NONLINEAR BLIND SOURCE SEPARATION USING A
HYBRID RBF-FMLP NETWORK

W.L. Woo and S.S. Dlay
School of Electrical, Electronic and Computer Engineering
University of Newcastle upon Tyne
United Kingdom
Tel: (+44) 191 222 6682
Fax: (+44) 191 222 8180
Email: w.l.woo@ncl.ac.uk and s.s.dlay@ncl.ac.uk

Index Terms — Independent component analysis, nonlinear source separation, signal reconstruction, multilayer perceptron (MLP), radial basis function (RBF) network.

Abstract — A novel scheme for blind source separation of nonlinearily mixed signals is developed using a hybrid system based on radial basis function (RBF) and feedforward multilayer perceptron (FMLP) networks. In this paper, the development of proposed RBF-FMLP network is discussed which hinges on the theory of nonlinear regularisation. The proposed network uses the local and global mapping bases simultaneously to perform both signals separation and reconstruction of continuous signals as well as signals that exhibit high degree of fluctuation. The parameters of the proposed system are jointly estimated using the generalised gradient descent approach and thereby renders the training process relatively simple and efficient in computation. Simulations of both synthetic and speech signals have been undertaken to verify the efficacy of the proposed scheme in terms of speed, accuracy and robustness against noise.
1. INTRODUCTION

For almost a decade, blind source separation (BSS) using Independent Component Analysis (ICA) has received considerable amount of attention because of its simplicity and versatility in many signal processing applications [1-2]. The goal of ICA is to recover independent sources given only sensor observations that are unknown linear superposition of the unobserved independent source signals. However, in general and for many practical problems, the mixed signals are more likely to be nonlinear or subject to some kind of nonlinear distortions due to sensory or environmental limitations which can be empirically modelled as

\[ x(t) = f(s(t)) + n(t) = B_2(h(B_1s(t)) + \xi) + n(t) \]  

(1)

where \( s(t) = [s_1(t) \ s_2(t) \ \ldots \ s_N(t)]^T \), \( x(t) = [x_1(t) \ x_2(t) \ \ldots \ x_N(t)]^T \) and \( n(t) = [n_1(t) \ n_2(t) \ \ldots \ n_N(t)]^T \) are the original source signals, spatially observed signals and sensory noise, respectively, \( \{B_1, B_2\} \) are the \( N \times N \) mixing matrices, \( \xi \) is the \( N \times 1 \) bias vector and \( h(\cdot) = [h_1(\cdot) \ h_2(\cdot) \ \ldots \ h_N(\cdot)]^T \) is a set of the nonlinear functions characterising the amount of nonlinearity in the mixing model. As linear BSS algorithms are not applicable in this model, the search for nonlinear solutions to the problem becomes paramount [3]. Therefore the need to study signal separation for nonlinear mixtures is of great importance at both theoretical and practical levels.

Extension of existing theories and methods to nonlinear BSS is not straightforward and so far, there have only been few initial attempts in this field [3-19]. However, the reported performances have been disappointing due to problem uniqueness and/or computational complexities. Kernel ICA proposed by Harmeling et al [4] which is a nonlinear extension of standard PCA uses a variant of canonical correlation analysis to separate nonlinearly mixed signals but the method is rather heuristic and the selection of nonlinear transformations to higher dimensional space is limited. Parra [5] uses the idea of volume conserving symplectic transformations which works well only for weakly nonlinear mixed signals. The self-organising
map (SOM) has been used in [6-7] but suffers from both network complexity and interpolation errors for continuous phase signals. It is also known that SOM is only applicable to a certain class of nonlinear functions in the mixture. Tan et al [8] proposed a radial basis function (RBF) network in which the hidden layer constitutes a set of Gaussian basis but the performance is limited to smooth nonlinear functions and continuous source signals. Higher order statistics combined with Genetic algorithm has also been proposed in [9] at the expense of immensely heavy computational complexity. Nested form of neural network models based on nonlinear ICA algorithms developed by Burel [10] which were later modified by Taleb and Jutten [11], Valpola et al [12-13] and Yang et al [14] are more structured and reported to produce better results than SOM models. These results were further enhanced through generalising the demixer architecture to a Feedforward Multilayer Perceptron (FMLP) employing the class of zero preserving continuously differentiable nonlinear function with variable gradient [17]. For a comprehensive survey on current and previous works in nonlinear BSS, readers are referred to [18].

The organisation of the paper is as follows: Section 2 examines the core problems facing current nonlinear ICA techniques and discusses the motivations behind the proposed work. The development of the proposed contrast function and the derivation of the update algorithm are detailed in Section 3 while Section 4 presents two experimental studies and compares the proposed algorithm with existing methods. Finally, Section 5 summarises the proposed work.

2. MOTIVATIONS

Nonlinear blind signal separation problem is fundamentally an ill-posed inverse problem in two aspects. Firstly, there exists infinite number of invertible mappings that can transform a set of independent source signals into another set of independent random variables. However, these random variables are not the original source signals but are nonlinearly related to the source signals; hence the uniqueness criterion is violated. Secondly, the unavoidable presence of additive noise or some form of imprecision in the observed
signals adds uncertainty to the signal reconstruction. In particular, if the noise level is too high, it is possible for the continuity criterion to be violated. A natural way of regularising the solution consists in looking for separating mappings belonging to a specific subspace \( \Xi(\Omega) \) parameterised by \( \Omega \). To characterise the indeterminacies for this specific model \( \Xi(\Omega) \) we need to examine the independence preservation equation which states that for all \( A \) within \( C_N \) where \( C_N \) is a \( \sigma \)-algebra on \( \mathbb{R}^N \), there exists

\[
\int_A dp_1(s_1)dp_2(s_2)\cdots dp_N(s_N) = \int_{H(A)} dp_1(y_1)dp_2(y_2)\cdots dp_N(y_N) \tag{2}
\]

Denoting \( T \) as the set of transforms that preserve independence and \( \Psi \) as the set

\[
\Psi = \{(p_1(s_1), p_2(s_2), \ldots, p_N(s_N)) \mid \exists H \in \Xi \setminus (T \cap \Xi), H(s) \text{ is independent}\} \tag{3}
\]

of all source signal distributions \( p(s) = \prod_{i=1}^N p_i(s_i) \) for which there exists a non-trivial mapping \( H : s \rightarrow y \) belonging to the model \( \Xi \) and preserving the independence of the components of the source signals \( s \). Ideally, \( \Psi \) should be empty but this cannot be achieved. According to Darmois theorem [18,19], in the case of linear model, \( \Psi \) will contain signal distributions with more than one gaussian signal and hence signal separation is not possible. Signal separation is possible only when \( p(s) \in \Psi \) where \( \Psi \) is the complement of \( \Psi \). The separability of the signals occurs when \( H \in T \cap \Xi \) is an identity matrix (which may also include the product of a permutation matrix and a diagonal matrix). The indeterminacy is then given by \( T \cap \Xi \) which represents the order, sign and scale ambiguities. In the general case where the mapping \( H \) has no particular form, which usually occurs in nonlinear model, independence preservation is a weak constraint for ensuring signal separability. Note that \( T \) is not limited to the set of linear transforms in order to preserve independence and in general, \( T \) does not depend on \( A \). By using the Gram-Schmidt orthogonalisation procedure in linear algebra space, it is possible to show that there exists infinite number of non-trivial
transformations $H$ which still ‘mix’ the source signals while preserving the statistical independence of transformed outputs [18]. Hence signal separation of general nonlinear mixture is impossible by resorting only to statistical independence. To overcome the ill-posed nature of the problem, some form of regularisations would need to be imposed. Ideally, the criterion are to design a demixer to output independent components that resemble the original source signals and the difference between these components and the source signals is smooth. One approach is to impose some form of signal constraints as regularisers to control the waveform of the estimated signals [8]. Another approach is to reduce the number of non-trivial mappings by imposing some form of structural constraints on the demixing model such that the global mixing-demixing system has certain desirable input-output mappings [20]. In this paper, both the signal and structural constraints are jointly used to regularise the effects of the indeterminacy resulted from the independence preservation rule. In terms of set theory, our aim is to reduce the cardinality in the set $\Xi \setminus (T \cap \Xi)$ that results in the transformed outputs $H(s)$ to be statistically independent where $H \in \Xi \setminus (T \cap \Xi)$. This is achieved by overlapping another set $Z$ that contains the signal constraints on the existing set $\Xi \setminus (T \cap \Xi)$ such that the cardinality in the new set $(\Xi \setminus (T \cap \Xi)) \cap Z$ is reduced. This will be detailed in Section 3 where signal constraints (as regularisers) in the form of cumulants matching are imposed on the estimated source signals. To further reduce the cardinality in $(\Xi \setminus (T \cap \Xi)) \cap Z$, the set $\Xi$ is confined to an even smaller set such that $\Xi$ contains only the smooth input-output mappings that will preserve the independence and the continuity in the reconstructed signals.

The theory of regularisation on smooth input-output mapping has led to the development of radial basis function (RBF) demixing system [8] in which the hidden nodes are constructed by local mapping basis e.g. gaussian activation functions. Although the theory of RBF is mathematically elegant, from our investigated studies, it is found that the RBF performance degrades dramatically when the source signals are characterised with frequent discontinuities. The local mapping basis used by the RBF can approximate any smooth function; however, it misses those points where discontinuities are present. Moreover, the number of hidden
nodes required to achieve a considerable level of performance is high and this subsequently leads greater computational complexity and stringent time requirement for training the network parameters. Besides, the utilisation of a large number of hidden nodes very often undermines the convergence of the blind RBF demixer especially when the nodes are not properly located in the input space as demonstrated in this paper. This also has a strong bearing on how the parameters of the network are being initialised. In [17], a generalised scheme based on FMLP is developed which uses global features to approximate the input-output mappings of the inverse of the nonlinear mixture. For the same degree of accuracy, the FMLP generally requires smaller number of parameters than the RBF. However, its weaknesses stem from the use of a black box fitting approach which causes the FMLP to suffer from the ‘overfitting’ phenomenon and the global mapping problem in which the hidden neurons in the network tend to interact globally with each other thus resulting in an improvement at one point but deterioration at some other points. In light of above, it is evident that the weaknesses associated with the RBF can be complemented by the strength of the FMLP and vice versa. A hybrid that merges RBF and FMLP networks into a single operating system is therefore proposed. The proposed RBF-FMLP uses local and global mapping bases to approximate the input-output mappings so that the hybrid system is endowed with both smoothness and squashing characteristics, which are the vital ingredients in nonlinear signal separation i.e. both local and global properties of each individual universal approximator can be extracted to improve performance in constructing continuous signals as well as signals that exhibit high degree of fluctuation. Other approaches of combining local and global bases for nonlinear BSS have also been reported where a set of adaptive B-splines [16] or higher order polynomials [20] are used as the hidden neurons activation functions. The use of polynomials, however, usually incurs a high computational cost which needs to be taken into account during real-time implementation.

3. DEVELOPMENT OF BLIND RBF-FMLP DEMIXING SYSTEM

The architecture of the proposed hybrid RBF-FMLP nonlinear demixer is shown in Figure 1 in which the input-output relationship can be expressed as
\[ y_i = \sum_{j=1}^{N} w_{kj}^{(2)} g_j \left( \sum_{k=1}^{N} m_j w_{jk}^{(1)} \exp \left( -\frac{1}{2\sigma_k^2} \|x - c_k\|^2 \right) + \theta_j \right) \quad \forall i = 1, 2, \ldots, N \]  

(4)

where \( \{w_{kj}^{(1)}\}_{j,k=1}^N \), \( \{w_{kj}^{(2)}\}_{i,j=1}^N \), \( \{m_j\}_{j=1}^N \) and \( \{\theta_j\}_{j=1}^N \) are the RBF weights, FMLP weights, gradients and biases, respectively while \( \{g_j(\cdot)\}_{j=1}^N \), \( \{c_k\}_{k=1}^N \) and \( \{\sigma_k^2\}_{k=1}^N \) are the nonlinear hidden neurons activation function, location centres of the neurons and the width of the activation function. In this setting, \( \{w_{kj}^{(1)}\}_{j,k=1}^N \), \( \{w_{kj}^{(2)}\}_{i,j=1}^N \), \( \{\theta_j\}_{j=1}^N \), \( \{g_j(\cdot)\}_{j=1}^N \), \( \{c_k\}_{k=1}^N \) and \( \{\sigma_k^2\}_{k=1}^N \) are the parameters that will be optimised according to the following contrast function:

\[
J(\Omega) = D \left( p_Y(y) \left\| \prod_j \hat{q}_j(y_j) \right\| \right) + \sum_i \alpha_i f_i(y_i, s_i) \quad \text{constraint}
\]

\[
= \int_{y \in \mathbb{R}^N} p_Y(y) \log \frac{p_Y(y)}{\prod \hat{q}_j(y_j)} \, dy + \sum_i \alpha_i f_i(y_i, s_i)
\]

\[
= \mathbb{E} \left[ \log \frac{p_Y(y)}{\prod \hat{q}_j(y_j)} \right] + \sum_i \alpha_i f_i(y_i, s_i)
\]

(5)

where

\[
f_i(y_i, s_i) = \sum_{j=1}^{M} \left\{ \text{cum} \left[ y_{i_1}, \ldots, y_{i_j} \right] - \text{cum} \left[ s_{i_1}, \ldots, s_{i_j} \right] \right\}^2
\]

(6)

In above, \( D(\cdot) \) is the mutual information of the demixer outputs [1-2] where \( p_Y(y) = p(y_1, y_2, \ldots, y_N) \) and \( \{\hat{q}_j(y_j)\}_{i=1}^N \) are the joint and marginal probability density functions (pdf) respectively, and \( \Omega = \{W_2, W_1, M, \theta, \sigma^2_i, c_i\} \) is the vector-matrix representation of the parameters. The marginal pdfs can be estimated in many ways e.g. parametric-based approach using probability series expansion. Readers
are referred to [1, 2] for further details. The term \( f_i(y_i, s_i) \) is the constraint (or regulariser) constructed from the \textit{a priori} information about the input signals and \( \alpha_i \) is the scalar constant chosen to provide the required amount of weighting on the constraints. The symbol ‘\text{cum}[\cdot]’ denotes the cumulants function and the order is defined by \( j \) (See Appendix B). The mutual information is invariant under any invertible nonlinear function and as a consequence, it has undesirable effects in that the outputs of the demixer will be related to the input signals via an indeterminate nonlinear mapping. The core idea is therefore to find \( \Omega \) such that (5) is minimised (to ensure statistical independence) while simultaneously ameliorating the effects of indeterminacy due to the trivial nonlinear mapping embedded in the combined mixing-demixing system. The latter being analogous to signal reconstruction [20] can be achieved by using the set of constraints which forces the demixer outputs to have identical cumulants up to the \( M^{th} \) order with respect to the original signals. It is not necessary that the signal constraints must be in the form of cumulants. Alternative measure can be based on moments matching. The motivation of using cumulants lies in its property of Gaussian noise suppression in the cumulant domain. In a compact expression, the outputs assume the form of

\[
y = W_2g(u + \theta) \quad \text{with} \quad u = MW_1v, \quad v = [v_1, v_2, \cdots, v_N]^T \quad \text{and} \quad v_i = \exp\left(-\frac{1}{2\sigma_i^2}\|x - c_i\|^2\right).
\]

In addition, the contrast function in (5) may be expressed up to constant proportionality as

\[
J(\Omega) = -E\left[\log\det\left|\frac{dy}{dx^T}\right|\right] - h(x) - E\left[\log\prod_i \tilde{q}_i(y_i)\right] + \sum_i \alpha_i f_i(y_i, s_i)
\]

\[
\approx -E\left[\log\det\left|\frac{dy}{dx^T}\right|\right] - \sum_i E[\log \tilde{q}_i(y_i)] + \sum_i \alpha_i f_i(y_i, s_i) \tag{7}
\]

\[
\approx -\log\det\left|\frac{dy}{dx^T}\right| - \sum_i \log \tilde{q}_i(y_i) + \sum_i \alpha_i f_i(y_i, s_i)
\]

where a instantaneous sample estimator is used in the last line of (7). The proportionality sign in the second line of (7) is used because \( h(x) = -E[\log p_X(x)] \) is independent of the network parameters and therefore can be treated as a constant. The derivatives of the outputs with respect to the inputs are derived as follow:
\[
\frac{dy}{dx^T} = \frac{dy}{du^T} \frac{du}{dv^T} \frac{dv}{dx^T} = -W_2 \ D_g \ M \ W_1 \ D_v \ D_\sigma \ A
\]

(8)

where \( D_v = \text{diag}[v_1, v_2, \ldots, v_N] \), \( M = \text{diag}[m_1, m_2, \ldots, m_N] \), \( D_g = \text{diag}[\hat{g}_1, \hat{g}_2, \ldots, \hat{g}_N] \) with

\[
\hat{g}_i = \frac{dg_i(u_i + \theta_i)}{d(u_i + \theta_i)} \quad D_\sigma = \text{diag} \left[ \frac{1}{\sigma_1^2}, \frac{1}{\sigma_2^2}, \ldots, \frac{1}{\sigma_N^2} \right] \quad \text{and} \quad A = \begin{bmatrix} (x - c_1^T) \\ \vdots \\ (x - c_N^T) \end{bmatrix}
\]

. Substituting (8) into (7), this results in

\[
J(\Omega) = -\log \det W_2 - \log \det W_1 - \log \det M - \sum_i \log \hat{g}_i(u_i + \theta_i) + \frac{1}{2} \sum_i \frac{\|x - c_i\|^2}{\sigma_i^2}
\]

\[
+ \sum_i \log \sigma_i^2 - \sum_i \left( \log \hat{g}_i(y_i) + \alpha_i f_i(y_i, s_i) \right) - \log \det A
\]

(9)

The RBF-FMLP network is a function of the weights \( W_1, W_2, \) the gradient \( M, \) the derivatives of the FMLP nonlinearity \( \{\hat{g}_i\}, \) the location of the centre \( \{c_i\} \) and the width of the neurons \( \{\sigma_i^2\}. \) Denoting the change in (9) as \( dJ(\Omega) = J(\Omega + d\Omega) - J(\Omega) \) due to infinitesimal change in the network parameters \( d\Omega, \) we arrive at:

\[
dJ(\Omega) = -tr[dW_2W_2^{-1}] - tr[dW_1W_1^{-1}] - tr[dMM^{-1}] - \sum_i d\left( \log \hat{g}_i(u_i + \theta_i) \right) + \frac{1}{2} \sum_i d \left( \frac{\|x - c_i\|^2}{\sigma_i^2} \right)
\]

\[
+ \sum_i d \left( \log \sigma_i^2 \right) - \sum_i d \left( \log q_i(y_i) + \alpha_i f_i(y_i, s_i) \right) - d \log |A|
\]

\[
= -tr[d\tilde{z}_2] - tr[d\tilde{z}_1] - tr[d\psi] + \lambda^T [d\alpha + d\theta] + \frac{1}{2} \sum_i d \left( \frac{\|x - c_i\|^2}{\sigma_i^2} \right) + \sum_i d \left( \log \sigma_i^2 \right)
\]

\[
+ \left[ \varphi + \alpha \otimes \mathbf{f} \right]^T dy - d \log |A|
\]

(10)

with \( d\tilde{z}_2 = dW_2W_2^{-1}, \quad d\tilde{z}_1 = dW_1W_1^{-1}, \quad d\psi = dMM^{-1}, \quad \alpha = [\alpha_1, \alpha_2, \ldots, \alpha_N]^T, \quad \mathbf{f} = [\hat{f}_1, \hat{f}_2, \ldots, \hat{f}_N]^T, \)

\[
\lambda = \begin{bmatrix} \hat{g}_1 \\ \hat{g}_2 \\ \vdots \\ \hat{g}_N \end{bmatrix}, \quad \varphi = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_N \end{bmatrix}, \quad \hat{f}_i = \frac{df_i}{dy_i} \quad \text{and} \quad \otimes \text{ represents the Hadamard product.}
The space of the contrast function in (5) with respect to the matrix parameter \( W_i \) has been proven to be curvilinear and thus it is characterised by the Riemannian geometry [2,14]. In (10), \( d\xi_i \) defines a non-integrable differential which spans a local linearised tangent space in the Riemannian curvilinear coordinates system. Hence, it is more appropriate to use the differential \( d\xi_i \) to locate the gradient of the contrast function. The term non-integrable means that we cannot find any matrix function \( \xi_i = \xi_i(W_i) \) that satisfies
\[
d\xi_i = dW_i W_i^{-1}
\] and being so, we need to re-define the differential \( d\xi_i \) in the space of \( dW_i \) so as to update matrix parameter \( W_i \) via \( d\xi_i = dW_i W_i^{-1} \) [23]. This approach follows closely with the concept of natural gradient used in linear ICA except that the present paper deals with a nonlinear demixing system based on the hybrid RBF-FMLP. The Riemannian geometry is not used for \( A \) since the latter is only used for representation convenience for the distance between an input vector \( x \) and the centres \( \{c_k\}_{k=1}^N \), and therefore it does not characterise the basis vectors for centres.

### 3.1 Parameters Update.

Equipped with the differentials, the derivations of the learning algorithms have now been substantially simplified. Starting with the output layer of the demixer, we have (see Appendix for derivation):

\[
\frac{dJ}{d\xi_2} = -I + [\phi + \alpha \circ f]y^T
\]  

(11)

Using the fact that \( d\xi_i = dW_i W_i^{-1} \) and therefore \( \frac{dW_i(t)}{dt} = \frac{d\xi_i(t)}{dt} W_i(t) = -\Lambda_i(t) \circ \frac{dJ}{d\xi_i(t)} W_i(t) \) where \( \Lambda_i \) is the matrix whose entries constitute the step size for each element of the parameters, the update equation for the FMLP weights has been derived as follow:
\[ W_z(t + 1) = W_z(t) + \frac{dW_z(t)}{dt} \]
\[ = W_z(t) - \Lambda_z(t) \circ \frac{dJ}{d\xi_z(t)} W_z(t) \]
\[ = W_z(t) + \Lambda_z(t) \circ \left[ I - \left[ \varphi(t) + \alpha \circ \hat{f}(t) \right] y^T(t) \right] W_z(t) \]
\[ = W_z(t) + \Lambda_z(t) \circ \left[ I - r^{(2)}(t)y^T(t) \right] W_z(t) \]  
\[ (12) \]

where \( r^{(2)}(t) = \varphi(t) + \alpha \circ \hat{f}(t) \) is the score function associated with the output of the FMLP network.

Proceeding in similar fashion, the update equation for the RBF weights becomes

\[ W_i(t + 1) = W_i(t) - \Lambda_i(t) \circ \frac{dJ}{d\xi_i(t)} W_i(t) \]
\[ = W_i(t) + \Lambda_i(t) \circ \left[ I - M(t) \left[ \lambda(t) + D_g(t) W_2^T(t)r^{(2)}(t) \right] v^T W_i^T(t) \right] W_i(t) \]
\[ = W_i(t) + \Lambda_i(t) \circ \left[ I - M(t)r^{(1)}(t) \left[ W_i(t)v(t) \right]^T \right] W_i(t) \]  
\[ (13) \]

where \( r^{(1)}(t) = \lambda(t) + D_g(t) W_2^T(t)r^{(2)}(t) \) is the score function associated with the output of the RBF network. On the other hand, the update for the gradient matrix \( M \) is given by

\[ M(t + 1) = M(t) - \Lambda_3(t) \circ \frac{dJ}{d\psi(t)} M(t) \]
\[ = M(t) + \Lambda_3(t) \circ \left[ I - \text{diag} \left\{ \lambda(t) u^T(t) + D_g W_2^T(t)r^{(2)}(t)u^T(t) \right\} \right] M(t) \]
\[ = M(t) + \Lambda_3(t) \circ \left[ I - \text{diag} \left\{ r^{(1)}(t) u^T(t) \right\} \right] M(t) \]  
\[ (14) \]

where \( \text{diag} \{ Q \} \) denotes extraction of the diagonal values of matrix \( Q \). The reason for extraction of the diagonal matrix is due to the structure that \( M \) is itself a diagonal matrix. As for the update of the threshold, we have
Similarly, the derivatives of the contrast function with respect to the width of the neurons take the form of

\[
\sigma_i^2(t + 1) = \sigma_i^2(t) - \mathbf{A}_s(t) \circ \frac{dJ}{d\sigma_i^2(t)} = \sigma_i^2(t) - \mathbf{A}_s(t) \circ \left[ \left( \left[ \mathbf{r}^{(1)}(t) \right]^{T} \mathbf{M}(t) \mathbf{w}_i^{(i)}(t) \mathbf{v}_i(t) - 1 \right) \frac{\| \mathbf{x}(t) - \mathbf{c}_i(t) \|^2}{2(\sigma_i^2(t))^2} + \frac{1}{\sigma_i^2(t)} \right]
\]

for all \( i \) where \( \delta_i \) is the delta function having a unit entry at the \( i^{th} \) position. The location of the hidden neurons \( \{ \mathbf{c}_i \}_{i=1}^{N} \) can be trained using the enhanced k-means clustering algorithm [21]. However, the overall convergence speed of the demixer and its efficiency in signal separation are highly dependent on the success of the RBF hidden neurons in capturing the underlying mixture distributions. Moreover, the clustering procedure can be an arduous task and is extremely slow especially in high dimensional space. To overcome these limitations, the centre location of the neurons can be jointly optimised with other parameters in which we have derived it to be

\[
\mathbf{c}_i(t + 1) = \mathbf{c}_i(t) - \mathbf{A}_s(t) \circ \frac{dJ}{d\mathbf{c}_i(t)} = \mathbf{c}_i(t) - \mathbf{A}_s(t) \circ \left[ \left( \left[ \mathbf{r}^{(1)}(t) \right]^{T} \mathbf{W}_i(t) \mathbf{\delta}_i(t) - 1 \right) \frac{(\mathbf{x}(t) - \mathbf{c}_i(t))}{\sigma_i^2(t)} + \mathbf{A}^{-1}(t) \mathbf{\delta}_i \right]
\]
However, from (17) the stability of the centre location update is directly conditional upon the inversion of matrix \( A(t) \), which is a function of \( \{x - c_i(t)\}_{i=1}^N \). To avoid inverting an ill-conditioned matrix, the computation for \( A^{-1} \) can be carried out by perturbing the matrix according to \( A^T \left[ AA^T + \epsilon I \right]^{-1} \) with \( \epsilon << 1 \).

Since \( AA^T \) is positive semi-definite, the perturbation will ensure that the inverse matrix is always stable.

Although the use of centres update algorithm based on (17) will accelerate the convergence speed, its intrinsic matrix inversion requires heavy computation. To reduce the complexity of the centres update algorithm, (17) can be jointly used with the enhanced k-means clustering algorithm where the initial phase of the update algorithm will be implemented using (17) but as soon as convergence to steady state is apparent, the update algorithm can switch to the enhanced k-means clustering algorithm for fine tuning and tracking.

The enhanced k-means algorithm does not interfere with the signal constraints since the latter will be dealt with by the FMLP sub-network.

### 3.2 Step Size Update.

Similar to parameters update in Section 3.1, the matrix of step sizes \( \{\Lambda_i(t)\}_{i=1}^6 \) can also be adapted iteratively to optimise the speed of convergence. This can be achieved by using the adaptive step size method which adapts each step size continuously during the learning phase. In this work, the adaptation of the step sizes follows closely with [22] which is outlined below:

\[
\Delta \eta_{jk}^{(i)}(t + 1) = \begin{cases} 
  b & \text{if } V_{jk}^{(i)}(t) - G_{jk}^{(i)}(t) > 0 \\
  -c \eta_{jk}^{(i)}(t) & \text{if } V_{jk}^{(i)}(t) - G_{jk}^{(i)}(t) < 0 \\
  0 & \text{otherwise}
\end{cases}
\]  

(18)
where $b$ and $c$ are two constants, $\Delta \eta_{jk}^{(i)}(t+1) = \eta_{jk}^{(i)}(t+1) - \eta_{jk}^{(i)}(t)$ and $\eta_{jk}^{(i)}(t)$ represents the $(j,k)^{th}$ element of $\Lambda_i(t)$. Note that $\{\Lambda_i(t)\}_{i=1}^3$ exist in the form of matrices whereas both $\Lambda_4(t)$ and $\Lambda_6(t)$ in vector form, and $\Lambda_5(t)$ scalar. The term $V_{jk}^{(i)}$ measures the average of previous gradients and $G_{jk}^{(i)}$ is the current gradient which is given as follows:

\[
G_{jk}^{(i)}(t) = \begin{cases} 
\left[ (I - r^{(2)}(t)y^{T}(t))W_{2}(t) \right]_{jk}, & i = 1 \\
\left[ (I - M(t)r^{(3)}(t))[W_{1}(t)v(t)]^{T}W_{1}(t) \right]_{jk}, & i = 2 \\
\left[ (I - \text{diag}[r^{(1)}(t)u^{T}(t)])M(t) \right]_{jk}, & i = 3 \\
-\left[ r^{(1)}(t) \right]_{j}, & i = 4 \\
-\left( \left[ r^{(1)}(t) \right]^{T}M(t)W_{1}(t)\delta_{j}v_{j}(t) - 1 \right) \frac{\|x(t) - c_{j}(t)\|^2}{2(\sigma_{j}^2(t))^2} + \frac{1}{\sigma_{j}^2(t)}, & i = 5 \\
-\left( \left[ r^{(1)}(t) \right]^{T}W_{1}(t)\delta_{j}v_{j}(t) - 1 \right) \frac{(x(t) - c_{j}(t))}{\sigma_{j}^2(t)} + [A(t) + \epsilon I]^{-1} \delta_{j}, & i = 6 
\end{cases}
\tag{19}
\]

From (18), it is noted that when $V_{jk}^{(i)}$ and $G_{jk}^{(i)}$ have the same sign, their product will be greater than zero which refers to the case of slow convergence and therefore, the step size should be arithmetically increased at each iteration. However, if their product results in negative sign, the step size will then be geometrically reduced as this implies that the parameters are oscillating. The slow increase and fast decrease of the step sizes is designed to avoid divergence. On the other hand, their product is zero this evidently means that either a solution is reached or that the algorithm has converged to one of the local minima. At this point, it is imperative to randomly perturb the current parameters a number of times to ascertain that a global minimum is attained. To reduce computational complexity, the average of gradients can be recursively estimated as


\[ V_{jk}^{(i)}(t) = \frac{1}{t} \sum_{l=1}^{t} G_{jk}^{(i)}(l) \]

\[ = \frac{1}{t} \sum_{l=1}^{t-1} G_{jk}^{(i)}(l) + \frac{1}{t} G_{jk}^{(i)}(t) \]

\[ = (1 - \gamma)V_{jk}^{(i)}(t-1) + \gamma G_{jk}^{(i)}(t) \]  \hspace{1cm} (20)

with \( \gamma = 1/t \) where the average gradient \( V_{jk}^{(i)}(t) \) is taken over the batch learning period of \([1,t]\).

Alternatively, \( V_{jk}^{(i)}(t) \) can be computed based on the average of the past \( p \) gradients rather than all previous gradients. In this case, to ensure stability \( \gamma \) must be bounded between 0 and 1 (i.e. \( 0 \leq \gamma \leq 1 \)). If \( \gamma = 1 \) is used, then \( V_{jk}^{(i)}(t) \) depends only on the current gradient \( G_{jk}^{(i)}(t) \). At the other extreme, if \( \gamma = 0 \) is selected then \( V_{jk}^{(i)}(t) \) remains unaltered from the previous average gradient \( V_{jk}^{(i)}(t-1) \) which effectively relates to the initial value.

### 3.3 Initialisations.

Among all initialisation of parameters, the initial value of the gradient matrix \( M \) and the corresponding step size are most crucial in order to control the overall amount of nonlinearity embedded in the proposed hybrid system. If the gradients are initialised too largely, then the signals will be heavily warped by the nonlinearity of the FMLP hidden nodes. In general, we initialise \( M(0) = \beta I \) with \( \beta \ll 1 \) and \( \{ A_3(0) \}_{jk} = n_{jk}^{(3)}(0) \in [0.1,0.001] \) so that a weakly nonlinear network is initially used in order to avoid early ‘overfitting’. The constants \( \varepsilon \) used in the matrix inversion, \( b \) and \( c \) in (18) are set to 0.1, 0.01 and 0.01, respectively. These values are obtained based on a set of Monte Carlo experiments carried out to determine the best range of step sizes that trades off between speed of convergence and accuracy. The weights in the FMLP and RBF sub-networks can be initialised as \( W_2(0) = W_1(0) = I \) while the bias weights \( \theta(0) = 0 \). The
width of the RBF hidden neurons can be set according to \( \{\sigma_i^2(0)\}_{i=1}^N \in [1/2, \ 2] \) and the centres \( \{c_i(0)\}_{i=1}^N \) can be randomly drawn from a uniform distribution. The step size matrices are initialised as follows:

\[
\Lambda_2(0) = \Lambda_4(0) = \left(\|x(0)\|^2 + d\right)^{-1} I \quad \text{where } d \text{ is a non-zero constant, } \Lambda_4(0) = \Lambda_6(0) = 0.0075 I \quad \text{(where } I \text{ is a column vector whose entries are all unity) and } \Lambda_4(0) = 0.005. \text{ These initial values are task dependent and by no means optimal. Nevertheless, these values have been repeatedly performed on various experiments and the results obtained have confirmed the validity of the usage of such values. The selection of } \{\alpha_i\}_{i=1}^N, \text{ in general, needs to take into account of functional form of the score function } \phi(t), \text{ the order } M \text{ used in the signal constraints and how accurately we want to trade off between achieving statistical independence and cumulants matching. From most experiments conducted so far by using the proposed technique, cumulants matching up to order } M = 6 \text{ and the weighting factors } \{\alpha_i\}_{i=1}^N = 1 \text{ being set such that each constraint in (5) is equally weighted have been used and the performances achieved have been very encouraging. However, we emphasise that } \{\alpha_i\}_{i=1}^N \text{ and the order } M \text{ used in the signal constraints are strongly interrelated and that the determination of the optimal values for both } \{\alpha_i\}_{i=1}^N \text{ and } M \text{ remains as an open issue. The following summarises the computational steps for the proposed scheme:}

\[
\text{Initialise: } \Omega(0) = \left\{W_2(0), W_4(0), M(0), \theta(0), \left\{\sigma_i^2(0)\right\}_{i=1}^N, \left\{c_i(0)\right\}_{i=1}^N \right\}.
\]

\[
\text{For } t = 0, 1, 2, \ldots
\]

\[
\text{Compute: } \mathbf{y}(t) = W_2(t)g(M(t)W_4(t)v(t) + \theta(t)), \quad v(t) = [v_1(t), v_2(t), \ldots, v_N(t)]^T, \quad v_i(t) = \exp\left(-\frac{1}{2\sigma_i^2(t)}\|x(t) - c_i(t)\|^2\right)
\]

\[
\text{Compute } \left\{G^{(i)}_{jk}(t)\right\}_{i=1}^6 \text{ and } \left\{V^{(i)}_{jk}(t)\right\}_{i=1}^6 \text{ using (19) and (20)}
\]

\[
\mathbf{r}^{(j)}(t) = \begin{cases}
\phi(t) + \alpha \circ \mathbf{f}(t), & j = 2 \\
\lambda(t) + D_\phi(t)W_2^T(t)\mathbf{r}^{(2)}(t), & j = 1
\end{cases}
\]

\[
\text{Update the following:}
\]
\[ \mathbf{W}_2(t+1) \leftarrow \mathbf{W}_2(t) \text{ using (12)} \]

\[ \mathbf{W}_1(t+1) \leftarrow \mathbf{W}_1(t) \text{ using (13)} \]

\[ \mathbf{M}(t+1) \leftarrow \mathbf{M}(t) \text{ using (14)} \]

\[ \mathbf{\theta}(t+1) \leftarrow \mathbf{\theta}(t) \text{ using (15)} \]

\[ \sigma^2(t+1) \leftarrow \sigma^2(t) \text{ using (16)} \]

\[ \mathbf{c}_i(t+1) \leftarrow \mathbf{c}_i(t) \text{ using (17) or/and enhanced k-means clustering algorithm} \]

End

Note also that in updating the centres \( \{\mathbf{c}_i\}_{i=1}^N \), it may be helpful to alternate between using the gradient adaptation as in (17) and the enhanced k-means clustering algorithm in order to preserve the accuracy in capturing the local variations of the observed signals and to enhance the convergence speed.

4. RESULTS

Two sets of experiments will be carried out in this section. Both are based on the MATLAB programming language. The first experiment uses source signals which are generated synthetically and are sub-gaussianly distributed whereas the second experiment uses recorded speech signals which are super-gaussianly distributed. In the first experiment, the source signals are given by

\[ \mathbf{s} = [s_1(t) \ s_2(t) \ s_3(t) \ s_4(t)]^T = [\text{sgn}[\cos(160\pi t)] \ \sin(60\pi t) \ \text{ramp}(t, \tau) \ \sin(800\pi t)]^T \]

where \( \text{ramp}(t, \tau) \) defines the sawtooth function with width \( \tau = 0.16667 \). The source signals are nonlinearly mixed according to

\[ \mathbf{x}(t) = \mathbf{V} \tanh(\mathbf{W}s(t)) + \mathbf{n}(t) \]  \hspace{1cm} (21)

where \( \mathbf{V} \) and \( \mathbf{W} \) are 4×4 non-singular matrices given as
\begin{align}
V &= \begin{bmatrix}
0.7349 & 0.1556 & 0.4902 & 0.4507 \\
0.6873 & 0.1911 & 0.8159 & 0.4122 \\
0.3461 & 0.4225 & 0.4608 & 0.9016 \\
0.1660 & 0.8560 & 0.4574 & 0.0056
\end{bmatrix},
W &= \begin{bmatrix}
0.2844 & 0.5828 & 0.4329 & 0.5298 \\
0.4692 & 0.4235 & 0.2259 & 0.6405 \\
0.0648 & 0.5155 & 0.5798 & 0.2091 \\
0.9883 & 0.3340 & 0.7604 & 0.3798
\end{bmatrix}
\end{align}

and \( n(t) \) is the noise component modelled as the white gaussian signal perturbing each sensor. Figures 2(a)-(b) show the original source signals and the nonlinearly mixed signals at SNR=35dB, respectively. The following demixers\(^2\) are used:

- Linear ICA [23]
- RBF (4-24-4) [8]
- FMLP (4-16-4) [17]
- Proposed RBF-FMLP (4-16-8-4)

The function \( \tilde{q}_i(y_i) \) in (5) is approximated by using a cubic function since the source signals are subgaussianly distributed. The recovered signals from each demixer are shown in Figures 3(a)-(f). Figure 3(a) shows the recovered signals from the Linear ICA where it is evident that linear scheme performs very poorly in retrieving the source signals. It is clearly visible from Figure 3(b) that the RBF demixer has successfully retrieved both sinusoidal sources but not those with rapid discontinuities such as the square and ramp functions. On the other hand, the FMLP demixer with signal constraints as shown in Figure 3(e) performs well in retrieving sources with rapid discontinuities but results in sharp distortion at some points of the sinusoidal sources. The best performance is achieved by the RBF-FMLP demixer with signal constraints (shown in Figure 3(f)) where continuous source signals as well as source signals that exhibit high degree of fluctuation are clearly retrieved. On the other hand, Figures 3(c) and (d) show the recovered signals when signal constraints are removed from the contrast function i.e. by setting \( \alpha_i = 0 \) for all \( i \). It is clearly perceptible from both figures that both FMLP and the proposed network still render acceptable signal separation despite the absence of the constraints. However, the results yield substantially better performance...
when the signal constraints are included as demonstrated in Figures 3(e) and 3(f). Table 1 shows the results of the signal constraints mismatches up to the $6^{th}$ order cumulants between the source signals and the recovered signals as computed from (6) and summed for all $i$. The large numbers in Table 1 are dominated by the mismatches of the $6^{th}$ order cumulant. Clearly with signal constraints, the recovered signals statistics are shaped more closely towards the original source signal statistics. The global rejection index proposed by Amari [23] is not appropriate as it is catered only for linear mixture. An alternative approach towards characterising the efficacy of each demixer, the following performance index is used:

$$\mathcal{P} = 2 \left(1 - \frac{1}{q} \sum_{i=1}^{q} |\rho_i|\right)$$  \hspace{1cm} (23)

where

$$\rho_i = \frac{E[(s_i - E[s_i])^\ast (y_i - E[y_i])]}{\sqrt{E[|s_i - E[s_i]|^2] E[|y_i - E[y_i]|^2]}}$$  \hspace{1cm} (24)

where $\rho_i$ is the normalised cross-correlation. In (24), the notations ‘$\ast$’ and ‘$|$’ denote the complex conjugate and absolute operation, respectively. The proposed performance index is essentially a variant of the mean square error criterion that implicitly takes into account the scale and phase reversal ambiguities. It is desirable to have the performance index as small as possible as this indicates the degree of similarity between the solution and the actual source signals. Figure 4 summarises the results of performance index based on a Monte-Carlo simulation of 100 realisations for each system under various signal-to-noise ratios (SNR). From the figure, it is clearly identified that the RBF-FMLP demixer has the lowest performance index under medium-to-high SNRs and degrades gracefully in the region of low SNRs. The FMLP is slightly worse than the proposed system under medium-to-high SNRs but is substantially degraded in low SNRs while the RBF performs better than the FMLP under noisy condition (low SNRs) due to its regularisation property but the achieved accuracy during signal reconstruction in the region of high SNRs is lower than the FMLP. Finally,
the performance attained by the linear ICA is the worst and falls far from optimal which points out the insufficiency of using linear model.

In the second experiment, real-life speech signals are used. Three speech signals are recorded during measurements and are non-linearly mixed according to \( \mathbf{x}(t) = \mathbf{V} \tanh(2\mathbf{W} \mathbf{s}(t)) \) where \( \mathbf{V} \) and \( \mathbf{W} \) are 3x3 non-singular matrices with elements randomly chosen in [0,1]. The speech signals and the observed signals are plotted in Figures 5(a)-(b). The observed signals are then segmented into blocks of 1024 samples (with 50% overlap between consecutive blocks) and are fed to the demixers. The previous four demixers are used with the following the structures: RBF (3-30-3), FMLP (3-21-3) and the proposed RBF-FMLP (3-18-12-3). All nonlinear demixers exploit the signal constraints. Linear ICA will be based on 3 input nodes and 3 output nodes. Since speech signal is super-gaussian, the marginal pdf can be approximated by

\[
\tilde{q}(y_i) = \frac{p_G(y_i)}{\text{sech}^2(y_i)} \quad \text{where} \quad p_G(y_i) = N(0,1)
\]

is the zero mean, unit variance normal distribution [24].

Figures 6(a)-(d) show the results of the recovered signals from each demixer. The extracted signals from the RBF-FMLP show a high degree of resemblance with the original speech signals although it is visible that some of the peaks have been squashed by the nonlinearity of the hidden neurons activation functions and also the insertion of some residual speech signals during the silent period. In terms of processing speed, the computational complexity of each method has been measured in terms of average processing time based on a Pentium IV 3GHz processor with 2GB RAM memory. The result shows that Linear ICA, RBF, FMLP and the proposed method take approximately 18.5\(\mu s\), 40.6\(\mu s\), 48.7\(\mu s\) and 61.6\(\mu s\) per iteration, respectively. Comparing these numbers, it is perceived that the latter three approaches are augmented with relatively high computational intensity especially the proposed approach. Figure 7 charts the convergence trajectory of each demixer which is obtained by evaluating the performance index in (23). The plot clearly identifies the effectiveness of the proposed RBF-FMLP in comparison with other schemes in terms of convergence speed and steady-state solution. In particular, the proposed scheme renders a convergent rate of approximately 0.075 unit/iteration, FMLP of 0.025 unit/iteration, RBF of 0.03 unit/iteration and Linear ICA of 0.011
unit/iteration. The fast convergence of the proposed method can be attributed to the productive interaction between the RBF and FMLP nodes in maximising the information transfer across the demixer through using the adaptive step sizes scheme. In terms of accuracy of the reconstructed signals as measured by the steady-state solution, RBF-FMLP is twice more accurate than the FMLP, 3 times in the case of RBF and 7 times in the case of Linear ICA. From both Figures 4 and 7, it is noted that the proposed method has improved performance in terms of speed, accuracy and robustness against noise when compared to other methods at the expense of higher computational intensity. On the other hand, one might argue that the performance of the FMLP can be further enhanced by using more neurons in the hidden layer. However, this is not necessarily the case since additional hidden neurons implies a large size network and that larger size network can perform more complex functions which could potentially lead to non-trivial mappings that would result in extraction of arbitrary independent components. The latter is highly undesirable since it will lead to incorrect estimation of the original source signals. So far, the proposed method uses the Gaussian kernel and the sigmoidal class of non-constant, monotonic and continuously differentiable functions with variable gradient as the nonlinear activation functions for the hidden neurons in the first and second each layer, respectively. The optimal selection of the number of hidden neurons required in each layer and the determination of the most appropriate function for the hidden neuron activation function will have direct impact on the performance of the proposed method in separating mixed signals and reconstructing the source signals. This will constitute the subject of our future research.

5. SUMMARY

A hybrid neural network based on RBF and FMLP has been presented for blind source separation of nonlinearly mixed signals. The proposed demixing system is characterised by the RBF network which is located at the first module and then cascaded with the second module FMLP. Simulation results indicate that the proposed scheme yields an improved performance over the conventional RBF and FMLP, and also reveal the inadequacy of using the Linear ICA model in separating nonlinearly mixed signals. The superiority of the
The proposed method can be accredited to the nonlinear regularisation property of the RBF-FMLP in constraining the demixing function to a smaller subset of solutions and thereby reducing the effects of non-trivial mappings that result in arbitrary independent components, the interplay of the local and global mapping bases in constructing continuous signals as well as signals that exhibit high degree of fluctuation and the robustness against noise via regularisation.

6. ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers for their helpful suggestions and comments on improving the readability of the paper.
7. REFERENCES


extended Infomax algorithm for mixed sub-gaussian and super-gaussian sources”, Neural
A. Differentiation of the contrast function with respect to the parameters.

Denoting the change in the contrast function as \( dJ(\Omega) = J(\Omega + d\Omega) - J(\Omega) \) due to infinitesimal change in the network parameters \( d\Omega \), we further arrive at the following from (10):

\[
dJ(\Omega) = -tr [d\xi_2] - tr [d\xi_1] - tr [d\psi] + \lambda^T [d\mu + d\theta] + \frac{1}{2} \sum_i \left( \frac{d}{\sigma_i^2} \|x - c_i\|^2 + \|x - c_i\|^2 \left( \frac{1}{\sigma_i^2} \right) \right) \\
+ \sum_i \frac{d\sigma_i^2}{\sigma_i^4} + \left[ \varphi + \alpha \circ \hat{f} \right]^T dy - d \log |A| \quad (25)
\]

By considering the differentials of the above with respect to the differentials of the parameter set \( \Omega = \{W_2, W_1, M, \theta, \{\sigma_i^2\}_{i=1}^N, \{c_i\}_{i=1}^N\} \), we obtain the following:

\[
\frac{dJ}{d\xi_2} = -I + \left[ \varphi + \alpha \circ \hat{f} \right] y^T \quad (26)
\]

\[
\frac{dJ}{d\xi_1} = -I + M \left[ \lambda + D_g W_2^T \varphi + \alpha \circ \hat{f} \right] v^T W_1^T \\
= -I + M \left[ \lambda + D_g W_2^T \rho^{(2)} \right] v^T W_1^T \quad (27)
\]

\[
\frac{dJ}{d\psi} = -\left[ I - \text{diag} \{\lambda u^T + D_g W_2^T \rho^{(2)} u^T\} \right] M \quad (28)
\]

\[
\frac{dJ}{d\theta} = \lambda + D_g W_2^T \rho^{(2)} \quad (29)
\]
\[
\frac{dJ}{d\sigma_i^2} = -\frac{1}{2(\sigma_i^2)^2} \|x - c_i\|^2 + \frac{1}{\sigma_i^2} + \frac{1}{2} \left[ \varphi + \alpha \circ \hat{f} \right] \mathbf{w}^{(i)} \frac{\nu_i}{(\sigma_i^2)^2} \|x - c_i\|^2
\]

\[
= \left( \left[ \varphi + \alpha \circ \hat{f} \right] \mathbf{M} \mathbf{w}^{(i)} \nu_i - 1 \right) \|x - c_i\|^2 + \frac{1}{\sigma_i^2}
\]

\[
= \left( \left[ \mathbf{r}^{(i)} \right]^T \mathbf{M} \mathbf{w}^{(i)} \nu_i - 1 \right) \|x - c_i\|^2 + \frac{1}{\sigma_i^2}
\]

\[
\frac{dJ}{dc_i(t)} = \left( \left[ \mathbf{r}^{(i)}(t) \right] \mathbf{W}_i(t) \delta_i(t) - 1 \right) \frac{(x(t) - c_i(t))^2}{\sigma_i^2(t)} + \mathbf{A}^{-1}(t) \delta_i(t)
\]

\[
(30)
\]

\[
(31)
\]

B. Cumulants.

The central moment for a random variable \( y \) is defined as \( \mu_x(n) = \mathbb{E} \left[ (y - m_y)^n \right] = \int_{-\infty}^{\infty} (y - m_y)^n p(y) dy \)

where \( m_y = \mathbb{E}[y] = \int_{-\infty}^{\infty} y p(y) dy \). By using this definition, we may define the cumulants up to the 8th order as follows:

1. \( \text{cum}[y] = m_y \)
2. \( \text{cum}[y, y] = \mu_y(2) \)
3. \( \text{cum}[y, y, y] = \mu_y(3) \)
4. \( \text{cum}[y, y, y, y] = \mu_y(4) - 3 \left( \mu_y(2) \right)^2 \)
5. \( \text{cum}[y, y, y, y, y] = \mu_y(5) - 10 \mu_y(3) \mu_y(2) \)
6. \( \text{cum}[y, y, y, y, y, y] = \mu_y(6) - 15 \mu_y(4) \mu_y(2) - 10 \left( \mu_y(3) \right)^2 + 30 \left( \mu_y(2) \right)^3 \)
7. \( \text{cum}[y, y, y, y, y, y, y] = \mu_y(7) - 21 \mu_y(5) \mu_y(2) - 35 \mu_y(4) \mu_y(3) - 210 \mu_y(3) \left( \mu_y(2) \right)^2 \)
8. \( \text{cum}[y, y, y, y, y, y, y, y] = \mu_y(8) - 28 \mu_y(6) \mu_y(2) - 56 \mu_y(5) \mu_y(3) - 35 \left( \mu_y(4) \right)^2 + 420 \mu_y(4) \left( \mu_y(2) \right)^2 + 560 \left( \mu_y(3) \right)^2 \mu_y(2) - 630 \left( \mu_y(2) \right)^4 \)
Figure 1
Figure 2
Recovered signals using the Linear ICA

Recovered signals using the RBF with signal constraints

Recovered signals using the FMLP without signal constraints

Recovered signals using the RBF-FMLP without signal constraints

Recovered signals using the FMLP with signal constraints

Recovered signals using the RBF-FMLP with signal constraints

Figure 3
Figure 4

Figure 5
Figure 6

(a) Recovered signals using the Linear ICA

(b) Recovered signals using the RBF

(c) Recovered signals using the FMLP

(d) Recovered signals using the RBF-FMLP

(time (s))
Figure 7

Convergence plot

Performance Index

- Linear ICA
- RBF
- FMLP
- RBF-FMLP

number of iteration
<table>
<thead>
<tr>
<th>Demixer</th>
<th>Without signal constraints</th>
<th>With signal constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMLP</td>
<td>$2.187 \times 10^3$</td>
<td>184.2</td>
</tr>
<tr>
<td>Proposed RBF-FMLP</td>
<td>$1.712 \times 10^4$</td>
<td>213.3</td>
</tr>
</tbody>
</table>

Table 1
Figure/Table Captions

Figure 1. Proposed hybrid RBF-FMLP nonlinear demixer.

Figure 2. Signals in experiment 1.
   (a) Original source signals.
   (b) Nonlinearly mixed (observed) signals.

Figure 3. Recovered signals in experiment 1.
   (a) Linear ICA.
   (b) RBF.
   (c) FMLP without signal constraints.
   (d) RBF-FMLP without signal constraints.
   (e) FMLP with signal constraints.
   (f) RBF-FMLP with signal constraints.

Figure 4. Performance index for each tested demixer in experiment 1.

Figure 5. Signals in experiment 2.
   (c) Original speech signals.
   (d) Nonlinearly mixed (observed) speech signals.

Figure 6. Recovered signals in experiment 2.
   (g) Linear ICA.
   (h) RBF.
(i) FMLP.

(j) RBF-FMLP.

Figure 7. Convergence of the performance index for each tested demixer in experiment 2.

Table 1. Signal constraints mismatches.
Footnotes

1 Throughout the paper, the symbol ‘.’ appearing on top of a variable denotes the derivatives operator. In addition, the number of dots represents the order of derivatives.

2 The notation (x-y-z) denotes the size of the neural demixer with x input nodes, y hidden nodes and z output nodes while (x-y_1-y_2-z) denotes x input nodes, y_1 hidden nodes in first layer, y_2 hidden nodes in second layer and z output nodes.