The INTERNATIONAL CONFERENCE ON ELECTRONICS
& OIL: FROM THEORY TO APPLICATIONS
March 05-06, 2013, Ouargla, Algeria

# Continuous-time Zhang Neural Networks for AR Spectral Estimator

Abderrazak Benchabane(1), Abdelhak Bennia(2), and Fella Charif(1)

(1) *Electrical Engineering Department, University of Ouargla, Algeria*
(2) *Signal processing Laboratory University of Constantine, Constantine Algeria*
*Benchabane.ab@univ-ouargla.dz, Abdelhak_bennia@laposte.fr, cherif.fe@univ-ouargla.dz*

***Abstract*—In this paper, we present an Auto-Regressive (AR) spectral estimator using a special kind of recurrent neural network proposed by Zhang called Continuous-time Zhang Neural Network (CZNN) to solve a system of linear equations. This neural network is characterized by an implicit dynamics and designed by defining a vector-valued error function instead of the usual scalar-valued norm-based error function used in the Gradient based Neural Networks (GNN). The output of the CZNN is the estimated AR coefficients so that the spectrum of the signal can be directly obtained in terms of the AR coefficients. For comparative purposes, the GNN model is also employed for AR parameters estimation.**

*Key-Words*— Spectral Estimation, AR model, Gradient-based neural network, Zhang neural network.

## I. INTRODUCTION

THE main objective of spectrum estimation is the determination of the power spectrum density from a finite set of measurements of a random process. Spectral estimation has been widely used in many practical applications such as radar, speech and communication, to mention a few [1]. Over the last century, a great of effort has been made to develop new techniques for high performance spectral estimation. Broadly, the developed techniques can be classified in two categories: nonparametric and parametric methods. The non parametric spectral estimation approaches are relatively simple, and easy to compute via the FFT algorithm. However, these methods require the availability of long data records in order to yield the necessary frequency resolution.

For the parametric approaches, we first design a model for the process of interest which is described by a small number of parameters. Based on this model, the spectral density estimate of the process can be obtained by substituting the estimated parameters of the model in the expression for the spectral density [1].

The most frequently used models in the literature are the autoregressive (AR), the moving average (MA), the autoregressive moving average (ARMA), and the sum of harmonics (complex sinusoids) embedded in noise. These parametric methods have a number of advantages as well as disadvantages over non-parametric methods. One of the advantages is their high resolution capability especially with a small number of data records. Also one of the disadvantages is the difficulty of determining a priori the order of the model for a given signal. In addition to these classical problems, many of the alternative spectral estimation methods require intensive matrix computation which my not be practical for real-time processing [2-4].

Since March 2001, a special kind of recurrent neural networks called the Zhang neural network (ZNN) has been proposed by Zhang and co-workers for solving online time-varying and/or static problems. Motivating by the fastness and robustness of these methods versus the gradient based neural networks [5-12]; we propose in this paper to employ the Continuous-time Zhang Neural Network (CZNN) for AR spectral estimator and compare it with the gradient-based neural network (GNN).

The major advantage of neural networks over other methods resides in their capability to perform more complex calculation in real times due to their parallel-disturbed nature. The neural network consists of a large number of simple devices; each one computes little more than weighted sums. Consequently the complexity of computation can be dramatically reduced and the total computation time is comparable to the response time of a single processor which can be very small [1,13]. The aim of our study is to provide an AR parameters estimator utilizing the real time characteristics of neural networks.

The paper is organized as follows: Section II states the AR parameters estimation problem. In section III, the dynamics of the CZNN and the GNN to solve this problem is developed. Computer simulation results for online spectral estimation based on the CZNN and the GNN models are presented in section IV followed by some concluding remarks.

## II. STATEMENT OF THE PROBLEM

Consider the parameter estimation problem of the noisy AR signal system [1,13]:

$$x(n) = \sum_{i=1}^{p} a_i x(n-i) + e(n) \quad (1)$$

where $a = [a_1, \ldots, a_p]^T$ is the unknown AR parameter vector; $x(n) = [x(n-1), \ldots, x(n-p)]^T$ is the regression vector of the AR process $x(n)$ of the known order $p$; $e(n)$ is white Gaussian process with variance $\sigma_n^2$. Our objective is to get an optimal estimate of the AR parameters using noisy observations $\{x(n)\}_{n=0}^{N-1}$, where $N$ is the number of data points. Generally $e(n)$ is small relative to $x(n)$, so we can estimate $x(n)$ using [1]:

$$\hat{x}(n) = x(n) - e(n) = \sum_{i=1}^{p} a_i x(n-i) \quad (2)$$

$$x(n) \approx \hat{x}(n) = \sum_{i=1}^{p} a_i x(n-i) \quad (3)$$

we first write (1) as linear equations in matrix and vector form [1,27]:

$$\begin{bmatrix} x(p+1) \\ x(p+2) \\ \vdots \\ x(N) \end{bmatrix} = \begin{bmatrix} x(p) & x(p-1) & \cdots & x(1) \\ x(p+1) & x(p) & \cdots & x(2) \\ \vdots & \vdots & \ddots & \vdots \\ x(N-1) & x(N-2) & \cdots & x(N-p) \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} \quad (4)$$

Let $\mathbf{x} = [x(p+1), x(p+2), \cdots, x(N)]^T$

$\mathbf{a} = [a_1, a_2, \cdots, a_p]^T$

$$\mathbf{S} = \begin{bmatrix} x(p) & x(p-1) & \cdots & x(1) \\ x(p+1) & x(p) & \cdots & x(2) \\ \vdots & \vdots & \ddots & \vdots \\ x(N-1) & x(N-2) & \cdots & x(N-p) \end{bmatrix}$$

then the above linear equations can be written[1,13]:

$$\mathbf{S} \cdot \mathbf{a} = \mathbf{x} \quad (5)$$

The AR coefficient $a_i$'s can be estimated by different ways, however, solving directly the linear equations instead to the computation of the inverse $\mathbf{S}^{-1}$ firstly, and then the multiplication $\mathbf{S}^{-1}\mathbf{x}$ is more efficient and accurate.

Once $a_i$'s computed, the estimated power spectrum can be written as [1]:

$$P(f) = \frac{\sigma_n^2}{\left|1 + \sum_{k=1}^{p} a_i e^{-j2\pi k f}\right|^2} \quad (6)$$

where $f$ is the normalized frequency.

## III. NEURAL NETWORK BASED SPECTRAL ESTIMATOR

To apply the neural network, the parameters estimation problem must be transformed to a minimization problem suitable for dynamic neural networks processing [14]. In this method the AR coefficients estimation problem is solved by constructing a network that has an energy function which is the same as the function to be minimised.

In the following, two kinds of recurrent neural networks for the AR parameters estimation will be presented. The former is based on the gradient-descent method in optimization to minimize a quadratic cost function [11]. The last, which is designed by Zhang et al , consists to minimize a vector-valued error function.

### *A. Continuous-time Zhang Neural Network*

Let $\mathbf{S} \in \Re^{N-p \times p}$ be the matrix data, $\mathbf{a} \in \Re^p$ the vector parameters to be estimated and $\mathbf{x} \in \Re^{N-p}$ the available signal samples. The CZNN system design is based on the set of linear equations $\mathbf{S}\mathbf{a}(t) - \mathbf{x} = 0$.

Commonly, the number of signal samples is larger than the model order. In the following, we suppose that $N > 2p$. As we can see, the system equation is over determined; therefore the equation can be transformed as [6,15]:

$$\mathbf{S}^T[\mathbf{S}\mathbf{a}(t) - \mathbf{x}] = 0 \quad (7)$$

where $\mathbf{S}^T$ denotes the transpose of $\mathbf{S}$

We can define now a vector –valued error function $\boldsymbol{\varepsilon}(t) \in \Re^p$ such as:

$$\boldsymbol{\varepsilon}(t) = \mathbf{S}^T[\mathbf{S}\mathbf{a}(t) - \mathbf{x}] \quad (8)$$

and then, we use the negative of the gradient as the descent direction:

$$\frac{d\boldsymbol{\varepsilon}(t)}{dt} = -\boldsymbol{\Gamma} f(\boldsymbol{\varepsilon}(t)) \quad (9)$$

where the parameter $\boldsymbol{\Gamma} \in \Re^{p \times p}$ is a positive-definite matrix used to scale the convergence rate of the solution and $f(\cdot): \Re^p \to \Re^p$ denote the activation function-vector.

Expanding (9) leads to:

$$\mathbf{S}^T\mathbf{S}\dot{\mathbf{a}}(t) = -\boldsymbol{\Gamma} f\left(\mathbf{S}^T[\mathbf{S}\mathbf{a}(t) - \mathbf{x}]\right) \quad (10)$$

To make every entry converges to zero at the same rate and at the same time, we take $\boldsymbol{\Gamma} = \eta\mathbf{I}$ with $\eta > 0$ which leads to:

$$\mathbf{S}^T\mathbf{S}\dot{\mathbf{a}}(t) = -\eta f\left(\mathbf{S}^T[\mathbf{S}\mathbf{a}(t) - \mathbf{x}]\right) \quad (11)$$

If we make $\mathbf{W} = \mathbf{S}^T\mathbf{S}$ and $\mathbf{b} = \mathbf{S}^T\mathbf{x}$, equation (11) becomes:

$$\mathbf{W}\dot{\mathbf{a}}(t) = -\eta f(\mathbf{W}\mathbf{a}(t) - \mathbf{b}) \quad (12)$$

In order to implement the CZNN, Eq. (12) can be written as the following explicit dynamics by using derivative and self-feedback for its $i$ th neuron's dynamics-equation [5]:

$$\dot{a}_i = -\eta f\left(\sum_{j=1}^{p} w_{ij} a_j - b_i\right) - \left(\sum_{j=1, j\neq i}^{p} w_{ij}\dot{a}_j\right) + (1 - w_{ii})\dot{a}_i \quad (13)$$

Figure (1) shows the architecture of the CZNN model for solving online the system (5) according to the equation (13). The circuit realizing the CZNN consists of $2p$ summers, $p$ integrators and $2p^2 + p$ weighted connections. As we know, the complexity of a neural network is defined as the total number of multiplications and additions per iteration, it can be seen that the CZNN requires $2p^2 + p$ multiplications and $2p$ additions per iteration.

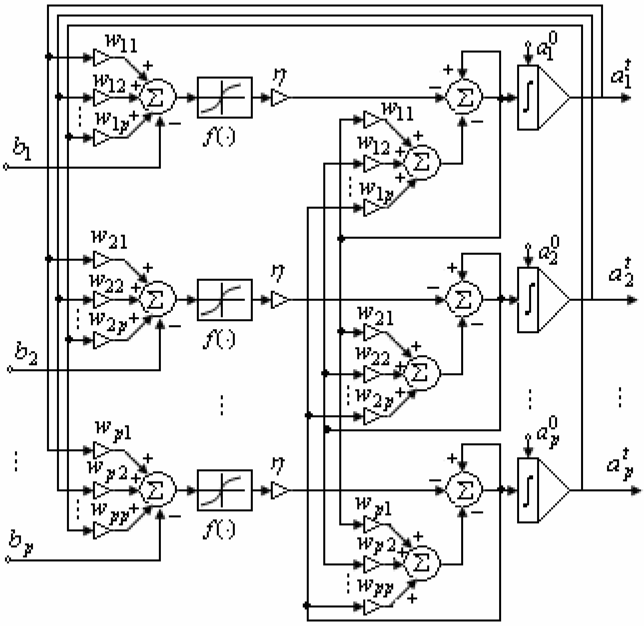


Fig. 1. Architecture of the CZNN model

### B. Gradient-based neural network

Conventional gradient-based neural networks (GNN) have been developed and widely investigated for online solution of the linear system [7,8,11,16]. The design procedure consists to define the norm-based scalar-valued error function $E(t)$ and then exploit the negative of its gradient as the descent direction to minimize $E(t)$ [11].

Consider the scalar-valued norm-based energy function

$$E = \frac{1}{2}\left\|\mathbf{S}\cdot\mathbf{a}-\mathbf{x}\right\|_2^2 \quad (14)$$

with $\|\cdot\|_2$ denoting the two norm of a vector. The minimum point of this cost function is the solution of the above linear system $\mathbf{x}=\mathbf{S}\cdot\mathbf{a}$.

To reach a minimum let us take the negative of the gradient of the energy function

$$-\frac{\partial E}{\partial \mathbf{a}} = -\mathbf{S}^{\mathbf{T}}\left(\mathbf{Sa}-\mathbf{x}\right) \quad (15)$$

By using a typical continuous-time adaptation rule, equation (15) leads to the following differential equation (linear GNN):

$$\dot{\mathbf{a}}(t) = \frac{d\mathbf{a}}{dt} = -\frac{\partial E}{\partial \mathbf{a}} = -\gamma\,\mathbf{S}^T\left(\mathbf{Sa(t)}-\mathbf{x}\right) \quad (16)$$

where $\gamma > 0$ is a design parameter used to scale the GNN convergence rate, and its should be set as large as hardware permits.

We could obtain the general nonlinear GNN model by using a general nonlinear activation function $f(\cdot)$ as follows [7]:

$$\dot{\mathbf{a}}(t) = -\gamma\, f\left(\mathbf{S}^T\mathbf{Sa(t)}-\mathbf{S}^T\mathbf{x}\right) \quad (17)$$

Or equivalently:

$$\dot{\mathbf{a}}(t) = -\gamma\, f\left(\mathbf{Wa}(t)-\mathbf{b}\right) \quad (18)$$

In the same manner as we have done in the first subsection, we could draw the structure of the GNN model as in figure (2). This second type of recurrent neural networks requires just $p^2+p$ multiplications and $p$ additions per iteration.

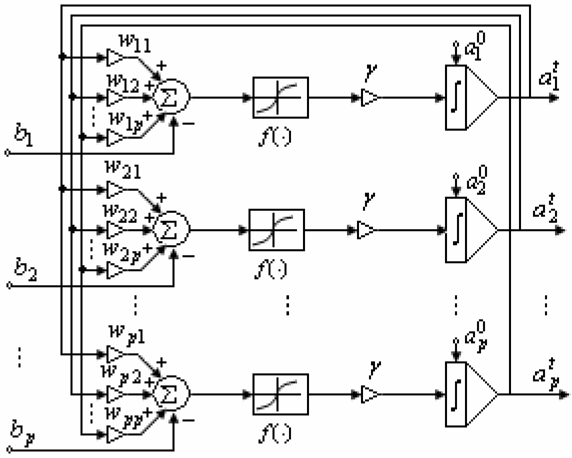


Fig. 2. Architecture of the GNN model

## IV. COMPUTER SIMULATIONS

In this section we present some examples to illustrate the accuracy and the fastness of the CZNN for AR coefficients estimation. In the first example we consider the AR signal

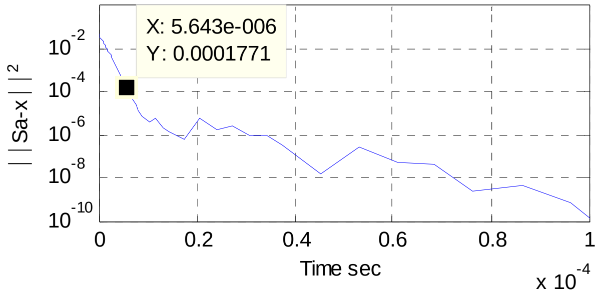
$$\begin{aligned} x(n) &= 2.0371x(n-1)-2.4332x(n-2) \\ &\quad +1.7832x(n-3)-0.7019x(n-4)+e(n) \end{aligned} \quad (19)$$

The input process $e(n)$ is white Gaussian process with variance $\sigma_n^2 = 4.10^{-5}$. To perform the CZNN for solving the parameters estimation problem, we let the number of process samples $N=64$ and the convergence rate parameter $\eta = 10^6$.

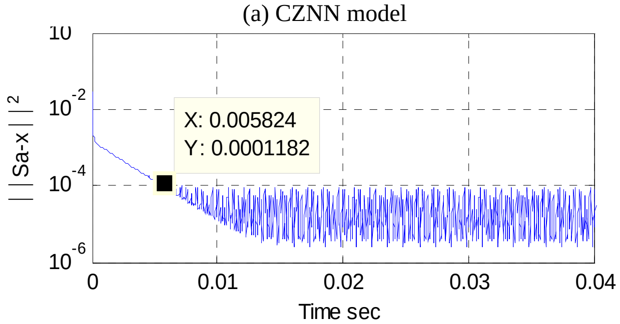
Figure (3) shows the convergence behavior of the state trajectory of the CZNN and GNN models. As we can see, starting from a random initial state $\mathbf{a}(0)$, the two networks converge to an optimal solution of the system (5). However, the CZNN is faster than the GNN. In figure (4), we can see that the CZNN reach an error over $10^{-4}$ in $5\mu s$. The same error is reached in $5ms$ by the GNN and just after the GNN model oscillates. We note here, to avoid the oscillatory phenomena, we can reduce the convergence rate parameter and the GNN becomes more slowly. In table (1), we summarize the computed results obtained by averaging 100 independent simulations and utilizing different algorithms. The Mean Square Error is defined as [13]:

$$MSE = \sqrt{\frac{1}{100}\sum_{i=1}^{100}\left\|\mathbf{a}^*-\mathbf{a}\right\|_2^2} \quad (20)$$

As we remark, the results obtained by the CZNN are similar to those obtained by the least square (LS) algorithm which is well known as an optimal estimator in presence of white Gaussian noise.

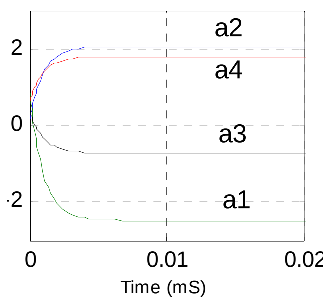



(a) CZNN model

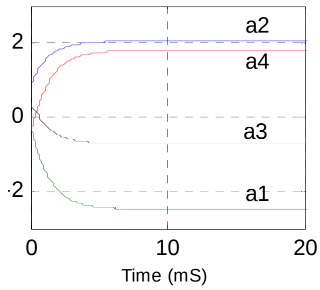



(b) GNN model

Fig. 4. Convergence behaviour of the norm error $E = \|\mathbf{S} \cdot \mathbf{a} - \mathbf{x}\|_2^2$ based on the (a) CZNN and (b) GNN model




(a) CZNN model




(b) GNN model

Fig. 3. Convergence behaviour of the state trajectory of the (a) CZNN and (b) GNN model

TABLE I COMPUTED RESULTS OF ESTIMATED PARAMETERS UZING DIFFERENT ALGORITHMS

| | $a_1$ | $a_2$ | $a_3$ | $a_4$ | MSE |
|---|---|---|---|---|---|
| **True** | -2.037 | 2.433 | -1.783 | 0.701 | 0 |
| **LS** | -2.020 | 2.387 | -1.729 | 0.683 | 0.2924 |
| **Burg** | -2.007 | 2.350 | -1.676 | 0.646 | 0.3222 |
| **Yule** | -1.774 | 1.933 | -1.280 | 0.455 | 0.9985 |
| **GNN** | -2.010 | 2.369 | -1.701 | 0.661 | 0.3175 |
| **CZNN** | -2.020 | 2.387 | -1.729 | 0.683 | 0.2924 |

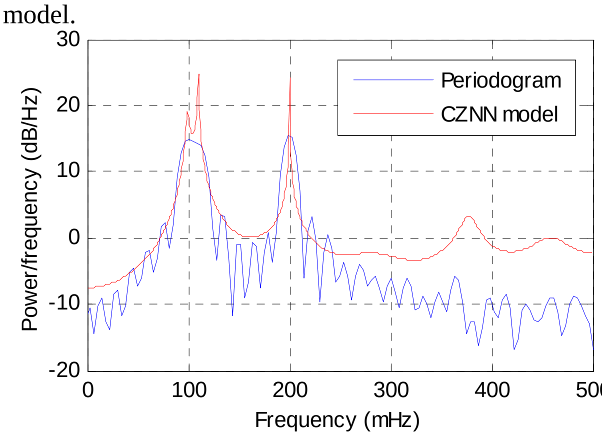
In the second example of computer simulations, we desire to estimate the spectrum of a signal. The signal test is a multiple sinusoids embedded in white Gaussian noise. The local SNR for the $k$ th sine wave is defined as [1]:

$$SNR_k = 10\log_{10}\frac{A_k^2}{2\sigma_n^2} \quad (21)$$

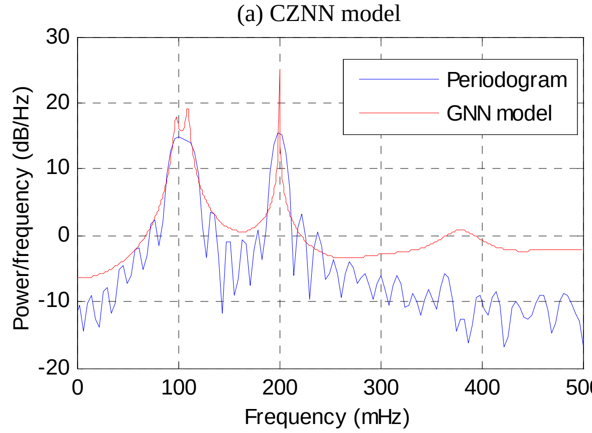
where $A_k$ is the amplitude of the $k$ th sinusoid.

The signal used in this simulation consists of three sinusoids. The normalized frequencies $f_1, f_2$ and $f_3$ were chosen as 0.1, 0.11 and 0.2 Hz respectively. The SNR was fixed to 15dB for all sinusoids. Also the model order was selected as 12 for the two networks. The convergence rate parameters were selected to $10^6$ for the CZNN and 10 for the GNN.

Using $N = 64$ data points involve that $f_1$ and $f_2$ are closer frequencies ( $f_2 - f_1 < 1/N$ ) and consequently the periodogram isn't able to separate the two frequencies. Figure (5) justifies the high resolution capability of the AR model.



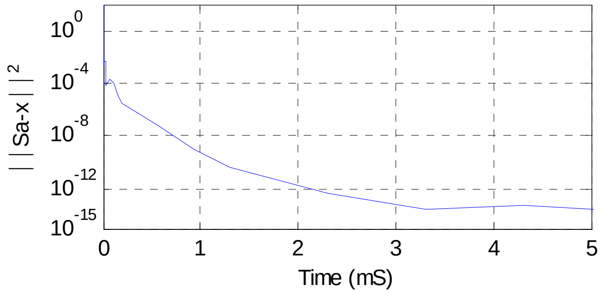

(a) CZNN model



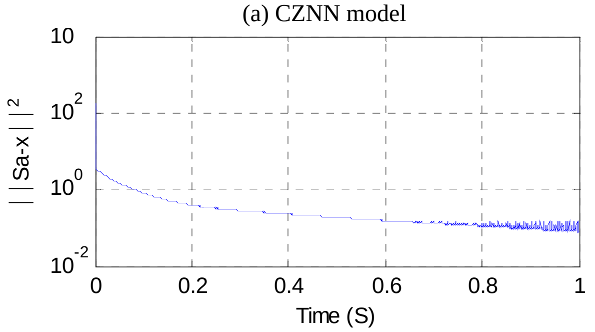

(b) GNN model

Fig. 5. The estimated spectrum by the (a) CZNN and (b) GNN model

As we can see from figure (6), the GNN model converge in over $1S$ to an error of $10^{-1}$, however, the CZNN converge to an error of $10^{-13}$ in $5mS$. Thus, we could summarize that the CZNN is more efficient and effective for online spectrum estimation, as compared with the conventional GNN.




(a) CZNN model




(b) GNN model

Fig. 6. Convergence behaviour of the norm error based on the (a) CZNN and (b) GNN model to estimate the spectrum.

## V. Conclusion

Recently, recurrent neural networks have been used as computational models for solving computationally intensive problems due to their inherent nature of parallel and distributed information processing. In this paper, an auto-regressive spectrum estimator using a new kind of recurrent neural network called continues Zhang neural network is proposed. This neural network is designed by defining a vector-valued error function instead of the usual scalar-valued norm-based error function used in the Gradient based Neural Networks. Computer simulations confirm the fastness and the accuracy of the proposed estimator.

## References

[1] S.K. Park, “Hopfield Neural Network for AR Spectral Estimator,” in Proc. IEEE'1990,1990, pp. 487-490.

[2] J. Marple., S. L., Adeli, H .Liu, “ Super-Fast Algorithm for Minimum Variance (Capon) Spectral Estimation,” Conference on Signals,Systems and Computers (ASILOMAR), 2010.

[3] G. O. Glentis, “ A Fast Algorithm for APES and Capon Spectral Estimation,” IEEE Transactions on Signal Processing. 56(9), 2008,pp.4207 - 4220.

[4] G. O. Glentis , “ Efficient Algorithms for Adaptive Capon and APES Spectral Estimation,” IEEE Transactions on Signal Processing. 58(1), 2010,pp.84-96.

[5] C. Yi and Y. Zhang, “Analogue recurrent neural network for linear algebraic equation solving,” ELECTRONICS LETTERS, Vol. 44(18),2008,pp.1-2.

[6] Y.Zhang, N.Tan, B.Cai, Z.Chen , “ MATLAB Simulink Modeling of Zhang Neural Network Solving for Time-Varying Pseudoinverse in Comparison with Gradient Neural Network,” Second International Symposium on Intelligent Information Technology Application.2008 pp.39-43.

[7] Y. Zhang , “Revisit the analog computer and gradient-based neural system for matrix inversion,” in Proc. IEEE Int. Symp. Intell. Control,Limassol, Cyprus, 2005, pp.1411–1416.

[8] Zhang. Y, Chen.K, Ma.W, and Xiao.L, “ MATLAB Simulation of Gradient-Based Neural Network for Online Matrix Inversion,” In: Huang, D. S., Heute, L. & Loog, M. (eds.), ICIC 2007, LNCS(LNAI), 4682, 2007,pp.98-109.

[9] L.Xiao ,Y.Zhang, “ Zhang neural network versus Gradient neural network for solving time-varying linear inequalities,” IEEE Transactions on Neural Networks, 22(10), 2011,pp.1676-1684.

[10] Y.Zhang , C.Yi, D.Guo,J. Zheng, “Comparison on Zhang neural dynamics and gradient-based neural dynamics for online solution of nonlinear time-varying equation,” Neural Comput & Applic ,2011,pp.1-7.

[11] Y. Zhang, K. Chen, and W. Ma, “MATLAB simulation and comparison of Zhang neural network and gradient neural network for online solution of linear time-varying equations,” in proceedings of International Conference on Life System Modeling and Simulation, Shanghai, China, September 2007, pp. 450-454.

[12] Y. Zhang and K. Chen, “Global exponential convergence and stability of Wang neural network for solving online linear equations,” Electronics Letters, vol. 44, no. 2, 2008,pp. 145-146.

[13] Y.Xia, M.S. Kamel ,“ A Cooperative Recurrent Neural Network Algorithm For Parameter Estimation of Autoregressive Signals,”2006 International Joint Conference on Neural Networks Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada July 16-21, 2006, pp. 2516- 2522.

[14] K. Chakraborty, K. Mehrotta, C.K. Mohan, S. Ranka, “An Optimization Network for Solving a Set of Simultaneous Linear Equations,” IJCNN., International Joint Conference on Neural Networks, vol. 4,1992,pp.516-521.

[15] Y.Xia, M.S. Kamel ,“ Cooperative Recurrent Neural Networks for the Constrained L1 Estimator,” IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 55, NO. 7, JULY 2007,pp. 3192- 3206.

[16] J.J. Hopfield, “Neural networks and physical systems with emergent collective computational abilities,” Proceeding of National Academic of Science in USA, Vol. 79, 1982,pp. 2554-2558.