ELSEVIER

Denoising using local projective subspace methods

P. Gruber[a], K. Stadlthanner[a], M. Böhm[a], F.J. Theis[a], E.W. Lang[a],*, A.M. Tomé[b],
A.R. Teixeira[b], C.G. Puntonet[c], J.M. Gorriz Saéz[c]

[a]*Institute of Biophysics, Neuro- and Bioinformatics Group, University of Regensburg, 93040 Regensburg, Germany*
[b]*Departamento de Electrónica e Telecomunicações/IEETA,Universidade de Aveiro, 3810 Aveiro, Portugal*
[c]*Departamento Arqitectura y Técnologia de Computadores,Universidad de Granada, 18371 Granada, Spain*

Available online 3 February 2006

Abstract

In this paper we present denoising algorithms for enhancing noisy signals based on Local ICA (LICA), Delayed AMUSE (dAMUSE) and Kernel PCA (KPCA). The algorithm LICA relies on applying ICA locally to clusters of signals embedded in a high-dimensional feature space of delayed coordinates. The components resembling the signals can be detected by various criteria like estimators of kurtosis or the variance of autocorrelations depending on the statistical nature of the signal. The algorithm proposed can be applied favorably to the problem of denoising multi-dimensional data. Another projective subspace denoising method using delayed coordinates has been proposed recently with the algorithm dAMUSE. It combines the solution of blind source separation problems with denoising efforts in an elegant way and proofs to be very efficient and fast. Finally, KPCA represents a non-linear projective subspace method that is well suited for denoising also. Besides illustrative applications to toy examples and images, we provide an application of all algorithms considered to the analysis of protein NMR spectra.
© 2006 Elsevier B.V. All rights reserved.

Keywords: Local ICA; Delayed AMUSE; Projective subspace denoising embedding

1. Introduction

The interpretation of recorded signals is often hampered by the presence of noise. This is especially true with biomedical signals which are buried in a large noise background most often. Statistical analysis tools like principal component analysis (PCA), singular spectral analysis (SSA), independent component analysis (ICA) etc. quickly degrade if the signals exhibit a low signal to noise ratio (SNR). Furthermore due to their statistical nature, the application of such analysis tools can also lead to extracted signals with a larger SNR than the original ones as we will discuss below in case of nuclear magnetic resonance (NMR) spectra.

Hence in the signal processing community many denoising algorithms have been proposed [5,12,18,39] including algorithms based on local linear projective noise reduction.

The idea is to project noisy signals in a high-dimensional space of delayed coordinates, called feature space henceforth. A similar strategy is used in SSA [9,20] where a matrix composed of the data and their delayed versions is considered. Then, a singular value decomposition (SVD) of the data matrix or a PCA of the related correlation matrix is computed. Noise contributions to the signals are then removed locally by projecting the data onto a subset of principal directions of the eigenvectors of the SVD or PCA analysis related with the deterministic signals.

Modern multi-dimensional NMR spectroscopy is a very versatile tool for the determination of the native 3D structure of biomolecules in their natural aqueous environment [7,10]. Proton NMR is an indispensable contribution to this structure determination process but is hampered by the presence of the very intense water (H_2O) proton signal. The latter causes severe baseline distortions and obscures weak signals lying under its skirts. It has been shown [26,29] that blind source separation (BSS) techniques like ICA can contribute to the removal of the water artifact in proton NMR spectra.

*Corresponding author.

E-mail addresses: elmar.lang@biologie.uni-regensburg.de
(E.W. Lang), ana@ieeta.pt (A.M. Tomé), carlos@atc.ugr.es (C.G. Puntonet).

ICA techniques extract a set of signals out of a set of measured signals without knowing how the mixing process is carried out [2,13]. Considering that the set of measured spectra X is a linear combination of a set of independent component (ICs) S, i.e. $X = AS$, the goal is to estimate the inverse of the mixing matrix A, using only the measured spectra, and then compute the ICs. Then the spectra are reconstructed using the mixing matrix A and those ICs contained in S which are not related with the water artifact. Unfortunately the statistical separation process in practice introduces additional noise not present in the original spectra. Hence denoising as a post-processing of the artifact-free spectra is necessary to achieve the highest possible SNR of the reconstructed spectra. It is important that the denoising does not change the spectral characteristics like integral peak intensities as the deduction of the 3D structure of the proteins heavily relies on the latter.

We propose two new approaches to this denoising problem and compare the results to the established Kernel PCA (KPCA) denoising [19,25].

The first approach Local ICA (LICA) concerns a local projective denoising algorithm using ICA. Here it is assumed that the noise can, at least locally, be represented by a stationary Gaussian white noise. Signals usually come from a deterministic or at least predictable source and can be described as a smooth function evaluated at discrete time steps small enough to capture the characteristics of the function. That implies, using a dynamical model for the data, that the signal embedded in delayed coordinates resides within a sub-manifold of the feature space spanned by these delayed coordinates. With local projective denoising techniques, the task is to detect this signal manifold. We will use LICA to detect the statistically most interesting submanifold. In the following we call this manifold the signal + noise subspace since it contains all of the signal plus that part of the noise components which lie in the same subspace. Parameter selection within LICA will be effected with a minimum description length (MDL) criterion [40,6] which selects optimal parameters based on the data themselves.

For the second approach we combine the ideas of solving BSS problems algebraically using a generalized eigenvector decomposition (GEVD) [28] with local projective denoising techniques. We propose, like in the algorithm for multiple unknown signals extraction (AMUSE) [37], a GEVD of two correlation matrices i.e, the *simultaneous* diagonalization of a matrix pencil formed with a correlation matrix and a matrix of delayed correlations. These algorithms are exact and fast but sensitive to noise. There are several proposals to improve efficiency and robustness of these algorithms when noise is present [2,8]. They mostly rely on an *approximative* joint diagonalization of a set of correlation or cumulant matrices like the algorithm second order blind identification (SOBI) [1]. The algorithm we propose, called delayed AMUSE (dAMUSE) [36], computes a GEVD of the congruent matrix pencil in a high-dimensional feature space of delayed coordinates. We show that

the estimated signal components correspond to filtered versions of the underlying uncorrelated source signals. We also present an algorithm to compute the eigenvector matrix of the pencil which involves a two step procedure based on the standard eigenvector decomposition (EVD) approach. The advantage of this two step procedure is related with a dimension reduction between the two steps according to a threshold criterion. Thereby estimated signal components related with noise only can be neglected thus performing a denoising of the reconstructed signals.

As a third denoising method we consider KPCA based denoising techniques [19,25] which have been shown to be very efficient outperforming linear PCA. KPCA actually generalizes linear PCA which hitherto has been used for denoising. PCA denoising follows the idea that retaining only the principal components with highest variance to reconstruct the decomposed signal, noise contributions which should correspond to the low variance components can deliberately be omitted hence reducing the noise contribution to the observed signal. KPCA extends this idea to non-linear signal decompositions. The idea is to project observed data non-linearly into a high-dimensional feature space and then to perform linear PCA in feature space. The trick is that the whole formalism can be cast into dot product form hence the latter can be replaced by suitable kernel functions to be evaluated in the lower-dimensional input space instead of the high-dimensional feature space. Denoising then amounts to estimating appropriate pre-images in input space of the non-linearly transformed signals.

The paper is organized as follows: Section 1 presents an introduction and discusses some related work. In Section 2 some general aspects about embedding and clustering are discussed, before in Section 3 the new denoising algorithms are discussed in detail. Section 4 presents some applications to toy as well as to real world examples and Section 5 draws some conclusions.

2. Feature space embedding

In this section we introduce new denoising techniques and propose algorithms using them. At first we present the signal processing tools we will use later on.

2.1. Embedding using delayed coordinates

A common theme of all three algorithms presented is to embed the data into a high-dimensional feature space and try to solve the noise separation problem there. With the LICA and the dAMUSE we embed signals in delayed coordinates and do all computations directly in the space of delayed coordinates. The KPCA algorithm considers a non-linear projection of the signals to a feature space but performs all calculations in input space using the kernel trick. It uses the space of delayed coordinates only implicitly as intermediate step in the non-linear

transformation since for that transformation the signal at different time steps is used.

Delayed coordinates are an ideal tool for representing the signal information. For example in the context of chaotic dynamical systems, embedding an observable in delayed coordinates of sufficient dimension already captures the full dynamical system [30]. There also exists a similar result in statistics for signals with a finite decaying memory [24].

Given a group of N sensor signals, $x[l] = [x_0[l], \ldots, x_{N-1}[l]]^T$ sampled at time steps $l = 0, \ldots, L-1$, a very convenient representation of the signals embedded in delayed coordinates is to arrange them componentwise into component trajectory matrices $X_i, i = 0, \ldots, N-1$ [20]. Hence embedding can be regarded as a mapping that transforms a one-dimensional time series $x_i = (x_i[0], x_i[1], \ldots, x_i[L-1])$ into a multi-dimensional sequence of lagged vectors. Let M be an integer (window length) with $M < L$. The embedding procedure then forms $L - M + 1$ lagged vectors which constitute the columns of the component trajectory matrix. Hence given sensor signals $x[l]$, registered for a set of L samples, their related component trajectory matrices are given by

$$
\begin{aligned}
X_i &= \begin{bmatrix} x_i[M-1] & x_i[M] & \ldots & x_i[L-1] \end{bmatrix} \\
&= \begin{bmatrix}
x_i[M-1] & x_i[M] & \cdots & x_i[L-1] \\
x_i[M-2] & x_i[M-1] & \cdots & x_i[L-2] \\
\vdots & \vdots & \ddots & \vdots \\
x_i[0] & x_i[1] & \cdots & x_i[L-M]
\end{bmatrix}
\end{aligned} \quad (1)
$$

and encompass M delayed versions of each signal component $x_i[l-m], m = 0, \ldots, M-1$ collected at time steps $l = M-1, \ldots, L-1$. Note that a trajectory matrix has identical entries along each diagonal. The total trajectory matrix of the set X will be a concatenation of the component trajectory matrices X_i computed for each sensor, i.e

$$
X = [X_1, X_2, \ldots, X_N]^T. \quad (2)
$$

Note that the embedded sensor signal is also formed by a concatenation of embedded component vectors, i.e. $x[l] = [x_0[l], \ldots, x_{N-1}[l]]$. Also note that with LICA we deal with single column vectors of the trajectory matrix only, while with dAMUSE we consider the total trajectory matrix.

2.2. Clustering

In our context clustering of signals means rearranging the signal vectors, sampled at different time steps, by similarity. Hence for signals embedded in delayed coordinates, the idea is to look for K disjoint sub-trajectory matrices to group together similar column vectors of the trajectory matrix X.

A clustering algorithm like *k-means* [15] is appropriate for problems where the time structure of the signal is irrelevant. If, however, time or spatial correlations matter, clustering should be based on finding an appropriate partitioning of $\{M-1, \ldots, L-1\}$ into K successive segments, since this preserves the inherent correlation structure of the signals. In any case the number of columns in each sub-trajectory matrix $X^{(j)}$ amounts to L_j such that the following completeness relation holds:

$$
\sum_{j=1}^{K} L_j = L - M + 1. \quad (3)
$$

The mean vector m_j in each cluster can be considered a prototype vector and is given by

$$
m_j = \frac{1}{L_j} X c_j = \frac{1}{L_j} X^{(j)}[1, \ldots, 1]^T, \quad j = 1, \ldots, K, \quad (4)
$$

where c_j is a vector with L_j entries equal to one which characterizes the clustering. Note that after the clustering the set $\{k = 0, \ldots, L-M-1\}$ of indices of the columns of X is split in K disjoint subsets K_j. Each trajectory sub-matrix $X^{(j)}$ is formed with those columns of the matrix X, the indices of which belong to the subset K_j of indices.

2.3. Principal component analysis and independent component analysis

PCA [23] is one of the most common multi-variate data analysis tools. It tries to linearly transform given data into uncorrelated data (feature space). Thus in PCA [4] a data vector is represented in an orthogonal basis system such that the projected data have maximal variance. PCA can be performed by eigenvalue decomposition of the data covariance matrix. The orthogonal transformation is obtained by diagonalizing the centered covariance matrix of the data set.

In ICA, given a random vector, the goal is to find its statistically ICs. In contrast to correlation-based transformations like PCA, ICA renders the output signals as statistically independent as possible by evaluating higher-order statistics. The idea of ICA was first expressed by Jutten and Hérault [11] while the term ICA was later coined by Comon [3]. With LICA we will use the popular FastICA algorithm by Hyvärinen and Oja [14], which performs ICA by maximizing the non-Gaussianity of the signal components.

3. Denoising algorithms

3.1. Local ICA denoising

The LICA algorithm we present is based on a local projective denoising technique using an MDL criterion for parameter selection. The idea is to achieve denoising by locally projecting the embedded noisy signal into a

lower-dimensional subspace which contains the characteristics of the noise free signal. Finally the signal has to be reconstructed using the various candidates generated by the embedding.

Consider the situation, where we have a signal $x_i^0[l]$ at discrete time steps $l = 0, \ldots, L-1$ but only its noise corrupted version $x_i[l]$ is measured

$$x_i[l] = x_i^0[l] + e_i[l], \tag{5}$$

where $e[l]$ are samples of a random variable with Gaussian distribution, i.e. x_i equals x_i^0 up to additive stationary white noise.

3.1.1. Embedding and clustering

First the noisy signal $x_i[l]$ is transformed into a high-dimensional signal $\mathbf{x}_i[l]$ in the M-dimensional space of delayed coordinates according to

$$\mathbf{x}_i[l] := [x_i[l], \ldots, x_i[l - M + 1 \bmod L]]^{\mathrm{T}} \tag{6}$$

which corresponds to a column of the trajectory matrix in Eq. (1).

To simplify implementation, we want to ensure that the delayed signal, like the original signal, (trajectory matrix) is given at L time steps instead of $L - M + 1$. This can be achieved by using the samples in round robin manner, i.e. by closing the end and the begin of each delayed signal and cutting out exactly L components in accord with the delay. If the signal contains a trend or its statistical nature is significantly different at the end compared to the beginning, then this leads to compatibility problems of the beginning and end of the signal. We can easily resolve this misfit by replacing the signal with a version where we add the signal in reverse order, hence avoiding any sudden change in signal amplitude which would be smoothed out by the algorithm.

The problem can now be localized by selecting K clusters in the feature space of delayed coordinates of the signal $\{\mathbf{x}_i[l] \mid l = 0, \ldots, L-1\}$. Clustering can be achieved by a *k-means* cluster algorithm as explained in Section 2.2. But *k-means* clustering is only appropriate if the variance or the kurtosis of a signal do not depend on the inherent signal structure. For other noise selection schemes like choosing the noise components based on the variance of the autocorrelation, it is usually better to find an appropriate partitioning of the set of time steps $\{0, \ldots, L-1\}$ into K successive segments, since this preserves the inherent time structure of the signals.

For other noise selection methods like choosing the noise components based on the variance of the autocorrelation it is usually better to find an appropriate partition of the set of time steps $\{0, \ldots, L-1\}$ into K successive segments, since this preserves the inherent time structure of the signal.

Note that the clustering does not change the data but only changes its time sequence, i.e. permutes and regroups the columns of the trajectory matrix and separates it into K sub-matrices.

3.1.2. Decomposition and denoising

After centering, i.e. removing the mean in each cluster, we can analyze the M-dimensional signals in these K clusters using PCA or ICA. The PCA case (Local PCA (LPCA)) is studied in [38] so in the following we will propose an ICA based denoising.

Using ICA, we extract M ICs from each delayed signal. Like in all projection based denoising algorithms, noise reduction is achieved by projecting the signal into a lower-dimensional subspace. We used two different criteria to estimate the number p of signal + noise components, i.e. the dimension of the signal subspace onto which we project after applying ICA.

- One criterion is a consistent MDL estimator of p_{MDL} for the data model in Eq. (5) [38]

$$
\begin{aligned}
p_{\mathrm{MDL}} = \ & \operatorname*{argmin}_{p=0,\ldots,M-1} \mathrm{MDL}(M, L, p, (\lambda_j), \gamma) \\
& \times \operatorname*{argmin}_{p=0,\ldots,M-1} \Bigg\{ -((M-p)L) \ln \left(\frac{\Pi_{j=p+1}^{M} \lambda_j^{(1/(M-p))}}{(1/(M-p))\sum_{j=p+1}^{M} \lambda_j} \right) \\
& + \left(pM - \frac{p^2}{2} + \frac{p}{2} + 1 \right) \left(\frac{1}{2} + \ln \gamma \right) \\
& - \frac{pM - (p^2/2) + (p/2) + 1}{p} \left(\frac{1}{2} \ln \frac{2}{L} + \sum_{j=1}^{p} \ln \lambda_j - \ln \sum_{j=1}^{M-1} \lambda_j \right) \Bigg\},
\end{aligned}
\tag{7}
$$

where λ_j denote the variances of the signal components in feature space, i.e. after applying the de-mixing matrix which we estimate with the ICA algorithm. To retain the relative strength of the components in the mixture, we normalize the rows of the de-mixing matrix to unit norm. The variances are ordered such that the smallest eigenvalues λ_j correspond to directions in feature space most likely to be associated with noise components only.

The first term in the MDL estimator represents the likelihood of the $m - p$ Gaussian white noise components. The third term stems from the estimation of the description length of the signal part (first p components) of the mixture based on their variances. The second term acts as a penalty term to favor parsimonious representations of the data for short time series, and becomes insignificant in the limit $L \to \infty$ since it does not depend on L while the other two terms grow without bounds. The parameter γ controls this behavior and is a parameter of the MDL estimator, hence of the final denoising algorithm. By experience, good values for γ seem to be 32 or 64.

- Based on the observations reported by [17] and our observations that, in some situations, the MDL estimator tends to significantly underestimate the number of noise components, we also used another approach: We clustered the variances of the signal components into two clusters using *k-means* clustering and defined p_{cl} as the number of elements in the cluster which contains the largest eigenvalue. This yields a

good estimate of the number of signal components, if the noise variances are not clustered well enough together but, nevertheless, are separated from the signal by a large gap. More details and simulations corroborating our observations can be found in Section 4.1.1.

Depending on the statistical nature of the data, the ordering of the components in the MDL estimator can be achieved using different methods. For data with a non-Gaussian distribution, we select the noise component as the component with the smallest value of the kurtosis as Gaussian noise corresponds to a vanishing kurtosis. For non-stationary data with stationary noise, we identify the noise by the smallest variance of its autocorrelation.

3.1.3. Reconstruction

In each cluster the centering is reversed by adding back the cluster mean. To reconstruct the noise reduced signal, we first have to reverse the clustering of the data to yield the signal $x_i^e[l] \in \mathbb{R}^M$ by concatenating the trajectory submatrices and reversing the permutation done during clustering. The resulting trajectory matrix does not possess identical entries in each diagonal. Hence we average over the candidates in the delayed data, i.e. over all entries in each diagonal:

$$x_i^e[l] := \frac{1}{M} \sum_{j=0}^{M-1} x_i^e[l + j \bmod L]_j, \tag{8}$$

where $x_i^e[l]_j$ stands for the jth component of the enhanced vector x_i^e at time step l. Note that the summation is done over the diagonals of the trajectory matrix so it would yield x_i if performed on the original delayed signal x_i.

3.1.4. Parameter estimation

We still have to find optimal values for the global parameters M and K. Their selection again can be based on a MDL criterion for the detected noise $e := x - x_e$. Accordingly we apply the LICA algorithm for different M and K and embed each of these error signals $e(M, K)$ in delayed coordinates of a fixed large enough dimension $\hat{M}$ and choose the parameters M_0 and K_0 such that the MDL criterion estimating the noisiness of the error signal is minimal. The MDL criterion is evaluated with respect to the eigenvalues $\lambda_j(M, K)$ of the correlation matrix of $e(M, K)$ such that

$$(M_0, K_0) = \underset{M, K}{\arg\min} \, \text{MDL}(\hat{M}, L, 0, (\lambda_j(M, K)), \gamma). \tag{9}$$

3.2. Denoising using delayed AMUSE

Signals with an inherent correlation structure like time series data can as well be analyzed using second-order BSS techniques only [22,35]. GEVD of a matrix pencil [33,37] or a joint approximative diagonalization of a set of correlation matrices [1] is then usually considered. Recently we proposed an algorithm based on a generalized eigenvalue decomposition in a feature space of delayed coordinates [35]. It provides means for BSS and denoising simultaneously.

3.2.1. Embedding

Assuming that each sensor signal is a linear combination $X = AS$ of N underlying but unknown source signals s_i, a source signal trajectory matrix S can be written in analogy to Eqs. (1) and (2). Then the mixing matrix A is a block matrix with a diagonal matrix in each block:

$$A = \begin{bmatrix} a_{11}I_{M \times M} & a_{12}I_{M \times M} & \cdots & a_{1N}I_{M \times M} \\ a_{21}I_{M \times M} & a_{22}I_{M \times M} & \cdots & \cdots \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1}I_{M \times M} & a_{N2}I_{M \times M} & \cdots & a_{NN}I_{M \times M} \end{bmatrix}. \tag{10}$$

The matrix $I_{M \times M}$ represents the identity matrix, and in accord with an instantaneous mixing model the mixing coefficient a_{ij} relates the sensor signal x_i with the source signal s_j.

3.2.2. Generalized eigenvector decomposition

The delayed correlation matrices of the matrix pencil are computed with one matrix X_r obtained by eliminating the first k_i columns of X and another matrix, X_l, obtained by eliminating the last k_i columns. Then, the delayed correlation matrix $R_x(k_i) = X_r X_l^T$ will be an $NM \times NM$ matrix. Each of these two matrices can be related with a corresponding matrix in the source signal domain:

$$R_x(k_i) = AR_s(k_i)A^T = AS_r S_l^T A^T. \tag{11}$$

Then the two pairs of matrices $(R_x(k_1), R_x(k_2))$ and $(R_s(k_1), R_s(k_2))$ represent congruent pencils [34] with the following properties:

- Their eigenvalues are the same, i.e., the eigenvalue matrices of both pencils are identical: $D_x = D_s$.
- If the eigenvalues are non-degenerate (distinct values in the diagonal of the matrix $D_x = D_s$), the corresponding eigenvectors are related by the transformation $E_s = A^T E_x$.

Assuming that all sources are uncorrelated, the matrices $R_s(k_i)$ are block diagonal, having block matrices $R_{mm}(k_i) = S_{ri} S_{li}^T$ along the diagonal. The eigenvector matrix of the GEVD of the pencil $(R_s(k_1), R_s(k_2))$ again forms a block diagonal matrix with block matrices E_{mm} forming $M \times M$ eigenvector matrices of the GEVD of the pencils $(R_{mm}(k_1), R_{mm}(k_2))$. The uncorrelated components can then be estimated from linearly transformed sensor signals via

$$Y = E_x^T X = E_x^T AS = E_s^T S \tag{12}$$

hence turn out to be filtered versions of the underlying source signals. As the eigenvector matrix E_s is a block

diagonal matrix, there are M signals in each column of Y which are a linear combination of one of the source signals and its delayed versions. Then the columns of the matrix E_{mm} represent impulse responses of finite impulse response (FIR) filters. Considering that all the columns of E_{mm} are different, their frequency response might provide different spectral densities of the source signal spectra. Then NM output signals y encompass M filtered versions of each of the N estimated source signals.

3.2.3. Implementation of the GEVD

There are several ways to compute the generalized eigenvalue decomposition. We resume a procedure valid if one of the matrices of the pencil is symmetric positive definite. Thus, we consider the pencil $(R_x(0), R_x(k_2))$ and perform the following steps:

Step 1: Compute a standard eigenvalue decomposition of $R_x(0) = V \Lambda V^T$, i.e, compute the eigenvectors v_i and eigenvalues λ_i. As the matrix is symmetric positive definite, the eigenvalues can be arranged in descending order $(\lambda_1 > \lambda_2 > \cdots > \lambda_{NM})$. This procedure corresponds to the usual whitening step in many ICA algorithms. It can be used to estimate the number of sources, but it can also be considered a strategy to reduce noise much like with PCA denoising. Dropping small eigenvalues amounts to a projection from a high-dimensional feature space onto a lower-dimensional manifold representing the signal + noise subspace. Thereby it is tacitly assumed that small eigenvalues are related with noise components only. Here we consider a variance criterion to choose the most significant eigenvalues, those related with the embedded deterministic signal, according to

$$\frac{\lambda_1 + \lambda_2 + \cdots + \lambda_l}{\lambda_1 + \lambda_2 + \cdots \lambda_{NM}} \geqslant TH. \tag{13}$$

If we are interested in the eigenvectors corresponding to directions of high variance of the signals, the threshold TH should be chosen such that their maximum energy is preserved. Similar to the whitening phase in many BSS algorithms, the data matrix X can be transformed using

$$Q = \Lambda^{-1/2} V^T \tag{14}$$

to calculate a transformed matrix of delayed correlations $C(k_2)$ to be used in the next step. The transformation matrix can be computed using either the l most significant eigenvalues, in which case denoising is achieved, or all eigenvalues and respective eigenvectors. Also note, that Q represents a $l \times NM$ matrix if denoising is considered.

Step 2: Use the transformed delayed correlation matrix $C(k_2) = Q R_x(k_2) Q^T$ and its standard eigenvalue decomposition: the eigenvector matrix U and eigenvalue matrix D_x.

The eigenvectors of the pencil $(R_x(0), R_x(k_2))$, which are not normalized, form the columns of the eigenvector matrix $E_x = Q^T U = V \Lambda^{-1/2} U$. The ICs of the delayed sensor signals can then be estimated via the transformation given below, yielding l (or NM) signals, one signal per row

of Y:

$$Y = E_x^T X = U^T Q X = U^T \Lambda^{-1/2} V^T X. \tag{15}$$

The first step of this algorithm is therefore equivalent to a PCA in a high-dimensional feature space [9,39], where a matrix similar to Q is used to project the data onto the signal manifold.

3.3. Kernel PCA based denoising

Kernel PCA has been developed by [19], hence we give here only a short summary for convenience. PCA only extracts linear features though with suitable *non-linear* features more information could be extracted. It has been shown [19] that KPCA is well suited to extract interesting non-linear features in the data. KPCA first maps the data x_i into some high-dimensional feature space Ω through a non-linear mapping $\Phi : \mathbb{R}^n \to \mathbb{R}^m, m > n$ and then performs linear PCA on the mapped data in the feature space Ω. Assuming centered data in feature space, i.e. $\sum_{k=1}^{l} \Phi(x_k) = 0$, to perform PCA in space Ω amounts to finding the eigenvalues $\lambda > 0$ and eigenvectors $\omega \in \Omega$ of the correlation matrix $\bar{R} = 1/l \sum_{j=1}^{l} \Phi(x_j) \Phi(x_j)^T$.

Note that all ω with $\lambda \neq 0$ lie in the subspace spanned by the vectors $\Phi(x_1), \ldots, \Phi(x_l)$. Hence the eigenvectors can be represented via

$$\omega = \sum_{i=1}^{l} \alpha_i \Phi(x_i). \tag{16}$$

Multiplying the eigenequation with $\Phi(x_k)$ from the left the following modified eigenequation is obtained

$$K \alpha = l \lambda \alpha \tag{17}$$

with $\lambda > 0$. The eigenequation now is cast in the form of dot products occurring in feature space through the $l \times l$ matrix K with elements $K_{ij} = (\Phi(x_i) \cdot \Phi(x_j)) = k(x_i, x_j)$ which are represented by kernel functions $k(x_i, x_j)$ to be evaluated in the input space. For feature extraction any suitable kernel can be used and knowledge of the non-linear function $\Phi(x)$ is not needed. Note that the latter can always be reconstructed from the principal components obtained. The image of a data vector under the map Φ can be reconstructed from its projections β_k via

$$\hat{P}_n \Phi(x) = \sum_{k=1}^{n} \beta_k \omega_k = \sum_{k=1}^{n} (\omega_k \cdot \Phi(x)) \omega_k \tag{18}$$

which defines the projection operator $\hat{P}_n$. In denoising applications, n is deliberately chosen such that the squared reconstruction error

$$e_{rec}^2 = \sum_{i=1}^{l} \| \hat{P}_n \Phi(x_i) - \Phi(x_i) \|^2 \tag{19}$$

is minimized. To find a corresponding approximate representation of the data in input space, the so-called pre-image, it is necessary to estimate a vector $z \in \mathbb{R}^N$ in

input space such that

$$\rho(\mathbf{z}) = \|\hat{P}_n \Phi(\mathbf{x}) - \Phi(\mathbf{z})\|^2 = k(\mathbf{z}, \mathbf{z}) - 2\sum_{k=1}^{n} \beta_k \sum_{i=1}^{l} \alpha_i^k k(\mathbf{x}_i, \mathbf{z})$$

(20)

is minimized. Note that an analytic solution to the pre-image problem has been given recently in case of invertible kernels [16]. In denoising applications it is hoped that the deliberately neglected dimensions of minor variance contain noise mostly and $\mathbf{z}$ represents a denoised version of $\mathbf{x}$. Eq. (20) can be minimized via gradient descent techniques.

4. Applications and simulations

In this section we will first present results and concomitant interpretation of some experiments with toy data using different variations of the LICA denoising algorithm. Next we also present some test simulations using toy data of the algorithm dAMUSE. Finally we will discuss the results of applying the three different denoising algorithms presented above to a real world problem, i.e. to enhance protein NMR spectra contaminated with a huge water artifact.

4.1. Denoising with Local ICA applied to toy examples

We will present some sample experimental results using artificially generated signals and random noise. As the latter is characterized by a vanishing kurtosis, the LICA based denoising algorithm uses the component kurtosis for noise selection.

4.1.1. Discussion of an MDL based subspace selection

In the LICA denoising algorithm the MDL criterion is also used to select the number of noise components in each cluster. This works without prior knowledge of the noise strength. Since the estimation is based solely on statistical properties, it produces suboptimal results in some cases, however. In Fig. 1 we compare, for an artificial signal with a known additive white Gaussian noise, the denoising achieved with the MDL based estimation of the subspace

dimension versus the estimation based on the noise level. The latter is done using a threshold on the variances of the components in feature space such that only the signal part is conserved. Fig. 1 shows that the threshold criterion works slightly better in this case, though the MDL based selection can obtain a comparable level of denoising. However, the smaller SNR indicates that the MDL criterion favors some over-modelling of the signal subspace, i.e. it tends to underestimate the number of noise components in the registered signals.

In [17] the conditions, such as the noise not being completely white, which lead to a strong over-modelling are identified. Over-modelling also happens frequently, if the eigenvalues of the covariance matrix related with noise components, are not sufficiently close together and are not separated from the signal components by a gap. In those cases a clustering criterion for the eigenvalues seems to yield better results, but it is not as generic as the MDL criterion.

4.1.2. Comparisons between LICA and LPCA

Consider the artificial signal shown in Fig. 1 with varying additive Gaussian white noise. We apply the LICA denoising algorithm using either an MDL criterion or a threshold criterion for parameter selection. The results are depicted in Fig. 2.

The first and second diagram of Fig. 2 compare the performance, here the enhancement of the SNR and the mean square error, of LPCA and LICA depending on the input SNR. Note that a source SNR of 0 describes a case where signal and noise have the same strength, while negative values indicate situations where the signal is buried in the noise. The third graph shows the difference in kurtosis of the original signal and the source signal in dependence on the input SNR. All three diagrams were generated with the same data set, i.e. the same signal and, for a given input SNR, the same additive noise.

These results suggest that a LICA approach is more effective when the signal is infested with a large amount of noise, whereas a LPCA seems better suited for signals with
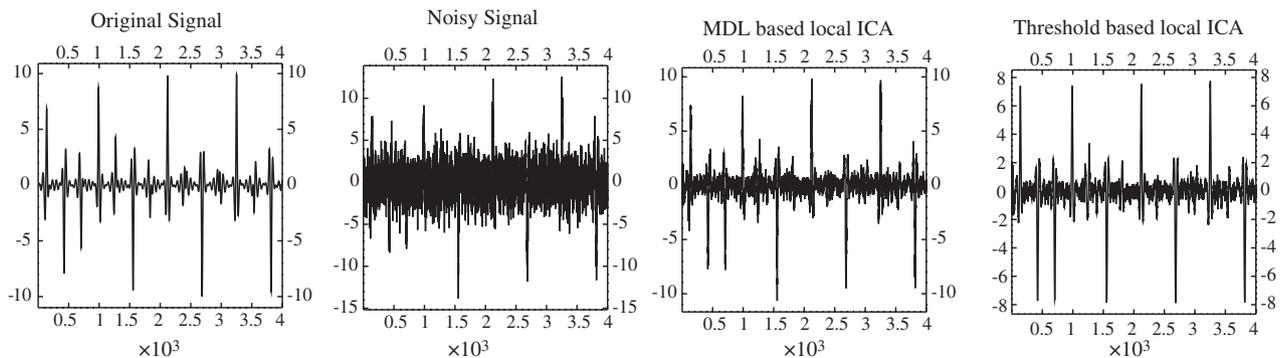
Fig. 1. Comparison between MDL and threshold denoising of an artificial signal with known SNR = 0. The feature space dimension was $M = 40$ and the number of clusters was $K = 35$. The (MDL achieved an SNR = 8.9 dB and the Threshold criterion an SNR = 10.5 dB).
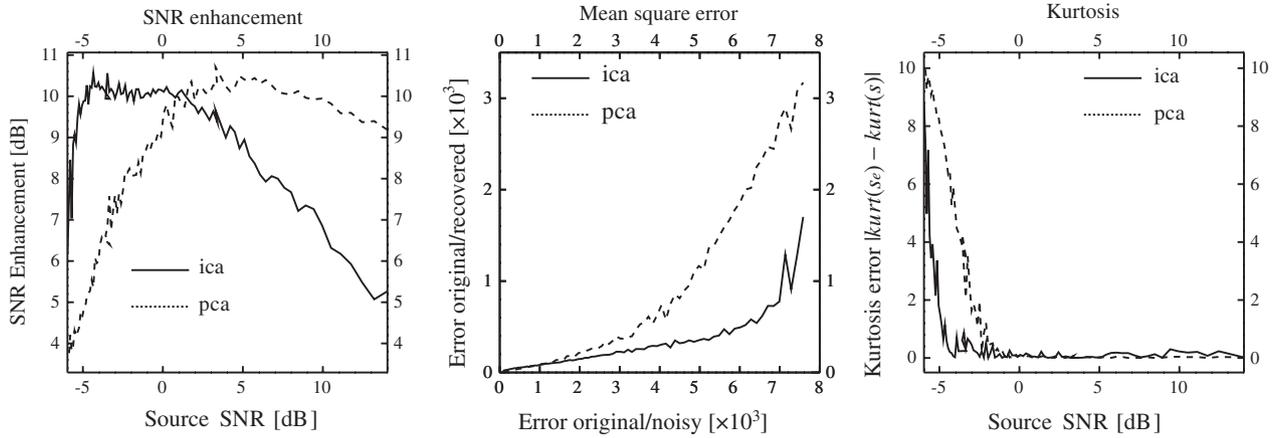
Fig. 2. Comparison between LPCA and LICA based denoising. Here the mean square error of two signals x, y with L samples is $(1/L)\sum_i \|x_i - y_i\|^2$. For all noise levels a complete parameter estimation was done in the sets $\{10, 15, \ldots, 60\}$ for M and $\{20, 30, \ldots, 80\}$ for K.

high SNRs. This might be due to the nature of our selection of subspaces based on kurtosis or variance of the autocorrelation as the comparison of higher statistical moments of the restored data, like kurtosis, indicate that noise reduction can be enhanced if we are using a LICA approach.

4.1.3. LICA denoising with multi-dimensional data sets

A generalization of the LICA algorithm to multi-dimensional data sets like images where pixel intensities depend on two coordinates is desirable. A simple generalization would be to look at delayed coordinates of vectors instead of scalars. However, this appears impractical due to the prohibitive computational effort. More importantly, this direct approach reduces the number of available samples significantly. This leads to far less accurate estimators of important aspects like the MDL estimation of the dimension of the signal subspace or the estimation of the kurtosis criterion in the LICA case.

Another approach could be to convert the data to a 1D string by choosing some path through the data and concatenating the pixel intensities accordingly. But this can easily create unwanted artifacts along the chosen path. Further, local correlations are broken up, hence not all the available information is used.

But a more sophisticated and, depending on the nature of the signal, very effective alternative approach can be envisaged. Instead of converting the multi-dimensional data into 1D data strings prior to applying the LICA algorithm, we can use a modified delay transformation using shifts along all available dimensions. This concept is similar to the multi-dimensional auto-covariances used in the multi-dimensional SOBI (mdSOBI) algorithm introduced in [31].

In the 2D case, for example, consider an $n \times n$ image represented by a matrix $\boldsymbol{P} = (a_{ij})_{i,j=1\ldots n}$. Then the transformed data set consists of copies of $\boldsymbol{P}$ which are shifted either along columns or rows or both. For instance, a translation $a_{ij} \rightarrow a_{i-1,j+1}$, $(i, j = 1, \ldots, n)$ yields the following transformed image:

$$\boldsymbol{P}_{-1,1} = \begin{bmatrix} a_{n,2} & \ldots & a_{n,n} & a_{n,1} \\ a_{1,2} & \ldots & a_{1,n} & a_{1,1} \\ \vdots & & \vdots & \vdots \\ a_{n-1,2} & \ldots & a_{n-1,n} & a_{n-1,1} \end{bmatrix}. \tag{21}$$

Then instead of choosing a single delay dimension, we choose a delay radius M and use all $\boldsymbol{P}_v$ with $\|v\| < M$ as *delayed* versions of the original signal. The remainder of the *LICA* based denoising algorithm works exactly as in the case of a 1D time series.

In Fig. 3 we show that this approach using the MDL criterion to select the number of components compared between LPCA and LICA. In addition we see that the algorithm also works favorable if applied multiple times.

4.2. Denoising with dAMUSE applied to toy examples

A group of three artificial source signals with different frequency contents was chosen: one member of the group represents a narrow-band signal, a sinusoid; the second signal encompasses a wide frequency range; and the last one represents a sawtooth wave whose spectral density is concentrated in the low frequency band (see Fig. 4).

The simulations were designed to illustrate the method and to study the influence of the threshold parameter TH on the performance when noise is added at different levels. In what concerns noise we also try to find out if there is any advantage of using a GEVD instead of a PCA analysis. Hence the signals at the output of the first step of the algorithm (using the matrix $\boldsymbol{Q}$ to project the data) are also compared with the output signals. Results are collected in Table 1.

Random noise was added to the sensor signals yielding a SNR in the range of $[0, 20]$ dB. The parameters $M = 4$ and
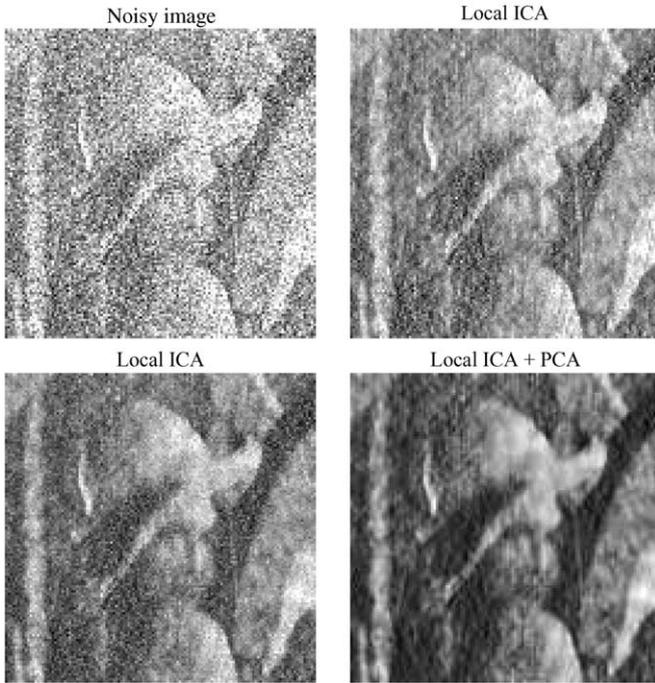
Fig. 3. Comparison of LPCA and LICA based denoising upon an image infested with Gaussian noise. Also note an improvement in denoising power if both are applied consecutively (Local PCA SNR = 8.8 dB, LICA SNR = 10.6 dB, LPCA and LICA consecutively SNR = 12.6 dB). All images where denoised using a fixed number of clusters $K = 20$ and a delay radius of $M = 4$, which results in a 49-dimensional feature space.

$TH = 0.95$ were kept fixed. As the noise level increases, the number of significant eigenvalues also increases. Hence at the output of the first step more signals need to be considered. Thus as the noise energy increases, the number (l) of signals or the dimension of matrix C also increases after the application of the first step (last column of Table 1). As the noise increases, an increasing number of ICs will be available at the output of the two steps. Computing, in the frequency domain, the correlation coefficients between the output signals of each step of the algorithm and noise or source signals we confirm that some are related with the sources and others with noise. Table 1 (columns 3–6) shows that the maximal correlation coefficients are distributed between noise and source signals to a

Table 1
Number of output signals correlated with noise or source signals after step 1 and step 2 of the algorithm dAMUSE

SNR (dB)	NM	1st step		2nd step		Total
		Sources	Noise	Sources	Noise	
20	12	6	0	6	0	6
15	12	5	2	6	1	7
10	12	6	2	7	1	8
5	12	6	3	7	2	9
0	12	7	4	8	3	11
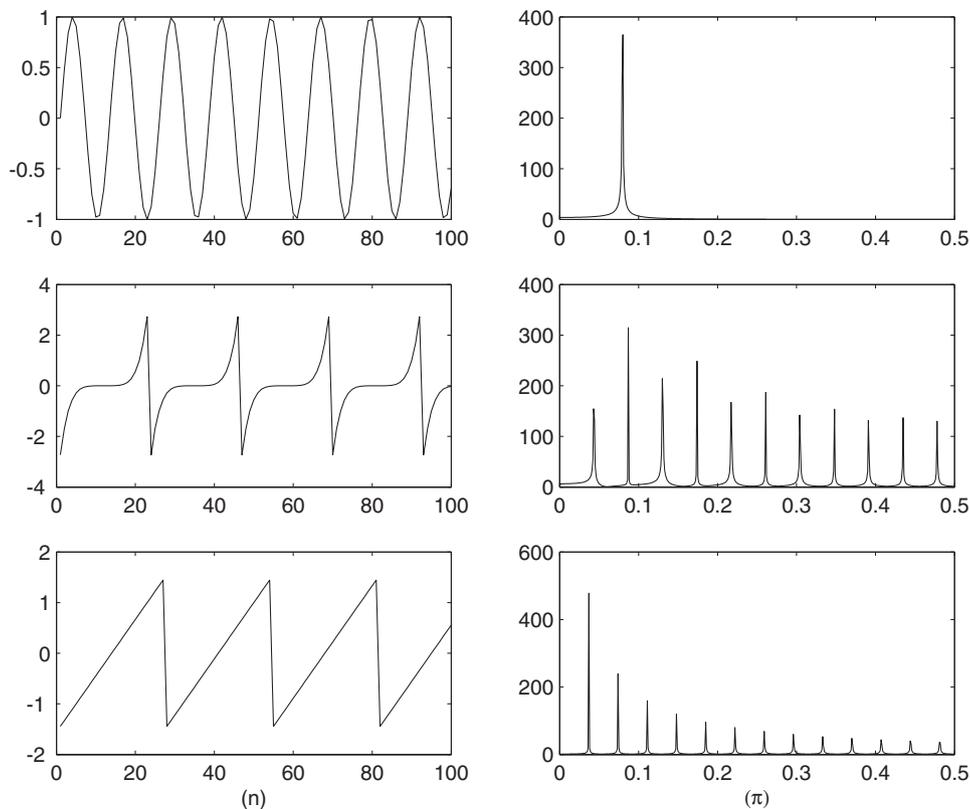
Fig. 4. Artificial signals (left column) and their frequency contents (right column).
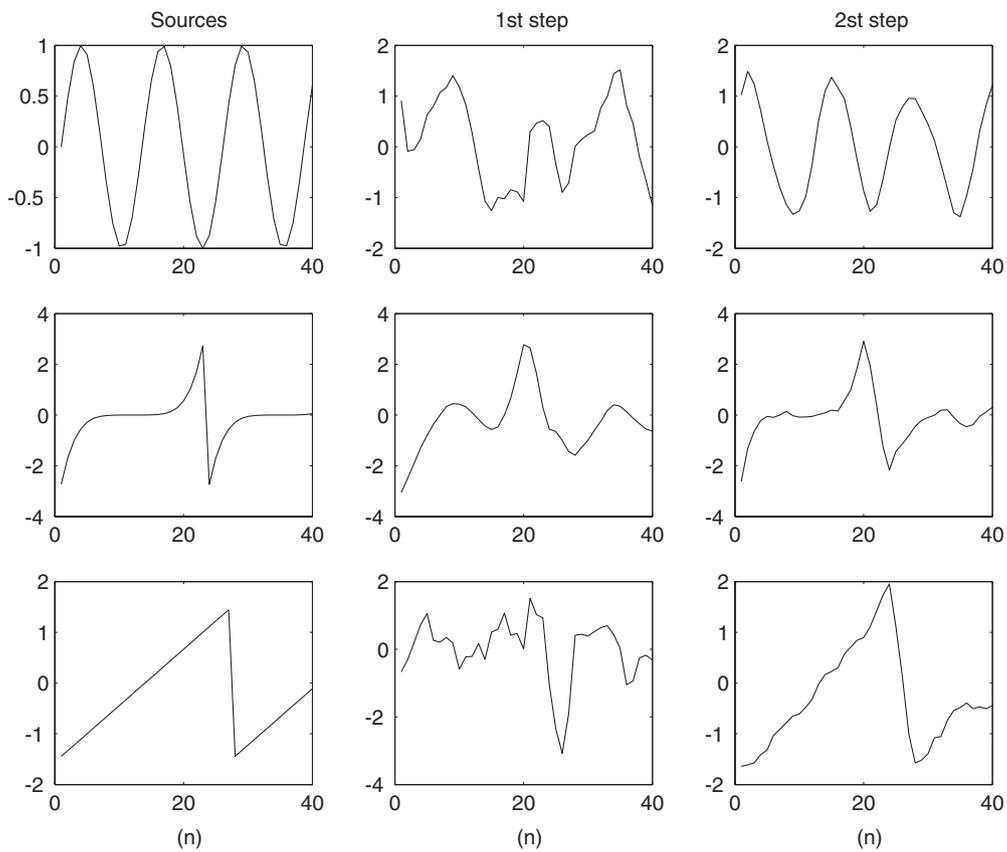
Fig. 5. Comparison of output signals resulting after the first step (second column) and the second step (last column) of dAMUSE.
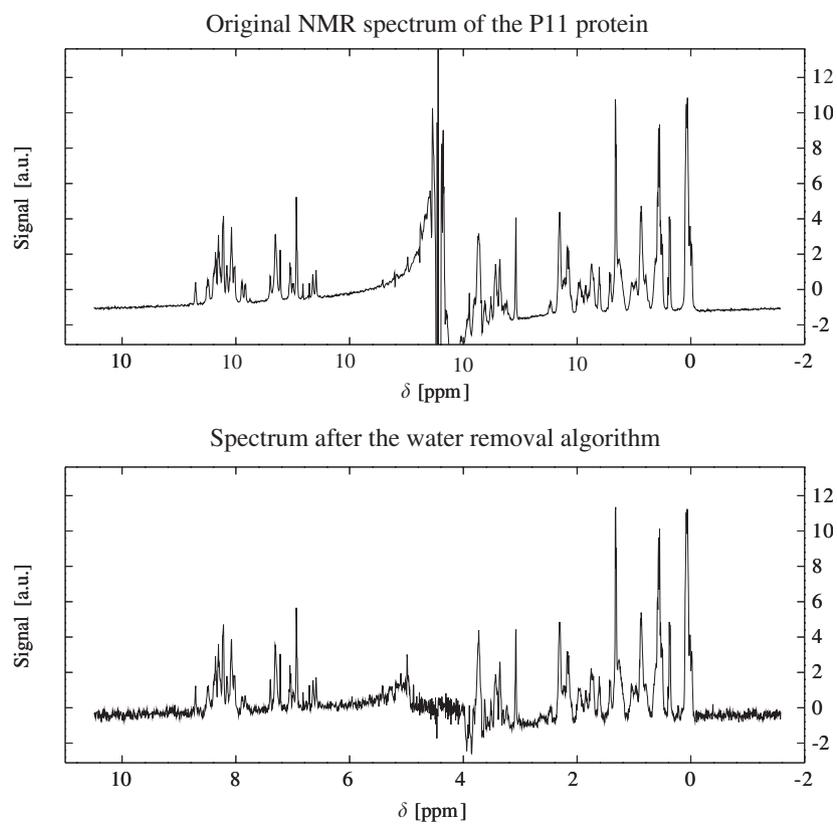
Fig. 6. The graph shows a 1D slice of a proton 2D NOESY NMR spectrum of the polypeptide P11 before and after removing the water artifact with the GEVD-MP algorithm. The 1D spectrum corresponds to the shortest evolution period t_1.
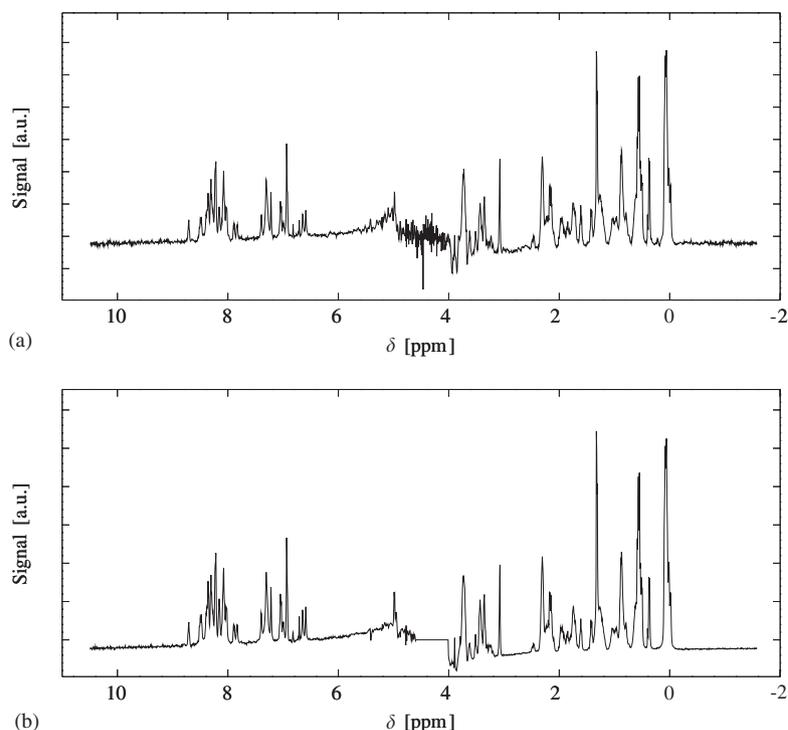
Fig. 7. The figure shows the corresponding artifact free P11 spectra after the denoising algorithms have been applied. The LICA algorithm was applied to all water components with M, K chosen with the MDL estimator ($\gamma = 32$) between 20 and 60 and 20 and 80, respectively. The second graph shows the denoised spectrum with a KPCA based algorithm using a Gaussian kernel: (a) LICA denoised spectrum of P11 after the water artifact has been removed with the algorithm GEVD-MP; (b) KPCA denoised spectrum of P11 after the water artifact has been removed with the algorithm GEVD-MP.

varying degree. We can see that the number of signals correlated with noise is always higher in the first level. Results show that for low noise levels the first step (which is mainly a PCA in a space of dimension NM) achieves good solutions already. However, we can also see (for narrow-band signals and/or M low) that the time domain characteristics of the signals resemble the original source signals only after a GEVD, i.e. at the output of the second step rather than with a PCA, i.e. at the output of first step. Fig. 5 shows examples of signals that have been obtained in the two steps of the algorithm for SNR $= 10$ dB. At the output of the first level the three signals with highest frequency correlation were chosen among the eight output signals. Using a similar criterion to choose three signals at the output of the 2nd step (last column of Fig. 5), we can see that their time course is more similar to the source signals than after the first step (middle column of Fig. 5).

4.3. Denoising of protein NMR spectra

In biophysics the determination of the 3D structure of biomolecules like proteins is of utmost importance. NMR techniques provide indispensable tools to reach this goal. As hydrogen nuclei are the most abundant and most sensitive nuclei in proteins, proton NMR spectra of proteins dissolved in water are recorded mostly. Since the concentration of the solvent is by magnitudes larger

than the protein concentration, there is always a large proton signal of the water solvent contaminating the protein spectrum. This *water artifact* cannot be suppressed completely with technical means, hence it would be interesting to remove it during the analysis of the spectra.

BSS techniques have been shown to solve this separation problem [27,28]. BSS algorithms are based on an ICA [2] which extracts a set of underlying independent source signals out of a set of measured signals without knowing how the mixing process is carried out. We have used an algebraic algorithm [32,33] based on second order statistics using the time structure of the signals to separate this and related artifacts from the remaining protein spectrum. Unfortunately due to the statistical nature of the algorithm unwanted noise is introduced into the reconstructed spectrum as can be seen in Fig. 6. The water artifact removal is effected by a decomposition of a series of NMR spectra into their uncorrelated spectral components applying a generalized eigendecomposition of a congruent matrix pencil [37]. The latter is formed with a correlation matrix of the signals and a correlation matrix with delayed or filtered signals [34]. Then we can detect and remove the components which contain only a signal generated by the water and reconstruct the remaining protein spectrum from its ICs. But the latter now contains additional noise introduced by the statistical analysis procedure, hence denoising deemed necessary.
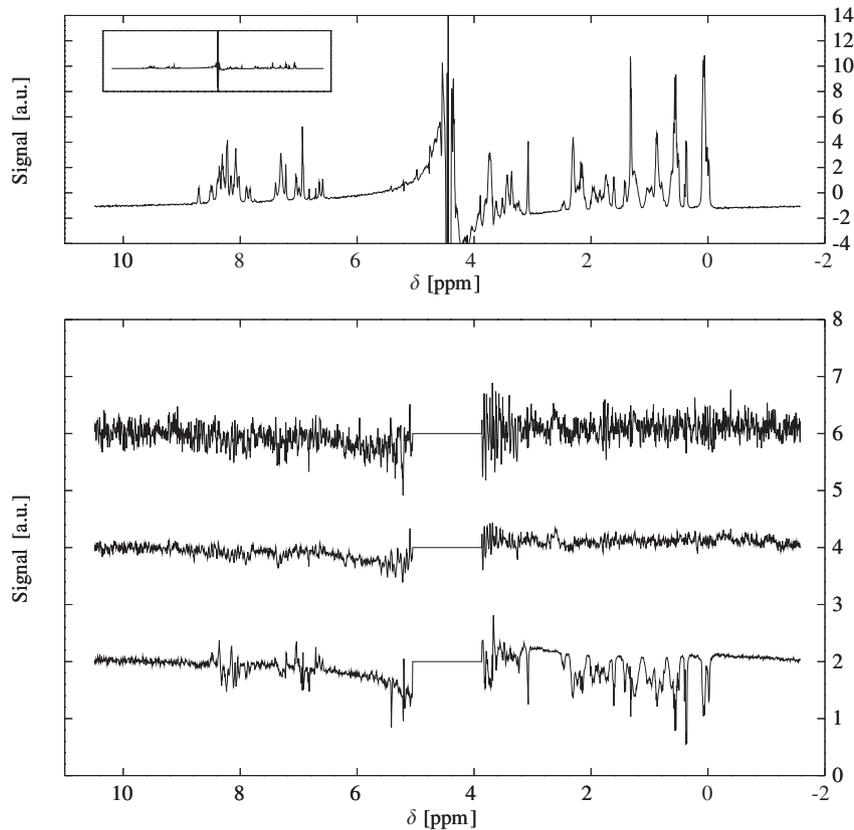
Fig. 8. The graph uncovers the differences of the LICA and KPCA denoising algorithms. As a reference the corresponding 1D slice of the original P11 spectrum is displayed on top. From top to bottom the three curves show: The difference of the original and the spectrum with the GEVD-MP algorithm applied, the difference between the original and the LICA denoised spectrum and the difference between the original and the KPCA denoised spectrum. To compare the graphs in one diagram the three graphs are translated vertically by 2, 4 and 6, respectively.

The algorithms discussed above have been applied to an experimental 2D nuclear overhauser effect spectroscopy (NOESY) proton NMR spectrum of the polypeptide P11 dissolved in water. The synthetic peptide P11 consists of 24 amino acids only and represents the helix H11 of the human glutathion reductase [21]. A simple pre-saturation of the water resonance was applied to prevent saturation of the dynamic range of the analog digital converter (ADC). Every data set comprises 512 free induction decays (FIDs) $S(t_1, t_2) \equiv x_n[l]$ or their corresponding spectra $\hat{S}(1_1, \omega_2) \equiv \hat{x}_n[l]$, with $L = 2048$ samples each, which correspond to $N = 128$ evolution periods $t_1 \equiv [n]$. To each evolution period belong four FIDs with different phase modulations, hence only FIDs with equal phase modulations have been considered for analysis. A BSS analysis, using both the algorithm GEVD using matrix pencil (GEVD-MP) [28] and the algorithm dAMUSE [36], was applied to all data sets. Note that the matrix pencil within GEVD-MP was conveniently computed in the frequency domain, while in the algorithm dAMUSE in spite of the filtering operation being performed in the frequency domain, the matrix pencil was computed in the time domain. The GEVD is performed in dAMUSE as described above to achieve a dimension reduction and concomitant denoising.

4.3.1. Local ICA denoising

For denoising we first used the LICA denoising algorithm proposed above to enhance the reconstructed protein signal without the water artifact. We applied the denoising only to those components which were identified as water components. Then we removed the denoised versions of these water artifact components from the total spectrum. As a result, the additional noise is at least halved as can also be seen from Fig. 7. On the part of the spectrum away from the center, i.e. not containing any water artifacts, we could estimate the increase of the SNR with the original spectrum as reference. We calculated a SNR of 17.3 dB of the noisy spectrum and a SNR of 21.6 dB with applying the denoising algorithm.

We compare the result, i.e. the reconstructed artifact-free protein spectrum of our denoising algorithm to the result of a KPCA based denoising algorithm using a Gaussian kernel in Fig. 8. The figure depicts the differences between the denoised spectra and the original spectrum in the regions where the water signal is not very dominating. As can be seen, the LICA denoising algorithm reduces the noise but does not change the content of the signal, whereas the KPCA algorithm seems to influence the peak amplitudes of the protein resonances as well. Further
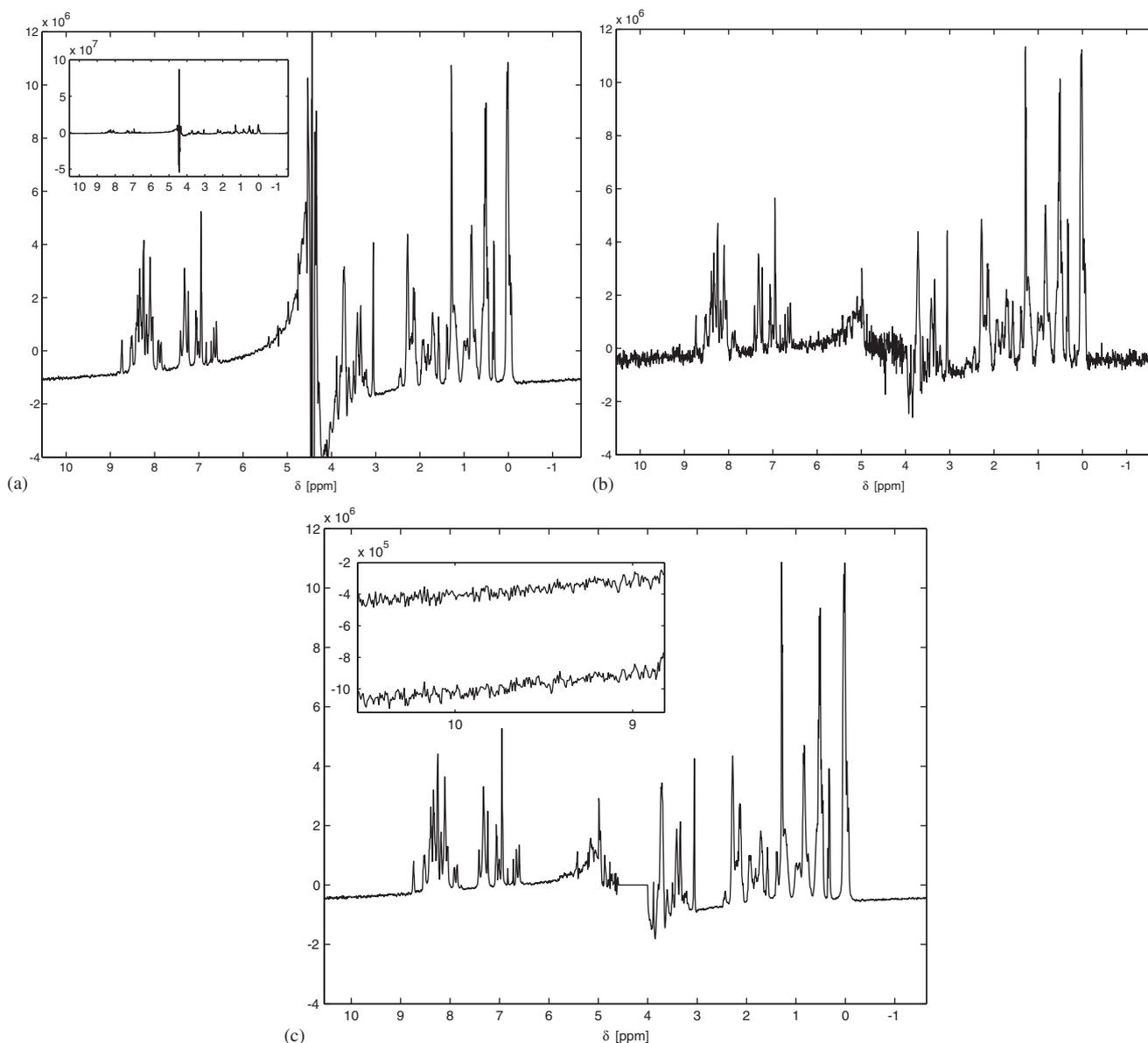
Fig. 9. 1D slice of a 2D NOESY spectrum of the polypeptide P11 in aqueous solution corresponding to the shortest evolution period t_1. The chemical shift ranges from −1 to 10 ppm roughly. The insert shows the region of the spectrum between 10 and 9 ppm roughly. The upper trace corresponds to the denoised baseline and the lower trace shows the baseline of the original spectrum: (a) Original (noisy) spectrum; (b) Reconstructed spectrum with the water artifact removed with the matrix pencil algorithm; (c) Result of the KPCA denoising of the reconstructed spectrum.

experiments are under way in our laboratory to investigate these differences in more detail and to establish an automatic artifact removal algorithm for multi-dimensional NMR spectra.

4.3.2. Kernel PCA denoising

As the removal of the water artifact lead to additional noise in the spectra (compare Fig. 9(a) and (b)) KPCA based denoising was applied. First (almost) noise free samples had to be created in order to determine the principle axes in feature space. For that purpose, the first 400 data points of the real and the imaginary part of each of the 512 *original* spectra were used to form a 400 × 1024 sample matrix $X^{(1)}$. Likewise five further sample matrices

$X^{(m)}$, $m = 2, \ldots, 6$, were created, which now consisted of the data points 401–800, 601–1000, 1101–1500, 1249–1648 and 1649–2048, respectively. Note that the region (1000–1101) of data points comprising the main part of the water resonance was nulled deliberately as it is of no use for the KPCA. For each of the sample matrices $X^{(m)}$ the corresponding kernel matrix K was determined by

$$K_{i,j} = k(x_i, x_j), \quad i, j = 1, \ldots, 400, \tag{22}$$

where x_i denotes the ith column of $X^{(m)}$. For the kernel function a Gaussian kernel

$$k(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right), \tag{23}$$
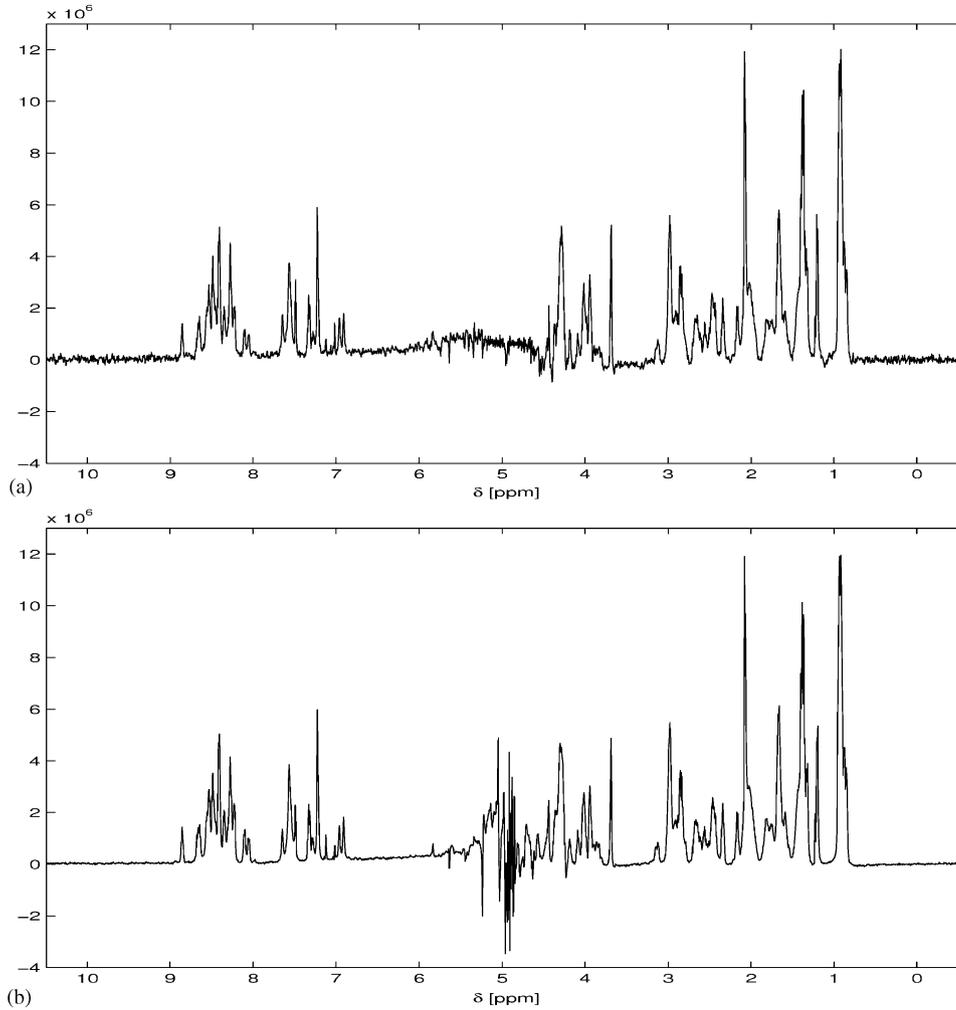
Fig. 10. Comparison of denoising of the P11 protein spectrum: (a) 1D slice of the NOESY spectrum of the protein P11 spectrum reconstructed with the algorithm GEVD-MP; (b) Corresponding protein spectrum reconstructed with the algorithm dAMUSE.

where

$$2\sigma^2 = \frac{1}{400 \times 399} \sum_{i,j=1}^{400} \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 \qquad (24)$$

is the width parameter σ, was chosen.

Finally the kernel matrix $\boldsymbol{K}$ was expressed in terms of its EVD (Eq. (17)) which lead to the expansion parameters $\boldsymbol{\alpha}$ necessary to determine the principal axes of the corresponding feature space $\Omega^{(m)}$:

$$\boldsymbol{\omega} = \sum_{i=1}^{400} \alpha_i \Phi(\boldsymbol{x}_i). \qquad (25)$$

Similar to the original data, the noisy data of the reconstructed spectra were used to form six 400×1024 dimensional pattern matrices $\boldsymbol{P}^{(m)}$, $m = 1, \ldots, 6$. Then the principal components β_k of each column of $\boldsymbol{P}^m$ were calculated in the corresponding feature space $\Omega^{(m)}$. In order to denoise the patterns only projections onto the first $n =$

112 principal axes were considered. This lead to

$$\beta_k = \sum_{i=1}^{400} \alpha_i^k k(\boldsymbol{x}_i, \boldsymbol{x}), \quad k = 1, \ldots, 112, \qquad (26)$$

where $\boldsymbol{x}$ is a column of $\boldsymbol{P}^m$.

After reconstructing the image $\hat{P}_n \Phi(\boldsymbol{x})$ of the sample vector under the map Φ (Eq. (18)), its approximate pre-image was determined by minimizing the cost function

$$\rho(\boldsymbol{z}) = -2 \sum_{k=1}^{112} \beta_k \sum_{i=1}^{400} \alpha_i^k k(\boldsymbol{x}_i, \boldsymbol{z}). \qquad (27)$$

Note that the method described above fails to denoise the region where the water resonance appears (data points 1001–1101) because then the samples formed from the original data differ too much from the noisy data. This is not a major drawback as protein peaks totally hidden under the water artifact cannot be uncovered by the presented BSS method. Fig. 9(c) shows the resulting denoised protein spectrum on an identical vertical scale

Table 2
Parameter values for the embedding dimension of the feature space of dAMUSE (M_{dAMUSE}), the number (K) of sampling intervals used per delay in the trajectory matrix, the number N_{pc} of principal components retained after the first step of the GEVD and the half-width (σ) of the Gaussian filter used in the algorithms GEVD-MP and dAMUSE

Parameter	N_{IC}	M_{dAMUSE}	N_{pc}	N_w(GEVD)
P11	256	3	148	49
Parameter	N_w(dAMUSE)	σ	$SNR_{GEVD-MP}$	SNR_{dAMUSE}
P11	46	0.3	18,6 dB	22,9 dB

as Figs. 9(a) and (b). The insert compares the noise in a region of the spectrum between 10 and 9 ppm roughly where no protein peaks are found. The upper trace shows the baseline of the denoised reconstructed protein spectrum and the lower trace the corresponding baseline of the original experimental spectrum before the water artifact has been separated out.

4.3.3. Denoising using delayed AMUSE

LICA denoising of reconstructed protein spectra necessitate the solution of the BSS problem beforehand using any ICA algorithm. A much more elegant solution is provided by the recently proposed algorithm dAMUSE, which achieves BSS and denoising simultaneously. To test the performance of the algorithm, it was also applied to the 2D NOESY NMR spectra of the polypeptide P11.

A 1D slice of the 2D NOESY spectrum of P11 corresponding to the shortest evolution period t_1 is presented in Fig. 9(a) which shows a huge water artifact despite some pre-saturation on the water resonance. Fig. 10 shows the *reconstructed* spectra obtained with the algorithms GEVD-MP and dAMUSE, respectively. The algorithm GEVD-MP yielded almost artifact-free spectra but with clear changes in the peak intensities in some areas of the spectra. On the contrary, the reconstructed spectra obtained with the algorithm dAMUSE still contain some remnants of the water artifact but the protein peak intensities remained unchanged and all baseline distortions have been cured. All parameters of the algorithms are collected in Table 2.

5. Conclusions

We proposed two new denoising techniques and also considered KPCA denoising which are all based on the concept of embedding signals in delayed coordinates. We presented a detailed discussion of their properties and also discussed results obtained applying them to illustrative toy examples. Furthermore we compared all three algorithms by applying them to the real world problem of removing the water artifact from NMR spectra and denoising the resulting reconstructed spectra. Although all three algorithms achieved good results concerning the final SNR, in case of the NMR spectra it turned out that KPCA seems to

alter the spectral shapes while LICA and dAMUSE do not. At least with protein NMR spectra it is crucial that denoising algorithms do not alter integrated peak intensities in the spectra as the latter form the bases for the structure elucidation process.

In future we have to further investigate the dependence of the proposed algorithms on the situation at hand. Thereby it will be crucial to identify data models for which each one of the proposed denoising techniques works best and to find good measures of how such models suit the given data.

References

[1] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, E. Moulines, A blind source separation technique using second-order statistics, IEEE Trans. Signal Process. 45 (2) (1997) 434–444.
[2] A. Cichocki, S.-I. Amari, Adaptive Blind Signal and Image Processing, Wiley, New York, 2002.
[3] P. Comon, Independent component analysis—a new concept?, Signal Process. 36 (1994) 287–314.
[4] K.I. Diamantaras, S.Y. Kung, Principal Component Neural Networks, Theory and Applications, Wiley, New York, 1996.
[5] A. Effern, K. Lehnertz, T. Schreiber, T. Grunwald, P. David, C.E. Elger, Nonlinear denoising of transient signals with application to event-related potentials, Physica D 140 (2000) 257–266.
[6] E. Fishler, H. Messer, On the use of order statistics for improved detection of signals by the MDL criterion, IEEE Trans. Signal Process. 48 (2000) 2242–2247.
[7] R. Freeman, Spin Choreography, Spektrum Academic Publishers, Oxford, 1997.
[8] R.R. Gharieb, A. Cichocki, Second-order statistics based blind source separation using a bank of subband filters, Digital Signal Process. 13 (2003) 252–274.
[9] M. Ghil, M.R. Allen, M.D. Dettinger, K. Ide, Advanced spectral methods for climatic time series, Rev. Geophys. 40 (1) (2002) 1–41.
[10] K.H. Hausser, H.-R. Kalbitzer, NMR in Medicine and Biology, Berlin, 1991.
[11] J. Hérault, C. Jutten, Space or time adaptive signal processing by neural network models, in: J.S. Denker (Ed.), Neural Networks for Computing, Proceedings of the AIP Conference, American Institute of Physics, New York, 1986, pp. 206–211.
[12] A. Hyvärinen, P. Hoyer, E. Oja, Intelligent Signal Processing, IEEE Press, NY, 2001.
[13] A. Hyvärinen, J. Karhunen, E. Oja, Independent Component Analysis, 2001.
[14] A. Hyvärinen, E. Oja, A fast fixed-point algorithm for independent component analysis, Neural Comput. 9 (1997) 1483–1492.
[15] A.K. Jain, R.C. Dubes, Algorithms for Clustering Data, Prentice-Hall, New Jersey, 1988.
[16] J.T. Kwok, I.W. Tsang, The pre-image problem in kernel methods, in: Proceedings of the International Conference on Machine Learning (ICML03), 2003.
[17] A.P. Liavas, P.A. Regalia, On the behavior of information theoretic criteria for model order selection, IEEE Trans. Signal Process. 49 (2001) 1689–1695.

[18] C.T. Ma, Z. Ding, S.F. Yau, A two-stage algorithm for MIMO blind deconvolution of nonstationary colored noise, IEEE Trans. Signal Process. 48 (2000) 1187–1192.

[19] S. Mika, B. Schölkopf, A. Smola, K. Müller, M. Scholz, G. Rätsch, Kernel PCA and denoising in feature spaces, Adv. Neural Inf. Process. Syst. NIPS11 11 (1999).

[20] V. Moskvina, K.M. Schmidt, Approximate projectors in singular spectrum analysis, SIAM J. Mat. Anal. Appl. 24 (4) (2003) 932–942.

[21] A. Nordhoff, Ch. Tziatzios, J.A.V. Broek, M. Schott, H.-R. Kalbitzer, K. Becker, D. Schubert, R.H. Schirme, Denaturation and reactivation of dimeric human glutathione reductase, Eur. J. Biochem. (1997) 273–282.

[22] L. Parra, P. Sajda, Blind source separation vis generalized eigenvalue decomposition, J. Mach. Learn. Res. 4 (2003) 1261–1269.

[23] K. Pearson, On lines and planes of closest fit to systems of points in space, Philos. Mag. 2 (1901) 559–572.

[24] I.W. Sandberg, L. Xu, Uniform approximation of multidimensional myoptic maps, Trans. Circuits Syst. 44 (1997) 477–485.

[25] B. Schoelkopf, A. Smola, K.-R. Mueller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Comput. 10 (1998) 1299–1319.

[26] K. Stadlthanner, E.W. Lang, A.M. Tomé, A.R. Teixeira, C.G. Puntonet, Kernel-PCA denoising of artifact-free protein NMR spectra, in: Proceedings of the IJCNN'2004, Budapest, Hungaria, 2004.

[27] K. Stadlthanner, F.J. Theis, E.W. Lang, A.M. Tomé, W. Gronwald, H.-R. Kalbitzer, A matrix pencil approach to the blind source separation of artifacts in 2D NMR spectra, Neural Inf. Process.—Lett. Rev. 1 (2003) 103–110.

[28] K. Stadlthanner, A.M. Tomé, F. Theis, E.W. Lang, W. Gronwald, H.R. Kalbitzer, Separation of water artifacts in 2D NOESY protein spectra using congruent matrix pencils, Neurocomputing, 69 (2006) 497–522.

[29] K. Stadlthanner, A.M. Tomé, F.J. Theis, W. Gronwald, H.-R. Kalbitzer, E.W. Lang, Blind source separation of water artifacts in NMR spectra using a matrix pencil, in: Proceedings of the Fourth International Symposium on Independent Component Analysis and Blind Source Separation, ICA'2003, Nara, Japan, 2003, pp. 167–172.

[30] F. Takens, On the numerical determination of the dimension of an attractor, Dynamical Systems Turbulence Ann. Notes Math. 898 (1981) 366–381.

[31] F.J. Theis, A. Meyer-Bäse, E.W. Lang, Second-order blind source separation based on multi-dimensional autocovariances, in: Proceedings of the ICA 2004, Lecture Notes on Computer Science, vol. 3195, Granada, Spain, 2004, pp. 726–733.

[32] A.M. Tome, Blind source separation using a matrix pencil, in: Proceedings of the International Joint Conference on Neural Networks, IJCNN'2000, Como, Italy, 2000.

[33] A.M. Tomé, An iterative eigendecomposition approach to blind source separation, in: Proceedings of the Third International Conference on Independent Component Analysis and Signal Separation, ICA'2003, San Diego, USA, 2001, pp. 424–428.

[34] A.M. Tomé, N. Ferreira, On-line source separation of temporally correlated signals, in: Proceedings of the European Signal Processing Conference, EUSIPCO2002, Toulouse, France, 2002.

[35] A.M. Tomé, A.R. Teixeira, E.W. Lang, K. Stadlthanner, A.P. Rocha, Blind source separation using time delayed signals, in: Proceedings of the International Joint Conference on Neural Networks, IJCNN'2004, vol. CD, Budapest, Hungary, 2004.

[36] A.M. Tomé, A.R. Teixeira, E.W. Lang, K. Stadlthanner, A.P. Rocha, R. ALmeida, dAMUSE—A new tool for denoising and BSS. Digital Signal Processing, 2005.

[37] L. Tong, R. wen Liu, V.C. Soon, Y.-F. Huang, Indeterminacy and identifiability of blind identification, IEEE Trans. Circuits Syst. 38 (5) (1991) 499–509.

[38] R. Vetter, Extraction of efficient and characteristic features of multidimensional time series, Ph.D. Thesis, EPFL, Lausanne, 1999.

[39] R. Vetter, J.M. Vesin, P. Celka, P. Renevey, J. Krauss, Automatic nonlinear noise reduction using local principal component analysis and MDL parameter selection, in: Proceedings of the IASTED International Conference on Signal Processing Pattern Recognition and Applications (SPPRA 02) Crete, 2002, pp. 290–294.

[40] P. Vitányi, M. Li, Minimum description length induction, bayesianism, and kolmogorov complexity, IEEE Trans. Inf. Theory 46 (2000) 446–464.

Peter Gruber was born in Bad Homburg, Germany, on April 12, 1976. He obtained a degree in Mathematics in 2002 at the University of Regensburg. He is currently working on his Ph.D. thesis at the Biophysics Department of the University of Regensburg. His research topics include statistical signal processing, linear and nonlinear independent component analysis and geometric measure theory.

Kurt Stadlthanner received his Diploma degree in Physics from the University of Regensburg in 2003. He is currently a doctoral student at the Institute of Biophysics, Neuro- and Bioinformatics group, at the University of Regensburg. His scientific interests are in the fields of biological data processing and analysis by means of blind source separation and support vector machines.

M. Bóhm received his physics diploma from the University of Regensburg in 2004. He is currently a doctoral student at the Institute of Biophysics, Neuro- and Bioinformatics Group, University of Regensburg. His scientific interests are in the fields of blind source separation, bio-inspired optimization and brain modelling.

Fabian J. Theis obtained M.Sc. degrees in Mathematics and Physics at the University of Regensburg in 2000. He also received a Ph.D. degree in Physics from the same university in 2002 and a Ph.D. in Computer Science from the University of Granada in 2003. He worked as visiting researcher at the department of Architecture and Computer Technology (University of Granada, Spain), at the RIKEN Brain Science Institute (Wako, Japan) and at FAMU-FSU (Florida State University, USA). Currently, he is heading the 'signal processing & information theory' group at the Institute of Biophysics, University of Regensburg, Germany. His research interests include statistical signal processing, linear and nonlinear independent component analysis, over-complete blind source separation and biomedical data analysis.

Elmar W. Lang received his physics diploma with excellent grade in 1977 and his Ph.D. in physics (summa cum laude) in 1980 and habilitated in Biophysics in 1988 at the University of Regensburg. He is apl. Professor of Biophysics at the University of Regensburg. He is currently associate editor of Neurocomputing and Neural Information Processing—Letters and Reviews. His current research interests focus mainly on machine learning and include biomedical signal processing, independent component analysis and

blind source separation, neural networks for classification and pattern recognition as well as stochastic process limits in queuing applications.

Ana M. Tomé received her Ph.D. in Electrical Engineering from University of Aveiro in 1990. Currently she is Associate Professor of Electrical Engineering in the Department of Electronics and Telecomunications/IEETA of the University of Aveiro where she teaches courses on Digital Signal Processing for Electronics and Computer Engineering Diploms. Her research interests include Digital and Statistical Signal Processing, Independent Component Analysis and Blind Source Separation as well as Classification and Pattern Recognition Applications of Neural Networks.

Ana Rita Teixeira received her diploma degree in mathematics applied to technology from University of Porto in 2003. Currently, she is finishing the M.Sc. of electronics and telecommunications at the University of Aveiro. Her research interests include biomedical digital signal processing as well as principal and independent component analysis.

Carlos G. Puntonet received a B.Sc. degree in 1982, an M.Sc. degree in 1986 and his Ph.D. degree in 1994, all from the University of Granada, Spain. These degrees are in electronics physics. Currently, he is an Associate Professor at the "Departamento de Arquitectura y Tecnología de Computadores" at the University of Granada. His research interests lie in the fields of signal processing, linear and nonlinear independent component analysis and blind separation of sources, artificial neural networks and optimization methods.

J.M. Górriz received the B.Sc. in Physics and Electronic Engineering from the University of Granada, Spain and the Ph.D. from the University of Cádiz, Spain in 2000, 2001, and 2003, respectively. He is currently Assistant Professor at the University of Granada. He is actually developing a Ph.D. in Voice Activity Detection, Robust Speech Recognition and Optimization Strategies. His present interests are in Statistical Signal Processing and its applications to speech.