Nonlinear Independent Component Analysis Using Generalized Polynomial Neural Network

P. GAO, W.L. WOO, and S.S. DLAY
School of Electrical, Electronic and Computer Engineering
University of Newcastle
Newcastle upon Tyne, NE1 7RU
UNITED KINDOM

Abstract: - In this paper, a new polynomial neuron-based network is proposed to tackle the problem of nonlinear Independent Component Analysis (ICA). We extend our research from a recently presented mono-nonlinearity mixture where a linear mixing matrix is sandwiched between two mutually inverse nonlinearities to a so-called multi-nonlinearity constrained mixing model. Our aim is to generalize the mono-nonlinearity mixing system to the situation that different nonlinearities are now allowed to be used for sources. Meanwhile, the theory of Series Reversion is adopted with the neural network demixer to make use of the a priori 'inverse' information between two layers of nonlinearities. The parameter learning algorithm for this special polynomial network demixer is also presented. Simulations have been carried out to verify the efficacy of the proposed approach. We demonstrate that the proposed network can successfully recover the original source signals in a blind mode under nonlinear mixing conditions.

Key-Words: - Independent Component Analysis, Blind Signal Separation, Series Reversion, Neural Network.

1 Introduction

With the rapid development during the last decade, Independent Component Analysis (ICA) has attracted considerable attention in both science and industry since it becomes one of the most powerful tools in Blind Signal Separation (BSS) [1-4 and reference therein]. The applications of ICA have involved in the area of wireless communications, biomedical signals analysis, and underwater acoustic signal processing *etc* [1, 3, 4]. Generally, the blind separation problem of independent sources can be defined as follows: Given a set of observations $\mathbf{x} = \begin{bmatrix} x_1(t) & x_2(t) & \cdots & x_p(t) \end{bmatrix}^T$ which are generated as a mixture of independent components $\mathbf{s} = \begin{bmatrix} s_1(t) & s_2(t) & \cdots & s_q(t) \end{bmatrix}^T$ according to

$$x_i = f_i\left(s_1, s_2, \ldots, s_q\right) \quad (1)$$

where f_i is an unknown differentiable bijective mapping, $i = 1, 2, \ldots, p$ and t is the time or sample index, the method of Independent Component Analysis (ICA) consists of estimating both the mixture mappings f_i 's and the original sources $s_i(t),\ i = 1, 2, \ldots, q$. In linear ICA, the mixing mapping takes the form of the linear combination as

$$x_i = f_i\left(s_1, s_2, \ldots, s_q\right) = m_{i1}s_1 + m_{i2}s_2 + \cdots + m_{iq}s_q \quad (2)$$

which is presupposed in most existing ICA algorithms. However, the linear assumption is always violated due to the existence of the nonlinear distortion in practice and the linear methods therefore fail to extract the original source signals [5]. For example, the auditory nervous system is modelled as a memoryless nonlinear system and many physiological signals in biomedical cases are nonlinearly distorted. Thus the identification of nonlinear dynamics should be taken into consideration in separation algorithms. Another instance is the recording of multiple speech source signals by carbon-button microphones which introduce some form of nonlinearity [6]. Hence, the search for a nonlinear solution becomes urgent and paramount in both theoretical and practical levels.

In this paper, based on an extension of the recently proposed mono-nonlinearity mixing model, we generalize the demixer to the situation that different nonlinearities can be utilized for the sources. A polynomial-based neural network is proposed as one of the solutions for nonlinear ICA. Furthermore, due to the special structure of the demixing network, the theory of Series Reversion is integrated into the network with modification to our current method.

2 Nonlinear Mixing and Demixing Model for ICA

Practically speaking, a realistic mixture needs to be nonlinear and concurrently capable of treating the linear mixture as a special case. In Nonlinear ICA, the separation problem become much more difficult than the conventional linear case since the independent property still holds even after the nonlinear transformation. Furthermore, it has been pointed out in [7] that there always exist infinite number of solutions in nonlinear separation of independent sources if the mixing functions f_i 's in (1) are not constrained. Recently, instead of using the general form of mixture displayed in (1), the so-called mono-nonlinearity mixing model which is originally stemmed from the theory of functional analysis is proposed in [8] and expressed as

$$\mathbf{x} = f\left(\mathbf{M}f^{-1}(\mathbf{s})\right) \tag{3}$$

where $\mathbf{M} = \begin{bmatrix} \mathbf{m}_1 & \mathbf{m}_2 & \cdots & \mathbf{m}_p \end{bmatrix}^T$ with dimension $p \times q$ and $\mathbf{m}_i = \begin{bmatrix} m_{i1} & m_{i2} & \cdots & m_{iq} \end{bmatrix}^T$ .For simplicity, we assume that the number of sources is equal to that of observations, i.e. $p = q = N$. It can be recognized that the structure of this model is actually one linear mixing matrix slotted into two layers of nonlinearities, one of which is the inverse function of the other. The term of 'mono-nonlinearity' results from the fact that identical nonlinear distortion is applied to each source signal. However, there is no guarantee that this condition is always fulfilled in practice. In fact, the channels between observations and sources are displayed to have arbitrary distortion due to the uncertainty of environment. Hence, in this paper, preserving the special relationship between the two layers in (3), we represent the 'multi-nonlinearity' constrained mixing system by the following model:

$$\mathbf{x} = \mathbf{F} \otimes (\mathbf{M}\mathbf{F}^{-1} \otimes \mathbf{s}) \tag{4}$$

where

$$\mathbf{F} = \mathrm{diag}\begin{bmatrix} f_1 & f_2 & \cdots & f_N \end{bmatrix},$$
$$\mathbf{F}^{-1} = \mathrm{diag}\begin{bmatrix} f_1^{-1} & f_1^{-1} & \cdots & f_N^{-1} \end{bmatrix},$$
$$\mathbf{F} \otimes \mathbf{u} = \begin{bmatrix} f_1(u_{11}) & f_1(u_{12}) & \cdots & f_1(u_{1T}) \\ f_2(u_{21}) & f_2(u_{22}) & \cdots & f_2(u_{2T}) \\ \vdots & \vdots & \ddots & \vdots \\ f_N(u_{N1}) & f_N(u_{N2}) & \cdots & f_N(u_{NT}) \end{bmatrix},$$

and $\mathbf{u}$ is a matrix with the dimension $N \times T$. Obviously, the expression in (4) will reduce to the mono-nonlinearity mixing model when $f_1 = f_2 = \cdots = f_N = f$ and can further treat the linear mixture as a special case if $\{f_i\}_{i=1}^N$ is linear. Moreover, the demixing system, shown in (5), is expected to inverse the mixing system and estimate the original sources directly.

$$\hat{\mathbf{s}} = \left[\mathbf{F}^{-1}\right]^{-1} \otimes \left(\mathbf{M}^{-1}\mathbf{F}^{-1} \otimes \mathbf{x}\right) = \mathbf{F} \otimes \left(\mathbf{W}\mathbf{F}^{-1} \otimes \mathbf{x}\right) \tag{5}$$

where $\mathbf{W}$ is the demixing matrix. From (5), it can be inferred that given the observed signals only, our aim is to estimate $\mathbf{F}$ and $\mathbf{W}$ such that the resulting transformed signals are mutually as independent as possible and statistically as close as possible to the source signals.

3 General Polynomial Neural Network Approach

In current literature, nonlinear ICA methods have mostly combined with different types of neural networks. Pajunen *et al* [9] provided one of the earliest nonlinear ICA solutions by using the Self-Organizing Maps (SOM). Since the theoretical foundation of the SOM algorithm is based on rectangular map, the main limitation of SOM lies in the inevitable distortion when the source signals differ considerably from the uniform distribution. To overcome the disadvantages associated with SOM, Pajunen and Karhunen [10] propose the generative topographic mapping (GTM) approach for nonlinear ICA. However, in order to apply non-uniformly distributed source signals, the GTM method requires the known probability density function (pdf) of the source signals, which may limit the applications of this method. Signal transformation methods based on Radial-Basis Function (RBF) [11] and Multilayer Perceptron (MLP) [5] neural networks have recently drawn a substantial amount of attention for their flexible nonlinear capability. Under the nonlinear condition, both methods provide acceptable performance. RBF-based system can provide fast convergence at the cost of less accuracy whereas MLP can recover the original signals more precisely but suffer from high computational complexity. Besides the structure of the network, the performance of the demixer also depends on the selection of the nonlinear activation function in the hidden neurons. Besides the structure of the network, the performance of the demixer also depends on the selection of the nonlinear activation function in the hidden neurons. Demixers using SOM [9], GTM [10], RBF [11] and MLP with sigmoidal nonlinearity [5] are intrinsically nonlinear because of the utilization of fixed nonlinearities in the hidden neurons. However, the execution by using the fixed degree of nonlinearity will lead to the oversized network, which inevitably subjects to huge computational complexity. Also, in [12], it is argued that the generation of arbitrary independent components is accentuated especially when an oversized network is used which subsequently leads to 'overfitting'. Hence, in order to regulate the outputs of the demixer into one unique solution, one approach is to allow the network to control its inherent capability and prevent the demixer from 'overfitting'. Therefore, instead of using a fixed form of nonlinearity in the hidden neurons, we propose to design a network whereby its intrinsic nonlinearity can be flexibly controlled. In this paper, based on the Weierstrass Approximation Theorem, polynomials are performed as the activation function in the hidden neurons. The coefficients in the polynomials are adaptively adjusted by the corresponding parameter learning algorithm to control the nonlinearities of the hidden layers in the network.

3.1 Polynomial Neuron-Based Network for Nonlinear ICA

In the Weierstrass Approximation Theorem [13], it is pointed out that for every continuous function $\Omega:[u,v]\to\mathbb{R}$, there always exists a polynomial series $p(u)=\sum_{m=0}^{M}\alpha_m u^m$, parameterized by $\boldsymbol{\theta}=\left\{M,\ \{\alpha_m\}_{m=0}^{M}\right\}$, which can uniformly approximate Ω with arbitrary accuracy. The Weierstrass Approximation Theorem plays an important role in the proposed approach since it guarantees the existence of the effective polynomial approximation and provides the theoretical foundation of the polynomial neural network. Therefore, according to the structure of the multi-nonlinearity constrained demixing system as in (5), a feedforward polynomial neuron-based network is proposed as shown in Fig. 1 where the hidden layer neurons in the network perform the polynomial series to approximate the mixing mapping functions $\{f_i\}_{i=1}^{N}$ and $\{f_i^{-1}\}_{i=1}^{N}$. Accordingly, in vector notation, the outputs of the demixer assume the following form of

$$\begin{aligned}
\mathbf{y}_{[3]} &= \mathbf{F}\otimes\mathbf{y}_{[2]} = \sum_{m=0}^{M_1}\left(\mathbf{a}_m \circ \mathbf{y}_{[2]}^{m}\right)\\
\mathbf{y}_{[2]} &= \mathbf{W}\mathbf{y}_{[1]}\\
\mathbf{y}_{[1]} &= \mathbf{F}^{-1}\otimes\mathbf{x} = \sum_{n=1}^{M_2}\left(\mathbf{b}_n \circ \left(\mathbf{x}-\mathbf{a}_0\right)^n\right)
\end{aligned} \tag{6}$$

where

$$\mathbf{y}_{[i]} = \left[y_{[i,1]} \ \cdots \ y_{[i,N]}\right]^T,$$

$$\mathbf{a}_m = \left[a_{[m,1]} \ \cdots \ a_{[m,N]}\right]^T,$$

$$\mathbf{b}_n = \left[b_{[n,1]} \ \cdots \ b_{[n,N]}\right]^T,$$

and $y_{[j,i]}$ denotes the i^{th} output of the j^{th} layer in the demixer, $\left\{a_{[m,i]} \,\middle|\, m\in[0,\ M_1];\ i\in[1,\ N];\ m,\ i\in\mathbb{Z}\right\}$ and $\left\{b_{[n,i]} \,\middle|\, n\in[1,\ M_2];\ i\in[1,\ N];\ m,\ i\in\mathbb{Z}\right\}$ are the coefficients while M_1 and M_2 represent the order of the series expansion; ' $\circ$ ' denotes the Hadamard product.

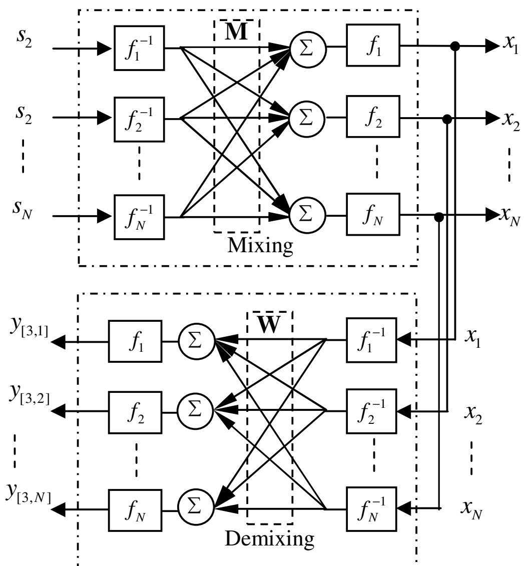

Fig. 1: Multi-nonlinearity Constrained Mixing Model and Neuron-based Nonlinear ICA Demixer

3.2 Series Reversion

As shown in Fig. 1, the implementation of the proposed demixer requires the inverse function of the polynomial series. It is possible to express the inverse function of a polynomial as a closed form when the order of the forward function is 3 or less; however, the problem becomes difficult and intractable as soon as the order increases. The theory of the Series Reversion provides an alternative solution and further establishes the foundation for computing the inverse function of a general polynomial expansion.

According to the theory of Series Reversion [14], if the function g has a polynomial expression as $g(u)=\sum_{m=0}^{M_1}\alpha_m u^m$, then the inverse function can be given by the similar form of $g^{-1}(u)=\sum_{n=1}^{+\infty}\beta_n\left(u-\alpha_0\right)^n$ with the coefficients computed by

$$\beta_n = \sum_{k_2,k_3,\ldots}\left[(-1)^{\sum_{i=2}^{M_1}k_i}\frac{\left(n-1+\sum_{i=2}^{M_1}k_i\right)!}{n!\prod_{i=2}^{M_1}\left(k_i!\right)}\left(\prod_{i=1}^{M_1}\alpha_i^{k_i}\right)\right] \quad (7)$$

where $k_2+2k_3+3k_4+\cdots=n-1,\ k_i\geq 0,\ i=2,3,4\ldots$ and $k_1=-\left(n+\sum_{i=2}^{M_1}k_i\right)$. In addition, the differential of β_n with respect to α_m's can be derived as

$$d\beta_n=\sum_{m=1}^{M_1}\left(\sum_{k_2,k_3,\ldots}(-1)^{\sum_{i=2}^{M_1}k_i}\frac{\left(-k_1-1\right)!}{n!\prod_{i=2}^{M_1}\left(k_i!\right)}\left(\prod_{\substack{i=1\\ i\neq m}}^{M_1}\alpha_i^{k_i}\right)k_m\alpha_m^{k_m-1}\right)d\alpha_m \quad (8)$$

Hence, the derivative of the reverse series with respect to the coefficients in the forward polynomial $\frac{\partial g^{-1}}{\partial\alpha_m}$ can be easily obtained by using $\frac{\partial g^{-1}}{\partial\alpha_m}=\sum_{n=1}^{M_2}\left(\frac{\partial g^{-1}}{\partial\beta_n}\frac{\partial\beta_n}{\partial\alpha_m}\right)$ which is necessary for the derivation of the parameter learning algorithm.

3.3 Parameter Learning Algorithm

In nonlinear ICA, the goal of the demixer is to obtain a set of signals as independent as possible and as close as possible to the original sources. The cost function rooted in the Kullback-Leibler Divergence (KLD) [1-2] is commonly used in most blind signal separation problem. However, under nonlinear condition, the independence preservation is not strong enough for ensuring signal separability and inadvertently results in non-uniqueness of solutions. Therefore, to reduce the indeterminacy of non-unique solutions, the cost function is modified by augmenting a set of signal constraints to the original KLD cost function as follows:

$$J=-\log\left|\det\frac{d\mathbf{y}_{[3]}}{d\mathbf{x}^T}\right|-\sum_{i=1}^{N}\log\left(p_i(y_{[3,i]})\right)+\underbrace{\sum_{i=1}^{N}\varepsilon_i f_i^{(c)}(y_{[3,i]},s_i)}_{\text{constraints}} \quad (9)$$

where $f_i^{(c)}(y_{[3,i]},s_i)=\sum_{j=1}^{D}\left[cum\left(y_{[3,i]},j\right)-cum\left(s_i,j\right)\right]^2$, ε_i's are a set of constants to control the importance of the additional constraints, $cum(u,j)$ represents the j^{th} order cumulant of u and D is the maximum order of the cumulant. In fact, these constraints imply the use of *a priori* information about the source distributions which is intended to match the outputs of the demixer to the original source signals in terms of cumulants. Making use of the relationship between the forward and reverse polynomial series as in (7)-(8), only the weight $\mathbf{W}$ and the coefficients $\left\{a_{[m,i]}\mid m\in[0,\ M_1];\ i\in[1,\ N];\ m,\ i\in\mathbb{Z}\right\}$ are the parameters which need to be updated to optimize the cost function.

Based on the architecture of the proposed polynomial neural network expressed as in (6), the derivative of the cost function with respect to the parameters can therefore be derived as

$$\begin{aligned}\frac{\partial J}{\partial\mathbf{W}}\mathbf{W}^T&=\mathrm{I}-\left(\sum_{m=1}^{M_1}m\left(m-1\right)\mathrm{diag}\left(\mathbf{a}_m\circ\mathbf{y}_{[2]}^{m-2}\right)\right)\eta_1\mathbf{y}_{[1]}^T\\&\quad-\left(\sum_{m=1}^{M_1}m\,\mathrm{diag}\left(\mathbf{a}_m\circ\mathbf{y}_{[2]}^{m-1}\right)\right)\mathbf{W}^T\rho\mathbf{y}_{[1]}^T\end{aligned} \quad (10)$$

$$\begin{aligned}\frac{\partial J}{\partial\mathbf{a}_0}&=\left[\mathrm{I}-\mathrm{diag}^{-1}\left(\eta_1\right)\mathbf{W}^T\mathrm{diag}^{-1}\left(\eta_2\right)\right]\rho+\mathrm{diag}^{-1}\left(\eta_2\right)\mathbf{W}^T\sigma\eta_1\\&\quad-\left(\sum_{n=1}^{M_2}n(n-1)\mathrm{diag}\left(\mathbf{b}_n\circ\left(\mathbf{x}-\mathbf{a}_0\right)^{n-2}\right)\right)\eta_2\end{aligned} \quad (11)$$

$$\begin{aligned}\frac{\partial J}{\partial\mathrm{diag}\left(\mathbf{a}_m\right)}&=\eta_1\left(m\mathbf{y}_{[2]}^{m-1}\right)^T+\mathrm{diag}\left[\rho\left(\mathbf{y}_{[2]}^{m}\right)^T\right]\\&\quad+\eta_2\left(\sum_{n=1}^{M_2}n\Psi_{[n,m,i]}\circ\left(\mathbf{x}-\mathbf{a}_0\right)^{n-1}\right)^T\\&\quad+\kappa\mathbf{W}^T\sigma\mathrm{diag}\left(\eta_1\right)-\mathrm{diag}^{-1}\left(\eta_1\right)\mathbf{W}^T\kappa\mathrm{diag}\left(\rho\right)\end{aligned} \quad (12)$$

where

$$\sigma=\sum_{m=1}^{M_1}m\left(m-1\right)\mathrm{diag}\left(\mathbf{a}_m\circ\mathbf{y}_{[2]}^{m-2}\right)$$

$$\kappa=\mathrm{diag}\left(\sum_{n=1}^{M_2}\Psi_{[n,m,i]}\circ\left(\mathbf{x}-\mathbf{a}_0\right)^n\right)$$

$$\eta_1=\left[-\frac{1}{\left(\sum_{m=1}^{M_1}ma_{[m,1]}y_{[2,1]}^{m-1}\right)}\ \cdots\ -\frac{1}{\left(\sum_{m=1}^{M_1}ma_{[m,N]}y_{[2,N]}^{m-1}\right)}\right]^T$$

$$\eta_2=\left[-\frac{1}{\sum_{n=1}^{M_2}nb_{[n,1]}\left(x_1-a_{[0,1]}\right)^{n-1}}\ \cdots\ -\frac{1}{\sum_{n=1}^{M_2}nb_{[n,N]}\left(x_N-a_{[0,N]}\right)^{n-1}}\right]^T$$

$$\Psi_{[n,m,i]}=\sum_{k_2,k_3,\ldots}\left[(-1)^{\sum_{i=2}^{M_1}k_i}\frac{\left(n-1+\sum_{i=2}^{M_1}k_i\right)!}{n!\prod_{i=2}^{M_1}\left(k_i!\right)}\left(\prod_{\substack{v=1\\ v\neq j}}^{M_1}a_{[v,i]}^{k_v}\right)k_m a_{[m,i]}^{k_m-1}\right]$$

$$\rho=\begin{bmatrix}-\frac{d\left[\log\left(p_1(y_{[3,1]})\right)\right]}{dy_{[3,1]}}+\varepsilon_1\frac{d\left[f_1^{(c)}\left(y_{[3,1]},s_1\right)\right]}{dy_{[3,1]}}\\ \vdots\\ -\frac{d\left[\log\left(p_N(y_{[3,N]})\right)\right]}{dy_{[3,N]}}+\varepsilon_N\frac{d\left[f_N^{(c)}\left(y_{[3,N]},s_N\right)\right]}{dy_{[3,N]}}\end{bmatrix}$$

By inserting (10)-(12) into (13)-(14), the gradient descent based learning algorithm can be obtained.

$$\mathbf{W}(t+1)=\mathbf{W}(t)-\mu_{\mathbf{W}}\frac{\partial J}{\partial \mathbf{W}}\mathbf{W}^T\mathbf{W}(t) \qquad (13)$$

$$\mathbf{a}_m(t+1)=\mathbf{a}_m(t)-\mu_{\mathbf{a}_m}\frac{dJ}{d\mathbf{a}_m};\ m=0,1,\ldots,M_1 \qquad (14)$$

4 Results

Five subgaussian signals are generated synthetically as the original sources in the experiment, expressed as

$$\mathbf{s}=\begin{bmatrix} s_1(t)\\ s_2(t)\\ s_3(t)\\ s_4(t)\\ s_5(t)\end{bmatrix}=0.95\times\begin{bmatrix}\text{Binary signal}\\ \sin(1600\pi t)\\ \sin\left[600\pi t+6\cos(120\pi t)\right]\\ \sin(180\pi t)\\ \text{Uniformly distributed signal}\end{bmatrix}$$

Simulation has been carried out iteratively with 2500 samples and the sampling frequency is 1kHz. The source signals are then mixed according to (4) where $\mathbf{M}$ is a 5×5 random mixing matrix, and

$$\begin{bmatrix} f_1(\cdot)\\ f_2(\cdot)\\ f_3(\cdot)\\ f_4(\cdot)\\ f_5(\cdot)\end{bmatrix}=\begin{bmatrix}\tanh(\cdot)\\ \sinh^{-1}(\cdot)\\ \tanh(\cdot)\\ \sinh^{-1}(\cdot)\\ \tanh(\cdot)\end{bmatrix}$$

The learning rates for the weights and the coefficients $\mathbf{a}_m$ are set to $\mu_{\mathbf{W}}=0.001$ and $\mu_{\mathbf{a}_m}=0.00003$, respectively.

In order to assess the performance of the proposed approach, we compare it with the well-known algorithms (Linear ICA, RBF and FMLP Network) by the performance index expressed as

$$\varphi=\frac{1}{NT}\sum_{t=1}^{T}\sum_{n=1}^{N}\left[\frac{s_i(t)}{\sqrt{E\left(s_i^2\right)}}-\frac{y_{[3,i]}(t)}{\sqrt{E\left(y_{[3,i]}^2\right)}}\right]^2 \qquad (15)$$

The original source signals, nonlinearly mixed signals, recovered signals by Linear ICA method [15] and the proposed polynomial-based network are shown in Fig. 2. The performance index of the tested algorithms is displayed in Fig. 3. As can be seen, the proposed approach has demonstrated its efficacy in separating signals under the nonlinear mixture. The success is consecutively followed by the MLP and RBF but the separation results achieved by the linear method falls far from optimal and this indicates the crucial need for nonlinear separation techniques.

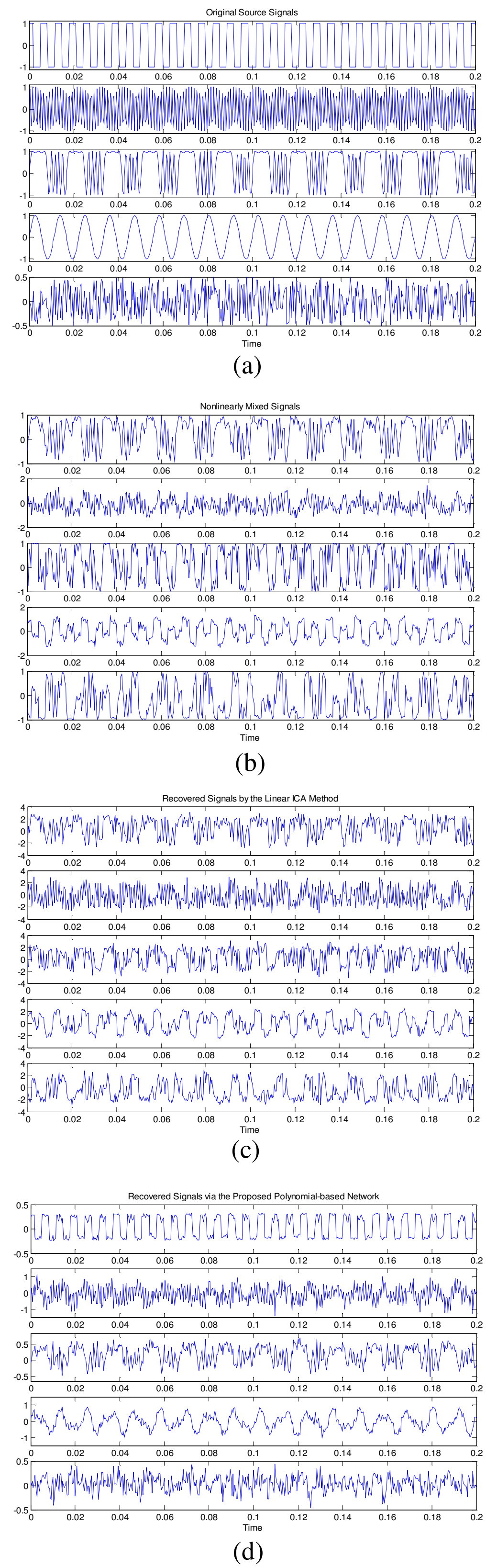

Fig. 2: (a) Original source signals.
(b) Nonlinearly mixed signals.
(c) Recovered signals by the linear ICA.
(d) Recovered signals by proposed method.

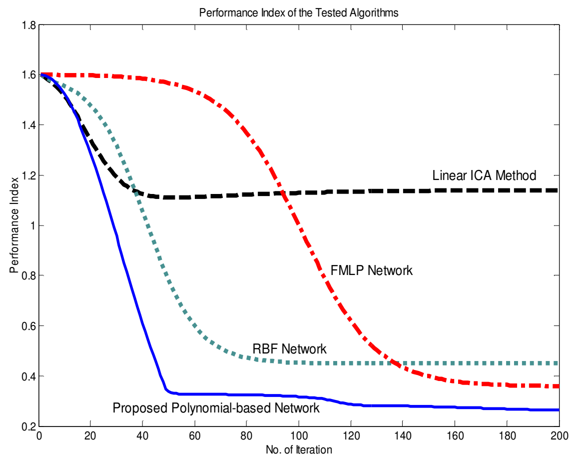

Fig. 3: Performance index of the tested algorithms

4 Conclusion

A new demixing scheme for blind separation of nonlinearly mixed signals is proposed in this paper. The key feature of the proposed approach is summarized as follows: (a) Compared with the mono-nonlinearity demixer, multiple nonlinearities are now allowed to be used for the sources since the multi-nonlinearity constrained mixing model is based. (b) A set of adaptively adjustable nonlinear functions is performed as the hidden neurons' activation function. (c) The theory of Series Reversion is integrated with neural networks to compute the inverse of the forward polynomial series.

References:

[1] S.I. Amari and A. Cichocki, Adaptive blind signal processing - Neural network approaches, *Proceedings of the IEEE*, Vol. 86, No. 10, 1998, pp. 2026-2048.

[2] J.F. Cardoso, Blind signal separation: Statistical principles, *Proceedings of the IEEE*, vol 86, No. 10, 1998, pp. 2009-2025.

[3] A. Koutras, E. Dermatas, and G. Kokkinakis, Simultaneous Speech Recognition in Noisy Reverberant Environments, *in 4th WSEAS Multi-Conference on Circuits, Systems, Communications and Computers (CSCC 2000)*, Athens, Greece, 2000.

[4] M. Kano, S. Hasebe, I. Hashimoto, and H. Ohno, Evolution of multivariate statistical process control: application of independent component analysis and external analysis, *Computers & Chemical Engineering*, vol. 28, No. 6-7, 2004, pp. 1157-1166.

[5] W.L. Woo and S. Sali, General Multilayer Perceptron Demixer Scheme for Nonlinear Blind Signal Seperation, *IEE Proc. On Vision, Image and Signal Processing*, Vol. 149, No. 5, 2002, pp. 253-262.

[6] T.F. Quatieri, D.A. Reynolds, and G..C. O'leary, Estimation of handset nonlinearity with application to speaker recognition, *IEEE Trans. on Speech and Audio Processing*, vol. 8, no. 5, 2000, pp. 567-584

[7] A. Hyvarinen, and P. Pajunen, Nonlinear independent component analysis: Existence and uniqueness results, *Neural Networks*, Vol. 12, No. 3, 1999, pp. 429-439.

[8] J. Eriksson, and V. Koivunen, Blind identifiability of class of nonlinear instantaneous ICA models, *in Proc. of the XI European Signal Processing Conference*, Toulouse, France, 2002, pp. 7-10.

[9] P. Pajunen, A. Hyvarinen, and J. Karhunen, Nonlinear Blind Source Separation by Self-Organising Maps, *Progress in Neural Information Processing 9*, 1996, pp. 1207-1210.

[10] P. Pajunen and J. Karhunen, A Maximum Likelihood Approach to Nonlinear Blind source Separation, *Artificial Neural Networks*, Lausanne, Switzerland, 1997, pp. 541-546.

[11] Y. Tan, J. Wang, and J. Zurada, Nonlinear Blind Source Separation using a Radial Basis Function Network, *IEEE Trans. on Neural Network*, Vol. 12, No. 1, 2001 pp. 124-134.

[12] W.L. Woo, and S.S. Dlay, Blind separation of nonlinearly mixed signals using maximum likelihood neural network, *WSEAS Trans. on Systems*, Vol. 2, No. 3, 2003, pp. 675-681.

[13] C.D. Cantrell, *Modern Mathematical Methods for Physicists and Engineers*, Cambridge University Press, 2000.

[14] P.M. Morse, and H. Feshbach, *Methods of Theoretical Physics*, McGraw-Hill Book Company, Inc., 1953.

[15] T.W. Lee, M. Girolami, and T.J. Sejnowski, Independent Component Analysis Using an Extended Infomax Algorithm for Mixed Subgaussian and Supergaussian Sources, *Neural Computation*, Vol. 11, 1999, pp. 417-441.