An improved 2-D lattice filter and its entropy relations

Ayşin Ertüzün
Boğaziçi University, Department of Electrical and Electronic Engineering, Bebek, Istanbul, Turkey

Ahmet H. Kayran and Erdal Panayirci
Istanbul Technical University, Faculty of Electrical and Electronics Engineering, Ayazağa Kampusu, Istanbul, Turkey

Received 30 May 1990
Revised 22 August 1991

Abstract. In this paper, an improved lattice filter structure to model two-dimensional (2-D) autoregressive (AR) fields is presented. This work is the generalization of the three-parameter lattice filter developed by Parker and Kayran. The proposed structure generates four prediction error fields (one forward and three backward prediction error fields) at the first stage. After the first stage, two additional prediction error fields are generated using two of the backward prediction error fields at the output of the first stage. This leads to six prediction error fields whose linear combination defines the successive lattice stages and the reflection coefficients. A recursive relationship between the reflection coefficients of the lattice filter and the AR coefficients is derived. In addition, the new structure and the three-parameter lattice filter are compared from information-theoretic point of view. The entropy calculations are carried out for Gaussian distributed data. It is concluded that the new structure approximates the maximum entropy more closely compared to the three-parameter structure. The increase in entropy naturally leads to a more reliable and better modelling of AR data fields.


Résumé. Nous présentons dans cet article une structure de filtre en treillis améliorée pour la modélisation de champs autogérés- sifs (AR) bi-dimensionnels (2-D). Ce travail constitue la généralisation du filtre en treillis à trois paramètres développé par Parker et Kayran. La structure proposée génère quatre champs d'erreur de prédiction (un champ d'erreur directe et trois champs d'erreur rétrograde) dans le premier niveau. Après celui-ci, deux champs d'erreur de prédiction additifs sont générés à l'aide de deux des champs d'erreur de prédiction rétrograde obtenus à la sortie du premier niveau. Ceci donne six champs d'erreur de prédiction dont la combinaison linéaire définit les niveaux de treillis successifs et les coefficients de réflexion. Nous dérivons une relation récursive entre les coefficients de réflexion du filtre en treillis et les coefficients AR. De plus, nous comparons la structure nouvelle et le filtre en treillis à trois paramètres du point de vue théorie de l'information. Les calculs d'entropie ont été effectués pour des données à distribution gaussiennne. Nous en concluons que la structure nouvelle approche l'entropie maximale de plus près que ne le fait la structure à trois paramètres. L'accroissement d'entropie conduit naturellement à une modélisation meilleure et plus fiable des champs de données AR.

Keywords. 2-D lattice filter, entropy, 2-D AR modelling.

Correspondence to: Erdal Panayirci, Istanbul Technical University, Faculty of Electrical and Electronics Engineering, Ayazağa Kampusu, Istanbul, Turkey.

0165-1684/92/$05.00 © 1992 Elsevier Science Publishers B.V. All rights reserved
1. Introduction

The field of multidimensional digital signal processing has become increasingly important in recent years due to a number of trends in digital signal processing. In many fields such as biomedicine, space imagery, nuclear physics and seismic prospecting, the data to be processed are inherently multidimensional in character.

The theory of one-dimensional (1-D) lattice filter is well developed and well known; however there have been very few approaches to the 2-D lattice filtering and its use in 2-D spectral estimation [1, 3, 4, 16, 17, 19, 20].

A basic approach to the modelling of 2-D fields by the reflection coefficients was made by Marzetta [16] who has extended some results in 1-D linear prediction to the 2-D case. He has also shown that 2-D linear prediction filters have the minimum phase property and the correlation-matching property if they have continuous support and the 2-D Levinson algorithm is applicable to such filters. The continuous support filter is defined to have infinite support in one of the variables which leads to an unrealizable filter in practice. As a consequence of this theory, Marzetta has given a reflection coefficient representation of 2-D minimum phase filters.

Parker and Kayran [19] have introduced the concept of four prediction error fields which are combined into a quarter-plane 2-D lattice filter structure which is then used for spectral estimation. The lattice structure thus developed is simple to implement and satisfactory results for data that have certain correlation properties are obtained. However, the lattice filter obtained this way lacks sufficient parameters to represent all classes of AR data. It can represent only a subset of all AR filters of the same order.

Later, Parker et al. [20] have extended the quarter-plane model to the asymmetric half-plane where five prediction error fields involving six reflection coefficients are defined.

McGuffin and Lui [17] have defined a so-called matrix normal equation where the normal equation is embedded in it. The solution of the matrix normal equation leads to an algorithm similar to the Burg's algorithm.

A relation between multichannel 1-D AR models and single channel 2-D models with quadrant support is proposed by Therrien [23] where simultaneous computation of parameters of all four quarter-plane filters is possible. This method is generalized to multichannel 2-D models and applied to the problem of estimation of the 2-D autospectral and cross spectral components [24, 25].

Nam and O'Neill [18] have extended the lattice filters to the 2-D/3-D case where the 3-D filter can adaptively track the variation of both spatial and temporal changes of moving images. Using the adaptive 2-D/3-D lattice filters, adaptive predictive-control schemes have been proposed.

Hsieh et al. [7] have extended the order update 2-D lattice filter model proposed by Parker and Kayran into order update and time update recursion for more efficient computation. They have used 2-D lattice filtering method to extract the prediction parameters of the 2-D linear predictor. It has been shown that it is possible to improve the visual quality of reconstructed images.

1-D lattice models have found numerous applications in system identification and control, spectral estimation, frequency tracking and line enhancement, channel equalization, noise cancellation and speech analysis and synthesis.

On the other hand, the applications of 2-D lattice filter are in the areas of cascade filter synthesis for implementation in dedicated VLSI hardware [14, 15], 2-D spectral estimation [10, 13, 23-25, 27], 2-D filter design [9, 26], image data compression [11, 12] and adaptive predictive image coding [7, 13, 18]. Encouraging results have been obtained using 2-D lattice filters in digital image processing [7, 11-13, 18].

The lattice structure obtained by combining four prediction error fields consists of successive stages and the input/output relation of the \((m_1, m_2)\)th stage of this filter is given as follows [19]:

Signal Processing
\[
\begin{bmatrix}
    f(m_1, m_2; n_1, n_2) \\
    b_{10}(m_1, m_2; n_1, n_2) \\
    b_{11}(m_1, m_2; n_1, n_2) \\
    b_{01}(m_1, m_2; n_1, n_2)
\end{bmatrix}
= \Gamma(m_1, m_2)
\begin{bmatrix}
    f(m_1 - 1, m_2 - 1; n_1, n_2) \\
    b_{10}(m_1 - 1, m_2 - 1; n_1 - 1, n_2) \\
    b_{11}(m_1 - 1, m_2 - 1; n_1 - 1, n_2 - 1) \\
    b_{01}(m_1 - 1, m_2 - 1; n_1, n_2 - 1)
\end{bmatrix},
\]
(1)

where \(\Gamma(m_1, m_2)\) is the coefficient matrix of the \((m_1, m_2)\)th stage and is given as follows:

\[
\Gamma(m_1, m_2) =
\begin{bmatrix}
    1 & -\Gamma_{10}(m_1, m_2) \\
    -\Gamma_{10}(m_1, m_2) & 1 \\
    -\Gamma_{01}(m_1, m_2) & -\Gamma_{11}(m_1, m_2) \\
    -\Gamma_{01}(m_1, m_2) & -\Gamma_{11}(m_1, m_2) \\
    -\Gamma_{01}(m_1, m_2) & -\Gamma_{11}(m_1, m_2) \\
    -\Gamma_{10}(m_1, m_2) & 1 \\
    -\Gamma_{10}(m_1, m_2) & 1
\end{bmatrix}
\]
(2)

This lattice filter, as seen from (2), has three parameters at each stage and will be referred to as the three-parameter lattice filter. Thus starting with zero order lattice \((m_1, m_2) = (0, 0)\); four prediction error fields are generated from the input field; namely

\[
f(0, 0; n_1, n_2) = b_{10}(0, 0; n_1, n_2) = b_{11}(0, 0; n_1, n_2) = b_{01}(0, 0; n_1, n_2) = u(n_1, n_2).
\]

Their linear combination is used to calculate the prediction error fields of higher order stages.

Ideally, if the input signal to the lattice filter is a 2-D AR field, the coefficients of the field will be recovered exactly if there is one-to-one correspondence between the lattice parameters and the data support.

However, as mentioned above, the three-parameter lattice filter has three coefficients which are not sufficient to model all AR data fields. This filter gives satisfactory results for data with certain statistics. However, it is attractive since it is easy to implement. The number of samples in the data support grows quadratically with the lattice order. The number of samples in the data support is related to the \((m, m)\)th order lattice filter as \((m+1)^2\). As the order of the lattice filter increases by one to \((m+1, m+1)\), the support size increases by \(2m+3\). This implies that to model a 2-D AR process with sufficient number of parameters, the number of lattice parameters should also be increased by \(2m+3\) as the order of the lattice filter increases. This way there will be a one-to-one correspondence between the number of lattice parameters and the size of the data support. The sufficient number of lattice parameters for exact modelling of the 2-D AR process is one less than the size of the data support and related to the order of the lattice filter as \((m+1)^2 - 1\). However, an \((m, m)\)th order lattice filter has \(3m\) independent parameters and other parameters are functionally related to these independent parameters. Thus, the three-parameter lattice filter lacks the sufficient parameters to represent all classes of 2-D quarter-plane AR fields. This results in information loss which increases as the number of lattice stages increases, since the 2-D AR field is tried to be modelled with a fixed number of parameters at each stage. The information loss can be reduced by increasing the number of backward prediction error fields and in turn increasing the number of reflection coefficients.

For the first order filter, the number of reflection coefficients and the number of required coefficients for the exact modelling of the AR field is three. Thus, there is no information loss for this filter. However, as the order of the lattice filter increases, the difference between the required number of parameters and the lattice reflection coefficients increases which in turn increases the information loss. Thus a new lattice filter structure is presented which improves the prediction and decreases the information loss by increasing the number of backward prediction error fields and hence the number
of reflection coefficients. The new structure is obtained by improving the lattice filter structure of Parker and Kayran [19]. Since the information loss starts at the second stage of the three-parameter lattice, the improvement is introduced after the first stage of the three-parameter lattice filter. This new filter will be referred to as the extended lattice filter of diagonal form (ELDF) [1].

The two types of filter structures are compared from spectral modelling and information-theoretic points of view. It is known that the three-parameter lattice filter introduces information loss due to insufficient number of parameters. In a recent work [2], this loss is calculated quantitatively. Later, this analysis is carried out for the ELDF [1] and the two lattice structures are compared from this point of view.

The organization of the paper is as follows. In Section 2, the ELDF will be presented and the computer simulations for modelling of the AR data fields using different lattice filter structures will be given. In Section 3, the information loss introduced by each of the lattice filters will be calculated. The relationship between the input vector and the backward prediction error vector from information-theoretic point of view is also presented in this section. An analysis for calculating the entropy of the input vector in terms of the entropy of the backward prediction error is given. Computer simulations are carried out to compare the information loss introduced by different lattice filter structures for Gaussian data. Finally, in Section 4, the conclusions are presented.

2. Extended lattice filter of diagonal form

To decrease the amount of information loss, and in turn to model the AR data field in the best possible way, the lattice structure of Parker and Kayran [19] is improved [1].

Since both the first order AR data field and the first order lattice filter have three coefficients, there is a one-to-one correspondence between them and the number of lattice parameters is sufficient to model the data field. The number of coefficients for the second order AR data field is eight. That means that five new coefficients have to be calculated at this stage. However, the three-parameter lattice filter calculates three new coefficients, leading to six parameters. There are two parameters that are not directly calculated. Here, it is proposed to realize a lattice filter with five backward prediction error fields instead of three at the input of the second stage. This way, the number of lattice filter parameters will also be increased since there is a linear relationship between the number of backward prediction error fields and the number of reflection coefficients. The two additional backward prediction error fields are obtained at the output of the first stage from the backward prediction error fields $b_{10}(1, 1; n_1, n_2)$ and $b_{01}(1, 1; n_1, n_2)$ as follows:

\begin{align}
    b_{21}(1, 1; n_1, n_2) &= b_{10}(1, 1; n_1, n_2), \quad (3a) \\
    b_{12}(1, 1; n_1, n_2) &= b_{01}(1, 1; n_1, n_2), \quad (3b)
\end{align}

where $b_{21}(1, 1; n_1, n_2)$ and $b_{12}(1, 1; n_1, n_2)$ are the auxiliary error fields generated. These fields are delayed appropriately and are inputted to the second stage.

Figure 1a illustrates the generation of the prediction error fields for the ELDF. Figure 1b shows how the prediction error fields are linearly combined in the $(i_1, i_2)$th stage where $i_1, i_2 \geq 2$ [1]. Thus, by the generation of the two auxiliary error fields, at the input of the second stage there will be six error fields, where one of them is the forward and the remaining five are the backward prediction error fields. Their linear combination will define the next lattice stage and will be used to calculate the reflection coefficients of the next stage. Thus, lattice stages greater than one have six inputs and six outputs as seen from Figs. 1a and 1b.

The input/output relation of the first stage of the ELDF is also given by (1); however, the input/output relations of the higher order stages are given
as follows:

\[
\begin{bmatrix}
  f(m_1, m_2; n_1, n_2) \\
  b_{10}(m_1, m_2; n_1, n_2) \\
  b_{11}(m_1, m_2; n_1, n_2) \\
  b_{01}(m_1, m_2; n_1, n_2) \\
  b_{21}(m_1, m_2; n_1, n_2) \\
  b_{12}(m_1, m_2; n_1, n_2)
\end{bmatrix}
= \Gamma(m_1, m_2)
\]

\[
\begin{bmatrix}
  f(m_1 - 1, m_2 - 1; n_1, n_2) \\
  b_{10}(m_1 - 1, m_2 - 1; n_1 - 1, n_2) \\
  b_{11}(m_1 - 1, m_2 - 1; n_1 - 1, n_2 - 1) \\
  b_{01}(m_1 - 1, m_2 - 1; n_1 - 1, n_2) \\
  b_{21}(m_1 - 1, m_2 - 1; n_1 - 1, n_2 - 1) \\
  b_{12}(m_1 - 1, m_2 - 1; n_1 - 1, n_2 - 1)
\end{bmatrix}
\]

(4)

where \( \Gamma(m_1, m_2) \) is the coefficient matrix of the \((m_1, m_2)\)th stage and is given as follows:

\[
\Gamma(m_1, m_2) =
\begin{bmatrix}
  1 & -\Gamma_{10} & -\Gamma_{11} \\
  -\Gamma_{10} & 1 & -\Gamma_{01} \\
  -\Gamma_{11} & -\Gamma_{01} & 1 \\
  -\beta_{21} & -\beta_{10} & -\beta_{00} \\
  -\beta_{12} & -\beta_{02} & -\beta_{11} \\
  -\Gamma_{01} & -\beta_{21} & -\beta_{12} \\
  -\Gamma_{11} & -\beta_{10} & -\beta_{02} \\
  -\Gamma_{10} & -\beta_{00} & -\beta_{11} \\
  1 & -\beta_{20} & -\beta_{01} \\
  -\beta_{20} & 1 & -\beta_{22} \\
  -\beta_{01} & -\beta_{22} & 1
\end{bmatrix}
\]

(5)

The indices of the entries of the matrix \( \Gamma(m_1, m_2) \) are dropped for convenience.

In general, the input/output relation is given as follows:

\[
e(m_1, m_2; n_1, n_2) = \Gamma(m_1, m_2)e(m_1 - 1, m_2 - 1; n_1, n_2),
\]

(6)
where matrix $\Gamma(m_1, m_2)$ is defined by (2) or (5) depending on the values of $m_1$ and $m_2$. The vectors $e(m_1, m_2; n_1, n_2)$ and $\delta(m_1-1, m_2-1; n_1, n_2)$ are the vectors of prediction error fields generated by the filters of order $(m_1, m_2)$ and $(m_1-1, m_2-1)$, respectively. Note the shifts in the elements of the vector $\delta(m_1-1, m_2-1; n_1, n_2)$. The dimensions of vectors and matrices related to the ELDF is given in Table 1 for different stages of the lattice filter.

As clearly seen from Table 1, the ELDF has more parameters compared to the three-parameter lattice filter; thus, the increase in the number of parameters will lead to a better and more reliable prediction. The ELDF can exactly model second order AR data fields in addition to the first order

<table>
<thead>
<tr>
<th>Lattice stages</th>
<th>$m_1, m_2 &lt; 2$</th>
<th>$m_1, m_2 \geq 2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$e(m_1, m_2; n_1, n_2)$</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>$\delta(m_1-1, m_2-1; n_1, n_2)$</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>$\Gamma(m_1, m_2)$</td>
<td>$4 \times 4$</td>
<td>$6 \times 6$</td>
</tr>
<tr>
<td>$\Phi(m_1, m_2)$</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>$\Phi(m_1, m_2)$</td>
<td>$3 \times 3$</td>
<td>$12 \times 12$</td>
</tr>
<tr>
<td>$r(m_1, m_2)$</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>$\Phi$</td>
<td>$4 \times 4$</td>
<td>$6 \times 6$</td>
</tr>
</tbody>
</table>
fields [1]. However, as the order of the AR field increases, two additional backward prediction error fields do not introduce sufficient parameters to model all the data points in the support. A similar situation with that of the three-parameter lattice filter arises; however, this problem can be eliminated by extending the lattice filter at every stage, which is possible because of the modularity of the ELDF [1]. So the number of prediction error fields can be increased by generating two additional error fields using the corresponding backward prediction error fields $b_{10}(\cdot ; n_1, n_2)$ and $b_{01}(\cdot ; n_1, n_2)$ at every stage.

In order to calculate the lattice parameters at each stage, the prediction errors should be minimized in the mean-squared sense. The optimization can be performed in one of the fields (that is, in the forward prediction error field or in one of the backward prediction error fields) or in all the error fields at the same time. To do this optimization, the following expression for the mean-squared prediction error has to be solved:

$$P(m_1, m_2) = E[e^T(m_1, m_2; n_1, n_2)a\tilde{e}(m_1, m_2; n_1, n_2)]$$

$$= E[\tilde{e}^T(m_1 - 1, m_2 - 1; n_1, n_2)\Gamma(m_1, m_2)$$

$$\times a\Gamma(m_1, m_2)\tilde{e}(m_1 - 1, m_2 - 1; n_1, n_2)],$$

(7)

where $E[ ]$ is the expectation operator and $a$ is a diagonal matrix. The diagonal elements ($a_i$s) can take on values of either 1 or 0, where 1 implies that the optimization is done in the $i$th field.

Minimum mean-squared solution of (7) will lead to the following expression for the optimum lattice filter parameters:

$$\Phi(m_1 - 1, m_2 - 1)\gamma(m_1, m_2) = r(m_1 - 1, m_2 - 1),$$

(8)

where

$$\gamma(m_1, m_2) = [\Gamma_{10} \Gamma_{11} \Gamma_{01} \beta_{21} \beta_{20} \beta_{12}$$

$$\beta_{02} \beta_{10} \beta_{00} \beta_{01} \beta_{11} \beta_{22}]^\top$$

(9)

is the vector of lattice parameters of stage $(m_1, m_2)$. Note that for the first stage, $\gamma$ consists of only the first three elements. $\Phi$ and $r$ consist of the prediction error field correlations. $\Phi$ is a symmetric matrix consisting of the prediction error field correlations and $r$ is a vector of prediction error field correlations. These correlations are calculated by time averaging the data field.

The coefficient matrix of the AR process is obtained by using the following procedure:

1. Define $A_{00}(m_1, m_2)$, $A_{10}(m_1, m_2)$, $A_{11}(m_1, m_2)$, $A_{01}(m_1, m_2)$, $A_{21}(m_1, m_2)$, $A_{12}(m_1, m_2)$ as the AR coefficient matrices of the forward and the backward prediction error filters, respectively.

2. Define the first stage coefficient matrix as follows:

$$A_{00}(1, 1) = \begin{bmatrix} 1 & -\Gamma_{01}(1, 1) \\ -\Gamma_{10}(1, 1) & -\Gamma_{11}(1, 1) \end{bmatrix}.$$  

(10a)

3. The coefficient matrices $A_{10}(1, 1)$, $A_{11}(1, 1)$, $A_{01}(1, 1)$ are defined as follows:

$$A_{10}(1, 1) = \begin{bmatrix} -\Gamma_{10}(1, 1) & -\Gamma_{11}(1, 1) \\ 1 & -\Gamma_{01}(1, 1) \end{bmatrix},$$

(10b)

$$A_{11}(1, 1) = \begin{bmatrix} -\Gamma_{11}(1, 1) & -\Gamma_{10}(1, 1) \\ 1 & -\Gamma_{01}(1, 1) \end{bmatrix},$$

(10c)

$$A_{01}(1, 1) = \begin{bmatrix} -\Gamma_{01}(1, 1) & 1 \\ -\Gamma_{11}(1, 1) & -\Gamma_{10}(1, 1) \end{bmatrix}.$$  

(10d)

4. Increment $m_1$ and $m_2$ by one and form the augmented coefficient matrices. The augmented matrices are defined as follows:

$$\hat{A}_{00}(m_1, m_2) = \begin{bmatrix} A_{00}(m_1 - 1, m_2 - 1) & 0 \\ 0 & 0 \end{bmatrix},$$

(11a)

$$\hat{A}_{10}(m_1, m_2) = \begin{bmatrix} 0^T & 0 \\ A_{10}(m_1 - 1, m_2 - 1) & 0 \end{bmatrix},$$

(11b)

$$\hat{A}_{11}(m_1, m_2) = \begin{bmatrix} 0 & 0^T \\ 0 & A_{11}(m_1 - 1, m_2 - 1) \end{bmatrix}.$$  

(11c)
\[ \hat{A}_{01}(m_1, m_2) = \begin{bmatrix} 0 & A_{01}(m_1 - 1, m_2 - 1) \\ 0 & 0 \end{bmatrix} \]

(11d)

(5) For stages two and higher, calculate the AR coefficients using the following recursion:

\[
\begin{bmatrix}
A_{00}(m_1, m_2) \\
A_{10}(m_1, m_2) \\
A_{11}(m_1, m_2) \\
A_{02}(m_1, m_2) \\
A_{12}(m_1, m_2)
\end{bmatrix} = \Gamma(m_1, m_2)
\begin{bmatrix}
\hat{A}_{00}(m_1, m_2) \\
\hat{A}_{10}(m_1, m_2) \\
\hat{A}_{11}(m_1, m_2) \\
\hat{A}_{02}(m_1, m_2) \\
\hat{A}_{12}(m_1, m_2)
\end{bmatrix},
\]

(12)

where \( \Gamma(m_1, m_2) \) is defined by (5) and each of its elements form a partition. The matrix-vector multiplication is carried out partition-wise.

As a consequence of (3a) and (3b),

\[
\begin{align*}
A_{21}(1, 1) &= A_{10}(1, 1), \\
A_{12}(1, 1) &= A_{01}(1, 1),
\end{align*}
\]

(13a)

(13b)

and the augmented matrices related to them are given as follows:

\[
\begin{align*}
\hat{A}_{21}(m_1, m_2) &= \begin{bmatrix} 0 & 0^T \\ 0 & A_{21}(m_1 - 1, m_2 - 1) \end{bmatrix}, \\
\hat{A}_{12}(m_1, m_2) &= \begin{bmatrix} 0 & 0^T \\ 0 & A_{12}(m_1 - 1, m_2 - 1) \end{bmatrix}.
\end{align*}
\]

(14a)

(14b)

(6) Repeat Steps 4 and 5 until the desired lattice order is obtained.

Thus, after the lattice parameters are calculated, the AR coefficients can easily be found using (10) and (14).

The ELDF structure performs as satisfactorily as another newly developed 2-D lattice filter structure [1, 3, 4]. The two structures differ only in the way the auxiliary error fields are generated and they both model first and second order AR fields exactly and can model different subsets of the higher order AR fields.

2.1. Computer simulations for modelling AR fields using lattice filter structures

Computer simulations are carried out to confirm the theory presented. The input data field to the lattice filter is an AR field given as follows:

\[ u(n_1, n_2) = \sum_{k_1=0}^{M_1} \sum_{k_2=0}^{M_2} a(M_1, M_2; k_1, k_2) \times u(n_1 - k_1, n_2 - k_2) + \sigma^2 w(n_1, n_2), \]

(15)

where \( w(n_1, n_2) \) is a white Gaussian noise of variance \( \sigma^2 \). The data field used constitutes the input to different lattice structures and the coefficients of the AR data field are modelled by the lattice reflection coefficients. The forward prediction error transfer function can be defined as follows:

\[
H(M, M; z_1, z_2) = E_{00}(M, M; z_1, z_2) / E_{00}(0, 0; z_1, z_2) = [1 \ z_1^{-1} \ \ldots \ z_1^{-M}] [A] [1 \ z_2^{-1} \ \ldots \ z_2^{-M}],
\]

(16)

where the matrix \( A \) consists of either the original or the modelled AR field coefficients, namely \( a(k_1, k_2) \)'s. This way, the transfer function of the modelled AR data field can be obtained by starting with the first order lattice filter and increasing the order until the desired order \( M \) is reached. For the modelled AR field coefficients, matrix \( A \) equals matrix \( A_{00} \) which can be found by utilizing (12) and (14).

The theory is tested for several different data fields and it is observed that the ELDF is superior.
over the three-parameter lattice filter. Two of these examples are given in the following section.

**EXAMPLE I.** The coefficient matrix of the data field is given as follows:

\[
A_{\text{org}} = \begin{bmatrix}
1.00000 & -0.03000 \\
-0.20000 & 0.00080 \\
-0.08000 & 0.23000 \\
0.00000 & 0.00000 \\
0.00150 & 0.00200 \\
-0.47000 & -0.01000 \\
0.01000 & 0.00000 \\
0.00000 & -0.00800
\end{bmatrix}
\]

Computed 2-D lattice parameters for the first four stages are given in Tables 2 and 3 for different lattice structures.

Table 2
Reflection coefficients of the three-parameter lattice filter in Example 1

<table>
<thead>
<tr>
<th>Lattice order</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\Gamma_{10})</td>
<td>-0.014530</td>
<td>0.092360</td>
<td>0.038270</td>
<td>-0.051670</td>
</tr>
<tr>
<td>(\Gamma_{11})</td>
<td>-0.014430</td>
<td>-0.000577</td>
<td>-0.252500</td>
<td>0.016330</td>
</tr>
<tr>
<td>(\Gamma_{12})</td>
<td>0.011840</td>
<td>-0.015830</td>
<td>-0.074340</td>
<td>-0.00019</td>
</tr>
</tbody>
</table>

Table 3
Reflection coefficients of the ELDF in Example 1

<table>
<thead>
<tr>
<th>Lattice order</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\Gamma_{10})</td>
<td>-0.014530</td>
<td>0.071190</td>
<td>-0.002715</td>
<td>-0.036840</td>
</tr>
<tr>
<td>(\Gamma_{11})</td>
<td>-0.014430</td>
<td>-0.010680</td>
<td>0.022300</td>
<td>-0.010790</td>
</tr>
<tr>
<td>(\Gamma_{12})</td>
<td>0.011840</td>
<td>-0.004899</td>
<td>-0.003240</td>
<td>-0.002182</td>
</tr>
<tr>
<td>(\beta_{21})</td>
<td>-0.264600</td>
<td>-0.001800</td>
<td>0.001724</td>
<td></td>
</tr>
<tr>
<td>(\beta_{22})</td>
<td>0.026340</td>
<td>0.012360</td>
<td>-0.002086</td>
<td></td>
</tr>
<tr>
<td>(\beta_{32})</td>
<td>0.491900</td>
<td>-0.017800</td>
<td>0.009911</td>
<td></td>
</tr>
<tr>
<td>(\beta_{42})</td>
<td>-0.004646</td>
<td>0.007217</td>
<td>0.026290</td>
<td></td>
</tr>
<tr>
<td>(\beta_{10})</td>
<td>0.020250</td>
<td>0.002502</td>
<td>-0.082440</td>
<td></td>
</tr>
<tr>
<td>(\beta_{20})</td>
<td>-0.016240</td>
<td>0.012470</td>
<td>-0.011970</td>
<td></td>
</tr>
<tr>
<td>(\beta_{30})</td>
<td>0.001992</td>
<td>0.021110</td>
<td>-0.036860</td>
<td></td>
</tr>
<tr>
<td>(\beta_{40})</td>
<td>0.011510</td>
<td>-0.007441</td>
<td>0.010040</td>
<td></td>
</tr>
<tr>
<td>(\beta_{22})</td>
<td>0.000115</td>
<td>0.015030</td>
<td>-0.014230</td>
<td></td>
</tr>
</tbody>
</table>

The coefficient matrix obtained at the output of a third order three-parameter lattice filter is

\[
A_{\text{oo}}(3, 3) = \begin{bmatrix}
1.00000 & -0.01188 & \\
0.01330 & 0.01478 & \\
-0.07115 & 0.25820 & \\
0.00271 & -0.00015 & \\
0.00486 & 0.00324 & \\
-0.48770 & 0.00154 & \\
-0.0012 & 0.01757 & \\
0.00198 & -0.02225 & \\
\end{bmatrix}
\]

The coefficient matrix obtained at the output of a third order three-parameter lattice filter is

\[
A_{\text{oo}}(3, 3) = \begin{bmatrix}
1.00000 & -0.01085 & \\
0.01670 & 0.01346 & \\
-0.09286 & 0.00068 & \\
-0.03827 & 0.00445 & \\
0.01493 & 0.07434 & \\
0.00148 & -0.02234 & \\
0.00401 & -0.00354 & \\
-0.00364 & 0.25250 & \\
\end{bmatrix}
\]

The magnitude and contour plots of the original spectrum are given in Fig. 2. It can be seen from these figures that there are three prominent peaks. A third order ELDF can model the AR data field with great accuracy. It can distinguish the three spectral peaks clearly and accurately. The location and the shape of all the peaks and other characteristics of the original spectrum are obvious from the spectrum modelled by the ELDF (Fig. 3). However, the three-parameter structure cannot resolve the three peaks and moreover it creates many small peaks of the same magnitude. Thus, the three-parameter structure does not reproduce the characteristics of the original spectrum. It is not possible to observe the characteristics of the original spectrum from the spectrum modelled by the three-parameter lattice filter (Fig. 4).
Fig. 2a. Magnitude plot of the original spectrum in Example 1.

Fig. 2b. Contour plot of the original spectrum in Example 1.
Fig. 3a. Magnitude plot of the spectrum approximated by a third order ELDF in Example 1.

Fig. 3b. Contour plot of the spectrum approximated by a third order ELDF in Example 1.
The coefficient matrix obtained at the output of a third order ELDF is

\[
A_{00}(3,3) = \begin{bmatrix}
1.00000 & -0.00491 \\
-0.62380 & 0.15920 \\
0.00245 & -0.38010 \\
-0.02923 & -0.00686 \\
0.04802 & -0.02708 \\
0.16700 & 0.01426 \\
-0.05544 & 0.09982 \\
-0.04478 & -0.04801
\end{bmatrix}
\]

The coefficient matrix obtained at the output of a third order three-parameter lattice filter is

\[
A_{00}(3,3) = \begin{bmatrix}
1.00000 & -0.00978 \\
-0.62720 & 0.10270 \\
-0.04992 & -0.00661 \\
0.06007 & 0.00440 \\
0.06837 & -0.08961 \\
0.00525 & 0.05530 \\
-0.07561 & -0.04144 \\
-0.00384 & -0.06791
\end{bmatrix}
\]

As seen from the magnitude and contour plots of the spectrum of the original and the modelled data fields (Figs. 5-7), the ELDF is superior to the three-parameter model. It can distinguish the two spectral peaks very clearly. However, the three-parameter lattice filter cannot perform a reliable modelling and cannot distinguish the two peaks. These results are consequences of the fact that the ELDF can model the AR data fields whose dominant coefficients lay along the axis indicated in Fig. 8 [1]. Thus, ELDF can model a wider range of AR fields compared to the three-parameter model.

In both examples, the optimization was done in the forward field to compute the lattice parameters. Since this filter generates more prediction error fields, there will be more cross correlations to be calculated. This is obvious from the increased dimensions of matrix \( \Phi \) and the vector \( r \). Thus, the calculation of the reflection coefficients needs an
Fig. 4a. Magnitude plot of the spectrum approximated by a third order three-parameter lattice filter in Example 1.

Fig. 4b. Contour plot of the spectrum approximated by a third order three-parameter lattice filter in Example 1.
Fig. 6a. Magnitude plot of the spectrum approximated by a third order ELDF in Example 2.

Fig. 6b. Contour plot of the spectrum approximated by a third order ELDF in Example 2.
Fig. 7a. Magnitude plot of the spectrum approximated by a third order three-parameter lattice filter in Example 2.

Fig. 7b. Contour plot of the spectrum approximated by a third order three-parameter lattice filter in Example 2.
inversion of the $12 \times 12$ matrix $\Phi$. Moreover, no four-quadrant symmetry conditions are imposed on the original AR field, thus $E[f_{b_1}] \neq E[b_{01} b_{10}]$ or $E[f_{b_10}] \neq E[b_{11} b_{01}]$ or $E[f_{b_0}] \neq [b_{11} b_{10}]$ and the correlations have to be calculated individually. In this method, the backward prediction error filter coefficient matrices $A_{10}, A_{01}, A_{11}, A_{21}, A_{12}$ cannot be obtained by simple row and/or column permutations from the coefficient matrix of the forward prediction error filter $A_{00}$. The matrix–vector multiplication given in (12) has to be performed.

3. Entropy relations of 2-D lattice filters

In this section, the statement that there is information loss between the input vector and the backward prediction error vector in 2-D lattice filtering is substantiated. The probabilistic relationship between the input and the backward prediction error vectors will be extended to the 2-D case where the input and the backward prediction error vectors are lexicographic ordered vectors [8]. A novel analysis is carried out to calculate the entropies of the input vector and the backward prediction error vector. The entropy of the backward prediction error vector is calculated for the three-parameter lattice filter and the ELDF. It is shown that the improved lattice filter does indeed bring an improvement over the three-parameter case. It is also shown that as the number of extensions are increased, in the limit, there is a one-to-one correspondence between the order of the input vector (in other words, the number of sample points in the data support) and that of the backward prediction error vector. This way both vectors can be regained from the knowledge of each other and the entropy of the backward prediction error vector will asymptotically reach the entropy of the input which is the maximum entropy we are searching for. These concepts are confirmed with an example and the relationship between the lattice order and the entropy of the backward prediction error vector is derived for Gaussian case.

Let the row ordered lexicographic input vector $u(M_1, M_2; n_1, n_2)$ be given as follows:

\[
u(M_1, M_2; n_1, n_2) = [u(n_1, n_2) \ u(n_1, n_2-1) \ldots \ u(n_1-M_2) \ u(n_1-1, n_2) \ldots \ u(n_1-1, n_2-M_2) \ldots u(n_1-M_2, n_2) \ldots \ u(n_1-M_1, n_2)]^T.
\]

(17)

The autocorrelation matrix $R_u$ of the input vector is defined as follows for the case $M_1 = M_2 = M$:

\[
R_u = \begin{bmatrix}
R_{00} & R_{01} & \ldots & R_{0M} \\
R_{10} & R_{00} & \ldots & R_{0M-1} \\
\vdots & \vdots & \ddots & \vdots \\
R_{0M} & R_{00} & \ldots & R_{00}
\end{bmatrix}.
\]

(18)

Since it assumed that $M_1 = M_2 = M$, the input vector will have $(M+1)^2$ elements and it will be denoted by $u(M, M; n_1, n_2)$. Thus, matrix $R_u$ will have $(M+1)^2$ blocks of dimension $(M+1) \times (M+1)$.

Also define a lexicographic ordered backward prediction error vector $b(n_1, n_2)$ for the 2-D lattice filter as follows:

\[
b(n_1, n_2) = \begin{bmatrix}
b_0(n_1, n_2) \\
b_1(n_1, n_2) \\
\vdots \\
b_M(n_1, n_2)
\end{bmatrix}.
\]

(19)
Note that the $i$th partition of vector $b_{i}(n_1, n_2)$ consists of the backward prediction errors obtained at the output of the $i$th stage.

Each partition consists of three elements for the three-parameter lattice filter as follows:

$$
    b_i(n_1, n_2) = \begin{bmatrix}
    b_{10}(i, i; n_1, n_2) \\
    b_{11}(i, i; n_1, n_2) \\
    b_{01}(i, i; n_1, n_2)
\end{bmatrix}, \quad i \geq 2.
$$  \hfill (20)

For the ELDF, the first and second partitions (in other words $i = 0, 1$) are given by (20). However, the partitions $i \geq 2$ are given as follows:

$$
    b_i(n_1, n_2) = \begin{bmatrix}
    b_{10}(i, i; n_1, n_2) \\
    b_{11}(i, i; n_1, n_2) \\
    b_{21}(i, i; n_1, n_2) \\
    b_{12}(i, i; n_1, n_2)
\end{bmatrix}, \quad i \geq 2.
$$  \hfill (21)

The autocorrelation matrix $R_b$ of the backward prediction error vector $b_{i}(n_1, n_2)$ is given as follows:

$$
    R_b = E[b_{i}(n_1, n_2)b_i^T(n_1, n_2)].
$$  \hfill (22)

The entropy, the information content, of a random input vector $u(M_1, M_2; n_1, n_2)$ is defined by the following multiple integral [22]:

$$
    H_u = -\int_{-\infty}^{\infty} p_u(u) \ln[p_u(u)] \, du,
$$  \hfill (23)

where $p_u(u)$ is the joint probability density function of the input vector $u(M, M; n_1, n_2)$. For the special case where the real valued random input vector $u(M, M; n_1, n_2)$ is Gaussian distributed, with zero mean and with correlation matrix $R_u$, the entropy can be written as follows [1]:

$$
    H_u = \frac{1}{2} \ln(\det[R_u]) + \frac{1}{2} L_u \ln(2\pi e),
$$  \hfill (24)

where $L_u$ denotes the number of elements of the input vector $u(M, M; n_1, n_2)$ which is $(M + 1)^2$.

Similarly, the entropy $H_b$ of the backward prediction error vector is given as follows:

$$
    H_b = \frac{1}{2} \ln(\det[R_b]) + \frac{1}{2} L_b \ln(2\pi e),
$$  \hfill (25)

where $L_b$ is the number of elements of the vector $b_{i}(n_1, n_2)$.

However, it should be noted that at the zeroth stage of the lattice filter the backward prediction error fields are generated from the input data field, namely

$$
    b_{10}(0, 0; n_1, n_2) = b_{11}(0, 0; n_1, n_2) = b_{01}(0, 0; n_1, n_2) = u(n_1, n_2).
$$

Thus, the first three rows of the autocorrelation matrix are identical which implies that the determinant is 0. However, this problem can be overcome by taking into account that the backward prediction errors of the zeroth stage are all equal to the input, and one prediction error is sufficient to represent the zeroth stage in entropy calculations. This assumption is also consistent with the corollary that if $\det[A] = 0$, any highest order non-vanishing principle minors of $\det[A]$ can be used in determining the entropy [21].

Thus, the first partition of the backward prediction error vector can be modified as follows:

$$
    b_{0}(n_1, n_2) = b_{10}(M, M; n_1, n_2) = b_{11}(0, 0; n_1, n_2).
$$  \hfill (26)

The other modification is that $L_b$ is now replaced by $L_b - 2$ in (25). Thus, the entropy of the backward prediction error vector is given as follows:

$$
    H_b = \frac{1}{2} \ln(\det[R_b]) + \frac{1}{2} (L_b - 2) \ln(2\pi e).
$$  \hfill (27)

Let the relation between the input vector and the backward prediction error vector be given as follows:

$$
    b(n_1, n_2) = Tu(M, M; n_1, n_2),
$$  \hfill (28)

where vectors $b_{i}(n_1, n_2)$ and $u(M, M; n_1, n_2)$ are
given by (17) and (19), respectively, and $T$ is the transformation matrix.

Since the case given by (28) is a linear transformation, the Jacobian $J$ is simply the determinant of the inverse of the transformation matrix $T$ [22]. In this case the entropy of the backward prediction is given by

$$H_b = H_u - \ln(\det[T^{-1}]).$$

(29)

Thus, in general, a change in the coordinates gives rise to a change in the entropy. Only in the case of rotation of coordinates (or any transformation that preserves the measure) $J = 1$ and hence $H_b = H_u$.

The relation given by (29) holds and hence there is a one-to-one correspondence between the two sets of vectors if and only if the transformation matrix has an inverse, that is, if the transformation is reversible.

For the zeroth order case, (28) can be written as follows:

$$b_{10}(n_1, n_2) = u(n_1, n_2).$$

(30)

For the first order case, (28) can be written as follows:

$$
\begin{bmatrix}
1 & 0 & 0 & 0 \\
-\Gamma_{10}(1) & -\Gamma_{11}(1) & 1 & -\Gamma_{01}(1) \\
-\Gamma_{11}(1) & -\Gamma_{01}(1) & -\Gamma_{10}(1) & 1 \\
-\Gamma_{01}(1) & 1 & -\Gamma_{11}(1) & -\Gamma_{10}(1)
\end{bmatrix}
\begin{bmatrix}
b_{10}(0, 0; n_1, n_2) \\
b_{10}(1, 1; n_1, n_2) \\
b_{11}(1, 1; n_1, n_2) \\
b_{01}(1, 1; n_1, n_2)
\end{bmatrix}
= 
\begin{bmatrix}
u(n_1, n_2) \\
u(n_1, n_2 - 1) \\
u(n_1 - 1, n_2) \\
u(n_1 - 1, n_2 - 1)
\end{bmatrix}.$$  

(31)

Since $\det[T^{-1}] = 1$ for the zeroth order filter (see (30)), the backward prediction error vector and the input vector are probabilistically equivalent in that they contain exactly the same amount of information. This is to be expected since the backward prediction error $b_{10}(0, 0, n_1, n_2)$ is obtained from the input. For the first order filter,

$$\det[T^{-1}] = (1 - \Gamma_{10}^2(1) - \Gamma_{11}^2(1) - \Gamma_{01}^2(1))^{-1}.$$  

(32)

Even though the backward prediction error vector and the input vector do not contain the same amount of information, one can be obtained from the knowledge of the other.

However, for the three-parameter lattice filter of order two or higher, the support of data varies as a quadratic function of the filter order, but the size of the backward prediction error vector increases by three since three new backward prediction error fields are generated at each stage. Thus, the input vector cannot be regained from the knowledge of the backward prediction error vector and their entropies cannot be related to each other using the relation given by (29). The entropy of the backward prediction error vector can only be determined by using the relation given by (27).

For the ELDF, the third partition ($i = 2$) has five elements due to the auxiliary error fields generated at the input of the second stage. Thus, (28) can be written as follows for the second stage of the ELDF:

$$\begin{bmatrix}
b_{10}(0, 0; n_1, n_2) \\
b_{10}(1, 1; n_1, n_2) \\
b_{11}(1, 1; n_1, n_2) \\
b_{01}(1, 1; n_1, n_2) \\
b_{11}(2, 2; n_1, n_2) \\
b_{12}(2, 2; n_1, n_2)
\end{bmatrix} = T 
\begin{bmatrix}
u(n_1, n_2) \\
u(n_1, n_2 - 1) \\
u(n_1 - 1, n_2) \\
u(n_1 - 1, n_2 - 1) \\
u(n_1 - 1, n_2 - 2) \\
u(n_1 - 2, n_2 - 1) \\
u(n_1 - 2, n_2 - 2)
\end{bmatrix}.$$  

(33)

The transformation matrix $T$ for the second order filter is a nonsingular $9 \times 9$ square matrix as...
follows:

\[
\mathbf{r} = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-\Gamma_{10}(1) & -\Gamma_{11}(1) & 0 & 1 & -\Gamma_{01}(1) & 0 & 0 & 0 & 0 \\
-\Gamma_{11}(1) & -\Gamma_{00}(1) & 0 & -\Gamma_{01}(1) & 1 & 0 & 0 & 0 & 0 \\
-\Gamma_{01}(1) & 1 & 0 & -\Gamma_{11}(1) & -\Gamma_{00}(1) & 0 & 0 & 0 & 0 \\
a_{10}(0,0) & a_{10}(0,1) & a_{10}(0,2) & a_{10}(1,0) & a_{10}(1,1) & a_{10}(1,2) & a_{10}(2,0) & a_{10}(2,1) & a_{10}(2,2) \\
a_{11}(0,0) & a_{11}(0,1) & a_{11}(0,2) & a_{11}(1,0) & a_{11}(1,1) & a_{11}(1,2) & a_{11}(2,0) & a_{11}(2,1) & a_{11}(2,2) \\
a_{00}(0,0) & a_{00}(0,1) & a_{00}(0,2) & a_{00}(1,0) & a_{00}(1,1) & a_{00}(1,2) & a_{00}(2,0) & a_{00}(2,1) & a_{00}(2,2) \\
a_{21}(0,0) & a_{21}(0,1) & a_{21}(0,2) & a_{21}(1,0) & a_{21}(1,1) & a_{21}(1,2) & a_{21}(2,0) & a_{21}(2,1) & a_{21}(2,2) \\
a_{12}(0,0) & a_{12}(0,1) & a_{12}(0,2) & a_{12}(1,0) & a_{12}(1,1) & a_{12}(1,2) & a_{12}(2,0) & a_{12}(2,1) & a_{12}(2,2)
\end{bmatrix},
\]

(34)

where the \( a \)'s are the coefficients of the relevant second order backward prediction error filters obtained by the recursive relations given by (10)–(12) and (14).

This will lead to a one-to-one correspondence between the input vector and the backward prediction error vector for the second order filter. In the ELDF, the irreversibility starts at the third stage unless the extensions are repeated stage after stage.

To check the validity of the theory presented in this section computer simulations are carried out.

The input data field is a fourth order AR process generated using the Gaussian number generators in the library routines. The input data field is fed into 2-D lattice filters of order four and the entropies of the input data field and the backward prediction errors are calculated and compared. The autocorrelation matrix values are obtained by time averaging the known data. The entropy of the backward prediction errors are calculated numerically by directly taking the time averages of the backward prediction errors obtained at the output of each stage and forming the autocorrelation matrix. As a check, the entropies are also calculated using the relation given by (29) if the transformation is reversible.

**Example 3.** The coefficient matrix of the fourth order AR data field used is given below. The data field has \( 70 \times 70 \) points. Thus the averaging is done using the known data points.

\[
A_{\text{org}} = \begin{bmatrix}
1.00000 & -0.44000 & 0.00740 \\
-0.44000 & 0.22000 & -0.00950 \\
-0.12000 & 0.06900 & 0.01400 \\
-0.05200 & 0.02000 & -0.00300 \\
0.00600 & 0.00270 & 0.00001 \\
0.16000 & -0.40000 \\
-0.07000 & 0.17000 \\
0.02000 & -0.04800 \\
0.00150 & 0.01800 \\
-0.00320 & 0.00700
\end{bmatrix}
\]

The AR data with coefficients given by the above matrix is used as an input to the different lattice structures and their reflection coefficients are given in Tables 6 and 7. Table 8 gives the entropies of the input and the backward prediction error vectors where calculations are done using different methods. It also gives a comparison between the different lattice filter structures. As seen from Figs. 9 and 11, the ELDF approximates the input entropy more closely compared to the three-parameter lattice filter. This result is not surprising since the introduction of the auxiliary error fields creates more degrees of freedom.

The entropy calculations show that ELDF is equivalent to the recently developed method [1, 3, 4].
Table 6
Reflection coefficients and the prediction error power of the three-parameter lattice filter in Example 3

<table>
<thead>
<tr>
<th>Lattice order</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \Gamma_{10} )</td>
<td>---</td>
<td>0.5734</td>
<td>0.192200</td>
<td>0.048830</td>
<td>-0.004577</td>
</tr>
<tr>
<td>( \Gamma_{11} )</td>
<td>---</td>
<td>-0.3148</td>
<td>-0.008330</td>
<td>-0.012600</td>
<td>-0.017380</td>
</tr>
<tr>
<td>( \Gamma_{01} )</td>
<td>---</td>
<td>0.4565</td>
<td>0.004835</td>
<td>0.001393</td>
<td>0.406200</td>
</tr>
<tr>
<td>PEP</td>
<td>2.153</td>
<td>1.2470</td>
<td>1.189000</td>
<td>1.187000</td>
<td>0.990200</td>
</tr>
</tbody>
</table>

PEP: Prediction error power

Table 7
Reflection coefficients and the prediction error power of the ELDF in Example 3

<table>
<thead>
<tr>
<th>Lattice order</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \Gamma_{10} )</td>
<td>---</td>
<td>0.5734</td>
<td>0.192200</td>
<td>0.045850</td>
<td>0.005791</td>
</tr>
<tr>
<td>( \Gamma_{11} )</td>
<td>---</td>
<td>-0.3148</td>
<td>-0.009800</td>
<td>-0.020970</td>
<td>-0.086520</td>
</tr>
<tr>
<td>( \Gamma_{01} )</td>
<td>---</td>
<td>0.4565</td>
<td>0.004604</td>
<td>0.001192</td>
<td>0.403500</td>
</tr>
<tr>
<td>( \beta_{21} )</td>
<td>---</td>
<td>-0.018860</td>
<td>0.019710</td>
<td>-0.019610</td>
<td></td>
</tr>
<tr>
<td>( \beta_{30} )</td>
<td>---</td>
<td>-0.012430</td>
<td>-0.022120</td>
<td>0.007122</td>
<td></td>
</tr>
<tr>
<td>( \beta_{12} )</td>
<td>---</td>
<td>0.012520</td>
<td>-0.008602</td>
<td>0.021200</td>
<td></td>
</tr>
<tr>
<td>( \beta_{02} )</td>
<td>---</td>
<td>-0.000525</td>
<td>-0.015460</td>
<td>0.035730</td>
<td></td>
</tr>
<tr>
<td>( \beta_{10} )</td>
<td>---</td>
<td>0.356000</td>
<td>0.006516</td>
<td>0.041140</td>
<td></td>
</tr>
<tr>
<td>( \beta_{00} )</td>
<td>---</td>
<td>-0.398100</td>
<td>0.793000</td>
<td>-0.850900</td>
<td></td>
</tr>
<tr>
<td>( \beta_{01} )</td>
<td>---</td>
<td>0.000683</td>
<td>-0.022820</td>
<td>-0.008963</td>
<td></td>
</tr>
<tr>
<td>( \beta_{11} )</td>
<td>---</td>
<td>0.528200</td>
<td>0.819800</td>
<td>-0.802400</td>
<td></td>
</tr>
<tr>
<td>( \beta_{12} )</td>
<td>---</td>
<td>0.103800</td>
<td>-0.499600</td>
<td>-0.877000</td>
<td></td>
</tr>
<tr>
<td>PEP</td>
<td>2.153</td>
<td>1.2470</td>
<td>0.995300</td>
<td>0.789500</td>
<td>0.900900</td>
</tr>
</tbody>
</table>

PEP: Prediction error power

Table 8
Backward prediction error entropies for different lattice structures in Example 3

<table>
<thead>
<tr>
<th>Lattice order</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Data support</td>
<td>1 × 1</td>
<td>2 × 2</td>
<td>3 × 3</td>
<td>4 × 4</td>
</tr>
<tr>
<td></td>
<td>Input entropy</td>
<td>1.747</td>
<td>6.531</td>
<td>14.34</td>
<td>25.25</td>
</tr>
<tr>
<td>Three parameter lattice</td>
<td>BPEE (numerical)</td>
<td>1.792</td>
<td>6.230</td>
<td>10.71</td>
<td>15.06</td>
</tr>
<tr>
<td></td>
<td>BPEE (eq. (29))</td>
<td>1.747</td>
<td>5.520</td>
<td>IRR</td>
<td>IRR</td>
</tr>
<tr>
<td>ELDF</td>
<td>BPEE (numerical)</td>
<td>1.792</td>
<td>6.230</td>
<td>13.19</td>
<td>19.22</td>
</tr>
<tr>
<td></td>
<td>BPEE (eq. (29))</td>
<td>1.747</td>
<td>5.520</td>
<td>12.30</td>
<td>IRR</td>
</tr>
</tbody>
</table>

BPEE: Backward prediction error entropy  
IRR: Irreversible

4. Conclusions

Here we presented a new and improved lattice structure developed from the three-parameter lattice filter by generating two auxiliary prediction error fields at the input of the second stage. The ELDF can model first and second order AR fields exactly and can model a subset of higher order AR fields. However, if the extensions are repeated after each stage and if \( M - 1 \) extensions are carried out for an \( M \)th order AR field, the ELDF can model every order AR field. The ELDF of particular order is a subset of all AR filters of the same order, where the three-parameter lattice filter is a subset of the ELDF. The ELDF is comparable with the filter recently developed \([1, 3, 4]\) both from spectral modelling and information-theoretic points of view. As mentioned in the text, the difference...
between these two newly developed methods lies in the way they generate the auxiliary error fields. Hence these two new structures span different subsets of the AR filters of the same order.

From the information-theoretic point of view, the ELDF is superior to the three-parameter lattice filter which is obvious from Fig. 11. The superiority of the ELDF is the consequence of the fact that it generates more backward prediction error fields and in turn more reflection coefficients compared to the three-parameter lattice filter.

If the extensions are repeated stage after stage, the entropy of the backward prediction error vector will reach the entropy of the input. If there are
as many backward prediction error fields as the number of points in the data support, there will be a one-to-one correspondence between the input data vector and the backward prediction error vector. Thus, the data field and the backward prediction error vector can be recovered from the knowledge of the other.

Contrary to the 1-D case [6], the entropies of the input and the backward prediction error vectors are not exactly equal even if the transformation is reversible. In other words, even though the input and the backward prediction error vectors can be obtained from each other, they do not have the same entropy. This implies that the 2-D lattice filters do not preserve the norm and they do not perform triangular decomposition on the autocorrelation matrix of the input. The triangular decomposition in 1-D lattice filtering leads to a Gram-Schmidt type of orthogonalization of the input; however, such an orthogonalization is not performed in 2-D lattice filtering.

This filter can be used in the areas of spectral estimation and image processing. Using this filter, a new algorithm can be developed to estimate the spectral characteristics of a 2-D random field from its autocorrelation function. Lattice parameters can be used to design 2-D recursive digital filters from prescribed magnitude characteristics. AR modelling for fine arts painting analysis and predictive-control can be mentioned as specific real world applications of this new lattice filter.

References


