

Fig. 3. Adaptive delta modulator for speech coding.

TABLE III
DETERMINED FIXED-POINT ATTRIBUTES FOR THE ADAPTIVE DELTA MODULATOR

desired SQNR	group	signals	integer word-length	minimum word-length	optimum word-length
19.0	u0	prediction filter	-1	8	9
	u013	syllabic filter	-4	8	8
	u007	DAC	-1	8	8
20.5	u0	prediction filter	-1	20	20
	u013	syllabic filter	-4	18	18
	u007	DAC	-1	14	15

efficient search methods. This software was applicable not only to digital filters but to complex systems, such as the 8×8 IDCT conforming to the IEEE specifications, CELP speech vocoder, and MPEG audio algorithms as well. Although the optimization results are dependent on the input signal samples, it was possible to obtain the robust results by choosing the input signal carefully, adding guard bits, using adders with saturation logic, or applying several input signal files.

ACKNOWLEDGMENT

The authors appreciate the support of the Alta Group and L. Herlitz for developing the SPW version.

REFERENCES

- [1] L. B. Jackson, "On the interaction of roundoff noise and dynamic range in digital filters," *Bell Syst. Tech. J.*, pp. 159-183, Feb. 1970.
- [2] F. Catthoor, J. Vandewalle, and H. De Man, "Simulated annealing based optimization of coefficient and data word-lengths in digital filters," *Int. J. Circuit Theory Applications*, vol. 16, pp. 371-390, Sept. 1988.
- [3] C. Caraiscos and B. Liu, "A roundoff error analysis of the LMS adaptive algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 32, no. 1, pp. 34-41, Feb. 1984.
- [4] *SPW-The DSP Framework Hardware Design System User's Guide*, Alta Group of Cadence Design Systems, Inc., 1994.
- [5] *DSP Station User's Guide*, Mentor Graphics Corp., 1993.
- [6] S. Kim and W. Sung, "A floating-point to fixed-point assembly program translator for the TMS 320C25," *IEEE Trans. Circuits Syst.*, vol. 41, no. 11, pp. 730-739, Nov. 1994.
- [7] K.-I. Kum and W. Sung, "VHDL based fixed-point digital signal processing algorithm development software," in *Proc. Int. Conf. VLSI CAD '93*, Nov. 1993, pp. 257-260.

- [8] W. Sung and K.-I. Kum, "Word-length determination and scaling software for a signal flow block diagram," in *Proc. Int. Conf. Acoust., Speech, Signal Processing '94*, Adelaide, Australia, vol. 2, Apr. 1994, pp. 457-460.
- [9] *Fixed-Point Optimizer User's Guide*, Alta Group of Cadence Design Systems, Inc., Aug. 1994.
- [10] G. De Micheli, *Synthesis and Optimization of Digital Circuits*. New York: McGraw-Hill 1994.

Parameter Estimation for Two-Dimensional Vector Models Using Neural Networks

Lin Xu and Mahmood R. Azimi-Sadjadi

Abstract— This correspondence addresses the problem of two-dimensional (2-D) vector image model parameter estimation using a new recursive least squares (RLS)-based learning method. Vector autoregressive (AR) models with various 1-D and 2-D, causal and noncausal regions of support (ROS) are considered. Numerical results are presented which demonstrate the usefulness of the proposed scheme for on-line implementation.

I. INTRODUCTION

Finite-order autoregressive (AR), moving average (MA) and autoregressive moving average (ARMA) models have found numerous applications in parametric representations of both 1-D and 2-D discrete time processes. For example, such statistical models can be used in the area of time series prediction [1], image restoration [2], image data compression [3], and texture analysis [4].

Until very recently, research on statistical image modeling was mainly focused on 2-D scalar [5], [6] or 1-D vector causal models [7], [8]. However, by using causal models, typically close to half of the adjoining pixels are ignored in the process of modeling. Using noncausal modeling, on the other hand, the correlational information is extracted from as many pixels, surrounding a given pixel (or a column of pixels) in a full plane ROS, as possible and these correlations are reflected in the image model. Therefore, one would expect that noncausal models provide substantially better representation of the image process than their causal counterparts.

Conventional modeling methods typically require prior statistical knowledge of the image process. As a result, on-line implementation of these methods becomes very difficult or virtually impossible as the whole image field must be available prior to the modeling process. Moreover, these methods can be computationally laborious especially for higher order models. A neural network-based modeling algorithm is developed in this paper for 1-D and 2-D AR vector modeling with various ROS geometries. The learning scheme is based upon RLS algorithm which possesses inherent fast convergence behavior. This new modeling scheme does not require any prior statistical knowledge of the process and thus, lends itself for on-line implementation.

Manuscript received April 10, 1994; revised May 23, 1995. The associate editor coordinating the review of this paper and approving it for publication was Prof. J. N. Hwang.

L. Xu is with Digital and Applied Imaging, Eastman Kodak Company, Billerica, MA 01821 USA.

M. R. Azimi-Sadjadi is with the Electrical Engineering Department, Colorado State University, Fort Collins, CO 80523 USA.

IEEE Log Number 9415278.

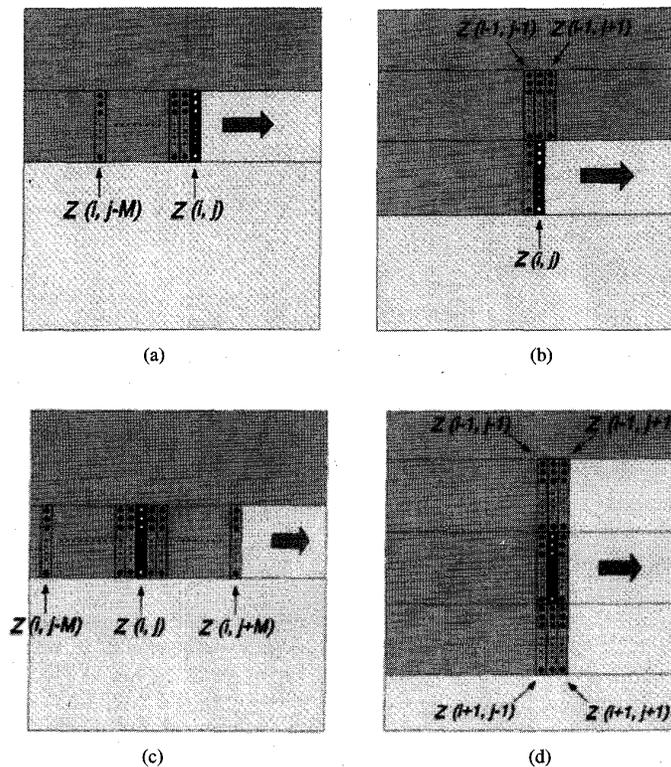


Fig. 1. (a) 1-D M th order causal ROS. (b) Extended 2-D first order causal NSHP ROS. (c) 1-D M th order noncausal ROS. (d) Extended 2-D first order noncausal ROS.

Simulation results for various 1-D and 2-D vector AR models are also presented.

II. IMAGE VECTOR MODELING AND PARAMETER ESTIMATION

Consider a zero mean image $\{x_{i,j}\}$, which is scanned vectorially in strips of width N . The direction of the scanning is assumed to be from left-to-right, and top-to-bottom. The strips may be overlapped to reduce the edge effects. Using such a scanning scheme, the image process can be represented by a vector or multichannel AR or ARMA process [7], [8]. A vector AR process with ROS, W , can be represented by

$$\underline{Z}(i, j) = \sum_{k, l \in W} \phi_{k, l}^t \underline{Z}(i - k, j - l) + \underline{E}(i, j) \quad (1)$$

where $\underline{Z}(i, j)$ represents an $N \times 1$ vector with the pixel intensity values as its elements, i.e., $\underline{Z}(i, j) = [x_{i,j} \ x_{i+1,j} \ \dots \ x_{i+N-1,j}]^t$, $x_{i+m,j}$, $m \in [1, N]$, denotes the intensity of the pixel at location $(i + m, j)$ of the column; vector $\underline{E}(i, j)$, which is defined similar to $\underline{Z}(i, j)$, represents the driving noise vector process and $\phi_{k, l}$'s, for $(k, l) \in W$, are $N \times N$ parameter matrices. The statistics of the driving process are

$$\begin{cases} \mathcal{E}[\underline{E}(i, j)] = \mathbf{0} \\ \mathcal{E}[\underline{E}(i, j)\underline{E}^t(i, j)] = Q_E \end{cases} \quad (2)$$

where Q_E is the covariance matrix of the error vector $\underline{E}(i, j)$ and $\mathcal{E}[\cdot]$ denotes the expectation operator.

The purpose of 2-D vector AR modeling is to find the parameter matrices $\phi_{k, l}$ for a given image and a choice of the geometry for ROS. Similar to 2-D scalar image modeling [6], the choice of ROS is one of the most important aspects in vector image model estimation. Scalar

ROS geometries can be extended to vector models. For a 1-D M th order vector causal ROS as shown in Fig. 1(a), W_1 is defined as

$$W_1 = \{(k, l) : k = 0, l \in [1, M]\}. \quad (3a)$$

For the extended 2-D M th order vector causal nonsymmetric half plane (NSHP) ROS as shown in Fig. 1(b) ($M = 1$), W_2 is defined as

$$W_2 = \{(k, l) : (k = 0, l \in [1, M]) \cup (k = 1, l \in [-M, M])\}. \quad (3b)$$

For the 1-D M th order vector noncausal ROS, as shown in Fig. 1(c), W_3 is defined as

$$W_3 = \{(k, l) : k = 0, l \in [-M, M]\}. \quad (3c)$$

For the extended 2-D M th order vector noncausal or full-plane ROS as shown in Fig. 1(d) ($M = 1$), W_4 is defined as

$$W_4 = \{(k, l) : k \in [-1, 1], l \in [-M, M], (k, l) \neq (0, 0)\}. \quad (3d)$$

Let us consider the causal vector model with ROS, as shown in Fig. 1(a). To estimate the parameter matrices ϕ_l for this model using the Yule-Walker method, transpose (1), premultiply both sides by $\underline{Z}(i, j - n)$, and then take the expectation. Using the orthogonality property [7], [8], $\mathcal{E}[\underline{Z}(i, j - n)\underline{E}(i, j)] = \mathbf{0}$, this yields

$$\rho_n = \sum_{l=1}^M \rho_{n-l} \phi_l + Q_E \delta(n) \quad (4)$$

where ρ_n is the covariance matrix of $\underline{Z}(i, j)$ and $\rho_n^t = \rho_{-n}$ for $n \in [0, M]$. The normal equation (4) can be arranged in a vector Yule-Walker system of equations from which ϕ_l 's, $l \in [1, M]$, and Q_E can be determined [8]. With the assumption that the image field is column wide-sense stationary within each strip [8], one may use

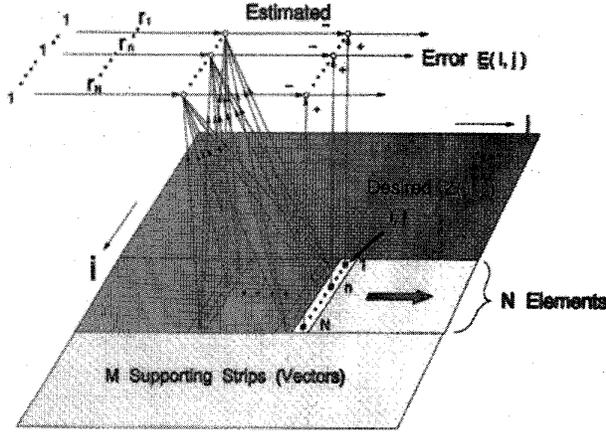


Fig. 2. Vector 2-D neural network modeling with 1-D M th order causal ROS with nonzero mean.

the ergodicity property to get reasonable estimates of ρ_n 's, $\hat{\rho}_n$'s for these parameters, i.e.,

$$\hat{\rho}_n = \frac{1}{(P-n) \times (P-N)} \sum_{i=0}^{P-1-N} \sum_{j=n}^{P-1} Z(i, j-n) Z^t(i, j) \quad (5)$$

where P is the dimension of the image. In [7], the elements of matrix ρ_n are estimated using a scalar version of (5) as the vector model is assumed to be formed of several scalar processes.

The conventional causal vector modeling method can easily be extended to the noncausal vector model with ROS, as shown in Fig. 1(c). In this case, the normal equation in (4) can be modified to

$$\rho_n = \sum_{l=-M}^M \rho_{n-l} \phi_l + Q_E \delta(n), \quad n \in [-M, M] \quad (6)$$

which can also be arranged in a vector Yule-Walker system of equations. Note that in the noncausal case the vector model is driven by a colored noise process [6], [8].

III. A NEURAL NETWORK-BASED SCHEME FOR PARAMETER ESTIMATION

In this section, a new approach for parameter estimation of vector image models is suggested using the RLS learning algorithm [9] which is inherently fast converging and suitable for on-line implementation.

Assuming that the image is wide-sense stationary within each strip, the vector process in (1) can be decomposed into N scalar processes [7]. Writing the AR models for the elements of vector $Z(i, j)$ yields

$$x_{i+n-1, j} = \sum_{k, l \in W} A_n^{k, l} Z(i-k, j-l) + e_{i+n-1, j}, \quad n \in [1, N] \quad (7a)$$

where $A_n^{k, l}$ represents the n th column of matrix $\phi_{k, l}$, i.e.

$$\phi_{k, l} = [A_1^{k, l}, A_2^{k, l}, \dots, A_N^{k, l}], \quad (k, l) \in W \quad (7b)$$

and

$$A_n^{k, l} = [a_{1, n}^{k, l}, a_{2, n}^{k, l}, \dots, a_{N, n}^{k, l}], \quad n \in [1, N] \quad (7c)$$

where $a_{m, n}^{k, l}$ is the m th element of vector $A_n^{k, l}$, $m \in [1, N]$.

From (7a), one can see that each element in $Z(i, j)$ is estimated based on all pixels inside the ROS, and the corresponding coefficients are the elements of parameter matrices $\phi_{k, l}$'s. Thus, a scalar 2-D neural network-based modeling scheme [10] can be used to extract



Fig. 3. Boat image.

the parameters of each scalar model. The structure of the network consists of a single layer of neurons as shown in Fig. 2 for the 1-D causal vector AR model with ROS of W_1 in Fig. 1(a). Note that since the ROS is identical for all the pixels in the vector, the network input vector is the same for all the elements in the vector. This input vector is given by

$$\underline{U}(i, j) = [x_{i, j-1}, x_{i+1, j-1}, \dots, x_{i+N-1, j-1}; \\ x_{i, j-2}, \dots, x_{i+N-1, j-2}; \dots; \\ x_{i, j-M}, \dots, x_{i+N-1, j-M}]^t \quad (8a)$$

and the corresponding parameter vector for the n th element of the vector, $\forall n \in [1, N]$, is

$$\underline{W}_n(i, j) = [a_{1, n}^{0, 1}(i, j), a_{2, n}^{0, 1}(i, j), \dots, a_{N, n}^{0, 1}(i, j); \\ a_{1, n}^{0, 2}(i, j), \dots, a_{N, n}^{0, 2}(i, j); \\ \dots, a_{1, n}^{0, M}(i, j), \dots, a_{N, n}^{0, M}(i, j)]^t \quad (8b)$$

In these vectors, (i, j) represents the position of the top element of the current vector where the estimation is taking place and $a_{m, n}^{k, l}(i, j)$, $m \in [1, N]$, represents the intermediate parameter for position (i, j) . Now using the RLS weight updating rule, the model parameters in $\underline{W}_n(i, j)$ can be estimated. This procedure involves the following steps [9]–[11]

$$\underline{K}_n(i, j) = \frac{P_n(i, j-1) \underline{U}(i, j)}{\lambda + \underline{U}^t(i, j) P_n(i, j-1) \underline{U}(i, j)} \quad (9a)$$

$$\underline{W}_n(i, j) = \underline{W}_n(i, j-1) + \underline{K}_n(i, j) \\ \times [x_{i+n-1, j} - \underline{W}_n^t(i, j-1) \underline{U}(i, j)] \quad (9b)$$

$$P_n(i, j) = \frac{1}{\lambda} [I - \underline{K}_n(i, j) \underline{U}^t(i, j)] P_n(i, j-1) \quad (9c)$$

where λ is the forgetting factor with $0 < \lambda \leq 1$, $\underline{K}_n(i, j)$ is the gain vector, and $P_n(i, j)$ is the inverse of data correlation matrix. The initial weights are either randomly set to small values or are initialized to zero and $P_n(0, 0) = \alpha I$, where α is a small value and I is the identity matrix. For a relatively large image, the effects of initial conditions would not impact the performance. The process is repeated for each new position of the window until the end of the strip is reached. At the end of each strip, a new strip is started in a zig-zag fashion to avoid discontinuities in the process. The strips are overlapped to reduce the edge effects. Generally speaking, better results are obtained for larger overlap.

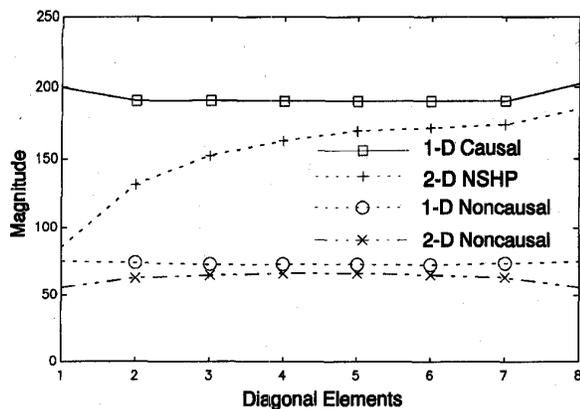


Fig. 4. Diagonal elements of error covariance matrix ($M = 1$).

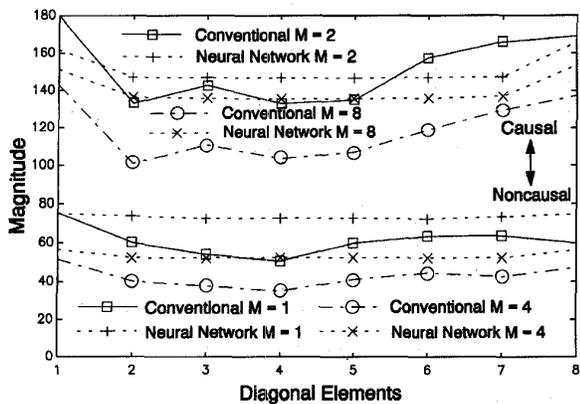


Fig. 5. Diagonal elements of error covariance matrices (conventional method versus neural network method).

The above procedure is repeated for each $n \in [1, N]$ sequentially. However, if a parallel processor is used, the procedure can be applied to all N elements in parallel. If the image has nonzero mean, a bias node can be added, as shown in Fig. 2, to compensate for this nonzero mean. With proper arrangement of the input and weight vectors, the parameters of other vector models, in Fig. 1, can also be estimated using the above procedure.

Although the above discussion only considered the case of causal and noncausal vector AR modeling, causal vector MA or ARMA model parameter estimation can be treated using similar methods as in the 2-D scalar MA or ARMA neural network-based modeling [10]. A similar scheme can be applied to each element of the vector with very little modifications in the structure of the network.

IV. SIMULATION RESULTS

The test image chosen was "Boat" image as shown in Fig. 3. This image has 256 gray levels and 512×512 pixels. Four cases of ROS in Fig. 1(a)-(d), i.e., 1-D causal and noncausal vector AR models, 2-D first order extended NSHP model and 2-D first order extended noncausal vector AR model were studied. In all cases, $N = 8$ with seven pixel overlap between strips and mean compensation input vector was also implemented. Initial weights were randomly initialized between -0.5 and 0.5 , $P(0,0) = 0.5I$ and $\lambda = 1$.

Under the assumption that the vector neural network parameter estimator is composed of several scalar parameter estimators, the

convergence behavior of the algorithm becomes similar to that of the scalar case [9]. The mean vector and the covariance matrix of the driving noise were computed to determine the performance of the algorithm for the four cases in Fig. 1(a)-(d). Fig. 4 shows the plots of the diagonal elements of the covariance matrix, Q_E , for the four cases when $M = 1$. Some 1-D causal and noncausal vector AR modeling results using the conventional methods are also presented for comparison. Fig. 5 gives the plots of the diagonal elements of the covariance matrix, Q_E , for various order $M = 1, 2, 4$, and 8 , 1-D causal and noncausal cases and using conventional and neural network schemes. From these results, the following observations can be made:

- 1) As expected, higher order models provide smaller error covariances (see Fig. 5), but the rate of improvement decreases as the order of the model increases.
- 2) Noncausal 2-D vector AR models have significantly smaller driving error covariances as they provide more accurate fit to the image data. Fig. 4 clearly discloses the difference between the causal and noncausal 1-D and 2-D vector AR models.
- 3) As can be seen from the plots in Fig. 4, for 1-D vector models with ROS inside a strip (Fig. 1(a) and (c)), the error covariances associated with the top and bottom pixels in the current vector are slightly larger than those in the middle. For 2-D causal vector models (Fig. 1(b)), the error covariances associated with lower pixels are significantly larger than those of the upper ones. For 2-D noncausal vector models (Fig. 1(d)), the error covariances are slightly smaller for those of the boundary pixels. The reason for this variation in the values of error covariances lies in the fact that each individual pixels has a different support region. Pixels with larger support region tend to have smaller error covariance values, while those with smaller support region have larger error covariance values. The off-diagonal elements of the error covariance matrices continued to decrease from causal 1-D vector models to 2-D noncausal vector models owing to the improvements in the accuracy in the modeling.
- 4) Comparing the results with those of the models obtained using the conventional method (Fig. 5) indicated that the models generated by the conventional method appear to have better fit. The reason for this discrepancy is that the conventional method computes the parameters of the model by solving a vector Yule-Walker in which ρ_n 's are obtained vectorially using (4); while using the neural network scheme the multichannel process is assumed to be decomposed into multiple scalar processes. Nevertheless, all models obtained using the neural network have much smaller covariance values at off-diagonal positions.
- 5) The values of elements in the bias compensation vector are close to the true image mean.

These observations indicate that the neural network-based vector modeling method provide a simple and efficient approach for no-line parameter estimation applications.

V. CONCLUSION

In this paper, a new method for image vector AR model parameter estimation was proposed. Several causal and noncausal ROS geometries were considered. The proposed scheme is based upon the application of a single layer neural network trained with a fast RLS-type algorithm. To apply the neural network modeling scheme developed in this correspondence, the vector model was decomposed into several scalar models. This algorithm is applicable to causal

as well as noncausal AR models with various ROS geometries. In addition, the algorithm is suitable for on-line implementation as prior statistical knowledge of the image is not needed. The simulation results show the usefulness of the proposed method.

REFERENCES

- [1] P. J. Brockwell and R. A. Davis, *Time Series: Theory and Methods*. New York: Springer-Verlag, 1991.
- [2] J. W. Woods and C. H. Radewan, "Kalman filtering in two dimensions," *IEEE Trans. Inform. Theory*, vol. IT-23, no. 4, pp. 473-482, July 1977.
- [3] E. J. Delp, R. L. Kashyap, and O. R. Mitchell, "Image data compression using autoregressive time-series models," *Patt. Recogn.*, vol. 11, pp. 313-323, Dec. 1979.
- [4] R. Chellappa and R. L. Kashyap, "Texture synthesis using 2-D noncausal autoregressive models," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 194-202, Feb. 1985.
- [5] R. L. Kashyap, "Characterization and estimation of two-dimensional ARMA models," *IEEE Trans. Inform. Theory*, vol. IT-30, no. 5, pp. 736-745, Sept. 1984.
- [6] S. Ranganath and A. K. Jain, "Two-dimensional linear prediction models—part I: Spectral factorization and realization," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, no. 1, pp. 280-299, Feb. 1985.
- [7] B. R. Suresh and B. A. Shenoi, "New results in two-dimensional Kalman filtering with applications to image restoration," *IEEE Trans. Circuits Syst.*, vol. CAS-28, pp. 307-319, Apr. 1981.
- [8] M. R. Azimi-Sadjadi, "New results in strip Kalman filtering," *IEEE Trans. Circuits Syst.*, vol. 36, no. 6, pp. 893-897, June 1989.
- [9] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice Hall, 2nd ed., 1991.
- [10] L. Xu and M. R. Azimi-Sadjadi, "Two-dimensional modeling of image random field using artificial neural networks," in *Proc. 1993 IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. I, Apr. 1993, pp. 581-584.
- [11] —, "Two-dimensional vector modeling of image random field using artificial neural networks," in *Proc. 1993 IEEE Int. Symp. Circuits and Systems*, vol. 1, Chicago, May 1993, pp. 838-841.

A New Polynomial Perceptron Based 64QAM Cellular Mobile Communications Receiver

Zeng-Jun Xiang and Guang-Guo Bi

Abstract—A new lattice polynomial perceptron (LPP) with faster convergence rate is introduced. The LPP based 64QAM cellular mobile communications receiver structure is described. Computer simulation results are given, which shows that, in 64QAM system, the performance of LPP is clearly superior to that of the other structures.

I. INTRODUCTION

Multilevel quadrature amplitude modulation (MQAM) is a bandwidth efficient transmission method for digital signals. However, the optimal receptions for MQAM mobile communication systems in the presence of frequency-selective fading and adjacent-channel

Manuscript received March 18, 1994; revised May 23, 1995. The associate editor coordinating the review of this paper and approving it for publication was Prof. J. N. Hwang.

Z.-J. Xiang is with INRS-Telecommunications, Verdun, Canada H3E 1H6. G.-G. Bi is with the Department of Radio Engineering, Southeast University, Nanjing, P.R. China.

IEEE Log Number 9415256.

interference (ACI) are inherently nonlinear problems. Nonlinear structures and the corresponding nonlinear processing methods are therefore required to achieve fully or nearly optimal performance. Neural network is one kind of nonlinear signal processing techniques and shows the potential for solving the above problems.

In [1], a model of polynomial perceptron (PP) is investigated in detail and a new fractionally spaced bilinear perceptron (FSBLP) model is introduced. Their nonlinear mapping abilities are evaluated. Applications of PP and FSBLP for fading channel equalization and co-channel interference suppression in a 16QAM receiver system are considered. Simulation results show that the performance of FSBLP is clearly superior to that of the other structures. However, when FSBLP is applied to 64QAM communication system with fading and ACI, its performance is worse.

In this paper, we introduce a new lattice polynomial perceptron model. The new model, as will be shown in Section IV, achieves the same performance as that of FSBLP in 16QAM system with fading and ACI. While in 64QAM system, its performance is clearly superior to that of the other structures.

The rest of the paper is organized as follows. The model of PP is presented in Section II. Section III introduces the model of LPP. Application examples, computer simulation results and discussions are given in Section IV. Section V includes concluding remarks.

II. POLYNOMIAL PERCEPTRON MODEL

The model of PP can be defined as [1], [2]

$$y(n) = f(P_W^L\{X(n)\})$$

where, $f(\cdot)$ is a nonlinear function of the sigmoid type. P_W^L is a degree- L polynomial function with coefficient vector W ,

$$\begin{aligned} P_W^L\{X(n)\} = & w_0 + \sum_{i_1=1}^N w_{i_1} x(n-i_1+1) \\ & + \sum_{i_1=1}^N \sum_{i_2=1}^N w_{i_1 i_2} x(n-i_1+1)x(n-i_2+1) \\ & + \cdots + \sum_{i_1=1}^N \sum_{i_2=1}^N \cdots \sum_{i_L=1}^N w_{i_1 i_2 \cdots i_L} \\ & \cdot x(n-i_1+1)x(n-i_2+1) \cdots x(n-i_L+1) \end{aligned}$$

$X(n) = [x(n), x(n-1), \dots, x(n-N+1)]^T$ is the N -dimensional input signal vector, and $W = [w_0, w_1, w_2, \dots, w_N, w_{11}, \dots, w_{NN}, \dots, \underbrace{w_{11 \cdots 1}}_L, \dots, \underbrace{w_{NN \cdots N}}_L]^T$ represents the

coefficient vector. When $L \rightarrow \infty$, $P_W^L\{X(n)\}$ is the well-known Volterra series.

III. LATTICE POLYNOMIAL PERCEPTRON

A. Model of LPP

The basic idea of the lattice realization of PP is to obtain the Gram-Schmidt orthogonal decomposition of the input signals of the polynomial perceptron, and then estimate the desired response signal as the output of the linear combination of the transformed signals through a nonlinear activation function $f(\cdot)$. For simplicity, we give the lattice realization method of the third order PP. Let $N = L = 3$