 1 & 0.6\end{pmatrix}$$

Simulations on both models are performed with the approach based on a 1D-slice and the approach based on the wslice.

<u>case 1</u>: for model 1, the matrix based on the 1D-slice is full rank. Results are given at Fig. 1. We note almost similar performances for both approaches. The convergence of the estimated parameters is a little slower for the w-slice method.

<u>case 2</u>: to show the necessity of the wslice method, model 2 has been chosen so that the 1D-slice based matrix has deficient rank. Fig. 2a shows that consitent AR parameters cannot be obtained from this single 1D-slice. The wslice method gives good results, as can be observed at fig. 2b.

Conclusion.

This paper deals with adaptive estimation of the AR parameters of a non gaussian multichannel ARMA processes. The proposed algorithm is a recursive computation of the pseudoinverse of a block Toeplitz matrix with rectangular blocks, and is based on a double lattice structure. In order to obtain consistent estimates of the AR parameters, the system to solve involves linear combinations of cumulants instead of single cumulants.

References:

[1] B. Priestley, "Spectral Analysis and Time Series", Academic Press, London, 1981.

[2] A. Swami, G. Giannakis, S. Shamsunder, "Multichannel ARMA Processes", IEEE Trans. on Signal Processing, vol. 42, n°4, April 94.

[3] J. Vidal, "Non Gaussian ARMA Estimation Based on Higher Order Statistics", PhD Thesis, UPC - Barcelona, Oct. 1993.

[4] J. W. Brewer, "Kronecker Products and Matrix Calculus in System Theory", IEEE Trans. on Circuits and Systems, vol. 25, pp. 772-781, 1978.

[5] H. Akaike, "Block Toeplitz Matrix Inversion", Siam. Journal on Appl. Math., vol. 24, n°2, March 1993.

[6] A. Swami, "System Identification Using Cumulants", PhD Thesis, USC - Los Angeles, Dec. 1988.

[7] A. Swami and J. Mendel, "Adaptive System Identification Using Cumulants", ICASSP 88, New york, April 1988.

[8] J. A. R. Fonollosa, J. Vidal and E. Masgrau Adaptive System Identification Based on Higher Order Statistics", ICASSP 91, Toronto, May 1991.

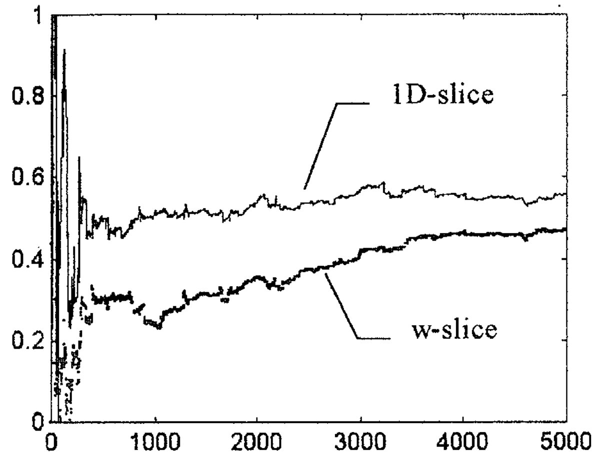

Fig 1: first coefficient of the A(1) matrix for model 1. The true value is 0.5.

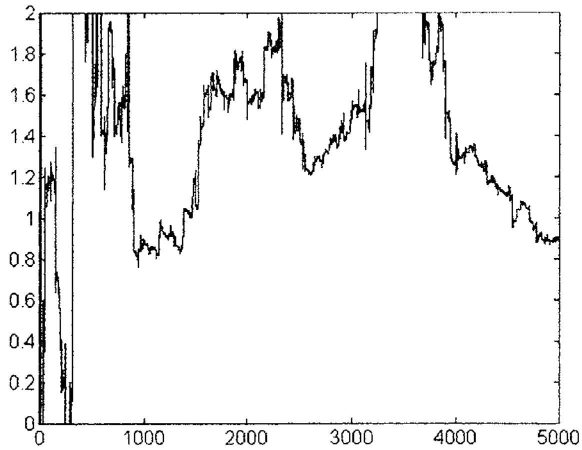

Fig 2a: first coefficient of the A(1) matrix for model 2, estimated using only one 1D-slice. The true value is 0.5.

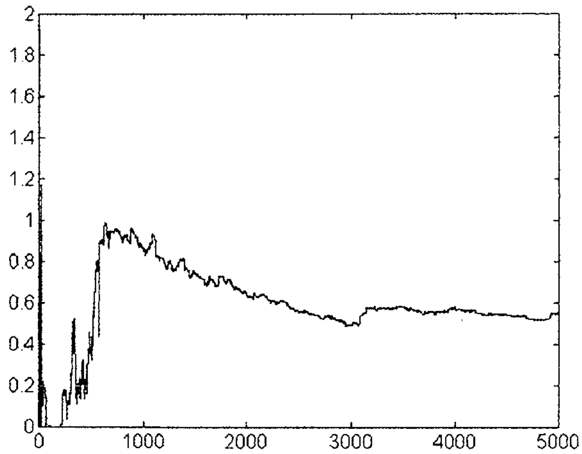

Fig 2a: first coefficient of the A(1) matrix for model 2, estimated using the w-slice method. The true value is 0.5

Time initialisations:

For m = 1 to p, do

$$\mathbf{F}_{m-1}(0)=\mathbf{B}_{m-1}(0)=\delta\begin{bmatrix}I_r\\0\\\vdots\\0\end{bmatrix} \qquad \delta \ll 1$$

$$\Delta^f_{m-1}(0)=\Delta^b_{m-1}(0)=0_{r\times v}$$

$$\gamma_m(0)=1$$

End of for loop

For n = 1 to Nmax, do:

Order initialisations:

$$\underline{f}_0(n)=\underline{b}_0(n)=\underline{y}(n)$$

$$\underline{\tilde{f}}_0(n)=\underline{\tilde{b}}_0(n)=\underline{z}(n)$$

$$\mathbf{F}_0(n)=\mathbf{B}_0(n)=\lambda\mathbf{F}_0(n-1)+z(n)y^T(n)$$

$$\gamma_0(n)=1$$

For m = 1 to p, do:

$$\Delta^f_{m-1}(n)=\lambda\Delta^f_{m-1}(n-1)+\frac{1}{\gamma_{m-1}(n-1)}\underline{\tilde{b}}_{m-1}(n-1)\underline{f}^T_{m-1}(n)$$

$$\Delta^b_{m-1}(n)=\lambda\Delta^b_{m-1}(n-1)+\frac{1}{\gamma_{m-1}(n-1)}\underline{\tilde{f}}_{m-1}(n)\underline{b}^T_{m-1}(n-1)$$

$$\Gamma^f_m(n)=-\mathbf{B}^{\dagger}_{m-1}(n-1)\Delta^f_{m-1}(n)$$

$$\Gamma^b_m(n)=-\mathbf{F}^{\dagger}_{m-1}(n)\Delta^b_{m-1}(n)$$

$$\tilde{\Gamma}^f_m(n)=-\mathbf{B}^{\dagger T}_{m-1}(n-1)\Delta^{bT}_{m-1}(n)$$

$$\tilde{\Gamma}^b_m(n)=-\mathbf{F}^{\dagger T}_{m-1}(n)\Delta^{fT}_{m-1}(n)$$

$$\underline{f}_m(n)=\underline{f}_{m-1}(n)+\Gamma^{fT}_m(n)\underline{b}_{m-1}(n-1)$$

$$\underline{b}_m(n)=\underline{b}_{m-1}(n-1)+\Gamma^{bT}_m(n)\underline{f}_{m-1}(n)$$

$$\underline{\tilde{f}}_m(n)=\underline{\tilde{f}}_{m-1}(n)+\tilde{\Gamma}^{fT}_m(n)\underline{\tilde{b}}_{m-1}(n-1)$$

$$\underline{\tilde{b}}_m(n)=\underline{\tilde{b}}_{m-1}(n-1)+\tilde{\Gamma}^{bT}_m(n)\underline{\tilde{f}}_{m-1}(n)$$

$$\mathbf{F}_m(n)=\mathbf{F}_{m-1}(n)-\Delta^b_{m-1}(n)\mathbf{B}^{\dagger}_{m-1}(n-1)\Delta^f_{m-1}(n)$$

$$\mathbf{B}_m(n)=\mathbf{B}_{m-1}(n)-\Delta^f_{m-1}(n)\mathbf{F}^{\dagger}_{m-1}(n)\Delta^b_{m-1}(n)$$

$$\gamma_m(n-1)=\gamma_{m-1}(n-1)-\underline{\tilde{b}}^T_{m-1}(n-1)\mathbf{B}^{\dagger T}_{m-1}(n-1)\underline{b}_{m-1}(n-1)$$

End of order loop

End of time loop

Table 1: the proposed algorithm.